AN ABSTRACT OF THE DISSERTATION OF

Duong Nguyen-Huu for the degree of Doctor of Philosophy in Electrical and Computer Science presented on May 9, 2016.

Title: Network Coding, Random Matrices, and Their Applications to Communication Systems

Abstract approved: _____

Thinh P. Nguyen

In this work, we study network coding technique, its relation to random matrices, and their applications to communication systems. The dissertation consists of three main contributions. First, we propose efficient algorithms for data synchronization via a broadcast channel using random network coding. Second, we study the resiliency of network coding based large distributed systems via characterization of minimum rank recovery of random matrices over finite field. Third, we propose a novel Location Assisted Coding technique to manage interference and increase capacity in Free Space Communication (FSO) systems. In the data synchronization problem, we assume there is a sender who broadcasts a set of packets to a number of receivers. Each receiver is assumed to have a random partial set of the desired packets. The goal is to devise network coding algorithms to minimize the number of packets transmitted by the sender until all the receivers successfully receive the entire set of packets. We establish probabilistic bounds and asymptotic results on the minimum number of transmitted packets for three randomized algorithms. In the minimum rank decoding problem, the goal is to recover the network coded packets from a malicious attacker who randomly corrupts the header of the packets with limited magnitude errors. We cast this problem as the problem of rank recovery of random matrices over finite field in presence of noise. We present some initial asymptotic results on joint distribution of weight and rank of random matrices for simple models which are useful for the rank recovery problem. We show that limited magnitude

noise is likely not to decrease the rank of low-rank matrices with uniformly distributed weights. Finally, we show that the proposed LAC technique can increase throughput and reduce interference for multiple users in a dense array of FSO femtocells. Our theoretical analysis and numerical experiments show orders of magnitude increase in throughput using LAC over traditional approaches.

Network Coding, Random Matrices, and Their Applications to Communication Systems

by

Duong Nguyen-Huu

A DISSERTATION

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Presented May 9, 2016
Commencement June 2016

Doctor of Philosophy dissertation of Duong Nguyen-Huu presented on May 9, 2016.

APPROVED:

Major Professor, representing Electrical and Computer Science

---

Director of the School of Electrical Engineering and Computer Science

---

Dean of the Graduate School

I understand that my dissertation will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my dissertation to any reader upon request.

Duong Nguyen-Huu, Author

ACKNOWLEDGEMENTS

First, I would like to show my great gratitude to my advisor, Prof. Thinh Nguyen, for giving me the opportunity to start my journey in US and for his insight, great guidance during my studies at Oregon State University.

Also, I would like to thank my committee members for reviewing this work and giving insightful and valuable comments. I am very grateful to my colleagues and members in my research group for their useful and interesting discussions.

Next, I would like to thank all my friends in Corvallis for not only helping me with life but also making the stay here enjoyable and memorable.

Last but not least, I would like to express my special thanks to my family for their continuous support and encouragement.

TABLE OF CONTENTS

Page

TABLE OF CONTENTS (Continued)

Page

TABLE OF CONTENTS (Continued)

Page

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF TABLES

Chapter 1: Introduction

1.1 Network Coding and Motivation

Since Internet appeared in the early 1960s, the number of users has been increasing considerably. Every aspects of Internet such as the underlying infrastructure, protocols, services, networks etc. also have been evolving continuously. Importantly, increasing the capacity has been always been one of the primary aims of making the Internet better. There have been many advances, from hardware and architecture to protocol design to boost the Internet capacity. Notably, the past decade has witnessed a spurt of research on using Network Coding (NC) to increase capacity, reliability, and security. The concept of NC was first introduced in 2000 by R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung [3]. In their work, NC is considered as a method of optimizing the digital data flow in a network by transmitting digital evidence about messages. The robustness of NC in term of bandwidth increase and attack resistance since then has been attracted many interests from research community. As a result, NC has been applied in many practical network systems and applications such as broadcast network, peer-to-peer network, distributed system and data centers, etc.

1.2 Contribution of This Dissertation

In this thesis, we study network coding aspects in three different settings. First, we propose efficient algorithms for data synchronization via a broadcast channel using random network coding. Second, we study the resiliency of network coding based large distributed systems via characterization of minimum rank recovery of random matrices over finite fields. Third, we propose a novel Location Assisted Coding technique to manage interference and increase capacity in Free Space Communication (FSO) systems.

1.2.1 Data Synchronization via Network Coding

In the first problem, we investigate the problem of data synchronization in which a sender has a set of packets to be distributed to all the receivers via a broadcast channel. Initially, each receiver has some fraction of the packets. At each time slot, the sender might broadcast a packet to all the receivers. The goal is to find a broadcast scheme that minimizes the number of time slots until all the receivers successfully obtain all the packets. We propose two probabilistic models on how the initial fractions of packets at receivers are distributed. These models arise naturally in many large scale systems such as Peer-to-Peer (P2P) networks, data centers, and distributed storage systems. Based on these models, we establish probabilistic bounds and asymptotic results on the minimum number of time slots to successfully transmit all the packets to all the receivers. Such bounds can shed lights on the benefits and limitations of using NC-based broadcast schemes in certain real-world settings. Next, we propose and analyze a number of random network coding algorithms for finding the approximately optimal solution. Our analysis provided quantitative performances in terms of expectation, variance, and tail probability on the number of time slots required to complete the synchronization for the proposed algorithms.

1.2.2 Data Recovery in Network Coding

In the second problem, NC technique is considered under attack and we show that the security can be improved using Minimum Rank Problem. In the minimum rank decoding problem, the goal is to recover the network coded packets from a malicious attacker who randomly corrupts the header of the packets with limited magnitude errors. We cast this problem as the problem of rank recovery of random matrices over finite field in presence of noise. We present some initial asymptotic results on joint distribution of weight and rank of random matrices for simple models which are useful for the rank recovery problem. We show that limited magnitude noise is likely not to decrease the rank of low-rank matrices with uniformly distributed weights.

1.2.3 Network Coding technique in WiFO system

In the third problem, we describe the WiFO system that can provide up to one Gbps per user while maintaining seamless mobility. While typical RF femtocells are non-overlapped to minimize inter-cell interference, there are advantages of using overlapped femtocells to increase mobility and throughput when the number of users is small. The contribution will be a novel location assisted coding (LAC) technique (a specific NC technique) used in the WiFO network that aims to increase the throughput and reduce interference for multiple users in a dense array of femtocells. Our theoretical analysis and numerical experiments show orders of magnitude increase in throughput using LAC over traditional approach, which verifies the robustness of NC technique.

1.3 Organization of This Dissertation

The thesis is organized as follows. We first present the background and preliminaries of our work in Chapter 2. In Chapter 3, we describe the problem of Data Synchronization via Network Coding. In Chapter 4, the problem of Data Recovery in Network Coding is introduced. Next, the novel WiFO systems and the use of Network Coding is described in Chapter 5. Finally, we conclude the thesis and propose future research directions in Chapter 6.

Chapter 2: Background

In this section, we briefly describe the definitions and preliminaries that will be used in this thesis. More details can be founded in [64], [88], [63], [36], [11].

2.1 Finite fields

Definition 1. *A finite field $\mathcal{F}$ or $GF(q)$ or $F(q)$ is a field that contains a finite number of elements (q) in which the four basic operations multiplication, addition, subtraction and division (excluding division by zero) are defined.*

The simplest type of finite fields is the prime fields.

Definition 2. *For each prime number q, the field $GF(q)$ can be presented by integers in the range $0, \ldots, q-1$ and can be constructed as the integers modulo q.*

Example 1. *When $q = 2$, the field $GF(2)$ contains only two elements: 0 and 1. The four operations are illustrated in the Table 2.1, Table 2.2, Table Table 2.3 (addition is identical to subtraction).*

$*$	0	1
0	0	1
1	1	0

Table 2.1: Multiplication in $GF(2)$

$+\|-$	0	1
0	0	1
1	1	0

Table 2.2: Addition in $GF(2)$

x/y	0	1
0	X	0
1	X	1

Table 2.3: Division in $GF(2)$

The non-prime fields $GF(q^n)$ can be constructed over prime fields $GF(q)$.

Definition 3. *The field $GF(q^n)$ is a finite field in which the elements are the polynomials of degree less than n over $GF(q)$.*

The $GF(q^n)$ is constructed using an irreducible polynomial P in $GF(q)[X]$ of degree n.

Example 2. *Let $q = 2, n = 2$, we construct $GF(2^2)$ using $P(X) = X^2 + X + 1$ and α where α is the root of P as in Table 2.4.*

$GF(2^2)$	α	1
0	0	0
1	0	1
α	1	0
α^2	1	1

Table 2.4: Construct $GF(2^2)$

The arithmetic operations for $GF(2^2)$ are illustrated in Table 2.5, Table 2.6, Table 2.7 and the subtraction operation is identical to addition.

$*$	0	1	α	α^2
0	0	0	0	0
1	0	1	α	α^2
α	0	α	α^2	1
α^2	0	α^2	1	α

Table 2.5: Multiplication in $GF(2^2)$

$+\|-$	0	1	α	α^2
0	0	1	α	α^2
1	1	0	α^2	α
α	α	α^2	0	1
α^2	α^2	α	1	0

Table 2.6: Addition in $GF(2^2)$

x/y	0	1	α	α^2
0	X	0	0	0
1	X	1	$1+\alpha$	α
α	X	α	1	α^2
α^2	X	α^2	α	1

Table 2.7: Division in $GF(2^2)$

2.2 Matrices and Rank

Definition 4. *The rank of a matrix is defined as the maximum number of linearly independent column (or row) vectors in the matrix.*

In this thesis, we use following notations:

- $rank(A)$: denote the rank of matrix A.
- Matrix $A \in F_q^{m \times n}$: denote that matrix A has m rows and n columns and all entries in matrix A belong to field $F(q)$ and we use $0, \ldots, q-1$ to represent elements in field $F(q)$.

Definition 5. *When all the column (or row) vectors in a matrix are linearly independent, the matrix is said to be full rank .*

2.3 Gaussian Elimination

Definition 6. *For a non-zero row in a matrix, the left-most non-zero entry is called the leading coefficient in this row.*

Definition 7. *A matrix is said to be in row echelon form if for any non-zero rows, the leading coefficient is to the right of the leading coefficient in the row above (if any).*

Definition 8. *A matrix is said to be in reduced row echelon form if it is in row echelon form and every leading coefficient is equal to* 1 *and is the only non-zero entries in its column.*

Example 3.

$$A = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

Matrix A is in row echelon form but not reduced row echelon form.

$$B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Matrix B is in reduced row echelon form. All the leading coefficients are in red color.

Following are three types of row operations which can be applied to a matrix:

1. Swap the position of two rows.
2. Multiply a row by a scalar.
3. Multiple a row by a scalar then add to other row.

We note that perform row operations to matrix doesn't change the rank of this matrix.

Definition 9. *Gaussian elimination or row deduction is an algorithm or process using row operations to change a matrix to the row echelon form.*

Example 4. *We show how to apply Gaussian elimination to change a matrix A to its*

row echelon form.

$$\begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \textit{(swap first row and third row)}$$

$$\rightarrow \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \textit{(subtract second row to third row)}$$

$$\rightarrow \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \textit{(subtract third row to first row)}$$

One of most well-known application of Gaussian elimination is to solve system of linear equations.

2.4 Linear Network Coding

To transmit a set of n packets $\mathcal{P} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_n\}$, we consider linear network coding (LNC) in $GF(q^n)$.

Definition 10. *A linear network coded (mixed) packet $\boldsymbol{c}$ is a linear combination of packets and is constructed as:*

$$\boldsymbol{c} = v_1\boldsymbol{p}_1 + v_2\boldsymbol{p}_2 + \cdots + v_n\boldsymbol{p}_n$$

where coefficients $v_i \in GF(q)$ and the addition operation is performed in $GF(q^n)$.

When packet $\mathbf{c}$ is transmitted, both the value of $\mathbf{c}$ and the set of coefficients $v_1, \ldots, v_D$ are forwarded. At the receivers, the original packets can be recovered when the matrix of coefficients are full rank in $GF(q)$.

The following data synchronization example is often used to describe the operation of network coding.

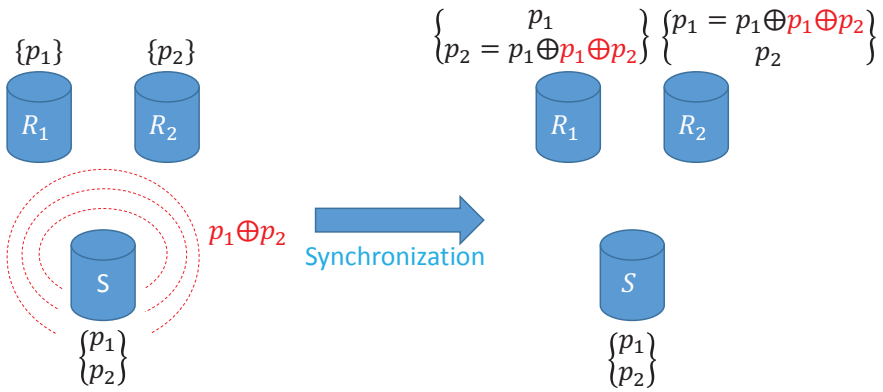


Figure 2.1: Network coding example

Example 5. *The example are illustrated in Fig. 2.1 using $GF(2)$. There are two receivers R_1 and R_2 and both receivers want to obtain both packets $\boldsymbol{p}_1, \boldsymbol{p}_2$. At each receiver already holds one of two packets. In this case, the coded packet $\boldsymbol{c}$ is generated and transmitted to both receivers in a broadcast channel as follows.*

$$\boldsymbol{c} = \boldsymbol{p}_1 + \boldsymbol{p}_2$$

At receiver R_1, the coefficient matrix

$$H_1 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

and at receiver R_2, the coefficient matrix

$$H_2 = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$$

Since $rank(H_1) = rank(H_2) = 2$ (full rank), both receivers can recover the original packets using addition operation in $GF(2)$.

Example 6. *The classic example to illustrate Linear Network Coding operation is presented in Fig. 2.2.*

In this network, source node S would like to transmit two packets a and b to sink nodes Y and Z while T, U, W are relay nodes. The mixing technique is performed first

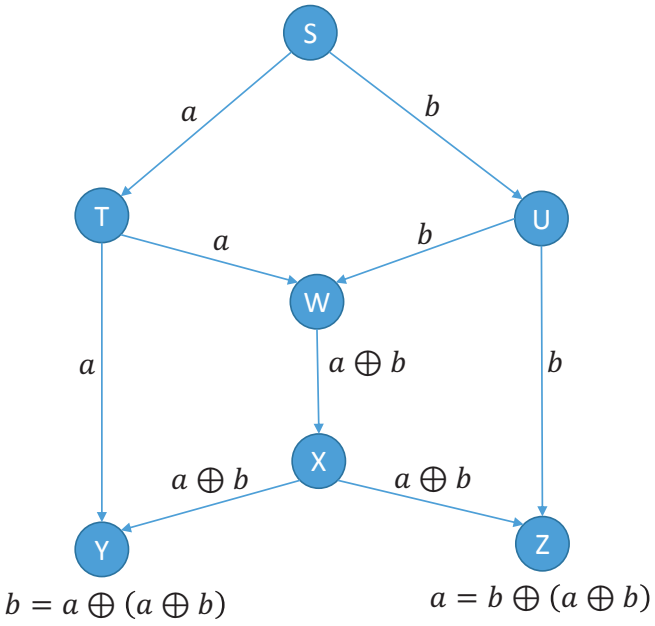


Figure 2.2: Butterfly topology - XOR Network Coding example

at node W where XOR operation of packet a and b are completed. At the sink nodes, XOR computations are applied again to retrieve packet b in node Y and packet a in node Z.

We note that the XOR is a special case of linear network coding while the addition are performed in $GF(2)$.

The random linear network coding can be considered as an advanced of linear network coding.

Definition 11. *In random linear network coding (RLNC), a coded packet is a linear combination of packets where the coefficients are generated randomly in a finite field.*

To present an example of RLNC, we also consider the same network topology in Fig. 2.3

Example 7. *In this example, some nodes produce random coded packets by generates random coefficients such as α_1, β_1 from node S to node T, α_2, β_2 from node S to node U and α_3, β_3 for node W to node X. At the sink nodes Y and Z. Now sink node Y*

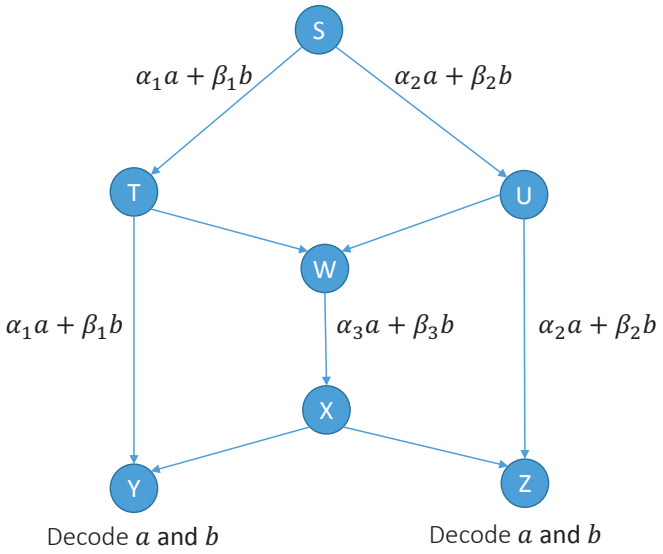


Figure 2.3: Butterfly topology - Random Network Coding example

receives two coded packets $\alpha_1 a + \beta_1 b$ and $\alpha_3 a + \beta_3 b$ while sink node Z receives two coded packets $\alpha_2 a + \beta_2 b$ and $\alpha_3 a + \beta_3 b$. Those coded packets form a system of linear equations and solving this give us packet a and b. The most significant advance of RLNC is that it does not depend on the network topology. However, the probability that those set of coefficients are independent need to be close to one so that at the sink nodes, the decoding process can be completed.

Next, we show that RLNC can be applied for any general-topology networks [57]. Consider a general network presented by a directed graph $G = (V, E)$. Assume that node v is the only source and z is the only sink in the network. Here, the input process of the source are presented as a vector $\mathbf{x}$

$$\mathbf{x} = [X(v, 1), X(v, 2), \dots]$$

and the output process of the sink are presented as a vector $\mathbf{z}$

$$\mathbf{z} = [Z(v', 1), Z(v', 2), \dots]$$

The relationship between input and output is represented by a transfer matrix M. The

RLNC solution exists only if the equation $\mathbf{z} = \mathbf{x}M$ has a solution. We show how to form the transfer matrix M with the following example.

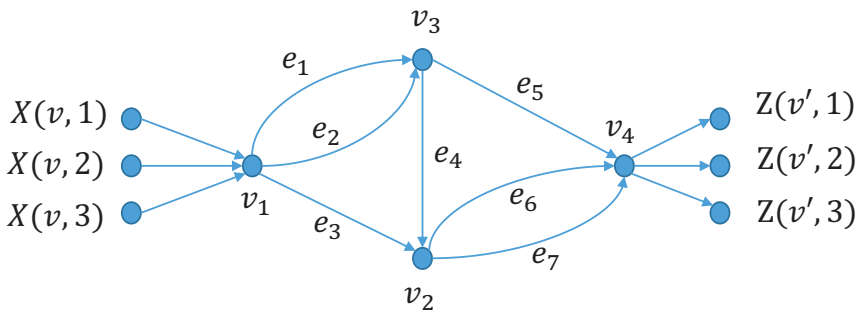


Figure 2.4: General topology - Random Network Coding example

Example 8. *The network topology are illustrated in Fig. 2.4. We have the following equations representing the random processes in the network:*

$$\begin{aligned}
Y(e_1) &= \alpha_{1,e_1}X(v,1) + \alpha_{2,e_1}X(v,2) + \alpha_{3,e_1}X(v,3) \\
Y(e_2) &= \alpha_{1,e_2}X(v,1) + \alpha_{2,e_2}X(v,2) + \alpha_{3,e_2}X(v,3) \\
Y(e_3) &= \alpha_{1,e_3}X(v,1) + \alpha_{2,e_3}X(v,2) + \alpha_{3,e_3}X(v,3) \\
Y(e_4) &= \beta_{e_1,e_4}Y(e_1) + \beta_{e_2,e_4}Y(e_2) \\
Y(e_5) &= \beta_{e_1,e_5}Y(e_1) + \beta_{e_2,e_5}Y(e_2) \\
Y(e_6) &= \beta_{e_3,e_6}Y(e_3) + \beta_{e_4,e_6}Y(e_4) \\
Y(e_7) &= \beta_{e_3,e_7}Y(e_3) + \beta_{e_4,e_7}Y(e_4) \\
Z(v',1) &= \epsilon_{e_5,1}Y(e_5) + \epsilon_{e_6,1}Y(e_6) + \epsilon_{e_7,1}Y(e_7) \\
Z(v',2) &= \epsilon_{e_5,2}Y(e_5) + \epsilon_{e_6,2}Y(e_6) + \epsilon_{e_7,2}Y(e_7) \\
Z(v',3) &= \epsilon_{e_5,3}Y(e_5) + \epsilon_{e_6,3}Y(e_6) + \epsilon_{e_7,3}Y(e_7)
\end{aligned}$$

Hence, the system matrix M can be computed as follows.

$$M = A \begin{bmatrix} \beta_{e_1,e_5} & \beta_{e_1,e_4}\beta_{e_4,e_6} & \beta_{e_1,e_4}\beta_{e_4,e_7} \\ \beta_{e_2,e_5} & \beta_{e_2,e_4}\beta_{e_4,e_6} & \beta_{e_2,e_4}\beta_{e_4,e_7} \\ 0 & \beta_{e_3,e_6} & \beta_{e_3,e_7} \end{bmatrix} B^T$$

where

$$A = \begin{bmatrix} \alpha_{1,e_1} & \alpha_{1,e_2} & \alpha_{1,e_3} \\ \alpha_{2,e_1} & \alpha_{2,e_2} & \alpha_{2,e_3} \\ \alpha_{3,e_1} & \alpha_{3,e_2} & \alpha_{3,e_3} \end{bmatrix}$$

and

$$B = \begin{bmatrix} \epsilon_{e_5,1} & \epsilon_{e_5,2} & \epsilon_{e_5,3} \\ \epsilon_{e_6,1} & \epsilon_{e_6,2} & \epsilon_{e_6,3} \\ \epsilon_{e_7,1} & \epsilon_{e_7,2} & \epsilon_{e_7,3} \end{bmatrix}$$

2.5 Convex Optimization

Definition 12. *A set $\mathcal{C}$ is called a convex set if and only if the convex combination of any two points in the set $\mathcal{C}$ also belongs to the set $\mathcal{C}$:*

$$\theta x_1 + (1-\theta)x_2 \in C \ \forall x_1, x_2 \in \mathcal{C}, \forall \theta \in [0,1]$$

Example 9. *Following are examples of convex set:*

- $\mathcal{C} = [a, b]$ *where* $a, b \in \mathbb{R}$.
- $\mathcal{C} = \mathbb{R}^2$.
- $\mathcal{C} = \{x \in \mathbb{R}^2 | \|x\|_2 \leq a\}$ *where* $a \in \mathbb{R}_{++}$.

Definition 13. *The domain of a function $f : \mathbb{R}^n \to \mathbb{R}$ is denoted as $\boldsymbol{dom}(f)$, and is defined as the set of points where f is finite:*

$$\boldsymbol{dom}(f) = \{x \in \mathbb{R}^n | f(x) < \infty\}$$

Definition 14. *A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if and only if $\forall x_1, x_2 \in \boldsymbol{dom}(f) \subseteq \mathbb{R}^n$ and $\forall \theta \in [0,1]$, we have*

$$f(\theta x_1 + (1-\theta)x_2) \leq \theta f(x_1) + (1-\theta) f(x_2)$$

Example 10. *Following are examples of convex function:*

- $f(x) = ax + b$ *where* $x \in \mathbb{R}$ *and given* $a, b \in \mathbb{R}$.

- $f(x) = e^{ax}$ *where* $x \in \mathbb{R}$ *and given* $a \in \mathbb{R}$.
- $f(x) = x\log(x)$ *where* $x \in \mathbb{R}_{++}$.

The standard form of convex optimization problem is written as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & g_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p. \end{aligned}$$

where:

- Objective function $f(x) : \mathbb{R}^n \to \mathbb{R}$ is a convex function
- Inequality constraints $g_i(x) \leq 0$ where g_i are convex
- Equality constraints $h_i(x) = 0$ where h_i are affine.

2.6 Probability Theory

Definition 15. *A random variable (RV)* $X : \Omega \to E$ *is a measurable function from the set of possible outcome* Ω *to set* E.

A RV X is called discrete RV or continuous RV depends on whether the set Ω is finite (or countably infinite) or uncountably infinite, respectively. Mathematically, a RV can be described by the probability distribution (or probability density function). Mean $\mathbf{E}[X]$ (expected value) and variance $\mathbf{Var}(X)$ are some of well-known metrics that can be used to describe a RV X.

Also, related to RV, Markov's inequality and Chebyshev's inequality are well-known results in probability theory that can be stated as follows.

1. **Markov's inequality**

   If X is a non-negative RV and for any $a > 0$, then we have

$$\mathbf{P}(X \geq a) \leq \frac{\mathbf{E}(X)}{a}.$$

2. **Chebyshev's inequality**

   If X is a RV and for any $a > 0$, then we have

$$\mathbf{P}\left(|X - \mathbf{E}[X]| \geq a\sqrt{\mathbf{Var}(X)}\right) \leq \frac{1}{a^2}.$$

Chapter 3: Data Synchronization via Random Network Coding

3.1 Introduction

Data synchronization plays a critical role in the performances of many emerging large scale distributed systems such as Peer-to-Peer (P2P) systems, distributed storage systems, and data centers. To provide high reliability in such systems, data are typically duplicated across multiple nodes in a network. In addition, many systems allow data to be updated asynchronously at individual nodes. As a result, potential data inconsistencies might arise across multiple nodes. For example, during the peak time, a data center [27], [46], [26] might allow data to be updated at individual servers autonomously for better performance. These changes are then propagated to other servers at an appropriate later time. During this interval, the data across the servers are inconsistent. In other systems, data inconsistencies at different nodes are resulted in a far less controllable way. Notably, in file sharing systems such as BitTorrent, peers might have different parts of the same file due to the random exchange of data among peers. Wireless broadcast is another example in which many users receive the same file broadcast from a base station. However, due to packet losses, for some given time, users might have different parts of the file. Thus, the aim of the data synchronization problem is to repair the data inconsistencies by broadcasting additional data to the receivers.

