# Network Coding

*Edited by:* Adele Kuzmiakova

# Network Coding

# Network Coding

*Edited by:*

**Adele Kuzmiakova**

ARCLER PRESS

www.arclerpress.com

## Network Coding

*Adele Kuzmiakova*

**Arcler Press**
224 Shoreacres Road
Burlington, ON L7L 2H2
Canada
www.arclerpress.com
Email: orders@arclereducation.com

**e-book Edition 2022**

ISBN: 978-1-77469-287-5 (e-book)



ISBN: 978-1-77469-106-9 (Hardcover)

Arcler Press publishes wide variety of books and eBooks. For more information about Arcler Press and its products, visit our website at www.arclerpress.com

# ABOUT THE EDITOR

**Adele Kuzmiakova** is a computational engineer focusing on solving problems in machine learning, deep learning, and computer vision. Adele attended Cornell University in New York, United States for her undergraduate studies. She studied engineering with a focus on applied math. While at Cornell, she developed close relationships with professors, which enabled her to get involved in academic research to get hands-on experience with solving computational problems. She was also selected to be Accel Roundtable on Entrepreneurship Education (REE) Fellow at Stanford University and spent 3 months working on entrepreneurship projects to get a taste of entrepreneurship and high-growth ventures in engineering and life sciences. The program culminated in giving a presentation on the startup technology and was judged by Stanford faculty and entrepreneurship experts in Silicon Valley. After graduating from Cornell, Adele worked as a data scientist at Swiss Federal Institute of Technology in Lausanne, Switzerland where she focused on developing algorithms and graphical models to analyze chemical pathways in the atmosphere. Adele also pursued graduate studies at Stanford University in the United States where she entered as a recipient of American Association of University Women International Fellowship. The Fellowship enabled her to focus on tackling important research problems in machine learning and computer vision. Some research problems she worked on at Stanford include detecting air pollution from outdoor public webcam images. Specifically, she modified and set up a variety of pre-trained architectures, such as DehazeNet, VGG, and ResNet, on public webcam images to evaluate their ability to predict air quality based on the degree of haze on pictures. Other deep learning problems Adele worked on include investigating the promise of second-order optimizers in deep learning and using neural networks to predict sequences of data in energy consumption. Adele also places an emphasis on continual education and served as a Student Leader in PyTorch scholarship challenge organized by Udacity. Her roles as the Student Leader were helping students debug their code to train neural networks with PyTorch and providing mentorship on technical and career aspects. Her hobbies include skiing, playing tennis, cooking, and meeting new people.

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF ABBREVIATIONS

| | |
|---|---|
| ACL | Access Control Lists |
| ACO | Ant Colony Optimization |
| ALM | Application Layer Multicast |
| ALMA | Application Layer Multicast Algorithm |
| AODV | Ad-hoc On-Demand Distance Vector |
| AODVM | Ad-hoc On-Demand Distance Vector Multipath |
| AOMDV | Ad-hoc On-Demand Multipath Distance Vector |
| App IF | Application Interface |
| ARP | Address Resolution Protocol |
| ARQ | Automatic Repeat Request |
| BER | Bit Error Rate |
| BS | Base Station |
| CA | Certifying Authority |
| CA | Collision Avoidance |
| CBMRP | Cluster-Based Multipath Routing Protocol |
| CB-PKC | Certificate-Based Public Key Cryptography |
| CC | Continuity Counter |
| CL-PKC | Certificateless Public Key Cryptography |
| CONCERTO | Control Over Network Coding for Enhanced Radio Transport Optimization |
| CPU | Central Processing Unit |
| CS | Compressed Sensing |
| CSMA | Carrier Sense Multiple-Access |
| D | Destination |
| D2D | Device-to-Device |
| DCF | Distributed Coordination Function |
| DDB | Dynamic Delayed Broadcasting |

| | |
|---|---|
| DF | Digital Fountain |
| DNs | Destination Nodes |
| DNS | Domain Name System |
| DSR | Dynamic Source Routing |
| EM | Electromagnetic |
| FAME | Firewall Anomaly Management Environment |
| FEC | Forward Error Correction |
| FMT | Fade Mitigation Techniques |
| GSE | Generic Stream Encapsulation |
| GSM | Global System for Mobile |
| HARQ | Hybrid ARQ |
| I/O | Input and Output |
| IB-PKC | Identity-Based Public Key Cryptography |
| IGMP | Internet Group Management Protocol |
| IM | Instant Messaging |
| KGC | Key Generation Center |
| LNC | Linear Network Coding |
| LP | Linear Program |
| LT | Luby Transform |
| LTE | Long-Term Evolution |
| MAC | Medium Access Control |
| MBMS | Multimedia Broadcast and Multicast Services |
| MDC | Multiple Description Coding |
| MDL | Minimum Description Length |
| MDS | Maximum Distance Separable |
| MPE | Multiple Protocol Encapsulation |
| MRNC | MAC-layer Random Network Coding |
| MRP | Multipath Routing Protocol |
| NACKs | Negative Acknowledgments |
| NC | Network Coding |
| NDMR | Node-Disjoint Multipath Routing |
| OLSR | Optimized Link State Routing |
| ONC | Opportunistic Network Coding |
| P2P | Peer-to-Peer |

| | |
|---|---|
| PEP | Packet Error Probability |
| PER | Packet Error Rate |
| PNC | Physical Network Coding |
| QoS | Quality of Service |
| R3E | Reliable Reactive Routing Enhancement |
| RAM | Read Access Memory |
| RLNC | Random Linear Network Coding |
| RN | Relay Network |
| RNC | Random Network Coding |
| ROVER | Robust Vehicular Routing |
| RREP | Route Reply |
| RREQ | Route Request |
| RRL | Rectangle Rule List |
| RRUs | Remote Radio Units |
| RS | Reed-Solomon |
| RSD | Robust Soliton Distribution |
| S | Source |
| SaaS | Storage as a Service |
| SE | Store and Encode |
| SGC | Subgraph Constructor |
| SHM | Shortest Hop Multipath |
| SIMD | Single Instruction Multiple Data |
| SLNC | Segment Linear Network Coding |
| SMR | Split Multi-Path Routing |
| SN | Source Node |
| TCP | Transmission Control Protocol |
| TNC | Triangular Network Coding |
| TS | Transport Stream |
| TTP | Trusted Third Party |
| TWRC | Two-Way Relay Channel |
| VANETs | Vehicular Ad-Hoc Networks |
| VM | Virtual Machine |
| VoIP | Voice Over Internet Protocol |
| WBA | Wireless Broadcast Advantage |

| | |
|---|---|
| WCDS | Weighted Connected Dominating Set |
| Wi-Fi | Wireless Fidelity |
| WLAN | Wireless Local Area Network |
| WMNs | Wireless Mesh Networks |
| WWW | World Wide Web |
| ZOR | Zone of Relevance |

# PREFACE

Network coding (NC) is a method of improving the flow of digital data in a network via transmitting the digital proof about the messages. The "digital evidence" is a combination of two or more messages. When the bits of the digital evidence reach the destination point, the transmitted message is deduced in spite of directly reassembled. Network coding was first introduced in 1999 as a substitute to routing.

This book takes the readers through a brief introduction into the network coding. This book sheds light on the history of network coding, theoretical framework for network coding, applications, and challenges of network coding, throughput benefits, use of network coding in wireless systems, and security against adversarial errors.

The first chapter stresses the basic overview of network coding so that readers are clear about the philosophies and history behind that form the utmost basics in the field. This chapter will also emphasize the development of network coding across the several years, and also explains some important network security protection models.

The second chapter takes the readers through the concepts network multicast model, along with the various frameworks such as algebraic framework and network coding framework. This chapter will provide highlights on the application of index coding with the help of linear programming as well as various network coding-based multipath routings.

Then, the third chapter explains the implementation of network coding in wireless systems. It also explains the various network coding schemes, wireless mesh networks, and wireless sensory networks. This chapter also sheds light on the theory behind the PNC.

The fourth chapter introduces readers to the use of network coding in mobile applications. This chapter also explains the significance of various transmission approaches such as unicast, multicast, systematic network coding, and pure network coding.

The fifth chapter throws light on the network coding for multicast networks, peer-to-peer (P2P) network coding, and application layer multicast (ALM). This chapter also addresses the difference between network coding and random coding, along with various protocols and algorithms that have been used in network coding.

The sixth chapter takes readers to the benefits of throughput in network coding, the concept of randomized network coding. Readers are then told about the various vector network coding algorithm, and code design algorithms such as code design for vector and scalar coding.

The seventh chapter explains the various applications and challenges in network coding that have been addressed in the last several years. This chapter also emphasize the application of network coding in mobile and wireless sensory systems, cloud, and distributed storage systems. This chapter also explains the various limitations of network coding.

The last chapter of this book sheds lights on the error correction bounds for centralized network coding. This chapter also mentions the detection of adversarial errors, various modeling of relaying strategies, as well as the benefits of randomized coding over routing.

This book has been designed to suit the knowledge and pursuit of the researcher and scholars and to empower them with various aspects of network coding, the development of network coding and its applications in various sectors so that they are updated with the information.

**CHAPTER 1**

# AN INTRODUCTION TO NETWORK CODING

## CONTENTS

The introduction chapter explains a brief history of NC along with the development of NC. This chapter also explains what is NC, which includes throughput, robustness, complexity, and NC security.

This chapter also emphasizes various kinds of NC, such as linear network coding (LNC), encoding, decoding, and linear combinations selection. This chapter addresses the various threats in network security, including passive attack, active attack, distributed attack, insider attack, along with other major security threats. This chapter provides highlights on the various protection models in order to improve the network security.

## 1.1. INTRODUCTION

It is neither easy nor straightforward to define network coding (NC). Over a period of time, numerous definitions have been given. According to Cai, Ahlswede, Yeung, and Li, as brought forth in their seminal paper, NC is used to "refer to coding at a node in a network," which to them means a casual arbitrary mapping from the inputs to the outputs.

Even though this is one of the most general definitions of NC, there is no distinction of network or the informational theory which is multi-terminal and the study of NC. Plenty of information is already available on network information theory, hence the same has not been dwelt upon, and here in this book, endeavor has been made to further this definition.

In general networks, every node is affected in all probability and arbitrarily by another node as far as the information theory paper goes, and this aspect has been distinguished in Ahlswede et al.'s paper where point-to-point links that are error-free interconnect the nodes in a network and these networks have been specifically studies.

Where the network information theory ordinarily studies the network models, Ahlswede et al.'s network model is a special case which is far more relevant to the current networks. Once error-free conduits have been obtained after abstracting the physical layer for carrying bits all wireline networks can easily fall within this category.

This essentially helps to distinguish the function of channel coding and NC for noisy links. Similarly, it becomes possible to distinguish source coding too from NC whereby NC is considered in the context of incompressible source processes that are independent.

The more frequently used definition of NC falls into network information theory's special case category. Even before 2000, this special case has been well studied, so that the novelty factor of NC reduces slightly however this definition can still be used further.

Random linear network coding (RLNC) is the specific form of NC around which most of the work related to NC is concentrated. When introduced, RLNC was shown as a randomized, simple method for coding whereby "a vector of coefficients for each of the source processes," was maintained and being "updated by each coding node."

Messages are communicated through a network in RLNC where they are accompanied by some additional information which is a vector of coefficients in this case. In the currently existing networks for communication packet networks is the network which is used extensively as extra information with links that are error-free can be easily accommodated on it.

These days packet headers are commonly used where packet headers can hold side information or extra information in packets (for instance, to keep track of orders, sequence numbers are usually kept in packet headers).

Hence, coding above the physical layer or more specifically coding at a node in packet network is the third definition with respect to NC (wherein the contents of the packets namely the data which is placed in these packets is applied with NC).

However, in the network information theory, the physical layer generally has the coding. This book uses this specific definition. In certain cases, the scope gets limited due to restriction of attention to packet network, and there may be non-reporting of certain results that have implications that are beyond packet networks. All the same, this definition brings the discussion to a concrete setting which is far more applicable in practice making this definition useful.

In order to reduce the gap that exists between a strategy and its execution, an infrastructure of Information Technology is being formulated by various companies. For the implementation to be successful, efficiency, and flexibility with a heightened need for the incorporation of insight are critical.

Costs can be minimized by the optimal usage of resources to achieve the objectives and goals of a business through efficiency. A timely response to the ever-changing policies of the company, convention, legislations of the government, and other registered areas with which the company's operations come into contact can be encouraged through flexibility.

Similarly, the company's systems shall be able to evaluate the software, evaluate programs and recognize initiatives through insight which shall also aid in the achievement of performance that is enhanced. The lower costs involved in an IT investment is one of the biggest advantages amongst the series of benefits that can be derived through IT, and in the current scenario where most people are working from home, it is pertinent that they are aware of such benefits.

To meet the demands for high mobility and spectral efficiency with a reduced consumption of energy, it is essential to have strong wireless networks in the future. With respect to the network resources like energy and bandwidth, it is essential to be aware of certain factors like the superior performance that can be maintained through the network as also how the wireless network design can be used to efficiently achieve this performance.

In order to overcome that challenges that are emerging new schemes and techniques are required in the current times where wireless network is only growing. For the improvement of network capacity, a new technique has appeared, namely NC.

This helps to increase the network's throughput. The incoming information is allowed to combine with the intermediate node through NC. The function of the intermediate node is restricted when compared to the conventional network.

The wireless network is now applied with NC, unlike earlier where it was used in the wired network during the earlier phases. The ability of the wireless transmission to broadcast is enhanced though the NC which has become a fundamental strategy for these networks.

During the last decade, the specific area of cooperative communications (namely relay network (RN)) within the wireless communication set up has been explored extensively. The spatial goals of diversity are realized where the nodes assist each other in a network architecture upon which the RN idea has been built.

Two end-users were the main points between which messages used to be transmitted traditionally. However, the data packet transmission cannot be completed properly where enough data packets cannot be received by the destination node (DN) owing to a series of factors like natural obstacles, the transmission range, and even the quality of a direct transmission link that could be quite poor.

Hence there is the need to bridge the connection between two nodes which can be done by deploying a node between them. Traditionally, information was repeated and forwarded to the nodes at the end through the relay nodes. the topology of the main network comprising of 3-nodes namely; the source(S), the relay® and the destination (D) has been discussed in and whereby the basic idea of relay channels and the concept per se has been introduced and explained in detail.

The transmission over networks that are wireless is loaded with a series of challenges like system performance, noise, bandwidth, interference, power constraints and fading. NC aids in overcoming these challenges. Whilst transmitting data packages from the source(s) to the receiver(s), the NC method aids in their integrated and accurate flow and at the same time achieve throughput that is max-min-cut.

Where the topology consists of a wireless network with three nodes NC can be implemented in a very straightforward manner as has been described by Katti et al. and Wu et al. Over the network topology that is provided, the implementation of NC helps in successful transmission for broadcast, unicast, and multicast. Can be referred to for details with respect to the advanced as well as fundamental concepts of NC.

## 1.2. HISTORY AND DEVELOPMENT OF COMPUTER NETWORKS

Industrial revolution was accompanied with a series of mechanical systems during the 18$^{th}$ century. Steam engines made their presence felt in the 19$^{th}$ century. Information gathering, its processing, and subsequent distribution was the main technology of the 20$^{th}$ century.

Worldwide telephone networks were installed amongst the many other developments that took place during this era along with the invention of the television and radio, the birth and growth of the computer industry which was unprecedented and the communication satellites being launched.

The difference between collecting, sorting, transporting, and processing of information are disappearing quickly due to the rapid convergence of these areas in a set up where technological progress is taking place at an extremely rapid pace.

At the mere push of a button, organizations with numerous offices spread all over the world at different geographical locations not only expect to but are also able to routinely examine the latest status of the remotest of the outputs.

The demand for more and more sophisticated processing of information is growing at a fast rate due to an increase in our ability to not only gather but also process and distribute information. Young though as it may be as compared to other industries, within a very short time, the computer industry has been able to make remarkable progress.

In the initial phases, the computer systems were highly restricted, being limited to a single room that was quite large. More often than not, visitors could view this most modern of the technologies through the glass walls of these rooms (Figure 1.1).

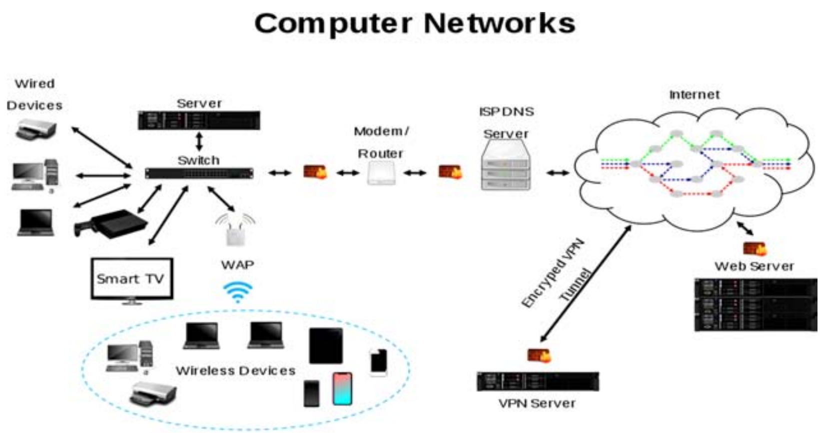



**Figure 1.1:** Computer network anchor.

*Source: Image by Wikimedia commons.*

A couple of dozen such computers were owned by large institutions, whereas the smaller ones had one or two of them. It was then merely the work of science fiction that within two decades, computers smaller than postage stamps and extremely powerful at that would be produced by the millions and not just a few hundreds.

The organization of computer systems has been influenced tremendously by the merger of communications and computers. The ‘computer center’ concept, which is essentially a room used by various individuals to process their work is but obsolete in the current times.

A large number of interconnected computers which though separate together meet the organization’s needs, have replaced the old model where a single room with a single computer met the organizational requirements.

These systems together comprise the computer networks. This book mainly deals with the organization and design of these networks. “Computer technology" is the term used where a single technology is used to interconnect the varied collection of autonomous computers (Figure 1.2).

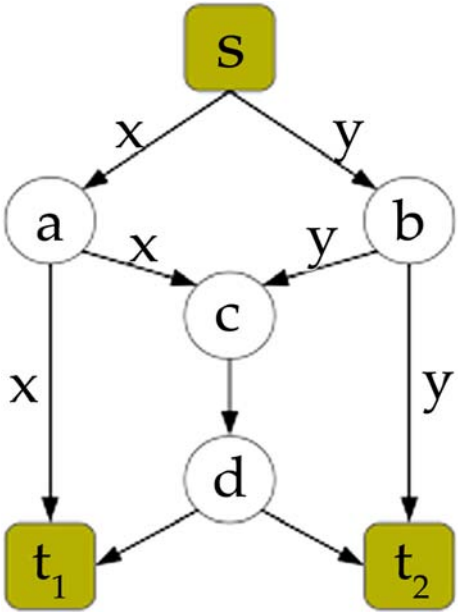



**Figure 1.2:** Networking coding example.

*Source: Image by Wikimedia commons.*

If exchange of information is possible between two computers, they are said to be interconnected. Other than be connected only through a copper wire, they can be connected using microwaves, communication satellites, fiber optics and infrared. Subsequently, we shall see that network can be in varied shapes, sizes, and forms.

Strange as it may seem, the World Wide Web (WWW) and the Internet both, are not computer networks. The reasons for this shall be clear towards the end of this book. In simpler words, a network of networks and not any single network makes the Internet and the web is essentially a distributed system running over the Internet.

In literature, a distributed network and a computer network have been quite confused. The main difference is that users see a collection of computers that are independent as a single system that is coherent in a distributed system. In most cases, the users are presented with a single paradigm or model.

Usually, this model is implemented through a middleware which is basically a software layer over the operating system. WWWis one of the most common examples of a distributed system where everything takes on the appearance of a document, also called the web page.

This kind of a software, model, and coherence are missing in a computer network. No attempt is made by the system whereby machines can be made to act and look coherent so that the actual machines are exposed to the users. A user needs to log into and run a machine where it is present where he/she wants to run a program which is remotely located.

Effectively, a software system that is built over a network is a distributed system. A high level of transparency and cohesiveness is given by the software. So essentially rather than the hardware, it is the software and mainly the operating system that distinguishes a network and a distributed system.

## 1.3. A BRIEF HISTORY OF NETWORK CODING (NC)

- **2000:** The concept was introduced for the first time in the year 2000. It was shown by Ning Cai, Raymond W. Yeung, Rudolf Ahlswede and Shuo-Yen Robert Li that NC has potential power in multicast networks where information that is identical is received by all the receivers. Even though they did not describe a method for designing an informative and good code, they did prove the existence of such a code.
- **2003:** Towards the practical implementation of NC noteworthy steps were taken. It was shown by Cai, li, and Yeung that the complexity involved in the designing of codes could be reduced by relying on mathematical functions that involved only multiplication and addition in the case of multicast networks. Further, for simplifying code design and analyzing various approaches towards coding Mdard and Koetter introduced an algebraic framework that was quite powerful.
- **2005–2006:** This period saw the publication of design algorithms that were valuable. In a multicast network, for designing the functions that are used by each node multicast low-complexity algorithms were published together by Sidharth Jaggi from Caltech, Effros, and Peter Sanders of the University of Karlsruhe in Germany both as collaborators as well as separately and Tracey Ho Caltech with the others.

- A systematic approach towards function designing was given in the first paper, and it was shown that random and independent choosing of functions for each node would work equally well in the second paper (a conference held in 2003 saw the early versions of both these contributions).
- **2006:** This phase saw the exploration for applications that could be used for networks that were wireless. The benefits of NC with regard to wireless applications were demonstrated by Christina Fragouli of the École Polytechnique Fédérale de Lausanne in Switzerland along with her collaborators in a conference in 2006. They also mentioned the characterized situations where the helpfulness of the approach could be high.

## 1.4. WHAT IS NETWORK CODING (NC) GOOD FOR?

Now that a definition is in place, the benefits or uses of NC can be discussed. The security, robustness, throughput, and robustness can all be enhanced through NC. These performance factors have all been discussed here.

### 1.4.1. Throughput

Enhanced throughput is perhaps the easiest to explain and is NC's most well-known utility. More efficient use of packet transmissions results in benefitting throughput whereby a far lesser number of packet transmissions can communicate more information.

Ahlswede et al. considered the multicast problem in a wireline network and have given an example of this benefit which is the most famous one. Butterfly network is used to refer to this example of theirs' where a multicast from a single source to two destinations or sinks is featured. The message at the source code is desired to be known in full at both sinks.

They consider the capacitated network only once one of the intermediate nodes breaks from the packet networks' traditional routing paradigm there can be establishment of the desired multicast connection where for the output alone the intermediate nodes have approval to make copies of the packets that have been received and a coding operation is performed.

By taking the XOR or the binary sum, it forms a new packet upon receiving two packets, it performs the coding operation, and the resulting packet is the output. A bitwise XOR of b1 and b2 is used to form the output packet b1 $\oplus$ b2 where b1 and b2 are the vectors contained in the two packets that have been received. Further coding operations are performed on the received packets by the sinks to decode them.

By taking the XOR of b2 and b1 ⊕ b2 sink t2 recovers b1 just the way b2 is recovered by sink t1 by taking the XOR of b1 and b1 ⊕ b2. Under routing one would be able to communicate further only one b2 or b1 to t2 where for instance b2 and b1 are communicated to t1.

An important point is illustrated through the butterfly network even though it has been created, namely that in a wireline network throughput for multicast can be increased through NC. The contents of two packets are communicated through butterfly network by using nine packet transmissions.

Additional transmissions would be required to supplement the nine transmissions, or else they would not be able to communicate in the absence of coding (for instance, an additional transmission from node 3 to node 4).

The throughput benefits of NC are not limited to wireline or multicast networks even though it helps to enhance throughput for multicast in wireline network. For instance, when the butterfly network is slightly modified two unicast connections are involved, which can be established with coding and cannot be established without it.

Two unicast connections are involved in this instance. So far, unicast has been considered in the lossless wireline networks, and to enable again to throughput as a result of NC, at least two unicast connections are required.

This section establishes in a more concrete manner that no advantage is derived by throughput through NC when routing is done over a single unicast connection in lossless wireline network.

### 1.4.2. Robustness

#### *1.4.2.1. Robustness to Packet Losses*

Various factors, such as buffer overflow, collision, and link outage cause packet loss in networks. These losses can be dealt with through a number of ways. Setting up a system of acknowledgments is one of the most simple and straightforward mechanism used by the transmission control protocol (TCP) wherein the source receives a message that acknowledges the receipt of the packets by the sink and the source retransmits the packet where no such acknowledgment is received as it means that the sink has not received the concerned packet.

Erasure coding sometimes also referred to as channel coding is also used at times as an alternate method. The source node (SN) applies erasure code so that even if the sink receives only a subset of the packets as have been sent by the source, the message can still be recovered due to a certain level of redundancy being introduced to the packets.

At the same time, one needs to consider NC or as to what happens to the coding that has been applied to the intermediate nodes as also whether packet losses can be reduced through NC. The evident answer is that NC is able to reduce such losses, and it does that. This can be understood through a simple illustration.

The illustration shows a two-link simple tandem network. With a probability of ε12 packets can be lost on the link that joins nodes 2 and 1, whereas the probability of such a loss in this network is ε23 on the link that joins nodes 3 and 2. Per unit time, information can be communicated at a rate of (1 – ε12) (1 – ε23) packets when erasure code is applied at node 1.

A code that has been suitably designed can be used to achieve an erasure channel with an erasure probability 1– (1 – ε12) (1 – ε23) existing between nodes 1 and 3 which has a capacity of (1 – ε12) (1–ε23). However, the system has a higher true capacity.

At a rate of 1 – ε12 packets per unit time information can be relayed between nodes 1 and 2 and at a rate of 1 – ε23 packets per unit time between nodes 2 and 3 by using an erasure code over the links joining nodes 2 and 3 and another one over the link joining nodes 1 and 2 in other words by applying two stages of erasure coding with complete decoding and re-encoding at node 2.

Furthermore, at a minimum rate of (1 – ε12, 1 – ε23) information can be communicated between nodes 1 and 3 which is far higher than (1 – ε12) (1 – ε23). However, the main reason for not using this solution in packet networks is the delay caused. Some sort of a delay is involved at every stage of erasure coding then be it the usage of convolutional code or the block code. This is mainly due to the reason that before the decoder can start decoding a certain number of packets need to be received at each stage. The total delay would be quite large if every link of the connection has erasure coding applied to it.

Coding is applied at intermediate nodes in a special kind of network nodding where extra stages of erasure coding are applied. Hence to avoid packet losses robustness can be provided by using NC from which throughput gains can be translated.

The NC solutions need to go further than just applying erasure coding at additional stages in addition to increasing throughput essentially a scheme for NC that without resorting to decoding applies additional coding at intermediate code.

#### *1.4.2.2. Robustness to Link Failures*

Protection from non-ergodic link failures is also provided by NC which is in addition to the robustness against random loss of packets. With no requirement for rerouting, a very fast recovery from failures is provided through live path protection where for each connection, a backup flow is transmitted along with a primary flow.

Resource usage can be improved through NC whereby network resources are allowed to be shared among varied flows. Recovery is allowed from any failure pattern in the set by a static NC solution where recovery is possible with an arbitrary rerouting for any failure pattern that is set in the case of a single multicast session.

### 1.4.3. Complexity

It is difficult to obtain the routing solution that is optimal even though a performance similar to the NC one is achieved by optimal routing. For instance, Steiner trees are involved for multicast routing in minimum-cost subgraph selection and even in a centralized setting Steiner tree is quite complex whereas distributed solutions with low-complexity are admitted in a linear optimization which is a NC for the corresponding problem.

Next section discusses this further. Whereas suboptimal solutions become a necessity due to practical limitations, performance can be improved substantially through with NC for instance, 802.11 wireless ad hoc networking and gossip-based data dissemination.

### 1.4.4. Security

Both drawbacks and benefits can be derived from NC as far as the security aspect is concerned. Hypothetically where only packet b1 $\oplus$ b2 is obtained by an adversary, that adversary cannot obtain either b2 or bi with this packet b1 $\oplus$ b2 by itself.

Thus, a secure means of communication is available, which is an advantage resulting from NC. Whereas if b1 $\oplus$ b2 is not sent out as node 3 is a malicious node and masqueraded packet is sent out as b1 $\oplus$ b2 such tampering becomes very difficult to detect as rather than being routed packets are coded. Potential security-related drawbacks result in this specific instance of NC (Figure 1.3).


**Figure 1.3:** Network coding security.

*Source: Image by Snappygoat.*

## 1.5. NETWORK CODING (NC)

In 1999, Zhang, and Yeung introduced the concept of NC for the first time as an alternate to routing. Like the blood corpuscles in bloodstream, data flows in to the destination from the source in discrete "pieces" that are defined within a packet-switched network that is traditional.

Some of the message data in an intact form is contained in each of the packets into which the outgoing message is broken at the transmitting station. All the packets arrive at the same destination even though they may follow different routes to reach there and the original message is obtained at the destination by reassembling through a receiving computer.

However, the main problem in this method is the long delays that are caused due to bottlenecks which commonly occur and the high volume of the overall network traffic. At times over and above the ability of a node to process them, packets bunch up at these nodes, whereas there may be underutilization of other nodes and routes.

Coders are the devices that replace switches and routers in NC. Whereas a system of arteries is used for the flow to the blood cells, metadata is transmitted through coders to the ultimate destination in digital evidence format about the message simultaneously through a number of paths.

On the other hand, a single packet may be the end result of the metadata arriving from a combination of sources which may be two or more. The effective capacity of the network can be increased through this method of distribution, whereby the severity and number of bottlenecks is reduced.

This effectiveness stands out especially when the volume of network traffic is close to the maximum capacity that can be obtained through traditional routing. The message packet that was intended can be computed when sufficient digital evidence is available to the receiver.

If the digital evidence that has been received is sufficient, the original message is comprehended even where some of the packets are mutilated or even lost on certain routes. The data depends upon the contents of all the messages that share a particular route during the period of transmission and not just on the one message that has been transmitted in NC.

This makes NC far more resistant to eavesdropping, hacking, and certain other forms of attack that could be seen in the transmission of data in the traditional manner. The severity of bottlenecks, their frequency, and the network topology, all of them affect the extent to which throughput can be improved.

In comparison to the routing method, the throughput does not get reduced through NC. In peer-to-peer (P2P) file-sharing, wireless sensor networks, multicast networks and digital file distribution, the usefulness of NC may be quite high.

### 1.5.1. Linear Network Coding (LNC)

Where a router, a node in a P2P distribution network or a node in an ad-hoc network acts as an information relay, the system needs to be considered here. This system would simply repeat the information packet that was destined to another node in a traditional set-up.

The packet(s) that have been created or received by the node are combined by it through NC into outgoing packages, which may be one or more. L bits are assumed to be contained in each packet. The shorter packets are padded with trailing 0s when packets of different sizes are combined (Figure 1.4).

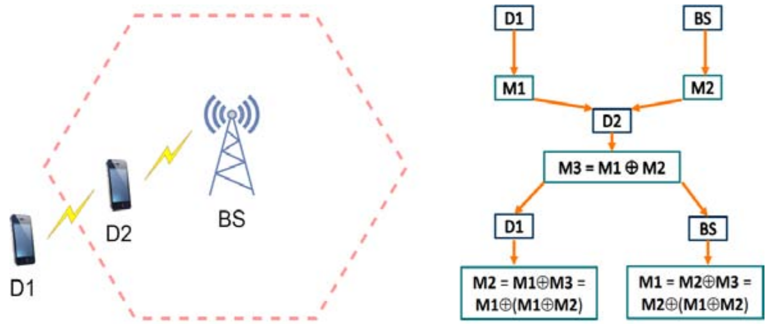



**Figure 1.4:** Network coding applied to device-to-device communication.

*Source: Image by Wikimedia commons.*

Consecutive bits of a packet can be interpreted as a symbol over the F2s field where a vector of L/s symbols comprises each packet. Original packets are combined linearly to form the outgoing packets with LNC wherein over the F2s field, multiplication, and addition are performed.

There is a clear understanding of the algorithms for decoding and coding in linear frameworks because of which they are chosen. Linear combination does not contain a series of interconnected linkages whereby the resulting encoded packet is of the same original size L as the original packets of length L that have been combined linearly.

Only a fraction of the information that is contained in the original packets subsists in the encoded packet, which is in contrast to concatenation. LNC would appear to be a form of spreading information.

## 1.5.2. Encoding

A sequence of coefficients $g_1 \ldots g_n$ are associated with each packet through a network in LNC and is equal to X=n i=1 $g_i M_i$ where one or several sources generate a series of original packets $M_1 \ldots M_n$. For every symbol position, the summation has to take place, namely, i = 1, $X_k$ = n where $X_k$ and $M_{ik}$ is the $k_{th}$ symbol of X and $M_i$ respectively.

For the example, a symbol is a bit, the field is $F_2 = \{0, 1\}$ and the linear combination sent by S after receiving $M_2 = b$ and $M_1 = a$ is $M_1 + M_2$ (in $F_2$, the +sign in addition, i.e., bitwise XOR). It is assumed for simplicity that both the coefficients g= $(g_1, \ldots g_n)$ are contained in a packet which is the encoding vector and the information vector is the encoded data X = n i = 1 $g_i$ $M_i$.

For instance, the information vector is equal to Mi (i.e., it is not encoded) where the encoding vector $e_i$ = (0, … 0, 1, 0, … 0) wherein at the $i^{th}$ position 1 is there. Encoding can be performed on packets that have already been encoded in other words, it can be done recursively.

Consider for instance where a set $(g_1, X_1)$, … $(g_m, X_m)$ of encoded packets have been received and stored by a node where for the $j_{th}$ packet the encoding [resp. information] vector is $g_i$ [resp. $X_j$]. Whereby computing the linear combination X = m j = 1 $h_j$ $X_j$ by picking up a set of coefficients $h_1$, …, $h_m$, a new encoded packet (g, X) is generated by this node.

As the coefficients are in regard to the original packets $M_i$,…, Mn, the corresponding encoding vector is not just equal to h; whereas it can be seen that $g_I$ = m j=I $h_j g_{ji}$ in simple algebra. Within the network at various nodes, this operation may be repeated.

### 1.5.3. Decoding

Here we need to assume that a set $(g_1, X_1)$…, $(g_m, X_m)$ has been received by a node. $M_i$ being the unknowns, the system {$X_j$ = n, i =1 $g_{ji}M_i$} needs to be solved so that the original packet can be retrieved. With n unknowns and m equations this is a linear system.

In order to recover all data, there is a requirement of getting m ≥ n. In other words, with relevance to the original packets, the received packets should be at least as large as the original number. On the other hand, a situation where m ≥ n it may not suffice as there may be a linear dependency of certain combinations.

### 1.5.4. Linear Combinations Selection

Selection of the linear combinations that need to be performed by each node is a problem that needs to be resolved by a network code design. Over the field F2s, a random and uniform selection of coefficients in a manner that is decentralized and totally independent by each of the nodes in the network is a simple algorithm.

There is a possibility of linearly dependent combinations being selected with random network coding (RNC). A field size of 2s is related to this probability. A field size of 2s is the base for such a probability. This probability is reduced to negligible even when the size of the field is small (like when s=8) as shown in certain simulations.

To design network codes alternatively deterministic algorithms can be used. For multicasting the polynomial-time algorithm, each code within the network has been examined sequentially, and as to what linear combinations can be performed by each node has been decided.

Only the information vector needs to be carried by the packets as fixed linear coefficients are used by each node. Additionally, restricted families of network configuration are applied with the existing decentralized algorithms.

## 1.6. NETWORK SECURITY THREATS

Attacks through the service provider, passive monitoring of communications, close: in attacks, active network attacks and exploitation by insiders are some of the ways in which various attacks can take place. Attractive targets in the form of networks and information systems are available to the hackers whereby nation-states should be resistant enough to attack from the threat agents' full range. Whenever attacks take place, the system must be able to rapidly recover from it and simultaneously must be able to limit the damage. There are namely five kinds of attacks which are discussed in subsections.

### 1.6.1. Passive Attack

Encrypted traffic is monitored in a passive attack where the sensitive information and clear text passwords are looked for so that they can be used in various other kinds of attacks. Traffic analysis, decrypting weakly encrypted traffic, monitoring of communication that is unprotected and authentication information like passwords being captured are all included in passive attacks.

The upcoming actions can be seen by adversaries through the passive interception of various operations of the network. Without the knowledge or consent of the user data files or information is disclosed to an attacker via passive attacks.

### 1.6.2. Active Attack

Attempts are made by the attacker to break into or bypass a secured system in an active attack. Trojan horses, viruses, stealth, or worms are the various means for achieving this. Attempts are made to modify or steal information, introduce malicious code, and break or circumvent various features for protection in active attacks.

In active attacks, a network backbone is aimed at, attempts are made to penetrate an enclave electronically, information that is in transit is exploited or an authorized remote is attacked whilst it attempts a connection with an enclave. Dissemination or disclosure of data files, modification of data or DoS are the results of an active attack (Figure 1.5).


**Figure 1.5:** DoS illustration.

*Source: Image by Pixabay.*

### 1.6.3. Distributed Attack

A "back door" program or a Trojan horse is introduced as an adversary code so that the software or component that is "trusted" is attacked and later many other users and companies are distributed with this.

During distribution or at the factory itself malicious software or hardware is the focus of these distribution attacks. In order to gain access which is not authorized with respect to the system function or information at a later stage, a malicious code like the back door to a product is introduced in these attacks.

### 1.6.4. Insider Attack

A disgruntled employee or someone else from the inside is involved in these kinds of attacks where the network is attacked. These attacks may be made with or even without any kind of a malicious intent. Information is stolen, damaged, eavesdropped upon intentionally, used in a manner that is fraudulent or denied access to by users who are authorized by malicious insiders. Lack of knowledge, carelessness, or circumvention of security intentionally for the purposes of task performance does not comprise malicious attacks.

### 1.6.5. Close: In Attack

An attempt to get close to network data, components, and systems physically so that more can be learnt about a network is involved in a close: in attack. Close physical proximity is achieved by regular individuals to facilities, systems, or networks so that they are able to modify, gather or deny access to information that would otherwise be available to the users. Open access, surreptitious entry into a network, and at times both are resorted to in order to achieve close physical proximity.

### 1.6.6. Major Security Threats

1. **Viruses and Worms:** Infecting files, removable media or installed software is replicated through a malicious computer programming code or program which is a virus. On the other hand, a worm is a script or a program that moves through a network by replicating itself and travels largely by sending its own copies through email.
2. **Trojan Horses:** Once a Trojan horse runs on a computer or if it is installed, it causes damage to the even though it may seem like innocuous software at the first instance. Trojans are at times designed to destroy the information or damage files that are there on the system being more than just an annoying factor.
3. **Spam:** Any online communication that is unwanted comprises spam.
4. **Phishing:** Attempts are made to obtain information that is sensitive, like passwords, user names, and details of credit cards or even money at times through phishing. This is done by making the system or the electronic communication perceive it as a trustworthy entity.
5. **Packet Sniffers:** For many years, packet sniffers have been used by administrators of the computer network to perform diagnostic tests, for their network monitoring, or even to troubleshoot problems.

6. **Maliciously Coded Websites:** When damage or a security breach is caused to a system the part of the software system or the code used is essentially a malicious code.
7. **Password Attacks:** In order to log in or find the password to a system so that access can be had to the computer, the predominant way is to make password attacks.
8. **Zombie Computers and Botnets:** When a computer is connected to the internet and a hacker, Trojan horse or virus has compromised it, the computer becomes a zombie in computer science parlance whereby malicious activities of some of the other can be performed it upon it remotely. Attacks in the form of denial of service and email spam can be performed through the botnets of computers that have turned zombie. There is complete ignorance on the part of most owners with respect to such a usage of their computers. This is also the main reason for their being compared metaphorically to zombies.

## 1.7. NETWORK SECURITY PROTECTION MODELS

The cyber-attack prevention models have been studied in detail by the research community investigating the cybercrimes. The main efforts in this were towards improving the access control lists (ACL) on the various devices with network infrastructure and automating Firewall rules for preventing these kinds of attacks.

All the anomalies that could possibly exist in an environment with a single or even a multi-firewall was identified by Alshaer et al. In order to detect a single firewall anomaly which could be intra-firewall as well as inter-firewall ones that exist between firewalls that are interconnected, they presented a set of algorithms.

A number of techniques have been provided to protect and purify the firewall policy from any kind of rule anomalies in the Firewall Policy Advisor. With no need for analyzing the filtering rules beforehand, the firewall policy advisor can be used by the administrator to manage the firewall policies.

A series of firewall policy anomalies that may exist in both distributed as well as centralized firewalls have been defined by the authors whereby, they have also proved that within the firewall policies, only these conflicts can exist.

Thereafter, to detect rule anomalies that can exist inter-firewall, i.e., between inter-connected firewalls and intra-firewall anomalies, i.e., within a single firewall, a set of algorithms were presented by them. The local consistency problems that are there with respect to the firewall rule sets have been analyzed, where the main focus is on the automatic frequent ruleset updates.

When the rules are inserted, removed, or even modified, inconsistencies can be present in the firewall rule sets, and a real time approach to detect these inconsistencies has been proposed by them. A data structure that is quite scalable has been presented in where a different approach has been used by the authors wherein to map the dependencies among the various policies of firewall only O (n) space is required.

Thereafter, an algorithm was designed to be repeated over the data so that any policy conflicts could be detected. They were able to prove in this way that the algorithm having an upper bound O (n 2 log n) is the fastest of all the known algorithms for the resolution and discovery of firewall rule anomaly.

Compared to the original policies, 87% improvement in the number of comparisons overhead was seen to have been achieved by their algorithm when they ran experiments on synthetic and various real-life firewall policies.

A systematic resolution as well as detection of firewall policy anomalies can be done through an innovative framework for firewall policy management called the Firewall Anomaly Management Environment or the FAME.

Further policies can be designed through its firewall policy analysis tool that is based on visualization. A firewall analysis tool too has been designed by researchers in which has also been implemented by them to enable the administrator as well to discover, design, and test the firewall policy that is global.

This can be either a planned policy or even a deployed one. The network topology's minimum description is used by this tool whereby it directly passes the different files that have low level configuration and are specific to the vendor.

A session-based on query-and-answer is used to interact with the user. An access control list's number of rules can be reduced tremendously through the framework proposed by the team of Alex Liu whereas the same semantics can be maintained and for the ACL compression problems that have a one-dimensional range an optimal algorithm can be given whilst presenting a solution that is systematic for compression of ACLs that are multidimensional having mixed field constraints.

Extensive experiments were conducted by them on synthetic as well as real-life ACLs. The TCAM Razor, a systematic approach that is efficient, effective as well as practical was also proposed by Liu and his team. This systematic approach was applied for packet classifiers to minimize TCAM rules.

The experimented with 40 real-life packet classifier groups that were structurally different, and even though optimal packet classifiers are not always produced by TCAM Razor, the TCAM Razor in their experiment achieved 31.3% and 29% of average compression ratio.

To cope with the tremendous range expansion in the current times, ISPs and network administrators can deploy TCAM Razor not requiring packet processing hardware or the modification of TCAM circuits as is required in other solutions.

A model of stateful firewalls was proposed by M. Gouda et al. whereby some packets earlier accepted by the firewall are stored when the same may be required at a later date. The stateful firewalls model as designed by them has various properties that are favorable.

The rich results in the analysis and design of stateless firewalls could be inherited through this. Furthermore, backward compatibility is provided for so that using the model a stateless firewall too can be specified. Also, stateful firewalls that were specified whilst using their model could be analyzed through the models presented by them.

Prior to any attempted access, a large percentage of misconfigurations can be eliminated by using an association rule mining which is a data-mining technique shown by LujoBauer et al.

The costly time-of access delay that can be incurred through accesses can be reduced by 43% by using their methods, whereas 58% of the intended policy can be predicted correctly.

Using the idea of resolve filters addition, a new scheme for resolution of conflict was proposed by B. Hari et al. With respect to a filter database, their main results are algorithms that can detect and resolve conflicts.

They found conflicts in the three firewall databases that were in existence and on which they tried their algorithms, and for each one of them, these are the potential security holes. For the common 2-tuple algorithm, which consists of both the addresses of the source and destination, an optimized version has been described and a general one for the k-tuple filter. Where a restricted set of values is attributed to the three tuples in a 5-tuple case, the usage of the 2-tuple algorithm has been shown by them.

An algorithm containing practical and intellectual combinations has been described by M. Waldvoge et al. They found that precomputation had to be used and markers had to be added on the intellectual side after the basic premise of binary searching on hash tables so that in the worst-case logarithmic time can be ensured.

Logarithmic time in worst-case cannot possibly be provided where only binary search of hash tables is used by algorithms. Mutating binary trees were singled out by them as an idea that pleased esthetically where the extra structure that was inherent in the specific kind of binary search done by them was leveraged off.

For IP lookups, on the practical side, this is a scalable and fast solution where implementation can be done on the hardware or the software. The software projections as given by them for IPv6 are 150–200ns and they are 80ns for IPv4.

Based on their examination of the structure of the routing databases that existed, the average case speed has been projected. Based on the basic algorithm which is already performing the overall performance can be easily restricted.

The designing and implementation in the NetBSD operating system kennel of an extended integrated service router software architecture that has a high performance and is modular was the main goal of the work in. Plugins or code modules which can be added dynamically and configured at run time can be allowed through this architecture.

Ethernet switches that are largely a commodity have been shown by M. Al-Fares et al. to be used to leverage so that the full aggregate bandwidth of clusters can be supported whereby tens of thousands of elements are contained in these clusters.

They projected that in comparison to the current solutions that are higher-end, more performance may be delivered at a lesser cost with commodity switches that have interconnected and architected appropriately in a manner similar to the one in which the more specialized MPPs and SMPs have been replaced to quite an extent by clusters of commodity computers.

The backward compatibility of this approach is full with respect to IP, Ethernet, and TCP, not requiring any modifications to the operating system, network interface or applications of the end host. Such anomalies can be detected and resolved through an automated process presented by Abedin et al. An easy to maintain and understand anomaly free ruleset which is compact should be produced by the anomaly resolution algorithm and merging algorithm.

There can be an integration of this algorithm into the editing tools and policy advisor. The relations between rules too can be completely defined and analyzed by this algorithm.

The anomalies of the XACML policy can be systematically detected and resolved through the innovative mechanism represented by H. Hu et al. An effective anomaly analysis' goals can be achieved through a segmentation technique introduced by them which is essentially policy-based.

The implementation of XAnalyzer which is a tool for the analysis of policy anomaly has also been described. Using XAnalyzer the anomalies existing in an XACML policy can be easily discovered and resolved by a policy designer.

To minimize the ACLs in network routers a geometrical model was considered by Applegate et al. Overwriting all the colors existing previously in the rectangle, the aim was the creation within a rectangular canvas which was initially white, of a colored rectilinear pattern and the basic operation was to paint a chosen sub-rectangle with a single color.

Finding the list of rules that is the shortest for the creation of a pattern that is given is the problem of rectangle rule list (RRL) minimization. In an ACL application, black and white are the only colors (permit or deny), whereas using strip rules, several achievable equivalent characterizations of the patterns are provided by RRL along with the presenting of polynomial-time algorithms whereby such patterns can be optimally constructed.

Through exploitation of their results about strip-rule patterns, approximation algorithms for ACL minimization and general RRL has been provided O (min (n1 = 3; OPT1 = 2)) by them whilst showing that RRL-minimization is NP-hard. However, the integrity of the ACLs of the router was not addressed even though this work was quite elaborate.

The integrity of the Access Control List of the router with respect to networks of large enterprise was investigated by Elnour and Ahmat. More so, they studied the issue of redundant ACLs, their discovery and elimination thereof from the configurations of multiple routers describing the methods that could be used to remove these kinds of redundancies efficiently.

Within network infrastructures that were complex, likely security holes could be discovered through their approach whereby they validated the practicality of their algorithms by not only proposing them but also actually implementing them.

An initial design of a prototype was presented by Y. Bartal et al. and further implemented as well for security management and firewall tools of a new generation whereby its usefulness could be seen in an example in the actual rather than assembly code, towards the convergence of network and security management the first important step was demonstrated whereby security management/configuration and firewall task can be successfully done at a level of abstraction that can be compared to the programming languages that are modern.

As a part of an explicit Internet content layer based on name-based routing, a content routing design was later described by M. Gritter et al. Being a natural extension of the routing systems and the Internet directory currently in use, content location can be done efficiently through content routing and the same can be to scale implementation with the Internet. Results have shown that client lookup is then far less variable and is much faster.

## 1.8. CONCLUSION

To conclude this chapter, a brief introduction to the NC along with significance of the NC has been discussed. This chapter discussed about the origin of NC, history of the computer networks, and the development of computer networks.

This chapter also discussed about the several numbers of applications that are effective for the users, such as throughput, robustness of the NC, complexity of the NC, along with the importance of security in NC.

It has been observed that there are various kinds of NC that are being used in a present interval of time, such as LNC. This chapter also explained the importance of linear combination selection, encoding, and decoding.

Towards the end of the chapter, various types of threats have been mentioned that are related to the network security, such as passive attack, active attack, distributed attack, close-in attack. In order to cope up and protect with these security threats, various network security protection models have also been mentioned.

## REFERENCES

1. Fragouli, C., Boudec, J., & Widmer, J., (2010). *Network Coding: An Instant Primer*. [Online] Eecs.harvard.edu. Available at: http://www.eecs.harvard.edu/~michaelm/CS222/primer.pdf (accessed on 3 May 2021).
2. Golchha, P., Deshmukh, R., & Lunia, P., (2014). *A Review on Network Security Threats and Solutions*. [Online] Ijser.in. Available at: https://www.ijser.in/archives/v3i4/IJSER1567.pdf (accessed on 3 May 2021).
3. Hassan, A., & Hassan, M., (2015). *Introduction and Applications of Network Coding*. [Online] Irjet.net. Available at: https://www.irjet.net/archives/V2/i8/IRJET-V2I878.pdf (accessed on 3 May 2021).
4. Ho, T., & Lun, D., (2007). *Network Coding: An Introduction*. [Online] Citeseerx.ist.psu.edu. Available at: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.926.1425&rep=rep1&type=pdf (accessed on 3 May 2021).
5. Koetter, R., Effros, M., & Médard, M., (2007). *A Brief History of Network Coding*. [Online] Scientific American. Available at: https://www.scientificamerican.com/article/history-of-network-coding/ (accessed on 3 May 2021).
6. Kumar, A., & Malhotra, S., (2012). *Network Security Threats and Protection Models*. [Online] Arxiv.org. Available at: https://arxiv.org/ftp/arxiv/papers/1511/1511.00568.pdf (accessed on 3 May 2021).
7. Kumar, D., (n.d.). *Computer Network*. [Online] Ddegjust.ac.in. Available at: http://www.ddegjust.ac.in/studymaterial/mca-5/mca-301.pdf (accessed on 3 May 2021).
8. SearchNetworking, (2007). *What is Network Coding?* Definition from WhatIs.com. [Online] Available at: https://searchnetworking.techtarget.com/definition/network-coding (accessed on 3 May 2021).

CHAPTER 2

# THEORETICAL FRAMEWORK FOR NETWORK CODING

## CONTENTS

The chapter theoretical framework for network coding (NC) emphasizes the various forms of a network multicast model such as the basic model, the line graph model. This chapter also explains the algebraic framework of NC, along with the different frameworks that are used in NC like start-up phase, and running phase.

This chapter also provide highlights on the trusted network routing and intermediate nodes routing table. This chapter addresses about multipath routing that are based on NC. The chapter of theoretical framework for NC provides a brief introduction about index coding with the help of linear programming, and multipath routing. This chapter also explains the information-theoretic frameworks that have been implemented in NC.

## 2.1. INTRODUCTION

Various research communities have studies network coding (NC) using varied theoretical frameworks. The preferences and background of the researcher help him/her decide upon the framework chosen for the research.

At the same time, it needs to be generally accepted that irrespective of the framework chosen, all the issues related to NC like throughput benefits, code design and complexity need to find a place within the framework so that their study can take place in the most efficient and natural manner.

Herein, the results obtained within the frameworks have been pointed out upon consideration of the tools from combinatorial, algebraic, linear programming and information-theoretic frameworks. This chapter has only examined traffic scenario of multicasting.

## 2.2. A NETWORK MULTICAST MODEL

### 2.2.1. The Basic Model

A source vertex $S \in V$ (on which h unit rate sources $S_i$ are collocated), a directed graph $G= (V, E)$ and a set $R = \{R_1, R_2\ldots, RN\}$ of N receivers have been used to describe a multicast scenario. All these three ingredients have been considered together here as (multicast) instance $\{G, S, R\}$.

The main theorem on NC reveals that each receiver's min-cut has to be equal to or greater than h, which is a condition both necessary and sufficient for the feasibility of rate h multicast. For rate h, this condition is called the multicast property.

From sources to receiver $R_j$, we can let the set of h edge-disjoint paths be $\{(S_i, R_j), 1 \leq i \leq h\}$. The min-cut max-flow theorem guarantees that such paths exist where the min-cut to each receiver is assumed to be equal to at least h.

There is no uniqueness to the paths that are chosen and the complexity of the network codes is affected by this. The same has been discussed later. Sub-graph G of G is the main object of study here whereby it contains hN paths $(S_i, R_j)$, $1 \leq i \leq h$, $1 \leq j \leq N$. the multicast property is also satisfied by the instance {G, S, R} (Figure 2.1).

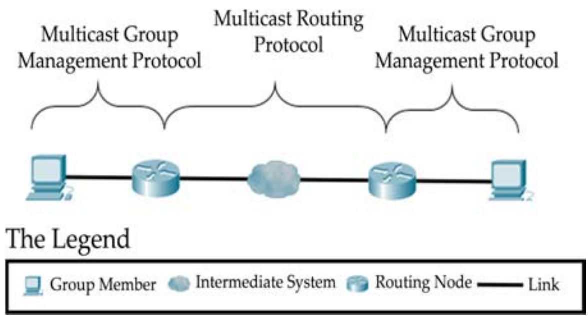



**Figure 2.1:** Multicast protocols.

*Source: Image by Wikimedia commons.*

Form each input edge, an element of $F_q$ is received by each node of G, and then the linear combinations (which may even be different) of these symbols get forwarded by it to its output edges.

A local coding vendor c(e) with a dimension of 1 × |In(e)| collect the coefficients that define the linear combination on edge e and the parent node of e receives the (e) which is the set of edges incoming to it.

A linear combination of the source symbols is carried by each edge and the h-dimensional global vector c (e)=[c1(e) ··· ch (e)] collects the coefficients that describe this linear combination as a result of this local linear combining. c1(e) σ1 + ··· + ch (e) $\sigma_h$ is the connotation of the symbol through edge e.

The rows of the matrix $A_j$ are formed by the h coding vectors that the receiver $R_j$ takes from the h input edges to solve the systems of linear equations. Each edge of the graph is assigned with a local coding vector or coding vectors equivalently through a network code design.

Where the conditions of the main NC theorem are satisfied by the multicast networks, it has been seen that over some fields that are large enough, there is the existence of a valid linear network code.

All the same, no valid linear network codes exist for certain other network scenarios and still some other network scenarios, there are no valid network codes then be it nonlinear or linear.

Solving all multicast instances linearly is thus possible. Linear combining at edge e needs to be performed in linear network coding (LNC) where two or more path exist using distinct edges of In(e) but are sharing e. Edge e then becomes the coding point.

Instead of simple forwarding, additional processing capabilities are required in practice at these places in the network. In the same way, minimal graphs or the ones not having redundant edges are a matter of interest.

The number of coding points would be reduced, and lesser network resources would be required if before multicasting minimal graph is identified. The same has been shown in the examples below. Using k coding points by stacking k butterfly networks, a non-minimal configuration can be created to extend the same construction.

Alternatively, it can be seen that at the most one coding point can be had by minimal configurations with two receivers and two sources. Hence, the number of coding points can be reduced tremendously by identifying minimal configurations.

Further, it shall show that in polynomial time identification of a minimal configuration can be done for an instance {G, S, R}. On the other hand, an NP-hard problem in most cases to identify a minimum configuration which has the least possible number of coding points.

## 2.2.2. The Line Graph Model

The linear combination of source symbols carried by each edge has to be specified eventually so that a network can be defined. A graph is thus a more transparent means to work with where an edge of ($S_i$, $R_j$) is represented by each vertex of L ($S_i$, $R_j$), the line graph of the path ($S_i$, $R_j$) is denoted by L ($S_i$, $R_j$) and only if the corresponding edges of the vertices of L ($S_i$, $R_j$) share a common vertex in ($S_i$, $R_j$) that the vertices of L ($S_i$, $R_j$) are adjacent. Three receivers and two sources of the network along with its line graph (Figure 2.2).

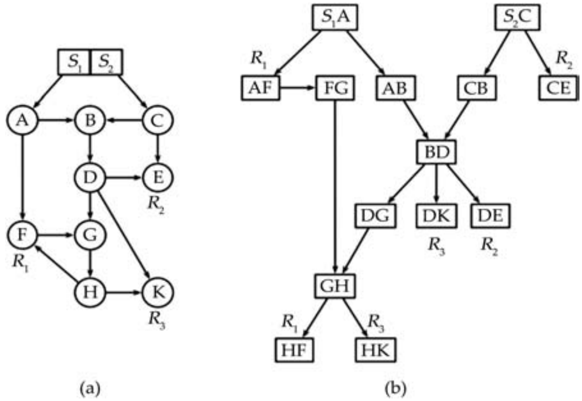



**Figure 2.2:** Line graph model.

*Source: Image by Soljanin and Fragouli (2007).*

It can be assumed that there are exactly h outgoing edges in graph G of the source vertex S where each h co-located source corresponds to one of them without the loss of generality (this may be possible by introduction of h auxiliary edges and an auxiliary node).

Corresponding to each of the h sources a node is contained in the line graph. S2C and S1A are source nodes (SNs). The input symbol is forwarded to the output edges by each node in γ which has a single input edge.

A coding operation is performed, in other words, linear combining is done by each node which has input edges that are two or more on the input symbols whereby the result is forwarded to all its output edges. In definition, these nodes form the coding points.

The definition of the coding points becomes transparent by using the line graph notation. GH and BD are the coding points. Eventually, the receiver node for receiver R j and source Si is the node that corresponds to the last edge of the path ($S_i$, $R_j$). N receiver nodes exist for a configuration that has N receivers and h sources. The receiver nodes, are AF, CE, DE, HF, and HK. The definition of minimality, valid, and feasible networks translate directly for line graphs.

The parent nodes of the vertex corresponding to e are the vertices that correspond to the edges of In (e). The coefficients of local coding vector c(e) label the edges that come into the node e. Hence, in the line graph, the values {αk} for the labels of the edges are essentially chosen in designing a network code.

## 2.3. ALGEBRAIC FRAMEWORK

Algebraic was one of the early approaches that stimulated the field of network coding.This approach has been used to explain the fundamental sasit requires very little back ground and is quite straight forward.

Within the framework of algebraic randomized coding has been developed, and the same is considered to be of a great importance in the field of practical NC. It is quite easy to explain this approach where the notion of line graph is used in which either a unique label corresponding to a variable in {αk} or a label 1 is carried by each edge.

Each vertex of the line graph, which is in other words the edge in the original graph, is thought of as a memory element where an intermediate information symbol is stored.

Further, the line graph acts as a linear system when a specific receiver $R_j$ is considered with h outputs (that are observed by the receiver) and h inputs (which h sources) right up to memory elements, m: = |E|. The set of state-space equations that are finite-dimensional describe this system.

$$s_{k+1} = As_k + Bu_k,$$
$$y_k = C_j s_k + D_j u_k,$$

where; A, B, $C_j$, and $D_j$ are matrices; $s_k$ is the m × 1 state vector; $u_k$ is the h × 1 input vector; and $y_k$ are the h × 1 output vector with dimensions that are appropriate as has been discussed below. A convolutional code can be described through the state space equations, which have been explained in detail.

The transfer matrix $G_j$ (D) is given as a standard result through the linear system theory where the indeterminate delay operator is D. the transfer matrix for receiver $R_j$ by using unit delay is:

$$G_j(D) = D_j + C_j(D^{-1}I - A)^{-1}B,$$

where; D is the indeterminate delay operator. Using unit delay, we obtain the transfer matrix for receiver $R_j$.

For all the receivers in matrix A is common and the network topology or the manner in which the states or memory elements are connected has been reflected. The nodes of the line graph, i.e., the state's index its elements and if between the indexing states there is an edge in the line graph, only then the element of A is non-zero.

The no-zero elements equal to either an unknown variable in {αk} or 1. For the variable entries in A, values are selected through network code design. The manner in which the inputs or the sources are connected on the graph is reflected through matrix B which is the same for all the receivers.

Depending upon the inputs and state variables, the manner in which the outputs receiver R j observes is expressed by the matrices $C_j$ and $D_j$, respectively. Through the introduction of edges and auxiliary vertices binary matrices can be chosen to be played by matrices like B, $C_j$, and $Da_j$.

The number of edges in the graph can be quite high, and the dimension of matrices A, B, $C_j$, and $D_j$ depends on this number. The manner in which this number can be reduced to quite an extent has been discussed in the next section.

For acrylic graphs, matrix A becomes strictly upper triangular due to the ordering of the elements of the state space vector therefore turning it nilpotent (for some positive integer n $A_n = 0$). If the longest path's length between the source and the receiver is denoted through L, AL+1=0. This goes to say that:

$$(I - A)^{-1} = I + A + A^2 + \cdots + A^L$$

The main theorem uses the proof that in the unknown variables {αk}, the polynomials are the elements of the transfer matrices $A_j$ as implied in this equation. Furthermore, an intuitive explanation has been offered.

Effectively, A can be easily seen to be an incidence matrix. All the paths connecting the network edges are accounted for in the series. The information that is brought from the sources to the receivers along these paths can be expressed through the transfer matrix.

## 2.4. NETWORK CODING (NC) FRAMEWORK

The lifetime of the wireless nodes can be increased, their availability can be enhanced, throughput increased, and the overall performance of the network can be improved when the multicasting approach is applied with NC, as can be seen through many researches.

If the malicious packets are stopped in the network once they have been identified, these characteristics of the network shall get enhanced. The incoming data should be validated by the sink node before the same is processed by it where intermediate nodes are being used to transmit data from the SN to the sink node.

It can thus be checked through this validation as to whether the data being transmitted is legitimate bearing origins in a real source or if it is being injected from the malicious nodes and is malicious data.

Different methods are used by researchers to create a trust relationship between the source and the sink so that the incoming data can be validated which may include: the generation of a sequence number between the SN and the sink node, encrypting the packets transmitted from the source, creation of a trust model between the SN and the sink node or using specific defined values to sign the data between the two ends like for a sink node, the MAC address.

Certain idealized assumptions form the base for NC for many researchers like for instance, the assumption that there is no existence of the transmission error and the malicious node.

These assumptions do not hold good in practical wired/wireless networks as the potential of NC shall get limited due to this. Due to this in the current times, the focus of many researches in the application of networking coding theory to networks that are practical.

### 2.4.1. Start-Up Phase

By using a separate packet to end the initial conditions of the chaotic system, initiation of trust with the sinkhole is initiated by the SN whereby chaotic codes can be generated independently by two ends during the start-up phase. A logistic map is used to generate a chaotic code as shown below:

$$X_{n+1} = \lambda X_n (1 - X_n)$$

Multiplying the output value X by 255 it can be scaled to 255, and the resulting value can then be rounded off. Due to certain advantages like deterministic behavior, its high sensitivity to the initial conditions, and unpredictability, the adoption of the chaotic generator is done in the framework that is proposed.

For example, the prediction of the code is very difficult when the X differs with 10–15 a completely different behavior is generated by the generator.

$$Y = [X_{n+1}] * 255$$

Further, to validate the incoming packets that come from the intermediate nodes, chaotic codes shall be created by the sink once the initial conditions have been sent to the sink node. The SN is then informed by the sink node of any malicious link that is then generated if the code is different from the one generated at the source site.

### 2.4.2. Running Phase

Verifying stage and the coding stage are the two stages into which the running phase is divided. Intermediate codes are used to encode the data at the coding stage. Using the chaotic code that is generated in the two-sided data is verified at the verifying stage on the side of the sink. Thereafter, the source is informed about the malicious code by the sink. The same has been shown in the following sub-sections.

#### *2.4.2.1. Coding Stage*

Using the packet data is sent to the sink by the source through the intermediate nodes. Within each packet is contained: data ID, data, packet type and chaotic code. The three bits filed is essentially the packet type which describes the kind of packets: data packet, start-up packet or the coded packet, etc., the chaotic code's sequence number is described by the data ID field. The number generated in the source is validated and forms the chaotic code.

Furthermore, if the intermediate node has more than one upstream link, the data and the chaotic code are coded by the node and using the packet, send them along with their data IDs. Till the time the packets reach the sink node packets are either forwarded or the incoming packets coded by each node. To check the packets' incoming source, the validation and verification of the packets is done at the sink node.

#### *2.4.2.2. Verifying Stage*

Before the packets are sent to the sink node data ID is inserted into the chaotic code by the source. The incoming packets' chaotic code is then checked by the sink when these packets are received by the sink node, and the code shall be accepted by the sink if the generated code and the incoming code as came from the SN are equal. Alternatively, a malicious code ID is sent by the sink to the source informing the neighbors so that the forwarding as well as receiving of the information that has used a malicious node can be stopped. Example 1 has been used to clarify the running phase idea.

*Example 1:* using packets, data along with the relevant chaotic code is sent by the source to the sink assuming the network. By XORing chaotic code fields and data fields, two packets are combined by a coded node like node F.

Till the packet reaches the sink node, the packet that has resulted is sent to the next hop. The chaotic code is read and validated in the sink node. Through the link, the sink continues to receive the information from the source as long as the chaotic code is the same.

On the other hand, is an invalid chaotic code where an invalid packet is received by the sink the malicious packets' path will be blocked by the sink and the same is informed to the source so that further data is stopped from being forwarded through that node?

The reliability and performance of the network can be improved upon by the usage of coding nodes. Hence without the need to record the data, by solving the linear independence relationship in the sink, lost data can be retrieved where a coded data has been lost in the system.

Whereat any point of time more than two data are lost simultaneously, the data needs to be reordered so that then one can be used to calculate the other as the system cannot regenerate the missing data lost at the same time. Hence when the linear independence relationship is to be solved in the sink node, the reliability and performance gets restricted.

Furthermore, receiving of the meaningless packets shall be stayed in the sink when corrupted linear independence data is sent by the two nodes. In order to maximize the knowledge of the event, one of the methods can be to let the coding nodes send data in a different time unit where the coding nodes are no longer coding nodes.

## 2.5. TRUSTED NETWORK ROUTING

### 2.5.1. Intermediate Nodes Routing Table

Routing tables are required at each node for forwarding the packet to the next hop where the intermediate nodes need to be thus modified so that the proposed framework can be applied in real networks.

For each intermediate node, three columns are contained in the routing table which are: neighbor id column which contains the ID of the nodes, binary values 0 1 contained in the send column which detail as to whether or not data is to be sent (which depends upon the information coming from the source) and the binary value 0/1, zero contained in the up/down column which represents the downstream links as well as one for upstream links.

The Up/Down column equals zero for the SN. Table 2.1. shows the routing table for node D. Two neighbor nodes, one in downstream link (T) and the other in upstream link (A) are contained in this. Data is received from A and sent to T which is the send column status for two nodes.

**Table 2.1:** Routing Table for Node D

| Neighbor ID | Send | Up/down |
|---|---|---|
| A | 1 | 1 |
| T | 1 | 0 |

## 2.5.2. The Sink Table

There being no routing table in the sink node, to the sink node creates the table that describes the cost between the sink and the nodes whereby information is contained with respect to every link that is connected to the sink. Four columns are contained in the sink's Table 2.2. Namely: node type, node ID, the link cost, and the data. Node ID connected to the sink is contained in the first column, node status is described in the node type wherein it takes one of the ensuing values: 1 (Backup node), 0 (that describes the legitimate node) and 3 (monitor node which has been discussed later). The data incoming from the intermediate node is stored in the data column and link cost from the source to the sink is represented by the last column. To receive all the knowledge packets, the sink node waits for a specific amount of time; however, if the packets are not received within that time, the missing data is reordered by the sink to be sent through a different path. All the same, once the event has been recovered, the table of the sink is flushed.

**Table 2.2:** Sink Table

| Node ID | Node Type | Data | Link Cost |
|---|---|---|---|
| D | 0 | 122 | 25 |
| E | 0 | 324 | 10 |
| F | 0 | 465 | 8 |
| G | 0 | 123 | 30 |

### 2.5.3. Coding Nodes Model

The coded packets are used by the sink to reconstruct the event from information that is meaningful. The data shall not be useful where the number of coded packets fall short of the number that is required and the knowledge of the vent cannot be increased.

Hence the coded packets need to be minimized so that |K| can be increased and change the behavior of the nodes so that the coding nodes can send information that is in varied time units and the node can be in any of the two packets that have been coded.

The maximum amount of time that is required for information to reach the sink from the source is called the threshold time and transition between the previous states depends on this.

Knowledge is attempted to be created by the sink from the incoming packets after the threshold time, and the node stays in a coding state if the packets are meaningful and shall help to enhance the knowledge. But if due to paucity of data, the information received by the sink cannot be decoded, the node's state is changed by the sink node to un-coding state, and the same is marked as a monitor node after $T_{th}$.

## 2.6. INFORMATION-THEORETIC FRAMEWORK

It would be helpful to decompose a network into smaller modules or subunits so that the structure of large-scale social, technological, or biological network can be understood. By capitalizing on regularities in the structure and finding an optimal compression of its topology, the modules which comprise the network can be identified. Through partitioning of a series of real-world and model networks, the advantages of this approach have been illustrated and explained.

Social, technological, or biological systems comprise the various groups in which most of the objects in nature varying from humans to proteins interact. A distinct intermediate level is formed by the groups between the systems' macroscopic and the microscopic descriptions, and the aspects of system function, which include robustness and stability, may often be coupled with the group structure. When a complex system's interactions among components are mapped to a network with nodes that are connected by links, highly connected modules are formed by these groups of interacting objects which are not connected to each other strongly. Thus, through the identification of the communities or modules which compose the complex network, the structure of this network can be comprehended. When a set of modules that are interconnected describe a network, certain regularities of the network's structure are highlighted while the details that are relatively unimportant are weeded out.

Hence, finding the structure an efficient compression can be seen to be the problem identifying itself as that of community identification and a lossy compression of the topology of the network can be seen as the modular description of a network. As per this view, the challenge of identifying a complex network's community structure can be approached as information theory's fundamental problem. The groundwork towards an information-theoretic approach towards community detection has been provided, along with an exploration of the advantages that this approach vis-à-vis the other methods that are used for community detection.

Through a simplified summary of the module structure, the process whereby a complex network is described has been envisioned as a communication process. A random variable X is the link structure of a complex network, the full form of the network X is known to a signaler and the aim is towards the conveyance of this information to the signal receiver in a reduced manner. The information about X is encoded by the signaler as Y, a simplified description. Through a communication channel that is noiseless, the encoded message is then sent. The message Y is observed by the signal receiver who "decodes" this message whereby it is used to make guesses Z about X, the original network's structure (Figure 2.3).

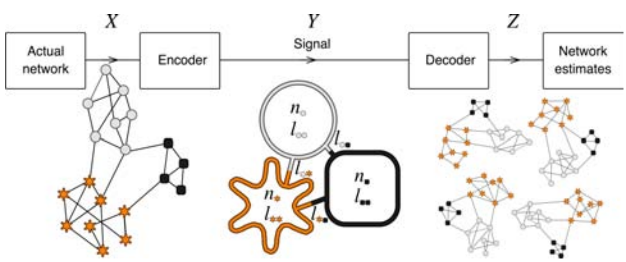



**Figure 2.3:** Information-theoretic framework.

*Source: Image by Rosvall and Bergstrom (2007).*

Using Y, a simpler description network X can be described in many varied ways. Depending upon what needs to be done with the description one can decide as to which of these descriptions is the best.

All the same, a general answer which is quite appealing has been provided by the information theory. From amongst all the descriptions $Y_i$, for a random variable *X,* the best description *Y* is the one through which maximum is told about *X* in other words, through which the mutual information *I (X; Y)* is maximized between the description and network.

The descriptions Y through which the structure of a network X has been summarized shall be explored here as the main interest is to identify community structure wherein the relations between the modules within X are described by enumerating the modules or communities within X.

One specific method of encoding the community structure of X shall be considered in this chapter. To be able to choose the best encoder suitable for the specific problem at hand, one should consider the various alternate "encoders" that can possibly be considered.

The adjacency matrix can be used to describe the undirected and unweighted network *X* with *l* links and of a size *n* that has been considered here.

$$A_{ij} = \begin{cases} 1 \text{ if there is a link between nodes i and j} \\ 0 \text{ otherwise} \end{cases}.$$

$$Y = \left\{ a = \begin{pmatrix} a_1 \\ \vdots \\ \vdots \\ a_n \end{pmatrix}, M = \begin{pmatrix} l_{11} & \cdots & l_{1m} \\ \vdots & \ddots & \vdots \\ l_{m1} & \cdots & l_{mm} \end{pmatrix} \right\}$$

Where the module assignment vector is a, form modules, $a_i \in \{1, 2, \ldots, m\}$ and M is the module matrix. The manner in which the m modules given by the assignment vector are connected in the actual network is described by the module matrix M=M (X, a). There are no nodes in the module I and $l_{ij}$ links connect it to module j (refer Figure 2.1).

The mutual information is maximized to find the best assignment a* over all the assignments that are possible of the nodes into m modules.

$$a^* = \underset{a}{\arg\max}\, I(X; Y).$$

Where the information required to describe X given Y is the conditional information H (X|Y) = H (Z) and the information essential for the description of X is H(X), the mutual information I (X; Y) = H(X) – H(X|Y) = H(X) – H(Z) (refer Figure 2.1).

Hence the effort is to minimize H(Z). in other words, this amounts to the construction of an assignment vector so that the possibility of the set of network estimates Z being small is the maximum in Figure 2.1. Where nodes to m modules are assigned by description Y:

$$H(Z) = \log\left[\prod_{i=1}^{m}\binom{n_1(n_1-1)1/2}{l_{ii}}\prod_{i>j}\binom{n_i n_j}{l_{ij}}\right]$$

Wherein the binomial coefficients are denoted by the parentheses and in base 2 the logarithm is taken. The varied modules that can be constructed with $l_{ii}$ links and $n_i$ nodes and the number thereof is given by each of the m binomial coefficients in the first product.

The module j and i can be connected to each other in a number of ways, and this is given in the second product by each of the m (m–1)/2 binomial coefficients.

The dolphin social network is applied to the cluster-based compression method. A division that differs by only one node from the division along which the split was observed in the actual dolphin groups has been selected in this method.

In order to search for a partition via which the mutual information between the original network and the description can be maximized, simulated annealing with the heat-bath algorithm has been used as checking of all possible partitions is computationally not feasible even where the networks are sized modestly.

Exhaustive searches with proximity to the Monte Carlo solutions have been done to confirm the results for the networks. Girvan and Newman introduced the modularity approach and the partition obtained by using this approach has been used here to compare the results thus obtained.

The simplicity involved in this approach is quite appealing, powerful numerical techniques by which large networks can be dealt with and its performance in benchmark tests make it possible to widely adopt this technique. The contributions from each module i add up to the modularity Q where the partitioning is into m modules.

$$Q = \sum_{i=1}^{m} l_{ii} / 1 - (d_i / 2l)^2,$$

Where between nodes in the $i^{th}$ module there are $l_{ii}$ number of links, the total number of links in the network is l and the total degree in module i is di.

Merely the number of links in the network is not minimized when the modularity is maximized. On the other hand, a configuration is found whereby in the actual network, the number of links within modules is maximized other than the number of links that are expected in a random network within comparable modules having the degree sequence that is same.

Further, the aim is to divide the network so that within the modules, the number of links is much more than was expected. For networks with similar-sized modules and degree sequence, the approach works quite well.

All the same, when the modularity approach is used to partition the dolphin network, the division that takes place in the network is quite different from the fission empirically observed in the dolphin group.

The choice of partition is highly influenced by and is sensitive to the system's total number of links due to the denominator in the second term of the definition modularity (as seen in the equation above). Groups with similar total degree are favored by the benefit function by which modularity is defined, which goes to say that the size of the network as a whole affects the size of the module.

The benchmark tests were conducted so that the quantitative comparison of the performance of the compression method that is cluster-based can be done the approaches that are modularity based. With an average degree 16 four equally sized groups have been obtained by the division of 128 nodes in these tests.

Identification of the underlying group structure becomes more and more difficult with an increase in the average number of links $k_{out} = 6, 7, 8$ from each node to nodes in other groups. Similar to the dolphin networks when there is a complete variation in the size of the group or in the total degree, it becomes even more difficult to resolve the community structure through the modularity approach.

A series of test networks was formed in the benchmark test by merging three of the four groups whereby a small group with 32 nodes and a large one with 96 nodes is formed in the test networks. Either approach finds it difficult to resolve these networks that as asymmetrically sized however, with a big margin, the underlying community structure are recovered through cluster-based compression more frequently as compared to modularity.

Using networks comprising of two groups where each of them has 64 nodes but where the average degrees were different being 8 and 24 links per node were used to conduct a set of benchmark tests. Community structure is again recovered more often by cluster-based compression as compared to modularity when $k_{out} = 2, 3, 4$ was used in these networks.

The challenge of model selection was addressed after this. There is generally a twofold task with respect to the resolving of resolving community structure as only in very special cases is it known a priori as to how many modules shall compose the sample network. The nodes need to be partitioned into the number of modules that have been determined in the network.

However, the point is that unless the assignment of nodes is considered, the optimal number of modules cannot be determined. Hence the need to simultaneous solve these problems.

A solution based upon the algorithmic information theory a solution has been provided below. To enable the decoder to make an estimate of the actual network that is the best, a compression of the network is attempted to be found by the encoder.

Letting the network being partitioned into n modules by the encoder where one module is for each node would be one approach. It is thus ensured that the network can be completely reconstructed by the decoder; however, nothing is gained in either module or compression identification under this approach.

Hence, to describe the network in modular form, the amount of information that is required should be balanced by the encoder in the manner given by the signal Y and as per the size of the set of networks, Z is estimated through the uncertainty that stays when the modular description is received by the decoder. Using the principle of minimum description length (MDL), there can be a resolution to the problem of optimal coding. Without overfitting the actual network X, the irregularities in the structure are aimed to be exploited and to summarize it in a condensed form.

Where for the journal citation network, a set of modules has to be chosen, so that on an annual repetition of the experiment, there is a likelihood of the same module being assigned to each journal, it implies overfitting. Even though the data for a specific year is captured quite a bit when data is overfitted, inadvertently, certain noise too is captured that does not recur in the ensuing year.

## 2.7. INDEX CODING VIA LINEAR PROGRAMMING

In the recent times, efficient communication in wireless networks, applications like the fast video-on-demand and the connection that index coding shares with NC have all led to considerable attention being levied on index coding.

In various settings, efficient heuristics and optimal encoding schemes were studied, and new results were obtained for NC like hardness of approximation as well as the gaps between non-linear and linear capacity improving.

With the fundamental parameter as broadcast rate β, the average cost for communication per bit for messages that are sufficiently long (namely the non-linear vector capacity), the problem input, the side-information relation is encoded by the basic setting of Index Coding.

Recently Bar-Yossef et al. (2006); Lubetzky and Stav (2007); and Alon et al. (2008) studied other Index Coding capacities (for instance the scalar capacity β1) and derived nontrivial bounds on β.

All the same light is shed on the behavior of β by these indirect bounds: in a general network for approximating β to within a nontrivial factor (i.e., o(n)) no known polynomial-time algorithm was there and where Index Coding is nontrivial the exact value of β remained unknown for any graph.

Using linear programs (LPs), the main contribution is an analysis of the broadcast rate β that is direct-information theoretic as opposed to the approaches used earlier in which β was compared with parameters that were graph-theoretic.

The two open questions mentioned above can thus be resolved. With a nontrivial approximation ratio, a polynomial-time algorithm is provided by which β can be computed in a general network and additionally for recognizing instances with β = 2, a polynomial-time decision procedure along with it.

Additionally, for different classes of graphs, β has been pinpointed precisely (for instance, for the cyclic groups' different Cayley graphs) whereby the upper and lower bounds for these graphs as were known earlier could be improved simultaneously.

Graphs can thus be constructed using this approach wherein in the number of vertices there is a linear difference between β and its trivial lower bound, and there is uniform bounding of β whereas the upper bound is polynomials worse being derived from the encoding scheme that is naïve.

A set of messages that need to be broadcast to the receivers over a noiseless channel in the Index Coding problem are held by a server. Some subset information of other messages is comprised in the side-information that a receiver has as each receiver has interest in one of the messages.

Devising an encoding scheme for the messages that is optimal (for instance one that minimizes the broadcast the length) is the objective where the input is the side-information map where the message allows the retrieval of all the relevant information by the receivers is allowed.

Where the side-information map of the clients is given the encoding, the scheme is optimized by this notion of source coding. Transmission of large files (for instance, video on demand) where the server is informed through a slow uplink of the side-information map is included in the motivating applications, which mainly include the identities of the files which due to past transmissions are stored at each client.

The shortest possible broadcast is then to be issued by the server so that while minimizing the overall latency, each client is allowed to decode its target file. When each message has exactly one receiver, a problem arises as an important special case, i.e., for all j f(j)=j and m=n. Binary relations consisting of pairs (i, j) can be used to equivalently describe the side-information map N(j) in this case such that $x_j \in N(i)$.

Where the relation is symmetric, these pairs can be taken as the edges of an undirected graph and on the vertex set [n] as the edges of a directed map. Birk and Kol introduced the original Index coding problem to which the special case of the problem corresponds (hereinafter identified by stating that G is a graph), and owing to its rich connections with Ramsey theory and graph theory, it has been studied extensively.

Simple relations between the other graph-theoretic parameters and the broadcast rates result in these connections. Where the independence and clique-cover numbers of G are denoted by α(G), χ(G) respectively, one has:

$$\alpha(G) \le \beta(G) \le \beta_1(G) \le \overline{x}(G).$$

Where an independent set is identified and there is a set of receivers but no mutual information, it gives rise to the first inequality above, and the last one is obtained due to the bitwise XOR of the vertices being broadcasted per clique in the clique-cover of G that is optimal.

Hinging on a greedy clique-cover (related to the bound), Reed-Solomon (RS) based protocols were proposed and the scalar capacity β1 and the framework of graph Index Coding were empirically analyzed and introduced.

$\beta_1 \leq (\overline{x})$

Based on a matrix rank minimization problem, a new linear index codes-based class was proposed by Bar-Yossef et al. in a paper that was a breakthrough. Minek, denoted the problem's solution where the optimal linear scalar capacity over GF was shown to have been achieved and was found to be superior particularly to the clique-cover method, i.e.:

$\beta_1 \leq \mathrm{Minrk}_2 \leq \overline{x}.$

In there was an extension of the parameter minrk2 to the general fields where it was seen through the Ramsey Theory arguments that a family of graphs is there for any ε > 0 on n vertices where β1 ≤ $n^{\varepsilon}$ whereas minrk2 ≥ n 1– ε for any fixed ε > 0.

In Alon et al. presented the first proof of a separation β < β1 for graphs; a new capacity parameter β ∗ has been introduced by the proof so that β ≤ β ∗ ≤ β1 and it has been shown that using a graph-theoretic characterization of $\beta^*$ the second inequality can be strict.

Further, Hypergraph Index Coding (as defined above the general broadcasting that also has problem of side information) was studied by the paper and for this there was the construction of numerous hard instances-ones where β = 2 whereas $\beta^*$ is unbounded, $\beta^*$< 3 whereas β1 is unbounded.

For graphs, a companion paper presented the first proof of a separation α < β; a new technique for bounding β from below has been used in the proof whereby a LP has been used where information inequalities are expressed by its constraints.

This separation is then amplified in the paper using lexicographic products whereby a sequence of graphs having the ratio β/α tending to infinity is yielded. In hypergraphs, an unbounded multiplicative separation between vector-linear and non-linear Index Coding too results through the same technique in which LPs are combined with lexicographic products.

The previous discussion makes it clear that there has been a high success rate in the prior Index Coding related work with respect to forming examples wherein separations between these parameters is exhibited and bounding the broadcast rate below and above by a number of parameters (however all of these are NP-hard to compute).

All the same general techniques whereby the broadcast rate β can be determined (not even to an approximation) for large classes of problems has not been successfully provided by this limitation has been clearly brought forth by the two facts that follow. Firstly, except for the graph for which there is a coincidence of the trivial upper and lower bounds $\chi(G)$ $\alpha(G)$, for every graph G, the exact value of $\beta(G)$ continued to stay unknown.

Secondly, where the trivial factor n (achieved by the broadcasting of all the n messages) can be improved by the approximation ratio of a polynomial-time algorithm, it was unknown as to whether the broadcast rate $\beta$ could be approximated by more than a constant factor 1.

The linear programming technique that was introduced in has been extended and applied in this chapter so that a series of new results can be obtained on Index Coding as also solving the two open questions that have been stated in the last few paragraphs.

The general problem of broadcasting with side information and the other one in which G is the graph have been discussed in detail in the two sections that follow:

$$\alpha(G) \leq \beta(G) \leq \beta_1(G) \leq \overline{x}(G).$$

$$\beta_1 \leq (\overline{x})$$

$$\beta_1 \leq \text{minrk}_2 \leq \overline{x}.$$

## 2.7.1. Multipath Routings

The reliability of end-to-end communications can be enhanced when the redundancy of wireless networks is leveraged whilst the transmission delay and control consumption get degraded.

Opportunistic routings, cluster-based multipath routing and braided multipath routing are the ones mainly included in multipath routing. The lifetime of the network can get extended and the energy efficiency improved effectively through multipath routing via cluster technology.

A single path (Single path) routing has been presented by Yin et al. where for the improvement of data transmission efficiency for wireless networks, this more deliberate scheme is combined with compressed sensing (CS).

Data is transferred to the base station (BS) from cluster head through joint multipath routing. A cluster-based multipath routing protocol (CBMRP) was proposed by Jena and Sharma whereby energy consumption was decreased utilizing the multipath and clustering techniques and the transmission reliability was increased. All the same, the CBMRP was essentially a single layer cluster.