The data synchronization problem is an instance of the index coding problem [6], [18] that consists of a sender and a number of receivers sharing a common broadcast channel. The sender has a set of packets $\mathcal{A}$. Each receiver has a random subset of $\mathcal{A}$. At each time slot, the sender broadcast a packet that can be received by all the receivers. The goal is to find a broadcast scheme that minimizes the number of time slots until every receiver successfully receive the set $\mathcal{A}$. An approach to this problem is to use the Network Coding (NC) framework. NC framework treats each packet as an element in a finite field. Each coded NC packet is a linear combination of other packets. It is shown that when the finite field size is larger than or equal to the number of nodes, the problem can be solved in polynomial time [58], [19]. However, for arbitrary field size,

the synchronization problem as an instance of the index coding problem has been shown to be NP-hard [77], [30], [59], [8]. As such a number of heuristic schemes have been proposed [31], [17].

Contributions. In this chapter, we study the synchronization problem from a probabilistic viewpoint. First, we describe two probabilistic models on how subsets of packets at receivers are distributed. These models arise naturally in many large scale systems such as Peer-to-Peer (P2P) networks, data centers, and distributed storage systems. For these two models, we establish probabilistic bounds and asymptotic results on the minimum number of time slots that the sender needs to successfully transmit all the packets to all receivers. Such bounds can shed lights on the benefits and limitations of using NC-based broadcast schemes in certain real-world settings. Second, while the probabilistic upper and lower bounds for the optimal solution can be found, finding the algorithms for achieving the optimal solution is not trivial. Therefore, we propose and analyze a number of random network coding (RNC) algorithms for finding the optimal solutions. Our analysis provides quantitative performances in terms of expectation, variance, and tail probability on the number of time slots required to complete the synchronization for the proposed algorithms.

Outline. We first discuss a few related work in Section 3.2, then present the problem formulation and notations in Section 3.3. In Section 3.4, we describe two common models in which data inconsistency can occur. Based on these models, we show the probabilistic bounds on the optimal solutions of any broadcast scheme in Section 3.5. We then describe three NC-based algorithms to perform synchronization and their theoretical performance analysis in Section 3.6 and Section 3.7. In Section 3.8, we provide the simulation results for the proposed algorithms and finally a few concluding remarks in Section 3.9.

3.2 Related Work

There exists rich literature on NC on which our work is built upon. Due to limited space, we will discuss the similarities and differences between our work and a few representative work. Our work is closely related to the index coding problem [6]. Both problems consists of a number of receivers who want to receive an identical set of packets $\mathcal{A}$ from a sender. All the receivers share the same broadcast channel, and have different subsets of $\mathcal{A}$. The goal of both problems is to minimize the number of broadcasts by the sender until all the

receivers successfully obtain the complete set $\mathcal{A}$. On the other hand, our work differs in the following ways. First, instead of assuming the subsets of packets at the receivers are given as in most network coding literature [6], [43], we propose two probabilistic models to characterize the distribution of the subsets of packets. Based on these two models, we further study the asymptotic bounds on the optimal solution which, to our knowledge, has not been investigated previously. Specifically, we study how the number of packets varies as a function of the number of receivers as both become large, can affect the solution. Second, instead of solving the problem in a deterministic manner, we propose randomized NC algorithms to find the approximate optimal solution with probabilistic guarantees.

We note that in many existing NC literature, the information about the partial sets of packet at the receiver is assumed to be available at server. For many large scale distributed systems consisting many users and large data, this assumption might be impractical since the central server might need to store a substantial large amount of information. This assumption is not required in our problem. Instead, we introduce three different levels of information exchange between the sender and the receivers. That said, our work is on the simplicity of randomized network coding techniques [20], [43], [21], [54] that can be implemented in real world settings. In addition, our theoretical results have probabilistic flavor as contrast to the work in [29].

Our work can also be viewed as an instance of the Direct Data Exchange (DDE) problem that was first proposed by El Rouayheb et al. [81]. The DDE problem has attracted much interest from the research community [87] [86] [24] [98]. While the goal of both problems is to synchronize data in multiple receivers, there are essential differences. In the the DDE problem, all the receivers have to participate in broadcasting their data while in our problem, only one sender can broadcast. In addition, in the DDE problem, the subset of packets at each receiver can only be original packets while in our setup, we allow both mixed (network coded) and original packets in the subsets of the packets. Furthermore, most existing solutions to the DDE problem take a deterministic approach while ours has a probabilistic flavor.

That said, our work is very similar to the problem of wireless broadcast using network coding via lossy channel. For example, in a single-hop wireless network, where communication channels are lossy, NC techniques are used to help the receivers to recover the lost packets quickly [72], [91]. In wireless ad hoc network, NC techniques have been

also applied to increase bandwidth efficiency [66], [82]. In wireless mesh network, the advantages of NC compared to traditional approach are presented [53], [4]. Majority of these schemes use the XOR operation since it can be implemented efficiently in practice. Our work extends the analysis and performance characterization of NC using a general finite field. It is motivated by the well-developed theory of linear network code [62], [57] and the robustness of applying random linear network coding into multicast application [49], [50].

3.3 Problem Formulation

3.3.1 Problem Description and Notation

Consider the following broadcast scenario with one sender who wants to broadcast two packets $\mathbf{p}_1$ and $\mathbf{p}_2$ to two receivers R_1 and R_2. We assume R_1 has packet $\mathbf{p}_1$ while R_2 has packet $\mathbf{p}_2$. The goal is to minimize the number of broadcasts by the sender so that each receiver will have both packets $\mathbf{p}_1$ and $\mathbf{p}_2$, hence their data is synchronized. A straightforward way is for the sender to broadcast $\mathbf{p}_1$ first then $\mathbf{p}_2$. Assuming no packet loss, R_1 and R_2 will have both packets in two time slots. However, a better way is for the sender to broadcast only one packet $\mathbf{c} = \mathbf{p}_1 \oplus \mathbf{p}_2$ where $\oplus$ denotes bit-wise exclusive OR of bits in the two packets. Upon receiving $\mathbf{c}$, R_1 and R_2 will be able to recover their missing packets, respectively as: $\mathbf{c} \oplus \mathbf{p}_1 = \mathbf{p}_1 \oplus \mathbf{p}_2 \oplus \mathbf{p}_1 = \mathbf{p}_2$, and $\mathbf{c} \oplus \mathbf{p}_2 = \mathbf{p}_1 \oplus \mathbf{p}_2 \oplus \mathbf{p}_2 = \mathbf{p}_1$. This example illustrates the benefit of network coding, i.e., mixing packets appropriately to reduce the number of broadcasts. In general, the problem of finding a broadcast scheme, i.e., the right "coded" packets that minimizes the number of transmissions for an arbitrary number of users with an arbitrary pattern of packets is an NP-hard problem [29]. As such, we consider a probabilistic approach to this problem as described shortly. We now use the following notations to describe the problem.

- There is one sender with a set of D original packets denoted as $\mathcal{P} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_D\}$, and N receivers denoted as $R_1, R_2, \ldots R_N$ that want to obtain these D packets.

- Each receiver R_i has a "Has" set $\mathcal{H}_i$ consisting of exactly $K \leq D$ packets. Denote $\mathcal{W}_i = \mathcal{P} \setminus \mathcal{H}_i$ as the "Want" set of packets that the receiver R_i wants but does not

have.

- A network coded (mixed) packet $\mathbf{c}$ is constructed as:

$$\mathbf{c} = v_1\mathbf{p}_1 + v_2\mathbf{p}_2 + \cdots + v_D\mathbf{p}_D \tag{3.1}$$

  with $v_i \in GF(\mathcal{F})$. Each $\mathbf{p}_i$ can be viewed as an element in $GF(\mathcal{F}^D)$. Consequently, we can view a packet as a row vector $\mathbf{v} = (v_1, v_2, \ldots, v_D)$, and the "Has" set $\mathcal{H}_i$ as a matrix $\mathbf{H}_i$ whose rows are $\mathbf{v}$'s. Also, for brevity, we denote $F = |\mathcal{F}|$.

- At each time slot, the sender is allowed to broadcast exactly one mixed or original packet to all the receivers. Furthermore, we assume no packet loss during broadcast.

- Let T_i denote the number of time slots until the receiver R_i receives a sufficient number of packets to be able to reconstruct all D original packets.

- Let $T = \max\{T_1, T_2, \ldots T_N\}$ denote the number of time slots until all the receivers are able to decode all the D original packets.

Note that a receiver will be able to reconstruct all the D original packets if it collects any D packets (mixed or original) that span a D dimensional space. Specifically, recall that a packet can be represented as a row vector $\mathbf{v}_i$, then if the matrix

$$\mathbf{V} = \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ \vdots \\ \mathbf{v}_D \end{pmatrix}$$

has rank D (full rank), then the original packets $\mathbf{p}_i$'s can be reconstructed via solving a set of linearly independent equations.

For simplicity, the packet length as defined above is artificially constrained to length $\lceil D \log F \rceil$ bits. In practice, a packet should be a vector of length $n \gg D$ whose each element is $\lceil D \log F \rceil$ bits long. Thus the number of bits to specify v_i in the packet header (necessary for the receivers to decode) is negligible. Finally, we note that the optimal scheme is the one that minimizes T.

3.3.2 Example

We now give an example to illustrate the notations and concepts. Let $D = 4, K = 2, \mathcal{F} = \{0, 1\}$, and thus $GF(2)$ is used for all the finite field computations. A receiver R_1 has $\mathcal{H}_1 = \{\mathbf{p}_1 \oplus \mathbf{p}_3, \mathbf{p}_2\}$, and its initial "Has" set $\mathcal{H}_1$ can be represented as a matrix:

$$\mathbf{H}_1 = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Note that since $GF(2)$ is used, each entry in the matrix can only be 0 or 1. If the sender broadcasts two packets $(\mathbf{p}_2 \oplus \mathbf{p}_4)$ and $\mathbf{p}_3$ which collectively can be represented as a matrix $\mathbf{S}$ below.

$$\mathbf{S} = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Assuming no packet loss, then the new "Has" set $\hat{\mathcal{H}}_1$ of receiver R_1 would have two more elements. Thus the corresponding new matrix $\hat{\mathbf{H}}_1$ is:

$$\hat{\mathbf{H}}_1 = \begin{pmatrix} \mathbf{H_1} \\ \mathbf{S} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Since the $rank(\mathbf{H}_1) = 4$ (full rank) in $GF(2^4)$, R_1 can reconstruct all original packets $\{\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4\}$.

As described, an optimal broadcast scheme is one with the minimum number of transmissions that enables all the receivers to obtain their corresponding full rank matrices. Clearly, the minimum number of transmissions depends on the initial "Has" sets of each receivers. In the next section, we will describe two models of the "Has" sets that arise naturally from real-world settings. We then use these models to characterize the optimality of the solutions for any broadcast scheme via probabilistic bounds in Section 3.5.

3.4 Models of the "Has" Set

We consider two models for the "Has" sets at individual receivers. We call these the "uncoded" and "coded" models of the "Has" set. These models aim to approximate the real-world scenarios.

Uncoded Model. The first model can be used to approximate a wireless broadcast scenarios in WiFi or cellular networks. Specifically, in the "Uncoded" model, each individual receivers has K_i original packets out of the D original packets $\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_D$, where K_i is random variable drawn from the Binomial distribution with parameters (D, α_i). This model arises from considering a scenario in which a sender broadcast D original packets to N receivers. Due to different channel conditions, each receiver has a different probability α_i of receiving the packets. Assuming that the outcomes of the D transmissions are independent across packets and receivers, then the number of packets received at the receivers follow D independent Binomial distributions. Starting at this point, the sender can employ an optimal transmission scheme that ensure all N receivers can receive all the D original packets in minimum number of transmissions. The optimal solution depends on the pattern of packets at the receivers, i.e., their "Has" sets. While finding the optimal scheme is NP-hard, given the probability model of the "Has" sets, it is possible to characterize the optimal solution, i.e., the minimum number of transmissions via probabilistic bounds as will be shown in Section 3.5. These bounds are useful in the sense that one can bound the optimal solution without knowing the optimal transmission scheme. Furthermore, in some cases, it is possible to determine whether network coding scheme is even useful.

Coded Model. In the "coded" model, each receiver is to assume to have S packets. However, these packets are network coded packets, defined previously as:

$$\mathbf{c} = v_1\mathbf{p}_1 + v_2\mathbf{p}_2 + \cdots + v_D\mathbf{p}_D.$$

Each coded packet is drawn randomly at uniform from $F^D - 1$ possible coded packets independently without replacement. The "Coded" model can be used to represent data stored Peer-to-Peer (P2P) network. In this setting, a file is first broken into D packets, then a number of coded packets are produced using coefficients v_i drawn uniformly at random. These coded packets are then distributed to the peers via some P2P transmission protocols. Each peer can also mix the packets it receives and forwards the mixed

(coded) packets to another peers. As a result, the S packets stored at a peer can be thought as S coded packets drawn randomly at uniform from the $F^D - 1$ possible coded packets.

3.5 Optimality Characterization of "Has" Set Models

In this section, we first discuss the trivial bound on the minimum number of transmissions T^* needed for the N receivers to recover all D original packets with each receiver R_i having its "Has" set $\mathcal{H}_i$. We then derive the probability distribution of T^* when packets in the set $\mathcal{H}_i$ are drawn according to the "Uncoded" and "Coded" models described in the previous section. In some cases, it is sufficient and simple to use the probability bounds, rather than a full distribution to characterize the optimality. We will provide these probabilistic bounds as well.

3.5.1 Trivial Bound

The trivial bound does not assume a probability model on the "Has" set. Instead, supposed there are N receivers $R_1, R_2, \ldots R_N$, each has a number of packets, i.e., "Has" set $\mathcal{H}_i$ which can be represented as a matrix $\mathbf{H}_i$. Let K_i be the rank of $\mathbf{H}_i$, and let $K = \min\{K_i\}$. Then we have the following Proposition on the minimum number of transmissions T^* needed for the N receivers to recover all D original packets:

Proposition 1.

$$D - K \leq T^* \leq D \tag{3.2}$$

Denote the lower bound of T^*: $T_l^* = D - K$, then the lower bound T_l^* can be established quite easily by considering the fact in order for a receiver whose matrix $\mathbf{H}_i$ with rank K_i to reconstruct all D packets, it needs to receive additional packets or rows sent by the sender (the matrix $\mathbf{S}$ in the example) such that $\begin{bmatrix} \mathbf{H}_i \\ \mathbf{S} \end{bmatrix}$ is full rank (rank D). Therefore, the minimum number of transmissions needed to be at least larger or equal $D-K$ that allows the receiver R_i with the lowest rank matrix $\mathbf{H}_i$ to recover the original packets. The upper bound $T_u^* = D$ is obvious since the sender can just send D original packets and every receiver can receive D original packets since by assumption there is no packet loss.

The trivial bound, however does not take the advantage of probabilistic models, thus can be quite loose. Next, we characterize the full probability distributions of T_l^* and give probabilistic bounds on T_l^*. Notably, we use these bounds to determine the effectiveness of any network coding scheme in the "Uncoded" model.

3.5.2 Analysis of the "Uncoded" Model

We will first determine the distribution of K, then the distribution of T_l^* can be completely characterized. However the closed-form distribution is a bit complicated that prevents us from drawing a good intuition. Therefore, we also provide probabilistic bounds for K that allows us to draw a better intuition.

3.5.2.1 Computing Distribution of K

To derive the distribution, we note that K_i is a Binomial random variable with D being the number of trials and α_i the probability of success. Thus, we have:

$$\begin{cases} Rank(\mathbf{H}_i) = K_i \\ \mathbf{P}(K_i = k) = f(k, D, \alpha_i) = \binom{D}{k}\alpha_i^k(1-\alpha_i)^{D-k} \\ \mathbf{P}(K_i \leq k) = F(k, D, \alpha_i) = \sum_{j=0}^{k}\binom{D}{k}\alpha_i^j(1-\alpha_i)^{D-j} \end{cases} \tag{3.3}$$

Now since $K = \min\{K_i\}$, one can find the cumulative probability distribution of K as follows.

$$F_K(k) = \mathbf{P}(K \leq k) = 1 - \prod_i^N (1 - \mathbf{P}(K_i \leq k)) \tag{3.4}$$

Then the probability distribution of K can be computed from the cumulative function:

$$\begin{aligned}
\mathbf{P}(K=k) &= F_K(k) - F_K(k-1) && (3.5)\\
&= \prod_{i=1}^{N}(1-P(K_i \le k-1)) - \prod_{i=1}^{N}(1-P(K_i \le k))\\
&= \prod_{i=1}^{N}\left(1-\sum_{j=0}^{k-1}\binom{D}{j}\alpha_i^j(1-\alpha_i)^{D-j}\right)\\
&\quad -\prod_{i=1}^{N}\left(1-\sum_{j=0}^{k}\binom{D}{j}\alpha_i^j(1-\alpha_i)^{D-j}\right) && (3.6)
\end{aligned}$$

We can see that the closed-form distribution does not provide a good intuition. Hence, we now provide some probabilistic bounds regarding K.

3.5.2.2 Probabilistic Bounds for K

Let $\alpha_{min} = \min\{\alpha_i\}$ and $\alpha_{max} = \max\{\alpha_i\}$. We have following Proposition regarding the tail bound for K.

Proposition 2. *(Tail bound) For $0 < k < D\alpha_{min}$, we have*

$$\boldsymbol{P}(K > k) \ge \left(1 - \exp\left(-\frac{1}{2\alpha_{min}}\frac{(D\alpha_{min}-k)^2}{D}\right)\right)^N \quad (3.7)$$

Proof. We have:

$$\begin{aligned}
\mathbf{P}(K>k) &= \prod_{i=1}^{N}\mathbf{P}(K_i>k) && (3.8)\\
&= \prod_{i=1}^{N}(1-\mathbf{P}(K_i \le k)) && (3.9)\\
&= \prod_{i=1}^{N}(1-F(k,D,\alpha_i)) && (3.10)
\end{aligned}$$

Also, $F(k, D, \alpha_{min}) \geq F(k, D, \alpha_i)$. Hence,

$$\mathbf{P}(K > k) = (1 - F(k, D, \alpha_i))^N \geq (1 - F(k, D, \alpha_{min}))^N \tag{3.11}$$

Also by Chernoff's inequality, we have:

$$F(k, D, \alpha_{min}) \leq \exp\left(-\frac{1}{2\alpha_{min}} \frac{(D\alpha_{min} - k)^2}{D}\right)$$

Plug in (3.11), we complete the proof. □

Since $T_l^* = D - K$, Proposition 2 indicates that minimum number of retransmission for the "Uncoded" model depends on the receiver with the smallest probability of successful packet reception.

Next, we have the following proposition regarding the asymptotic behavior of D and N.

Proposition 3. *(Asymptotic) For $N \to \infty$ and any k, α_{min} such that $0 < k < D\alpha_{min}$, we have:*

$$\begin{cases} \boldsymbol{P}(K > k) \to 0 \text{ for } D = o(\log(N)) \\ \boldsymbol{P}(K > k) \to c \text{ where } c \in (0, 1) \text{ for } D = \Theta(\log(N)) \\ \boldsymbol{P}(K > k) \to 1 \text{ for } D = \omega(\log(N)) \end{cases} \tag{3.12}$$

(Using Bachmann-Landau notations for $o(), \Theta(), \omega()$).

Proof. We first show the case when $D = \Theta(\log(N))$.

$$\begin{aligned} \exp\left(-\frac{1}{2\alpha_{min}} \frac{(D\alpha_{min} - k)^2}{D}\right) &= exp\left(-\Theta(\log(N))\right. \\ &= \Theta\left(\frac{1}{N}\right) \leq c_1 \frac{1}{N} \end{aligned}$$

for some $0 < c_1 < \infty$. Hence,

$$\left(1 - \exp\left(-\frac{1}{2\alpha_{min}} \frac{(D\alpha_{min} - k)^2}{D}\right)\right)^N \geq \left(1 - \frac{c_1}{N}\right)^N. \tag{3.13}$$

Now,

$$\lim_{N\to\infty}\left(1-\frac{c_1}{N}\right)^N = e^{-c_1}, \tag{3.14}$$

and from (3.7), (3.13), and (3.14), when $N \to \infty$, we obtain

$$\mathbf{P}(K > k) \geq e^{-c_1} > 0. \tag{3.15}$$

We note that in (3.15), $\mathbf{P}(K > k)$ is strictly greater than 0.

On the other hand, using $\binom{D}{l} \geq 1$, we have

$$\begin{aligned} F(k, D, \alpha_{max}) &= \sum_{l=0}^{k} \binom{D}{l} \alpha_{max}^l (1-\alpha_{max})^{D-l} && (3.16)\\ &\geq \sum_{l=0}^{k} \alpha_{max}^l (1-\alpha_{max})^{D-l} && (3.17)\\ &= (1-\alpha_{max})^D \sum_{l=0}^{k} \left(\frac{\alpha_{max}}{1-\alpha_{max}}\right)^l && (3.18)\\ &\geq (1-\alpha_{max})^D. && (3.19) \end{aligned}$$

Since $D = \Theta(\log(N))$, and $0 < 1 - \alpha_{max} < 1$, we have

$$(1-\alpha_{max})^D = (1-\alpha_{max})^{\Theta(\log(N))} = \Theta(\frac{1}{N}).$$

Hence, $F(k, D, \alpha_{max}) \geq c_2(\frac{1}{N})$ for some $\infty > c_2 > 0$. Therefore,

$$(1 - F(k, D, \alpha_{max}))^N \leq (1 - \frac{c_2}{N})^N = e^{-c_2} < 1$$

for $N \to \infty$. Similar to (3.11), we have:

$$\mathbf{P}(K > k) = (1 - F(k, D, \alpha_i))^N \leq (1 - F(k, D, \alpha_{max}))^N \tag{3.20}$$

Combine these two above equations, we have:

$$\mathbf{P}(K > k) < 1. \tag{3.21}$$

Now, from (3.15) and (3.21), we have $0 < \mathbf{P}(K > k) < 1$. This completes the proof for $D = \Theta(\log(N))$.

For the case $D = \omega(\log(N))$, similarly we have:

$$\begin{aligned} \exp\left(-\frac{1}{2\alpha_{min}}\frac{(D\alpha_{min}-k)^2}{D}\right) &= \exp(-\omega(\log(N)) \\ &= \omega(\frac{1}{N}) \leq c_3\frac{1}{N}. \end{aligned}$$

for any $\infty > c_3 > 0$. Now,

$$\begin{aligned} \mathbf{P}(K > k) &\geq (1 - \exp\,(-\frac{1}{2\alpha_{min}}\frac{(D\alpha_{min}-k)^2}{D}))^N \\ &\geq (1 - \frac{c_3}{N})^N \to e^{-c_3} \to 1 \end{aligned} \tag{3.22}$$

for $N \to \infty$ and $c_3 \to 0$.

Also $\mathbf{P}(K > k) \leq 1$, then $\mathbf{P}(K > k) \to 1$ for $D = \omega(\log(N))$.

Finally, for $D = o(\log(N))$, similarly we have

$$\begin{aligned} F(k, D, \alpha_{max}) &\geq (1 - \alpha_{max})^D = (1 - \alpha_{max})^{o(\log(N))} \\ &= o(\frac{1}{N}) \geq c_4\frac{1}{N} \end{aligned}$$

for any $\infty > c_4 > 0$. Hence,

$$\mathbf{P}(K > k) \leq (1 - F(k, D, \alpha_{max}))^N \leq (1 - \frac{c_4}{N})^N \to e^{-c_4} \to 0 \tag{3.23}$$

for $N \to \infty$ and $c_4 \to \infty$.

Also $\mathbf{P}(K > k) \geq 0$ then $\mathbf{P}(K > k) \to 0$ for $D = o(\log(N))$. □

Using the parameters $\alpha_{min} = 0.3; \alpha_{max} = 0.7; k = \frac{D\alpha_{min}}{2}$, Fig. 3.1 shows the empirical probability $\mathbf{P}(K > k)$ that is accurately predicted by Proposition 3.

There is an interesting point implied by Proposition 3. If the number of packets sent (D) is on the order of log of the number of receivers (N), then the probability $\mathbf{P}(K > k)$ does not approach 0 or 1 when N and D approach infinity. Rather, this probability approaches a number between 0 and 1. On other the hand, probability $\mathbf{P}(K > k)$ approaches 0 or 1 when the $D = o(\log(N))$ and $D = \omega(\log(N))$, respectively.

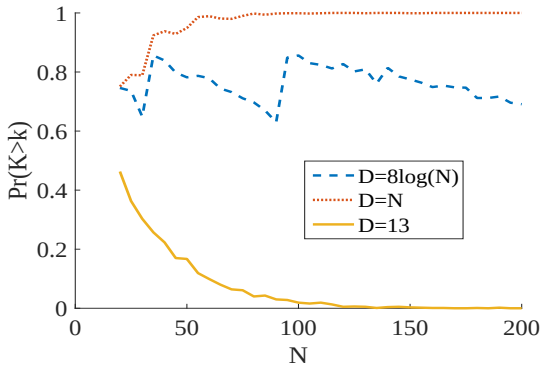


Figure 3.1: Empirical $\mathbf{P}[K > k]$ vs. N

Essentially, this implies that there is a phase transition that depends on on how large D is, compared with N in the asymptotic sense.

Consider the special case where $k = 0$, we have:

$$\begin{aligned}\mathbf{P}(K=0) &= 1-\mathbf{P}(K>0)\\ &= 1-\prod_{i=1}^{N}(1-F(0,D,\alpha_i)\\ &\geq 1-(1-(1-\alpha_{max})^D)^N.\end{aligned}$$

From the above equation, we have the following corollary:

Corollary 4. *For fixed the number of packets D, and the number of receivers $N \geq \log_{[1-(1-\alpha_{max})^D]}\epsilon$,*

$$\mathbf{P}(K=0) \geq 1-\epsilon$$

where $\epsilon > 0$.

The corollary above can be interpreted as follows. When the number of receivers is sufficiently large, there exists a receiver which hasn't received any packet with almost certainty. Therefore, the senders needs to re-send all packets. Also in this scenario, the lower bound equals the upper bound $T_l^* = D - K = D = T_u^*$ which implies that network coding does not bring any benefit.