The scalability was poor due to the single cluster's simple structure so that the application range got limited. The large computation complexity led to the transmission overhead increasing rapidly as a result of the inevitable large computation complexity.

Based on an ant colony optimization (ACO) and dynamic clustering a multipath routing protocol (MRP) was proposed so as to monitor burst events in a somewhat reactive WSNs.

The lifetime of the network was maximized and the energy consumption degraded by the MRP as a result of selecting cluster heads as per the residual energy, choosing a route dynamically to transmit data packets and the ACO algorithm searching the multiple paths.

All the same, the transmission delay was increased and the transmission reduced as it was essential to get for each node on the path the state information for the calculation of the pheromone. Unpredictable and unreliable wireless links can be handled well through opportunistic routings. A representative opportunistic routing protocol is ExOR.

Numerous opportunities are provided for each spread through the strategy of postponing selection whereby progress is made, thus enabling ExOR to make use of long radio links with link loss rates that were high.

The DN received a relatively high number of duplicate packets as a result of coordination mechanism and effective mutual recognition that was lacking between the various forwarding nodes.

To find a robustness guidance path during the route-discovery phase, a biased back off scheme was proposed via which for any pairwise nodes between two adjacent loops cooperative forwarding opportunities were provided.

The problem of data collision between two adjacent loops was however not dealt with well by reliable reactive routing enhancement (R3E) due to which the transmission gain of opportunistic routing got degraded. Node failures and link losses in WSNs are tolerated in braided multipath routing as it is a powerful scheme whereby energy-efficient and reliable transmission ways are provided by it.

Links or nodes are allowed to interleave over the transmission paths through braided multipath braiding whilst taking many partially disjoint alternation paths rather than completely disjoint paths.

For routing or data transmission with the iterative method, a novel shortest hop multipath (SHM) algorithm was presented by Yilmaz et al.

which generated the shortest hop braided multipath efficiently and without support of NACK/ACK, the packets are relayed to the sink by the nodes with less hops.

The addition of only one hop's marginal nodes into the network is helped at a network level by each iteration from the sink. Poor scalability results from the new network flooding that inevitably results when each time a new node intends to connect the WSNs.

The convergence of algorithms is affected due to frequent packet losses, which cause poor performance by SHM's construction due to high link losses thus caused.

A braided MRP that is reliable was developed by Wang et al. wherein the next-hop node is dynamically selected by the parent nodes as per the current link quality that is achievable so that at the same time reliability is guaranteed. The transmission overhead is effectively degraded by this scheme.

## 2.8. NETWORK CODING (NC)-BASED MULTIPATH ROUTINGS

NC in WSNs has been studied quite elaborately in the recent times. The received packets can be re-encoded or aggregated by the intermediate nodes through NC and therein lies its main contribution.

For data coding, a theoretically efficient method is provided by random linear network coding (RLNC). The origin packets that are received by the sink can have their encoding coefficients linearly correlated as the locality encoding coefficients of each intermediate node are generated independently in the RLNC scheme so that until the packet is routed to the sink, it should be re-encoded at each hop.

The decoding operation's computing complexity is the attached deficiency as a result of which the encoding vector results in the accompanying of extra transmission overhead.

A segment linear network coding (SLNC) scheme was proposed by Guo et al. The complexity involved in decoding reduces tremendously as a result of division into simple inverse operation of the complex matrix inverse operation and constraints being added to the encoding coefficients.

All the same, the reliability of the transmission cannot be ensured as a couple of issues like the packet loss rate are ignored by this method.

The lifetime of the network was prolonged whilst degrading of energy consumption and data traffic flow is aimed at through the proposed broadcast algorithm that uses NC for GBR GBR-NC.

The strategy of hop-by-hop retransmission is adopted by GBR-NC so that the reliability of transmission is guaranteed as in end-to-end communications this easily results in higher resource consumption and data redundancy.

Based on a novel moving window NC technique MWNCast, a cooperative multicast protocol was proposed whereby a better balance between transmission delays and throughput can be achieved, which at the same time without explicit feedback kept very low levels of decoding complexity and the probability of packet loss.

All the same, the window size negatively influenced the reliability of the transmission in this method so that choosing an appropriate window size continues to be a challenge. According to Jiang et al. a certain distribution that is random satisfied the network nodes.

NC techniques require a network topology and a multicast routing algorithm that is energy-efficient was presented by the authors whereby the throughput increases effectively. All the same, the computing complexity of such an algorithm is quite high.

Dynamic source routing (DSR) was developed by Johnson et al. The data packets could be delivered successfully due to the highly reactive service provided by the DST, which could quickly react to changes and had a low overhead.

Depending on the mechanism of automatic data transmission, the transmission efficiency got enhanced by the EARQ or the end-to-end automatic repeat request (ARQ) that was proposed. The transmission overhead to saw an increase along with the reliability of the data increasing through this retransmission mechanism.

The existing protocols and the NC-BMR that is proposed have certain differences, namely: the decoding success rate improves due to the new encoding matrix that is selected by the RLNC scheme that is based on data compression for guaranteeing the linear correlation of the encoding coefficients.

HMPN technology is for many-to-one pattern in addition to the end-to-end pattern which is one of the WSNs' popular communication patterns; with respect to parent nodes, optimum path width being selected so that a decrease in the overhead transmission and the transmission reliability is guaranteed; and the best main route is chosen dynamically and locally against the data retransmission by the parent node due to coordinating data forwarding that is HMPN-based.

## 2.9. CONCLUSION

To conclude this chapter, it has been observed that there are two types of network multicast models that are the basic model, and the line graph. In this chapter, the basic significance of algebraic framework has been discussed, along with the different phases of framework that are used in network modeling like start-up phase and running phase.

Towards the end of the chapter, it has been observed that Index Coding has received more attention in the recent interval of time by the application in the real-world and in part by its relation to the NC in very considerable amount.

In this chapter it has been discussed that how the index coding could be effective applied with the help of linear programming in multipath routing, along with the trusted network routings and intermediate nodes routing. This chapter also discussed the various types of multipath routings that are entirely based on NC. This chapter also provide highlights on the significance of information-theoretic frameworks.

## REFERENCES

1. Al-Najjar, H., & Rousan, N., (2012). *Trusted Network Coding Framework*. [Online] Ieeexplore.ieee.org. Available at: https://ieeexplore.ieee.org/document/6506542 (accessed on 3 May 2021).
2. Blasiak, A., Kleinberg, R., & Lubetzky, E., (2010). *Index Coding via Linear Programming*. [Online] Math.nyu.edu. Available at: https://www.math.nyu.edu/~eyal/papers/bkl-beta.pdf (accessed on 3 May 2021).
3. Li, Z., Xu, M., Liu, T., & Yu, L., (2019). A network coding-based braided multipath routing protocol for wireless sensor networks. *Wireless Communications and Mobile Computing, 2019*.
4. Rosvall, M., & Bergstrom, C. T., (2007). An information-theoretic framework for resolving community structure in complex networks. *Proceedings of the National Academy of Sciences,104*, 7327–7331.
5. Soljanin, E., & Fragouli, C., (2007). *Network Coding Fundamentals*. [Online] Citeseerx.ist.psu.edu. Available at: https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.185.2865&rank=1&q=Network%20Coding%20Fundamentals&osm=&ossid= (accessed on 3 May 2021).

# CHAPTER 3

# NETWORK CODING IN WIRELESS SYSTEM

## CONTENTS

The chapter provides a brief introduction to NC and the theory behind NC in wireless systems. It also includes various kinds of schemes that are used in NC, such as random linear network coding (RLNC), triangular network coding (TNC), an opportunistic network coding (ONC).

This chapter also highlights the history and theory behind the physical-layer NC. This chapter also addresses the challenges in physical-layer NC and benefits of the physical-layer NC. Last but not least, various applications of NC in wireless networks such as file download, instant messaging (IM), live broadcast, and video on demand, are emphasized.

## 3.1. INTRODUCTION

Network coding (NC) is a fairly recent subset of network information theory that has given space to immense advancements in the optimization of network throughput. It takes part in performing operations except just forwarding and replicating the nodes which constitute a network.

The theory behind Network Coding (NC), the various NCschemes which have been propose dand made use of over theyears, including the very emergenceof applying NC at the physical layer of networks,and varying selected applications of NC in the wireless net works.

It is important to note the developments in this field and examine its impact on wireless networks; in terms of both the improvements as well as advancement it has made, along with its concluding application to various categories within wireless networking.

In the past, the transmission of data through a network was solely viewed as a commodity flow, which is an exchange of commodities without the ability to process these commodities themselves.

NC changed this, suggesting operations which are more complicated than simple replication and the transmission of data packets, such operations could be performed at the nodes that make up a given network.

This led to rapid advancements and initiated the use of new mathematical tools, in fields such as algebra, matroid theory, geometry, graph theory, combinatorics, and optimization theory, among others. This resulted in NC as it is today: a complex field rich in mathematics.

While NC is a complicated topic to discuss without imperative mathematical background knowledge, in this introduction, the aim is to give a concise description of what NC entails of, and in doing so, throwing light on the various characteristics defining this technology.

In order to achieve this, some background is provided regarding NC and its underlying theory in this section, discussions on popular coding schemes that are currently in use and brief analysis of physical layer-based NC, focusing on physical network coding (PNC) since it is most used in wireless networks. Wireless networks have been made into a design by using wired networks as its blueprint. The design is abstract towards the wireless channel as a point-to-point link, and grafts wired network protocols onto the wireless environments.

For example, routing uses the shortest path protocols, routers forward packets but does not modify the data, and reliability still relies on the retransmissions. The design has worked well for wired networks, but even less so for the unpredictable and unreliable wireless mediums.

The wireless medium is primarily different. While wired networks have reliable and predictable links, wireless links have a higher bit error rate (BER), and their characteristics could vary over very short time scales.

Furthermore, wired links are unicast links, but the majority of wireless links (with Omni-directional antennas) are broadcast links. Transmissions in a wired network do not interfere with each other, whereas the interference is a generic case for the wireless medium.

Wired nodes are commonly static, while wireless was built in order to support mobility as well as portability. The wired network design is conflicting with the properties of the wireless medium. As a result, current wireless networks suffer a low throughput, dead spots, and inadequate mobility support.

The characteristics of wireless networks may all seem disadvantageous at first sight, but a newer perspective reveals that some of them can be used to our advantage, albeit with a fresh and new design.

The broadcast nature of wireless provides an opportunity to deal with unreliability; when a node broadcasts a packet, it is likely that at least one nearby node is on the receiving end, which can therefore function as the next-hop and forward the packet.

This is in stark contrast to the present wireless design, where there is a single designated next-hop, and when it does not receive the packet, the previous hop has to retransmit it. The property is called spatial diversity and has been explored further in the text provided.

Interestingly, wireless networks exhibit significant data redundancy, i.e., there is a large overlap in the information available to the nodes. First, as a packet goes through multiple hops, its content becomes acquainted to many nodes.

Further, wireless broadcast amplifies this redundancy because at each hop, it delivers the same packet to multiple nodes within the transmitter's radio range. Here the question arises, can an alternative design of wireless networks exploit their intrinsic characteristics, such as spatial diversity and data redundancy, rather than foisting an artificial wireline abstraction? NC may be the answer.

This chapter would further explore the case for NC as a design platform for wireless networks, diving more into how NC can improvise the throughput, reliability, fairness, and management of wireless networks.

## 3.2. DEFINITION OF NETWORK CODING (NC)

NC is a networking technique where operations, which in practice tend to be algebraic algorithms, are performed on data while it passes through nodes within a defined network.

While, in theory any manner of algorithm could be performed on the data at a node, current NC algorithms tend to be more concerned with gathering the various transmissions which pass through a given node. In traditional routing networks, packets are simply cached and there on forwarded to the next node downstream in the network.

As such, if a routing node receives two packets from two distinct sources, it will forward them in sequence, even if they are both addressed towards the same destination, while filling in any other ones it may receive in the meantime.

This results in the node formulating separate transmissions for each and every message being delivered, which in turn results in a decrease in the network efficiency. NC is used to lessen this by merging relevant messages at the relay node, using a given encoding and then forwarding this accumulated result to the destination for decoding.

In order for this to work properly, the destination node (DN) needs to be synchronized with the transmitting nodes, a constraint especially vital when it comes to NC done at a physical layer.

## 3.3. THEORY BEHIND NETWORK CODING (NC)

In the introduction defining the beginning of NC research, a multicast session over a directed graph with lossless links was considered. The results are expressed as the max-flow min-cut theorem in network theory.

This theorem states that in a flow network, where information flows from one node to the other, the maximum amount of flow from the source to the sink (max-flow) is equivalent to the minimum capacity that would not allow any flow to pass from the source to sink when cut/removed from the network in a specific manner (min-cut).

When operations at intermediate nodes were allowed to pass, the maximum multicast rate was equivalent to the minimum min-cut from the source to each receiver. Basically, if all receivers have the same min-cut from the source, then NC would allow all nodes to achieve the min-cut capacity concurrently.

This capacity would also be corresponding to the maximized flow rate which each receiver gets if it were the only receiving node in the network. The simplest example which demonstrates the key idea as well as the benefits of NC in the multicast case is the butterfly network depicted in Figure 3.1.

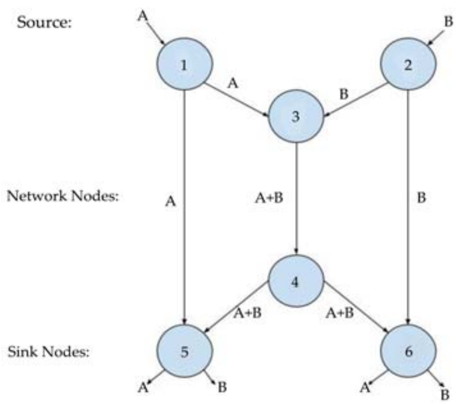



**Figure 3.1:** Message transmission through different nodes.

*Source: Image by Network Coding Meets Multimedia: A Review (Magli, Wang, Frossard, and Markopoulou, 2012).*

Two of the source nodes (SNs), 1 and 2, have messages A and B respectively, which both need to be transmitted to the two destination/sink nodes, 5 and 6. Each edge in this network can only carry a single message.

If nodes could only route/retransmit, then the central link would only be able to carry message A or message B, but not both of them at the same time.

Assuming a person sends message A across the central link between nodes 3 and 4, then node 5 would receive message A twice and would never receive message B. The same problem would occur at node 6 if a person sends message B through the central link, with node 6 never receiving message A. One can immediately see here that routing would be insufficient to transmit both messages since no routing scheme can concurrently transmit A and B to both sink nodes. This is where the operation on data at relay nodes comes into play; a simple linear code is demonstrated here, where A and B are encoded using their sum. Similarly, in this example, node 5 would receive both A and A + B, from which it can decode B by subtracting these two values. Node 6 would go through the same procedure to decode A after receiving both B and A+B. From this simple example, it can be observed that various other encoding techniques could be applied to a varying number of packets, in varying network configurations.

These techniques could exceedingly increase the amount of information which could be transmitted in a single instance, thus significantly improvising the throughput and power efficiency of networks which implement them. This result can also be applied to wireless networks with two simultaneous unicast connections, as shown in Figure 3.2.

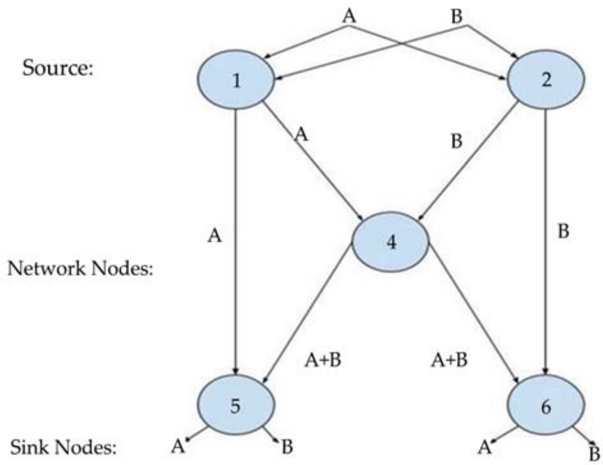



**Figure 3.2:** Wireless networks with two simultaneous unicast connections.

*Source: Image by Network Coding Meets Multimedia: A Review (Magli, Wang, Frossard, and Markopoulou, 2012).*

The modified wireless butterfly network shown in Figure 3.2, is differing from the original butterfly network in the sense that packet transmissions can be transmitted from the SN to more than one node. Therefore, transmissions are represented through hyper-arcs, instead of just arcs. These examples give a brief and fairly accessible overview of the theory behind, and its nature of NC.

## 3.4. NETWORK CODING (NC) SCHEMES

The manner in which nodes encode and decode the packets which they transmit/receive can have a huge and long-lasting impact on the resulting throughput of the network. The majority of the NC schemes in use in this day and age have their basis in algebraic theory.

Whereas earlier schemes, such as the conventional XOR-coding scheme and the deterministic linear network coding (LNC) scheme were destined in nature, the more common schemes in use today are non-deterministic, which means that they are free from the constraints of having packet feedback information for each and every transmitted packet from all its receivers.

In this section, different coding schemes such as random linear network coding (RLNC), triangular network coding (TNC) and opportunistic network coding (ONC) are discussed in detail.

### 3.4.1. Random Linear Network Coding (RLNC)

Here, application or use of a network model for a general communication system that consists of a source, network, and sink nodes connected via channels which are potentially lossy, is given.

Such a system can be represented as a directed graph $G = \{V, E\}$ where the vertices $V$ represent the various nodes in the network and the set of edges $E$ consist of arcs between the nodes and denote links in the network. In RLNC, coded packets are random linear combinations of the original packets over a finite field with size $q$. Each coded packet is in the form:

$$\sum_{i=1}^{i=k} \alpha_i \times P_i$$

Here; $\alpha_i$ and $P_i$ are random coefficients chosen from a Galois field and the packet respectively; while $k$ is the number of packets which are coded together in that specific single transmission. A Galois field is any field that contains a finite number of elements.

If the field size is large enough, the probability that the receiver(s) will obtain linearly independent combinations approaches 100%. Linear independence is imperative in ensuring uniform distribution, thus reducing unnecessary information being forwarded to a given node.

The smaller the field size is, however, the less complex the resulting system is, which is especially preferable in wireless networks. Instead of transferring the original packets, the SN is generating and transmitting randomly-coded packets over the *k* packets.

In order to decode these coded packets and retrieve the original ones, the DNs need to receive *k* linearly independent coded packets. They can use Gaussian elimination in order to decode these coded packets.

As power-packed as this encoding scheme is, it does have its disputes, most important among them is the fact that if a receiver gets an insufficient number of packets, it is exceedingly unlikely that it can recover any of these original packets.

This phenomenon is known as the index coding problem. It can be fixed by sending through additional random linear combinations until the receiver gets the required number of packets. Other major challenges include the high decoding computational complexity due to using the Gauss-Jordan elimination method as well as the high transmission overhead due to adding large coefficient vectors to encoded blocks.

LNCcomputational complexity makes it unsuitable for practical use in devices that operate on battery power, such as mobile phones and wireless sensors.

### 3.4.2. Triangular Network Coding (TNC)

In order to fundamentally fix the problem with RLNC detailed above, where receivers that get an insufficient number of packets and cannot recover the original packets, TNC was proposed thereon.

The triangular pattern-based packet coding scheme is performed in two stages. First, redundant bits are categorically added to the head and tail of each packet to make sure that all packets are of uniform bit length. The packets then are XOR-coded bit by bit where the other bits are added in such a way that they generate a triangular pattern which is known as triangularization, as shown in Figure 3.3.

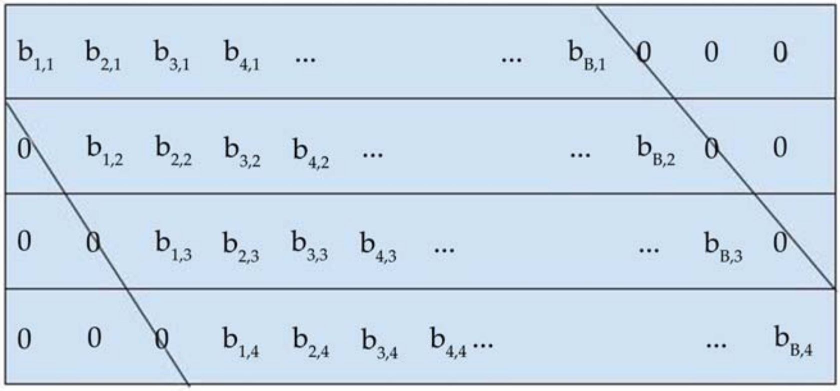



**Figure 3.3:** Triangularization.

*Source: Image by Optimal Solution for the Index Coding Problem Using Network Coding (Qureshi, Foh, and Cai, 2012).*

The decoding process has similarities to the LNC decoding process, which involves Gaussian elimination. However, in this specific case, it is simplified since the coded packets are in a triangular pattern, and as such, the receiver only needs to perform back-substitution, where each row is solved from the last to the first one.

This is far less computationally intensive and gives triangular coding the bandwidth performance of LNC, while being able to afford the computation cost of XOR coding.

### 3.4.3. Opportunistic Network Coding (ONC)

In ONC, the sensor nodes can snoop on all transmissions in their neighborhood and store the overheard data packets, whether or not they are intended for them. As such, the sensor nodes know the overheard and routed packets that each neighboring node possesses, and can perform NC operations based on this very information.

Each node has its own queue of received uncoded packets $p_1$, $p_2$..., $pv$ whose destinations which are determined by their packet headers and the node's routing table, are $r_1$, $r_2$..., $r_n$. The node thereon dequeues the first packet, $p_1$ and, while making sure all of the next-hop recipients can instantly decode the resulting combination, steps throughout the queue to willingly add packets for combinations.

A recipient can immediately decode a combined packet if it knows all but one uncoded packet. For example, a combined packet $p_1 + p_2 + p_3 + p_4$ is valid if node $r_1$ knows $p_3$and $p_4$, node $r_3$ knows $p_1$ and $p_4$, and node $r_4$ knows $p_1$ and $p_3$.

Being familiar with some of the common coding schemes in use helps us see that most are geared towards packet-based networks. In wireless networks, however, there are a significant number of gains to be made at layers besides the application layer, which is where the bulk of NC is done at present.

### 3.4.4. Physical Layer Network Coding (NC)

Physical-layer NC conceptually was proposed in 2006 for application in wireless networks. The bedrock idea is to take advantage of the natural NC operation, which occurs when electromagnetic (EM) waves are overlaid on one another.

This simple idea led to numerous developments, with subsequent works by different researchers, which leads to many new results in the various domains of wireless communication, wireless information theory, as well as wireless networking. An attempt to give a brief overview of both the theory behind this concept and the implications of the results in the three fields are listed above.

## 3.5. THEORY BEHIND PNC

Interference is often treated as a destructive concept in wireless communication networks till date. While multiple transmitters send radio waves to their respective receivers, a receiver may get signals from its transmitter as well as from various other transmitters at that very time.

The radio waves are treated as interference which corrupts the intended signal, as a result of their overlapping. In Wi-Fi (wireless fidelity) networks, for instance, when multiple nodes transmit together, packet collisions can occur, which leads to none of the packets being received in the correct manner.

PNC takes advantage of the fact that when multiple EM waves come together within the same physical space, they add or superimpose, therefore increasing in amplitude. The mixing of EM waves is a form of organic NC, which is utilized in PNC.

The concept of PNC can be easily shown within a network model known as a two-way relay channel (TWRC). TWRC is a three-node linear network in which two end nodes, nodes $T_1$ and $T_2$, want to communicate via a relay node $R$. This is illustrated in Figure 3.4.


**Figure 3.4:** Two-way relay channel.

There is no straight signal path between nodes $T_1$ and $T_2$. A real-world example of such a system is a satellite network in which nodes $T_1$ and $T_2$ are the ground stations, and the relay $R$ is the satellite. The half-duplex constraint is often forced on wireless communication systems in order to ease engineering design.

With the half-duplex constraint, a node cannot therefore transmit and receive at the same time. The relay in TWRC thus cannot receive from node $T_1$ or node $T_2$ and transmit towards them at the same time.

This means that each packet from node $T_1$ to node $T_2$ (and similarly, each packet from node $T_2$ to node $T_1$) must use at least two time slots to reach its destination. Therefore, the best possible packet exchange throughput is two packets for each of the two slots, one in each direction.

Hence, it can be said that the use of natural NC, through taking advantage of the superimposition of waves could indeed be a major breakthrough. It is however subject to certain challenges.

## 3.5.1. Advantages and Challenges in PNC

PNC can possibly achieve 100% and 50% throughput increases compared with traditional transmission and straightforward NC, respectively, in multi-hop network settings. PNC achieves this doubling of the TWRC throughput by reducing the needed timeslots, for the exchange of one packet from four to two.

Unfortunately, PNC is yet to be in widespread use in wireless networks, since in a general multi-hop network, MAC-layer, and network-layer issues take on an increasingly vital role, particularly with regards to the management of complexity when there are many simultaneous flows in the same network.

One of the important issues in PNC is how to deal with the asynchronies between signals transmitted simultaneously by two end nodes. Symbols transmitted by these two end nodes could potentially arrive at the receiving end with symbol misalignment as well as relative carrier-phase offset.

This could result in significant penalties to perform. Reliability of transmission is also one of the challenges which face PNC. Channel coding is typically used to solve this issue, and its application to PNC is the subject of research today.

With this in mind, PNC is the future of NC in wireless networks, and therefore, it is important to understand it. Having given this overview, now some of the applications that have been realized using NC in wireless networks in the next section are pointed out further.

## 3.6. APPLICATIONS OF NC TO WIRELESS NETWORKS

It is improbable that the incorporation of NC at the physical layer will be practical in the near future, for various reasons. It is however fairly feasible to build NC into overlay networks. In overlay networks, nodes are applications running in computers, and edges are the transport-level connections between computers.

Overlay networks can be infrastructure-based, as shown by content distribution networks like Akamai. They can be ad-hoc or peer-to-peer (P2P) networks of end hosts temporarily linked together in order to perform a particular communication task, for example, file download, live broadcast, media on-demand, instant messaging (IM), conferencing, or gaming.

A quick review of such applications of NC, as well as applications to wireless and sensor networks are further explained.

### 3.6.1. File Download

Downloading files from a server to a client's computer is one of the most laymen tasks which occurs in network communication. While the downloaded file is traditionally unicast from the server to the client, if people ignore delay, this can also be seen as a multicast of the file from the server to a large group of clients using a relatively large amount of buffering.

Using this simplification, one can see that the utilization NC would potentially increase the throughput and therefore reduce the average downloading time. Consider this, for example, downloading a file over a

P2P network comprising all nodes which are currently downloading the file.

In this model, freshly arriving nodes join the network by connecting to a subset of the already existing nodes. Accordingly, the difficulty of P2P content distribution is searching for an optimal block scheduling algorithm, which should ideally minimize the file downloading time in a distributed manner.

A real-life example is BitTorrent, the original, and still the most famous protocol for P2P file downloading. In BitTorrent, this file is evenly distributed into *h* pieces. Each node negotiates to get pieces of the file from its neighbors, unless the node obtains all *h* pieces and can leave the network.

After a node acquires a fresh piece, it announces this acquisition to its neighbors, so that every node knows which pieces every neighbor has. Nodes typically request a local rarest block, which is a block which is least common amongst all of the node's neighbors.

Without NC, each node would have to decide which of the blocks to download based merely on the information it could obtain from its neighboring nodes. This could lead to unproductive downloading blocks, since blocks which are rare in its immediate neighboring nodes are not necessarily rare blocks when considering the entire network.

However, with NC, as coded blocks are aggregates, all coded blocks are almost equally useful to any node, and as such, there is no need to locate and request global rarest blocks in the network. This prevents an information clog and decreases the file downloading time.

### 3.6.2. Video on Demand, Live Media Broadcast, and Instant Messaging (IM)

Video on demand can be looked at as a specialized form of file download where pieces of the downloaded file have to appear in order and should be decoded in real-time, taking into account some trivial delay.

NC can be applied in this case by breaking the file into chunks, which can be downloaded in a sequence. A similar technique can be used with live media broadcast. Earliest decoding is used to further reduce delay.

IM is also similar to live broadcast, but with a lower bit rate, hence only text is sent usually. Text messages are also often sent in bursts or clumps and have less strict delay constraints. However, IM now frequently includes a larger bulk of messages such as images and audio clips.

Flooding, which is usually used for IM in P2P networks, is inefficient while used on larger files. NC can be applied in all of the cases in order to improve efficiency in overlay networks.

## 3.7. WIRELESS MESH NETWORKS (WMNS)

Apart from an application-level overlay network, a second convenient place where NC can be applied is a link-layer network such as a wireless mesh network. Mesh networks consist of mesh routers, which supply access to an existing infrastructure, and mesh clients, which both provide multi-hop connectivity to the mesh routers and utilize the connectivity provided by other mesh clients.

It has already been observed how the numerous transmissions required for the two wireless nodes to exchange packets through an intermediate node is reduced from four to three using NC. This can be further extended to varying hops.

If a wireless node *a* sends a packet stream to a wireless node *b* over a series of hops, then *b* can send an equally rated stream of packets in the reverse direction for free. This means that additional transmissions on the intermediate hops would not be necessary. This could lead to significant savings in the total amount of data that needs to be transmitted.

## 3.8. WIRELESS SENSOR NETWORKS

For the final application of NC, it is best to consider wireless sensor networks, in which very small sensors can be spread onto surfaces, and harvests energy from the environment in order to formulate sensing surfaces on a built-in communication network.

Sensor nodes are primarily equipped with a radio transceiver, a microcontroller, a memory unit, and a set of transducers with which they acquire and process data with. To reduce each node's size and energy requirements, the transceiver's oscillator is replaced by an on-chip resonant circuit.

However, the center frequency of the resonant circuit is random, which means that each node picks a random channel on which to transmit, and a random channel on which to receive. The nodes can self-organize themselves to form a multi-hop network and transmit the data to a sink node.

The throughput between any of the two nodes is consistent only if routing is used, but grows in a linear manner in the number of channels if NC is used, and the radio ranges are chosen ideally. The reason NC which helps greatly here is that the randomly mixed packets can then find their way towards the destination without the need for explicitly informing nodes where their destinations are. This is especially useful since explicitly identifying routes is difficult when graph connectivity is not known.

In this section, some of the areas where NC can be applied in wireless networks and can broadly classify its benefits resulting in improvement in a network's throughput, efficiency, and scalability, are discussed. Further benefits include improving resilience to attacks and eavesdropping as a side effect of the encoding schemes used.

In wireless communication networks, network performance is massively affected by varying network characteristics, such as limited channel bandwidth, an unstable signal transmission, a serious power constraint, high node unreliability, and eased interception of wireless signals.

NC has lately emerged as a new coding paradigm which has demonstrated a wide range of potential applications for improvising network performance in wireless communication networks.

The core notion of NC is to permit the information (or data) received from multiple links to be mixed at intermediate network nodes for following transmissions so that the amount of data in the network is lessened and the network performance in terms of network throughput is improvised.

This notion can also be applied to the data received on a single link within one stream or across differing streams, and even to the physical layer, where different signals can be used for further transmission.

In contrast to traditional network operations that try to keep away from crash of data streams through resource management, this primary principle not only paves a new way to improvise network throughput but also brings a bulk of other surprising benefits, such as energy efficiency, network robustness, and network security.

Due to its wide range of potential applications, NC has recently received an increase in attention from the research commune. To better take advantage of this promising coding paradigm in wireless communication networks, various technological issues have to be studied further, which revitalizes a sizable amount of research activities in the area.

In the paper "*Bounds on the Throughput Gain of Network Coding in Unicast and Multicast Wireless Networks*," Liu et al. studied bounds on the throughput gain of NC in unicast and multicast wireless networks.

For any irregular networks of dimensions under either the protocol or physical communication model which was introduced by Gupta and Kumar, it showed that NC and broadcasting lead to at-most a consistent factor improvement in per-node throughput.

For the protocol model, they provided bounds on this very factor. They also established boundaries on the throughput benefit of NC and broadcasting for various source multicast in random networks. For an arbitrary network deployment, they showed that the coding benefit ratio is at most O(logn) for both the protocol and physical models.

The results provide further guidance on the application of NC, and more generally indicate difficulty in improving the escalating behavior of wireless networks without modification of the physical layer.

Wireless networks have been designed using the wired network as the blueprint. The design abstracts the wireless channel as a point-to-point link, and cuts wired network protocols onto the wireless environments. For example, routing uses the shortest path protocols, routers forward packets but does not modify data, and reliability is dependent on retransmissions.

The design has worked well for wired networks, but less so for the undependable and unpredictable wireless medium. This is fundamentally different. While wired networks have both reliable and predictable links, wireless links have a higher error rate, and their characteristics vary over short time scales.

Furthermore, wired links are unicast in nature, but the majority (with Omni-directional antennas) are broadcast links. Transmissions through a wired network does not interfere with one another, whereas interference is in commonality for the wireless medium.

Wired nodes are usually static, while wireless was built to support potency and transferability. The wired network design conflicts with the features of the wireless medium. As a result, the current wireless networks go through low throughput, dead spots, and inadequacy in mobility support.

The features of wireless networks might all seem to be at a disadvantage at first sight, but a newer perspective reveals that some of them can be used to our advantage, even if with a fresh design. The broadcasting nature of wireless provides an opportunity to deal with unreliability; when a node broadcasts a packet, it is likely that at least one nearby node receives it, which can therefore function as the next-hop and forward the packet.

This is in sharp contrast to the present wireless design, where there is a single designated next-hop, and when it does not receive the packet, the previous hop has to retransmit it. The property is called 'spatial diversity.'

Interestingly enough, wireless networks exhibit a significant amount of data redundancy, i.e., there is a large overlap in the information available towards the nodes. Firstly, as a packet travels multiple hops, its content becomes known to many nodes.

Furthermore, wireless broadcast intensifies this redundancy due to each hop it delivers. The same packet delivers to multiple nodes within the transmitter's radio range. Can an alternative design of wireless networks expose their intrinsic characteristics, such as spatial diversity and data redundancy, instead of foisting an artificial wireline abstraction?

NC can possibly be the answer. For instance, redundancy can be misused in order to compress data, increasing the information flow per transmission, and thus improvising the overall network throughput.

## 3.9. OPPORTUNITIES

The following simplified examples revolve around the same mainstream theme, which covers the potential benefits of building future wireless networks around NC.

### 3.9.1. Throughput

The throughput of today's wireless networks leaves a lot to be desired; it is outlined here how a wireless architecture designed around NC can help improve throughput. The intuition is that NC increases wireless throughput because coding allows the routers to compress the transmitted information given what is known at various nodes.

By matching what each neighbor has with what another neighbor wants, a router can deliver multiple packets to different neighbors in a single transmission. This style of coding is known as inter-flow NC because the coding is done over packets that differ in their next hop, and thus from different flows.

It is an important extension because in a real wireless network, there might be only a small number of flows crossing the reversing path of each other a la Alice-Bob, but one would expect many flows to intersect at a relay, and therefore can be coded together.

For more general topologies, COPE leads to larger bandwidth savings which are apparent from the above example. It can XOR more than a pair of packets and produce a multifold increase in the throughput. To summarize, COPE is a MAC extension that has two components:

1. **Opportunistic Listening:** COPE uses the shared nature of the wireless medium which, for free, broadcasts each packet in a small neighborhood around its path. Each node stores the packets it overhears for only a limited period. It also tells its neighbors which packet it has heard by explicating the packets it sends through.

   This creates an environment that is accepting to coding because nodes in each area have a huge and partially overlapping reservoir of packets, they can use in order to decode.

2. **Opportunistic Coding:** When a node transmits a packet, it utilizes information based on what its neighboring nodes have heard so as to deliver various packets in a single transmission. The node XORs multiple packets, when each of the intended next-hop has the correct amount of information in order to decode the encoded packet. Even more precisely, each node uses the following coding rule:

To transmit n packets, $p_1,\ldots, p_n$, to n next hops, $r_1,\ldots, r_n$, a node can XOR the n packets together only if each next node has all n – 1 packet except the packet it wants.

Theoretically, COPE leads to reduction of the number of transmissions by a factor of two, and therefore should duplicate the throughput. Katti et al. have shown, however, that in practice the throughput gain is much more expensive.

Experimental throughput gain passes through the theoretical gain because COPE eases hot-spots in a network. Categorically, in an ad hoc network, most paths intersect at the center. As a result, nodes central to the network experience congestion, build queues, and drop packets.

The dropped packets have consumed bandwidth in order to reach the center of the network. Dropping them centrally reduces network resources and significantly reduces the overall throughput. Contrasting with coding, congested nodes placed central to the network have the opportunity to send various packets in one single transmission, allowing them to drain the queues quicker and thus avoid dropping packets.

### 3.9.2. Content Distribution

Many people today like to work in coffee shops, while listening to music. Imagine a scenario in which a music store like iTunes partners with a popular coffee store like Starbucks in order to provide music hot-spots.

A person can listen to any song over WIFI. iTunes and Starbucks may charge customers per song or provide the ability to hear personalized music as a value-added service. Bandwidth usage efficiency will be integra to scale services for other people.

Consider a scenario where x and y are customers in Starbucks, where each of them has some song on their laptop, but are currently interested in listening to a song that they do not have. They hence contact the access point in Starbucks, which has access to the iTunes music inventory.

Each customer requests an access point for the song they want to download, and also tells it the list of songs he/she already has. Say that Q wants song sA and P wants song sB. Q however happens to have sB on her laptop, and P happens to have sA on his laptop.

Instead of sending separate streams to Q and P, the access point can use NC to compress the information and improve bandwidth utilization. Specifically, to deliver packet pA to Q and pB to P, the access point XORs the two packets and broadcasts the XOR-ed packet.

Since Q has the song P is downloading, she can use the meta-data in the XOR-ed packet's header to locate the packet P is playing, XOR it with the received packet, and obtain the packet she wants. P can do the same to remove Q's packet from the XOR-ed packet and obtain the packet he wants.

In this example, NC doubles the throughput. Though the coding techniques used in this section are similar to COPE (in the above section), it extends the work towards a fresh application. Specifically, while COPE requires a multi-hop network, this application runs on WLAN.

Further, COPE exploits data redundancy at the network layer, i.e., it is exploitative of the fact that the same packet is known to multiple nodes since it travels along multiple hops. In contrast, the data redundancy is exploited at the application level. The objective here is to show that NC is a very versatile technique which can be customized to differing applications.

### 3.9.3. Reliability

The primary means for ensuring reliability in the existing structure is retransmission of the lost packets. This works well in wired networks where the error rate is extremely low, but is ineffective over error-prone wireless channels. This section presents a few accepted examples displaying how NC provides a more efficient approach to reliability. In contrast to §III-A, the examples in this section take up intra-flow NC, i.e., routers mix packets heading in the same destination. As a result of this mixing, every received packet contains a certain amount of information about all packets in the original file, and therefore, no coded packet is special. Put differently, without coding, a transmitter needs to know which exact packets the destination (or the next-hop) misses so that it may retransmit them. When the network is unreliable, the communication network for this feedback reliably consumes a significant bandwidth. In the presence of coding, no specific packet is essential, and as a result, a transmitter need not learn which particular packet the destination misses, it only needs to get feedback from the destination once it has received enough packets in order to decode the whole file. The reader may have noticed that the above applies to erasure-correcting coding which is applied at the source too. In fact, source coding is just a special case of intra-flow NC, where the source is the only node allowed to mix the packets in the flow. There are however, benefits to allowing other nodes to perform such mixing, as seen in the examples below. Finally, it is important to note that a natural synergy exists between reliability and throughput. Claiming that the NC provides efficient reliability means that for the same level of reliability, it shows the network achieving higher throughput (Figures 3.5(a)–(c)).

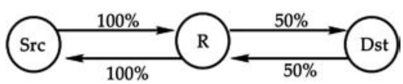



**Figure 3.5(a):** Network coding allows R to improve reliability with low complexity.

*Source: Image by Wireless Network Coding: Opportunities and Challenges (Christina Fragouli et al., 2007).*

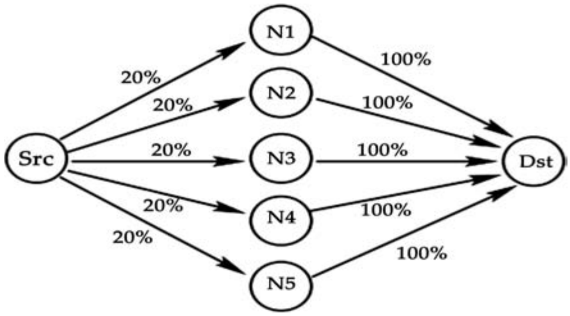



**Figure 3.5(b):** Network coding removes the need for the middle nodes to co-ordinate.

*Source: Image by Wireless Network Coding: Opportunities and Challenges (Christina Fragouli et al., 2007).*

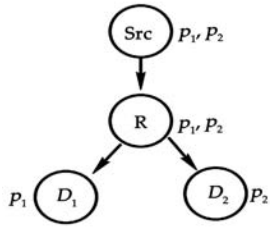



**Figure 3.5(c):** Network coding allows multiple receivers to efficiently recover lost pockets.

*Source: Image by Wireless Network Coding: Opportunities and Challenges (Christina Fragouli et al., 2007).*

### *3.9.3.1. Single Path*

Consider the easily explained example in Figure 3.5(a), where the labels on the links refer to their delivery rate. The source has a perfected link to the router R. In contrast to it, R's link to the destination has a probability of 50% in both directions.

If one wants to transfer a n-packet file, the transmission from the source to node R is perfect and so focusing on the cost of delivering these n packets from node R to the destination. 802.11 unicast in today's wireless networks will require every packet to be retransmitted unless it is successfully received at the sender, for an expected number of 4 transmissions per packet, and 4n packets for the whole file, which is far greater than the lower bound of 2n + 2 transmissions (for the n packets and 1 acknowledgment) for the transfer.

Ideally, one would want a scheme that achieves this performance with zero complexity. A possible non-coding scheme could proceed in rounds, where the sender first transmits all n packets, waits for a batched acknowledgement from the receiver detailing which packets it has received, and then proceeds similarly with the packets that were not successfully received.

This scheme requires 2n + 2 log2 n, which, while far from ideal, is a significant improvement over the current scenario. Unfortunately, this improvement is achieved at the cost of increased protocol complexity, as well as larger sized acks from the receiver.

NC, however, gives the ideal solution in this case and that too with very low complexity. Consider this an approach where R transmits a random linear combination of the packets. In particular, R transmits to the destination packets of the form $p_i = c_{j1}p_1 + \ldots + c_{in}p_n$, where $p_j$ is the jth packet in the file, and $c_{ijs}$ are random coefficients that R picks.

The destination awaits to receive n such coded packets and recovers the original packets in the file, as follows:

$$\begin{pmatrix} p_1 \\ \vdots \\ p_n \end{pmatrix} = \begin{pmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \\ c_{n1} & \cdots & c_{nn} \end{pmatrix}^{-1} \begin{pmatrix} p'_1 \\ \vdots \\ p'_n \end{pmatrix}$$

- Once the destination has decoded these packets, it immediately sends R one acknowledgment for the whole file. Since the ack on average needs two trials, and assuming R continues transmitting during that period, R will deliver this file to its destination in 2n + 2 transmissions, on an average.
- Dead Spots: NC also assists in helping understand dead spots. Consider the scenario in Figure 3.5(b), where say P would like to transfer a file to R. P is not within R's radio range, and thus P needs the help of an intermediate node.
- There are five nearby wireless nodes which could relay P's packets to R. Unfortunately, P is in a dead spot with 80% loss rate to every nearby wireless node. In today's 802.11 networks, P will pick the best path to R for her transfer. But since all paths are lossy, each packet will have to be transmitted six times (5 times from P to the relay and once from the relay to R.

A better approach would make use of spatial diversity to improve P's throughput. P broadcasts her packets, and any relay that hears a packet can forward it to R. In this case, the probability of delivering a packet to the relay increases from 20% to (1 – 0.85) × 100 ≈ 67.2%. Thus, on average, a packet is transmitted 2.5 times (1.5 times from P to the relays and once from the receiving relay to R. This increases P's throughput by 2.4x.

But using diversity without coding may create a new set of problems. Multiple relay nodes can hear the same packet and thus attempt to transmit it to R. This creates spurious transmissions and wastes the wireless bandwidth. The combination of spatial diversity and NC solves this issue. P broadcasts her packets. Any relay can participate in forwarding P's packets to R.

To do so, the relay creates a non-specified linear combination of the packets it has received from P so far. Specifically, a relay transmits to R packets of the form $p_i = c_{ijpj}$, where pj is one of P's packets, and cij's are random coefficients. If the file contains n packets, then R can decode the whole file after receiving any n coded packets, using the matrix inversion in 1. Once R decodes the file, he immediately broadcasts an acknowledgment for the whole transfer, causing the relays to stop their transmissions.

- Multicast: Consider the example in Figure 3.5(c), where the source wishes to multicast a video stream to nodes D1 and D2. Say for instance source transmits packets p1 and p2 to the router R, which broadcasts the two packets to the destinations, since wireless receptions at different nodes are highly independent, it is possible that when R broadcasts the two packets, node D1 receives only p1, while D2 receives only p2.

In this case, node R has to retransfer both packets in order to allow the destinations to recover their respective losses. But, if node R is given permission to code, then it can XOR the two packets (i.e., p1 $\oplus$ p2), and broadcast the XOR-ed version on the wireless medium.

This individual transmission gives space to both the destinations in order to recover their corresponding losses and at the same time providing efficient reliability. Ghaderi et al. have studied this problem in theory and have shown that NC improvises the expected number of transmissions by a factor Θ (log K log cijs K), where K is the fan-out of the multicast tree.

MORE has been developed, which is an opportunistic routing protocol that exploits intra-flow NC. The first protocol, known as Srcr, distributes the traffic to multicast destinations along the edges of a tree rooted at the same source.

The second protocol, called ExOR, exploits the broadcasted nature of the medium in order to deliver a packet to various nodes simultaneously (Figure 3.6).

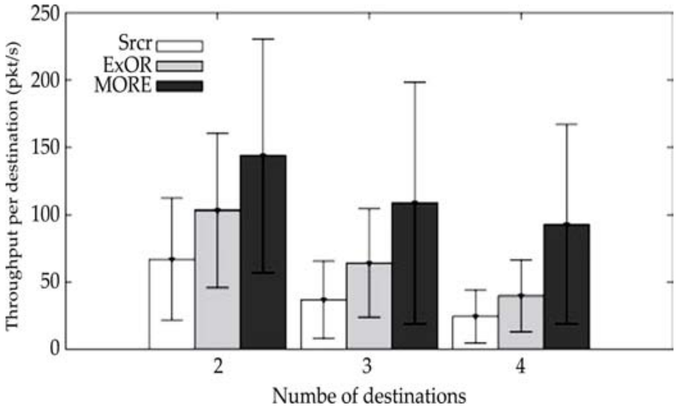



**Figure 3.6:** The per-destination multicast throughput of MORE, ExOR, and Srcr (thick bars= average per-destination throughput taken over 40 runs with different nodes and thin lines= the standard deviation).

*Source: Image by Wireless Network Coding: Opportunities and Challenges (Christina Fragouli et al., 2007).*

Figure 3.6 shows the average per-destination throughput in a 20-node wireless testbed. In this very scenario, the source multicasts a file in order to reach a variety of destinations. As expected, the per-destination average throughput decreases with again in the number of destinations.

However, Figure 3.6 depicts that MORE's throughput gain increases with an increase in the number of destinations. Intra-flow NC, and specifically MORE, has 35–200% throughput gain over ExOR and 100–300% gain over Srcr.

### *3.9.3.2. Fairness*

An important characteristic of wireless networks is its time variability and that of the received signal quality due to interference and attenuation effects (fading). These random fluctuations at the physical layer are looked at as packet erasures at higher layers and might result in variability of the reception rates over short period spans.

Real-time applications, on the other hand, tend not to tolerate them too well, and there is an integral research body in order to deal with such effects.

Moreover, such changeability may also require a scheduled scheme in order to ensure fairness over a short span of time.

Finally, reduction of the variability of packet delivery may also serve to only to lessen the problem of windows closing, for instance in TCP. It is important to note here that combining NC and broadcasting in such an environment allows us to naturally "smooth over" rates which receivers experience over a short span of time and relieve these problems as well.

## 3.10. CODING CHALLENGES

The coding challenges arise from the desire to combine several attractive properties, such as lowering complexity, delay, and memory requirements, high achievable rates, and adaptability to unpredictable channel conditions. In general, there is a trade-off between these properties.

### 3.10.1. Fast Coding

The requirement of NC through intermediate nodes in the network in order to perform operations over finite fields in real-time. While costing inter-flow coding is usually low, the general linear codes used could be computationally not feasible.

In this case, the encoding algorithms require polynomial time complexity, which is bounded as O(n2), where n is the number of the linearly combined packets. Nodes may have to test whether the received packets are innovative (bring in new information), in which case an additional complexity of O(n2) operations is required.

Decoding amounts in order to solve a system of linear equations, which typically requires O(n3) operations. There is an intensified effort to design lower complexity encoding and decoding algorithms inspired by low density codes, but this effort is still at its initial steps.

Coding however is not a deployment hurdle in the current wireless networks. Chachulski et al. report that MORE, which is a NC protocol, can sustain a throughput of 44 Mb/s on low-end machines with Celeron 800 MHz CPU and 128 KB of cache.

### 3.10.2. Forced Reliability

Intra-flow NC does not organically lend itself to effortless performance degradation: receiving only n – 1 linear combinations of n linearly combined packets is practically useless. This imposes strict requirements for reliable delivery and is in stark contrast to uncoded transmission.

### 3.10.3. Real-Time Traffic

Intra-flow coding across n packets shows that a receiver needs to collect all n of them before extracting the data. In real-time applications such as audio and video, the associated delay might be excessive for large values of n, which are indicative of the fact that small values of n would be of use.

Using these small values of n, on the other hand, may not allow us to mix the information and come to the realization of the theoretically promised benefits. Therefore, there is a tension in balancing these two opposing requirements.

## 3.11. CONCLUSION

NC enables more skilled, scalable, and reliable wireless networks. These opportunities come with a need to rethink our MAC, routing, and transport protocols. However, it is believed that future research will overcome such challenges and integrate NC into the wireless network design.

## REFERENCES

1. Feng, C., & Li, B., (2012). *Network Coding for Content Distribution and Multimedia Streaming in Peer-to-Peer Networks*. [Online] Researchgate.net. Available at: https://www.researchgate.net/publication/267839813_Network_Coding_for_Content_Distribution_and_Multimedia_Streaming_in_Peer-to-Peer_Networks (accessed on 3 May 2021).
2. Fragouli, C., et al., (2007). *Wireless Network Coding: Opportunities and Challenges.* [Online] Available at: http://www.nms.lcs.mit.edu/6829-papers/coding-milcom.pdf (accessed on 3 May 2021).
3. Liew, S., Zhang, S., & Lu, L., (n.d.). *Physical-Layer Network Coding: Tutorial, Survey, and Beyond*. [Online] Arxiv.org. Available at: https://arxiv.org/ftp/arxiv/papers/1105/1105.4261.pdf (accessed on 3 May 2021).
4. Magli, E., Wang, M., Frossard, P., & Markopoulou, A., (2012). *Network Coding Meets Multimedia: A Review*. [Online] Arxiv.org. Available at: https://arxiv.org/pdf/1211.4206.pdf (accessed on 3 May 2021).
5. Qureshi, J., Foh, C., & Cai, J., (2012). *Optimal Solution for the Index Coding Problem Using Network Coding Over GF (2)*. [Online] Arxiv.org. Available at: https://arxiv.org/pdf/1209.6539v1.pdf (accessed on 3 May 2021).
6. Rout, R., & Ghosh, S., (2013). *Enhancement of Lifetime Using Duty Cycle and Network Coding in Wireless Sensor Networks*. [Online] Cs.bgu.ac.il. Available at: https://www.cs.bgu.ac.il/~segal/PAPERS/06399490.pdf (accessed on 3 May 2021).

# CHAPTER 4

# NETWORK CODING: MOBILE APPLICATION

## CONTENTS

This chapter explains the brief introduction into the network coding (NC) and how NC can be applied in mobile applications. This chapter also sheds light on the various transmission approaches, such as unicast, broadcast, and pure NC.

The chapter also provides highlights on the coding and cooperation in a network, along with the various protocols that require consideration. The chapter emphasizes the key management schemes that have been used in networking coding and small mobile cells.

This chapter also addresses the significance of multi-hop wireless network and packet relaying in multi-hop wireless networks. The chapter also explains the various routing protocols used in NC as well as multipath routing in wireless mesh networks (WMNs).

## 4.1. INTRODUCTION

Network coding (NC) has received a lot of attention since the term was "coined." Several research works have investigated and implemented NC to prove the feasibility of this novel technique. NC can be applied in various communication scenarios such as multicast or meshed networking, where NC without fail delivers promising results for throughput as well as reliability.

While most codes are end-to-end in nature, with NC packets can be recorded at every node in the network, which can be of special interest in multi-hop networks. This concept of NC has been proven to work in theory, some of the current questions are how to design NC algorithms and whether these algorithms are too complex for a given platform.

In References, it has been shown that NC can be applied to sensor networks and meshed networks formed by mobile phones. One finding was that NC techniques must be designed with care if they are to be applied to the mobile or implanted domain.

These platforms have a limited number of resources such as energy, memory, and computational power in addition to the general problems in mobile networking such as a limited wireless capacity.

The use of NC is motivated by the fact that the transmissions from one source to many sinks must be done in a dependable and well-organized manner. NC enables this as it gives space for an efficient spectrum usage and a low-level complexity error control system.

NC can be applied at differing protocol layers, ranging from the physical layer over the network layer to the application layer. In this work, the major focus is on the application layer. Furthermore, the chapter provides some execution guidance on how to keep the complexity of NC low (Figure 4.1).


**Figure 4.1:** Network coding plays a very significant role in mobile application.

New technologies are emerging in order to create the next generation 5G network. These latest technologies will deliver a higher network capacity, allowing the support of more users, lower the cost per bit, enhance energy efficiency and provide with the adaptability to introduce future services and devices.

It is envisioned that the 5G network will support data rates reaching speeds up to 10 Gb/s and reducing the latency to as low as 1 millisecond end-to-end. One approach of ascending throughput inside the 5G network is by utilizing the NC.

NC is an emerging network technology, which no longer treats data, moving through the network from sender to receiver, as mere commodities. Traditional routers inside a network can duplicate and can forward incoming data packets, but NC allows multiple packets at a router to be encoded together before being forwarded.

> The concept of network coding was first introduced in.It is an emerging communication paradigm which has the potential in or der to provide a significant benefit to networks in terms of band width,energy consumption,delay,and strength to packet losses.

Another emerging technological trend for the 5G network is small cells. The small cell technology is the most successful solution to deliver omnipresent 5G services in a very cost-effective and energy-efficient manner to its own users. Mobile small cells are proposed to cover the urban landscape and can be set up on the fly based on demand, using mobile devices (i.e., user equipment) or remote radio units (RRUs).

Moreover, mobile small cells are networks which consist of mobile devices which are within a relatively proximity to one another, and thus, it allows device-to-device (D2D) communication which enable high data rate services such as video sharing, gaming, and proximity-aware social networking.

As a result, end-users are provided with this abundance of 5G broadband services while the D2D communications extemporize throughput, energy efficiency, latency, and civility.

One of the scenarios being examined in SECRET centers on secure NC-enabled mobile small cells. In this case's architecture, the technologies of mobile small cells, NC and D2D communications are put together. The cellular network, consisting of macrocells, is ram shackled into mobile small cells.

Each of the mobile's small cells are controlled by a hotspot (or cluster-head). This is a mobile node (device) within the cluster of mobile nodes which is selected to become the local radio manager in order to control and maintain the cluster.

In addition, the hotspots of the differing clusters are supervised by a centralized software-defined controller. Through a cooperative effort, these hotspots form a wireless network of stationery small cells which have several gateways/entry points to the mobile network using intelligent high-speed connections. Data traffic between mobile nodes is established through D2D communications and optimized by utilization of NC.

Imagine that a mobile node wants to share a multimedia file with two other mobile nodes. The mobile node in possession of the multimedia file, the source node (SN), sends through this file to the mobile nodes requesting the file and the destination nodes (DNs).

Note that these mobile nodes are not required to be in the same mobile small cell. Through D2D communications, the multimedia file-using multiple hops is being routed by the mobile nodes, through a network of small mobile cells from the SNs to the DNS.

This architectural structure has various advantages if compared to the currently employed architecture. By allowing multi-hop D2D communications throughout a network of small mobile cells, data traffic within this scenario is no longer required to be routed through the base station (BS).

This means that the data is no longer required to travel long distances to and from the BS, but has a more direct route. This leads to a significant reduction in latency. Since the transmissions travel shorter distances, the transmissions require less amounts of power to reach its destination.

This means that this architecture also allows data transmission to be more efficient, energy-wise. This architecture also reduces the workload of the BS, which mitigates stress on the cellular network.

## 4.2. TRANSMISSION APPROACHES

Different approaches for transmitting the data are feasible and depicting certain possibilities about the same. Let us assume that a single source broadcasts data to N syncs $t_1 \ldots t_N$ and that the source has a direct wireless link to the sinks.

The data can be divided into several packets, for example, transmitting packets over the wireless link might lead to packet loss due to the attributes of the wireless channel; thus, an error control system is needed in place.

### 4.2.1. Unicast

The easiest solution is for the source to send in the data in a round-robin fashion using a dependable unicast protocol, e.g., transmission control protocol (TCP). Such an approach is completely reliable as each sink is served and looked at individually.

Each sink acknowledges the received packets, and thus, the source device can determine when all the sinks have received packets. This solution is simple and the computational complexity of the process is low. However, if N is high and ascending, the amount of unimportant information sent from the source becomes significant (Figure 4.2).

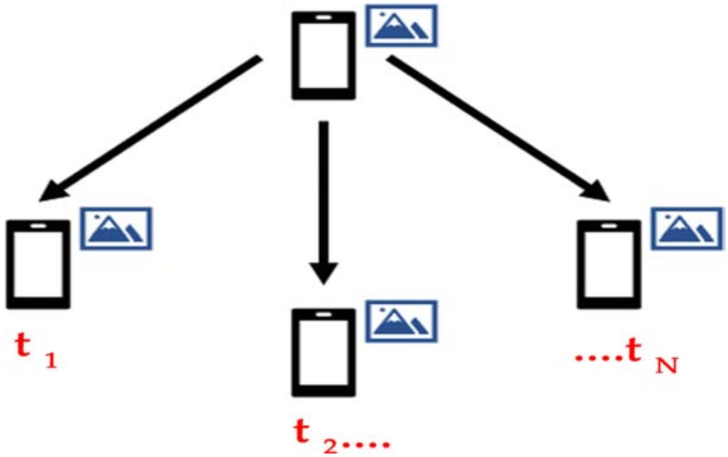



**Figure 4.2:** The basic picture viewer setup.

### 4.2.2. Broadcast

Instead of sending a broadcast to each device one by one, the source would broadcast data towards the sinks. This approach is exceedingly efficient if no errors take place on the wireless link. However, when packet losses occur, some sort of error correction is needed.

To achieve reliability, the source needs to take cognizance of which packets have been lost by one or more of the sinks and those should be retransmitted as this introduces the need for feedback information, which consumes spectrum as well as time.

The amount of feedback information depends on N and the packet error probability (PEP). The feedback messages can be small and as such do not require a lot of diversity. However, they possibly introduce collisions in the network as both the source and sinks will attempt to transmit packets simultaneously.

Thus, the performance of such a broadcast approach depends on the effectiveness of the medium access control (MAC). Moreover, the retransmissions by themselves are suboptimal as not all sinks will lose the same packets, thus each retransmitted packet will only be useful for a subset of the sinks.

For example, if mobile devices 1, 2, and 3 have lost packet 17, 21, and 16 respectively, three broadcast packets need to be transmitted, and each retransmitted packet is solely useful for a single sink. Generally, broadcast can be faster than unicast if N > 1 and its performance is less sensitive in comparison pointing towards the number of sinks.

### 4.2.3. Pure Network Coding (NC)

One NC approach that puts itself in this scenario is random linear network coding (RLNC). With this approach, coding is used in order to ease the issue of correcting lost packets at the sinks and moreover reduces the requirement for feedback.

In NC, nodes can combine this information in the network in order to create new packets. Hence, the source codes g + r packets from the g original packets and broadcasts these packets. r is the number of unnecessary packets and should be chosen according to the PEP of the link.

Each sink must be on the receiving end of any g linear independent packets, which can therefore be decoded in order to recreate the original packets. The advantage of NC can be illustrated by the previously mentioned example.

In this case, the source could code packets 16, 17, and 21 together into a fresh new packet of the same length as the original ones. This packet is broadcasted towards the three sinks, which each remove from the coded packet and the packets they already got and therefore decode it into the packet they lost (Figure 4.3).


**Figure 4.3:** An illustration of pure network coding.

*Source: Image by unsplash.com.*

Thus, the retransmission that needed three transmissions using broadcast can be done by a single transmission using NC. As the coding and decoding operations introduces complexity, the computational requirement is increased.

These operations will increase the central processing unit (CPU) load and thus the energy consumption. However, the number of redundant packets transmitted from the source and feedback messages sent from the sinks can be decreased, which help to decrease energy consumption.

### 4.2.4. Systematic Network Coding (NC)

To decrease the complexity, systematic NC can be used. Systematic NC combines the broadcast and NC approaches. As there is no obvious gain in coding the first g packets, the source broadcasts these packets and codes the remaining r packets.

Each uncoded packet is useful for all N syncs as they are linear independent. The following r packets are coded and have a high probability of being independent of the n uncoded packets. This approach decreases the computational complexity at the source and the sinks as only r packets must be coded and decoded.

## 4.3. CODING AND COOPERATION IN A NETWORK

To observe the effect of recording, it is required to test a setup where the sinks are being cooperated by forwarding recorded packets to each other. The easiest approach is to let each sink recode as well as forward whenever it receives a packet from its source, with some fixed probability of pR.

This probability should be chosen in accordance to the PEP of the sinks in the cluster, and will also depend on what parameters a person wishes to improvise, wherever he or she have chosen 5% and 10%. This simple protocol allows us to observe the effect of recording.

Enabling the recording should offload the source, in terms of both energy and computations, by moving some of the coding to the cluster. The effect should be highest when the PEP is high, and especially visible in cases where the channel between the source and a sink is weaker, but the channels between cooperating sinks are stronger.

In this case, a fixed pR was used; however, for a real protocol implementation, it will be imperative to consider when exactly a sink has enough information to be a useful ‘recorder.’ In the test, the measurement is from the first packet, after receiving the packet which completes the decoding for the generation which was on the receiving end (Figure 4.4).

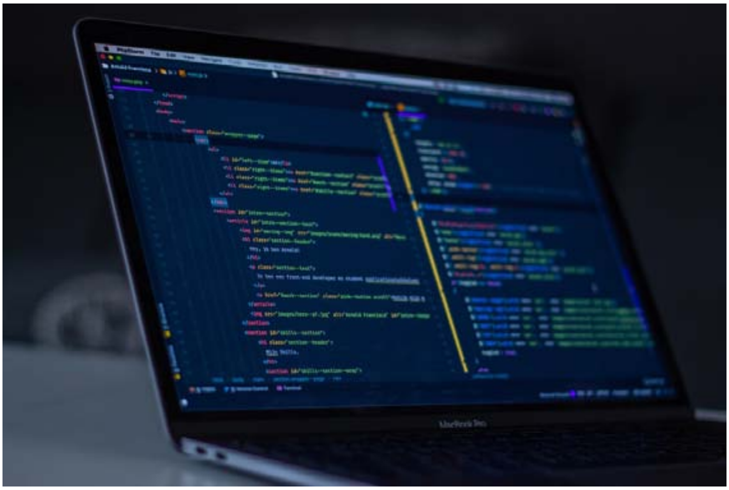


**Figure 4.4:** Coding and cooperation in a network.

*Source: Image by unsplash.com.*

This allows us to obtain the performance as if there is a perfect feedback channel and feedback scheme. The sink records the following parameters-time per generation, PEP, total packets, uncoded packets, coded packets, linear dependent packets, relayed coded packets, relayed linear dependent packets.

As in the previous setup, the test was conducted using three Nokia N95s, one source and two sinks. Packets were coded using the generation size q = 64 and a packet size of 1200 bytes. Approximately 9.000 test runs were completed in totality for each pR.

However, in the relay communication, it can be observed, as predicted, the relay becomes an ascendingly better source of information as the PEP increases. The main reason for this is that the relay will only be able to mend uncorrelated losses, that is, losses which occurred only on the other sink.

In addition, since RLNC is using the relay which will randomly pick whichever packets to recode, furthermore minimizing the probability of selecting a useful packet. However, as the PEP increases, so does the probability that one relay has useful packets in order to offer the second relay.

This tendency can be seen for pR = 5%, where the ratio of import packets changes from 19% to 73% and for the pR = 10% case where the ratio changes from 21% to 58%.

## 4.4. PROTOCOL CONSIDERATIONS

The challenge of making sure dependable multicast transmission in arbitrary networks is an open problem with no solution within reach. In order to make a usable application, this problem needs to be addressed at least for the scenario where the application is deployed.

The solution in this prototype is simply present to overshoot, therefore sending additional packets for each generation in order to compensate for the packet losses. Such an approach is, for example, used in multimedia broadcast and multicast services (MBMS) systems where the overshooting is tuned based on uncommon feedback from nodes in the network, such that an already defined fraction of the sinks may decode (Figure 4.5).


**Figure 4.5:** Protocol consideration in network coding.

Because the overshooting is fixed at a certain level, the sinks that experience a packet loss below this level will be equipped in order to decode the data, while the remaining sinks will not. This approach is uncomplicated and works well if the sinks have relatively uniform and static channel conditions.

If the feedback channel is weak or non-existing, this may be the only available solution. Another approach is to let the sinks request for more data if they need it. The source sends data from a generation, and alternatively, it also sends some overhead, and thereon proceeds to the next generation.

If any of the sinks were unable to decode the generation, they signal that they need additional information which the source sends through. This approach adapts better to changing channel conditions and, as such, can utilize the channel better.

However, the feedback from the sinks introduces the exposure problem and the crying baby problem. Therefore, this approach works best if the sinks have relatively uniform channel conditions, and if the number of sinks is moderate in nature.

As the links to sinks are independent, they will hold differing information when the source has transmitted data. Thus, an interesting approach is to let the sinks cooperate and thereon exploit the connection diversity. Instead of a sink requesting additional data specifically from the source, any node which received the request could respond, therefore more than one node could potentially attempt to answer this request, which would introduce the implosion problem.

One of the main setbacks of this approach is the high complexity it introduces itself with, one technique to pacify this could be the NC. Additionally, if done correctly, it could potentially allow for transmission in partially connected networks.

Thus, in addition to the overall system operation, there are several problems with reliable transmission in a broadcast network which must be addressed, namely the implosion, exposure, and crying baby problem.

Furthermore, a range of protocol functionality is integral or beneficial, such as service discovery, cluster forming, multi-hop routing, connection loss and reconnection, TCP friendliness, and security, especially when partial connected mesh networks and cooperation is considered.

## 4.5. KEY MANAGEMENT SCHEMES

The proposed scenario architecture brings a set of new technologies in conjunction, and with that, it also comes with several security and privacy challenges. This section will explore all kinds of challenges that every individual technology poses in the proposed architecture, its effect on proposing a key management scheme, and state-of-the-art key management schemes proposed for a similar networking architecture which can be used as inspiration to design our own (Figure 4.6).


**Figure 4.6:** Key management schemes in network coding.

### 4.5.1. Network Coding (NC)-Enabled Network

A NC-enabled network allows the encoding of data packets at routers inside of the network and decoding at the receiver's end. This provides integral beneficiaries to networks in terms of bandwidth, energy consumption, delay, and robustness to packet losses.

Despite these tremendous advantages, networks utilizing NC technology are vulnerable to the so-called pollution attack. In this attack, a malicious adversary controls a router such that it can mutate data packets by introducing pollution in the original data packet.

NC causes this pollution to spread downstream by encoding proper data packets with polluted data packets. This leads to the inability to properly decode and retrieve the information at the intended receivers.

A successful pollution attack wastes a lot of costly network resources. The challenge posed by this attack is like the vulnerability of data modification in any wireless network. Data integrity is required to prevent any polluted data packets from being transmitted further through the network.

The research community proposed various integrity schemes to solve this problem. However, the efficiency and effectiveness of the integrity schemes are closely related to the key management schemes, which are responsible for the generation, distribution, use, and update of the cryptographic keys used by the integrity schemes.

In the literature, there are various proposed schemes for key distribution which are used in NC-enabled networks, but they suffer from drawbacks that limit their effectiveness and reliability.

Therefore, it is of utmost importance the design of novel key management schemes that can overcome the limitations and drawbacks of the existing key management schemes in order to support efficiently and effectively existing and new integrity schemes against pollution attacks in NC-enabled mobile small cells.

### 4.5.2. Mobile Small Cells

The introduction of mobile small cells in the researched scenario architecture makes the network dynamic in nature. Every mobile node inside the network can always be on the move. Certain mobile nodes were able to leave the macro cell, and other mobile nodes could join the macro cell.

This network thus has a constantly changing topology, and hereon, it poses a problem when it comes to key management. Traditional certificate-based public-key cryptography (CB-PKC) relies on a trusted third party (TTP) called a 'certifying authority' (CA).

The CA issues certificates to users inside the network premises. These certificates are used to verify the identity and provide a cryptographic key at the same time. This CA can be interpreted as the key manager, and it is a concretely central control point that every node in the network trusts.

However, a CA does not fit in the studied scenario architecture due to the lack of infrastructure. On the other hand, identity-based public-key cryptography (IB-PKC) unfasten the requirement of certificates, since public keys in IB-PKC are equivalent to the identity of the mobile nodes.

However, private keys are obtained from the key generation center (KGC). KGC holds a master key from which it produces private keys. Consequently, a completely bargained KGC implies that the whole framework is undermined.

This implies that IB-PKC experiences a solitary mark of disappointment, alongside the key escrow issue. At last, certificateless public key cryptography (CL-PKC) is acquainted with addressing these issues. With CL-PKC, private keys are developed by both the KGC and the versatile client mentioning the private key.

The KGC produces the initial segment of the private key, and the portable client finishes the private key by consolidating it with his own private key. The assignments of the KGC can be conveyed among versatile hubs utilizing certain mysterious sharing.

An undermined KGC utilizing CL-PKC just furnishes the assailant with halfway private keys. CL-PKC not just addresses the single place of disappointment and the key escrow issue; however, it can likewise fulfill the powerful geography of the organization.

Certificateless key administration conspires in this manner appear to be a decent contender for the considered situation engineering, anyway many proposed plots experience the ill effects of the private key dispersion issue, or they need key update or key disavowal methodology.

### 4.5.3. Cryptographic Security Arrangements

Having investigated the difficulties which are delivered by the thought over situation design, this part examines how these difficulties can be tackled. By permitting the situation engineering to perform multi-bounce D2D correspondence, a range of security and protection challenges emerge.

Notwithstanding, cryptographic methods and mysterious shared validation can give mystery and secrecy. Gatherings wishing to convey safely require a common cryptographic key to exploit the cryptographic strategies and unknown shared verification.

Key administration plans are liable for the age, appropriation, stockpiling, use, denial, and update of these cryptographic keys. It is subsequently critical to research the plan of novel key administration conspires that fit with our situation engineering and give every one of these functionalities in a proficient and compelling way.

Also, to completely abuse the benefits of organization coding in our situation design, novel key administration plans are needed as most of the current ones cannot completely uphold the information respectability plans proposed in the writing to forestall contamination assaults in NC-empowered organizations.

What is more, the security of portable little cells is additionally influenced by key administration. The powerful geography that versatile little cells bring to our situation engineering represents the test of a reasonable group of key administration plans.

As examined, CB-PKC IB-PKC, and their separate key administration plans are not reasonable. Then again, CL-PKC and certificateless key administration plans appear to be a decent applicant.

Notwithstanding, existing certificateless key administration conspires either need key update or key denial methods, or they require a protected station for (incomplete) key circulation which is hard to acknowledge in our situation design.

In this way, it is critical to plan novel key administration plans for our situation engineering. These plans ought to give powerful and low intricacy key administration including secret key dividing between portable hubs, key repudiation, key update, and versatile hub verification.

At last, they ought to likewise uphold existing and new uprightness plans against contamination attacks in NC-enabled mobile small cells in an efficient and effective manner.

## 4.6. MULTI-HOP WIRELESS NETWORK

Permitting information parcels in transmission to navigate different jumps to arrive at the expected collector brings a range of security dangers. These security dangers can be part into two classes, information protection and personality protection.

Information protection dangers cover all assaults wherein the aggressor attempts to reveal data about the information communicated to the expected recipient. The aggressor utilizes strategies, for example, listening in and character pantomime. These assaults are all around examined, and different cryptographic procedures have been created to keep these assaults from being successful. These cryptographic procedures can give information classification utilizing information encryption plans, substance validation utilizing distinguishing proof plans, and information confirmation utilizing mark plans.

These methods counter all the previously mentioned difficulties. Be that as it may, a significant number of these countermeasures require both the sender and the planned recipient to be in control of a common cryptographic key (Figure 4.7).


**Figure 4.7:** Multi-hop wireless network.

Accordingly, clearly key administration assumes a basic part to accomplish information protection. Personality protection is the other class of security dangers in a multi-bounce remote organization. The test of giving personality protection lies in the foundation of secure correspondence between two portable hubs.

To build up secure correspondence between two versatile hubs, the two hubs are needed to demonstrate their personality to one another. This necessity keeps any assailant from utilizing a pantomime assault. Be that as it may, both versatile hubs wish to stay unknown to the moderate hubs directing the recognizing data.

This test can be addressed with mysterious common confirmation. With mysterious shared verification, both versatile hubs take part in an intelligent zero-information evidence of personality convention. This convention includes trading difficulties to demonstrate their character to one another, without really sending any private distinguishing data.

Nonetheless, every one of the zero-information verification of personality convention either requires both portable hubs to have a pre-set-up secret or relies upon a TTP. This TTP is a focal control point that each hub in the organization trusts yet does not fit in our proposed engineering because of the absence of framework. Both versatile hubs are hence needed to have a pre-set-up secret (like a common cryptographic key) to impart.

### 4.6.1. Packet Relaying in Multi-Hop Networks

In remote multi-bounce organizations, hubs speak with one another utilizing remote channels and do not have the requirement for basic framework or incorporated control. Hubs may help one another by sending or handing off every others' parcels, perhaps including many middles of the road transfer hubs.

This empowers hubs that cannot hear each other straightforwardly to convey over halfway transfers without expanding transmission power. Such multi-bounce handing-off is an extremely encouraging answer for expanding throughput and giving inclusion to a huge actual zone.

By utilizing a few middles of the road hubs, the sender can diminish transmission power in this manner restricting obstruction impacts and empowering spatial reuse of recurrence groups. In specially appointed organizations, the medium is shared, and hubs mastermind admittance to the medium in an appropriated path free of their present traffic interest.

Specifically given standard specially appointed steering conventions that attempt to limit handing-off hubs on the way, hubs nearer to the organization community are bound to turn into a transfer hub. This has the characteristic downside that a hub that fills in as a hand-off hub for transmissions of various adjoining hubs is inclined to turn into an exhibition bottleneck.

As it is important to comprehend the execution of such transfer organizations, the following sub-area gives an outline on execution investigation of a hand-off hub. At the point when different transfers are included across a start to finish way, it is essential to control overhead for each single bundle transmission.

Lamentably, current MAC and actual layers for wireless local area network (WLAN) based multi-bounce networks force high overhead for the transmission of little information parcels, which is normal for voice over internet protocol (VoIP).

By joining a few little bundles into bigger ones, per parcel transmission overhead can be decreased essentially. Accordingly, the accompanying subsections give an outline on proficient parcel conglomeration systems.

### 4.6.2. Bundle Aggregation for VoIP in Wireless Meshed Networks

The arrangement of VoIP in remote cross-section networks is a significant assistance for the future remote web. Notwithstanding, the transmission of little (voice) parcels forces high MAC and physical layer overhead, which prompts low limit with respect to VoIP over IEEE 802.11-based multi-jump network organizations.

The possibility of bundle accumulation is to join a few little parcels into a bigger totaled one so overhead on the remote medium can be fundamentally decreased. While such total components have been proposed for single-jump framework remote neighborhood, planning a total methodology for multi-bounce remote cross-section networks is a difficult issue.

In foundation remote neighborhood, the sender has total information about the connection attributes of one jump neighbors and would thus be able to ascertain an ideal parcel size for conglomeration. In a multi-bounce climate, signal quality and clog for each connection are extraordinary.

At the point when lattice hand-off hubs total little parcels, there is an innate compromise in regards to bundle size. Totaling more parcels prompts bigger amassed ones, lessens the general number of bundles in the cross-section and prompts diminished multi-jump dispute and parcel misfortune because of crashes.

Notwithstanding, such bigger amassed bundles can bring about higher parcel misfortune for a connection that works at low sign quality. For such connections, amassing fewer bundles can be useful. For proficient parcel accumulation, it is fundamental to have enough bundles in the nearby line to be totaled.

In this way, parcels are falsely postponed to expand the conglomeration proportion, which may prompt better quality to end delay. Then again, conglomeration lessens the general number of parcels in an impact area, diminishing multi-bounce disputes, crashes, retransmissions, and, subsequently, MAC layer use, which may decrease the start to finish delay.

Packet aggregation can be classified as end-to-end or hop-by-hop. In end-to-end aggregation, all packets towards a common destination are aggregated. In hop-by-hop aggregation, aggregation and de-aggregation are done at every node, which leads to higher complexity and potentially higher delay.

However, it yields better aggregation possibilities as packets for different destination addresses but with the same next-hop could be aggregated. In a realistic wireless mesh network deployment, link characteristics and load will be different for each hop.

Therefore, a hop-by-hop aggregation scheme enables an optimization of the packet size used for aggregation for each hop. This allows to trade-off packet loss due to contention and bit errors.

Hop-by-hop aggregation outperforms end-to-end aggregation strategies, because the overall aggregation along a whole path will not be constrained by the weakest link, leading thus to significant performance improvement compared to end-to-end aggregation mechanisms.

## 4.7. ROUTING PROTOCOLS

The objective of routing is to route data from a sender to one or more destinations. Routing in a mobile wireless multi-hop network, and in mobile ad-hoc networks is a challenging task. Routing protocols in mobile ad-hoc networks are usually divided into proactive, reactive, and hybrid routing.

A proactive protocol evaluates routes to all reachable nodes and attempt to maintain consistent, up-to-date routing information. In a reactive protocol, routing paths are searched only when needed. Hybrid protocols combine proactive routing with reactive routing in hierarchical network structures (Figure 4.8).


**Figure 4.8:** Routing protocols in network coding.

The mobility of nodes in combination of the noisy links calls for new approaches in order to obtain optimal network performance. Also, new applications and systems require more than the traditional unicast routing protocols. For example, broadcasting and multicasting protocols targeted at mobile wireless networks are needed. In this section, various investigations of routing protocols for mobile wireless multi-hop networks are discussed.

First, real experiments with three of the most popular mobile ad-hoc network routing protocols are described: Ad-hoc On-demand Distance Vector protocol (AODV), optimized link-state routing protocol (OLSR), and dynamic source routing protocol (DSR).

The main focus of the experiments was to evaluate the reactivity of the protocols compared to power and bandwidth consumption. The next section then discusses the issue of broadcasting for multi-hop wireless networks.

It also proposes and evaluates a new protocol for stateless broadcasting, the dynamic delayed broadcasting (DDB) protocol. Multipath routing allows the establishment of multiple paths between source and destination in wireless mesh networks (WMNs).

Then, multicast routing for mobile ad-hoc networks including two new protocols is presented: QAMNet, which is an approach to improve the quality of service (QoS) for multicast communication, and RObust VEhicular Routing (ROVER), which is a reliable multicast protocol for vehicular networks.

Finally, an intelligent navigation system based on traffic monitoring with multi-hop communication for vehicular networks is proposed. It is shown that with the use of multicast routing, intelligent navigation systems that make re-routes in case of accidents or traffic congestion can be developed.

### 4.7.1. Multipath Routing in Wireless Mesh Networks (WMNs)

WMNs provide a cost-efficient way to interconnect existing wireless networks as well as to supply larger areas with network access. WMNs offer a more robust and redundant communication infrastructure than the wireless networks deployed today.

They offer communication facilities in situations where certain systems, e.g., global system for mobile communications (GSM), might be overloaded. The unreliability of the wireless medium, resource-constrained nodes and dynamic topologies make wireless mesh and mobile ad-hoc networks prone to transmission failures, node failures, link failures, route breaks, and congested nodes or links.

One important approach to overcome this problem and to exploit WMNs for robust real-time communication is path diversity. For each destination, multiple routes are provided by a multipath routing protocol, e.g., to support real-time data transfer.

Appropriate coding and path allocation is selected for the given network conditions and, therefore, the degree of redundancy in transmission is set. Multipath routing allows the establishment of multiple paths between source and destination.

This provides increased reliability of the data transmission and fault tolerance or load balancing. Several multipath routing approaches enhance the well-known single path routing protocols AODV protocol or DSR protocol.

Split multipath routing (SMR) with maximally disjoint paths in ad hoc networks (SMR) extends DSR to create two maximally disjoint paths. The routing scheme prohibits intermediate nodes from replying on route requests (RREQ).

Intermediate nodes forward duplicate RREQ messages, if they arrive through a different link and if their hop count is equal or lower than the previously received one(s). The destination responds to the first RREQ with a route reply (RREP) message as it represents the shortest delay path.

From subsequently received RREQs the destination selects the maximally disjoint path and establishes a second path by sending a RREP. Both paths are then equally used for data transmission. Node-disjoint multipath routing

(NDMR) adapts the same Split Multipath scheme for AODV protocol.

The criteria for forwarding the RREQs are the same as in Split Multi-Path, but the behavior of the destination is changed. After setting up the shortest delay paths, the destination only selects paths that are node-disjoint to the already established one(s).

Ad-hoc on-demand multipath distance vector protocol (AOMDV) [Mar02] and ad-hoc on-demand distance vector multipath protocol (AODVM) [Ye03] represent other multipath variants of AODV protocol.

AOMDV protocol discovers multiple loop-free paths during a single route discovery. AOMDV replaces the hop count of the AODV protocol by an advertised hop count to a destination, which represents the maximum hop count for all available routes to the destination.

The routing entries further contain a list of next hops with jump checks rather than one basic next bounce for every objective. RREQ or RREP bundles update the directing data at a hub either for a converse or forward way. Copies of such course notices may characterize substitute ways to objective or source. Like in AODV protocol, grouping numbers ensure the newness of the directing data.

To try not to defeat circles, substitute ways are possibly acknowledged whether their bounced check is more modest than the promoted jump mean a similar objective arrangement number. The gathering of a more up-to-date objective grouping number reinitializes the promoted jump consider well as the following bounce list for this objective.

AOMDV protocol may either discover hub disjoint ways or connection disjoint ways. For hub disjoint ways, every hub just acknowledges RREQs showing up from various neighbors.

### 4.7.2. Multicast Routing

In portable specially appointed organizations, effective help of multipoint interchanges is fundamental to offer types of assistance like gathering sound and video conferencing, spread of information to a bunch of beneficiaries or joint effort of a gathering of clients.

Additionally, most of the significant intelligent gathering administrations like gaming or conferencing have extremely solid QoS necessities with respect to defer and data transfer capacity. Multicast steering conventions for versatile impromptu organizations can be characterized into three- or lattice put together depending with respect to the basic sending structure that they use. Tree-based plans, for example, [Roy99, Ji98, Jet01b] develop a multicast tree from every one of the sources to every one of the beneficiaries utilizing either source-based trees or shared trees. Cross section-based methodologies, for example, [Lee02, Gar99] figure a few ways among sources and objections. Crossbreed conventions, for example, [Bom98, Sin99] attempt to join the strength of lattice-based specially appointed directing and the low overhead of tree-based conventions.

At last, stateless multicast conventions, for example, [Ji01, Jet01a] do not keep up sending states on the hubs concerning model the arrangement of hubs to navigate is remembered for the information bundles themselves.

Numerous vehicular organization applications require position-based multicasting, e.g., for spreading traffic data to vehicles moving toward the current situation of the source [Sic07]. Geo-cast conventions that forward messages to all hubs inside a zone of relevance (ZOR) [Mai04] are the regular counterpart for this kind of directing.

A few applications will require multicast transmission with start to finish QoS. Flooding-based geo-cast conventions are not proposed for these kinds of uses. Hence, there is a need to create multicast conventions for vehicular ad-hoc networks (VANETs) that can uphold start to finish QoS systems actualized in a vehicle layer convention.

## 4.8. CONCLUSION

In the conclusion of the chapter, various aspects of NC have been mentioned throughout the chapter, starting with the basic introduction to the application of NC in mobile application. This chapter also discussed about the several different approaches of transmission such as unicast, broadcast, pure NC, and systematic NC.

The importance of coding and cooperation of NC in a network has been also discussed in the chapter. This chapter also discussed about the various protocols that needs a deep consideration in NC. Towards the end of the chapter, various key schemes related to the management of coding have been mentioned.

This chapter also discussed about the significance and application of multi-hop wireless network with an example of packet relaying in multi-hop networks. In the end of the chapter, various types of routing protocols have been discussed.

## REFERENCES

1. Fitzek, F., Pedersen, M., Heide, J., & Médard, M., (2010). Network coding. *Proceedings of the 5th ACM Workshop on Performance Monitoring and Measurement of Heterogeneous Wireless and Wired Networks-PM2HW2N '10*, [Online] Available at: https://www.researchgate.net/publication/229015784_Network_coding_applications_and_implementations_on_mobile_devices (accessed on 3 May 2021).
2. ITN – Secret, (2019). *Secure Network Coding for Mobile Small Cells: Survey and Reference Architecture*. [eBook] ITN-Secret. Available at: https://ec.europa.eu/research/participants/documents/downloadPublic?documentIds=080166e5c0566c6b&appId=PPGMS (accessed on 3 May 2021).
3. Pedersen, M., Heide, J., Fitzek, F., & Larsen, T., (2010). A mobile application prototype using network coding. *European Transactions on Telecommunications*, [Online] *21*(8), 738–749. Available at: https://sci-hub.do/10.1002/ett.1448 (accessed on 3 May 2021).
4. Vasilios, S., Geert, H., Torsten, B., Andreas, K., Maria, K., & Veselin, R., (n.d.). *Chapter 5: Multi-Hop Wireless Networks*. [eBook] Available at: http://www2.aueb.gr/users/vsiris/publications/k5_COST290_Book_Ch5.pdf (accessed on 3 May 2021).