3.5.3 Analysis of the "Coded" Model

In the "coded" model, each receiver stores S vectors and each vectors would be drawn randomly in $GF(F^D)$ (including both original and combined packets). First, we need to find the distribution of $K_i = rank(\mathbf{H}_i)$ and then one can compute the distribution of K by using order statistics. Still, the formula is too complex. Hence, we also provide upper bound for expectation of K_i. By these bounds, one can establish bound for K by Markov's inequality.

3.5.3.1 Computing Distribution of K

Consider any receiver R_i, we choose randomly S vectors in $GF(F^D)$ and there are $rank(\mathbf{H}_i) = K_i$ linearly independent vectors. We can compute $\mathbf{P}(K_i = k)$ by a recursive approach as follows.

Consider any node i, let $f(k, s)$ be the probability that $S = s$ and $rank(\mathbf{H}_i) = K_i = k$. Obviously, we can have (for simple cases):

$$\begin{cases} f(0,0) = 1 \\ f(k,s) = \prod_{j=1}^{j=k} p_j \text{ for } 1 \leq k = s \leq S \\ f(k,s) = 0 \text{ for } k > s. \end{cases} \tag{3.24}$$

The probability that we have k linearly independent vectors after picking up s random vectors is equal to sum of two probability: first is the probability that we have $k-1$ linearly independent vectors in $s-1$ random vectors and the s-th vector is linearly independent with existing vectors; second is the probability that we have k linearly independent vectors in $s-1$ random vectors and the s-th vector is dependent with existing vectors.

We have the inductive step for $0 < k < s \leq S$ as follows.

$$f(k,s) = f(k-1, s-1)p_k + f(k, s-1)(1 - p_{k+1}) \tag{3.25}$$

where

$$p_j = 1 - \frac{F^{j-1} - 1}{F^D - 1} = \frac{F^D - F^{j-1}}{F^D - 1} \tag{3.26}$$

We can rewrite the function $f(k,s)$ as follows. In case $s < k$, $f(k,s) = 0$. In case $s \geq k$, we have

$$f(k,s) = \sum_{\sum_j^{k+1} \alpha_j = s-k} \left(\prod_{i=1}^{k} p_i \prod_{j=2}^{k+1} (1-p_j)^{\alpha_j} \right) \tag{3.27}$$

$$= \prod_{i=1}^{k} p_i \sum_{\sum_j^{k+1} \alpha_j = s-k} \left(\prod_{j=2}^{k+1} (1-p_j)^{\alpha_j} \right) \tag{3.28}$$

where $\alpha_j = 0, 1, \ldots, s-k$ for $2 \leq j \leq k+1$. From the above formula, one can apply order statistics for i.i.d discrete variables [28] to compute the distribution of K.

3.5.3.2 Probabilistic Bounds on K

Proposition 5. *(Tail bound) For $S \leq D$ and any $k > 0$, we have:*

$$\boldsymbol{P}(K_i \geq k) \leq \frac{(\beta^S - 1)}{(\beta - 1)} \frac{1}{k} \tag{3.29}$$

where $\beta = \frac{F^D - F}{F^D - 1}$

Proof. Let denote $E_j = \mathbf{E}[K_i | S = j]$ be the expected rank of matrix $\mathbf{H}_i$ given that $\mathbf{H}_i$ has j rows (or the number of independent packets). Let q_j be the probability that the j-th row is linearly independent with the previous $j-1$ rows.

$$\begin{aligned} E_j &= q_j(E_{j-1} + 1) + (1 - q_j)E_{j-1} \\ &= q_j + E_{j-1} \\ &= q_j + q_{j-1} + E_{j-2} \\ &\cdots \\ &= \sum_{j=1}^{i} q_j \end{aligned}$$

since $E_0 = 0$.

Now, in each receiver R_i, we have S packets. Hence,

$$E_S = \sum_{j=1}^{S} q_j \tag{3.30}$$

Now, consider q_j. The necessary condition for j-th row to be linearly independent with previous $j-1$ rows is that j-th row needs to be linearly independent with each row in $j-1$ rows.

$$q_j \leq \left(\frac{F^D - F}{F^D - 1}\right)^{j-1} \tag{3.31}$$

for $j \geq 1$.

Combine (3.30) and (3.31), we have:

$$\sum_{j=1}^{S} \left(\frac{F^D - F}{F^D - 1}\right)^{j-1} \geq E_S = \mathbf{E}[K_i] \tag{3.32}$$

Hence, we can have an uppper bound U for $\mathbf{E}[K_i]$:

$$U = \sum_{j=1}^{S} \left(\frac{F^D - F}{F^D - 1}\right)^{j-1} = \frac{\beta^S - 1}{\beta - 1}$$

where $\beta = \frac{F^D - F}{F^D - 1}$.

One now can use Markov's inequality to complete the proof. □

Using the parameters $S = 3; F = 2; k = 2$, Fig. 3.2 shows the empirical probability $\mathbf{P}(K_i > k)$ and the upper bound for different values of D that match the prediction of the Proposition 5.

Since $K = \min\{K_i\}$, we have: $P(K \geq k) \leq P(K_i \geq k)$. However, one can establish a tighter bound for K by applying the inequality for N independent random variables with identical mean and variance in [5].

Proposition 6. *(Asymptotic) Consider where $D \to \infty$ and $T = D - S$ is a constant,*

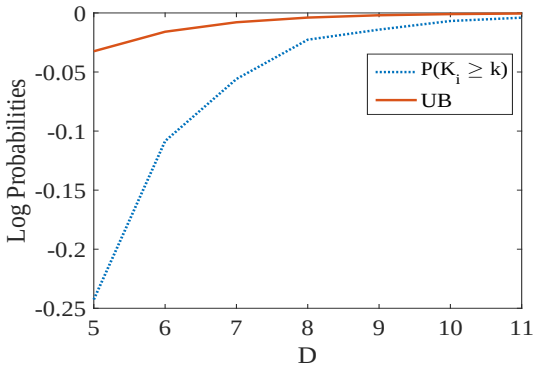


Figure 3.2: Empirical $\mathbf{P}[K > k]$ vs. D

using the result given in [22, Theorem 1], we have:

$$\lim_{D\to\infty} \boldsymbol{P}(K_i = k) = \begin{cases} \prod_{j=T+1}^{\infty} (1 - (\frac{1}{F})^j) \ \textit{for } k = 0 \\ \frac{\prod_{j=T+k+1}^{\infty} (1-(\frac{1}{F})^j)}{\prod_{j=1}^{k} (1-(\frac{1}{F})^j)} (\frac{1}{F})^{k(T+k)} \ \textit{for } k \geq 1 \end{cases}$$

Hence, one can compute the probability distribution of K by using order statistics.

3.6 Algorithms

In the previous section, we characterize the optimal solution via asymptotic and probabilistic results. In this section, we describe three random network coding algorithms to approximate the optimal solution: the Simple Random Network Coding Algorithm (SRNC), the Informed Random Network Coding Algorithm (IRNC), and the Refined Random Network Coding Algorithm (RRNC). We start with the simplest one: SRNC algorithm.

3.6.1 Simple Random Network Coding Algorithm (SRNC)

The SRNC algorithm is described as follows.

SRNC algorithm assumes that the sender has no knowledge about the subsets of packets at the receivers at any given time. At every time slot, the sender broadcasts a

Algorithm 1: SRNC Algorithm

```
   Data: The sender has no knowledge about packets at receivers
1  while there exists one receiver that can't recover the original packets do
2  |  Sender generates and broadcasts a mixed packet;
3  |  Each receiver R_i updates its "Has" set and corresponding matrix H_i;
4  |  if H_i is full rank then
5  |  |  R_i can recover the original packets and sends acknowledgment to the
   |  |   sender;
6  |  end
7  end
```

mixed packet (line 2)

$$\mathbf{c} = v_1\mathbf{p}_1 + v_2\mathbf{p}_2 + \cdots + v_D\mathbf{p}_D,$$

where v_i's are drawn uniformly at random from the finite field $GF(\mathcal{F})$. The sender will continue to broadcast these packets until it receives all the acknowledgments from each receiver, indicating that all the receivers have successfully obtained all the packets.

At the receiver, upon receiving a mixed packet $\mathbf{c}$, the "Has" set of a receiver R_i is updated as:

$$\mathcal{H}_i = \mathcal{H}_i \bigcup \{\mathbf{c}\},$$

and the corresponding matrix $\mathbf{H}_i$ is constructed (line 3). Next, the Gaussian elimination algorithm is applied to $\mathbf{H}_i$ to find linearly independent columns and the missing original packets. If $\mathbf{H}_i$ is full rank, then receiver R_i can recover the original packets. In this case, the receiver sends an acknowledgment to the sender indicating that it has successfully recovered all the original packets (line 5). Otherwise, it waits for the next packet from the sender. The process repeats until the receiver is able to recover all the original packets. The SRNC algorithm is simple since the sender does not require information from the receivers. Rather, only one acknowledgement from each receiver is sufficient to complete the synchronization process.

3.6.2 Informed Random Network Coding (IRNC)

The IRNC algorithm is described as follows.

The IRNC algorithm requires a bit more information. Specifically, all receivers send

Algorithm 2: IRNC Algorithm

```
   Data: The sender has knowledge about "Want" sets at receivers only in the
         beginning
1  while there exists one receiver that cannot recover the original packets do
2      Sender generates and broadcasts a mixed packet based on the initial "Want"
       sets at receivers;
3      Each receiver R_i updates its "Has" set and corresponding matrix H_i;
4      if H_i is full rank then
5          R_i can recover the original packets and sends an acknowledgment to the
           sender;
6      end
7  end
```

the information on their "Want" sets to the sender only once in the beginning. The sender uses this information to construct and broadcast the mixed packets without further collaboration from the receivers except the final acknowledgements from each receiver indicating that they have successfully obtained all the packets.

The "Want" set at each receiver R_i is constructed as follows. First, the Gaussian elimination algorithm is applied to $\mathbf{H}_i$ to find the missing original packets. Next, R_i sends this information to the sender. The sender then constructs a union set $\mathcal{W} = \bigcup_i \mathcal{W}_i$ where $\mathcal{W}_i$ consisting of the missing original packets for R_i. Let $M = |\mathcal{W}|$, the sender broadcasts a mixed packet constructed as:

$$\mathbf{c} = v_1\mathbf{p}_1 + v_2\mathbf{p}_2 + \cdots + v_M\mathbf{p}_M,$$

where $\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_M \in \mathcal{W}$, and v_i's are drawn uniformly at random from the finite field $GF(\mathcal{F})$. The only difference between IRNC and SRNC algorithms is that the IRNC algorithm generates mixed packets from $\mathcal{W}$ (line 2) while the SRNC algorithm generates mixed packets from all the original packets in $\mathcal{P}$.

As an example, consider a scenario with five original packets and two receivers R_1 and R_2. R_1 has three packets, each is a linear combination of the five original packets.

Thus, $\mathbf{H}_1$ is a 3×5 matrix of the form:

$$\mathbf{H}_1 = \begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{pmatrix},$$

where * denotes values from $GF(\mathcal{F})$. Assume that $\mathcal{F} = \{0, 1\}$, then each row in $\mathbf{H}_1$ represents a packet of R_1 which is a linear combination of the five original packets. Since $GF(2^5)$ is used, a 1 or a 0 in the i-th column and j-th row indicates that the original packet $\mathbf{p}_i$ is present or not in the j-th mixed packet, respectively. Now R_1 applies the Gaussian elimination algorithm, and suppose it produces the following upper diagonal matrix:

$$\mathbf{H}_1' = \begin{pmatrix} 1 & * & * & * & * \\ 0 & 0 & 1 & * & * \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

Based on $\mathbf{H}_1'$, the "Want" set of R_1 includes $\mathbf{p}_2$ and $\mathbf{p}_4$. R_1 then sends this information to the sender. Similarly, if R_2's "Want" set contains only $\mathbf{p}_3$, it will send this information to the sender. The sender will now generate the mixed packets that are random linear combinations from the set $\mathcal{W} = \{\mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4\}$.

Receivers in the IRNC algorithm also behaves similarly to those in the SRNC algorithm. Since the IRNC algorithm generate packets based on the missing packets at the receivers, the sender avoids sending redundant information to the receivers. Therefore, the IRNC algorithm should perform better than the SRNC algorithm.

3.6.3 Refined Random Network Coding Algorithm (RRNC)

We now introduce the RRNC algorithm which can be shown theoretically better than the SRNC and IRNC algorithms. The RRNC algorithm is described as follows.

Compare to the previous two algorithms, the RRNC algorithm requires a bit more information exchange between the sender and receivers, but they all are very similar. Specifically, the sender receives the information on the "Want" sets from each receiver after transmitting each packet. It then constructs the union set $\mathcal{W} = \bigcup_i \mathcal{W}_i$, and gener-

Algorithm 3: RRNC Algorithm

```
   Data: The sender has knowledge about "Want" sets at receivers at each time slot
1  while there exists one receiver that cannot recover all the original packets do
2  |  Sender generates and broadcasts a mixed packet based on the "Want" sets;
3  |  Each receiver R_i updates its "Has" set and corresponding matrix H_i;
4  |  if H_i is full rank then
5  |  |  R_i can recover the original packets and sends acknowledgment to the
   |  |   sender;
6  |  else
7  |  |  R_i computes and sends its "Want" set to the sender;
8  |  end
9  end
```

ates mixed packets based on $\mathcal{W}$ in the exact manner as the IRNC algorithm. The only difference is that after receiving a new packet, each receiver recomputes its "Want" set and sends its updated "Want" set to the sender (line 7). The sender then constructs a new $\mathcal{W}$ and uses it to generate and broadcast the next packet (line 2). The process repeats until all the receivers can successfully recover all the original packets.

Intuitively, the RRNC algorithm is better than the IRNC and SRNC algorithms because at each time slot, the RRNC algorithm uses more information about the missing packets at each receiver. As a result, a mixed packet generated by the RRNC algorithm has a higher chance of adding more new information to the receivers than the others two. We will show the theoretical analysis in the next section.

3.7 Theoretical Performance of the Proposed Algorithms

In this section, we provide a number of theoretical results on the performances for the proposed SRNC, IRNC, and RRNC algorithms in terms of the number of time slots for completing the data synchronization. First the performances of algorithms are considered from the viewpoint of a single receiver R_i. Recall in Section 3.3 that a packet can be represented as a vector $\mathbf{v}$. Thus, a group of packets are considered as mutually linearly independent if theirs vector representations are mutually linearly independent. We now consider a receiver R_i who wants to recover all $D = |\mathcal{P}|$ original packets. Given that R_i currently obtains $K \leq D$ linearly independent packets, we want to know on average how

many time slots it takes for R_i to recover all D original packets using the SRNC, IRNC, and RRNC algorithms.

3.7.1 Single User's Perspective

Let $T_i^{(S)}$, $T_i^{(I)}$, and $T_i^{(R)}$ be the random variables denoting the number of packets sent out by the sender, i.e., the number of time slots required so that R_i is able to recover all the original D packets using the SRNC, IRNC, and RRNC algorithms, respectively. Let denote $|\mathcal{F}| = F$ and also $L = D - K$ be the cardinality of the individual "Want" set for each receiver R_i. Then, we have the following Propositions to characterize the performances of the proposed algorithms.

Proposition 7. *(Performance of the SRNC algorithm)*

$$\mathbf{E}[T_i^{(S)}] = \sum_{j=1}^{L} \frac{F^D - 1}{F^D - F^{K+j-1}} \tag{3.33}$$

$$\mathbf{Var}[T_i^{(S)}] = \sum_{j=1}^{L} \frac{(F^{K+j-1} - 1)(F^D - 1)}{(F^D - F^{K+j-1})^2} \tag{3.34}$$

Let $M = |\mathcal{W}|$ be the cardinality of the combined "Want" set, then the performance of the IRNC algorithm is characterized by the following Proposition.

Proposition 8. *(Performance of the IRNC algorithm)*

$$\boldsymbol{E}[T_i^{(I)}] = \sum_{j=1}^{L} \frac{F^M - 1}{F^M - F^{M-L+j-1}} \tag{3.35}$$

$$\boldsymbol{Var}[T_i^{(I)}] = \sum_{j=1}^{L} \frac{(F^{M-L+j-1} - 1)(F^M - 1)}{(F^M - F^{M-L+j-1})^2} \tag{3.36}$$

Next, the following Proposition characterizes the performance of the RRNC algorithm.

Proposition 9. *(Performance of the RRNC algorithm)*

$$\boldsymbol{E}[T_i^{(R)}] \leq \sum_{j=1}^{L} \frac{F^{M-j+1}-1}{F^{M-j+1}-F^{M-L}} \tag{3.37}$$

$$\boldsymbol{Var}[T_i^{(C)}] \leq \sum_{j=1}^{L} \frac{(F^{M-j+1}-1)(F^{M-L}-1)}{(F^{M-j+1}-F^{M-L})^2}. \tag{3.38}$$

The proofs of all these Propositions can be found in the Appendix.

The following Proposition supports our intuition that the RRNC algorithm is better than the IRNC algorithm which in turn is better than the SRNC algorithm.

Proposition 10. *(Performance Comparison)*

$$\boldsymbol{E}[T_i^{(R)}] \leq \boldsymbol{E}[T_i^{(I)}] \leq \boldsymbol{E}[T_i^{(S)}] \tag{3.39}$$

$$\boldsymbol{Var}[T_i^{(R)}] \leq \boldsymbol{Var}[T_i^{(I)}] \leq \boldsymbol{Var}[T_i^{(S)}]. \tag{3.40}$$

Proof. For the expected value, let us consider the following function:

$$f(x) = \frac{F^x - 1}{F^x - F^{x-a}},$$

where $1 \leq a \leq L$ is a constant. We have:

$$f'(x) = \frac{lnF}{F^x - F^{x-a}} > 0.$$

where $x > a$. Therefore, $f(x)$ is a monotonically increasing function in x.

Now, from (3.33), (3.35), (3.37) the upper bound of $\mathbf{E}[T_i^{(R)}]$, $\mathbf{E}[T_i^{(I)}]$ and $\mathbf{E}[T_i^{(S)}]$ is the sum of functions of the form $f(M-j+1)$, $f(M)$ and $f(D)$, respectively. Also, $M-j+1 \leq M \leq D$. Thus, we have $\mathbf{E}[T_i^{(R)}] \leq \mathbf{E}[T_i^{(I)}] \leq \mathbf{E}[T_i^{(S)}]$.

For the variance, consider the following function:

$$g(x) = \frac{(F^x-1)(F^{x-a}-1)}{F^x - F^{x-a}} \tag{3.41}$$

where $1 \leq a \leq L$ is a constant. We have

$$g'(x) = \frac{\ln(F)(F^{2x} - F^a)}{F^x(F^a - 1)} > 0 \tag{3.42}$$

where $x > a$. Hence, $g(x)$ is a monotonically increasing function in x.

Now, from (3.34), (3.36), (3.38) the upper bound of $\mathbf{Var}[T_i^{(R)}]$, $\mathbf{Var}[T_i^{(I)}]$, and $\mathbf{Var}[T_i^{(S)}]$ is the sum of functions of the form $g(M-j+1)$, $g(M)$ and $g(D)$, respectively. Also, $M - j + 1 \leq M \leq M$. Thus, we have $\mathbf{Var}[T_i^{(R)}] \leq \mathbf{Var}[T_i^{(I)}] \leq \mathbf{Var}[T_i^{(S)}]$.

□

3.7.2 Sender's Perspective

We now consider the performance of the entire system, i.e., the sender's perspective. Let $T_{max}^{(S)}$, $T_{max}^{(I)}$, and $T_{max}^{(R)}$ be the random variables denoting the numbers of time slots until the sender receives all the acknowledgments from all N receivers using the SRNC, IRNC, and RRNC algorithms, respectively. Then clearly,

$$T_{max}^{(S)} = \max_i T_i^{(S)} \tag{3.43}$$

$$T_{max}^{(I)} = \max_i T_i^{(I)} \tag{3.44}$$

$$T_{max}^{(R)} = \max_i T_i^{(R)}, \tag{3.45}$$

for $i = 1, 2, \ldots, N$.

The performances of all three algorithms are characterized by the following Proposition.

Proposition 11. *(Tail probability)*

$$\mathbf{P}(T_{max} > a) \leq 1 - (1 - \frac{\sigma^2}{(a-\mu)^2})^N \tag{3.46}$$

for $a > \mu = \mathbf{E}[T_i]$ *and* $\sigma^2 = \mathbf{Var}[T_i]$ *for each algorithm, respectively.*

Alternatively, one can find the upper bound of $\mathbf{E}[T_{max}]$ by applying the inequality

for N independent random variables with identical mean and variance in [5] as follows.

$$\mathbf{E}[T_{max}] \leq \mu + \sigma\sqrt{N-1}. \tag{3.47}$$

3.8 Performance Results

In this section, we present the performance evaluations of the proposed algorithms for various settings, and verify the agreement between the theoretical and empirical results.

Fig. 3.3 shows the empirical $\mathbf{E}[T_i^{(S)}]$, $\mathbf{E}[T_i^{(I)}]$, $\mathbf{E}[T_i^{(R)}]$, i.e., the average numbers of time slots needed for a receiver to recover all $N = 50$ original packets using the SRNC, IRNC, and RRNC algorithms, respectively, as a function of K, the number of packets initially at a receiver. The value range for K is from 20 to 30. As seen, the value of $\mathbf{E}[T_i^{(S)}]$ and $\mathbf{E}[T_i^{(I)}]$ are not much different to each other, while $\mathbf{E}[T_i^{(R)}]$ is slightly smaller. This complies with our intuition since the RRNC algorithm has more information than the others two. Despite of a modest improvement in mean of time slots needed to recover all the original packets for the RRNC algorithm, we note that the variance of $T_i^{(R)}$ is also smaller than those of the others two. This is quite important as we consider the performance from the sender's perspective as shown in Fig. 3.4.

Fig. 3.4 shows empirical $\mathbf{E}[T_{max}^{(S)}]$, $\mathbf{E}[T_{max}^{(I)}]$, $\mathbf{E}[T_{max}^{(R)}]$ as the numbers of time slots needed for the sender to complete the synchronization process for the SRNC, IRNC, and RRNC algorithms. Now, one can see that the RRNC algorithm achieves a much better performance than those of the others two. We argue that this is due to smaller variance produced by the RRNC algorithm. This can be seen from Eq. (3.47) that $\mathbf{E}[T_{max}]$ for all three algorithms depends on the square root of the number of the receivers times the variance. Thus, a small change in variance can greatly affect $\mathbf{E}[T_{max}]$ for a large number of receivers.

Next, we verify our theoretical results with simulations for various parameters. Using $D = 100, F = 2, K = 30 \rightarrow 70$, Fig. 3.5 and Fig. 3.6 show the correctness of analytical performance of the SRNC algorithm. As seen, the number of time slots decreases while K increases. Intuitively, the more information a receiver has, the less information the sender needs to broadcast to complete the synchronization process at this receiver.

Using $N = 10, D = 10, F = 2, K = 5$, Fig. 3.7 and Fig. 3.8 verify the agreement between theoretical and simulated performance results of the IRNC algorithm as function

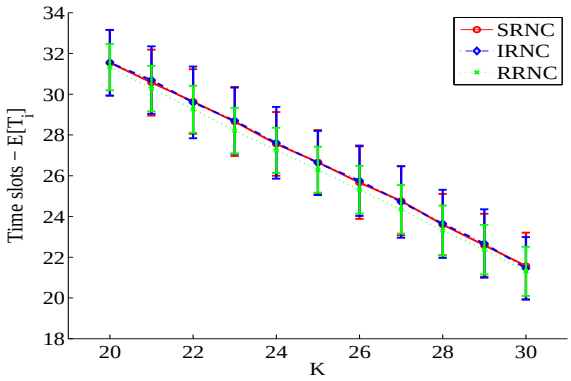


Figure 3.3: Empirical $\mathbf{E}[T_i]$ vs. K

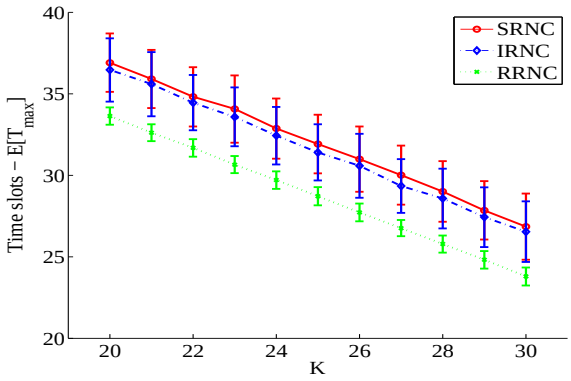


Figure 3.4: Empirical $\mathbf{E}[T_{max}]$ vs. K

of M (cardinality of the union set). As seen, the smaller cardinality of the union set is, the better performance can be achieved.

Using $N = 50, D = 30, F = 2, K = 10$, the validity of the upper bound on the expectation, and the variance of the RRNC algorithm are shown in Fig. 3.9 and Fig. 3.10.

Fig. 3.11 and Fig. 3.12 show the performance of three algorithms with different value of F (the field size) using $N = 30, D = 20, K = 10, F \in \{2, 3, 5, 7, 11, 13\}$. As seen, while F grows the performance of the proposed algorithms is improved substantially. Intuitively, with a larger field size the probability that a new generated packet is dependent with the packets in "Has" sets at receivers will decrease, leading to higher

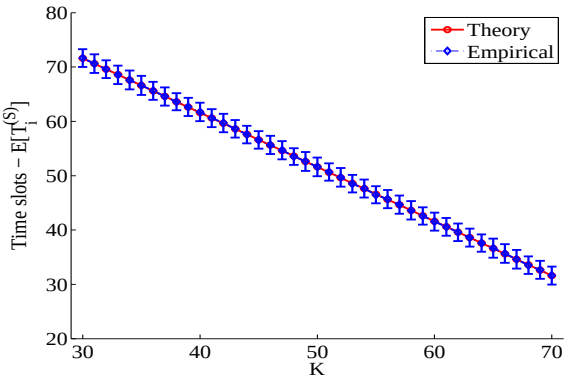


Figure 3.5: Theoretical and empirical performance $\mathbf{E}[T_i^{(S)}]$ vs. K for algorithm SRNC

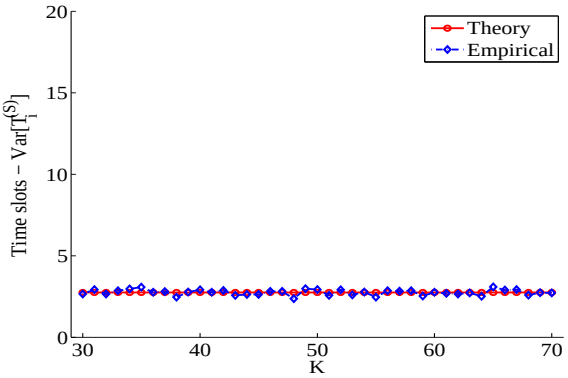


Figure 3.6: Theoretical and empirical performance $\mathbf{Var}[T_i^{(S)}]$ vs. K for algorithm SRNC

chance recovering the all original packets faster.