**CHAPTER 5**

# NETWORK CODING IN APPLICATION LAYER MULTICAST

## CONTENTS

The chapter of network coding (NC) in application layer multicast (ALM) explains the significance of peer-to-peer (P2P) networks and various protocols of ALM. This chapter provides a brief introduction to the NC for multicast networks and linear network coding (LNC) for multicast.

Also, in this chapter, the detailed differences between deterministic NC and random network coding (RNC) have been mentioned. This chapter also provides highlights about the NC in P2P file sharing. This chapter addresses the deterministic linear coding over combination networks, which includes peer joining, and peer leaving.

This chapter emphasizes the evaluation of various strategies that are used for optimal routing, such as the maximum flow problems, and evaluation of the transmission topology. This chapter explains the application or use of multicast in cloud networks. This chapter also mentions the use of NC multicast in satellite network, that emphasizes NC in wireless networks, and NC in satellite networks.

In this chapter, various reliable multicasts have been explained that are used in fixed and land-mobile satellite services, such as transmission reliability, retransmission strategies, packet-level coding, alternative approaches of coding, and impact of packet-level coding on retransmission. This chapter also includes various protocols and algorithms that have been used in ALM.

## 5.1. INTRODUCTION

The two major reasons due to which application-layer multicast (ALM) protocols suffer from delay are data and security. The main purpose behind introducing network coding (NC) techniques is to address these issues. NC is too complex in these topology structures. There is a novel by the name POSET protocol stack in which all the problems related to this has been addressed.

Control protocol, POSET structure protocol and data distribution protocol are the three protocols which a novel consists of. So, join, leave, and prune operations is controlled by control protocols, matrix property is used by the POSET structure and finite filed encoding, intermediate encoding and finite field decoding modules distribution protocol consist of.

After the introduction of this protocol, the results have been compared with the existing nice protocol. Therefore, there is a decrease in delay by 33% and increase in 8.25% of data. To deliver the message from one end to another end today internet uses routing. Encoding, duplicating, and forwarding of messages is done by the relay nodes. Simplification of routing is knowing as NC which allows relay nodes to perform these functions.

**Multicast capacity can be achieved by the network coding and because of this amount of multicast network has improved. Due to two reasons, application layer multicast (ALM) is considered as a perfect candidate to apply network coding.**

The first reason is that ALM is built on peer-to-peer (P2P) networks whose topology can be arbitrary, so it is easy to tailor the topology to facilitate NC, and the second reason is that the nodes in ALM are end hosts which are powerful enough to perform complex encoding and decoding operations.

So, the following contributions to the theory and practice of NC and its application to ALM is present in this thesis. To multicast the networks, a general approach to apply linear network coding (LNC) is proposed. A series of minimal NC problems is investigated by the user, and to solve them under a unified framework, a systematic approach is proposed.

In P2P file-sharing system and the P2P media streaming system, LNC is applied by the user, respectively. Every system has its own features and requirements, and also these two systems have.

The main focus is on throughput and reliability, in the P2P file-sharing system. It has been seen that the construction of overlay topology is done in a way that it can be looked as a union of multiple combination networks.

A general LNC scheme for combination networks is proposed by the user and adapt it to the P2P file-sharing system. As by comparing with other system without NC, the result is that simulation shows great improvement in both throughput and reliability.

The main focus is on heterogeneity and bandwidth utilization of the access links of peers in the case of P2P media streaming system. The network model is adapted by the user, which consists of a bandwidth bottleneck lies only at the edge of the network.

Through multiple description coding (MDC) media content is encoded into multiple stripes. On the basis of downloaded bandwidths peers subscribe to the stripes. Within the same stripe, random linear network coding (RLNC) is performed. Peers achieve much higher satisfaction in terms of received downloading rate by combining MDC and NC.

Apart from this, the investigation is done by the user between the inter-session LNC problem and multiple simultaneous multicast sessions. To evaluate the NC benefit and it is based on a practical inter-session NC scheme for multicast networks. Around 30% of the system throughput is increased, as in most cases when it is compared to intra-session NC.

## 5.2. PEER-TO-PEER (P2P) AND ALM

The evolution of P2P technology offers a great alternate solution for multicast communication. Basically, P2P is referred to as a completely distributed network architecture which is quite different from the traditional server-client model. In the case of the server-client model, a server provides a centralized service that is requested by different clients, such as hosts (Figure 5.1).

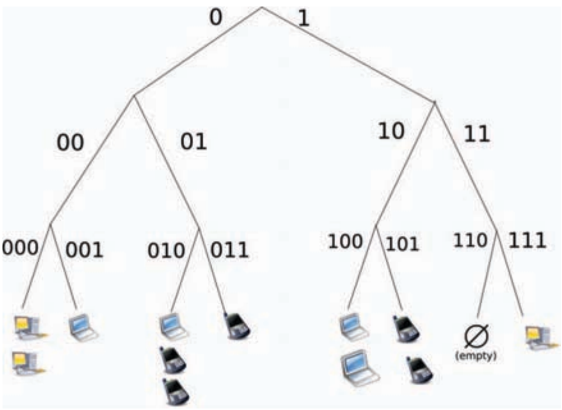



**Figure 5.1:** Peer-to-peer network diagram.

*Source: Image by Wikimedia commons.*

In general, the hosts already know the address of the server, that too in advance. One of the most noticeable drawbacks of server-client model is its limited bandwidth as well as resource on the server-side. As the bandwidth and resource of the server is shared by all the hosts, the server can consequently be dazed by a huge number of hosts.

P2P model usually allows the host to create an ad-hoc logical overlay network on the top of the physical network. The links of the overlay network are actually the logical links, each of which can be plotted to a physical path in the physical network.

Hosts can easily share the information along with bandwidth with each other via overlay network. In turn, this necessitates hosts to be capable of performing much more complex operations like routing and overlay topology construction or maintenance.

The hosts are known as peers as they are similar in terms of their specific functionalities. The benefits or advantages of P2P systems are very clear. First of all, the larger the size of the system is, the large the total bandwidth is. Peers can contribute their bandwidth to the system, as they can easily share their bandwidth with each other. It is clear that more peers join the system, there would be more bandwidth the system has.

Secondly, the processing is well-distributed, that is, there is no single point of failure issue. In the server-client model, as compared to the hosts, the server is much more important. In case, the server is down, the whole system would be down. This is further known as single point of failure problem.

While, in P2P systems, peers are of equal significance. The overly network formed in a distributed way is such that the system can sustain working in an effective way even if some peers leave the system. Practically, a lot of P2P systems adopt a hybrid model.

Still, there is a server holding the resource which is requested by a lot of hosts. In order to make the system scalable, the hosts create a P2P network and aid each other to extract the resource. Consequently, the server can serve much more hosts as compared to that of the server-client model.

ALM is proposed to avoid multicast support in routers by executing multicast-related functionalities in the application layer of the host. A P2P network is created between the source and all the receivers. Then, a multicast tress is constructed over the P2P network.

In a similar way, the messages are transmitted from the root to the leaves. On the level of the physical network, the messages are transmitted via unicast along the paths designated by the tree. As traditional multicast routing executes multicast support in the network layer, often it is denoted as network-layer multicast.

ALM is a great alternative for multicast communication over a large-scale network. It does not require the router support; hence it is easy to deploy. In theory, it can support infinite receivers. As receivers support forwarding messages for each other, the more the receivers, the higher the total uploading bandwidth.

## 5.3. NETWORK CODING (NC) FOR MULTICAST NETWORKS

NC is recently proposed as a generalization of routing. Routing allows relay nodes to duplicate or forward messages. As NC allows relay nodes to encode messages, duplicating or forwarding is considered as a special case of encoding (Figure 5.2).


**Figure 5.2:** Network coding for multicast networks.

NC has a great potential application for wires as well as wireless networks both. Multicast is one of the key applications of NC. With NC, a multicast network can attain its maximum throughput.

Depending on the type of the encoding function at the relay nodes, NC can further be categorized into two types:

1. **Linear Network Coding (LNC):** Encoding functions are linear functions; and
2. **Non-Linear Network Coding:** Encoding functions are non-linear functions.

Li et al. proved that LNC is enough for a multicast network to attain its maximum throughput.

## 5.4. LINEAR NETWORK CODING (LNC) FOR MULTICAST

The Max-Flow Min-Cut Theorem: It provides a simple way of calculating the maxflow just by finding the minimum cut. In a unicast session, the maximum transmission rate between the source and the receiver is the actual max-flow between them.

Multicast capacity, in a multicast session is defined as the minimum of the max-flows between the source and the receivers. If the source transmits the message to all the receivers at the same rate, it is really easy to observe that the multicast capacity is the upper bound of the transmission rate.

As a result, multicast capacity is considered as the maximum throughput that a multicast network can attain. With NC, the maximum throughput can be attained for a multicast network, that is, NC can completely take advantage of multicast capacity (Figure 5.3).


**Figure 5.3:** Linear network coding for multicast.

LNC is enough to attain this aim and even a stronger result: with LNC, each receiver can get the messages at the rate determined by the max-flow between the source and the receiver at the same time.

In a LNC assignment W, the rank of a node v, $rank_v$. refers to the rank of a linear space spanned by the global coding vectors for incoming edges of v, i.e.:

$rank_v(W) = rank\ \{q_e(W), head\ (e) = v\}$

For a receiver to decode the messages received successfully, the rank of the receiver should be no less than that of coding dimension h. A valid LNC assignment is basically a LNC assignment that all the receivers can easily decode.

A capacity-achieving LNC assignment for an acyclic graph G = (V, E) exists if and only if we can assign vectors to the edges that satisfy:

$$q_e \in \text{span}\{q_{e'},\ \text{head}(e') = \text{tail}(e), \forall e \in E\}$$

$$\text{rank}\{q_{e'},\ \text{head}(e') = t\} = \text{multicast\_capacity}, \forall t \in T.$$

## 5.5. DETERMINISTIC NETWORK CODING (NC) VS. RANDOM NETWORK CODING (RNC)

To apply LNC to multicast networks is to get a capacity-attaining LNC assignment for a multicast network. There are basically two ways of fulfilling this:

- Deterministic network coding; and
- Random network coding (RNC).

Deterministic NC basically adopts a centralized method of calculating the encoding mixing coefficients for each node in the network given the information related to network topology, source, and receivers. Jaggi et al. proposed a polynomial deterministic LNC construction algorithm.

Once the LNC assignment is determined, it is well distributed around the network. Then, each node encodes the incoming messages depending upon the LNC assignment during the complete session of multicast.

As the LNC assignment is calculated depending upon the complete information regarding the multicast session, all the receivers are certain to be capable of decoding properly. In addition to that, the needed field size can be as small as the number of receivers.

One of the drawbacks of deterministic LNC is its dependency on the stability of the system topology. Once the typology is altered, the complete LNC assignment requires to be calculated again.

On the other hand, RNC makes use of a complete distributed way of determining the encoding mixing coefficients for each node. Even, each node does not need to gather any kind of local information. The coefficients are generated in a random way for each node.

The random coefficient is linked to the corresponding encoded messages. After getting the encoded messages, the relay nodes again encode them using a set of new generated random coefficients and replace the coefficients in the message with the new ones.

Eventually, the receivers will try to decode the messages depending on the coefficients linked in the messages. Because of the coefficient randomness, there is a non-zero probability that the receiver cannot decode effectively. In such cases, the receiver has to receive more messages in order to perform decoding.

It is clear that, if field size is increased for the coefficients, the chances of failing to decode is decreased to a greater extent. The strength of RLNC is its resilience under the dynamic network topology. The change in topology has no impact on the generation of coefficients as each node can generate the coefficients randomly as well as independently.

## 5.6. NETWORK CODING (NC) IN PEER-TO-PEER FILE SHARING (PPFEED)

NC can be regarded as a great improvement of routing to enhance network throughput and offer high reliability. Further, it allows a node to generate output messages by encoding its received messages.

P2P networks are the most appropriate place to apply NC because of the two key reasons-the topology of a P2P network is arbitrarily constructed and thus easy to adopt the topology in order to allow NC; the nodes in a P2P network are end hosts that can actually perform much more complex operations like encoding and decoding than just storing and forwarding messages.

There are various schemes to apply NC to P2P file-sharing that employs a P2P network in order to distribute files resided in a web server or a file server. The schemes exploit a special kind of network topology known as combination network (Figure 5.4).


**Figure 5.4:** Network coding in peer-to-peer file sharing (PPFEED).

It is evidenced that the combination networks can attain an unbounded NC gain calculated by the ratio of network throughput along with NC to that without NC. The scheme encodes a file into multiple messages and then divide the peers into multiple groups, with each group being responsible for relaying one of the messages.

The encoding schemes are mainly designed to content the property that any subset of the messages can be utilized in order to decode the original file as long as the size of the subset is large enough. To meet this need, first of all, a deterministic LNC scheme is defined which satisfies the desired property.

After that, peers are connected in the same group to flood the corresponding message and then connecting peers in different groups in order to distribute messages for getting decoded. In addition to that, the scheme can be extended readily in order to support topology awareness to enhance the performance of the system in terms of reliability, throughput, and link stress.

Already there are P2P systems available, each one with different kinds of approaches as well as services. Further, P2P systems can be divided into two different types:

- Structured peer-to-peer systems; and
- Unstructured peer-to-peer systems.

In the case of structured P2P systems, the connection amongst the peers in the network are fixed. Basically, peers maintain the information regarding the resources such as shared content, that their neighbor peer contains. As a result, the data queries can effectively be directed to the neighbor peers having the required data and information, even if the data is very rare.

While on the other hand, in unstructured P2P systems, the connection amongst peers in the network are arbitrarily formed in either flat hierarchical manners. To find as many peers that possess the required content as possible, peers in unstructured P2P systems query data based on different techniques like flooding, expanding ring, etc.

Further, there are three different designs of unstructured P2P systems:

- Centralized unstructured peer-to-peer systems;
- Decentralized (or pure) unstructured peer-to-peer systems; and
- Hybrid unstructured peer-to-peer systems.

In a centralized unstructured P2P system, a central entity is basically utilized for the purpose of indexing and bootstrapping the whole system. An example of a centralized unstructured P2P network is Bit Torrent Network. Napster is another example which pioneered the idea of P2P file sharing.

In Napster, a server or server farm is used to offer a central directory. A peer in the network informs the directory server of its IP address along with the names of the contents which makes it easily available for sharing. Hence, the directory server knows which kind of objects each peer in the network actually has and after that, develops a dynamic and centralized database that efficiently maps the content name into a list of IPs.

The decentralized or pure unstructured P2P network is an overlay network. An overlay network can be defined as a logical network. As edge in this network is present between any pair of the peers that preserves a TCP connection. An example of a decentralized unstructured P2P network is Gnutella.

To join the Gnutella network, initially, a user connects to one of the various known bootstrapping peers. Then, the bootstrapped peers respond with the information regarding one or more peers present in the overlay network. Such information comprises IP address along with the port of each peer. In Gnutella, the peers are only aware of their neighbor peers.

A hybrid unstructured P2P network permits the existence of infrastructure nodes, generally referred to as super-peers or overlay nodes or supernodes. This eventually develops a hierarchical overlay network that addresses the problems related to scaling on pure unstructured P2P networks like Gnutella.

In such networks, a peer can usually change the roles over time. An example of a hybrid unstructured P2P network is $K_aZ_aA$. In order to overcome the limitations of the current network system, a proposed system of P2P file sharing using NC is executed. In literature, NC has evolved as a major information-theoretic approach which can be used to enhance the performance of P2P networks and wireless networks both. On a large scale, it has been accepted as well as acknowledged theoretically that NC can improve the network throughput of multicast sessions in acyclic graphs, attaining their cut-set capacity bounds.

Some of the recent studies too have supported the claim that NC is advantageous for large-scale P2P content distribution, as it deals with the issue of locating the last missing blocks for completing the download. The proposed system makes use of link capacity by using encoding message in order to get max-flow through link.

As per the theory, file sharing based on the NC has proven to be effective. The effect of NC can be realized in large networks. BitTorrent is another P2P file-sharing network, which splits the file into pieces. However, it has some key limitations. It is a decentralized P2P network, and therefore, it is quite hard to administer. In addition, it also causes degradation in internet speed as well as overall computer performance. The proposed network deals effectively with such limitations, just by proposing new protocols.

## 5.7. DETERMINISTIC LINEAR CODING OVER COMBINATION NETWORKS

### 5.7.1. Peer Joining

Let us assume that the server is well-known whose IP address is known to all peers by some kind of address translation services like DNS (domain name system). When a peer wants to recover a file hosted by the server, it initiates a join process by sending a JOIN request to the server.

### 5.7.2. The Server Maintains Different Lists and Counters

The server maintains various counters and lists. For each group $G_i$, the server maintains a counter $gc_i$ to store the number of peers in the group and a list $gl_i$ to store the list of clusters which include the peers in the group. For each cluster $C_i$, the server maintains a list $cl_i$ to store the groups which include peers in the cluster. In addition, the server maintains a list of existing peers along with their respective residue upload bandwidths and IP addresses for each group.

As a greater number of peers join the system, it is resource consuming to sustain a complete list of peers for each group. The server will sustain a partial list of peers with the major residue upload bandwidths. At the same time, peers will report to the server their upgraded residue upload bandwidths occasionally to upgrade the partial list on the server.

However, the server is largely responsible for bootstrapping the peers, it will not be the holdup of the system as once each peer gets the list, it communicates with other peers for the purpose of data dissemination and topology construction.

When the server gets the peer's join request, it allocates the peer to a group. After that, the server sends the list of peers of that particular group to the joining peer and also updates the number of peers in that group. The server will allocate the new peer to a group depending upon some of the following factors given below.

First, the peer is allocated to a cluster depending upon its coordinate. If the number of groups which comprise peers in the cluster is less than the new peer, then it will be allocated to one of the groups that comprise no peers in the cluster. When there are various groups like this, the group spans the least number of clusters.

Tie is broken by selecting the group having a smaller number of peers in the group. The validation behind this is the need to reduce the logical links between different clusters and assure that peers can get enough innovative messages, that is, messages from different groups, in the cluster in order to perform decoding.

Peer Joining Algorithms:

INPUT: joining peer v

OUTPUT: updated overlay network

BEGIN

//suppose the cluster corresponding to peer v is $C_i$

If $|cl_i| < k$

$S_i$ = the set of groups not in $cl_i$;

Pick a group $g_i$ S such that $gl_i$ is the smallest;

If multiple groups have the same smallest $g_l$, pick a group $g_i$ with less

$gc_i$;

Peer v is assigned to group $g_i$;

END

After getting the list of peers, the new peer will contact them and then develop create overlay links with them. Such peers are known as intra-neighbors of the new peer as they are within the same group. While the neighbors which are in different groups are known as inter-neighbors.

The new peer requests one of its intra-neighbors in order to provide a list of its inter-neighbors. While picking the intra-neighbor, high priority is given to the peer in the similar cluster. Then, the new peer takes the list of peers as its inter-neighbors.

The topology of the P2P network can be regarded as a key combination of multiple unstructured P2P networks, each of which comprised of the peers within the same group. The topology within one group is random as long as it is connected.

The only limitation is on the edges between distinct groups. It is needed that each peer is connected to at least k–1 peer in k–1 distinct group respectively. Here, k is multicast capacity of network. When more than k–1 peer is connected together, the system reliability can be significantly enhanced.

### 5.7.3. Peer Leaving

There are two kinds of peer leaving:

- Friendly; and
- Abruptly (terminating or changing unexpectedly).

For the friendly leaving, the leaving peer will start a leaving process by sending the LEAVE messages to its intra-neighbors as well as inter-neighbors. So that the system is aware of its leaving and can make important updates consequently.

For the abruptly leaving, the leaving peer will start by leaving process not send any notification messages to both of its intra-neighbors and inter-neighbors. This is mainly because of the link crash or computer crash.

## 5.8. COMPUTING THE OPTIMAL ROUTING STRATEGY

In computing the optimal routing strategy, the key aim of increasing end-to-end session throughput is achieved in two phases:

- Constructing the transmission topology; and
- Designing the suitable coding strategy for data dissemination (by using a randomized code assignment algorithm).

The NC theorem establishes the fundamental connection between multicast flow routing and network flows. As a result, the computation of the multicast rate and the optimal multicast topology is discrete into a number of maximum flow computations (Figure 5.5).


**Figure 5.5:** Computing the optimal routing strategy.

The maximum attainable throughput of a multicast session is the smallest throughput amongst all source-destination pairs. Here, the question arises how the data should be disseminated and coded. In the next section, algorithms for all phases are given that might be applied realistically to compute the optimal transmission topology for coded overlay flows.

### 5.8.1. The Maximum Flow Problem

In the theory of network flows, maximum flow is quite a well-studied issue. Given a directed network G= (V, A) and nodes u, v ∈ V, the maximum flow from u to v is considered to be the maximum rate at which flows can be shipped from u to v along capacitated arcs in directed network, G.

In the problem of min-cost flow, which is quite a more general version of the max-flow problem, a definite cost is linked with every unit flow which gets shipped through an arc. In addition, the given flow rate needs to be attained while introducing the minimum link costs.

A min-cost flow algorithm can be used to compute the maximum flow just by incorporating a virtual arc (in general, a feedback link) from the receiver v to the sender u with cost-1, while on the other hand, setting all other costs to zero. An ϱ-relaxation-based algorithm is employed in order to compute the max-rate multicast topology with minimum bandwidth consumption.

The algorithm is amenable to fully asynchronous and fully distributed implementations. In the notation, each link (i, j) ∈ A is linked with bandwidth capacity $b_{ij}$, $f_{ij}$ is the flow rate from node i to node j, $c_{ij}$ is the cost of transmitting a unit flow via link (i, j), gi is the flow excess on node i, and pi is the dual variable acting as unit price charged for flow excess at node i.

### 5.8.2. Computing the Transmission Topology

Towards attaining maximum rate multicast transmission, the first step is to compute a routing topology signifying the amount of bandwidth needed on each link in the network. Given this topology, flows are assigned on each link as per the allotted bandwidth and then ultimately transmit the data. In this section, the focus will be on computing the transmission topology and allocation of bandwidth.

Unicast sessions: In unicast sessions, each link (i, j) ∈ A is initialized with flow $f_{ij} = 0$ and cost $c_{ij} = 0$. After that, a feedback link is added with bds = f d s = α, and c d s = –|D|, where α is a constant with any value known to be larger than the attainable maximum flow rate and |D| is the maximum diameter of the network (in number of hops).

For each node i, the flow excess gi is calculated as P (j, i) ∈ A $f_{ji}$ – P (i, j) ∈A $f_{ij}$, and the price pi is initialized to 0. After the process of initialization, each node having a positive flow excess ($g_i$> 0) implements the algorithm. The algorithm terminates when every node has zero flow excess.

Multicast sessions: In a multicast session, data are sent to a group of interested receivers from the source, at the same rate in the overlay. In order to attain maximized multicast throughput, the first need is to recognize the maximum attainable throughput between each source-destination pair, by using the algorithm described above for unicast sessions.

Given the maximum attainable throughput for every destination, the throughput of a multicast session links to the smallest throughput attainable to all the destinations. As the maximum throughput for each destination may be distinctive from each other, they are highly required to be reduced in order to match the dominant multicast rate of flow.

A set of variables are used in order to maintain the status of each link in context to each destination. The bandwidth capacity and cost are still denoted by $b_{ij}$ and $c_{ij}$, respectively, on each directed link (i, j). let f $k_{ij}$ be the rate of flow on the arc (i, j) serving different destinations k, g $k_i$ be the flow excess on the node i in serving destination k, while p $k_i$ be the price on node i in serving destination k.

However, the min-cost flow algorithm remains unchanged, except that the algorithm is independently applied for each of the source-destination pair. When the min-cost flow algorithm terminates for each and every destination, the maximum attainable throughput $f_k$ of a source-destination pair is the flow rate on the feedback link.

The maximum attainable throughput is fmax = min ($f_k$). In order to tune the transmission topology to conserve needless bandwidth, the flow from the source to destination k is reduced by δ = $f_k$ – fmax. The flow reduction process is initiated by reducing the flow on each feedback link by δ.

## 5.9. MULTICAST IN CLOUD NETWORKS

In a cloud environment, there are many applications that require broadcasting or multicasting messages. There are many applications in cloud environments that require broadcasting or multicasting messages. Some of these applications are:

- To send broadcast or multicast messages into the cloud network, user applications drive their host virtual machine (VM). To satisfy the needs of distributed databases, these packets can be generated, file-sharing services, audio/video streaming, or audio/video conferencing.

- By flooding it through the network to get information about how to map the destination's logical IP address to the corresponding physical IP address, a VM sends out a broadcast or multicast message.
- To support standard protocols, a VM sends a broadcast or multicast message into the network, e.g., address resolution protocol (ARP). For instance, some data center virtualization standards such as VXLAN and NVGRE convert broadcasting in the virtualized subnet into multicasting in the physical network.

There is a method by the name one to all method in which a common approach is followed to multicast in cloud environment. In this method, sources establish the connections to all the destination nodes (DNs) sends/unicasts the message to them either one by one or simultaneously (Figure 5.6).


**Figure 5.6:** Multicast in cloud network.

To reduce the distribution, delay this approach does not use the upload capacity of the DNs and along with this, approach puts lots of pressure on some physical links such as the one connected to the network interface of the source node (SN) (several copies of the message have to go through that link).

Though, because of its simplicity, the above approach is attractive, and there is only one node which sends the packets. That one node is termed as SN. DNs do not participate in sending or forwarding packets.

It was revised by Chu. et al. that whether the multicast related services should be implemented in the network layer or the application layer. A comparison is made by the author by using both simulations and experiments on the Internet to compare the performance of their ALM solution with that of IP Multicast.

So, after the comparison, the result indicates that in small to medium-sized multicast groups, ALM can achieve low link stress. Narada is the term refers to the maximum link stress in their ALM algorithm and for the multicast group size of 16, it was reported 5 and 7 for the multicast group size of 100. The results show that the maximum link stress can always be capped at three at the time when ALM is done carefully. This means ALM is measurable with the regards to link stress.

## 5.10. NETWORK CODING (NC) MULTICAST IN SATELLITE NETWORKS

Content distribution has been a major satellite-based service for so many years because of the inherent broadcast nature of satellite communications. Terrestrial-based alternatives have acquired grip recently, mainly because of the decreasing cost of the broadband communications, linked with disruptive technologies like P2P content distribution.

On the other hand, satellite-based platforms present various key benefits that are a bit hard to match in a straightforward way. First of all, the scalability is assured because of the broadcast nature of the satellite communications, while considering the multi-beam satellite systems.

Secondly, the independence of terrestrial infrastructure is provided by the satellite coverage ubiquity. Third is that typical satellite networks present a kind of star topology that actually simplified the multicast services as there is no routing. Finally, assured bandwidth is generally inherent to most of the satellite systems that eventually makes it basic in developing a content distribution platform (Figure 5.7).


**Figure 5.7:** Network coding multicast in satellite networks.

In general, the disruptive technologies are at the core of the modern content distribution concerns and solutions. In relation to the communication networks and protocols, NC provides an arguably interesting concept of networking: data throughput and the network robustness can considerably be improved by allowing the intermediate nodes to mix distinctive data flows in a network via algebraic combinations of different datagrams.

Such a major idea, which breaks with the standard store-and-forward paradigm of current routing solutions is especially valid for satellite networks where guaranteed patterns in huge terminal populations can be disputed by broadcasting encoded packets to different nodes at the same time until the DNs have sufficient level of freedom, to decode as well as recover the original data.

### 5.10.1. Network Coding (NC) in Wireless Networks

It is common in wireless networks to have multiple connections because of the broadcast nature of the network. Just one path is selected with the protocols of routing and therefore restricting the capacity to the attainable capacity in that path.

With the application of NC, it is quite possible to reach the min-cut capacity just by transmitting the linear encoded symbols via different paths. Nevertheless, the messages are decoded when sufficient linear independent messages have been received.

This ultimately links to the min-cut capacity, that is to the maximum flow between two different nodes when all of the possible paths are used. Multicasting over the wireless networks can be improved efficiently with NC.

Then the messages are encoded linearly so that the receivers can easily decide them after enough independent symbols have been received. This basically allows the receiver to moderate the erasures mainly when a feedback channel is present. In addition to that, linearly encoded symbols can be simultaneously optimized for all of the receivers as per the actual patterns.

### 5.10.2. Network Coding (NC) in Satellite Networks

Let us now consider a content distribution multicast scenario for satellite network topologies. There are various return channel technologies in the satellite network, however the return channel over satellite is the most common reference scenario. Usually, these satellite networks are characterized by having extremely asymmetrical links with long round-trip delays. In context to fixed satellite terminals, the rates of erasure are quite small, though slow fading events caused by the weather conditions can affect the group of terminals within the same geographical region.

The rates of erasure can vary from the unidirectional satellite systems, if in case, the fade mitigation techniques (FMT) are used. These can effectively be used in order to adapt the signal protection levels, which ultimately can affect the erasure rates along with the available capacity.

It is predicted that the multicast services make use of population management techniques so as to assure low rates of erasures for most of the receivers. This is possible only with the FMT techniques. Either with fixed or variable capacity systems, NC is highly advantageous as it can provide time-diversity for non-real-time services by integrating prior transmitted symbols with the new ones.

In addition to that, it can make effective use of the feedback channel in order to optimize the linear combinations as per the specific erasures reported by the receivers. While employing multicast techniques of population management, it is possible to regulate the performance as NC decreases the need of high protection levels at the physical layer and therefore increasing the available capacity.

In context to mobile satellite terminals, both of these are highly affected by slow fading events that occurred in fixed satellite terminals together with fast-fading events that are the key features of mobility-related impairments.

NC can offer short-scale time diversity that ultimately allows the receivers to overcome the patterns of erasure caused by fast fading events. Nevertheless, the feedback channel can be utilized to optimize the linear combinations as per the specific erasures reported by the terminals.

So, because of the fast-fading events that are distinctive of the mobile satellite terminals, only the approaches based on FEC were possible along with traditional approaches. As a result, the approaches of NC can present great performance gains with the elimination or reduction of the FEC-related overhead.

It is likely to introduce few NC redundancies similar to FEC, so as to decrease the erasures and restrict the use of the feedback channel. This may be endorsed for networks where the return link resources are limited which is a bit common for the satellite network.

In addition, they can also give local content which is not carried over the satellite. Multipath capabilities are forecasted for satellite networks, for example, in the railway scenario where train stations, tunnels, and urban environment produce channel losses for a long period of time.

In such a scenario, NC can represent substantial improvements by taking advantage of multiple capabilities. Independent linear combinations diffused via multiple paths allow the receivers to decode the messages as early as it has received enough encoded symbols. It actually means that multiple paths not just provide redundancy but also an improved performance by attaining the min-cut capacity.

## 5.11. RELIABLE MULTICAST FOR FIXED AND LAND-MOBILE SATELLITE SERVICES

### 5.11.1. Transmission Reliability

A key goal or aim of any modification of a system that must provide reliable multicast is to reduce the chances of erroneous reception of these services. Transmission reliability for multicast is hard from the packet error rate (PER) approach of broadcast, where one sets a tolerable PER, which should hold for all users, however the erroneous receptions or outages are classically at different points in time for different users.

For reliable multicast, a transmission gets failed, if even just one user of the group did not successfully get all of the content. Therefore, the chances of unsuccessful transmission $P_{un}$, and thus the need to transmit additional information, scales with the number of receivers r:

$$P_{un} = 1 - p_{suc}{}^{r}$$

where; $p_{suc}$ is the probability of individual successful reception.

This outcome for multicast services in the requirement of improved forward error correction (FEC) schemes, which can assure a lower PER than for the broadcast or for the point-to-point case, if one wants to attain a high reliability (Figure 5.8).

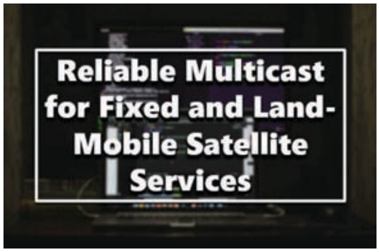



**Figure 5.8:** Reliable multicast for fixed and land-mobile satellite services.

### 5.11.2. Retransmission Strategies

Completely reliable data transmission cannot be attained by pure FEC, as for the realistic systems, there is always a likelihood larger than zero for the unsuccessful reception. A solution is to make use of automatic repeat request (ARQ) schemes. These schemes are also used for point-to-point connections.

In literature, the optimized strategies, that are based on combinations of FEC and ARQ are usually denoted as a hybrid ARQ (HARQ) scheme. HARQ can be classified into two categories: type-I and type-II.

Type I HARQ encodes each data packet by means of a combination of codes for error detection and correction. In case, the receiver fails to decide the packet, it will eventually request a retransmission of the erroneous packet.

On the other hand, the hybrid Type II scheme can be defined as an adaptive ARQ system. In case, the errors are detected in the data packets that are received, then they are not discarded, but the receiver saves them in buffer and requests the gender for extra redundancy developed by an error-correcting code.

Then the receiver makes use of the erroneous packet along with the packet comprising redundancy to reconstruct the original information. A key difference between multicast services and point-to-point connections is that the multicast services retransmission can be received by all the users.

As a result, it is not ideal to resend missing packets individually, but to send additional information that can be used by each and every receiver. A coding scheme which can make use of this feature is proposed in the section below.

### 5.11.3. Packet Level Coding

It is proposed to add a FEC mechanism at the packet layer. At this layer 'h,' the redundancy packets are added, in a similar manner as the redundancy bytes are added in an individual DVB-S packet by the Reed-Solomon (RS) code.

This ultimately results in a double structure where normal DVB-S channel encoding is done for each packet and on the top of its 'h' redundancy packets are added to the 'k' normal DVB-S data packets. Eventually, this results in the transmission of n = k + h packets.

Using maximum distance separable (MDS) codes, similar to the RS, allows each and every receiver to reconstruct the original information if at least 'k' out of 'n' packets of a packet group are received. As a result, the receiver can deal with erasures as long as they result in a total loss, not surpassing 'h' packets. One essential capability for this scheme is to consistently detect erroneous packets. This can be done at the physical layer through the RS code of the DVB-S channel code.

The RS-Decoder can effectively determine if its error-correction capability is exceeded. This characteristic is a good standard to detect the erroneous packets if the number of errors before a RS-Decoder is small. In case, if the number of errors at the input is too large, the likelihood that an erroneous packet is corrected into a wrong, but legal, codeword increases.

For such a scenario, the capability of error correction of the RS-Code could be decreased, in order to increase its precision for detecting the errors. In addition, one should detect, if because of the slow fading, the signal level is too low for correct reception. The availability of erroneous packets can be signaled through the Transport Error Indicator of the MPEG-transport stream (TS) header. Another key point is the knowledge of the position of each packet in the stream. This can be done by the continuity counter field (CC), however based on the execution, an additional byte overhead has to be added before the payload, to upsurge the range of the counter.

Concerning the protocol stack, there are different possible implementations. Either a fixed size (k, n) packet group can be utilized, where the MPEG TS is segmented into different groups of constant length for which redundancy is added, or, the coding can be done at the application level, where each file is encoded as one large packet group. Despite the fact, the second approach is more optimal in principle, the variable-length generates problems for RS-based approaches and alternative codes are then required.

### 5.11.4. Alternative Coding Approaches

As an alternative to using a RS code, one can effectively use other codes that are no longer separable, and thus, to receive more than 'k' packets in average, they can decode all erasures correctly. An interesting approach is the use of irregular low-density parity-check codes; they can be generated as well as decoded for different file sizes, where the sender and the receiver only need to utilize the same pattern for encoding or decoding.

If one allows a certain loss in performance, the decoding and encoding complexities increases linearly with the length of the packet by n ln(1/²), which makes it suitable for long file transfers. The number of required packets $h_{elp}$ to reach the same performance as the RS-Code increases to:

$$h_{lp} = h + \epsilon \cdot n.$$

### 5.11.5. Effect of Packet Level Coding on Retransmissions

In multicast case, one has to consider the effect of the additional requested transmission not only on a single receiver nut on a group of receivers. For packet-level coding, the one and only criterion for successful reception is that 'k' out of 'n' packets are decoded in the correct manner, independent of the ordering.

If additional redundancy is desired, new coded packets $h*$ are broadcast to all users. This not just help the receiver, who requested the information, but all the receivers who missed the information.

After the packets have been sent, all of the users now can decode the transmission successfully as they have received k out of $k + h + h*$ packets correctly. Also, this feature decreases the required data rate in the return link, as a receiver no longer has to transmit back the identification number of all erroneously received packets, but just the total number of packets that are missing. At the satellite gateway station, the negative acknowledgments (NACKs) are received, and the multicast protocol only need to consider the maximum number of missing packets, sending an apt number of redundancy packets.

A key point to be considered is that sufficient redundancy packets 'h' is available. If one would not have packets, one could re-transmit the packets sent previously again. Then, such retransmissions would be no longer important for all the receivers but only too few of them where the specific packet is not available. Eventually, this would reduce the efficiency of the ARQ.

## 5.12. PROTOCOLS AND ALGORITHMS

Protocols: in general, satellite systems make use of encapsulation protocols to carry the network datagrams, named as IP packets. The need for an encapsulation protocol for supporting the NC are bit straightforward-sequence or generation number, coefficients matrix and payload.

Using the same methodology for the FEC support over multiple protocol encapsulation (MPE), it is probable to redefine some header fields that are not utilized in data cast applications. In case of generic stream encapsulation (GSE), the mechanism of extension header allows for the introduction of novel features of characteristics like NC support.

The use or application of the current encapsulation protocols for supporting NC allows for simplified deployment. In addition to that, the location of such protocols below the network layer in the protocol stack allows for NC to be transparent to the higher layers such as the network layer.

NC algorithms: The NC algorithms are considered as the key element for acquiring the promised gains of the performance. Further, some of the fundamental needs are mentioned which are required while designing NC algorithms for the satellite networks.

First of all, the transmission must have a low decoding delay that is similar to the intermediate performance of the fountain codes. This low decoding delay is mainly needed for near real-time services, while it carries a small loss of efficiency.

It has already been shown that, for fountain codes, the intermediate performance for a very a smaller number of received coded symbols is gained with a degree of d=1. In context to NC, this means that most of the symbols must be transmitted as it is that is without combining them with other symbols.

The second key requirement is that the symbols which were already seen must not be again transmitted. This can be attained by using a sliding window mechanism, the scheduler will just transmit the symbols within this sliding window. The last requirement is that the error recovery process must use a linear combination in a manner that each and every symbol can be recovered completely with a minimum number of transmitted symbols.

### 5.12.1. A Simple ALM Algorithm

As discussed earlier, it is slower in application layer than in the network layer, a single packet multicast can be a factor. The implementation results in that fast when the message is large ALM can be fairly fast (e.g., when the message is a large file). Efficient pipelining of packets is the reason behind the question that ALM is fast when the message is large.

BitTorrent's tuned version is the algorithm, in which the number of connections of each peer (node) is set to the maximum (i.e., it is set to n in the SN, and n–1 in the DNs), and the message is divided into an equal chunk.

Connecting to all n DNs the algorithm starts by the source, and in parallel sending packets from a distinct chunk to everyone. A DN opens connections to all the remaining DNs, and starts forwarding the packets received from the source to all those nodes after receiving the first packet of the chunk from the source.

About the network topology, the above algorithm does not require any kind of information. However, the algorithm's networking load is at most six times that of any NLM, as the algorithm ensures that only one copy of a packet is sent to any DN.

### 5.12.2. A Simple Analysis of the ALM Algorithm

At the SN, the size of the file is denoted. This can be explained more by considering a simple model in which sending the file from any node to any other node takes t seconds. Also, suppose that transferring a file of size size/n from one node to any other node can be done in t/n seconds (Figure 5.9).


**Figure 5.9:** A simple analysis of the ALM algorithm.

The real environment is always different from assumption, so the above mention assumption in the real environment may not be accurate as different links may have different capacities and different congestions. From one node to another node, the nodal delay can vary. The hop distance between a pair of nodes may be different than that of another pair. The nodal delay can vary from one node to another node. Also, the hop distance between.

Therefore, after t seconds, every DN has received all the packets of a distinct chunk of the file. A DN starts forwarding packets from the source to all other DNs as soon as it receives the first packet from the source.

Every DN needs to forward a chunk of size ‘f’ to ‘n–1’ other node; this takes about ‘t’ seconds. Therefore, the whole process of disseminating the file would take about t seconds as the source and DNs send packets simultaneously. In other words, by the simplified model, the tuned-version of BitTorrent needs about t seconds to distribute the file, which is about the time needed to send the file to a single DN!

## 5.13. CONCLUSION

In conclusion, this chapter has proposed a set of protocols and algorithms in order to enhance the end-to-end multicast session in a considerable amount. This chapter also proposes the application-layer multicast algorithm (ALMA), and it has been seen that it is arguably the best application layer protocol with respect to most of the metrics that are considered in ad hoc networks. ALMA relies on developing a logical tree in the ad hoc network.

In this chapter, it has been discussed that several various kinds of NC can be used for multicast networks such as LNC. This chapter also discussed the major differences between deterministic NC and RNC, along with the application of deterministic linear coding over the combination networks.

Towards the end of the chapter, various strategies have been discussed in order to evaluate the optimal routing, and that can be achieved with the help of computing the transmission topology. Also, it has been observed that there are various multicasts that are reliable for fixed and land mobile satellite services.

# REFERENCES


1. Choudhary, A., Akhade, N., Narke, A., & Deshmane, A., (2015). *Peer to Peer File Sharing Using Network Coding*. [Online] Mjret.in. Available at: http://www.mjret.in/V2I3/M6-2-3-7-2015.pdf (accessed on 3 May 2021).
2. Ernst, H., Donner, A., & Shabdanov, S., (2003). *Reliable Multicast for Fixed and Land-Mobile Satellite Services*. [Online] Researchgate.net. Available at: https://www.researchgate.net/publication/224794102_Reliable_Multicast_For_Fixed_And_Land-Mobile_Satellite_Services (accessed on 3 May 2021).
3. Vieira, F., & Barros, J., (n.d.). *Network Coding Multicast in Satellite Networks*. [Online] av.it.pt. Available at: http://www.av.it.pt/conftele2009/Papers/14.pdf (accessed on 3 May 2021).
4. Vijay, R., & Chirchi, V., (2015). *File Sharing between Peer-to-Peer Using Network Coding Algorithm*. [Online] citeseerx.ist.psu.edu. Available at: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.741.9467&rep=rep1&type=pdf (accessed on 3 May 2021).
5. Wang, M., Li, Z., & Li, B., (n.d.). *A High-Throughput Overlay Multicast Infrastructure with Network Coding*. [Online] Iqua.ece.toronto.edu. Available at: https://iqua.ece.toronto.edu/papers/mwang-iwqos05.pdf (accessed on 3 May 2021).
6. Yang, M., & Yang, Y., (2008). *Peer-to-Peer File Sharing Based on Network Coding*. [Online] Ieeexplore. Available at: https://www.researchgate.net/publication/4365068_Peer-to-Peer_File_Sharing_Based_on_Network_Coding (accessed on 3 May 2021).

**CHAPTER 6**

# THROUGHPUT BENEFITS OF NETWORK CODING

## CONTENTS

In this chapter, throughput benefits of network coding (NC) and the actual functioning of throughput are discussed. The chapter also highlights the meaning of randomized NC and how it works. It also explains vector NC algorithms with several formulas.

In addition, it also demonstrated code design algorithm in which code design for vector coding, code design for scalar coding and vector solutions with the help of subspace codes is explained. Furthermore, it also shed some light on the average throughput coding benefits. It also explained the throughput benefits of NC for communication. In the end, it explains MAC-layer random network coding (MRNC) and throughput analysis.

## 6.1. INTRODUCTION

Within the past few years, there has been a significant increase in the interest in understanding the possible performance benefits ensuing from the use of network coding (NC) in multi-hop wireless environments. Especially, majority of the military battlefield scenarios display two features that believes to stimulate the use of NC: (a) the dependence on bandwidth-constrained, ad hoc wireless links (e.g., making use of MANETs produced by vehicle attached radios in urban insurgencies) and (b) the necessity to distribute data and information (e.g., mission commands, maps) to numerous recipients.

The preliminary outcomes on the dominance of NC, such as the original validation of Ahlswede et al. of how in-network mixing of packets by temporary nodes aids to attaining a communication capacity that is not practicable exclusively through routing were attained for the case of a wireline, lossless network.

In the present scenario, several groups have examined the probable performance gains achieved by NC for both multicast and unicast traffic in wireless environments, for a number of application setups.

The majority of these approaches profoundly aimed at manipulating the wireless broadcast advantage (WBA) by making use of, whenever possible, a solo link-layer broadcast transmission (of a packet which is created by a linear grouping of separate packets) to get manifold neighboring nodes.

By reducing the totality of autonomous transmissions required, network-coding approaches efficiently lessen the part of time the wireless channel is held by a solo transmitting node, resulting in enhancing the throughput of the overall network.

In addition, it is also believed that there is another degree of freedom in wireless environments, explicitly link-layer rate diversity, which is so far failed to exploit by the NC approaches. It is now seen that majority of the commodity wireless cards has the potential to adopt modulation to differ the link rate in reply to the signal-to-interference levels at the receiver.

Link rate multiplicity archetypally demonstrates a rate-range trade-off: in case the identical transmission power is used for entirely link transmission rates, then, overall, the quicker the transmission rate, the lesser is the transmission range (although, the rate-distance deviation in real life is fairly uneven.

While this rate variety has been broadly utilized for unicast traffic and is frequently standardized, its use in link-layer broadcasting is comparatively restricted. For instance, while the present IEEE 802.11a/b/g standards authorize the transmission of the control frames (e.g., CTS/RTS/ACK) at the lowermost rate (e.g., 6 Mb per second for IEEE 802.11a), transmission rates for broadcast data are most of the times application-specific.

Although, currently, there has been some work that validates that effectual manipulation of such rate diversity by directing algorithms for link-layer broadcasts can conducive to substantial (often 6-fold) diminution in the latency rate of broadcast and surge in the overall attainable throughput.

The research around NC was inspired by the seminal paper that established that, overall, the use of in-network encoding of packets could achieve an ideal capacity that cannot be comprehended via any viable routing-only scheme.

For multicast traffic, the 'capacity' is outlined as the greatest data rate that a sender can send to all associates of a set of receivers. It is given by the lowest of greatest flow (s, t) between sender, denoted by s and each receiver, called t.

It was revealed that NC can attain multicast capacity. Li et al. displayed that it is adequate for the encoding function to be linear. Along with throughput, NC delivers extra benefits, such as sturdiness (by making the way for nodes to obtain possibly multiple copies of a single packet).

In wireless environments, NC has been determined to deliver various benefits, such as enriched energy efficiency (by lessening the number of distinct transmissions) and greater throughput.

For unicast applications, Katti et al. have currently exhibited that the prudent use of NC can, in general, enhance wireless network throughput.

Furthermore, there was also an understanding of random linear coding for multi-hop wireless multicast applications that showed that how such randomized coding could advance the, in general, download latency for file-sharing applications. There was also use of a linear programming formulation to compute the ideally maximum throughput that may be attained for a wireless multicast flow under the influence of original multicast and NC.

Although, majority of this scrutiny do not take into consideration the impact of transmission rate diversity at the link-layer level. In this, the major emphasis was on a rate-diversity aware broadcast tree construction heuristic, known as WCDS (weighted connected dominating set), that was exposed to lessen the broadcast latency (described as the worst-case distribution delay of a packet to a faction of receivers) by 3 to 5 times, in comparison to the traditional diversity-unaware routing strategies.

It is generally seen that rate control is tackled at the transport layer in order to amend source rates. Khreishah et al. established disseminated rate allocation coding and algorithms schemes, but for broadcasts aimed at multiple unicast flows. In this chapter, an examination on the bound on the benefit of multi-rate diversity and NC is also discussed. Liu et al. established the bounds on the gain of NC at a factor of 2 for the solo multicast case.

It is also determined that pairwise inter-session NC can enhance the throughput of routing-based solutions, irrespective of the fact that whether perfect scheduling is used. Le et al. (1985) originated a stricter upper bound on the throughput gain for a common wireless network dependent on the encoding number, for instance, the total quantity of packets that can be encoded by using a coding node in every transmission.

For a single coding structure with n flows, the maximum throughput gains for both the non-coded and coded flows is upper bounded by $2n / (n + 1)$. Although, most of these analyzes did not take into consideration the impression of transmission rate range at the link-layer level and rather believes that the transmission rate is autonomous of link distance. Currently, Cui et al. deliberate disseminated scheduling of broadcast links. The LP formulation plays an important role in delivering scheduling rate at each link to attain this optimum.

## 6.2. RANDOMIZED NETWORK CODING (NC)

For straightforwardness, it is essential to take into notice a network represented as a graph, where nodes correspond to terminals, and edges correspond to channels. It is presumed that time is planned, and during each slot of time, one can send through each edge 1 symbol over a determinate field $F_q$ (for instance, if one is operating over the binary field $F_2$, he or she can send only one field per time slot). One can label this as having unit capacity edges.

One of the easy and straightforward question that one can examine in information flow through networks is how to disseminate information from a sole source to an only receiver. It is referred as a unicast connection. How much information that one can send was replied in 1956, from Ford-Fulkerson and self-sufficiently from Feinstein, Elias, and Shannon, in the well-known min-cut maxflow theorem. This theorem is of the view that the maximum quantity of information that one can send equivalents to the value of the least cut that splits the source from the receiver. A cut is basically referring to a set of edges whose removal separates the source from the receiver.

The minimum cut (mincut) value h equals the minimum numeral of edges that is required to eliminate to separate the source from the receiver. For instance, in Figure 6.1. the mincut value to receiver R equals h = 2, since eliminating the edges (B2, R) and (B3, R) separates the receiver from the network. How one can proficiently disseminate information at the mincut rate was also replied by Ford Fulkerson by delivering a polynomial-time algorithm.

This algorithm recognizes h edge-disjoint paths that attach the source to the target. In this, the only remaining tasks is to simply route the information along these paths. From every such path, the receiver needed one symbol per time slot. Figure 6.1. shows two edge-disjoint paths, that can be made into use to provide the direction to symbols u1 and u2 to the receiver.

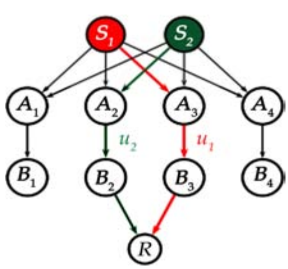



**Figure 6.1:** Two edge-disjoint paths towards receiver R.

*Source: www.researchgate.net.*

Although, this algorithm, as well as routing in overall, presumes that there is a need to recognize in advance the paths that are intended to use, which in turn undertakes that the network is stationary for the concerned objective.

This is not always the case in practical networks, where there would be need of substantial distinctions of the network connectivity. In Figure 6.2, it is assumed that during each time slot, the receiver R links to two unique nodes Bi and B j. For instance, R could be connected to both B1 and B2 at time t, to B2 and B4 at time t + 1, vice versa.

This could work as a replica of a wireless environment, with a mobile receiver. It could also be a representation of a lossy environment, where all nodes B convey information, but only two of the conveyed symbols are acknowledged correctly at each time slot. If the network nodes are trigger to perform routing, then one can route through each edge AB either symbol u1 or symbol u2, as depicted in Figure 6.3. Let us consider these symbols as colors, green, and red, that we require to allocate to edges AB. As, it is well known that there are 4 edges (Ai, Bi) and two colors, it follows that there will be presence of minimal two edges that will acquire the same color. Explicitly, they provide the way for the similar information.

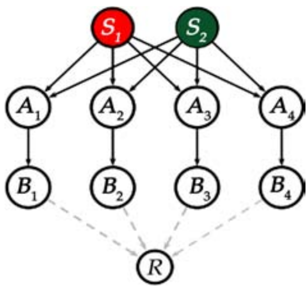



**Figure 6.2:** A dynamic unicast connection: receiver R at each time slot connects to two different nodes Bi and Bj. For example, R could be connected to B1 and B2 at time t, to B2 and B4 at time t + 1, etc.

*Source: www.researchgate.net.*

If the receiver then occurs to make a connection to these two edges, it will obtain the identical information from both of them. If one takes the assumption that each node A chooses consistently at random which of the two signs to forward, then the chance of failure equals 1:2. If one does not take into consideration a randomized scheme, instead use a deterministic scheme, then again get a persistent likelihood of error.

For instance, if we deterministically allot to 50% of the edges (Ai, Bi) the symbol u1 and to the other half the symbol u2, then the error's probability will be equal to k–1 2k, where k is denoted as a number of (Ai, Bi) edges. This chance again goes to 1 2 with the increase in k. Now, let us taken an assumption that nodes A is engaging in linear coding. That is, each node A consistently at random picks and transmits to node B one linear combination of the symbols u1 and u2, say xiu1 + xju2, using the coefficients xi, xj ∈ Fq, the determinate field of operation.

It is then adequate that the receiver R gathers two linearly autonomous equations, as it can compute these to recover u1 and u2. Figure 6.4. demonstrates a probable selection of linear combinations. It is important to note that, no matter to which two nodes B the receiver links, it can recover two linearly autonomous equations to solve for both u1 and u2. In specific, R is required to compute a set of equations of the form:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_1 & x_2 \\ x_3 & x_4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix},$$

where; {y1, y2} are the two symbols receiver R receives; and {xi} are the constants for the linear combinations that is taken into use to generate {y1, y2}.

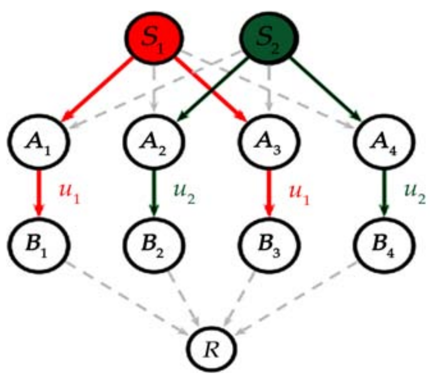



**Figure 6.3:** Routing sends one symbol, either u1 or u2 through each edge (Ai, Bi).

*Source: www.researchgate.net.*

For instance, if R observes edges (A3, B3), and (A1, B1) it requires to compute the equations:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix},$$

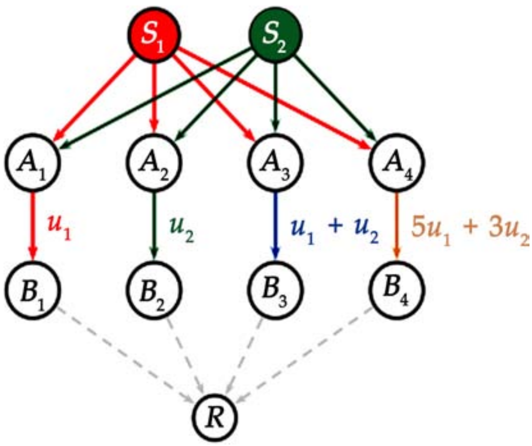



**Figure 6.4:** Coding sends a linear combinations of the symbols u1 and u2 through each edge (Ai, Bi).

*Source: www.researchgate.net.*

If the A-nodes choose the coefficients {xi} evenly at random over a field F q, the likelihood of failure equals to:

$$(1-\frac{1}{q^2})\frac{q-1}{q^2} \approx \frac{1}{q}.$$

That is, we now have attained a regionalized operation, with the likelihood of error fall as low as zero with the increase in the operation of the field sizes. Thus, using coding allows operation in dynamically varying environments with almost near-optimal performance and no centralized knowledge. It is important to note that it is possible to connect an arbitrary number of receivers on the B-nodes of our network, and it is explained in Figure 6.5.

Each respective receiver would still take two linearly autonomous equations and thus rate equal to its mincut. This network demonstrates an exceptional instance of the chief multicasting theorem in NC, which demonstrates that overall networks, if one permits an in-between network node to execute linear coding, we can then transmit rate h concurrently to an arbitrary number of receivers, given that the mincut concerning each receiver is equivalent to at least h. Furthermore, one can also attain this mincut in a regionalized manner by making use of randomized NC.

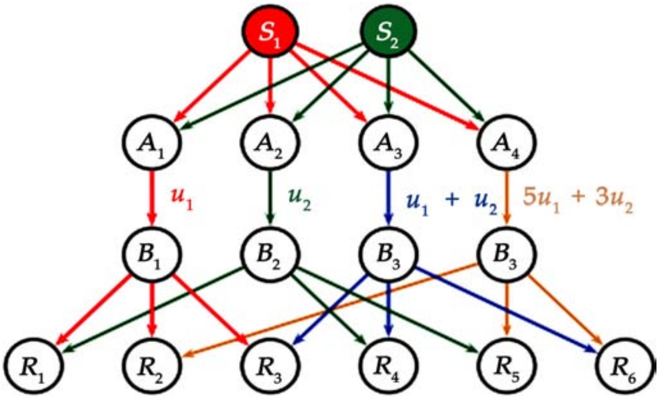



**Figure 6.5:** We can connect an arbitrary number of receivers to the B nodes. Provided that each receiver connects to at least two different such nodes, they can all simultaneously retrieve symbols u1 and u2, with the same code.

*Source: www.researchgate.net.*

As it is well known that one can treat unicasting and multicasting in the similar manner, in the next sections, we will talk about only a unicast connection; although, all the subject matter would also be applied to the case of multicasting the same information to an arbitrary number of receivers.

## 6.3. VECTOR NETWORK CODING (NC) ALGORITHMS

In vector coding, the source concurrently transfers h vectors of length L to the final destination, where L refers to a design parameter. These vectors will be denoted as {u1,…, uh}. These vectors will take the values over a determined field F q.

For instance, in the majority of this chapter, the main area of focus will be on binary vector coding, where Fq = F2. The midway network nodes gather vectors of length L, linearly process them by multiplying them with coding matrices having the values in the field Fq, and then further proliferate them. The L×L coding matrices will be denoted as $\{X_k\}$.

It is important to note that in order to transmit a binary vector of length L from an input x to an output y over the binary deterministic network, there is need to use the input h times, each time carrying a single bit, and consequently, amass h bits from the output y.

In a similar way, just like in the scalar coding, one can subsidiary a state variable with every edge of the network, where now each state variable is a vector of length L, and transcribe the states pace equations for receiver j as:

$$s_{k+1} = As_k + Bu_k$$
$$y_k = C_j s_k + D_j^B u_{k.}$$

In case the network has m = |E| edges, in the equations given above, u k is the L h × 1 input vector that embraces of the h vectors {u1, uh}, s k is the L m × 1 vector that comprises the m state vectors, and yk is a L h × 1 output vector. Matrices A, B, D j, and Cj are basically block matrices of adequate dimension which comprises of blocks with size L × L.

Without loss of generalization, it can be taken into the assumption that D j is the all-zero matrix. It is important to note that matrices B and Cj are fixed block matrices that have as elements either the L × L all zero matrix 0 or the L × L identity matrix I.

Matrix A is shared for all of the receivers and rejects the network topology, that is, the method or criteria with which way the edges (memory elements) are linked. The entries of this matrix are either persistent, or the unidentified coding matrices {Xk}, and it is assumed that we have ν such unknowns.

## 6.4. CODE DESIGN ALGORITHM

In this section, our main area of focus is to develop our own algorithms. Both for scalar NC and vector, the first step is to formulate the algebraic expression. That is, to create the transfer matrices Mj, $1 \leq j \leq N$, and M. As the major area of interest is in polynomial time algorithms, the multivariate polynomials $f(X_1, \ldots, X_v)$ are not calculated explicitly. There are basically two steps that are involved in code design:

- **Step 1:** Each variable is expressed as Xi as a polynomial of a single variable X, and these polynomials are chosen carefully in a way that ensures the polynomial $f(X_1, \ldots, X_v)$ does not become identically 0; and
- **Step 2:** As far as scalar NC is concerned, the first step is to select a scalar value for the variable X from a finite field of size q as

trivial as feasible, and for vector NC, it is important to choose a L × L matrix in ML(F2) for the variable X of size L as trivial as feasible, so that the polynomials assess to a nonzero value for scalar coding, and to an invertible matrix for vector coding.

The second step is the most critical one as it is the one that differentiates this algorithm from the algebraic code designs in the literature: these algorithms are the first, as per the present knowledge, that cooperatively aimed at minimizing the field size while recognizing valid solutions.

## 6.4.1. Code Design for Vector Coding

The first way to start is to describe the given algorithm, and then examine its functioning:

- **Step 1:** Assignment of polynomials to {Xi}:
  - Let us take into assumption that the variables {Xi} take scalar values. By taking into consideration matrix completion methods, one can easily identified an assignment of values to the variables {Xi = αi}, with {αi} in a finite field F q of size q > 2 log N, in order to make sure that all matrices M j become invertible, i.e., det(M)% = 0, and det (j)% = 0, for j = 1…N. That is, f (X1 = α1,…, X ν = αν)% = 0.(5).
  - Let us take into assumption that the field F q, where the values {αi} fit, has size q = 2k with k = &log N' + 1. By taking into consideration a standard representation of extension fields, one can express each value αi ∈ F2k, acknowledged in the preceding step, as a binary polynomial pi(X) of degree at most k–1 in an unspecified X. These polynomials can be substituted in place of the variables {Xi} in the transmission matrices Mj and the transmission matrix M.
  - The determinant of the transfer matrix M is calculated. It is important to note that the entries of M are polynomials in a single variable X, thus the determinant can be efficiently calculated. Then we can get a solo variable polynomial f(X) which is equal to f(X)! f (X1 = p1(X), $X_v = p_v$ (X)). (6) We know from (5) that the polynomial f(X) in (6) is not identically zero. Furthermore, one can easily visualize that it has degree at most N(k–1) hΛ in the variable X, where Λ is considered as lengthiest path length from the source to a receiver.

Now, let us take into consideration the variables {Xi} as L × L matrices, and undertake each such matrix is expressed as the polynomial pi(X) that have been identified previously, of an L × L matrix X. It has been ensured in this assignment that the resulting matrix polynomial f(X) in (6) is not identically 0. The existing code design problem is now abridged to choosing the size parameter L and a single matrix X = A so that the matrix f(A) is invertible.

- Step 2:Assignment of Value to X:
- First and foremost is to find a polynomial g(X) that is co-prime with f(X), of degree m as trivial as possible. It will be proved in the interpretation of our algorithm that one can easily identified such a g(X) of degree $m \leq \log(Nh \Lambda \log(N))$ in polynomial time.
- If g(X) has degree m, create an m × m matrix A so that g(A)=0, using for instance the well-recognized construction in Lemma III.
- Select L = m and X = A. The following Lemma III. demonstrates that for this selection, f(A) is an invertible m × m matrix. Therefore, every coding matrix Xi is assigned the L × L matrix pi(A).
- **Step 3:Algorithm Analysis:** The algorithm that are discussed here are aimed at minimizing the size L of the executed coding matrices that is equivalent to least degree polynomial g(X) co-prime to f(X) that one can identified in polynomial time.

In Theorem III.3 an upper bound is provided on the degree m of g(X) that is required to be execute (hard guarantees). In Lemma III.4, it has been shown that the portion of polynomials that have a co-prime factor of degree at most m, meets doubly exponentially (with m) to one. It intensely suggests that the given algorithm in the majority of instances will result in a size much smaller than the upper bound.

## 6.4.2. Code Design for Scalar Coding

- **Assignment of Polynomials to {Xi}:** Just like in Step 1, the not identically zero polynomial f(X) is created. The code design problem is thus reduced to the problem of finding a value X = α so that f(α) not equal to 0.
- Assignment of Value to X
  - Similar to Step 2, the aim is to identify an irreducible polynomial g(X) that will be co-prime T (X) of degree at most m = log n.
  - The finite field has been taken into consideration of size Fm produced by the polynomial g(X). The assignment Xi = pi(X) mod g(X) is made. Therefore, each Xi is allotted a value in the field $F_2^m$. The polynomial f(X) assesses to the nonzero value f(X) mod g(X).

That is, the value α is assigned to X in the limited field made by g(X), parallel to the indeterminate X.

- **Analysis:** For instance, for 75% of polynomials, engaging a binary alphabet for scalar NC is adequate.
- **Alphabet Size in Network Coding:** It is essential to take into notice that our algorithm lessens the problem of lessening the alphabet size (finite field of operation) in scalar NC, to the problem of discovering a reduction of the polynomial f ($X_1$,…, $X_v$) to a solo variable polynomial f(X) that has a co-prime factor of degree as lesser as possible.

One major challenge in this preparation is the way the decline to a solo variable polynomial is accomplished. This decline can be achieved in numerous ways, and what is the ideal way is not clear. For instance, a polynomial f1(X) can have greater degree than a polynomial f2(X), although, f1(X) may have a lesser degree co-prime factor than f2(X), resulting in a tinier field of operation.

The algorithm that is discussed here does not guarantee to find the optimal alphabet size, but still delivers a method to minimize the employed alphabet in a large fraction of cases.

It is well known that, if our planned algorithm for vector NC identifies a solution of size L = m, then the algorithm for scalar NC will ascertain an explanation over a finite field of size $F_2^m$.

Therefore, these algorithms would result in equivalent solutions. The next two theorems make this correspondence more common, and up to some extent, self-governing of the method executed to recognize the scalar solution or vector.

## 6.4.3. Vector Solutions Using Subspace Codes

These formulations from the preceding sections are based on rank-metric codes, but can be viewed as an exceptional case of a broader construction on the basis of subspace codes. In the follow-on process, the simple preparation of this construction was explained, exhibit how one of such constructions can be amended by making use of subspace codes, and offer a broader question on subspace codes which is resultant from this discussion.

Lastly, multicast network was shown that is accompanied with three messages in which vector NC outclasses scalar NC, where the major area of focus is to make use of special classes of subspace codes.

The construction with subspaces can be attained by observing that the rows of the matrix, where C is a $t \times n$ matrix, is a foundation stone of a subspace of dimension t in $F^{(t+n)}_q$ and the set of all such matrices in the network code forms a code in $G_q(t + n, t)$. For several constructions and networks, it is important to comprehend what sort of code is needed for each network. For instance, Construction 2 in Section V-C can be enhanced by making use of a code in $G_q(4t, 2t)$ with minimal subspace distance 2t.

A foundation for a codeword is a $2t \times 4t$ matrix and the matrices that develops the footing for the codewords can substitute the $2t \times 4t$ matrices of the form [I2t Ci] in Construction 2. Such a code will provide the reason to make use of more nodes in the middle layer of the network. Formations of bigger codes for this objective can be found.

Although, the improvement is not huge since asymptotically the code gotten from an MRD code that was used in Construction 2 is ideal and can be enhanced by at most a factor of four. In addition, for the other constructions, for instance., the simplifications in Section V-E, subspace codes can be taken into consideration.

For these constructions and other variations, the needed large subspace code is labeled as follows. For a given $\rho$, $0 \leq \rho \leq \ell - 2$, there is a need to look for a bigger code in G q ($\ell$t, t) such that the linear span of the rows of any $\ell$ codewords is a subspace whose dimension is at least $(\ell - \rho)$ t.

One can make use of such code when $\rho$ links interlink the source with each receiver. More simplifications will be reviewed in the full version later on. One perfect instance of such construction that needed a novel type of subspace codes is a multicast network having three messages in which vector NC outdoes scalar NC.

It is generally seen that network is a simple modification of the $N_{3,r,3}$-combination network. The new network $N_{e3,r}$ comprises of $N_{3,r,3}$ with an extra link from the source to each receiver. For scalar NC, each edge comprises of three coefficients that can be deemed as a 1-d (one dimensional) subspace of $F^3 q_s$. At most two edges from the three edges, deriving in the middle layer and finishing in the same receiver, can transmit the similar 1-d subspace. Henceforth, $r \leq 2(q^2_s + q_s + 1)$. We validate the benefit of vector NC on scalar NC on a particular instance.

Let us take into assumption q = 4 = 22, i.e., $r \leq 42$, and cogitate now vector NC, where the mails are binary vectors of length t = 2. Henceforth, the edges will carry 2-D subspaces of $F^6_2$. Vector NC will outclass scalar NC if one will able to identify more than 42 2-D subspaces of $F^6_2$ such that any three 2-D subspaces will span at least a 4-D subspace, consequently they will be accomplished by the extra link from the source to the receiver. Such a code with more than 42 subspaces can be originate using some spreads.

## 6.5. AVERAGE THROUGHPUT CODING BENEFITS

Let us take into consideration a communication network exemplified as a directed graph G = (V, E) with unit capacity edges, and h unit rate info sources $S_1$, …, $S_h$ that concurrently convey information and data to N receivers $R_1$,…, $R_N$ situated at distinct nodes. Assume that the min-cut between each receiver node and the sources is h.

**According to the Ford-Fulkerson's min-cut, max-flow theorem, it was stated that if a single receiver could exploit the network reserves by itself, it would be able to attain info at rate h.**

Currently, it has been comprehended that letting network nodes to re-encode the information they obtain (Apart from re-routing) permits each receiver to recover info at rate h, even when N receivers instantaneously assign the network resources.

Such sort of coding is now known by the name as NC. Furthermore, it was shown that by linear network coding (LNC), the min-cut rate can be attained in multicasting to manifold sinks. It is not always the scenario when network nodes are only permissible to accelerative the info they obtain, and in general, NC delivers throughput benefits in comparison to routing.

One common question that is being asked is that how sizeable these throughput advantages are. Let Tc = h represents the rate that the receiver know-how when NC is taken into use.

Generally, two types of routing are taken into consideration: integral routing, that needed that through every unit capacity edge we route at most one-unit rate source, and fractional routing that made the way for numerous fractional rates from distinct sources that add up to at most one.

It was demonstrated that, for undirected graphs, if one permits the fractional routing, the throughput benefit that is being offered by the NC over routing is restricted by a factor of two, i.e., T f/T c $\leq$ 2.

Experimental results in over the network graphs of six Internet service providers also presented slight throughput benefits in this event. This result does not transmission to directed graphs. The authors specify an instance of a directed graph (also known by the name combination network in the NC literature) where the essential throughput benefits scale in proportion to the number of sources, explicitly, Ti/Tc = 1/h.

It has been shown in this chapter that an alike result is true even if one permits fractional routing. In other way, if make a comparison about the common rate ensured to all receivers under routing with the rate that can be offered by NC, the benefits which are offered by the NC are in proportionate to the number of sources h.

In addition to it, it was also demonstrated that for both undirected as well as directed graphs, Tf/Tc equivalents the integrality gap of a standard linear programming construction for the directed Steiner tree problem. Two well recognized lower bounds on the integrality gap for directed graphs are $\Omega$ ((log n/log integrity n)$^2$) and $\Omega$ ($\sqrt{}$ N), where n refers to the total number of nodes in the underlying graph.

For undirected graphs, a known gap is 8/7. In this chapter, the main area of focus is on several benefits of throughput that NC can offers when multicasting to a certain group of receivers that have the identical min-cut. Work in the literature has also initiated to interpret throughput benefits that can be delivered by NC for other types of traffic.

Even for the multicasting, there is still inadequate comprehension about structural properties of multicast configurations that necessitate NC (rather than plain routing) to attain near-optimal or optimal rates.

In a way to enhance our knowledge in this aspect, there is need to give some freedom about the prerequisite that routing has to transfer the identical rate to every receiver of the multicast session, and interpret the maximum average throughput achievable with fractional and integral routing where the average is executed over the rate that every individual receiver experiences.

These quantities can be denoted by $T_{av}$ i = maxA∈$A_i$ 1 N $P_j$=1…N $T_{ji}$(A) and $T_{av}$ f = maxA∈Af 1 N P j=1…N $T_j$ f (A), correspondingly, where the intensification is overall probable routing tactics. By dissociate the challenge of attaining a high average rate from the difficulty of harmonizing the rate towards distinctive receivers, one can expect to enhance its intuition of when NC delivers throughput benefits from a theoretical point of view.

In addition, with respect to a practical approach, for applications that are vulnerable to loss of packets such as real time video and audio, the average throughput is a more adequate measure of performance. It is also valid when (as in the combination network example, there are a huge number of receivers, and the throughput they witness inclines to focus around the average value. In fact, multicast sessions where distinct types of receivers witness distinctive rates is the standard rather than the exclusion in practical scenarios, and erasure coding schemes (e.g., Fountain codes have been advanced to solve this situation. In this, an approach is outlined that combines erasure coding and vector routing to decipher the average to common throughput for an arbitrary multicast configuration.

Now let us take a look at some of the advantages of coding for average throughput over a multicast configuration {G, S, R} and a group of non-negative capacities c on the edges of G. There is need to assume, from a technical perspective, that the min-cut from S to every of the terminals is identical. It can be arranged effortlessly by adding dummy terminals. That is, if the min-cut to a receiver Ri is greater than needed, one can make the connection of the receiver node to a new dummy terminal through an edge of capacity which is equivalent to the min-cut. Then the NC throughput is identical to the communal min-cut value. The greatest attainable average throughput with routing is specified by the greater fractional packing of partial Steiner trees. It is important to note that the partial Steiner tree t initiate from the source S and spans all or only a division of the terminals. With each respective tree t, a variable yt is subordinated signifying a fractional flow through the tree. Let τ be the set of all partial Steiner trees in {G, S, R}, and not the number of terminals in t. Then the greatest fractional packing of partial Steiner trees is offered by the linear program (LP) discussed as follows:

$$\begin{aligned}
&\text{Packing Partial Steiner Trees:}\\
&\text{maxinize} \sum_{t \in \tau} \frac{nt}{N} yt\\
&\text{subject to}\\
&\sum_{t \in \tau : å \in t} yt \le c_{å} \quad \forall\, å \in E\\
&yt \ge 0, \forall\, t \in \tau
\end{aligned}$$

## 6.6. THROUGHPUT BENEFITS OF NETWORK CODING (NC) FOR SW ARQ COMMUNICATION

It is generally seen that flow error control mechanisms allow destination entities to govern the movement of packets directed by source entities.

This process can include buffering control and data rate in addition to error control. Techniques such as automatic repeat request (ARQ) can be used in order to make sure high consistency communication in noisy channels. These methods are generally used to administer the retransmission of the corrupted packets. The retransmission occurs with respect to the three major ARQ schemes: Go-back-N, Stop-and-wait, and Selective Repeat.

It is important to note that the Stop-and-Wait data transfer protocol is the basic category of the ARQ scheme. A receiver initiates the task of error detection on the frames acknowledged from the transmitter using the procedure of error detection. If there is no event of any error occurrence, the receiver answers with a response (ACK) signal through a feedback channel. It is important on the part of transmitter to wait for the response of the transmitted frame.

On the arrival of ACK, the transmitter ensures to send the next frame. Although, in case any error arises in the received frame, the receiver transmits a negative response signal (NAK) requesting for the same frame to be retransmitted. Consequently, the transmitter is required to retain a copy of the conveyed frame until an ACK is received.

The procedure is recurrent in nature as it can be done as many times as needed until a receiver received an error free message. When engaged in communication applications with an extended roundtrip, ARQ is combined with error correction to address the retransmission delay to yield a scheme known by the name hybrid ARQ (HARQ).

Although, in the current time frame, NC has been projected in information theory to propose disputably optimal channel capacity. This is a new model in data transport that has been initiated to enhance network throughput and functioning. With NC, data packets are no longer viewed as a commodity, to be only amassed and then furthered, rather nodes are permitted to amend the packets before advancing them. Ahlswede offered the butterfly model. This model is an engaged graph G= (N, E) where E and N are the sets of edges and nodes in the network. In physical terms, E, and N are the set of routers and links joining pairs of routers, correspondingly.

The source $S \in N$ issues information symbols ($b_1$, $b_2$) at a rate of 2/time unit. The sinks Y, Z are only able to obtain $b_1$ and $b_2$ after two-time units. Although, when NC is espoused, sink Y can obtain symbols and $b_1 \oplus b_2$, and from there sink Y can recuperate the new symbols from a very naive algebraic operation. The same is applicable for sink Z.

Consequently, this will enhance the throughput by 33% at every transmission and attain the min-cut capacity between both the receivers, Z, and Y and the sender S. NC permits us to intensifies the throughput, attaining its optimality when looking for the method or way to transmit the information symbols at rate identical to the source rate asymptotically. It has been revealed that it is adequate for the encoding function at the intermediate nodes to be linear.

The random network coding (RNC) concept was also proposed. In this method, a node outputs the inward packets after employing a linear combination over a determinate field with the help of randomly selected coefficients. Practical NC has also been offered in order to address the requirement for central knowledge of the graph topology and its decoding and encoding functions.

Equivalent to all variants is that the nodes between the destination and the source are permissible to engage in mathematical operations using (probably random) coding coefficients apart from their replication and relaying of data packets. Practical NC can attain throughput intimate to capacity with low delay. In other ways, there are several methods to combine, and later abstract, the liberated data streams formed and expended in the network.

Currently, it has been demonstrated that NC proposes imperative advantages over HARQ in WiMAX communication. A naive MAC layer protocol, also known as, MAC-layer random network coding (MRNC) was offered which substitutes the HARQ protocol. This scheme uses the "generation" or data segment that was projected by Chou et al. where each data segment comprises of n blocks and each block b has a static number of bytes indicated to as the block size k.

While solving problems of such types of linear systems, it is often seen that Gauss-Jordan elimination is a perfect method in this case. This approach is engaged in practical NC which has the benefit of employing progressive decoding by which the deciphered blocks can be created as early as the encoding matrix is abridged to an echelon form. Many other papers have validated the benefits of NC in wireless scenarios.

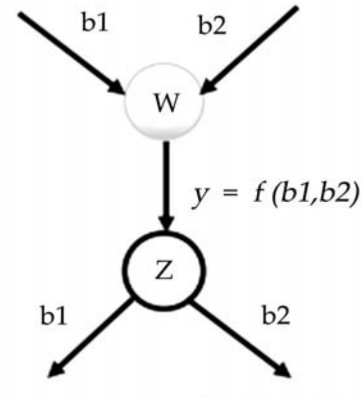
## 6.6.1. MAC-Layer Random Network Coding (MRNC)

In realistic RNC, a file is categorized into generations or segments, every data segment being further categorized into n blocks with a predestined size k, as follows:

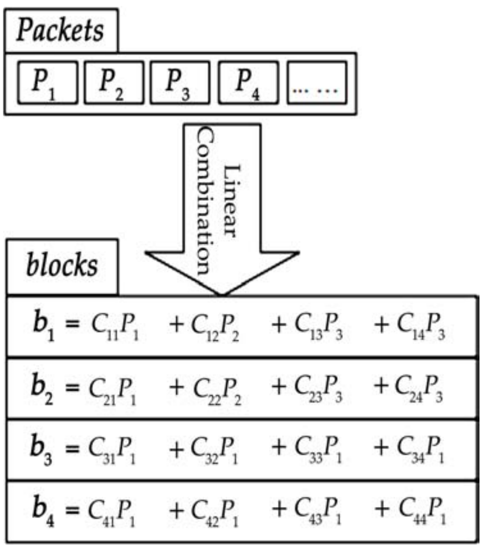
$$\text{data segment} = [b_1, b_2, \cdots, b_n]$$

The sender arbitrarily selects a group of coding coefficients $[c_1, c_2, \ldots,$ cn] to create a linear combination x of the original data block.

$$x = \sum_{t=1}^{n} c_i b_i$$




In MRNC, the sender continues to convey coded blocks from the existing segment, until an ACK is attained from the receiver for the whole segment. As demonstrated in the below figure, packets at the transmitter are merged linearly with arbitrary coefficients $C_{ij}$ then further into the channel.




Based on MRNC, SW-NC ARQ deciphers n packets at an intermediary node using (n, n) encoding matrix. This results in n linearly connected packets which can be sent as one frame y. At the receiver ends, a gradual decoding process with the help of Gauss-Jordan elimination is taken into consideration for each block.

In this process, decoding happens as coded blocks are being attained, that suggests that the decoding time overlays with the time needed to obtain the block. Instantly after linearly liberated coded blocks have been obtained for a segment, the receiver would be able to recuperate the whole original segment and transmits an ACK to the sender. In such a scenario, it is adequate for the receiver to obtain back only one NAK/ACK signal for the whole frame that subsequently lessens the transmitter idle time from 2n to (n–1) time units.

### 6.6.2. Throughput Analysis

It is assumed that the receiver will send back ACKs at a rational power. Consequently, the feedback channel is believed to be noise-free so that the response signals will forever be presented to the transmitter. The time needed to convey one bit from the basis to the final destination is known as propagation time $T_{prop}$.

The transmission delay $T_{prop}$ is outlined as the time expended in releasing all the bits containing the block into the communication channel. Given the bit error rate (BER), the chance that the conveyed block of size k comprises an error FER, is given by:

$$\mathrm{FER} = 1 - (1 - \mathrm{BER})^k$$

## 6.7. CONCLUSION

At the end, it is concluded that the use of internet has been significantly increased in the past few years. With this surge in the usage of the internet, all the elements that has been responsible for making the internet better and advanced has been evolved. With this development, there was a greater increase in the interest about knowing the performance benefits from the use of NC in multi-hop wireless environments.

One common and widely adopted method in this is the use of randomized NC. It is basically known for its easiness. In this, a network is usually outlined in the form of a graph, where nodes resemble to terminals, and edges resemble to channels. It is presumed that time is planned, and during each slot of time, one can send through each edge 1 symbol over a determinate field $F_q$.

Vector NC is another type that has been widely used in the NC. In this, the source synchronously transmits h vectors of length L to the final destination, where L refers to a design parameter. In an akin way, just like in the scalar coding, one can subsidiary a state variable with every edge of the network, where now each state variable is a vector of length L. There are several throughput coding benefits that outweighs it from other comparable methods.

In addition, the flow error control mechanisms permit destination entities to regulate the movement of packets directed by source entities. This approach usually comprises of buffering control and data rate in addition to error control. Several techniques such as ARQ can be used with the objective to make sure high uniformity communication in noisy channels. Such approaches are usually used to administer the retransmission of the corrupted packets.

## REFERENCES

1. Tuvi Etzion, Antonia Wachter-Zeh, (2016). *Vector Network Coding Based on Subspace Codes Outperforms Scalar Linear Network Coding*. [eBook] Tuvi Etzion and Antonia Wachter-Zeh. Available at: https://arxiv.org/pdf/1604.03292.pdf (accessed on 3 May 2021).
2. Alsebae, A., Leeson, M., & Green, R., (2021). *The Throughput Benefits of Network Coding for SW-ARQ Communication*. [Online] Ieeexplore.ieee.org. Available at: https://ieeexplore.ieee.org/document/6550502 (accessed on 3 May 2021).
3. Chekuri, C., Fragouli, C., & Soljanin, E., (2021). On average throughput and alphabet size in network coding. [eBook] *IEEE Transaction on Information Theory*. Available at: https://chekuri.cs.illinois.edu/papers/throughput.pdf (accessed on 3 May 2021).
4. Ebrahimi, J., & Fragouli, C., (2021). *Vector Network Coding Algorithms*. [Online] Ieeexplore.ieee.org. Available at: https://ieeexplore.ieee.org/document/5513771 (accessed on 3 May 2021).
5. Fragouli, C., & Soljanin, E., (2006). Network coding fundamentals. *Foundations and Trends® in Networking, 2*(1), 1–133.
6. Fragouli, C., (2011). Network coding: Beyond throughput benefits. *Proceedings of the IEEE, 99*(3), 461–475.
7. Vieira, L., Gerla, M., & Misra, A., (2013). Fundamental limits on end-to-end throughput of network coding in multi-rate and multicast wireless networks. *Computer Networks,57*(17), 3267–3275.

**CHAPTER 7**

# NETWORK CODING: APPLICATIONS AND CHALLENGES

## CONTENTS

This chapter emphasizes NS applications, including NC in overlay networks and mobile devices, NC for LTE, and CONCERTO systems based on NC.

The chapter also mentions NC applications in wireless networks, including file download, video on demand, live media broadcast, wireless mesh networks (WMNs), and wireless sensor networks. This chapter explains the use of NC in cell phone and in wireless sensors, along with some issues that have been addressed in mobile NC.

The chapter also provides the information about the application of NC in cloud storage and distributed storage and its several architectures, protocol consideration, complexity, and green consideration, and where to apply NC.

This chapter also addresses the challenges that have been faced in wireless NC along with the new challenges that have been addressed in modern time computer systems, that includes the difficulties in broadcast network, along with the several challenges that come up in coding.

This chapter emphasizes the limitation of NC, limitations in network topology, limitations of NC in NC flooding, limitations of NC in peer-to-peer (P2P) networks, as well as the limitations of a random network coding (RNC).

## 7.1. INTRODUCTION

Since, Network Coding (NC) has gained huge amount of attention from the research community, and the reason behind this is the potential to enhance the throughput, energy performance, and delay in a considerable amount, along with the simplify protocol design and providing the security support by its own.

The opportunities in code design have produced a huge amount of stream of new ideas as well as tactics in order to tackle the power of NC.

But, what sort of designs are entirely efficient, when it comes to practice? And what sort of designs will not match up to their claimed theoretical gains because of the real-world limitations?

Without trying all-inclusive view of all practical pitfalls, this chapter tries to find out the meaningful building blocks with respect to an effective or successful design, critical, and common constraints to most intra-session NC systems, along with the effective methods, tactics, and philosophies in order to guide the future models and research problems that are grounded on the practical concerns.

## 7.2. APPLICATIONS OF NETWORK CODING (NC)

Talking about the time interval of the last 10 years, the NC was presented as the phenomenon in the community of research. NC has spreader to several numbers of various discipline of science, that starts from the physical layer to the upper layer. It consists of channel coding, computer science, switching theory, wireless communications, data storage, computer networks, and cryptography.

In addition to this, with respect to mathematics, NC has played a significant role together with game theory, graph theory, matroid theory, and optimization theory. Along with that, with respect to physics, quantum NC was cogitated about, and with respect to the biology, the linear network coding (LNC) has been used for modeling intracellular communication (Figure 7.1).