The robustness of random network coding techniques can be verified in Fig. 3.12. Here, we compare proposed algorithms with an efficient deterministic algorithm in which the sender only broadcasts M original packets in union set $\mathcal{W}$. Obviously, the number of time slots to complete synchronization process for the deterministic algorithm is M. It can be seen that the deterministic algorithm outperforms SRNC and IRNC for some small values of F, however from the range where $F \geq 7$, IRNC has better performance and the performance of SRNC: $\mathbf{E}[T_{max}^{(I)}]$ is very close to M.

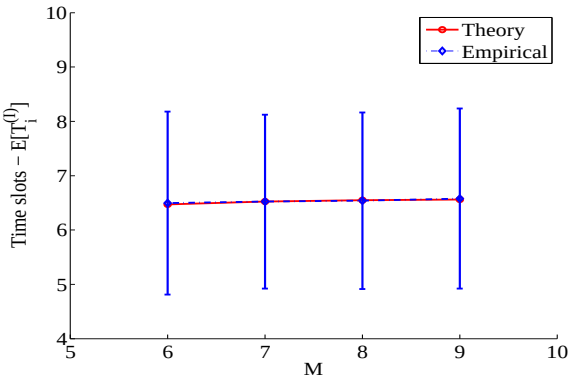


Figure 3.7: Theoretical and empirical performance $\mathbf{E}[T_i^{(I)}]$ vs. M for algorithm IRNC

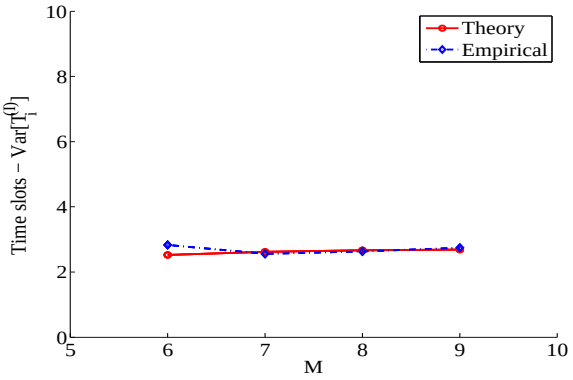


Figure 3.8: Theoretical and empirical performance $\mathbf{Var}[T_i^{(I)}]$ vs. M for algorithm IRNC

3.9 Conclusion

In this chapter, we describe the problem of efficient data synchronization/ broadcast for a large number of nodes with disparate data. The synchronization problem arises naturally in many applications, including Peer-to-Peer networks, data centers, and distributed storage systems with asynchronous updates. Two probabilistic models are considered on how the initial fractions of packets at receivers are distributed and according to different practical scenarios. Also, we propose and analyze a number of random network coding algorithms and verify their performances via theoretical analysis and simulations.

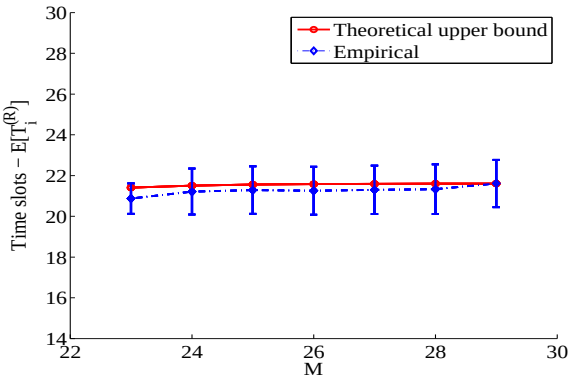


Figure 3.9: Theoretical upper bound and empirical performance $\mathbf{E}[T_i^{(R)}]$ of algorithm RRNC

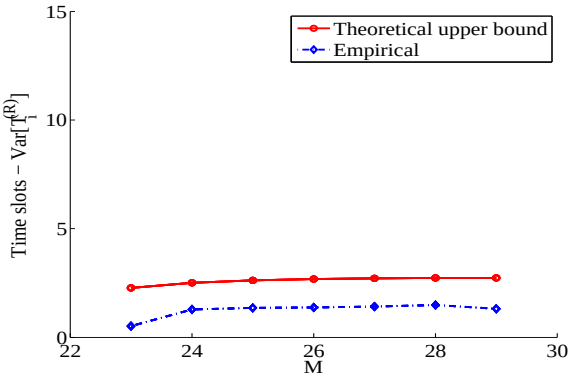


Figure 3.10: Theoretical upper bound and empirical performance $\mathbf{Var}[T_i^{(R)}]$ of algorithm RRNC

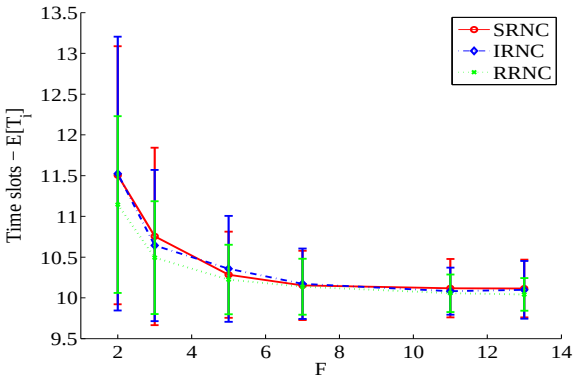


Figure 3.11: Empirical performance $\mathbf{E}[T_i]$ of receiver while increasing F

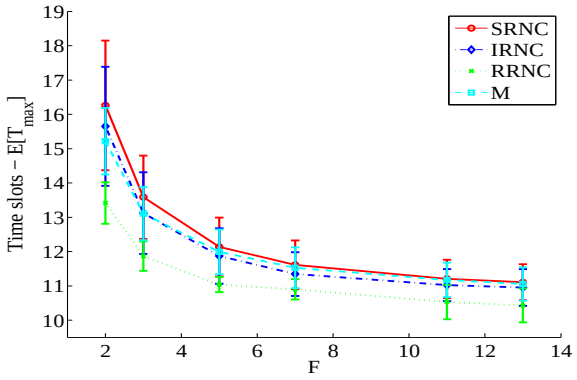


Figure 3.12: Empirical performance $\mathbf{E}[T_{max}]$ of sender while increasing F

Chapter 4: On Perturbation of Minimum Rank Matrices with Application to Matrix Recovery

4.1 Motivation

In recent years, there have been a lot of interests in studying network coding (NC), especially random network coding (RNC) [21], [65] in the context of network security. As discussed in the previous chapter, the RNC technique mixes a number of "packets" together, i.e., linearly combines a number of packets where the random coefficients are drawn uniformly from a finite field, to produce a networked coded packet. To decode the original information packets, the receivers need to know the random coefficients in order to find the solution (information packets) to the system of linear equations. These information is sent together with the packets in the header, or ahead of time.

Now, we consider an example of RNC setting in which sender nodes (R_1, R_2) send the RNC packets to receiver nodes (R_3) (illustrated in Fig. 4.1).

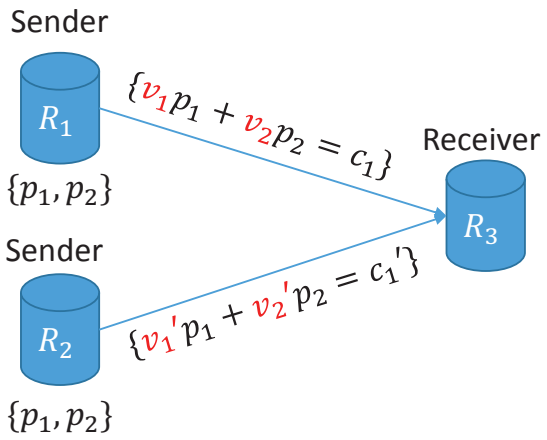


Figure 4.1: Random Network Coding example

When a node (e.g. R_1 or R_2 in Fig. 4.1) is compromised, a malicious attacker can change the random coefficients (v_1, v_2, v_1', v_2') in such a way that the linear system is no

longer full-rank:

$$rank\left(\begin{bmatrix} v_1 & v_2 \\ v_1' & v_2' \end{bmatrix}\right) < 2$$

and thus the original data cannot be recovered, even when the network coded packets are successfully transmitted.

In this chapter, we cast the problem of information recovery using NC technique under a malicious attack as the problem of minimum rank decoding of matrices over finite fields.

4.2 Related Work

The miminum rank decoding problem in our work is based on the theory of matrix rank which many research and applications have been built upon [74], [95]. In many engineering applications, the model or design of a system can be represented in a matrix form. The rank of this matrix can be used to express order, complexity, or dimension of the system [33].

For this reason, there exists rich literature on the applications of rank matrix. We also note that in most of the applications, rank of matrices are considered in the real/complex fields. Due to limited space, we only disscuss a few. One of the most well-known applications of calculating the rank of a matrix is to solve a system of linear equations [75], [48]. In the field of control theory, in order to determine whether a linear system is controllable, or observable, the rank of system matrix can be used [35], [84], [69]. Also, the rank of the communication matrix is an important factor in the field of communication complexity. It is used as a function gives bounds on the amount of communication needed for two parties [83], [73]. In graph theory, the rank of matrix has been used extensively as graph metrics, e.g., minimum rank of an undirected graph is the smallest rank of any generalized adjacency matrix of the graph [32], [96]. Rank of matrices has also been used in Coding Theory to define Rank Metric [85], [39] and then apply to several applications [80], [40].

Our work is also similar to the Minimum Rank Problem (MRP) which have been applied to Matrix Recovery/Completion research [14], [12]. The MRP has been known to be NP-hard [79]. In real fields, the problem can be solved approximately using log-det function or trace function of the matrix [34]. However, these heuristic approaches are

not as efficient when the problem is considered in Finite Fields.

That said, the theory of MRP in Finite Fields is far from completion. Motivated by the well-developed theory of MRP in real fields, our work extends the idea of using MRP in Matrix Recovery to Finite Fields and apply to the problem of Information/Matrix Recovery under malicious attack in NC-technique.

4.3 Min-rank Decoding Problem

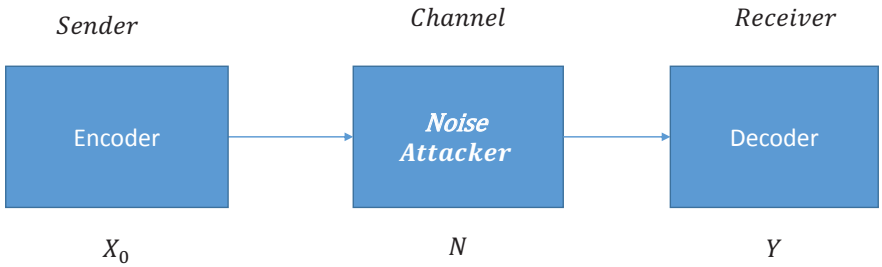


Figure 4.2: Channel model

The minimum rank decoding problem is illustrated in the Fig. 4.2. First, the sender sends a packet through a noisy channel to the receiver. The noise models the random changes that the attacker made to the packet header. Let X_0 be the matrix representing the header, N be the matrix representing noise, then a receiver receives a corrupted header as $Y = X_0 + N$. We study the conditions under which one can recover the rank of matrix X_0 or the original matrix X_0 with high probability.

We use the following notations.

- All matrices are considered in $F_q^{n\times n}$
- $\|.\|$ denotes the Hamming distance.
- X_0 denotes the original matrix (uncorrupted header).
- N denotes the limited magnitude noise, $\|N\| \leq \epsilon$.
- $Y = X_0 + N$ denotes the observed matrix (corrupted header).

Our goal is to recover $rank(X_0)$ or X_0 when possible.

4.3.1 Min-Rank Properties and Decoder

The minimum rank decoding problem is formulated as:

$$\begin{aligned} &\text{Minimize} && rank(X) \\ &\text{subject to} && \|X - Y\| \leq \epsilon \end{aligned} \tag{4.1}$$

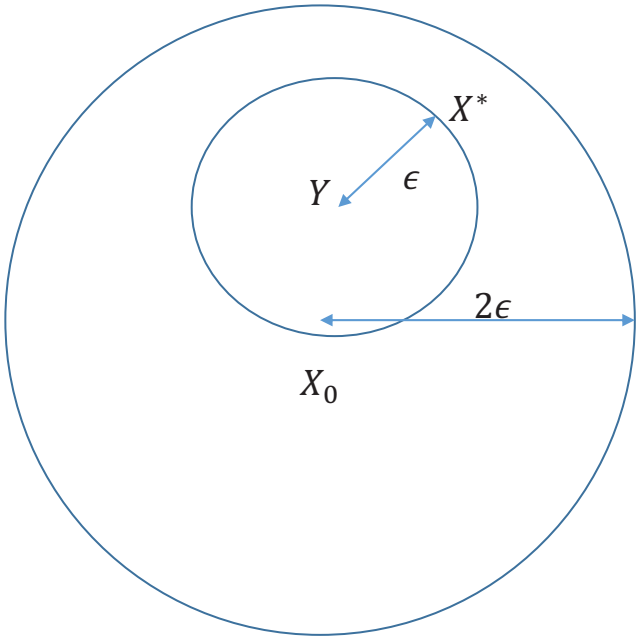


Figure 4.3: Min-rank decoding problem

Let X^* be the solution of (4.1) then we have

$$rank(X^*) \leq rank(X_0). \tag{4.2}$$

Definition 16. ***The min-rank property***

X_0 satisfies the min-rank property if and only if that for any X such that

$$||X - X_0|| \leq 2\epsilon$$

then

$$rank(X) \geq rank(X_0)$$

The strictly min-rank property is an extension of the min-rank property which is defined as follows.

Definition 17. ***The strictly min-rank property*** X_0 *satisfies the strictly min-rank property if and only if that for any* X *such that*

$$0 < ||X - X_0|| \leq 2\epsilon$$

then

$$rank(X) > rank(X_0)$$

We have the following Theorem related to the min-rank property.

Theorem 12. *The (strictly) min-rank property can lead to matrix (rank) recovery using the min-rank decoder in problem (4.1).*

Proof. Assume that X_0 satisfies the min-rank property then

$$||X^* - X_0|| \leq ||X^* - Y|| + ||Y - X_0|| = 2\epsilon$$

then we have

$$rank(X^*) \geq rank(X_0) \tag{4.3}$$

From (4.2) and (4.3), we can conclude that:

$$rank(X^*) = rank(X_0)$$

In addition, the min-rank property is illustrated in Fig. 4.3. Now if matrix X_0 satisfies the strictly min-rank property, which is:

$$rank(X_0) < rank(X) \ \forall \ X \neq X_0 : \|X - X_0\| \leq 2\epsilon$$

Then the solution matrix X^* is in the within 2ϵ (in Hamming distance) from matrix X_0 and has the same rank as X_0. Hence, X^* is exactly equal to X_0. We can conclude that the min-rank decoder in (4.1) can recover the rank of original matrix. □

Example 11. *Let*

$$X_1 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

We have $rank(X_1) = 1$ *then changing one or two entries of* X_1 *can only increase rank of* X_1*. Hence,* X_1 *satisfies the strictly min-rank property for* $\epsilon = 1$*.*

Also, let

$$X_2 = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$$

Then we have $rank(X_2) = 2$ *and rank of matrix* X_2 *can only be increased or remains the same when we change one or two entries of* X_2*. Hence,* X_2 *satisfies the min-rank property with* $\epsilon = 1$*.*

To exactly recover the original X_0, X^* must be unique. We also investigate these conditions in probabilistic settings.

4.3.2 Min-rank property in Uniform Model

Let us denote:

- $\mathcal{R}(r, n)$ is the set of of matrices $X \in F_q^{n \times n}$ such that $rank(X) = r$
- $\mathcal{M}(r, n, 2\epsilon)$ is the set of matrices $X \in \mathcal{R}(r, n)$ such that X satisfies min-rank property.

Now, we can establish a lower bound for the min-rank property in Uniform model as follows.

Theorem 13. *Suppose matrix* X *is drawn uniformly at random in the set of matrices* $\in \mathcal{R}(r, n, 2\epsilon)$ *with condition that* $r(2\epsilon + 1) \leq n$ *then we have:*

$$\boldsymbol{P}\left(X \in \mathcal{M}(r, n, 2\epsilon)\right) \geq \frac{A_q(r, r)^{(\epsilon'+1)} q^{r(n-r(\epsilon'+1))}}{A_q(n, r)}$$

where $\epsilon' = 2\epsilon$ *and*

$$A_q(n,k) = (q^n - 1)(q^n - q)\dots(q^n - q^{k-1})$$

is the number of ordered k*-tuples of linearly independent vectors in* F_q^n*.*

We also have the following asymptotic result where n is large.

Corollary 14. *Define* LB *as the lower bound in the theorem (13), we have:*

$$\lim_{n\to\infty} LB = \frac{A_q(r,r)^{(\epsilon'+1)}}{q^{r^2(\epsilon'+1)}}$$

which means that the lower bound depends only on r, ϵ *and* q*.*

Proofs of Theorem 13 and Corollary 14 can be found in Appendix.

4.3.3 Complexity of min-rank decoder

Definition 18. *Define the set* $\mathcal{S}_X(\epsilon) = \{Y : ||Y - X|| \leq \epsilon\}$*. (the norm here is Hamming distance). The complexity of min-rank decoder is defined as the size of the set* $\mathcal{S}_X(\epsilon)$*.*

Let $m = n^2$. The size of the set:

$$|\mathcal{S}_X(\epsilon)| = \sum_{i=0}^{\epsilon} \binom{m}{i}$$

Let $\epsilon = \lfloor \alpha m \rfloor$, and use bounds on the Binomial sum [37, pg 427], we have

$$\sum_{i=0}^{\epsilon} \binom{m}{i} \leq 2^{H(\alpha)m} = 2^{H(\frac{\epsilon}{m})m}$$

where entropy $H(\alpha) = -\alpha\log(\alpha) - (1-\alpha)\log(1-\alpha)$.

4.4 Random Matrix Model

4.4.1 Model Description

In this section, we introduce a random matrix model whose entries are i.i.d variables. Specifically, let X be a $n \times n$ random matrix over $GF(q)$ generated by a probability p

such that:

$$X_{i,j} = \begin{cases} k \text{ with prob } \frac{p}{q-1} \ \forall k = 1, \ldots, q-1 \\ 0 \text{ with prob } 1-p \end{cases}$$

Also, denote $W(.)$ as the weight of a matrix, i.e., the number of non-zero entries and $R(.)$ as the rank of a matrix.

4.4.2 Previous Results

In this section, we present some previous results related to rank and weight of the random matrix model.

- Relation between the probability p and the rank $R(X)$ [9]:

$$\mathbf{E}\left[\frac{1}{q^{R(X)}}\right] = \frac{1}{q^n}\left(1 + \sum_{k=1}^{n}\binom{n}{k}\gamma^k(1-\gamma)^{n-k}\left[1+(q-1)(1-p/\gamma)^k\right]^n\right) \quad (4.4)$$

  where $\gamma = 1 - 1/q$. Fig. 4.4 shows the empirical rank vs probability p which follows the Eq. 4.4.

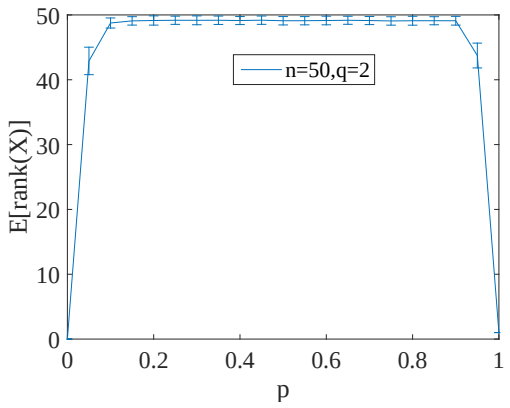


Figure 4.4: Empirical expected rank

- On the other hand, the expected weight given rank of matrix r, with the matrices are drawn uniformly at random in the set of matrices with given rank r is given

[70]:

$$E[W|R=r] = \frac{m(1-1/q)(1-1/q^r)}{(1-1/q^n)^2}$$

where $m = n^2$

- Expected weight in random matrix model:

$$\mathbf{E}[W(X)] = p \times n^2$$

- Trivial relation:

$$\|X - Y\| \leq \epsilon \quad \leftrightarrow \quad W(X) - \epsilon \leq W(Y) \leq W(X) + \epsilon \tag{4.5}$$

$$\rightarrow \quad R(X) - \epsilon \leq R(Y) \leq R(X) + \epsilon \tag{4.6}$$

To the best of our knowledge, there is no known results about the closed-form distribution of rank, or the joint distribution of weight and rank for the described model, except for case of the the elements are uniformly distributed. In the next section, we present our initial results on the joint distribution between weight and rank of matrix and apply the results to show the min-rank property of uniform noise model.

4.5 Main Results

4.5.1 Uniform Noise Model

Since the weight of N: $W(N) \leq \epsilon$ then we have $W(X_0) - \epsilon \leq W(Y) \leq W(X_0) + \epsilon$. In this model, we assume that the noise matrix N behaves such that the received matrix Y are uniformly random in the set of matrices with weight $\in (W(X_0) - \epsilon, W(X_0) + \epsilon)$. The model is illustrated in Fig. 4.5.

Theorem 15. ***(Number of matrices related to weight and rank)*** *Denote $C(W \geq w, R = r)$ as the number of matrices in $F_q^{n \times n}$ which have weight that is larger or equal to w and rank is equal to r. Then*

$$C(W \geq w, R = r) \leq \frac{m(q-1)(q^r-1)q^{2n-r-1}}{w(q^n-1)^2} \frac{\prod_{i=0}^{r-1}(q^n - q^i)^2}{\prod_{i=0}^{r-1}(q^r - q^i)} \tag{4.7}$$

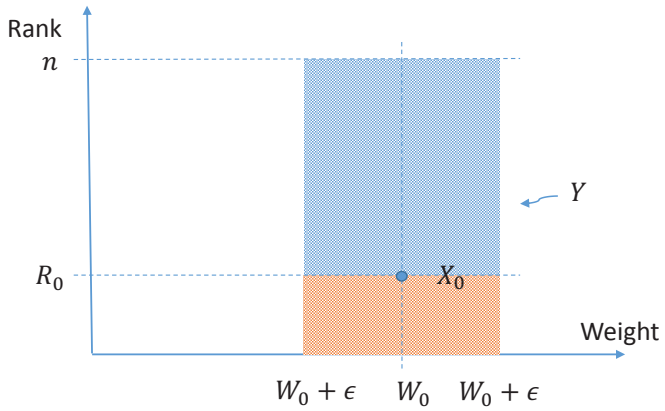


Figure 4.5: Uniform Noise Model

For the case $q = 2$ and $p = \frac{1}{2}$, we have:

$$C(W \geq w, R = r) \quad \leq \quad \frac{m}{w} \frac{2^{2n}}{2^{r+1}} \prod_{i=1}^{r-1} \frac{(2^n - 2^i)^2}{(2^r - 2^i)} \tag{4.8}$$

The upper bound on the joint distribution for the case $n = 10$ is illustrated in Fig. 4.6.