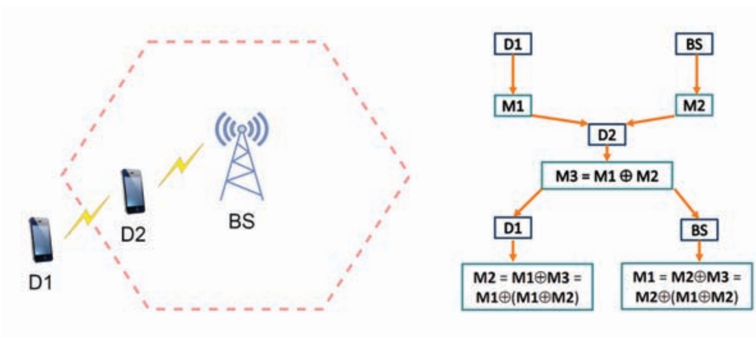



**Figure 7.1:** An illustration of linear network coding.

*Source: Image by Wikipedia.*

### 7.2.1. Overlay Networks

The frequently asked questions from the researchers are, how can NC be applied to peer-to-peer (P2P) distribution of content? Why is NC cooperative in P2P distribution of content? How can NC be applied to P2P multimedia streaming? Why is NC helpful in the context of multimedia streaming?

Talking about the distribution of P2P content, the block scheduling issue finds the keys to utilize the NC in a significantly simple and effective way. The outcomes have been shown that the lower latency of the file downloads and better robustness to peer departures.

On the other hand, talking about the P2P multimedia streaming, the protocols of streaming necessitate to redesign in order to get the occupied advantages of NC. Particularly, a new streaming design which is consist of NC is provided. The application of NC is having the potential, in order to entirely utilize the available bandwidth resources. In this way, it increases or enhances the overall performance of the system.

### 7.2.2. Mobile Devices

This section of the topic, give the idea of the picture viewer application of NC with respect to the mobile phones. The prime thought is that the consumers can share the content over the technologies which are of very short range, for example, Wi-Fi. It has been seen that the consumption of energy elevates with the generation size (Figure 7.2).

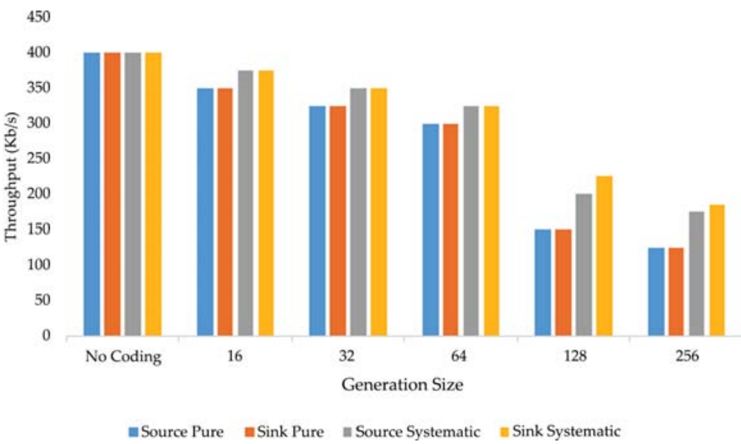



**Figure 7.2:** Throughput vs. generation with or without network coding.

### 7.2.3. Network Coding (NC) for LTE

This bit layout the access of NC in LTE (long-term evolution) networks. NC appears to find the solution for the difficulties, for example, streaming, download services with respect to the future mobile communication systems, the bandwidth and energy.

This can be done with the help of using simulation tool; the NC can save up to 80% of the unwanted or unnecessary information of the cellular link if there are at least two cooperative mobile devices. In addition to this, if the number of devices in the cooperation cluster increases to four, then NC can save up to more than half of the traffic in the short-range.

The linear combinations of the packet that are being transmitted from the sender are known as layer-1. The increment and multiplication are performed over GF, which is having a similar size as the original packets.

In the context of random linear network coding (RLNC), the coefficients of coding are preferred at random. Refer to middle layer and because of the channel situation in wireless network, the packets may tend to lose. The decoder received the encoded message from the sender, which is having the potential to recover or retrieve the original data packets after receiving at least ‘g’ linearly independent packets.

In addition to this, the receiver nodes are having the potential to produce and send new encoded packets, even before decoding the entire generation. They figure the fresh linear set of the packets that they have previously received.

This process is called recoding, and the operation of recoding is an exclusive feature of NC. The traditional coding schemes requires the original message that needs to be entirely decoded before it can be encoded again.

### 7.2.4. CONCERTO System Based on Network Coding (NC)

The CONCERTO system was developed by a team of BAE Systems and the CONCERTO system is a solution of MANET system. It stands for “control over NC for enhanced radio transport optimization.” It was shown to support two to three times more video throughput as compared to a state-of-the-art set of protocols, as well as up to 7 times distance-utility product (Table 7.1).

**Table 7.1:** Emergence Projects Article in Chronological Order

| **2006–2009** | **2008–2010** | **2010–2012** | **2008–2011** | **2011–2015** |
|---|---|---|---|---|
| DARPA CB-MANET/CONCERTO | ONR Adaptive networks/BRAVO | BRAVO NG | DARPA IAMANET/PIANO | DARPA SAFER/SONATA |
| N C for MANETs | N C using local info, high mobility | | N C security | |

The development of projects article in chronological order article that belong to the communications and networking categories is shown in Table 7.1 in chronological order.

In order to supports the entire range of network types, that is, unicast, multicast, and broadcast, CONCERTO system is become visible to joint routing and transport protocol that are having the potential to exchange the packets between the nodes.

Basically, there are two basic models of CONCERTO system:

- First is to replaces the traditional forwarding of data packets with transmission of information with the help of network coded data; and
- Second is, data propagation over sub-graphs.

The CONCERTO systems are made up with the help of several numbers of components that are described below:

1. **Application Interface (App IF):** It receives the packets from the applications running on PCs which are linked to the CONCERTO with the help of an Ethernet cable, which categorizes them into traffic types and passes them to the NC module.
2. **NC Module (Net Coder):** The role of the NC module is to group the application packets into the form of generations, performs random linear coding over packets in a generation, provides coded packets to the forwarding engine upon request, and decodes the packets at destination nodes (DNs).
3. **Subgraph Constructor (SG Constructor):** This uses the information about network topology, multicast groups, as well as source and DNs in order to select the nodes that will forward network coded packets for each application session as well as to find out the forward traffic amount for every single of these nodes.
4. **Forwarding Engine (Master Fwder, Fwdr):** These uses the novel transport protocols in order to decide about which packets are required to be encoded, forwarded, decoded, discarded, retransmitted, and timed-out.
5. **Neighbor Discovery, Topology Discovery (Topology, Topo Cache):** This component of the CONCERTO system finds out the connectivity of node and evaluates the quality of the link. This component uses the effective optimized link state routing (OLSR) protocol in order to spread or distribute the connectivity information across the network in order to support the construction of the sub-graph and to distribute multicast group membership.

6. **Group Manager (Internet Group Management Protocol (IGMP) Module, Group Cache):** Group manager, in CONCERTO system keeps the track of each destinations that are present in each multicast group and helps in providing this information to the sub-graph constructor.

## 7.3. APPLICATIONS OF NETWORK CODING (NC) TO WIRELESS NETWORKS

It is very unexpected that integrating the NC at the physical layer will be practical, in the context of coming interval of time, because of several numbers of various reasons. Nevertheless, it is quite reasonable to build NC into overlay network.

With respect to the overlay networks, nodes are the applications that are running in the computers, and edges are the transport-level linkages between the computers. Overlay networks can be based on the infrastructure, as it has been mentioned by content distribution networks such as Akamai (Figure 7.3).

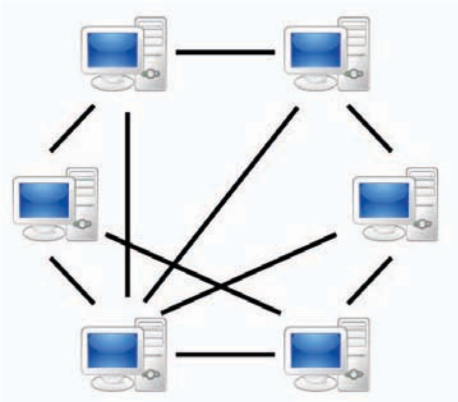


**Figure 7.3:** An illustration or peer-to-peer network.

*Source: Image by Wikipedia.*

They can be used when necessary or P2P networks of end-hosts that are connected to each other in order to execute a specific type of communication task for a short interval of time, for example, live broadcast, instant messaging (IM), file download, gaming, media on demand, or conferencing.

In this topic of the chapter, let us take a brief knowledge of these sort of applications of NC, along with the applications to wireless networks and sensor networks.

### 7.3.1. File Download

Downloading various sorts of file from a server to a client computer is one of the most common job to perform that take place in network communication. On the other hand, the file that has been downloaded is conventionally unicast from the server to the client, if the delay has been neglect or overlook the delay, this can also be viewed in the form of multicast of the file from the server to a large group of clients with the help of a proportionally large amount of buffering.

With the help of simplification, it can be observed that using of NC could elevate the throughput with a very significant amount and, in this way, it decreases the average time of downloading. Let us assume an example, downloading a file over a P2P network can be made of all the nodes that are presently downloading the file.

With respect to this model, newly arriving nodes join the network with the help of linking to a subset of the nodes that are already existing. The difficulty in P2P content distribution is looking for an optimal block scheduling algorithm, which should decrease the time of downloading in a distributed manner.

In addition to this, a real-life example is Bit-Torrent. Bit-Torrent is the original and yet is the most protocol for P2P file downloading. In the context of Bit-Torrent, the file is equally divided into 'h' number of pieces. Each node negotiates in order to obtain pieces of the file from its adjacent nodes, unless, and until the node gets all the 'h' number of pieces and can leave the network.

After a node gets a new piece, it declares this possession to its adjacent nodes, in such a way that node knows which pieces every neighbor node is having. Conventionally, the nodes request a local rarest block, that is, a block that is least common between all the neighboring nodes.

Without NC, each node would have to pick which blocks to download, and the decision is entirely based on the information it could gather from the adjacent nodes. This could take into the direction of an inadequately downloading blocks, as the blocks that are rare in its immediate bordering nodes are not essentially rare blocks when taking into the consideration about the whole network.

On the other hand, with the help of NC, never the less, as the coded blocks are aggregates, every coded block are near about equally significant or useful to any node, and as there is no requirement to seek out and request global rarest blocks in the network. This helps in stop the hampering of an information bottleneck, and this elevates the time of downloading a file.

### 7.3.2. Video on Demand, Live Media Broadcast, and Instant Messaging (IM)

Video on demand can be well thought out in the form of a specialized kind of the file download. In video on demand, the pieces of the downloaded files must arrive in order and must be decoded in near real-time, considering some small delay. In this case, NC can be applied by disintegrating the file in the form of chunks, which can be download following a logical order or sequence.

An identical methodology can be applied with live media broadcast. Earliest decoding can be applied in order to decrease the delay time. IM is also very identical as compared to the live broadcast. But also, in live broadcast has a lower bit rate as compared to the IM, and the reason behind this is, only text is usually being sent.

In addition to this, text messages are also more frequently sent in the form of clusters or bursts, and they have a less strict delay constraint, comparatively. Nevertheless, IM now often is consisting of a larger message, for example, they include images and audio clips as well. Generally, flooding is used for the IM in P2P networks, which comes out as an ineffective when used on larger files.

NC can be applied in all the cases that have been mentioned above in order to elevate the effectiveness in overlay networks.

### 7.3.3. Wireless Mesh Networks (WMNs)

Other than an application-level overlay network, there is another suitable or appropriate place where NC can be used, and is known as a link-layer network. The example of a link-layer network is a wireless mesh network.

Wireless mesh networks (WMNs) are consisting of mesh routers which allows the access to an existing infrastructure, and mesh clients, which both supply multi-hop connectivity to the mesh routers and uses the connectivity which is being supplied by the other mesh clients.

It has been observed that, how the several transmissions are needed for two wireless nodes in order to exchange the packets through an intermediate node can be reduces from four to three with the help of NC. This can be expanded to multi hops, furthermore (Figure 7.4).

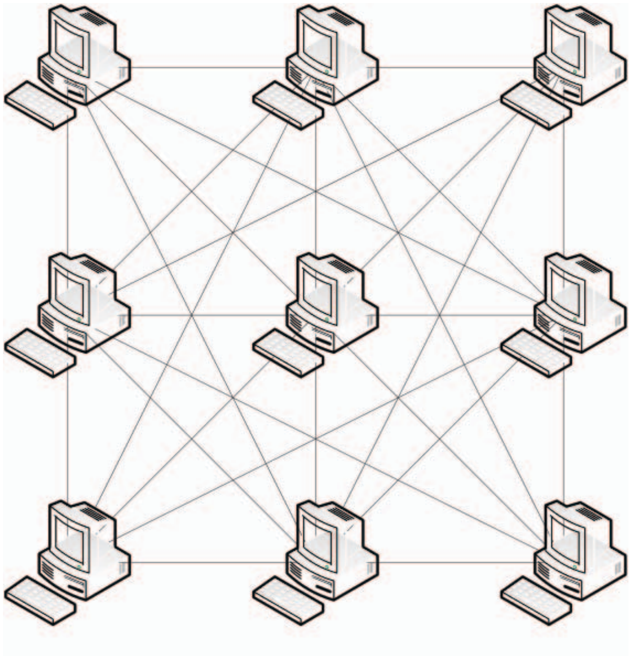


**Figure 7.4:** An illustration of true mesh network.

*Source: Image by Wikimedia commons.*

If a wireless node 'a' sends a packet stream to a wireless network node 'b' over a long series of hops, then node 'b' can send packets in the opposite direction to a for free with an equal rate of stream. Meaning, additional transmission on the intermediate hops would not be essential. This could take to a huge amount of savings with respect to the total amount of data that is required to be transmitted.

## 7.3.4. Wireless Sensor Networks

For the final application of the NC, let us assume a wireless sensor network. In the wireless sensor network, a very small sensors can be dispersed on to the surface, and can produce energy with the help of the environment in order to form a sensing surface over a built-in communication network.

Usually, the sensor nodes are consisting of a radio transceiver, a microcontroller, a memory unit, along with a set of transducers with the help of which they can obtain the data and process it. In order to decrease the size of the node as well as the requirement of energy, the transceiver oscillator is being substituted by an on-chip resonant circuit.

Never the less, the center frequency of the resonant circuit is random. Meaning, each node chooses a random channel on which they transmit, and the other random channel on which they receive. The nodes can self-organize themselves in order to form a multi hop network and send the data to a sink node.

The throughput among any two nodes remains constant if only routing is being used. Also, the throughput grown linearly in the number of channels if the NC is being applied, and the radio ranges are picked optimally. The reason NC aids significantly, in such a way that randomly mixed packets can then find their way to the desired places without the need for explicitly informing nodes where their destinations are.

Especially, this is very useful as the explicitly identifying routes are hard when the graph connectivity is unknown.

In this section of the chapter, it has been notices that some of the areas where NC can be utilized in wireless networks and can extensively categorize its advantages in the form of an improvement throughput of networks, effectiveness of networks and scalability of networks.

In addition to this, there are several benefits that are consisting of improving resilience to attacks and eavesdropping in the form of a side effect of the encoding schemes used.

## 7.4. NETWORK CODING (NC) IN MOBILE AND WIRELESS SENSOR NETWORKS

Mobile network presents an overt but worthwhile hurdle for NC execution in order to overcome. If mobile or cell phones are compared to a laptop or personal computer, then they have considerable smaller life of battery along with the small computational power, which are two resources NC necessitates to improve the performance of an individual system (Figure 7.5).

**Figure 7.5:** A picture depicting network coding in mobile and wireless networks.

*Source: Image by unsplash.com.*

Nevertheless, broadcasting data to various numbers of mobile devices could prove to be a compelling reason in order to pick NC with the propagation of data.

Various cell phones can be placed in the same region at any given point of time. If more than one cell phone streaming the similar data, stated how costly the NC can be, it would be best to stave off the utilization of an overlay network and simply share the data directly.

In addition to this, if the cell phones are kept apart at a significant distance, unicast or broadcast content exchange could be the other option. There are various advantages or benefits for using it, and it relies on the number of receivers. A unicast is appropriate if the numbers of target mobile devices are small-sized, and they can save on the consumption of energy when it is compared with a broadcast implementation.

Never the less, broadcasting is very efficient or optimal solution, but also, it necessitates error-correcting code upon arrival to destination. Forwarding to close neighbors decreases the load on the server in order to broadcast.

### 7.4.1. Mobile Network Coding (NC) in Developing Countries

At the present interval of time, emerging nations may not have the means, in order to install various numbers of wireless access points across the land. In this way, routers, and the data sharing are very crucial with respect to the data in order to spread to all network users.

Generally, emerging nations are using mesh network, and it is very common. Mesh networks are depending on the nearby routers in order to send the data from wireless access points; data can reach various numbers of users at very optimal cost. NC can assist in elevating the effectiveness and throughput in these special networks.

Users, that are present near to the hot-spot, or a wireless access point, can broadcast the data they receive to the nearby users as well. Users, that receives the data can collect, and can forward the extra data they have received. With the help of NC, nearby users can receive more data more often.

Moreover, multipath reception is an addition that increases the frequency of transmission of data. Although, air interfaces are added in order to have more ways a node can retrieve wireless data. Never the less, a Wi-Fi connection could be incorrect several times. But, the application of NC in order to duplicate and propagate packets decreases the number of errors that could take place.

### 7.4.2. Problems of Mobile Network Coding (NC)

Generally, the RLNC is well thought out for mobile NC. This necessitates a random co-efficient be generated at the place of relay nodes when combining packets. Nevertheless, built-in random number generators are not well-grounded.

Apart from this, the other problem comes up from executing the arithmetic operations. These can be resolved with the help of the Euclidean algorithm for division, and Gauss-Jordan algorithm for decoding. Various specifications of various cell phone devices can produce additional difficulty with incorporating the NC. And, these arose struggles can change the effectiveness of the speed of computation along with the speed of data processing.

## 7.5. NETWORK CODING (NC) IN CLOUD AND DISTRIBUTED STORAGE

Storage systems play a very significant role of storage as a service (SaaS), and a multitude of services provided by Clouds. As, one attribute of Cloud service is location transparency, and, Cloud storage can be called clout repositories, or in simple words, a part of the cloud (Figure 7.6).

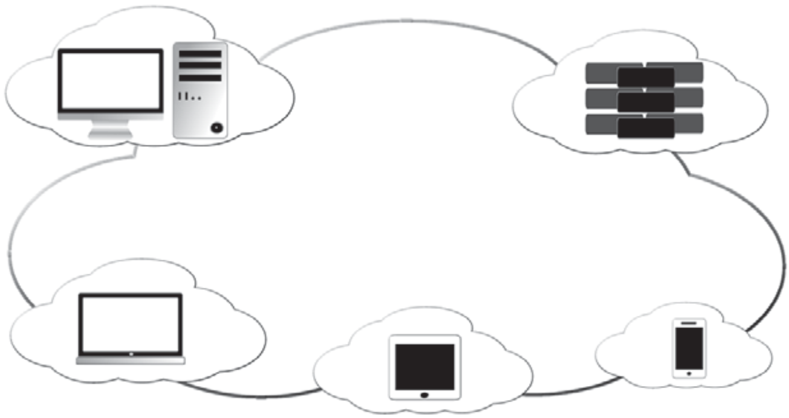


**Figure 7.6:** An illustration of cloud and distributed storage.

*Source: Image by pixabay.com.*

At a present interval of time, there are three kinds of cloud models, which serve to the requirements of the consumers. These three types are known as public clouds, private clouds, and hybrid clouds. Public clouds serve to a general number of people, whereas the private clouds are accessed by a specific group, such as an organization or a company. On the other hand, the third: hybrid clouds are consisting of flexible data capabilities which have the potential to move among the clouds.

In all models, the storage system must correctly store and quickly retrieve data as per the requirement of the consumer.

## 7.5.1. Various Architectures

The architectures of the cloud storage fluctuate; apart from the requirement of the repositories, there does not exist a standard architecture of cloud storage. Nevertheless, a common characteristic between all clouds is the existence of storage management. This can come as a middle-ware or a lead server.

These communicate with the repositories of cloud storage, and would be the location of the NC implementations to the systems. In addition to the, talking about the significance of the primary cloud storage and backup cloud storage, the primary cloud storage can be used for more frequent and more smaller reads and writes.

On the other hand, the backup cloud storage is used to store much larger, less used data as compared to the primary cloud storage. Primary cloud storage can be compared to RAM (read access memory), whereas backup cloud storage can be compared to magnetic disks on a computer (Figure 7.7).

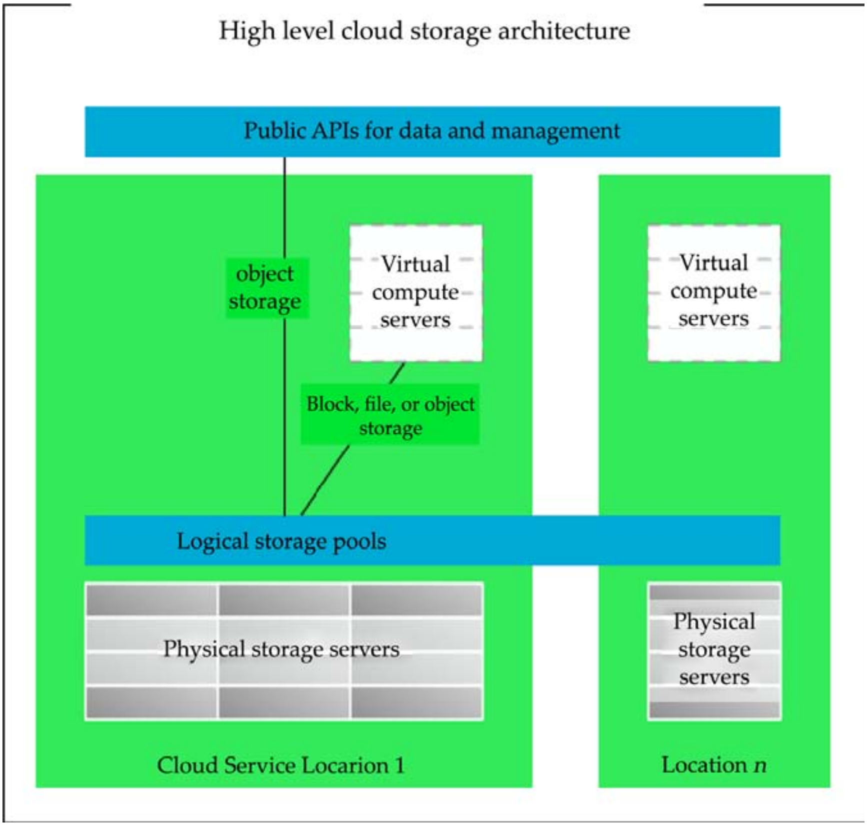



**Figure 7.7:** High level cloud storage architecture.

*Source: Image by Wikimedia commons.*

Understanding these two forms of cloud storage can be the key for optimum performance in a specific cloud system. Moreover, in order to increase the security, some clouds are also consisting of a hypervisor. A hypervisor monitors the access requests along with the state of the data that is present in the cloud.

## 7.6. NETWORK CODING (NC) DESIGNS SUITED FOR THE REAL WORLD

Since its introduction in the year of 2000, the NC has been shown to provide theoretical growth with a considerable amount in several numbers of various fields, from multicasting of data in wireline and wireless networks, to more effective designs for P2P communication and distributed storage systems, to wireless meshed networks even with highly dynamic changes.

> **The major philosophy of network coding is to stimulate the system in order to mix various numbers of data packets at intermediate nodes with the help of the coding, despite a storing and forwarding the copies of packets that are routed across the network.**

In this context, it is no longer needed for the system to keep track of which packets have been received. In fact, the main goal of the receivers is accumulating enough coded packets in order to retrieve the information. In this way, the NC provides resilient and throughput effective ways for mobile, heterogenous, and/or networks that are driven by content to collect, store, and transmit information in various conditions (Figure 7.8).

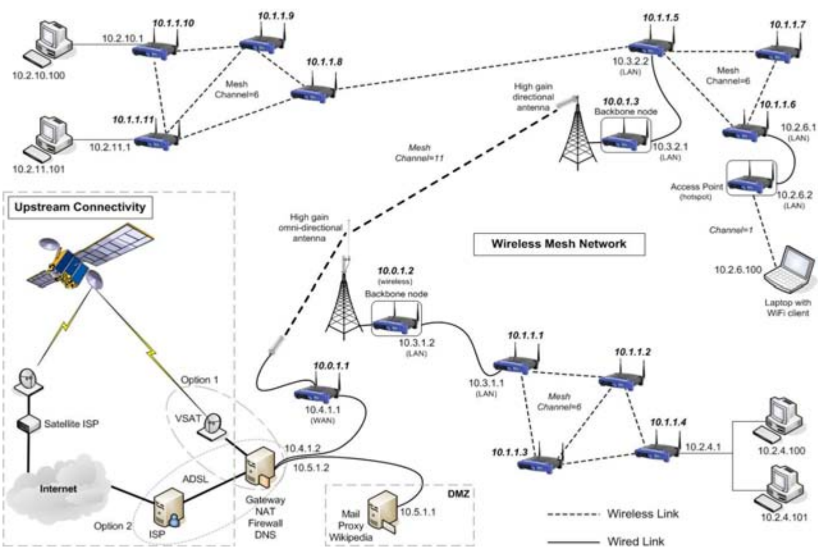



**Figure 7.8:** Wireless mesh network diagram.

*Source: Image by Wikimedia commons.*

A simple, distributed, and asymptotically optimal solution is RLNC. Random liner NC depends on producing packets in the form of linear combinations of original packets with the help of selecting the coefficients of these combinations uniformly at random from the elements of finite field.

Nevertheless, some practical executions have been observed, for example, COPE, CATWOMAN, MORE has recently made its way into the Linux kernel, which is an extensive adoption of NC in the real-world systems and off the shelf devices remain difficult to track down.

One of the reasons why NC is not into the action yet on several numbers of various platforms is the believe that NC will add extra complication to the installed communication systems. The complexity is an outcome of how NC is executed and the choice of coding parameters.

Let us have a look at these two problems one by one:

1. Until now, there was no available or completely tested cross-platform software library that grants system integrators or researchers to utilize or research NC without executing their own solutions. This has changed with the introduction of Kodo. Kodo comes up with a dual license for both researches use as well as commercial use.

In this section, all the outcomes and findings are based upon the utilization of Kodo software library. In this way, it has been claimed that the implementation complexity of basic RLNC algorithms for a huge number of commercially available platforms is mostly solved with the help of the Kodo software library.

Never the less, since the hardware of devices improves the algorithms will also necessitates the adaptation in order to make use of the system in the best possible manner, for example, utilization of single instruction multiple data (SIMD) on the devices of Intel can increase the speed of large finite field computation.

2. Whereas, the selection of the parameters of coding majorly relies on the application and network topology. A prime goal of this topic of the chapter is to introduce to the reader with the most significant aspect to consider under several numbers of various situations. Many times, the selection of the parameters fluctuates significantly between theoretical research and execution of it (Figure 7.9).

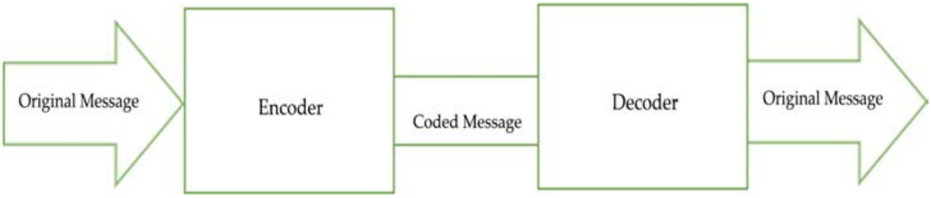



**Figure 7.9:** Random linear network coding diagram.

On the other hand, coping up with the execution and the complexity of the NC, practical adoption of NC also necessitates the adaptation of existing communication protocols and integration with the legacy systems.

Many times, this introduces the requirements to cope up with the time variable delays, heterogeneous receivers, dynamic topologies, and unreliable feedback. These characteristics are discussed in more details in the sections below.

## 7.6.1. Complexity and Green Considerations

As the performance gains of NC over the other tactics that are well understood from a research point of view, the question come to light is how complicated NC really is and how does this affect or influence the platform it is running on.

Understanding the complicatedness is very important in order to derive the relevance to the platforms that are already existing, for which the complexity can be added is limited because of the cost or resources that are available. In addition to this, the additional complexity or computation is directly connected to elevate the consumption of energy in order to carry out those operations.

As the NC has the potential in order to decrease the consumption of energy with the help of decreasing the number of transmissions in some application fields,

there is a clear trade-off between the complicatedness that is required to find out into the NC and the potential energy gain an individual get by decreasing the transmissions.

The complicatedness of the NC is dominated with the help of three major aspects. These three major aspects are the size of generation, the size of field, and the sparsity of the network codes.

### 7.6.2. Processing in the Real World

With respect to the real world, the speed of processing has a large dependence on the specific platform as well as the software implementation. Never the less, this is a connection between theoretical measures of the complicatedness, for example, the number of operations to execute the process of encoding or decoding, and the actual speed of processing, the former is not a direct indication to the latter.

Generally, the former allows to anticipate the behavior of scaling, for example, what happens when the number of packets combined is large, but this is hardly the best indicator in more practical settings.

The underlying hardware structure, which is consisting of cache and memory management is a very censorious problem which is not generally combined in the form of assumptions in the designing the algorithms. This can translate in theoretically effective algorithms performing poorly in practice.

One of the prime characteristics of designing an effective system and algorithm is the basic knowledge or basic understanding of how modern-day computers are designed. The conventional single CPU (central processing unit), storage, and I/O (input and output) (for example, network cards, graphic cards) is no longer an accurate representation of what a computer looks like.

In the present interval of time, a computer emphasizes several numbers of cores (which are the part of the CPU) linked with the help of several memory caches to the main memory and input/output devices.

In this way, effectively using the hardware of a modern computer necessitates algorithms in order to execute multiple cores and optimizing the usage of memory to effectively take benefits of caching.

### 7.6.3. Protocol Considerations

There is a significant difference in concept and feedback requirements between standard routing protocol design and effective NC design. The issue is quite complex, and the reason behind this is the usual assumptions in NC research may not be directly applicable in practice.

For instance, feedback is no longer reliable or immediate. Nevertheless, recent work has contemplated these sorts of effects, and for example, there are additional effects in practice, such as time-varying delays because of the medium access control (MAC) and to buffering in the protocol stack, that can have a significant impact on the overall performance.

In the end, both protocol as well as the design of the code are complex, and the reason behind this is that devices are no longer homogenous in their capabilities, in their channels, or even in the energy available to them, for example, remaining battery of a mobile device or sensor.

According to the most current approaches assume that the system picks a single code configuration, for example, a one-size-fits-all methodology, but this can cause harm to the NC systems for little adaptability.

### 7.6.4. Where to Use Network Coding (NC)

Although some might contemplate the NC in the form of Holy Grail for all kind of setups, there are circumstances where NC will not perform between as compared to the other schemes, but at least perform equally good.

In this section, some examples have been mentioned that show where to use NC and where not to use NC. In case of point-to-point to multi-point communication, the NC will perform as good as any other advanced coding scheme performs, but also, there is no additional enhancement can be expected.

Nonetheless, with respect to some point-to-point scenarios, online NC can offer advantages for applications, for example, coded TCP (transmission control protocol) or video streaming. In the context of multipath communication, generally, the NC provides gains if there are losses, the feedback is delayed, or the capacity of the links are very diverse.

If none of those three took place, then no gain can be expected. In the present time, people can see these applications. For example, channel bundling of LTE (Long Term Evolution) and Wi-Fi for cell phone devices in order to serve small scenarios. As both links will have independent aspects with respect to the probability of packet loss and delay in feedback, gains are anticipated with the help of NC.

Multi-hop networks will gain with the help of NC as far as there are losses. The recoding potential of the NC makes sure that the end-to-end delay and overabundance used on each and every link is minimized-but again there should be no losses as well as there should be no gains as well.

In the context of cooperative settings, for example, wireless meshed networks, recoding is being used with the NC providing gains as far as there are losses or there is no immediate and loss free or delayed feedback.

Scheduling would be equally good if there is perfect understanding and knowledge involved among all the communication nodes (for example, index coding or simple relaying topologies).

In a general manner, this is the very strong use case for NC as it takes away the requirement for overwhelming signaling between communication nodes, and usually, the losses exist in these kinds of scenes.

With respect to the multi-source meshed network and multi-destination meshed network (for example, distributed storage), NC will always offer a gain as far as there are dynamic changes in the network. These dynamic changes can be change of routes, failure or adding of source and/or destination, and change of route characteristics.

## 7.7. NEW CHALLENGES FOR WIRELESS NETWORK CODING (NC)

A new architecture of network that utilizes the NC and make use of the broadcast nature of wireless medium would essentially necessitates the research community to think again about the network stack. The MAC, routing, and the transport protocols are all imported with the help of the wired domain, with minor modifications.

They are optimized to work over point-to-point links, let us consider a single pre-arranged path and a layered architecture. The cost of redesigning the network stack is non-negligible. But, on the other hand, the wireless throughput is very limited, which mandates efforts in order to find out the potential of new high-throughput architectures.

In addition to this, the wireless environment is more controllable with respect to the deployments as compared to the wired environment. Generally, these sorts of deployment can entirely depend upon the software updates.

### 7.7.1. Challenges of a Broadcast Network

The advantages of the NC are, NC utilizes the broadcast nature of wireless medium in order to transmit or propagate a single packet to several receivers at the same interval of time. Most of the new challenges are not produces through NC, but in the matter of fact, they are rather a by-product of depending on the broadcast channel, which has consequences on MAC, routing, along with the transport protocols.

1. **MAC (Medium Access Control):** The standard access mode of 802.11 and similar MACs is a distributed coordination function

(DCF) combining carrier sense multiple-access (CSMA) with collision avoidance (CA). A node that desires to propagate senses the channel and starts sending only if it notices that the channel is clear for a pre-addressed interval.

It then waits for a grant from the receiver. If the sender does not receive any sort of permission or acknowledgment within a specific interval of time, it assumes that there was a collision and selects a random back-off timer which is distributed uniformly within a variance window or contention window.

The contention window increases and get two times for every failed transmission to decrease the chances of collisions. On the other hand, there is no acknowledgment for the broadcast packets. In addition to this, it is uncertain how to add this functionality without making a considerably complexity and the potential for ack implosion.

The deficiency of acknowledgment for broadcast packets shows that the MAC does not have any congestion avoidance function in these kinds of mode. In this way, the application of unmodified 802.11 style MAC leads to an elevated number of collisions, and a decrement in the throughput of the system.

This compound to the problem, and the reason being is that several numbers of various 802.11 cards seem to have a faulty implementation of CSMA. In addition to this, the transmitter is not having the potential to find out the successful reception in the absence of broadcast.

In this way, the MAC does not retransmit the broadcast packets and does not provide any sorts of advantages that are reliable. The absence of MAC reliability is not the problem for intra-flow NC. But inter-flow NC requires to remunerate for it with the help of other mechanisms.

2. **Routing:** The conventional protocols of routing carry out a point-to-point abstraction of wireless network. And reduce routing in a wireless network to a shortest path computation on these directed links, in the manner like wired network. Nevertheless, in the context of broadcast, multiple nodes could receive a packet at one and the same time, and one or more of them might pick to transmit in the form of an outcome.

This changes the notion of routing from a single shortest path to a multipath problem, where the decisions are made after the receiving of packets, and not at the time when they are being transmitted. Multipath routing can be formulated in the form of a linear program (LP).

Nevertheless, the struggle arises from the broadcast property of the wireless links, which gives the LP formation a huge number of constraints. The MORE protocol presents a practical low-complexity heuristic that cope up with this problem.

It has been believed that the MORE shows an important first step in the direction of a general routing protocol which is based on broadcast. In addition to this, more improvements would combine COPE with MORE, improves round-robin service of multi-flow of MORE, and rethink the auto-rate.

3. **Transport:** The 802.11 MAC, which is used in broadcast mode does not accomplish the usual link layer functionality of contention resolution and retransmission of lost packets.

With respect to inter-flow coding, the subsequent high loss rate is required to be label. Othherwise, it would be mistaken as a sign of congestion by transport protocols such as TCP, producing them to decrement in their rate of sending unnecessarily. In the context of intra-flow coding, reliability comes for free, but a different problem surfaced. Coding and decoding involve linear operations over batches of packets. Batching does not work well with window-based transport protocols such as TCP, which are based on window.

In this way, the transport protocols would require to be designed again in order to be efficient in the presence of these changed forwarding semantics.

## 7.7.2. Coding Challenges

The difficulties in coding arises from the desire to integrate various attractive aspects, for example, low complexity, high achievable rates, delay, and memory requirements, and adaptability to unknown channel situations. Generally speaking, there is a trade-off between these properties.

1. **Fast Coding:** NC necessitates intermediate nodes in the network in order to execute the operations over finite fields in real-time. On the other hand, the cost of inter-flow coding is typically quite less, the general linear codes used in intra-flow coding could be expensive in terms of computationally.

In this case, the encoding algorithms necessitate polynomial time complexity that is bounded as O ($n^2$), in which 'n' is the number of the linearly combined packets. Nodes may require to inspect whether received packets are innovative (bring new information), in which case an additional complexity of O ($n^2$) operations is required.

Decoding amounts in order to solve a system of linear equations, that generally necessitates O ($n^3$) operations. There is an increment in the effort in order to design lower complexity encoding and decoding algorithms that are inspired by low density codes, but this try is still at its first steps.

Never the less, the coding is not a deployment hurdle in current wireless networks. According to the reports of Chachulski et al. in which he said about MORE, a NC protocol, can sustain a throughput of 44 Mega-bites per second on low-end machines with Celeron 800 Mega-Hertz CPU and 128 Kilo Bites of cache.

2. **Forced Reliability:** In the context of Intra-flow NC, intra-flow NC does not naturally offer itself to graceful performance degradation: receiving only 'n–1' linear combinations of n linearly combined packets are practically useless. This carry out the stringent necessities for reliable delivery, and is in stark contrast to uncoded transmission.
3. **Realtime Traffic:** With respect to the intra-flow coding, Intra-flow coding across n packets implies that a receiver desires in general in order to collect all 'n' of them before extracting the data. In real-time applications, for example, audio, and video, the related delay might be prohibitive for large values of n, which seems to depict that small values of n would be desirable.

With the help of very small values of 'n,' on the other hand, may not give permission to optimally mix the information and realize the theoretically promised benefits. In this way, there is a tension in balancing these two opposing requirements.

## 7.8. LIMITATIONS OF NETWORK CODING (NC)

On the other hand, the NC has been shown to offer several numbers of advantages, for example, greater energy efficiency, faster download times, there are specific faults which, at the present interval of times, excluding the NC for competing with the conventional store-and-forward schemes.

The network topology itself could be an obstacle with respect to the performance of the NC. The nodes in these sorts of network that do not have the chance to encode the data from several various incoming neighbors are possibly better using a conventional routing protocol.

In the context of NC flooding, the network-coding-enhanced protocol cannot do well in the networks where nodes have more than two neighbors.

With respect to NC in P2P networks, the main reason behind the lack of popularity is the expensiveness or high computational cost for encoding the data and decoding the data. As the potential of computation of peers is not known yet, but a peer may perform worse in a NCP2P network as opposed to a conventional P2P network.

## 7.8.1. Network Topology

Majorly, NC outshines in network topologies which have little bottleneck. According to the theorem of Menger, which states that, a topology which is having minimal links or edges from source to sink nodes may not have an adequate flow to see a considerable increment in the performance.

For example, a chain topology only has ‘n–1’ edges, as mentioned in Figure 7.10. If ‘node 1’ must send a message to ‘node 4,’ no NC would elevate the throughput. If the message has to reach its destination, which is ‘node 4,’ then it must pass through ‘node 2’ and then ‘node 3,’ and then its desired node 4.

Only erasure coding, which is encoding and decoding that takes place only at the source and the destination, respectively, can elevate the value of throughput of the network topology when it has been compared to a conventional store and forwarding routing method. In this way, NC is of little use in topologies with minimal edges between any pair of nodes.


**Figure 7.10:** Chain topology.

## 7.8.2. Limitations of Network Coding (NC) in NC-Flooding

NC flooding does not perform that good with network topologies in which more than two links per node are common. And the reason behind this is that, a rule for simple NC which must be stick to in order to make sure the successful decoding of packets.

According to Figure 7.11, ‘node 1’ has just received m 2, m 3, m 4, and m 5 (m 1, m 2, m 3, m 4, and m 5 are messages from the corresponding nodes in the network). The next outgoing packet of ‘node 1’ must be decodable by all nodes that are present in the network.

If any of the two incoming messages were integrated and broadcasted, the two nodes would only be decoded, on the other hand, the rest of the nodes will drop the packets, and the reason being is that it is not decodable.

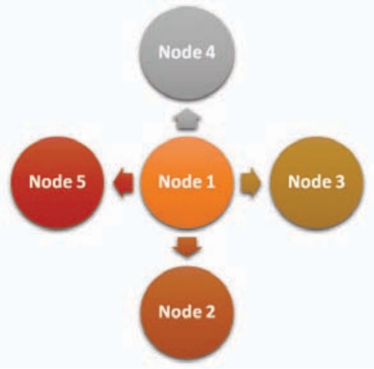



**Figure 7.11:** Four-way bottleneck.

In this way, only one message per time unit can be broadcasted to all the other nodes that are present in the network. An improvement in order to decrease the time is to send varying encoded or native packets on each link.

### 7.8.3. Limitations of Network Coding (NC) in Peer-To-Peer (P2P) Networks

Bit Torrent remains more noteworthy and more well-kwon P2P network as compared to Avalanche. Perhaps the computational overhead is the prime reason that NCP2P networks are not that common.

The specifications of the target computer can fluctuate from peer to peer. In this way, slower computers can impact the performance of said peer in the network, as it would take longer in order to decode and encode the data blocks.

## 7.9. CONCLUSION

In the conclusion of this chapter, the future of NC systems is consisting of a combination of existing and new technologies. This chapter have quantified the advantages of the NC concept as utilized in the wireless network. This chapter illustrates the application of NC in the overlay network, cell phones, LTE, and CONCERTO systems. Moreover, in the coming interval of time, the applications of NC in multicast communication can be seen.

From this chapter, a deep understanding of NC and what it implies have been gained, along with the origin of this field through its different applications in wireless networks. In this chapter, both theoretical aspects as well as practical aspects of NC have been observed.

This chapter presented some of the designing rules that are used for NC in the real world. Particularly, this chapter focused on what sorts of designs, network topologies, and protocols are best to befit for the real world, that is, designs that are simple enough yet effective and having the potential to obtain the advantages of NC.

The practical applications of the NC have been shown to have the capability to reduce the time of download, increment in the data throughput in networks, and enhance the overall theoretical performance of a distributed algorithm. The major problem with the NC is the amount of computational power which is required to evaluate these condensed packet payloads from the senders in a network.

NC allows more effective, scalable, and well-grounded wireless network. These occasions come with a requirement to think again about the MAC, routing, and transport protocols. It has been believed that, never the less, the future research will overcome these all difficulties and combine the NC into the design of wireless network.

## REFERENCES


1. Ahmed, H. M. H., (2015). Introduction and applications of network coding (8th edn.). [eBook] *International Research Journal of Engineering and Technology (IRJET*), 471–480. Available at: https://www.irjet.net/archives/V2/i8/IRJET-V2I878.pdf (accessed on 3 May 2021).
2. Bilal, M., & Kang, S., (2019). Network-coding approach for information-centric networking. *IEEE Systems Journal,* [Online] *13*(2), 1376–1385. Available at: https://ieeexplore.ieee.org/document/8451876 (accessed on 3 May 2021).
3. Christina, F., Dina, K., Athina, M., Muriel, M., & Hariharan, R., (n.d.). *Wireless Network Coding: Opportunities and Challenges*. [eBook] pp. 1–8. Available at: http://www.nms.lcs.mit.edu/6829-papers/coding-milcom.pdf (accessed on 3 May 2021).
4. Elom, W., (2016). *Network Coding for Wireless Applications: A Review*. [Online] Cse.wustl.edu. Available at: https://www.cse.wustl.edu/~jain/cse574-16/ftp/netcode/index.html#magli13 (accessed on 3 May 2021).
5. Ephremides, A., Jaggi, S., Ho, T., Shrader, B., & Chung, S., (2008). Network coding. *Journal of Communications and Networks,* [Online] *10*(4), 367–370. Available at: https://ieeexplore.ieee.org/document/6389852 (accessed on 3 May 2021).
6. Fragouli, C., & Soljanin, E., (2007). Network coding applications. *Foundations and Trends® in Networking,* [Online] *2*(2), 135–269. Available at: https://www.cse.iitd.ac.in/~vinay/courses/CSV887/NCApp.pdf (accessed on 3 May 2021).
7. Jonny, L. W., (2015). *A Survey of Network Coding and Applications*. UNLV Theses, Dissertations, Professional Papers, [Online] Available at: https://digitalscholarship.unlv.edu/cgi/viewcontent.cgi?article=3506&context=thesesdissertations#:~:text=Network%20coding%20aims%20to%20improve,received%20data%20packets%20before%20forwarding.&text=NC%2DFlooding%20is%20introduced%2C%20which,or%20time%20complexity%20of%20flooding (accessed on 3 May 2021).
8. Pedersen, M., Lucani, D., Fitzek, F., Sorensen, C., & Badr, A., (2013). Network coding designs suited for the real world: What works, what does not, what's promising. *2013 IEEE Information Theory Workshop (ITW),* [Online] Available at: https://www.researchgate.net/

publication/261056038_Network_coding_designs_suited_for_the_real_world_What_works_what_doesn't_what's_promising (accessed on 3 May 2021).

9. Yuan, L., (2018). *The Limitation of Random Network Coding* (pp. 1–6). [eBook]. Available at: https://arxiv.org/pdf/1210.1915.pdf (accessed on 3 May 2021).

**CHAPTER 8**

# SECURITY AGAINST ADVERSARIAL ERRORS

## CONTENTS

The chapter explains the basic significance of security against adversarial errors. This chapter also provides highlights on the error correction bounds for centralized network coding (NC), such as upper bounds, generic linear network codes, and lower bounds.

This chapter also explains the basic ideology behind finding out the adversarial errors. This chapter gives a brief introduction to the random linear network coding (RLNC) for arbitrarily correlated sources. This chapter includes several various relaying strategies of modeling. This chapter addresses the rare-less codes that have been used in NC, such as streaming protocols. This chapter also emphasizes the various advantages of randomized coding over routing, such as dynamically varying connections and distributed settings.

## 8.1. INTRODUCTION

Distributed network coding (NC) along with its robustness can benefit among other application areas multicast in decentralized settings like peer-to-peer (P2P) networks and wireless ad hoc to deal with packet losses and failures.

The end hosts code and forward the packets to other end hosts in these settings. Compromised nodes then need to be considered diligently with respect to the security aspect. Even as its capabilities get enhanced, network security faces new challenges due to NC.

The usage of a subgraph that contains multiple paths leading to the sink node is facilitated through multicast NC which is one of its main advantages. Information-theoretic security is provided with many beneficial possibilities through coding across multiple paths against competitors that control or observe transmissions/arcs in the network that are a limited subset.

Adistributed multi cast scheme that is based on randoml in ear network codingcan be added with capabilities for error correction or error detection by the addition of designe dre dundancy that is appropriate.This has been explained here in after.

Traditional security techniques face additional problems posed by coding at intermediate nodes. For example, more erroneous packets result from coded combinations that involve packets that are erroneous as a result of which there is a lesser efficacy of the traditional codes for error correction dealing with erroneous packets in limited proportion. Additionally, coding is not allowed at intermediate nodes that are not trusted in the traditional signature schemes. Nodes can sign packets with linear combinations through an elliptic curve-based homomorphic scheme.

Corruption of the packets is detected and forging of signatures is prevented where the assumption is about the hardness of the computational co-Diffie-Hellman problem on elliptic curves. Following an information-theoretic approach, the focus in this chapter is on multicast network's detection problem and correction of adversarial errors.

## 8.2. ERROR CORRECTION BOUNDS FOR CENTRALIZED NETWORK CODING (NC)

The most direct generalization to NC from the traditional algebraic coding theory is where central designing of the network code is done and all parties are aware of it (source, adversaries, and sinks).

The problem formulation here is quite similar as the same capacity is assumed to exist in all arcs (a pair of nodes can be connected through multiple arcs) and the interest is in knowing as to relative to the arc capacity the rates is how large.

To refer to equivalently (N, A, s, T) or to a tuple (G, s. T), the term network is used. Without generality loss, the assumption is that in-degree 0. ‡ is there in the source node (SN). A NC model that is delay-free can be adopted as the network is acrylic which implies that only after the nth symbol of the input process is received by the start node o (1) is the arc 1 transmitted.

Consideration is restricted to scalar NC, i.e., for all n, the function is the same where the nth symbol transmitted on an arc l is a function of the nth symbol of each input process of node o(l). For some prime power q, the symbols transmitted are from Yan arc alphabet which is assumed to be equal to Fq.

There is a resemblance of the coding model with the delay-free scalar linear coding model in these aspects. All the same arbitrary coding operations that are likely to be nonlinear have been allowed here, and the arc alphabet can be different from the source alphabet rather than having a number that has source processes of a fixed rate. Relative to arc capacity log |Y| a single-source process with a rate log |X | is sought to be bound.

For each arc and the source, a single symbol can be focused. The coding operation for an arc l is a function $\varphi_l$: X → Y if o(l) = s, or φl: k: d(k)=o(l) Y → Y if o(l) = s.

A network code φ = {$\varphi_l$: l ∈ A} is defined for all network arcs by the set of coding operations. Where Y1 denotes the random symbol which the end node d (1) of arc 1. § receives and the random source symbol is denoted by X, let the set of input symbols of an arc 1 be denoted by:

$$Y_{I(l)} := \begin{cases} X & \text{if } o(l) = s \\ \{Y_k : d(k) = o(l)\} & \text{if } o(l) = s \end{cases}$$

The arcs l ∈ A are indexed and considered in topological order wherein compared to the higher-indexed arcs, the lower indexed arcs are upstream. The arc and index are referred to interchangeably for the sake of brevity.

$Y_l = \varphi_l\ (Y_{I(l)})$ where 1 has no arc error occurring on it. In case $Y_l = \varphi_l(Y_{I(l)})$ an arc error is considered to have occurred on 1. If on the z of the arc's errors occur, it is said that a z-error has occurred in the network. If, despite the error occurrence, the source symbol can still be reproduced by each sink in T, it can be said that the network code can correct the z-error.

If all the z-errors can be corrected by the network code it is said to be z-error-correcting for all z ≤ z. a Z error is considered to have occurred where in each arc l ∈ Z an error occurs for a set Z of arcs and on the other arcs, no error occurs and the error pattern of the error is said to be set Z. all Z-errors can be corrected by a Z-error-correcting code.

A series of additional definitions are used in the network error correction bounds' proofs below. The (error-free) global coding function φ˜ l: X → Y is defined for each arc l ∈ A where when all arcs are error-free φ˜ l(X) = $Y_l$; |+(Q)| is called the size of the cut and the set of forward arcs of a cut Q are denoted by +(Q): = {(i, j): i ∈ Q, j ∈/Q} (Figure 8.1).

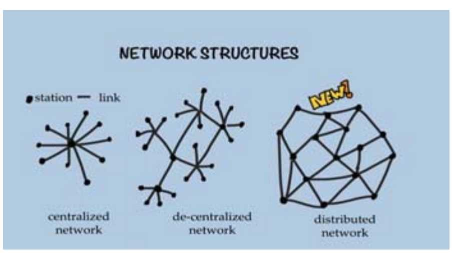



**Figure 8.1:** Network structures.

*Source: Image obtained from Flickr.*

## 8.2.1. Upper Bounds

The size of the alphabet's upper bounds is presented in this section which for the error-correcting codes that are point-to-point are analog of the classical Hamming and Singleton bounds.

Arbitrary coding functions, $\varphi_1$ which are likely to be non-linear have been considered here and the error value el associated with arc1 has been defined as:

$$\mathring{a}_1 := (Y_1 - \phi_1(Y_{I(1)})) \bmod q.$$

It may be noted that only relative to the values of the arc inputs $Y_{I(1)}$ $e_1$ has been defined. For certain given arc error values and a given code el can thus be simply considered as an input of node d (1) and the arc values $Y_1$ can be inductively determined in topological order by using:

$$Y_1 = (\phi_1(Y_{I(1)} + \mathring{a}_1)) \bmod q.$$

For an error-free network if the arc values are first found, the same results are obtained thereafter the arc errors are applied in topological order, i.e., $e_1$ is added to $Y_1$ mod q for each arc 1 where el=0 and the values of the higher-indexed arcs are changed accordingly.

It may be noted that the lower-indexed arc is not affected by an error on arc 1. Vector e: = ($e_1$: 1 ∈ A) ∈ Y|A| defines a (network) error where the corresponding vector of an error and the error are referred to interchangeably.

## 8.2.2. Generic Linear Network Codes

The notion of a generic linear network code is introduced before the lower bounds are developed on the source alphabet size in the ensuing section, and the same shall be utilized for construction of the codes for network error-correcting by which the bounds are proved.

The following maximal independence property is satisfied by a generic linear code: the global coding vectors are linearly independent if they can be linearly independent in some network code for every subset of arcs.

In comparison with multicast linear codes, a stronger linear independence requirement is satisfied by the generic linear codes where only a full rank set of inputs is required by each sink node. Hence even though a multicast linear code is not a generic code, a generic code is essentially a multicast linear code.

Similar to the techniques seen for the construction of multicast linear codes, techniques analogous to these can be used for the construction of generic linear codes.

For the construction of multicast codes, there was the introduction of the technique for random linear coding, and this can also be used for the construction of generic linear network codes and even though field sizes that are much larger may be required, the probability is of asymptomatically approaching 1 in the field size so that in a given network in comparison to multicast codes, for the construction of generic codes similar success probabilities can be achieved (Figure 8.2).

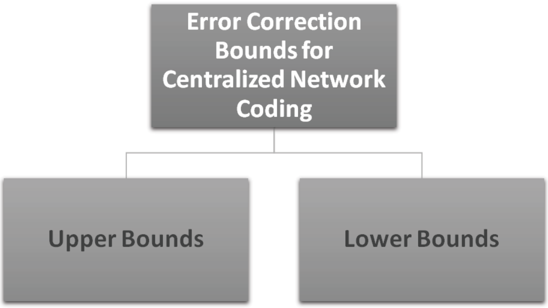



**Figure 8.2:** Hierarchy of error connection bounds.

A deterministic approach that is centralized and similar to it can also be taken. Over a finite field F, a generic linear network code is constructed by the following algorithm for an acrylic network (N, A, s, T) and any positive integer n.

For the construction of a multicast linear code, the network arcs' coding vectors are set in a topological order ∗ by the algorithm, making it similar to Algorithm 1 where the virtual SNs is connected to the actual SNs starting with n virtual arcs.

### 8.2.3. Lower Bounds

For point-to-point error-correcting codes, the classical Gilbert and Varshamov bounds are generalized next by the lower bounds derived on the size of the source alphabet. With parameters that satisfy the bounds, sufficient conditions are given by these bounds for the existence of codes that are for network error-correcting.

Construction gives the proofs. In this section generalization of scalar linear network coding (LNC) is used for the construction of network error-correcting codes where X the source alphabet is a subset of $F_n$ q an n-dimensional linear space and Y the arc alphabet is Fq a finite field where n equal to the minimum source-sink cut size m = min t∈T R (s, t) is set.

Where the number of source processes is equal to n, the entire n-dimensional linear space $F_n$ q is the source alphabet for a basic scalar NC. For some k ≤ m, X is a k-dimensional subspace of Fm q for a linear network error-correcting code.

The source values x ∈ X as length-m row vectors with entries in Fq will be considered here. The set of arc coding operations $\varphi = \{\varphi_l : l \in A\}$ have been referred to as the underlying (scalar linear network) code to distinguish it from the complete network code wherein the choice of X ⊂ Fm q is also included.

A generic code that can be constructed for a specific network is used as the underlying code φ and the same has been described in the preceding section. Choosing X in a manner that under a specific set of error events its values can be distinguished is the remaining task.

## 8.3. DETECTION OF ADVERSARIAL ERRORS

In this section, a scenario has been discussed where an adversary introduces information-theoretic detection errors when except for some of the coding coefficients that are random, the entire coding strategy and message are known to that adversary.

An adversary may be able to create a situation like this when a sink node originally intended as the source message's recipient is compromised by the adversary. Hypothetically speaking, we can assume that z erroneous adversarial packets are sent by the adversary.

Random linear combinations of these r+z packets are received by each sink. Where there is an error connection, the number of transmissions/arcs controlled by the adversary guides the minimum overhead or the proportion of the information that is redundant, which is in proportion to the source-sink minimum cut.

There is no minimum overhead in the error detection case, and this can be flexibly traded off against the coding field size and detection probability.

During normal conditions, for low overhead monitoring when no adversary is known to be present, an error detection scheme can be used along with a high overhead error correction scheme which gets activated when an adversarial error is detected.

In each packet a flexible number of hash symbols is included to add error detection capacity to random linear coding. With high probability, a sink node can thus detect adversarial modifications with this approach.

Incomplete knowledge of the adversary while designing its packets with respect to the random network code is the only condition required to make it precise. The sink can even receive each packet that is corrupted with a packet from the adversary where in comparison to the source, the adversary may have the same or greater transmission capacity.

## 8.4. RANDOM LINEAR NETWORK CODING (RLNC) FOR ARBITRARILY CORRELATED SOURCES

Linearly or independent correlated sources have been considered so far. Using random linear network coding (RLNC), transmission of arbitrarily correlated sources has been considered next where compressions may be required over networks.

Joint decoding of two sources with identical distribution of output values that have been drawn independently in each unit time period from the same joint distribution Q as well as the problem of distributed encoding have been considered analogously to Wolf and Slepian.

Transmission occurring across intermediate nodes in an arbitrary network whereby NC can be performed is the main difference in the problem here. This gets reduced to the original problem of Slepian-Wolf in a special case where from each source, there is one direct link to a common receiver in the network.

A vector linear network code operating on clocks of bits has been considered here. Where the source $X_i$ has the bit rate $r_i$ linear coding is done in F2 with blocks that from each source $X_i$ contain $m_i$ bits.

For each block, $nc_i$ bits are transmitted by each node on each of the incident outgoing links 1, where the corresponding source bits form these by random linear combinations which originate at the nodes and if any, on the incident incoming links the bits are transmitted (Figure 8.3).

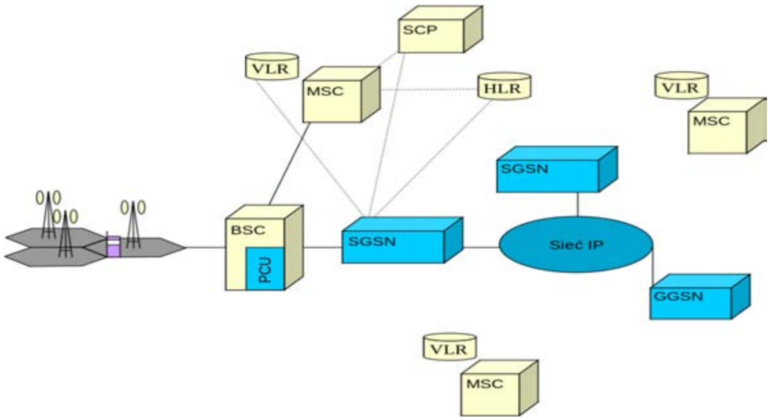



**Figure 8.3:** Network architecture.

*Source: Image by Wikimedia Commons.*

## 8.5. RANDOM NETWORK CODING (RNC) SCHEME FOR DATA DISTRIBUTION

The efficiency of content distribution applications can be increased through coding techniques in the application-level multicast or in ad hoc networks. In this case, these applications benefit from increased efficiency.

The seminal promises of NC have catalyzed most of the results because the forwarded information can be combined by the nodes.

Incoming packets are combined linearly and transmitted by each peer in random network coding (RNC) with a random selection of the coefficients. Due to linear coding, the computational cost causes the deployment of NC to be limited at the application level, for example in video streaming or P2P file sharing.

Currently, rateless erasure codes (a specific a class of erasure codes) have been designed in response to these complexity issues. Regarding the transmission of coded data, this has turned out to be a practical tool.

The number of transmitted and coded symbols can be adjusted easily in this family of erasure codes. An arbitrary number of coded symbols can be generated by a rateless encoder, unlike the rate characterized in the standard channel codes, which is selected in the design phase.

Digital fountain (DF) is the approach used for the transmission of such codes as till the time all the sinks or interested receivers receive the requisite number of symbols for decoding to be successful the transmitter, like a fountain keeps emitting the coded symbol.

In a wireless broadcasting application, many recipients are served by a single source whereby different channel conditions are experienced by the various recipients, and these kinds of applications are benefitted by such paradigm.

On the encoder side, among the original packets, simple XOR operations are required to be performed by the rateless codes as far as the added computational costs are concerned so that on the decoder side in a Galois field sparse linear system of order 2 can be solved.

From multiple peers' content is downloaded concurrently by a peer in most of the P2P applications that are deployed, which is then uploaded towards multiple peers.

There is a requirement of content reconciliation policies even though the network dynamics can be counteracted and the overall utilization of the bandwidth too can be improved. At the same time, irrespective of the peer who has contributed the coded information, the usefulness of each of it is equal and herein lays the advantage of coding whereby it simplifies the content reconciliation problem.

In principle, the content reconciliation issue gets relaxed when coding is used as the original set of packets may generate any number of uniquely coded packets without any limit at all. As a result, to propagate the information there can be the adoption of a much simpler push approach.

However, novel issues too can arise due to coding. Specifically, the computational complexity involved in encoding and decoding the coding blocks into which information flows have to be divide may in practice render the approach unfeasible.

Using rateless codes, a RNC Strategy in GF (2), which is practical, has been proposed in this chapter due to these motivations. For low communication overhead due to an excessive amount of transmission of coded packets that are redundant and propagation of information at a fast pace, the goal here is the designing and exploitation of DF principle that has a higher efficiency.

For the pre-encoding of the source information Luby Transform (LT) codes have been used. Most of the times, a node can possibly send a packet through linear combinations of the packets that it has received earlier where it has not as yet decoded them by calculating in GF (2). This chapter contributes mainly by:

- Modeling the strategies analytically so that they can be used to combine and forward coded packets.
- Exploiting the analytical results from earlier so that novel approaches can be designed whereby the aim is to fill the performance gap between the general RNC and LT-based solutions. It has been shown specifically that the number of duplicate packets can be reduced tremendously when in the initial phase itself, the upload bandwidth of the peers is throttled.
- Showing that there can be modification of the OFG decoding algorithm that has been proposed previously so that a linear combination of the coded packets that are received can be obtained at a single XOR operation’s cost, which is over and above the computational savings that result from using LT codes.
- The derivation of a large number of experiments in the results section where in terms of the computational costs as well the delivery times, the proposed approaches have been compared with the literature.

A reconciliation algorithm has been proposed that trades off complexity and accuracy. An attempt to coordinate the downloading of the content by the peers that are participating to the overlay through a design family of reconciliation techniques that has been tested in a real test-bed has been designed by Byer et al. where the coordination is attempted through both coding of the original packets as well as recoding of the data that has already been coded.

Based on a distributed rateless code’s optimal design wherein by construction there is a guarantee of the coded packets being as useful as they are independent, a complementary approach has been presented whereby the need for reconciliation has been avoided.

All the same, the solution can be generalized only once it is assumed that the overlay connections’ perfect knowledge is available and is limited to common relay node single network topology.

In a real dynamic overlay, this second assumption is not really feasible. Furthermore, as the number and size of the connections increase in the overlay, the complexity of the optimization algorithm also increases.

Another coding approach that is optimal has been proposed. Re-encoding, where while moving from hop to hop several coding stages are cascaded forms the basis of this solution. In the intermediate node, major computational expenses are involved in recoding which is over and above being asymptomatically optimal which may result in excessive overhead in coding at the practical level with limited code block length and several hops.

Optimal peer selection and adoption of rateless coding is adopted by P2P streaming application wherein an approach through which reconciliation phase can be canceled and which is simpler has been presented.

Propagation of the coded information is not allowed to the peers before the entire block is decoded and every P2P connection is applied with the DF approach. Hence, every peer sends LT encoded packets that are independent only after waiting for the block of original packets to get completely decoded even though it is involves an additional delay.

Due to the LT codes' performance decoding complexity and low decoding is involved in this strategy, and as before the nodes start relaying, they have to wait for the whole block to get decoded, a large amount of time is required to spread information in a network.

In order to reduce the end-to-end latencies to the minimum high-quality streaming topologies are aimed to be constructed through an optimal overlay formation strategy which is coupled with the push approach that is proposed.

Better performance of the combined packets is proposed to be achieved through what is called the RNC strategy. In (GF)q, using a linear combination of its input symbols, a new packet is created by the seeded where a sufficiently large integer is represented by q.

By a linear combination of the packets that have been received previously in GF(q), new encoded packets are created in the network's other nodes. As brought forth, the computational complexity involved in a real scenario makes this strategy difficult to apply as the chapter has proven that the value of q should be more than the number of peers that are there in the network.

Till the time a peer receives a specific number of packets, it waits to relay packets so that it can deal with this problem. Even if the performance of RNC is continued to be considered poor, the time required for spreading information can thus be reduced, but as q increases, in GF(q), the computational complexity of encoding and decoding continues to increase and stays very costly.

## 8.6. MODELING OF RELAYING STRATEGIES

The design of relaying strategies that are effective with respect to coded packets in the overlay network between two peers have been supported by two analytical models that have been developed in this section. In the first place, a set of coded packets that are to be relayed are randomly selected to quantify the effect on nodes by saturating their upload bandwidth.

It has been proven time and again that the probability of providing useful packets to the receiving nodes is higher when before forwarding the received packets, their linear combinations are made, implying that there can be a profitable exploitation of RNC in GF (2) as well.

Where in a P2P overlay network T, the components of a distributed application have organized have been considered here. Between the cycles and pairs of peer's multiple paths are allowed as no hypothesis has been made on the formation of T.

For all the peers, valuable information is held by a single peer (the original source). All the other peers cooperate in obtaining the information at the start-up as all of them have an interest in retrieving it. Coded packets that are useful in the buffer OB are stored in every peer.

A received packet with no linear dependency on the previously received packets is said to be a useful coded packet. From its buffer OB packets are allowed to be combined and forwarded by each peer. The download (Bd) and upload (Bu) bandwidths expressed in bps characterizes the peers.

For a given packet size, the peer bandwidths need to be expressed in packets per second (pps) in the sequel. The peer bandwidths are denoted as Nu and Nd in this case. The number of neighbors to which the information is uploaded ($z_u$) as well as number of from which information is downloaded ($z_d$) also characterizes each peer.

## 8.7. RATE-LESS CODES NETWORK CODING (NC)

Video and TV services are being offered over the internet through SopCast, PPLive, Coolstreaming, and TUV player and a number of other P2P streaming applications that are becoming increasingly popular and present an innovative way to do the same. A lot of research is being conducted in the field of P2P video streaming solutions' designs that are ISP friendly and efficient even as many issues are still to be resolved. Successful P2P file sharing experience is one of the starting points wherefrom a number of solutions have been designed.

A pull-based approach is used to exchange chunks of video stream (into which it has been divided) across the overlay. Exploitation of parallel downloads is allowed to each participant for which the updated information about the chunks owned by the involved peers must be signaled by them.

The system is made robust to the overlay dynamics caused by peers' churning by the adoption of solutions like reconciliation at the receiver and delivery redundancy. Some of the open issues in P2P video streaming can be tackled by allowing the combining of information packets rather than their forwarding simply by the nodes in the network in the NC paradigm.

To take an example, a push P2P streaming algorithm is designed by using RLNC without there being a requirement of explicit chunk requests. For P2P data distribution, an efficient and practical form of NC has been recognized as the usage of rate-less codes.

From k (1+), a random set of coded packets, there can be the recovery of k information packets by using rate-less coding wherein for large values of k 0 is approached. Furthermore, the data source can generate coded packets of an arbitrarily large number on the fly.

LT codes which are a particular class of rate-less codes has been used in this work. The issue of reconciliation can be eliminated by using the principle of rate-less coding as up to the collection of k (1+) packets coded packets can be relentlessly downloaded so that the retrieval of the corresponding k video chunks and decoding LT is allowed.

As from the point of view of decoding each packet sent by a node is equally important the algorithm becomes robust to peer churning by using rate-less codes, which makes it easier in a random overlay neighborhood to find information that is useful.

The real-time constraints of the video experience can be impacted by the decoding delay that is introduced when the k video packets are applied with LT coding, e.g., playback delays and excessive start-up. As far as the original k video chunks have been decoded, a new source of LT-coded packets rests in every peer in the overlay.

Depending on the code block size k, some delay in data transmission is introduced through this approach. All the same, only where the value of k is large LT coding behaves optimally. The coded symbols that have already been coded are allowed to be re-encoded by a coding technique that has been proposed.

In terms of rate overhead and complexity, this solution is not really practical, and it causes several LT coding stages to cascade. Presents one distributed rate-less code design which, however, can be generalized only when perfect knowledge of the overlay is assumed and is limited to the case with a common relay node and a single network topology.

It is guaranteed that an asymptotically optimal rate-less code is exploited by every node in the network through an efficient relaying policy which is coupled with the proposed protocol that is based on LT encoding. As a result, the solution becomes quite general and very simple.

## 8.7.1. Streaming Protocol

LT coding is applied by dividing the video stream into generation; a sequence of k packets xm i, i = 0,…, k–1, m ≥ 0, of constant size L constitute the m-th generation. A video stream access unit (i.e., the pictures' group via which the video frames are encoded), is corresponded with a generation in practical applications. The yield is an arbitrarily large number of packets $y_m$ j = k–1 i=0 gm i, $j_{xm}$ i, j ≥ 0 as within each generation LT coding is performed.

The robust soliton distribution (RSD) τδ, c(d), d=1,…, k where δ is the allowed failure probability at the decoder and c is a positive constant that is suitable must be followed by the packet degree d, i.e., the number of information packets that are used for the generation of a coded packet.

By the usage of a simple message passing decoder, the k xm I can be retrieved through any set of (1=) k coded packets using the RSD guarantees. Waiting for degree 1 packets, the coded packets $y_{mj}$ are buffered at any receiving node. The degree of the incoming ones and all the other coded packets, namely the ones that have already been buffered, is lowered by using the corresponding video packet, which is decoded as soon as a degree 1 packet is observed. With a limited overhead the k video packets can be decoded by iterating this procedure.

The decoder must be provided with the packet indexes that are used to generate a coded packet, i.e., i: gm i, j = 1. Selection of packets on both the decoder and encoder side and synchronized pseudo-random generators for the degree are used to accomplish this.

A sequential counter allows the generation of the pseudorandom outcomes that are the same on the decoder side via which every coded packet is identified and each peer is associated to a unique random set for this purpose.

To take an example, independent of the value k, the generation of combinations of 65536 coded packets is permitted by a 16-bit counter's signal, and for reasonably sized packets, the overhead rate is negligible.

Signaling of the gm i, j would cost k bits per packets and for this approach is far more convenient with regard to it. Where the application-level connections are modeled as edges between peers and a peer is represented by a vertex, a size N graph T can be modeled for the P2P network with active peers, N.

The content producer is s ∈ T, the SN which aims that the video is streamed to all the other overlay peers. It is assumed here that a path T connects to any peer p ∈T \ s. a static overlay topology is assumed here for the purpose of this study particularly, a static list of the nodes is there in each p ∈ T which are connected to its outgoing links.

A push protocol has been developed, which is referred to as the store and encode (SE), and this is used as the starting point. A number of coded packets y0 j are generated by s, the only data source at start-up whereby the upload capacity Bu gets saturated.

In a round-robin fashion, a packet scheduling policy which is quite simple, is assumed so that coded packets can be sent by s to all its outgoing edges. Progressive LT decoding is executed by every receiver p which waits for the coded symbols (store phase).

P starts behaving as a new source code as soon as it decodes the first packet generation x0 i, i = 0,…, k–1 as it starts to generate coded symbols that are novel and independent. The unique random generator is used to achieve this.

As a result, irrespective of all the other information that is being propagated in the overlay, these new packets have a high probability of being quite independent and the nodes that are still waiting for the generation m=0 can use them, whereby through the exploitation of the independent and multiple sources on their incoming edges their download rate can be increased.

A limited download capacity Bd characterizes every peer p to make the simulations realistic and the exceeded packets are dropped when Bd is less than the aggregated upload rate from its source.

The node that has decoded the m-th generation starts storing the generation m+1 and becomes a source for ∀xl i, l ≤ m so that real-time video streaming can be allowed. It is assumed that a peer requests for the generation's periodic signaling. The coded information is propagated and encoded independently by every peer in the network so that the SE protocol can be taken to be a kind of NC. It is always preferred that a node propagates coded information immediately, however, block decoding delays hamper SE.

This chapter proposes encode (RE) protocol and LT relay. The aim is to enable the instantaneous propagation of the received packets by the node. With the sole constraint that only once (relay phase) can every ym j be forwarded, the received ym j is allowed to be relayed by p the peer that waits for the m-th generation.

This simple rule assures that in the absence of loops optimal LT coded symbols are carried by the $p \in T$ received coded data and guarantees that across the overlay links, no duplicated packets flow. It is worthwhile to note that the upload rate during the relay phase gets limited due to this constraint.

Coded packets of generation m+1 can be relayed concurrently by p as it switches to the encode phase when decoding of generation m is possible for it like in SE thereby making it an independent source for packet generation $l \leq m$. As far as the propagation delay is concerned, the performance can be really improved by the relaying opportunity. The adoption of a technique via which duplicated packets can be detected there can be a relaxation in the hypothesis on the absence of loops in the overlay.

The address of the source of the incoming packets is stored by every peer and a ranking index is associated with each sender wherein the associated sequential counter recognizes the duplicated packets. On the receipt of every original packet their sender ranking index gets incremented, whereas on the other hand, the same index gets decremented by every duplicate packet.

When a certain source the ranking index is below the threshold, i.e., a lot of packets are being pushed by the sender, there is pruning of the incoming links from the overlay.

## 8.8. BENEFITS OF RANDOMIZED CODING OVER ROUTING

In networks with a special structure, major capacity gains can be seen through NC as a superset of routing. Even though centralized optimal routing is difficult to set up, it offers several benefits. Two types of network scenarios where specific usefulness can be availed from distributed random linear coding have been considered in this section (Figure 8.4).

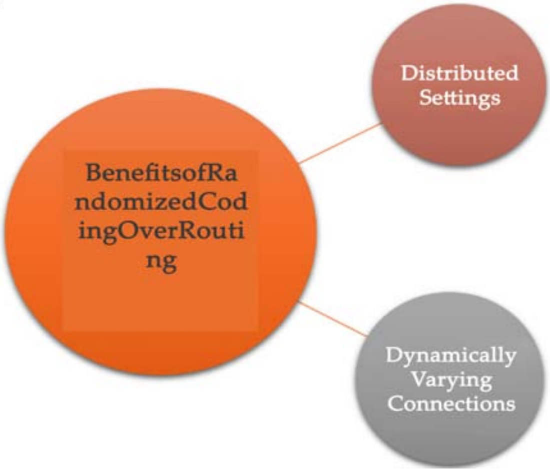



**Figure 8.4:** Benefits of randomized coding in different settings.

### 8.8.1. Distributed Settings

At the level of network nodes, the maintaining of reliable routing state may be infeasible or even expensive where the networks have changing topologies or/and a large number of nodes. Such issues are addressed in the proposed distributed randomized routing schemes.

High efficiency can still be achieved by not allowing different signals to be combined. For instance, let us consider a problem where the on a rectangular grid network, two processes need to be sent to the receiver nodes in unknown locations from the SN.

The acrylic delay-free case is analyzed for the sake of simplicity where burst, synchronized or pipelined operation is corresponded to where when on all the incident incoming links of v reception of transmission takes place, each transmission at a node v occurs.

Where no coordination is involved in the routing state or among nodes and a distributed transmission scheme is desired to be used, it is aimed by the network that the probability of receipt of two distinct processes by any node is maximum.

For instance, in the random flooding scheme RE that follows, this can be done in a manner that flooding preserves message diversity:

- One process is sent in both directions on one axis and in both directions along the other axis in the other process by the SN.

- The information received by a node on one link is sent via three outgoing links (these nodes are the ones that pass through the SN along the grid).
- The information received by a node on two links is sent separately with equal probability one process on one of the outgoing links and the other on the link that remains.

With the following simple random coding scheme RC, the same rectangular grid problem has been considered for the sake of comparison:

- One process is sent on one axis in both directions and the other process along the other axis in both directions by the SN.
- The information received by the node on one link is sent on its three outgoing links.
- On each of the two outgoing links a random linear combination of the source processes is sent by the node that receives information on two links.

### 8.8.2. Dynamically Varying Connections

Multisource multicast with connections that vary dynamically is another scenario where the RLNC can be of advantage. Distributed randomized coding is compared to an approximate online Steiner tree routing approach wherein a centralized fashion a tree is selected for each transmitter.

One tree per connection has been used as in the dynamic scenario considered the complexity of setting up each connection is a consideration of major significance. Compared to NC, a smaller performance gap may be achieved by using multiple Steiner trees in more complicated than online routing approaches.

A simulation-based approach has been used as it is difficult to analyze sophisticated routing algorithms. Trials were run on graphs that had been randomly generated with parameters like: transmission range, number of sources, number of nodes, number of receivers, maximum in-degree and out-degree.

There was a uniform scattering of nodes over a unit square for each trial. In order to create an acyclic graph, the direction of each link was chosen to b from the lower numbered node to the higher numbered node and nodes were ordered by their coordinate.

Up to a maximum out-degree and in-degree of the nodes involved, within Euclidian distance of each other, any pair of nodes was connected by a link. From amongst the lower numbered half of the nodes, random choosing (with replacement) of the SNs was done, and from amongst the highest-numbered nodes, the receiver nodes were chosen.

The values of the parameters were chosen in a manner that ensured the resulting random graphs were connected in general, and some of the connections that were desired could be supported, whereas they were simultaneously small enough so that there was efficient running of the simulations.

A series of time slots were present in each trial. For each time slot, the source that was transmitting was either off, i.e., when information was not being transmitted, or it was on when the source information is transmitted. Each source that was on for the approximate Steiner tree routing algorithm was associated with a Steiner tree the link-disjoint from the others, connecting it to all the receivers.

With a probability of 1-$p_0$ a source stayed on when it was on at the beginning of each time slot alternatively it turned off and with a probability of 1-$p_0$ the source either stayed off when it was off or it underwent the following procedure in turn:

- To search for a Steiner tree, the algorithm was applied for the approximate Steiner tree routing algorithm, with the other sources that were on at that time the link-disjoint with their Steiner trees and that source was connected to all the receivers. If the Steiner tree was not found, the source stayed off or was blocked, and where it was found, the information was sent to all the receivers through the Steiner tree by turning on the source.
- Up to three random linear network codes were selected for NC. The source was turned on where for transmitting information from that as also the other sources to all the receivers if one of them (the random linear network codes) was valid alternatively, the source was blocked.

For windows of 250 time slots, the average throughput was calculated till the time these measurements reached a state where they were steady, i.e., to avoid transient initial start-up behavior the measurements in three consecutive windows were within a factor of 0, 1 from each other and the frequency of blocked requests were used as performance metrics.

The difference in performance was not quantified precisely by these simulations. Neither the online routing nor the overhead of random linear coding have been very useful as far as the preliminary indication goes.

RLNC has shown to outperform the Steiner tree heuristic in simulations with respect to the blocking probability and throughput on a randomly constructed graph's non-negligible set showing that benefits can accrue from coding, which are not restricted to a few examples that have been carefully chosen when connections vary dramatically.

The coefficient vector sent with each packet or block, the decoding operations at the receivers and the linear coding operations at each node give the additional overhead to NC.

The coding field size affects each of these kinds of overheads. For large enough field sizes, the optimality of random linear coding is guaranteed by the previous section's theoretical bounds; however, for the problems of worst-case network connection, they are tight.

For networks that have a moderate number of nodes, the simulations illustrate as to what kind of field size would be required in practice.

For this purpose, small field sizes are used whereby random linear coding is allowed to match generally the Steiner heuristic's performance, and also where Steiner tree routing is difficult due to the topology of the networks, it allows the same to be surpassed. For networks of 8–10 nodes the applicability of network with shortcode lengths of 4–5 bits have been shown in the simulations.

## 8.9. CONCLUSION

In conclusion, this chapter has presented an overview of the significance of the security against the adversarial errors. Several error correction bounds used in centralized NC have been mentioned, such as upper bounds, lower bounds, along with the generic linear codes.

In this chapter, detection of adversarial errors as well as RLNC for arbitrarily correlated sources has been discussed. Various strategies exists that can be used in the modeling of a relay.

This chapter also discusses various NC schemes that can be implemented in RLNC for data distribution. Finally, a brief understanding of rate-less codes used in NC has been discussed.

# REFERENCES

1. Bioglio, V., Grangetto, M., Gaeta, R., & Sereno, M., (2013). A practical random network coding scheme for data distribution on peer-to-peer networks using rateless codes. *Performance Evaluation, 70*(1), 1–13.
2. Grangetto, M., Gaeta, R., & Sereno, M., (2009). *Rateless Codes Network Coding for Simple and Efficient P2P Video Streaming*. [Online] Ieeexplore.ieee.org. Available at: https://ieeexplore.ieee.org/document/5202788 (accessed on 3 May 2021).
3. Ho, T., & Lun, D., (2008). *Network Coding: An Introduction*. [Online] Researchgate.net. Available at: https://www.researchgate.net/publication/265229690_Network_Coding_An_Introduction (accessed on 3 May 2021).
4. Ho, T., Médard, M., Koetter, R., Karger, D., Effros, M., Shi, J., & Leong, B., (2006). *A Random Linear Network Coding Approach to Multicast*. [Online] Authors.library.caltech.edu. Available at: https://authors.library.caltech.edu/5107/1/HOTieeetit06.pdf (accessed on 3 May 2021).

# Index

# Network Coding

Network coding (NC) is a method of improving the flow of digital data in a network via transmitting the digital proof about the messages. The "digital evidence" is a combination of two or more messages. When the bits of the digital evidence reach the destination point, the transmitted message is deduced in spite of directly reassembled. Network coding was first introduced in 1999 as a substitute to routing. This book takes the readers through a brief introduction into the network coding. This book sheds light on the history of network coding, theoretical framework for network coding, applications, and challenges of network coding, throughput benefits, use of network coding in wireless systems, and security against adversarial errors. The first chapter stresses the basic overview of network coding so that readers are clear about the philosophies and history behind that form the utmost basics in the field. This chapter will also emphasize the development of network coding across the several years, and also explains some important network security protection models. The second chapter takes the readers through the concepts network multicast model, along with the various frameworks such as algebraic framework and network coding framework. This chapter will provide highlights on the application of index coding with the help of linear programming as well as various network coding-based multipath routings. Then, the third chapter explains the implementation of network coding in wireless systems. It also explains the various network coding schemes, wireless mesh networks, and wireless sensory networks. This chapter also sheds light on the theory behind the PNC. The fourth chapter introduces readers to the use of network coding in mobile applications. This chapter also explains the significance of various transmission approaches such as unicast, multicast, systematic network coding, and pure network coding. The fifth chapter throws light on the network coding for multicast networks, peer-to-peer (P2P) network coding, and application layer multicast (ALM). This chapter also addresses the difference between network coding and random coding, along with various protocols and algorithms that have been used in network coding. The sixth chapter takes readers to the benefits of throughput in network coding, the concept of randomized network coding. Readers are then told about the various vector network coding algorithm, and code design algorithms such as code design for vector and scalar coding. The seventh chapter explains the various applications and challenges in network coding that have been addressed in the last several years. This chapter also emphasize the application of network coding in mobile and wireless sensory systems, cloud, and distributed storage systems. This chapter also explains the various limitations of network coding. The last chapter of this book sheds lights on the error correction bounds for centralized network coding. This chapter also mentions the detection of adversarial errors, various modeling of relaying strategies, as well as the benefits of randomized coding over routing.
This book has been designed to suit the knowledge and pursuit of the researcher and scholars and to empower them with various aspects of network coding, the development of network coding and its applications in various sectors so that they are updated with the information.

Adele Kuzmiakova is a computational engineer focusing on solving problems in machine learning, deep learning, and computer vision. Adele attended Cornell University in New York, United States for her undergraduate studies. She studied engineering with a focus on applied math. While at Cornell, she developed close relationships with professors, which enabled her to get involved in academic research to get hands-on experience with solving computational problems. She was also selected to be Accel Roundtable on Entrepreneurship Education (REE) Fellow at Stanford University and spent 3 months working on entrepreneurship projects to get a taste of entrepreneurship and high-growth ventures in engineering and life sciences. The program culminated in giving a presentation on the startup technology and was judged by Stanford faculty and entrepreneurship experts in Silicon Valley. After graduating from Cornell, Adele worked as a data scientist at Swiss Federal Institute of Technology in Lausanne, Switzerland where she focused on developing algorithms and graphical models to analyze chemical pathways in the atmosphere. Adele also pursued graduate studies at Stanford University in the United States where she entered as a recipient of American Association of University Women International Fellowship. The Fellowship enabled her to focus on tackling important research problems in machine learning and computer vision. Some research problems she worked on at Stanford include detecting air pollution from outdoor public webcam images. Specifically, she modified and set up a variety of pre-trained architectures, such as DehazeNet, VGG, and ResNet, on public webcam images to evaluate their ability to predict air quality based on the degree of haze on pictures. Other deep learning problems Adele worked on include investigating the promise of second-order optimizers in deep learning and using neural networks to predict sequences of data in energy consumption. Adele also places an emphasis on continual education and served as a Student Leader in PyTorch scholarship challenge organized by Udacity. Her roles as the Student Leader were helping students debug their code to train neural networks with PyTorch and providing mentorship on technical and career aspects. Her hobbies include skiing, playing tennis, cooking, and meeting new people.

ISBN 978-1-77469-287-5

ARCLER PRESS