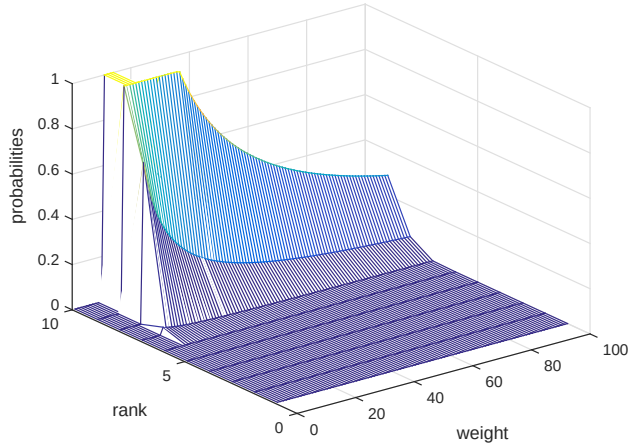


Figure 4.6: Upper bound for $P(W, R)$ with $n = 10$

Theorem 16. ***(Min-rank property in Uniform Model)*** *Draw matrix Y uniformly at random in the set of matrix with weight $w \in (w_0 - \epsilon, w_0 + \epsilon)$ (all matrices have equal probability). Then the probability P that Y have lower rank than matrix X_0 with $w(X_0) = w_0$ and $rank(X_0) = r_0$:*

$$\begin{aligned} \boldsymbol{P}(rank(Y) < rank(X_0)) &\leq \frac{\frac{m}{w_0-\epsilon}\sum_{r=0}^{r_0-1}\frac{2^{2n}}{2^{r+1}}\prod_{i=1}^{r-1}\frac{(2^n-2^i)^2}{(2^r-2^i)}}{\sum_{w=w_0-\epsilon}^{w_0+\epsilon}\binom{m}{w}} && (4.9)\\ &= O\left(\lambda\frac{n}{a^{n^2}}\right) \to 0 && (4.10)\end{aligned}$$

as $n \to \infty$ with condition that

$$\beta, \zeta, \eta \in (0,1) \quad (4.11)$$

$$\eta < \zeta \quad (4.12)$$

$$\gamma > 0 \quad (4.13)$$

with following notation:

$$\begin{aligned} \beta &= \frac{r_0}{n} && (4.14)\\ \zeta &= \frac{w_0}{n^2} && (4.15)\\ \eta &= \frac{\epsilon}{n^2} && (4.16)\\ \theta &= \begin{cases} \zeta - \eta & \text{if } \zeta < 1/2 \\ \zeta + \eta & \text{if } \zeta \geq 1/2 \end{cases} && (4.17)\\ \alpha &= H(\theta) && (4.18)\\ \lambda &= \frac{\beta}{32(\zeta-\eta)\eta} && (4.19)\\ \gamma &= \alpha + \beta^2 - 2\beta && (4.20)\\ a &= 2^\gamma && (4.21)\end{aligned}$$

Corollary 17. *Denote $b = a - 1$. For any $\delta > 0$ and $n \geq \frac{\lambda}{b\delta}$, we have*

$$\boldsymbol{P}\left(rank(Y) < rank(X_0)\right) = O(\delta).$$

Example 12. *Suppose* $\frac{r_0}{n} = 0.3; \frac{w_0}{n^2} = 0.4; \frac{\epsilon}{n^2} = 0.2$, *we illustrate Theorem 16 in Fig. 4.7. We can see that the event* $\{rank(Y) \geq rank(X_0)\}$ *occurs with almost certainty as the size of matrix increases (min-rank property).*

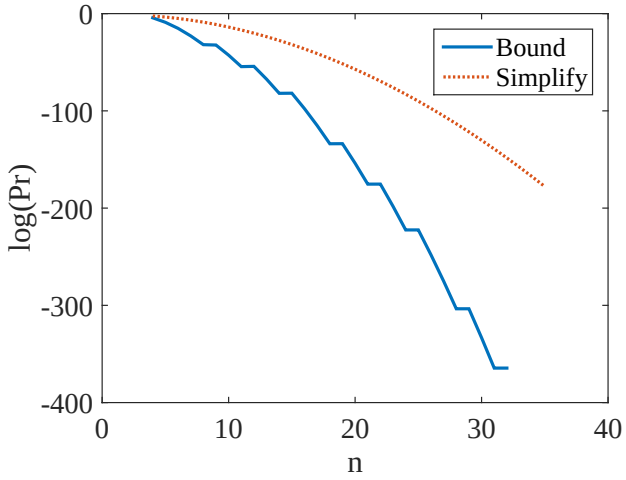


Figure 4.7: Bound and its simplified form on the min-rank property of uniform model

Theorem 18. *Change uniformly at random* ϵ *entries of any matrix* $X \in F_q^{n\times n}$ *with small weight* $w < n$ *and get matrix* Y. *We have:*

$$\boldsymbol{P}\left(rank(Y) \geq rank(X)\right) \geq \left(1 - \frac{w^2}{m - \epsilon + 1}\right)^{\epsilon}.$$

4.5.2 Other results

Definition 19. *Define the set* $\mathcal{X}_i^j = \{X \in F_q^{n\times n} : rank(X) = i; rank(X + N) = i + j \ \forall w(N) = j\}$

We have the following theorem

Theorem 19.

$$\begin{cases} |\mathcal{X}_1^1| = (2^n - n - 1)^2 \\ |\mathcal{X}_2^1| = \sum\limits_{t_1+t_2+t_3+t_4=n} \frac{n!}{t_1!t_2!t_3!t_4!} \end{cases} \tag{4.22}$$

Theorem 20. ***Number of matrices of given rank and weight over*** $F_q^{n\times n}$

$$C(W=w,R=r)=\begin{cases}0 & \text{if } w<r\\ a!\binom{n}{a}^2 & \text{if } w=r=a\\ (n-1)a\times a!\binom{n}{a}^2 & \text{if } w=a+1; r=a\end{cases} \tag{4.23}$$

4.6 Open Problem

Problem 1. *Denote* $S\subset F_q^{n\times n}$ *as the set of minimizers of min-rank decoder. If* S *is a singleton set then min-rank decoder has unique optimal solution, otherwise we can define the error event* $\mathcal{E}$ *that the min-rank decoder problem cannot recover exactly original matrix as:* $\mathcal{E}=\{|S|>1\}\cup\{\{|S|=1\}\cap\{X^*\neq X\}\}$. *Find the asymptotic analysis of* $\boldsymbol{P}(\mathcal{E})$*?*

Problem 2. *Find the efficient algorithms to solve the min-rank decoder?*

Following are three proposed algorithms to solve the min-rank optimization problem:

- *Algorithm 1: Search through all the set* $\mathcal{S}_Y(\epsilon)$ *and find the min-rank matrix.*
- *Algorithm 2: Sample in the set* $\mathcal{S}_Y(\epsilon)$ *and record the minimum rank matrix.*
- *Algorithm 3: Use decomposition and enumerate basis*

$$X=\sum_{l=1}^{r}u_lv_l^T=UV^T$$

where $u_l,v_l\in F_q^n$ *and* $U,V\in F_q^{n\times n}$.

Problem 3. *Find the joint distribution of weight and rank in term of the defined random matrix probability* p*?*

$$\boldsymbol{P}(W=w,R=r)=f(p)$$

Problem 4. *Find the set of strictly min-rank matrices* $\mathcal{M}^*(r,n,2\epsilon)$ *? as defined below:*

Definition 20. *The set of strictly min-rank matrices is defined as* $\mathcal{M}^*(r,n,2\epsilon)=\{X\in F_q^{n\times n}: rank(X)=r; rank(X+N)>r\ \forall N\in F_q^{n\times n}: 0<W(N)\leq 2\epsilon\}$

Chapter 5: Location Assisted Coding (LAC) for WiFO: A Hybrid WiFi and Free Space Optical High Speed WLAN of Femtocells

5.1 Introduction

In a report by Strategy Analytics, the numbers of homes with WiFi and public WiFi hotspots have increased steadily as the price of broadband access service is becoming more affordable. In fact, wireless service providers have taken advantage of the wide spread WiFi deployments to carry their cellular traffic, alleviating the problem of limited radio frequency (RF) spectrum. Cisco reported that 46 percent of the total cellular data traffic was offloaded through WiFi or femtocells in 2014, and the monthly global mobile data traffic will surpass 24.3 exabytes by 2019. That said, WiFi devices are projected to continue their significant growth trend, fueled by the emerging markets for smart homes and the Internet of Things (IoT). Consequently, the limited wireless capacities of the current WiFi systems will not be able to support many wireless devices and bandwidth intensive applications in the near future.

While much research has focused on 802.11a to increase the current WiFi capacity, it is noted that such an approach typically requires complex circuitry power modulators/demodulators due to sophisticated modulation schemes to obtain high bit rates. On the other hand, recent advances in free space optical (FSO) technology promise a complementary approach to increase wireless capacity with minimal changes to the existing wireless technologies and simpler designs. The solid state light sources such as Lighting Emitting Diode (LED) and Vertical-cavity Surface-Emitting Laser (VCSEL) are now sufficiently mature that it is possible to transmit data at high bit rates reliably at low power consumption using simple modulation scheme such as ON-OFF Keying. Importantly, the FSO technologies do not interfere with the RF transmissions. However, such high data rates are currently achievable only with point-to-point transmissions and not well integrated with existing WiFi systems. This drawback severely limits the mobility of the free space optical wireless devices.

That said, our work provides the following contributions. First, we describe a hybrid

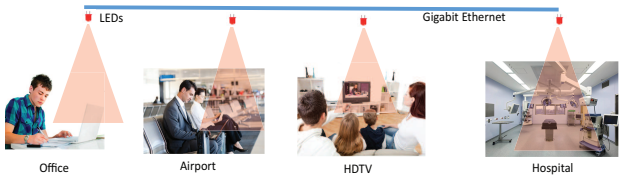


Figure 5.1: Use Scenario

WiFi-FSO (WiFO) [94] WLAN that can provide orders of magnitude increase in throughput over existing WiFi systems while maintaining seamless mobility. The proposed WiFO architecture is based on the femtocell architecture [15], [16] in which transmissions take place in confined areas (non-overlapped cells) to reduce interference. On the other hand, using a dense deployment of overlapped femtocells can result in higher bandwidth and greater mobility. In particular, our second contribution is a novel cooperative transmission scheme, known as Location Assisted Coding (LAC) technique that takes advantage of the receiver's location information to eliminate interference and achieve high bit rates. LAC allows multiple receivers including ones in an overlapped areas to obtain data from multiple transmitters simultaneously without interference. We also provide theoretical and numerical results of LAC technique for random deployment topologies. Third, we formulate the multi-user rate allocation problems and present the solutions together with their optimality analyses.

We also note that the LAC technique in WiFO system is an advanced version of the NC techniques (mentioned in Chapter 2) since the system is able to process different data flows (the RF packets and FSO packets) simultaneously. Also, based on the locations of all receivers (topology), data at transmitters are coded (mixed). Hence, each receiver need to decode the desired data from the coded packets, which is identical to NC techniques.

The paper is organized as follows. In Section 5.2, we discuss related work on free space optical communication and coding techniques. In Section 5.3, we provide an overview of WiFO that serves as a background for the proposed LAC technique in Section 5.4. Section 5.5 provides the theoretical and numerical analysis of the proposed LAC technique for a number of random topologies. In Section 5.6, we formulate and solve the problem of efficient rate allocation for multiple receivers based on LAC coding scheme. Finally, we provide a few concluding remarks in Section 5.7.

5.2 Related Work

In this section, we first briefly discuss a few related work on hybrid RF-FSO communication systems then highlight the differences between our work on LAC and the popular cooperative transmission techniques Multiple Input Multiple Output (MIMO) as well as the classic results on multiuser communication theory.

Hybrid RF-FSO communication systems. There have been several studies on RF-FSO hybrid systems. The majority of these studies, however are in the context of outdoor point-to-point FSO transmission, using a powerful modulated laser beam. Due to the instability of the FSO link over long distance transmission, an RF link is often used as a back up link [97], [10], [55]. Usually, a low capacity RF link is used when the primary FSO link fails or degrades significantly due to rain, fog or other environmental conditions. [52] provided a routing framework that maximizes the fairness index, which defined as the minimal ratio of data transmitted and data required among all traffic profiles. Backup RF link is made more available when the traffic is more delay sensitive. There are also recent literature on joint optimization of simultaneous transmissions on RF and FSO channels. For example, in [1], [61], [2], [89], and [93], the authors considered a joint coding schemes for both FSO and RF channels. [1] proposed a rateless coding scheme and the advantage of rateless coding scheme is proved. In [61], the authors studied the outage probability in a FSO/RF hybrid system and presented a power allocation scheme to minimize the outage probability. [2] optimized a FSO/RF hybrid network with respect to the location of the optical transceivers. A more comprehensive optimization problem is presented in [89]. Many aforementioned FSO/RF systems are designed for outdoor environments where attenuation/fading is due mainly to the weather conditions or scintillations. In contrast, our work is focused on Wi-Fi and FSO system for indoor environments where fading is due mainly to geometry of the cone beams.

Cooperative Transmissions. LAC is similar to Multiple-Input Multiple-Output (MIMO) techniques that have been used widely in communications systems to achieve significantly higher data rates than traditional single-input and single-output systems [76], [7], [44]. In both LAC and MIMO, the spatial dimension is key to increase data rate. On the other hand, due to the simplicity of On-Off Keying, or more generally, the Pulse Amplitude Modulation (PAM) used in WiFO, LAC's spatial dimension is gained through the receiver's information location. More importantly, majority of MIMO

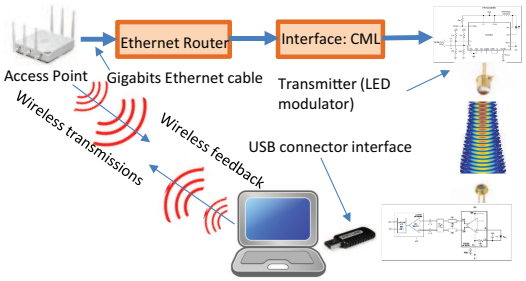


Figure 5.2: WiFO architecture

coding techniques are focused on multiple transmit and receive antennas for a single user [38],[42], [90], [78]. On the other hand, LAC's aim is to use multiple transmitters for multiple receivers simultaneously.

Multiuser information theory. From the information theory perspective, LAC is related to the well-known broadcast channel problem [25]. In this setting, single data source tries to transmit a common message to all receivers at the same time. The capacity for discrete memoryless channel is derived by Marton in [68] which generalizes the results in [25]. The achievable throughput of Gaussian broadcast channel is shown in [13] using dirty-paper coding technique. The idea behind dirty-paper coding [23] is that if the interference is known, then by adapting to the interference, the transmitter still can transmit at maximum rate despite of the interference. This result is extended to multiple receivers in [56]. LAC technique is different from these classic techniques in several ways. Specifically, LAC is designed for the proposed WiFO system [94] with short distance transmissions under well-controlled environments. Additionally, LAC directly relies on amplitude modulation and base band transmission which are not typically used in high-rate RF transmissions. Finally and importantly, LAC makes use of explicit location information of the receivers rather than channel information that are typically used in other coding schemes.

5.3 WiFO architecture

The WiFO system architecture is based on the femtocell architecture consisting of an array of triangular-lattice FSO transmitters deployed in the ceiling to provide FSO coverage for the floor area directly below. The WiFO system is designed to overcome the

capacity overload problem of the existing WiFi networks. The capacity overload problem arises due to the competitions among many users for a limited shared wireless bandwidth. Most often, the capacity overload problems are due to large down-link traffic since for many wireless applications, the amount of downlink traffic is on orders of magnitude larger than that of the uplink traffic. Although users can move around, they are often stationary, e.g., sitting on terminal benches at airports or lounges in hotel lobbies (Fig. 5.1). As such, a network of LEDs/VCSELs with the high-speed Ethernet infrastructure can be deployed directly above the appropriate spots to provide local high rate FSO transmissions, in addition to the WiFi transmissions. The current FSO technologies are inexpensive with the transmitters and receivers using LEDs/VCSELs and silicon photodiodes (PDs) that cost less than $20. In addition, they operate around 20 mW with good SNR and well within the eye safety (850 nm). Also, VCSEL-based transmitter can theoretically provide up to 1 Gbps, without interfering with WiFi.

The operations of the WiFO system are simple as shown in Fig. 5.2. All the data from the Internet to the devices in a WiFi network is first traversed through the Access Point (AP). For an IP packet of a given flow, the AP will decide whether to send the data on the WiFi or FSO channels. If it decides to send the data on the FSO channel for a particular device, the data will be encoded appropriately, and broadcast on the Gigabit Ethernet network with the appropriate information to allow the right FSO transmitter to receive the data. Upon receiving the data, the FSO transmitter relays the data to the intended device. If the AP decides to send the data on the WiFi channel, then it just directly broadcasts the data through the usual WiFi protocol. Upon receiving the data from the FSO channel, the receiver decodes the data, and sends a feedback/ACK to the AP via the WiFi channel. Feedback/ACK will allow the system to adapt effectively to the current network conditions.

We have successfully built a WiFO prototype from off-the-shelf components consisting of a single sender and a few receivers with limited mobility. Each receiver is capable of receiving data 50-100 Mbps simultaneously over both WiFi and FSO channels. The FSO transmitter used the LEDs (LED851L) to modulate light. For the FSO receiver, we used the FDS-100 silicon pin photodiodes. A demo can be seen at http://www.eecs.oregonstate.edu/$\sim$ thinhq/WiFO.html. Additionally our work was highlighted by NSF at http://news.science360.gov/archive/20150515.

In order to support seamless mobility as a device moves from one light cone to

another, we design an association protocol in which a device is associated with one or more FSO transmitters simultaneously when the user is in an overlapped coverage. Multiple devices can also associate with a single light cone. The transmitter associated with a mobile device is responsible for transmitting data for that device. When multiple FSO transmitters associate with a single device, they can send data simultaneously to the device. This contrasts with WiFi or cellular networks where each device can only be associated with a primary AP at any point in time. This salient feature is unique to the proposed WiFO network since as will be shown in Section 5.4, multiple WiFO devices associated with multiple transmitters will be able to receive and decode their data simultaneously from these FSO transmitters to increase the overall capacity using the LAC technique.

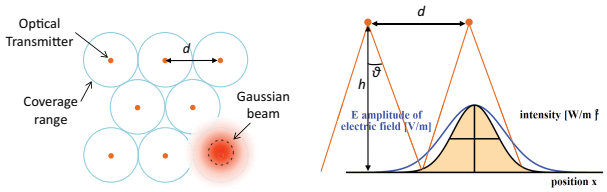
To establish association, each FSO transmitter broadcasts a beacon signal consisting of a unique ID periodically. A WiFO device automatically associates with one or more transmitters that provide sufficient SNRs. Upon receiving a beacon signal from a transmitter, the device sends back *alive* heartbeat messages that include the transmitter ID and the MAC address to the AP using WiFi channel. The AP then updates a table whose entries consist of the MAC address and the transmitter IDs which are used to forward the packets of a device to the appropriate transmitters. If the AP did not receive a heartbeat from a device for some period of time, it will disassociate that device, i.e., remove its MAC address from the table. We note that the association protocol requires messages exchanged between the AP and the mobile device. Since the FSO channel is a one-directional channel, the messages exchanged during the association protocol are sent using the WiFi channel.

5.4 Location Assisted Coding (LAC)

To aid our discuss on LAC, we provide a brief background on free space optical transmissions.

5.4.1 Optical Transmission

Fig. 5.3(a) shows a topology of non-overlapped triangular-lattice FSO transmitter array, i.e., FSO femtocells. The spacing between each transmitter is determined by:



(a) (b)

Figure 5.3: (a) Configuration of the optical transmitter array; (b) coverage of optical transmitters with a divergent angle of ϑ

$d = 2h \tan \vartheta$, where h is the height of the ceiling, and ϑ is the divergent angle of the transmitter. Using $h = 5$ meters (approximate height of ceilings in typical buildings) and $\vartheta = 7.5$ degrees, the coverage area for a single FSO transmitter is approximately 1.36 meter squares. The light from the optical transmitter is a Gaussian beam with a divergent angle of ϑ as shown in Fig. 5.3 (b). A large ϑ will cover a larger floor area and thus reduce the total number of FSO transmitters. However, the transmit power and the minimum optical power required at the optical receivers set the upper limit of ϑ. If two transmitters are co-located, then the received signal power for a user will be doubled, and thus higher data rates can be achieved. However, such simple deployment would increase the number of transmitters by two, without improving the mobility since there are still gaps between the circles as seen in Fig. 5.3. Although WiFi transmission can cover those gaps, the bit rates might be reduced in these areas.

One can use dense deployment of transmitter array to ensure no gaps. Using overlapped coverage will increase mobility and reduce bit error rate for a single receiver if two or more transmitters are used to send data to the single receiver. On the other hand, to avoid multi-user interference, transmitting data in overlapped areas may require TDMA or FDMA, which effectively reduces the overall capacity. This is typically done in WiFi or cellular networks. We show that this limitation is not necessary when the side information, specifically the user location, is used.

5.4.2 Problem Formulation

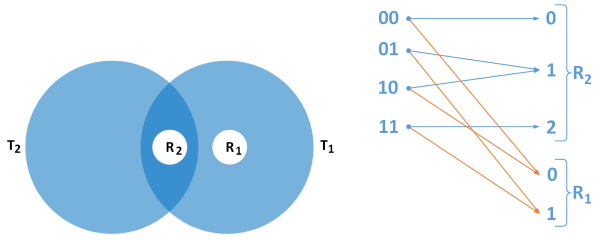
Assume that there are n FSO transmitters $T_1, T_2, \ldots T_n$, each produces a light cone that overlaps each other. We also assume that there are m receivers $R_1, R_2, \ldots R_m$, located

in the coverage areas. An FSO transmitter is assumed to use On-Off Keying (OOK) modulation where high optical power represents "1" and low power represents "0" [47]. On the other hand, a receiver is assumed to be able to detect different levels of light intensities. For example, if two transmitters send a "1" simultaneously to a receiver, the receiver would be able to detect "2" as light intensities from two transmitters add constructively. On the other hand, if one transmitter sends a "1" while the other sends a "0", the receiver would receive a "1". As an example, Fig. 5.4(a) shows a topology consisting of two FSO transmitters and two receivers. In this setting, the interference will occur at receiver R_2 if the transmitter T_1 and T_2 sends independent bits to R_1 and R_2.

The goal is to design a cooperative transmission scheme that allows the AP to send independent information to the receivers at the maximum rates. We begin with the channel model. We note that our problem of characterizing the achievable region appears to be similar to the well-known broadcast channels. Specifically, when the channel is a Degraded Broadcast Channel (DBC), the capacity region has been established [25], [71], [41]. However, we can show that WiFO channel is not a degraded broadcast channel, thus the well-known results on DBC are not applicable.

5.4.3 Channel Model

We first consider the topology shown in Fig. 5.4(a). Receiver R_2 is in the overlapped area, and therefore, can receive signals from both transmitters while receiver R_1 can receive signal from only one transmitter. Cooperative transmission scheme uses both



(a) (b)

Figure 5.4: (a) Topology for two FSO transmitters and two receivers; (b) Broadcast channels for two receivers.

transmitters to send independent information to each receiver simultaneously. This cooperative transmission scheme can be viewed as a broadcast channel in which the sender can broadcast four possible symbols: "00", "01", "10", and "11" with the left and right bits are transmitted by T_1 and T_2, respectively. Thus, there is a different channel associated with each receiver. Fig. 5.4(b) shows the broadcast channels for the two receivers R_1 and R_2. There are only three possible symbols for R_2 because it is located in the overlapped coverage of two transmitters. Therefore, it cannot differentiate the transmitted patterns "01" and "10" as both transmitted patterns result in a "1" at the receiver due to additive interference. On the other hand, there are only two symbols at receiver R_1 because it is located in the light cone of a single transmitter.

It is straightforward to see that the channel matrices for R_1 and R_2 associated with Fig. 5.4(b) are:

$$A_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

We note that the entry $A(i, j)$ of the channel matrix denotes probability that a transmitted symbol i to turn a symbol j at the receiver. Since we assume all sources of error are due to multi-user interference, $A(i, j)$ is either 0 or 1.

We note that it is straightforward to construct the channel matrices for scenarios with transmission errors. In particular, if we consider following simple i.i.d channel model. The probability of that a transmitted bit is flipped at a receiver in its transmission cone is α, and is identical for all transmitters. Furthermore, the transmissions are independent across transmitters and time slots. Thus, for the scenario in Fig. 5.4(b), channel matrices for R_1 and R_2 can be written as:

$$A_1 = \begin{bmatrix} 1-\alpha & \alpha \\ \alpha & 1-\alpha \\ 1-\alpha & \alpha \\ \alpha & 1-\alpha \end{bmatrix}, \quad A_2 = \begin{bmatrix} (1-\alpha)^2 & 2\alpha(1-\alpha) & \alpha^2 \\ \alpha(1-\alpha) & (1-\alpha)^2+\alpha^2 & \alpha(1-\alpha) \\ \alpha(1-\alpha) & (1-\alpha)^2+\alpha^2 & \alpha(1-\alpha) \\ \alpha^2 & 2\alpha(1-\alpha) & (1-\alpha)^2 \end{bmatrix}.$$

The same method can be used to construct channel matrices for other topologies. We note that due to the limited scope of the work and simplicity, we will not discuss the

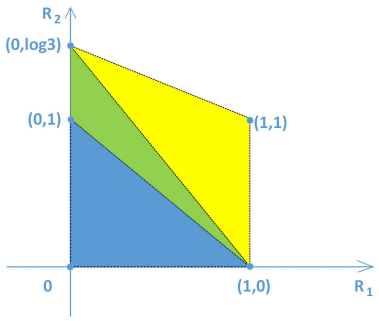


Figure 5.5: Achievable rate region for R_1 and R_2.

channels due to errors other than interference errors.

5.4.4 Achievable Rate Region

For the given channels in Fig. 5.4, Fig. 5.5 shows the three achievable rate regions for R_1 and R_2: blue, green, and yellow with each one is larger than the previous one. Each point (x, y) denotes the achievable rate, i.e., bits per transmission for R_1 and R_2, respectively. The blue region is achievable by simply using TDMA. Specifically, (1,0) is achievable by sending bits to R_1 exclusively and zero bits to R_2. Similarly, (0,1) is achievable by sending all the bits to R_2 and zero bits to R_1. Therefore, using TDMA and varying the fraction of time we use the strategy (0,1) and the remaining time we use the strategy (1,0), the blue achievable region can be achieved.

Such a scheme can be further enlarged by noting that the point $(0, \log 3)$ can be achieved. Indeed, $\log 3$ bits per transmission is achievable for R_2 if two transmitters are used to transmit the bits to R_2. Specifically, there are three distinct symbols at the output for R_2, namely: 0, 1, 2. Therefore, using the basic result in information theory, the maximum achievable rate is $\log 3$. Finally using TDMA between the strategies $(0, \log 3)$ and $(1, 0)$, the green region is achievable.

It is not obvious to see why one can further enlarge the achievable rate region as shown in yellow. In fact, we develop the LAC technique for general topologies of multiple transmitters and receivers. The achievable point (1,1) in Fig. 5.5 responsible for enlarging the rate region, is just a special case of the LAC technique to be discussed next.

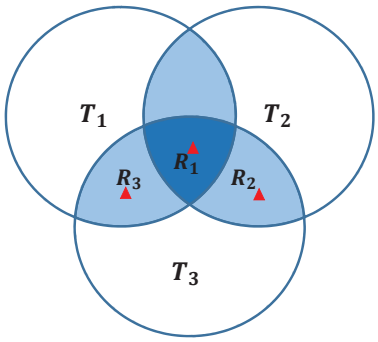


Figure 5.6: Example of three cones with interference

5.4.5 Encoding/Decoding Algorithms

In this section, we show the LAC encoding algorithm that under some conditions allows multiple receivers to receive independent bits simultaneously.

For simplicity, assume there are n transmitters and n receivers. Receiver R_i wants to receive bits b_i, $i = 1, 2, \ldots, n$. The goal is for the transmitters $T_1, T_2, \ldots, T_n$ to transmit bits $t_1, t_2, \ldots t_n$ simultaneously, but yet all the receivers R_i's will be able to recover their intended bits b_i's from the received signals r_i's. By assumption, b_i, $t_i \in \{0, 1\}$. On the other hand, $r_i \in \{1, 2, \ldots, n\}$ since the received signals add constructively.

Definition 21. *Let H be the matrix whose entry $H(i, j)$ is equal to 1 if receiver i can receive signal from transmitter j and 0 otherwise. H is called a topology matrix.*

For example, the topology matrix associated with Fig. 5.6 is:

$$H_3 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}.$$

Definition 22. *The system is said to achieve full rate if every receiver R_i can achieve 1 bit per transmission simultaneously.*

Note that the Definition 22 is meant for On-Off Keying modulation in which, at most one bit of information can be sent by any transmitter.

We have the following Proposition:

Proposition 21. *If the topology matrix H has full rank in* $\mathbf{GF}(2)$*, then it is possible for the system to achieve full rate.*

The proof for Proposition 21 is best presented via the following encoding and decoding algorithm that achieve full rate.

5.4.5.1 Encoding Algorithm

Let $b_1, b_2, \ldots, b_n \in \{0, 1\}$ be the bits wanted by receivers $R_1, R_2, \ldots, R_n$, and H is a full rank topology matrix.

Consider the following system of equations in $\mathbf{GF}(2)$:

$$\begin{cases} H(1,1)t_1 \oplus H(1,2)t_2 \oplus \ldots \oplus H(1,n)t_n = b_1 \\ H(2,1)t_1 \oplus H(2,2)t_2 \oplus \ldots \oplus H(2,n)t_n = b_2 \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \ldots \\ H(n,1)t_1 \oplus H(n,2)t_2 \oplus \ldots \oplus H(n,n)t_n = b_n \end{cases} \tag{5.1}$$

where $\oplus$ is addition in $\mathbf{GF}(2)$, i.e. $a \oplus b = (a + b) \mod 2$. Since H is a full-rank matrix in $\mathbf{GF}(2)$, we can solve the system of equations (5.1) above for unique t_1, t_2, ..., t_n in terms of b_1, b_2, ..., b_n. The solution for t_1, t_2, ..., t_n is a linear combination of b_1, b_2, ..., b_n. We claim that if the transmitters T_1, T_2, ..., T_n transmit the bits $t_1, t_2, \ldots, t_n$, respectively, then all the receiver $R_1, R_2, \ldots, R_n$ will be able to receive their desired bits $b_1, b_2, \ldots, b_n$, even if a receiver is in the overlapped area cover by multiple transmitters. We note that in WiFO, the AP having access to all the flows of data, transmits $t_1, t_2, \ldots t_n$ to the transmitters $T_1, T_2, \ldots, T_n$, respectively. T_i then transmits t_i. Thus, the encoding procedure involves solving a system of linear equations. One assumption is that the AP knows which regions the receivers are in, and therefore it can construct the topology matrix H. The AP obtains this information from the mobility protocol described briefly in Section 5.3. If a receiver is associated with two given transmitters then the AP knows that the receiver is in an overlapped region of those two transmitters. When all receivers are in separate non-overlapped regions, the H matrix is an identity matrix, and therefore full-rank. Thus, $t_i = b_i$.

Table 5.1: Transmitted signals, received signals and recovered bits in **GF**(2) for three cones in Fig. 5.6

b_1	b_2	b_3	t_1	t_2	t_3	r_1	r_2	r_3	$\hat{b}_1$	$\hat{b}_2$	$\hat{b}_3$
0	**0**	**0**	0	0	0	0	0	0	**0**	**0**	**0**
0	**0**	**1**	0	1	1	2	2	1	**0**	**0**	**1**
0	**1**	**0**	1	0	1	2	1	2	**0**	**1**	**0**
0	**1**	**1**	1	1	0	2	1	1	**0**	**1**	**1**
1	**0**	**0**	1	1	1	3	2	2	**1**	**0**	**0**
1	**0**	**1**	1	0	0	1	0	1	**1**	**0**	**1**
1	**1**	**0**	0	1	0	1	1	0	**1**	**1**	**0**
1	**1**	**1**	0	0	1	1	1	1	**1**	**1**	**1**

5.4.5.2 Decoding Algorithm

A receiver R_i needs to be able to recover the bit b_i from the received signal r_i which can be represented as:

$$\begin{cases} r_1 = H(1,1)t_1 + H(1,2)t_2 + \ldots + H(1,n)t_n \\ r_2 = H(2,1)t_1 + H(2,2)t_2 + \ldots + H(2,n)t_n \\ \quad \ldots \\ r_n = H(n,1)t_1 + H(n,2)t_2 + \ldots + H(n,n)t_n \end{cases} \tag{5.2}$$

The receiver recovers b_i by performing

$$r_i \mod 2 = \hat{b}_i. \tag{5.3}$$

We claim that $b_i = \hat{b}_i$. This can been seen by performing a mod 2 operation on both sides of equations (5.2) which results in the equations (5.1). Or simply, if r_i is even then R_i decodes bit b_i as "0", and "1" otherwise. As a result, each receiver can decode its bits correctly and independently in presence of interference. Furthermore, no other information regarding other users is required. Therefore, the decoding procedure is very simple.

Example 1. Consider the overlapped regions as shown in Fig. 5.6. The topology matrix for this case is:

$$H_3 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

This matrix is also full-rank, therefore using LAC, one can transmit data at full rate. Specifically, we solve the following system of equations for t_1, t_2, t_3 in $\mathbf{GF}(2)$.

$$\begin{cases} t_1 \oplus t_2 \oplus t_3 = b_1 \\ \quad t_2 \oplus t_3 = b_2 \\ \quad t_1 \oplus t_3 = b_3 \end{cases} \tag{5.4}$$

or

$$\begin{cases} t_1 = b_1 \oplus b_2 \\ t_2 = b_1 \oplus b_3 \\ t_3 = b_1 \oplus b_2 \oplus b_3 \end{cases} \tag{5.5}$$

Now, if the three transmitters transmit bits as shown in (5.5), then at the receivers, the received signals are:

$$\begin{cases} r_1 = t_1 + t_2 + t_3 \\ r_2 = t_2 + t_3 \\ r_3 = t_1 + t_3 \end{cases} \tag{5.6}$$

The received signals and the recovered bits using (5.3) for all cases are shown in Table 5.1. We can see that the recovered bits are exactly the intended bits.

5.4.6 Coding Scheme for GF(q)

The coding scheme shown in previous sections uses $\mathbf{GF}(2)$. This is based on the assumption that the transmitters can only transmit "0" and "1" using OOK modulation. If the transmitters can transmit with q levels from "0" to "$q-1$" where q is a prime number, we can extend the coding scheme to $\mathbf{GF}(q)$. Specifically,

1. At the transmitter, we still use the system of equations (5.1) except that the

addition now is computed over $\mathbf{GF}(q)$, i.e. $a \oplus b = (a+b) \mod q$. Using Gaussian elimination, solution for $t_1, t_2, \ldots, t_n$ could be achieved if the matrix H is full-rank in $\mathbf{GF}(q)$.

2. At the receiver, we still have the system of equations (5.2) which is now in $\mathbf{GF}(q)$. By taking $\mod q$ operation on both sides of equations (5.2), we have $r_i \mod q = b_i$. Therefore, we can recover the transmitted bits by computing:

$$\hat{b}_i = r_i \mod q. \tag{5.7}$$

Advantages of using multiple levels q are twofold. First, it increases the capacity with fewer number of transmitters. Second, it is easier to have full-rank topology matrix. A matrix that is not full-rank in $\mathbf{GF}(2)$ might be full-rank in $\mathbf{GF}(q)$ with $q > 2$. Therefore, by letting the transmitters send signal with multiple levels, we might be able to transmit at full rate. This could be seen in the following example.

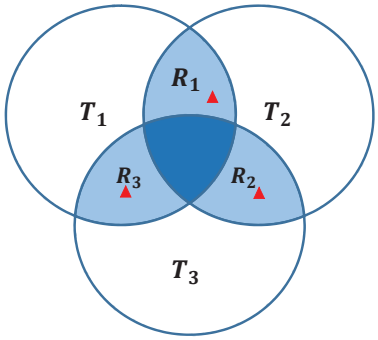


Figure 5.7: Example of three cones with rank 2 topology matrix

Example 2. Consider the following topology matrix

$$H_3' = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

This is the topology matrix for the case illustrated in Fig. 5.7. Clearly, H_3' is not full-rank in $\mathbf{GF}(2)$ since $rank(H_3') = 2$. However, in $\mathbf{GF}(3)$, it is a full-rank matrix. Therefore,

we can make use of the proposed coding scheme to achieve full rate. In this case, the transmitters can transmit levels $0, 1$ or 2.

The received signals and the recovered bits using for all cases are shown in Table 5.2. We can see that the recovered bits are exactly the intended bits.

5.4.7 Extended LAC

The proposed coding schemes in Section 5.4.6 are for n transmitters and n receivers with full-rank topology matrix. Now, we extend the coding schemes to the general case where there are n transmitters and m receivers. The topology matrix H is now of size $m \times n$ and has rank k ($k \leq \min(m, n)$). As a result, only k equivalent independent single channels are used at any point of time [44]. Therefore, our goal is to derive a coding scheme to achieve k/n of the full rate R. This is also the theoretical maximum rate.

Specifically, since the topology matrix has rank k in $\mathbf{GF}(q)$, we can pick k linearly independent rows out of n rows in the matrix. Denote U as a set of k linearly independent rows in $\mathbf{GF}(q)$: $U = \{u_1, u_2, \ldots, u_k\}$ and V as the set of the other $m - k$ rows: $V = \{v_1, v_2, \ldots, v_{m-k}\}$. As a result, $v_1, v_2, \ldots, v_{m-k}$ could be represented as linear combinations of $u_1, u_2, \ldots, u_k$ in $\mathbf{GF}(2)$. We assume that the matrix H has no row with all zero entries since if there is such a row, i.e., the corresponding receiver does not receive any signal from any cone, we just remove that receiver/user from the system. Notice that for each v_i, there always exists a row u_j such that if we swap the two rows, we still have a set of k linearly independent rows $\{u_1, u_2, \ldots, u_{j-1}, v_i, u_{j+1}, \ldots, u_k\}$. This property is proved in Proposition 22.

Proposition 22. *Consider a $m \times n$ matrix H of rank k in $\mathbf{GF}(q)$ ($k \leq \min(m, n)$). Assume that the matrix H has no row with all zero entries. Let $u_1, u_2, \ldots, u_k$ be k linearly independent row vectors and $v_1, v_2, \ldots, v_{m-k}$ be the other $m - k$ row vectors in the matrix.*

For each v_i, there always exists a row vector u_j such that if the two vectors are swapped, $u_1, u_2, \ldots, u_{j-1}, v_i, u_{j+1}, \ldots, u_k$ are still linearly independent.

Proof. See Appendix. □

Using this property, we can derive a coding scheme to achieve rate of $\frac{k}{n}R$ as follows.

Table 5.2: Transmitted signals, received signals and recovered bits in **GF**(3) for three cones in Fig. 5.7

b_1	b_2	b_3	t_1	t_2	t_3	r_1	r_2	r_3	$\hat{b}_1$	$\hat{b}_2$	$\hat{b}_3$
0	**0**	**0**	0	0	0	0	0	0	**0**	**0**	**0**
0	**0**	**1**	2	1	2	3	3	4	**0**	**0**	**1**
0	**0**	**2**	1	2	1	3	3	2	**0**	**0**	**2**
0	**1**	**0**	1	2	2	3	4	3	**0**	**1**	**0**
0	**1**	**1**	0	0	1	0	1	1	**0**	**1**	**1**
0	**1**	**2**	2	1	0	3	1	2	**0**	**1**	**2**
0	**2**	**0**	2	1	1	3	2	3	**0**	**2**	**0**
0	**2**	**1**	1	2	0	3	2	1	**0**	**2**	**1**
0	**2**	**2**	0	0	2	0	2	2	**0**	**2**	**2**
1	**0**	**0**	2	2	1	4	3	3	**1**	**0**	**0**
1	**0**	**1**	1	0	0	1	0	1	**1**	**0**	**1**
1	**0**	**2**	0	1	2	1	3	2	**1**	**0**	**2**
1	**1**	**0**	0	1	0	1	1	0	**1**	**1**	**0**
1	**1**	**1**	2	2	2	4	4	4	**1**	**1**	**1**
1	**1**	**2**	1	0	1	1	1	2	**1**	**1**	**2**
1	**2**	**0**	1	0	2	1	2	3	**1**	**2**	**0**
1	**2**	**1**	0	1	1	1	2	1	**1**	**2**	**1**
1	**2**	**2**	2	2	0	4	2	2	**1**	**2**	**2**
2	**0**	**0**	1	1	2	2	3	3	**2**	**0**	**0**
2	**0**	**1**	0	2	1	2	3	1	**2**	**0**	**1**
2	**0**	**2**	2	0	0	2	0	2	**2**	**0**	**2**
2	**1**	**0**	2	0	1	2	1	3	**2**	**1**	**0**
2	**1**	**1**	1	1	0	2	1	1	**2**	**1**	**1**
2	**1**	**2**	0	2	2	2	4	2	**2**	**1**	**2**
2	**2**	**0**	0	2	0	2	2	0	**2**	**2**	**0**
2	**2**	**1**	2	0	2	2	2	4	**2**	**2**	**1**
2	**2**	**2**	1	1	1	2	2	2	**2**	**2**	**2**

Algorithm 4: k-bit Coding Algorithm

1. Find a set U of k linearly independent rows of H: $U = \{u_1, u_2, \ldots, u_k\}$. Put the other $m - k$ rows to a set V.
2. From the set of linearly independent rows $X = U$, create a $k \times n$ matrix $\tilde{H}$. $\tilde{H}$ has rank k. Therefore, we can pick k columns from $\tilde{H}$ to create $k \times k$ matrix H' that has rank k.
3. Deploy the proposed coding schemes (Section 5.4.6) for k transmitters and k receivers corresponding to the full-rank matrix H'.
4. Take a row v_i out from V, search through the set U and find a row $u_j \in U$ such that if we replace u_j by v_i in the set U, we obtain a set of linearly independent rows U'.
5. Go to step 2) with $X = U'$.
6. Keep doing that until V is empty.

The result we obtain here is a sequence of $m - k + 1$ pairs. Each pair includes a set of k transmitters and a set of k receivers with their full-rank $k \times k$ topology matrix H' from Step 3. They are the sets of active transmitters and receivers allowing k receivers to decode its signal correctly in a time slot. By periodically using the pairs of active transmitter set and receiver set in the sequence with the proposed coding schemes in Section 5.4.6, we can achieve rate of $\frac{k}{n}R$ and allow m receivers share the bandwidth. After the algorithm terminates, each receiver appears in at least one pair of transmitter and receiver sets and therefore has a chance to receive signal and decode it. Nevertheless, this coding scheme does not guarantee the throughput fairness among all users in the system.

5.5 Performance Analysis of LAC for Various Topologies

In this section, we present the performance analysis of LAC using OOK modulation. In order to show the robustness of LAC, we compare it with a basic code (BC). BC can only work under the condition that each receiver is located in a non-overlapped area.

The number of pairs of receiver and cone that satisfies this condition is the maximum number of bits that can be transmitted at a time.

5.5.1 Bernoulli Model

Suppose n receivers are located in n cones such that receiver i is belonged to cone j with probability p for any i, j. Then the topology matrix H is an $n \times n$ matrix in $\mathbf{GF}(2)$ such that:

$$H_{ij} = \begin{cases} 1 \text{ with probability } p \\ 0 \text{ with probability } 1-p. \end{cases}$$

Let q be the size of the finite field, and $\gamma = 1 - 1/q$. Denote

$$h = \sum_{k=1}^{n} \binom{n}{k} \gamma^k (1-\gamma)^{n-k} [1 + (q-1)(1-p/\gamma)^k]^n.$$

We have the following proposition regarding the achievable rate.

Proposition 23. *For the model above, the achievable rate R, defined as the average number of bits can be received per time slot, can be approximated as :*

$$R_{LAC} \approx n - \log_q (h+1) \tag{5.8}$$

for sufficiently large n.

Proof. See Appendix. □

5.5.2 Uniform Model

In this model, we set $p = \frac{1}{2}$ so that each entry in the topology matrix H has equal probabilities of having values 0 or 1.

Proposition 24. *For sufficiently large number of transmitters and receivers in a small area, the probability P_{LAC} of achieving full rate, i.e. $R_{LAC} = n$, approaches a constant. Specifically,*

$$P_{LAC} = 0.289. \tag{5.9}$$

Furthermore, the average achievable rate is:

$$R_{LAC} = \frac{1}{2^{n^2}} \sum_{k=1}^{n} k \prod_{i=0}^{k-1} \frac{(2^n - 2^i)^2}{2^k - 2^i}. \tag{5.10}$$

Proof. See Appendix. □

We now consider the basic coding (BC) scheme that does not tolerate interference. In other words, BC can only transmit at full rate if each receiver is strictly located in each non-overlapped region. We have the following proposition regarding BC performance for the Uniform model.

Proposition 25. *The probability that the BC scheme is able to transmit a full rate (n) is:*

$$P_{BC} = \frac{n!}{2^{n^2}}. \tag{5.11}$$

Furthermore, the average rate R for the BC scheme is:

$$R_{BC} = \frac{1}{2^{n^2}} \sum_{k=1}^{n} k! \binom{n}{k}^2 2^{(n-k)^2}. \tag{5.12}$$

Proof. See Appendix. □

To verify the theoretical analyses, Fig. 5.8 shows the probability of being able to send bits at full rate for LAC and BC schemes as a function of n. As seen, this probability decreases and approaches 0.289 for the LAC scheme as predicted. On the other hand, the same probability decreases to zero quickly for the BC scheme.

Also, in Fig. 5.9, the average rate of the LAC scheme is much larger than that of BC. In addition, the rate of LAC shows an roughly linear relation to the number of cones while the rate of BC decreases as the number of cones increases.

5.6 Time Minimization and Rate Allocation

So far, we have demonstrated that LAC performs very well in term of bandwidth efficiency since the system always operates at full rate using the $k - bit$ Coding Algorithm.

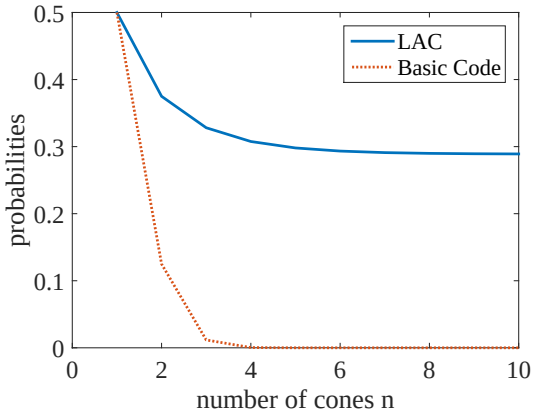


Figure 5.8: Full rate transmission probabilities versus different number of cones

However, it is not clear how LAC can be used in the scenarios where multiple receivers request different transmission rates. We first begin with the time minimization problem.

5.6.1 Time Minimization

The time minimization problem can be described as follows.

- There are m receivers $R_1, R_2, \ldots, R_m$. Each receiver requires a different number of bits, i.e., receiver R_i needs b_i bits.
- Let k be the rank of the topology matrix H. Therefore, in each round, the transmitters can collectively transmit no more than k information bit to the receivers.
- Let N be the number of rounds required for the transmitters to send the requested bits to all the receivers. The goal is to find the coding/scheduling scheme that minimizes the number of rounds N.

To formulate the problem, let us denote the set $\mathcal{U} = \{u_1, u_2, \ldots, u_m\}$ as the set of all rows of matrix H. Furthermore, denote $\mathcal{D} = \{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_d\}$ where $|D| = d$ as the set that contains all distinct non-empty subsets $\mathcal{V}_i \subset \mathcal{U}$ such that all vectors in $\mathcal{V}_i$ are linearly independent. Obviously, since $rank(H) = k$ we have $|\mathcal{V}_i| \leq k$ and $0 < d \leq \sum_{i=1}^{i=k} \binom{m}{k}$.

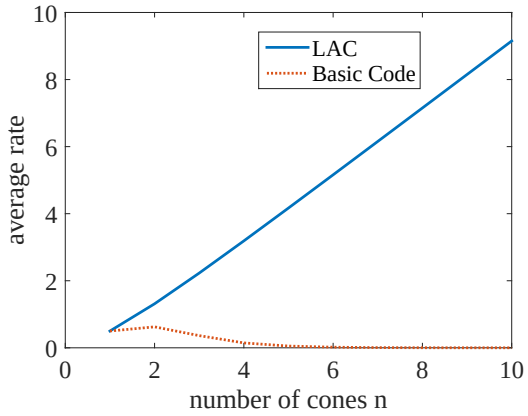


Figure 5.9: Average rate versus different number of cones

We can construct any LAC-based coding scheme C as follows. At each round, we choose a subset $\mathcal{V}_i$ for any $1 \leq i \leq d$ where $|\mathcal{V}_i| = l$ then l receivers corresponding to l independent vectors in $\mathcal{V}_i$ will be served using LAC. The process repeats until all receivers receive their desired number of bits (some receivers can receive more bits than their desired number of bits).

Let $A \in [0,1]^{m \times d}$ be the matrix which represents the set $\mathcal{D}$

$$A_{ij} = \begin{cases} 1 \text{ if } \mathcal{V}_j \text{ includes receiver } R_i \\ 0 \text{ otherwise} \end{cases}$$

Define $x = [x_1, x_2, \ldots, x_d]^T$ where $x_i \in \mathbb{Z}^+$ denotes the number of rounds that we choose subset $\mathcal{V}_i$. The total number of rounds:

$$N = \sum_{i=1}^{m} x_i$$

Therefore, the number of bits that receiver R_i receives which requires to be no less than b_i can be written as the constraint:

$$\sum_{j=1}^{d} x_j A_{ij} \geq b_i \ \forall i \tag{5.13}$$

$$\leftrightarrow \quad Ax \succeq b \tag{5.14}$$

where $b = [b_1, b_2, \ldots, b_m]^T$ is the vector representing the desired number of bits for each receiver, and $\succeq$ represents element-wise comparsion.

Given this notation, the Integer Linear Program for the time minimization problem can be formulated as follows.

Problem P1:

$$\begin{aligned} &\text{Minimize} && \sum_i x_i \\ &\text{Subject to} && \begin{cases} x \succeq \mathbf{0} \\ Ax \succeq b \end{cases} \end{aligned} \tag{5.15}$$

with variable $x \in \mathbb{Z}^{+d\times 1}$ and given $A \in [0,1]^{m\times d}, b \in \mathbb{Z}^{+m\times 1}$

We illustrates the problem in the following example.

Example 13. *We have* $m = 4, n = 3$ *and suppose* $(b_1, b_2, b_3, b_4) = (2, 2, 1, 1)$. *Assume that the topology matrix is*

$$H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

Then $rank(H) = 3$.

- *There are* $d = 12$ *feasible subsets in* $\mathcal{D}$: $\{(R_1); (R_2); (R_3); (R_4); (R_1, R_2); (R_1, R_3); (R_1; R_4)$ $(R_2, R_3); (R_2, R_4), (R_3, R_4), (R_1, R_2, R_3), (R_1, R_2, R_4)\}$.
- *The optimal scheme* C^* *would need* 2 *rounds:* $(R_1, R_2, R_3), (R_1, R_2, R_4)$.
- *Compare to some other scheme* C *would need* 3 *rounds:* $(R_1, R_2), (R_1, R_2), (R_3, R_4)$.

We note that (5.15) is the generalized form of the Covering Integer program [92] in which $b_i = 1\ \forall i$. Furthermore, the Covering Integer program is shown to be equivalent to Set Cover problem which has been shown to be NP-hard [51]. Thus, the Time Minimization problem is NP-hard, and many heuristic algorithms can be used to solve this problem. Therefore, we will now focus on a proportional rate allocation problem in the next section.

5.6.2 Proportional Rate Allocation

Recall that if $rank(H) = k$ then by using the $k - bit$ Coding algorithm in LAC, we can serve k receivers in each round. Certainly, we would prefer to transmit k bits per time slot in any round. That said, there are many coding schemes that can achieve the maximum rate in which we would prefer a coding scheme that can also achieve target rate of each receivers. To do so, we will use the randomized approach to design our scheduling/coding schemes.

To illustrate our approach, suppose there are 3 receivers R_1, R_2, R_3 and their associated topology matrix H, with $rank(H) = 2$ such that the systems can serve both R_1 and R_2 or both R_2 and R_3 in a round. A coding scheme C can be implemented as follows. In each round with probability of 0.5, we choose the subset $\mathcal{V}_1 = (R_1, R_2)$ to serve and with remaining probability of 0.5, we choose the subset $\mathcal{V}_2 = (R_2, R_3)$ to serve. By applying coding scheme C with probabilistic policy $x = [0.5, 0.5]$, the resulted rate distribution r of over three receivers R_1, R_2, R_3 is $[0.5, 1, 0.5]$ respectively or can be normalized as $[\frac{1}{4}, \frac{1}{2}, \frac{1}{4}]$.

Due to the weak law of large numbers, it is guaranteed that the average rate distribution achieved by a randomized policy/schedule x would converge to its true target rate in probability, i.e, when the number of rounds n applying policy x is large enough, the average rate distribution $\bar{r}$ would be within ϵ close to the resulted rate distribution r:

$$\lim_{n\to\infty} P(|\bar{r} - r| \geq \epsilon) = 0$$

Suppose the number of bits that three receivers R_1, R_2, R_3 requires are $[500, 1000, 500]$ respectively then the desired rate distribution b can also be normalized as $[\frac{1}{4}, \frac{1}{2}, \frac{1}{4}]$. Hence, this desired distribution can be achieved by applying the above coding scheme $C(x)$.

Now, let us formulate an optimization problem for this proportional rate allocation problem. The notations are similar to the Time Minimization problem.

- In this case, the set $\mathcal{D}$ only includes subset $\mathcal{V}_i$ such that $|\mathcal{V}_i| = k$. Also $0 < d \leq \binom{m}{k}$
- Matrix A is defined the same as previous section

$$A_{ij} = \begin{cases} 1 \text{ if the set } \mathcal{V}_i \text{ includes receiver } R_j \\ 0 \text{ otherwise} \end{cases}$$

- Let $x = [x_1, x_2, \ldots, x_d]^T$ where x_i be the probability that $\mathcal{V}_i$ is chosen at each round. Also,

$$\begin{cases} x \geq 0 \\ \sum_i x_i = 1. \end{cases}$$

- Hence, the resulted rate distribution $r(x) = \frac{1}{k}Ax$.

Our first goal is to find a randomized coding scheme that can operate in the maximum rate (full rate) transmission while achieving as close as possible to a given target rate allocation. The problem can be described as follows.

- Let $b = [b_1, b_2, \ldots, b_m]^T$ be the desired rate distribution over all m receivers. Also, b is normalized such that $\sum_{i=1}^{m} b_i = 1$.
- The goal of this problem is to find a coding scheme x such that target rate distribution $r(x) = b$ or as close as possible to b.
- The distance from the obtained rate allocation distribution $r(x)$ to the target distribution b is defined by using vector norm:

$$||r(x) - b|| = ||\frac{1}{k}Ax - b||.$$

Thus, the problem can be formulated as a convex optimization form as:

Problem P2:

$$\begin{aligned} &\text{Minimize} && ||\tfrac{1}{k}Ax - b|| \\ &\text{Subject to} && \begin{cases} x \succeq \mathbf{0} \\ \mathbf{1}^T x = 1 \end{cases} \end{aligned} \tag{5.16}$$

The Problem **P2** is a non-negative least square problem (NNLS) which can be solved by active set method [60].

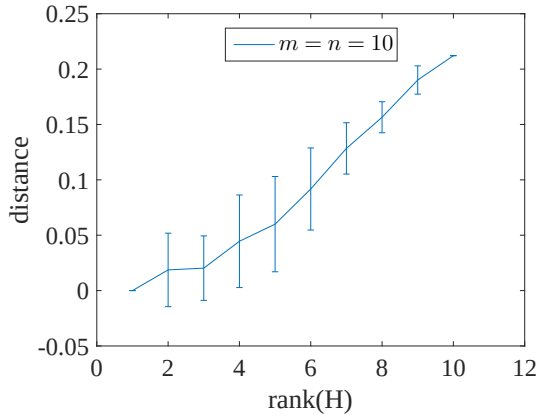


Figure 5.10: Average optimal value versus k (rank of topology matrix H)

Fig. 5.10 plots the average optimal values of the objective function in **P2** vs. rank of H's. In this simulation, the number of transmitters is equal to the number of receivers ($m = n = 10$). The topology matrices H are generated uniformly at random, and the their ranks are noted. However, for any H, we require that any receiver is covered by at least one transmitter. The desired rate allocation is a constant vector:

$$b = \begin{bmatrix} 0.19 & 0.21 & 0.18 & 0.02 & 0.1 & 0.05 & 0.07 & 0.08 & 0.01 & 0.09 \end{bmatrix}^T$$

The small distance implies that the solution of the corresponding randomized policy approximates the desired rate allocation well. As seen in Fig. 5.10, when the rank of topology matrix H increases, the system can transmit at full rate which exactly equal to the rank. However, all receivers need to participate in transmission in every time

slot. Thus, the policy is not sufficiently flexible to achieve the propotional target rate allocation. For example, when the matrix H is full rank, the only possible solution is $[0.1, 0.1, \ldots, 0.1]$ ($n = 10$) which can be far way from the given target rate allocation. On the other hand, when the topology matrix H has low rank, at each time slot, there are several options of choosing which receivers to serve. Hence, the system would have more flexibility to allocate the rate as desired.

When the systems operated in full-rate, it is possible that the desired rate allocation cannot be reached. In the case, the desired rate allocation is required, it can be achieved at the cost of reducing the overall transmission rate. In general, we would like to optimize the operation such that the overall rate is as high as possible while the desired rate allocation is obtained. We formulate this problem as follows.

- Denote $\mathcal{D}^{(k)} \subseteq \mathcal{D}$ as the set which only includes $\mathcal{V}_i$ such that $|\mathcal{V}_i| = k$. Note that when $\mathcal{D}^{(k)}$ is used, the transmission rate would be k.
- Let $A^{(k)}$ be the matrix representing set $\mathcal{D}^{(k)}$ (similar to previous A and $\mathcal{D}$) and $A^{(k)} \in [0,1]^{m \times d^{(k)}}$ where $d^{(k)} = |\mathcal{D}^{(k)}|$.
- A policy can be represented by vector $x = [x^{(1)}, x^{(2)}, \ldots, x^{(k)}]$ where $x^{(i)}$ is a $d^{(i)}$-vector corresponding to the probability that $A^{(i)}$ is chosen.
- The average rate of system is

$$R = kx^{(k)} + (k-1)x^{(k-1)} + \cdots + x^{(1)} = \sum_{i=1}^{k} ix^{(i)}$$

- The rate allocation distribution is

$$\frac{1}{k}A^{(k)}x^{(k)} + \frac{1}{k-1}A^{(k-1)}x^{(k-1)} + \cdots + A^{(1)}x^{(1)} = \sum_{i=1}^{k} \frac{1}{i}A^{(i)}x^{(i)} = b$$

The problem can be formulated as follows.

Problem P3:

$$\text{Maximize} \quad \sum_{i=1}^{k} i x^{(i)}$$

$$\text{Subject to} \quad \begin{cases} x \succeq \mathbf{0} \\ \mathbf{1}^T x = 1 \\ \sum_{i=1}^{k} \frac{1}{i} A^{(i)} x^{(i)} = b \end{cases} \tag{5.17}$$

The problem **P3** can be efficienttly solved via convex optimization framework [45].

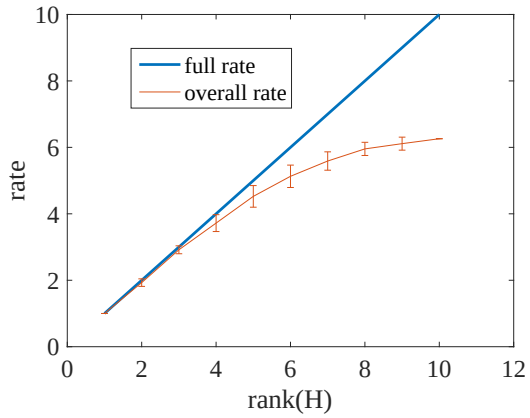


Figure 5.11: Average rate versus k (rank of topology matrix H)

Fig. 5.11 shows the overall rate of the system vs. the rank of the topology matrix H. The simulation setup/parameters are identical to that of Fig. 5.10. The proportional rate is now guaranteed to be the exact target rate allocation. On the other hand, the solution cannot achieve full rate. In fact, as the rank increases, hence the full rate increases, the gap between the overall resulted rate and the full rate increases.

5.6.3 Analytic Solution and Relaxation Algorithm

The convex optimization framework can help us solve the problem (5.16) and (5.17) but cannot give an analytic solution. We address this solution in the this section.

Let $\bar{A} = \frac{1}{k}A$. Due to [88], the equation systems

$$\bar{A}x = b$$

would exist a solution if

$$\bar{A}\bar{A^+}b = b \tag{5.18}$$

where $\bar{A^+} \in \mathbb{R}^{d\times m}$ is the Moore-Penrose pseudoinverse of $\bar{A}$. The solution (if exist) would be in the form

$$x = \bar{A}^+b + (I - \bar{A}^+\bar{A})w \tag{5.19}$$

for any $w \in \mathbb{R}^{d\times 1}$, which is the solution of optimization problem (5.16) if there is no constraint. With the introduce of the non-negative constraints, we still able to find the analytic solution for the NNLS problem in some special cases as follows.

Proposition 26. *Suppose $b \in Range(\bar{A})$ or there exist x^* such that $\bar{A}x^* = b$ then*

$$\boldsymbol{1}^T x^* = 1.$$

Proof. See Appendix. □

Proposition 27. *If (5.18) satisfies and there exists $x^* \geq 0$ in form of (5.19) then x^* would be a solution of problem (5.16).*

Proof. Since x^* is a feasible point and also the objective function of problem (5.16) at x^* achieves the minimum value:

$$||\frac{1}{k}Ax^* - b|| = 0$$

then x^* is the optimal solution.

□

Proposition 28. *If $m \geq d$ and $rank(A) = d$ and there exists x^* such that $\bar{A}x^* = b$ then*

$$x^* = x_{LS} = C^{-1}\bar{A}^T b$$

where $C = \bar{A}^T\bar{A} \in \mathbb{R}^{d\times d}$ and $rank(C) = d$.

Proof. Since

$$\bar{A}x^* = b$$

We have

$$\begin{aligned} \bar{A}^T A x^* &= \bar{A}^T b \\ \Rightarrow C^{-1} C x^* &= C^{-1} \bar{A}^T b \\ \Rightarrow x^* &= C^{-1} \bar{A}^T b. \end{aligned} \tag{5.20}$$

□

Proposition 29. *If $m \geq d, b \in Range(A)$ and $rank(A) = d$ and $x_{LS} = C^{-1}\bar{A}^T b \succeq 0$ then x_{LS} is the solution of the optimization problem (5.16).*

Proof. Obviously, since $x_{LS} = C^{-1}\bar{A}^T b$ is a feasible point and also the objective function of problem (5.16) at x_{LS} achieves the minimum value:

$$||\frac{1}{k} A x_{LS} - b|| = 0$$

then x_{LS} is the optimal solution. □

Suppose x^* is the optimal solution of problem (5.16) and

$$||\frac{1}{k} A x^* - b|| > 0$$

meaning that the optimal policy x^* can not achieve exactly the target distribution b. With the assumption that there exists a solution for $\bar{A}x = b$, we can find a sub-optimal policy x' to achieve exactly the target distribution b (with a smaller number of transmission bits per round compared to k bits using policy x^*). The procedure to find a such policy is described in the Relaxation Algorithm as follows.

Here are some notations that will be used in the algorithm:

- Denote a_i as the column vector i in matrix A, we can see that each a_i corresponds to a subset $\mathcal{V}_i$ and also there are k one entries in a_i: $\sum_{j=1}^{m} a_i(j) = k$.

- Denote $e_i \in [0,1]^{m\times 1}$ as the unit vector

$$e_i(j) = \begin{cases} 1 \text{ where } i = j \\ 0 \text{ otherwise.} \end{cases}$$

Proposition 30. *If $d = \binom{m}{k}$ then the system of linear equations $\bar{A}x = b$ exists a solution.*

Proof. See Appendix. □

Note that: Proposition 30 only states one special condition to guarantee the existence of solution to the equation $\bar{A}x = b$ which is an important requirement that we can apply the Relaxation Algorithm. In fact, the equation $\bar{A}x = b$ can still have solution in other cases as well.

Algorithm 5: Relaxation Algorithm

1. Find a solution of $\bar{A}x = b$, denote as x. Let $\mathcal{N} = \{x_i : x_i < 0\}$ be the subset of negative entries and $\mathcal{P} = \{x_i : x_i \geq 0\}$ be the subset of positive entries.
2. Find the smallest subsets $\mathcal{S}$ of $\mathcal{P}$ such that:

$$c = \sum_{x_i \in \mathcal{N}} x_i a_i + \sum_{x_i \in \mathcal{S}} x_i a_i \geq 0$$

3. Decompose c by using the set of unit vectors $\{e_1, e_2, \ldots, e_m\}$

$$c = \sum_{i=1}^{m} y_i e_i$$

   where $y_i \geq 0 \ \forall i$

4. Formulate the new policy by using y_i and $x_i \in \mathcal{P} \setminus \mathcal{S}$ such that ky_i is the probability that e_i is used (the systems only serve the service R_i) and $x_i \in \mathcal{P} \setminus \mathcal{S}$ is the probability that subset $\mathcal{V}_i$ is used in a round.

Proposition 31. *The policy formulated by the Relaxation Algorithm would achieve exactly the target rate distribution b but the average transmission bits in each round would decrease to a value that is smaller than k.*

Proof. See Appendix. □

Example

Topology matrix where $m = 4, n = 3$.

$$H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

Then

$$A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

where $rank(A) = 3$. Suppose

$$b = \begin{bmatrix} 1/3 & 2/15 & 4/15 & 4/15 \end{bmatrix}^T$$

Hence,

$$x_{LS} = C^{-1}\bar{A}^T b = \begin{bmatrix} 1/5 & 1/5 & 3/5 \end{bmatrix}^T \succeq 0$$

Then x_{LS} is the solution. However, if

$$b = \begin{bmatrix} 1/3 & 2/15 & 2/15 & 2/5 \end{bmatrix}^T$$

Then

$$x_{LS} = C^{-1}A^T b = \begin{bmatrix} -1/3 & 3/5 & 3/5 \end{bmatrix}^T \not\succeq 0$$

Hence, we need to solve the optimization problem. Now, we use convex solvers such as CVX [45] and the solution is

$$x^* = \begin{bmatrix} 0 & 1/2 & 1/2 \end{bmatrix}^T$$

which produce $||Ax - b|| \approx 0.0816$.

Since x^* can not achieve exactly target rate distribution. We can use the above procedure to find policy x' from x_{LS}. We have:

$$y = \begin{bmatrix} 1/3 & 2/15 & 2/15 & 2/5 \end{bmatrix}$$

The policy x' using y_i as the probability that e_i is used would achieve exactly the target rate distribution. However, the average rate of policy x' in this case is 1 bit per round instead of 3 bits per round (when using policy x^*).

5.7 Conclusions

In this work, we briefly introduce WiFO, a hybrid WiFi-FSO network for Gbps wireless local area network (WLAN) femtocells that can provide up to one Gbps per user while maintaining seamless mobility. While typical RF femtocells are non-overlapped to minimize inter-cell interference, there are advantages of using overlapped femtocells to increase mobility and throughput when the number of users is small. We present LAC, a novel coding technique used in the WiFO network that aims to increase bandwidth and reduce interference for multiple users in a dense array of femtocells. Both theoretical analysis and numerical experiments show orders of magnitude increase in throughput using LAC over the basic code. In addition, we introduce Time Minimization and Rate Allocation problem. Algorithms and solutions are also presented to verify the robustness of LAC technique in practical scenario.

Chapter 6: Conclusion and Future Work

6.1 Conclusion

In the thesis, we study random matrices and their applications to network coding techniques. In the scope of this work, network coding techniques are applied and investigated in three different scenarios:

- **Data synchronization problem.** The problem of efficient data synchronization for a large number of nodes with disparate data are introduced. We propose two probabilistic models on how the initial fractions of packets at receivers are distributed. These models arise naturally in many large scale systems such as Peer-to-Peer networks, data centers, and distributed storage systems. Based on these models, we establish probabilistic bounds and asymptotic results on the minimum number of transmission to complete the data synchronization process. Next, we propose and analyze a number of random network coding algorithms and verify their performances via theoretical analysis and simulations.

- **Data recovery problem.** The problem of information recovery in network coding systems is introduced. This problem arises when there is a node of the system under malicious attack. We show that the security can be improved using Minimum Rank Decoding Problem. In the minimum rank decoding problem, the goal is to recover the network coded packets from a malicious attacker who randomly corrupts the header of the packets with limited magnitude errors. We cast this problem as the problem of rank recovery of random matrices over finite field in presence of noise. We present some initial asymptotic results on joint distribution of weight and rank of random matrices for simple models which are useful for the rank recovery problem. We show that limited magnitude noise is likely not to decrease the rank of low-rank matrices with uniformly distributed weights.

- **Data transmission problem in WiFO systems.** The WiFO system is introduced as a hybrid WiFi-FSO network for Gbps wireless local area network (WLAN)

femtocells that can provide up to one Gbps per user while maintaining seamless mobility. While typical RF femtocells are non-overlapped to minimize inter-cell interference, there are advantages of using overlapped femtocells to increase mobility and throughput when the number of uses are small. We present LAC, an instance of network coding technique used in WiFO network network that aims to increase bandwidth and reduce interference for multiple users in a dense array of femtocells. Both theoretical analysis and numerical experiments show orders of magnitude increase in throughput using LAC over the basic code.

6.2 Future work

Network coding techniques and theory of random matrices will still be an attractive topic for the research community. In this section, we would like to discuss a few research directions that can be studied in the future:

- **Random matrices** While some results on the relation between rank and weight of random matrices are introduced in our work, the research for this relation are far from completion. The number of matrices with certain rank and certain weight or the distribution of matrix with given rank and weight still remains an intractable problem. Solving this problem will boost up a number of random matrices applications.

- **Min rank decoding problem** The min rank optimization problem in real fields can be solved efficiently by several heuristic approaches. However, these approaches do not perform well in the finite fields. Algorithms and solutions to the min rank decoding problem therefore will be an interesting research problem.

- **Network coding** It is shown that the LAC coding can achieve the capacity of 1 bit per user which has not been proved to be the maximum capacity of WiFO systems. Hence, we would like to extend and advance LAC coding or find other coding techniques to increase the capacity.

Bibliography

[1] A. Abdulhussein, A. Oka, Trung Thanh Nguyen, and L. Lampe. Rateless coding for hybrid free-space optical and radio-frequency communication. *Wireless Communications, IEEE Transactions on*, 9(3):907–913, March 2010.

[2] F. Ahdi and S.S. Subramaniam. Optimal placement of fso links in hybrid wireless optical networks. In *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, pages 1–6, Dec 2011.

[3] Rudolf Ahlswede, Ning Cai, Shuo-Yen Robert Li, and Raymond W Yeung. Network information flow. *Information Theory, IEEE Transactions on*, 46(4):1204–1216, 2000.

[4] Anwar Al Hamra, Chadi Barakat, and Thierry Turletti. Network coding for wireless mesh networks: A case study. In *Proceedings of the 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks*, pages 103–114. IEEE Computer Society, 2006.

[5] Barry C. Arnold and Richard A. Groeneveld. Bounds on expectations of linear systematic statistics based on dependent samples. *The Annals of Statistics*, 7(1):220–223, 01 1979.

[6] Z. Bar-Yossef, Y. Birk, T. S. Jayram, and T. Kol. Index coding with side information. *Information Theory, IEEE Transactions on*, 57(3):1479–1494, March 2011.

[7] Ezio Biglieri, Robert Calderbank, Anthony Constantinides, Andrea Goldsmith, Arogyaswami Paulraj, and H Vincent Poor. *MIMO wireless communications*. Cambridge University Press, 2007.

[8] Anna Blasiak, Robert Kleinberg, and Eyal Lubetzky. Index coding via linear programming. *arXiv preprint arXiv:1004.1379*, 2010.

[9] Johannes Blomer, Richard Karp, and Emo Welzl. The rank of sparse random matrices over finite fields. *Random Structures and algorithms*, 10(4):407–420, 1997.

[10] S. Bloom and W. Hartley. *The last-mile solution: hybrid FSO radio*. AirFiber Inc., May 2002.

[11] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2009.

[12] Jian-Feng Cai, Emmanuel J Candès, and Zuowei Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[13] Giuseppe Caire and Shlomo Shamai. On the achievable throughput of a multiantenna gaussian broadcast channel. *Information Theory, IEEE Transactions on*, 49(7):1691–1706, 2003.

[14] Emmanuel J Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

[15] Vikram Chandrasekhar, Jeffrey G Andrews, and Alan Gatherer. Femtocell networks: a survey. *Communications Magazine, IEEE*, 46(9):59–67, 2008.

[16] Vikram Chandrasekhar, Jeffrey G Andrews, Tarik Muharemovict, Zukang Shen, and Alan Gatherer. Power control in two-tier femtocell networks. *Wireless Communications, IEEE Transactions on*, 8(8):4316–4328, 2009.

[17] M. A R Chaudhry and A. Sprintson. Efficient algorithms for index coding. In *INFOCOM Workshops 2008, IEEE*, pages 1–4, April 2008.

[18] Mohammad Asad R Chaudhry, Zakia Asad, Alex Sprintson, and Michael Langberg. On the complementary index coding problem. In *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on*, pages 244–248. IEEE, 2011.

[19] Kaikai Chi, Xiaohong Jiang, and Susumu Horiguchi. Network coding-based reliable multicast in wireless networks. *Computer Networks*, 54(11):1823–1836, 2010.

[20] Philip A Chou and Yunnan Wu. Network coding for the internet and wireless networks. *IEEE Signal Processing Magazine*, 24(5):77, 2007.

[21] Philip A Chou, Yunnan Wu, and Kamal Jain. Practical network coding. In *Proceedings of the annual Allerton conference on communication control and computing*, volume 41, pages 40–49. The University; 1998, 2003.

[22] Colin Cooper. On the distribution of rank of a random matrix over a finite field. *Random Structures and Algorithms*, 17(3-4):197–212, 2000.

[23] Max HM Costa. Writing on dirty paper (corresp.). *Information Theory, IEEE Transactions on*, 29(3):439–441, 1983.

[24] Thomas A. Courtade and Richard D. Wesel. Coded cooperative data exchange in multihop networks. *arXiv:1203.3445 [cs, math]*, March 2012. 00006.

[25] Thomas Cover. Broadcast channels. *Information Theory, IEEE Transactions on*, 18(1):2–14, 1972.

[26] Yong Cui, Hongyi Wang, and Xiuzhen Cheng. Wireless link scheduling for data center networks. In *Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication*, page 44. ACM, 2011.

[27] Yong Cui, Hongyi Wang, Xiuzhen Cheng, and Biao Chen. Wireless data center networking. *Wireless Communications, IEEE*, 18(6):46–53, December 2011.

[28] Herbert Aron David and Haikady Navada Nagaraja. *Order statistics*. Wiley Online Library, 1970.

[29] S. El Rouayheb, A. Sprintson, and C. Georghiades. On the index coding problem and its relation to network coding and matroid theory. *Information Theory, IEEE Transactions on*, 56(7):3187–3195, July 2010.

[30] Salim Y El Rouayheb, Mohammad Asad R Chaudhry, and Alex Sprintson. On the minimum number of transmissions in single-hop wireless coding networks. In *Information Theory Workshop, 2007. ITW'07. IEEE*, pages 120–125. IEEE, 2007.

[31] A. Eryilmaz, A. Ozdaglar, M. Medard, and Ebad Ahmed. On the delay and throughput gains of coding in unreliable networks. *Information Theory, IEEE Transactions on*, 54(12):5511–5524, Dec 2008.

[32] Shaun M Fallat and Leslie Hogben. The minimum rank of symmetric matrices described by a graph: a survey. *Linear Algebra and its Applications*, 426(2):558–582, 2007.

[33] Maryam Fazel. *Matrix rank minimization with applications*. PhD thesis, 2002.

[34] Maryam Fazel, Haitham Hindi, and S Boyd. Rank minimization and applications in system theory. In *American Control Conference, 2004. Proceedings of the 2004*, volume 4, pages 3273–3278. IEEE, 2004.

[35] Maryam Fazel, Haitham Hindi, and Stephen P Boyd. A rank minimization heuristic with application to minimum order system approximation. In *American Control Conference, 2001. Proceedings of the 2001*, volume 6, pages 4734–4739. IEEE, 2001.

[36] William Feller. *An Introduction to Probability Theory and Its Applications. Volume I.* John Wiley & Sons London-New York-Sydney-Toronto, 1968.

[37] Jörg Flum and Martin Grohe. Parameterized complexity theory, volume xiv of texts in theoretical computer science. an eatcs series, 2006.

[38] Gerard J Foschini and Michael J Gans. On limits of wireless communications in a fading environment when using multiple antennas. *Wireless personal communications*, 6(3):311–335, 1998.

[39] Ernest Mukhamedovich Gabidulin. Theory of codes with maximum rank distance. *Problemy Peredachi Informatsii*, 21(1):3–16, 1985.

[40] Ernst M Gabidulin, AV Paramonov, and OV Tretjakov. Ideals over a non-commutative ring and their application in cryptology. In *Advances in CryptologyEUROCRYPT91*, pages 482–489. Springer, 1991.

[41] Robert G Gallager. Capacity and coding for degraded broadcast channels. *Problemy Peredachi Informatsii*, 10(3):3–14, 1974.

[42] David Gesbert, Mansoor Shafi, Da-shan Shiu, Peter J Smith, and Ayman Naguib. From theory to practice: an overview of mimo space-time coded wireless systems. *Selected Areas in Communications, IEEE Journal on*, 21(3):281–302, 2003.

[43] Christos Gkantsidis and Pablo Rodriguez Rodriguez. Network coding for large scale content distribution. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 4, pages 2235–2245. IEEE, 2005.

[44] Andrea Goldsmith, Syed Ali Jafar, Nihar Jindal, and Sriram Vishwanath. Capacity limits of mimo channels. *Selected Areas in Communications, IEEE Journal on*, 21(5):684–702, 2003.

[45] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 1.21. `../../cvx`, April 2011.

[46] Daniel Halperin, Srikanth Kandula, Jitendra Padhye, Paramvir Bahl, and David Wetherall. Augmenting data center networks with multi-gigabit wireless links. In *ACM SIGCOMM Computer Communication Review*, volume 41, pages 38–49. ACM, 2011.

[47] Hennes Henniger and Otakar Wilfert. An introduction to free-space optical communications. *Radioengineering*, 19(2):203–212, 2010.

[48] Francis Begnaud Hildebrand. *Advanced calculus for applications*, volume 63. Prentice-Hall Englewood Cliffs, NJ, 1962.

[49] Tracey Ho, M. Medard, R. Koetter, D.R. Karger, M. Effros, Jun Shi, and B. Leong. A random linear network coding approach to multicast. *Information Theory, IEEE Transactions on*, 52(10):4413–4430, Oct 2006.

[50] Tracey Ho, Muriel Médard, Jun Shi, Michelle Effros, and David R Karger. On randomized network coding. In *Proceedings of the Annual Allerton Conference on Communication Control and Computing*, volume 41, pages 11–20. The University; 1998, 2003.

[51] Dorit S Hochbaum. *Approximation algorithms for NP-hard problems*. PWS Publishing Co., 1996.

[52] A. Kashyap and M. Shayman. Routing and traffic engineering in hybrid rf/fso networks. In *Communications, 2005. ICC 2005. 2005 IEEE International Conference on*, volume 5, pages 3427–3433 Vol. 5, May 2005.

[53] Sachin Katti, Dina Katabi, Hari Balakrishnan, and Muriel Medard. Symbol-level network coding for wireless mesh networks. In *Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication*, SIGCOMM '08, pages 401–412, New York, NY, USA, 2008. ACM.

[54] Sachin Katti, Hariharan Rahul, Wenjun Hu, Dina Katabi, Muriel Médard, and Jon Crowcroft. Xors in the air: practical wireless network coding. *IEEE/ACM Transactions on Networking (TON)*, 16(3):497–510, 2008.

[55] I. Kim and E. Korevaar. Availability of free space optics (fso) and hybrid fso/rf systems. *Proc. Optical Wireless Commun. IV*, Aug 2001.

[56] Young-Han Kim, Arak Sutivong, and Styrmir Sigurjonsson. Multiple user writing on dirty paper. In *Information Theory, 2004. ISIT 2004. Proceedings. International Symposium on*, page 534. IEEE, 2004.

[57] Ralf Koetter and Muriel Médard. An algebraic approach to network coding. *Networking, IEEE/ACM Transactions on*, 11(5):782–795, 2003.

[58] Ho Yuet Kwan, Kenneth W Shum, and Chi Wan Sung. Generation of innovative and sparse encoding vectors for broadcast systems with feedback. In *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on*, pages 1161–1165. IEEE, 2011.

[59] Michael Langberg and Alex Sprintson. On the hardness of approximating the network coding capacity. In *Information Theory, 2008. ISIT 2008. IEEE International Symposium on*, pages 315–319. IEEE, 2008.

[60] Charles L Lawson and Richard J Hanson. *Solving least squares problems*, volume 161. SIAM, 1974.

[61] N. Letzepis, K.D. Nguyen, A. Guillen i Fabregas, and W.G. Cowley. Outage analysis of the hybrid free-space optical and radio-frequency channel. *Selected Areas in Communications, IEEE Journal on*, 27(9):1709–1719, December 2009.

[62] S.-Y.R. Li, R.W. Yeung, and Ning Cai. Linear network coding. *Information Theory, IEEE Transactions on*, 49(2):371–381, Feb 2003.

[63] Shuo-Yen Robert Li, Raymond W Yeung, and Ning Cai. Linear network coding. *Information Theory, IEEE Transactions on*, 49(2):371–381, 2003.

[64] Rudolf Lidl and Harald Niederreiter. *Finite fields*, volume 20. Cambridge university press, 1997.

[65] Luısa Lima, Muriel Médard, and Joao Barros. Random linear network coding: A free cipher? In *Information Theory, 2007. ISIT 2007. IEEE International Symposium on*, pages 546–550. IEEE, 2007.

[66] Junning Liu, D. Goeckel, and D. Towsley. The throughput order of ad hoc networks employing network coding and broadcasting. In *Military Communications Conference, 2006. MILCOM 2006. IEEE*, pages 1–7, Oct 2006.

[67] David JC MacKay. Fountain codes. *IEE Proceedings-Communications*, 152(6):1062–1068, 2005.

[68] K. Marton. A coding theorem for the discrete memoryless broadcast channel. *Information Theory, IEEE Transactions on*, 25(3):306–311, May 1979.

[69] Mehran Mesbahi. On the rank minimization problem and its control applications. *Systems & control letters*, 33(1):31–36, 1998.

[70] Theresa Migler, Kent E Morrison, and Mitchell Ogle. Weight and rank of matrices over finite fields. *arXiv preprint math/0403314*, 2004.

[71] DL Neuhoff, RM Gray, LD Davisson, et al. A coding theorem for the discrete memoryless broadcast channel. *Trans. Inform. Theoty*, 21:511–528, 1975.

[72] Dong Nguyen, T. Tran, Thinh Nguyen, and B. Bose. Wireless broadcast using network coding. *Vehicular Technology, IEEE Transactions on*, 58(2):914–925, Feb 2009.

[73] Noam Nisan and Avi Wigderson. On rank vs. communication complexity. *Combinatorica*, 15(4):557–565, 1995.

[74] Ben Noble and James W Daniel. *Applied linear algebra*, volume 3. Prentice-Hall New Jersey, 1988.

[75] Christopher C Paige and Michael A Saunders. Solution of sparse indefinite systems of linear equations. *SIAM journal on numerical analysis*, 12(4):617–629, 1975.

[76] Arogyaswami J Paulraj, Dhananjay A Gore, Rohit U Nabar, and Helmut Bölcskei. An overview of mimo communications-a key to gigabit wireless. *Proceedings of the IEEE*, 92(2):198–218, 2004.

[77] René Peeters. Orthogonal representations over finite fields and the chromatic number of graphs. *Combinatorica*, 16(3):417–431, 1996.

[78] Lei Poo. Space-time coding for wireless communication: a survey. *Report from Standford University*, 2002.

[79] Benjamin Recht, Maryam Fazel, and Pablo A Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

[80] Ron M Roth. Maximum-rank array codes and their application to crisscross error correction. *Information Theory, IEEE Transactions on*, 37(2):328–336, 1991.

[81] Salim El Rouayheb, Alex Sprintson, and Parastoo Sadeghi. On coding for cooperative data exchange. *arXiv:1002.1465 [cs, math]*, February 2010. 00038.

[82] Yalin Evren Sagduyu and Anthony Ephremides. On joint mac and network coding in wireless ad hoc networks. *Information Theory, IEEE Transactions on*, 53(10):3697–3713, 2007.

[83] Louis L Scharf. *Statistical signal processing*, volume 98. Addison-Wesley Reading, MA, 1991.

[84] Suresh P Sethi and Gerald L Thompson. *What is Optimal Control Theory?* Springer, 2000.

[85] Danilo Silva, Frank R Kschischang, and Ralf Koetter. A rank-metric approach to error control in random network coding. *Information Theory, IEEE Transactions on*, 54(9):3951–3967, 2008.

[86] A. Sprintson, P. Sadeghi, G. Booker, and S. El Rouayheb. A randomized algorithm and performance bounds for coded cooperative data exchange. In *Information Theory Proceedings (ISIT), 2010 IEEE International Symposium on*, pages 1888–1892, June 2010.

[87] Alex Sprintson, Parastoo Sadeghi, Graham Booker, and Salim El Rouayheb. Deterministic algorithm for coded cooperative data exchange. In Xi Zhang and Daji Qiao, editors, *Quality, Reliability, Security and Robustness in Heterogeneous Networks*, number 74 in Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, pages 282–289. Springer Berlin Heidelberg, January 2012. 00011.

[88] Gilbert Strang and Wellesley-Cambridge Press. *Introduction to linear algebra*, volume 3. Wellesley-Cambridge Press Wellesley, MA, 1993.

[89] Yi Tang, M. Brandt-Pearce, and S.G. Wilson. Link adaptation for throughput optimization of parallel channels with application to hybrid fso/rf systems. *Communications, IEEE Transactions on*, 60(9):2723–2732, September 2012.

[90] Vahid Tarokh, Nambi Seshadri, and A Robert Calderbank. Space-time codes for high data rate wireless communication: Performance criterion and code construction. *Information Theory, IEEE Transactions on*, 44(2):744–765, 1998.

[91] Tuan Tran, Thinh Nguyen, Bella Bose, and Vinodh Gopal. A hybrid network coding technique for single-hop wireless networks. *Selected Areas in Communications, IEEE Journal on*, 27(5):685–698, 2009.

[92] Vijay V Vazirani. *Approximation algorithms*. Springer Science & Business Media, 2013.

[93] Di Wang and A.A. Abouzeid. Throughput capacity of hybrid radio-frequency and free-space-optical (rf/fso) multi-hop networks. In *Information Theory and Applications Workshop, 2007*, pages 3–10, Jan 2007.

[94] Qiwei Wang, Thinh Nguyen, and Alan X Wang. Channel capacity optimization for an integrated wi-fi and free-space optic communication system (wififo). In *Proceedings of the 17th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*, pages 327–330. ACM, 2014.

[95] John HENRY WILKINSON, Friedrich Ludwig Bauer, and C Reinsch. *Linear algebra*, volume 2. Springer, 2013.

[96] AIM Minimum Rank-Special Graphs Work et al. Zero forcing sets and the minimum rank of graphs. *Linear Algebra and its Applications*, 428(7):1628–1648, 2008.

[97] Haiping Wu, B. Hamzeh, and Mohsen Kavehrad. Achieving carrier class availability of fso link via a complementary rf link. In *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Eighth Asilomar Conference on*, volume 2, pages 1483–1487 Vol.2, Nov 2004.

[98] Muxi Yan and A. Sprintson. Algorithms for weakly secure data exchange. In *Network Coding (NetCod), 2013 International Symposium on*, pages 1–6, June 2013.

APPENDICES

Appendix A: Proofs of Propositions and Theorems

A.1 Proof of Proposition 7

Proof. Let $t_j^{(S)}$ be the random variable representing the number of time slots to collect the j-th linearly independent packet after $(j-1)$ linearly independent packets has been added to $\mathcal{H}_i$ (in addition to K linearly independent packets in $\mathcal{H}_i$ at initial). In $\mathcal{H}_i$, there are $(K+j-1)$ linearly independent packets so there are $(F^{K+j-1}-1)$ dependent vectors with $\mathcal{H}_i$ in total of (F^D-1) nonzero vectors in $GF(\mathcal{F}^D)$.

Let $p_j^{(S)}$ be the probability the j-th linearly independent packet is received at each time slot. We have:

$$p_j^{(S)} = 1 - \frac{F^{K+j-1}-1}{F^D-1} = \frac{F^D - F^{K+j-1}}{F^D-1}$$

Then $t_j^{(S)}$ has geometric distribution with expectation $\mathbf{E}[t_j^{(S)}] = \frac{1}{p_j^{(S)}}$ and variance $\mathbf{Var}[t_j^{(S)}] = \frac{1-p_j^{(S)}}{{p_j^{(S)}}^2}$. Since $\mathbf{H}_i$ needs exactly L new linearly independent packets to be full rank, the number of broadcasts $T_i^{(S)}$ that receiver R_i can recover all D original packets is equal the time it receives L-th new linearly independent packet:

$$\mathbf{E}[T_i^{(S)}] = \sum_{j=1}^{L} E[t_j^{(S)}] = \sum_{j=1}^{L} \frac{1}{p_j^{(S)}}$$

$$\rightarrow \mathbf{E}[T_i^{(S)}] = \sum_{j=1}^{L} \frac{F^D-1}{F^D - F^{K+j-1}} \tag{A.1}$$

Also, for the variance of $T_i^{(S)}$:

$$\mathbf{Var}[T_i^{(S)}] = \sum_{j=1}^{L} \mathbf{Var}[t_j^{(S)}] = \sum_{j=1}^{L} \frac{1-p_j^{(S)}}{{p_j^{(S)}}^2}$$

$$\rightarrow \mathbf{Var}[T_i^{(S)}] = \sum_{j=1}^{L} \frac{(F^{K+j-1}-1)(F^D-1)}{(F^D - F^{K+j-1})^2} \tag{A.2}$$

□

A.2 Proof of Proposition 8

Proof. Consider the behavior at receiver R_i, let $\mathcal{S}_i$ be the intersection (share) set between $\mathcal{H}_i$ and the union set $\mathcal{W}$ at the sender. We have

$$|\mathcal{S}_i| = |\mathcal{W} \cap \mathcal{H}_i| = |\mathcal{W}| + |\mathcal{H}_i| - |\mathcal{P}| = M + K - D = M - L.$$

We use the similar approach as in proof of Proposition 7 except that we randomly choose non-zero vectors in $\mathcal{W}$. Also in $\mathcal{H}_i$, there are $(M-L)$ packets that are linearly dependent with $\mathcal{W}$. Hence, the probability $p_j^{(I)}$ that the j-th linearly independent packet is received at R_i can be computed as follows.

$$p_j^{(I)} \quad = \quad 1 - \frac{F^{M-L+j-1}-1}{F^M-1} = \frac{F^M - F^{M-L+j-1}}{F^M-1}$$

Now, for the expectation and variance of $T_i^{(I)}$

$$\mathbf{E}[T_i^{(I)}] = \sum_{j=1}^{L} \frac{1}{p_j^{(I)}} = \sum_{j=1}^{L} \frac{F^M-1}{F^M - F^{M-L+j-1}}. \tag{A.3}$$

$$\mathbf{Var}[T_i^{(I)}] = \sum_{j=1}^{L} \mathbf{Var}[t_j^{(I)}] = \sum_{j=1}^{L} \frac{1-p_j^{(I)}}{{p_j^{(I)}}^2}$$

$$\rightarrow \mathbf{Var}[T_j^{(I)}] = \sum_{j=1}^{L} \frac{(F^{M-L+j-1}-1)(F^M-1)}{(F^M - F^{M-L+j-1})^2}. \tag{A.4}$$

□

A.3 Proof of Proposition 9

Proof. After each transmission, every receiver R_i recomputes its "Want" set $\mathcal{W}_i$, and then the sender recomputes $\mathcal{W}$, so the cardinality $M = |\mathcal{W}|$ will decrease by at least one. Let W_j be the updated union set at the sender after R_i receive the $(j-1)$-th linearly independent packets. We have $|\mathcal{W}_j| < |\mathcal{W}_{j-1}| < \cdots < |\mathcal{W}_1| = |\mathcal{W}|$ and $|\mathcal{W}_j| = M_j \leq M - (j-1) = M - j + 1$. Now, the intersection (share) set $\mathcal{S}_{i,j}$ between $\mathcal{H}_i$ and the union set $\mathcal{W}_j$ is $\mathcal{S}_{i,j}$. We have

$$\begin{aligned} |\mathcal{S}_{i,j}| &= |\mathcal{W}_j \cap \mathcal{H}_i| = |\mathcal{W}_j| + |\mathcal{W}_i| - |\mathcal{P}| \\ &= M_j + K - D = M_j - L. \end{aligned}$$

Then the probability $p_j^{(R)}$ such that the j-th new linearly independent packet is received can be computed as follows.

$$p_j^{(R)} = 1 - \frac{F^{M_j - L + j - 1} - 1}{F^{M_j} - 1} = \frac{F^{M_j} - F^{M_j - (L-j+1)}}{F^{M_j} - 1}$$

Consider the following function:

$$f(x) = \frac{F^x - F^{x-a}}{F^x - 1}$$

where $a = L - j + 1$ then $1 \leq a \leq L$. We have:

$$f'(x) = -\frac{(F^a - 1)\ln(F) F^{x-a}}{(F^x - 1)^2} \leq 0.$$

Hence, $f(x)$ is monotonically decreasing. Since $M_j \leq M - j + 1$, we have:

$$p_j^{(R)} = f(M_j) \geq f(M - j + 1) = \frac{F^{M-j+1} - F^{M-L}}{F^{M-j+1} - 1} \tag{A.5}$$

Therefore,

$$\mathbf{E}[T_i^{(R)}] = \sum_{j=1}^{L} \frac{1}{p_j^{(R)}} \leq \sum_{j=1}^{L} \frac{F^{M-j+1} - 1}{F^{M-j+1} - F^{M-L}} \tag{A.6}$$

For the variance of $T_i^{(I)}$, we have

$$\mathbf{Var}[T_i^{(R)}] = \sum_{j=1}^{L} \frac{1 - p_j^{(R)}}{{p_j^{(R)}}^2} \tag{A.7}$$

Consider the following function

$$g(x) = \frac{1-x}{x^2}$$

We have

$$g'(x) = \frac{x-2}{x^3} < 0$$

where $0 \leq x \leq 1$. Hence, $g(x)$ is a monotonically decreasing function in $0 \leq x \leq 1$. Combine with (A.5), we have:

$$\mathbf{Var}[T_i^{(R)}] = \sum_{j=1}^{L} g(p_j^{(R)}) \leq \sum_{j=1}^{L} \frac{(F^{M-j+1}-1)(F^{M-L}-1)}{(F^{M-j+1}-F^{M-L})^2}.$$

□

A.4 Proof of Proposition 11

Proof. The proof approaches are similar for all three algorithms. Here, the general notation T_{max} can be applied to each algorithm, respectively. We have

$$\mathbf{P}(T_{max} > a) = 1 - \mathbf{P}(T \leq a).$$

Also, $\mathbf{P}(T_{max} \leq a) = \mathbf{P}(\bigcap_{i=1}^{N} T_i \leq a)$.

Since $a > \mu$, let $a = b\sigma + \mu$ where $b > 0$ then

$$\begin{aligned} \mathbf{P}(T \leq a) &= \mathbf{P}(\bigcap_{i=1}^{N} T_i \leq \mu + b\sigma) \\ &= \mathbf{P}(\bigcap_{i=1}^{N} T_i - \mu \leq b\sigma) \\ &\geq \mathbf{P}(\bigcap_{i=1}^{N} |T_i - \mu| \leq b\sigma) \end{aligned}$$

Apply two-sided Chebyshev's inequality with N independent random variables $T_1, T_2, \ldots, T_N$:

$$\mathbf{P}(\bigcap_{i=1}^{N} |T_i - \mu| \leq b\sigma) \geq \prod_{i=1}^{N} (1 - \frac{1}{b^2}) = (1 - \frac{1}{b^2})^N$$

Note: the bound is only meaningful where $b \geq 1$.
Hence,

$$\mathbf{P}(T > a) \leq 1 - (1 - \frac{1}{b^2})^N$$

Plug $b = \frac{a-\mu}{\sigma}$ back, we have:

$$\mathbf{P}(T > a) \leq 1 - (1 - \frac{\sigma^2}{(a-\mu)^2})^N \tag{A.8}$$

□

A.5 Proof of theorem 13

Proof. Suppose row vectors of matrix X belongs to a given r-dimensional subspace W in F_q^n and these row vectors of X satisfy:

- There are $\epsilon' + 1$ bases of W where each base includes r linearly independent vectors F_q^n in W. Since we have the number of bases of any given subspace W is $A_q(r, r)$, then the number of choices is $A_q(r, r)^{\epsilon'+1}$ [70].

- The rest $(n - r(\epsilon' + 1))$ vectors of X also belong to W and can be zero vector. Hence, the number of choices for these vector is $q^{n-r(\epsilon'+1)}$.

Also, the number of matrices of rank r formulated by a given subspace W is $A_q(n,r)$.

We known that when ϵ' entries of matrix X are perturbed, there exists a basis of W that doesn't change. Hence the rank of matrix is at least r. Therefore, we can complete the proof. □

A.6 Proof of Corollary 14

Proof. We have:

$$\lim_{n\to\infty} \frac{A_q(n,r)}{q^{nr}} = \prod_{i=0}^{i=r-1} (\frac{q^n - q^i}{q^n}) \tag{A.9}$$

$$= \prod_{i=0}^{i=r-1} (1 - \frac{q^i}{q^n}) \tag{A.10}$$

$$= 1^r \tag{A.11}$$

$$= 1. \tag{A.12}$$

This completes the proof. □

A.7 Proof of Theorem 15

Proof. Suppose all the matrices have equal probability then due to [70], we have:

$$\mathbf{P}(R=r) = \frac{1}{q^m} \frac{\prod_{i=0}^{r-1} (q^n - q^i)^2}{\prod_{i=0}^{r-1} (q^r - q^i)}$$

Also, the expectation of weight with given rank [70]:

$$\mathbf{E}[W|R=r] = \frac{m(1-1/q)(1-1/q^r)}{(1-1/q^n)^2} \tag{A.13}$$

$$= \frac{m(q-1)(q^r-1)q^{2n-r-1}}{(q^n-1)^2} \tag{A.14}$$

Use Markov inequality, we have:

$$\begin{aligned} \mathbf{P}(W \geq w, R = r) &= P(W \geq w | R = r) P(R = r) && \text{(A.15)} \\ &\leq \frac{E[W|R=r]}{w} P(R = r) && \text{(A.16)} \end{aligned}$$

Plug in:

$$\mathbf{P}(W \geq w, R = r) \leq \frac{m(q-1)(q^r-1)q^{2n-r-1}}{w(q^n-1)^2} \frac{1}{q^m} \frac{\prod_{i=0}^{r-1}(q^n-q^i)^2}{\prod_{i=0}^{r-1}(q^r-q^i)}$$

Hence,

$$C(W \geq w, R = r) \leq \frac{m(q-1)(q^r-1)q^{2n-r-1}}{w(q^n-1)^2} \frac{\prod_{i=0}^{r-1}(q^n-q^i)^2}{\prod_{i=0}^{r-1}(q^r-q^i)}$$

□

A.8 Proof of Theorem 16

Proof. Let denote P as $\mathbf{P}(rank(Y) < rank(X_0)$, we have

$$\begin{aligned} P &= \mathbf{P}(rank(Y) < r_0 | w(Y) \in \{w_0 - \epsilon, w_0 + \epsilon\}) && \text{(A.17)} \\ &= \frac{\mathbf{P}(\{rank(Y) < r_0\} \cap \{w(Y) \in \{w_0 - \epsilon, w_0 + \epsilon\})}{\mathbf{P}(w(Y) \in \{w_0 - \epsilon, w_0 + \epsilon\})} && \text{(A.18)} \\ &= \frac{\sum_{w=w_0-\epsilon}^{w_0+\epsilon} \sum_{r=0}^{r_0-1} C(w,r)}{\sum_{w_0-\epsilon}^{w_0+\epsilon} C(w)} && \text{(A.19)} \\ &\leq \frac{\sum_{w=w_0-\epsilon}^{m} \sum_{r=0}^{r_0-1} C(w,r)}{\sum_{w_0-\epsilon}^{w_0+\epsilon} C(w)} && \text{(A.20)} \\ &= \frac{\sum_{r=0}^{r_0-1} C(W \geq w_0 - \epsilon, r)}{\sum_{w_0-\epsilon}^{w_0+\epsilon} C(w)} && \text{(A.21)} \\ &= \frac{\frac{m}{w_0-\epsilon} \sum_{r=0}^{r_0-1} \frac{2^{2n}}{2^{r+1}} \prod_{i=1}^{r-1} \frac{(2^n-2^i)^2}{(2^r-2^i)}}{\sum_{w=w_0-\epsilon}^{w_0+\epsilon} \binom{m}{w}} && \text{(A.22)} \end{aligned}$$

Consider function $f(x) = \frac{(2^n-2^x)^2}{(2^r-2^x)}$ where $n > r$ then we have $f'(x) > 0$ for $x \in (1, r-1)$. Then we have

$$\begin{aligned}
A &= \prod_{i=1}^{r-1} \frac{(2^n-2^i)^2}{(2^r-2^i)} && \text{(A.23)}\\
&\leq \left(\frac{(2^n-2^{r-1})^2}{(2^r-2^{r-1})}\right)^{r-1} && \text{(A.24)}\\
&= \left((2^n-2^{r-1})\left(\frac{2^n-2^{r-1}}{2^{r-1}}\right)\right)^{r-1} && \text{(A.25)}\\
&= \left((2^n-2^{r-1})(2^{n-r+1}-1)\right)^{r-1} && \text{(A.26)}\\
&\quad \text{(leave } 2^{r-1} \text{ and 1)} && \text{(A.27)}\\
&< (2^{(2n-r+1)})^{(r-1)} && \text{(A.28)}\\
&= 2^{2nr-r^2+r-2n+r-1} && \text{(A.29)}\\
&= 2^{2nr-2n-r^2+2r-1} && \text{(A.30)}
\end{aligned}$$

Consider:

$$\begin{aligned}
B &= \sum_{r=0}^{r_0-1} \frac{2^{2n}}{2^{r+1}} \prod_{i=1}^{r-1} \frac{(2^n-2^i)^2}{(2^r-2^i)} && \text{(A.31)}\\
&= \sum_{r=0}^{r_0-1} 2^{2nr-r^2+r-2} && \text{(A.32)}
\end{aligned}$$

Let $g(r) = 2nr - r^2 + r - 2$ then $g'(r) = 2n - 2r + 1 > 0$ for $r < n$, then:

$$\begin{aligned}
B &\leq r_0 2^{2n(r_0-1)-(r_0-1)^2+r_0-1-2} && \text{(A.33)}\\
&= r_0 2^{2nr_0-2n-r_0^2+3r_0-4} && \text{(A.34)}
\end{aligned}$$

Let $C = \sum_{w=w_0-\epsilon}^{w_0+\epsilon} \binom{m}{w}$, use the advanced Stirlings bound:

$$\frac{2^{mH(w/m)}}{m+1} \leq \binom{m}{w}$$

Recall that:

$$\theta = \begin{cases} \frac{w_0-\epsilon}{m} & \text{if } w_0 < m/2 \\ \frac{w_0+\epsilon}{m} & \text{if } w_0 \geq m/2 \end{cases}$$

Then

$$C \geq (2\epsilon+1)\frac{2^{mH(\theta)}}{m+1}$$

Now, we have:

$$\begin{aligned} P &\leq \frac{m}{w_0-\epsilon}\frac{r_0 2^{2nr_0-2n-r_0^2+3r_0-4}}{(2\epsilon+1)\frac{2^{mH(\theta)}}{m+1}} && \text{(A.35)} \\ &= \frac{m(m+1)r_0}{32(w_0-\epsilon)\epsilon}\frac{1}{2^{n^2H(\theta)-2nr_0+2n+r_0^2-3r_0}} && \text{(A.36)} \end{aligned}$$

(Use: $2\epsilon < 2\epsilon+1$).
Denote: $\alpha = H(\theta), \beta = \frac{r_0}{n}, \zeta = \frac{w_0}{n^2}, \eta = \frac{\epsilon}{n^2}$. Then $\alpha, \beta, \zeta, \eta \in (0,1)$. We have:

$$\begin{aligned} P &\leq \frac{1}{32(\zeta-\eta)\eta}\frac{\beta n^3(n^2+1)}{n^4 2^{\alpha n^2-2n\beta n+2n+\beta^2n^2-3\beta n}} && \text{(A.37)} \\ &= \frac{\beta}{32(\zeta-\eta)\eta}\frac{n(1+\frac{1}{n})}{2^{(\alpha+\beta^2-2\beta)n^2+(2-3\beta)n}} && \text{(A.38)} \end{aligned}$$

Denote $\lambda = \frac{\beta}{32(\zeta-\eta)\eta}, \gamma = \alpha+\beta^2-2\beta > 0, a = 2^\gamma > 1$. If then we have:

$$\begin{aligned} P &\xrightarrow{n\to\infty} \lambda\frac{n}{a^{n^2}} && \text{(A.39)} \\ &\xrightarrow{n\to\infty} 0 && \text{(A.40)} \end{aligned}$$

□

A.9 Proof of Corollary 17

Proof. Since $a > 1$ then $b > 0$, we have:

$$a^{n^2} = (1+b)^{n^2} > bn^2 \quad \text{(A.41)}$$

Then we have:

$$n \geq \frac{\lambda}{b\delta} \tag{A.42}$$

$$\rightarrow \frac{1}{bn} = \frac{n}{bn^2} \leq \frac{\delta}{\lambda} \tag{A.43}$$

$$\rightarrow \frac{n}{(1+b)^{n^2}} \leq \frac{\delta}{\lambda} \tag{A.44}$$

$$\rightarrow \lambda \frac{n}{a^{n^2}} \leq \delta \tag{A.45}$$

$$\rightarrow P = O(\lambda \frac{n}{a^{n^2}}) = O(\delta). \tag{A.46}$$

□

A.10 Proof of Theorem 18

Proof. X has w non-zero entries so there are at most w non-zero rows and w non-zero columns. If we can choose all ϵ entries out of these non-zero rows and columns, then matrix Y can only have greater rank than X. Hence, we have:

$$\begin{aligned}
P(rank(Y) \geq rank(X)) &\geq \frac{\binom{m-w^2}{\epsilon}}{\binom{m}{\epsilon}} && \text{(A.47)}\\
&= \frac{(m-w^2)!(m-\epsilon)!}{m!(m-w^2-\epsilon)!} && \text{(A.48)}\\
&= \frac{(m-w^2-\epsilon+1)\ldots(m-w^2)}{(m-\epsilon+1)\ldots m} && \text{(A.49)}\\
&= \prod_{i=0}^{i=\epsilon-1} \frac{m-w^2-i}{m-i} && \text{(A.50)}\\
&= \prod_{i=0}^{i=\epsilon-1} (1-\frac{w^2}{m-i}) && \text{(A.51)}\\
&\geq (1-\frac{w^2}{m-\epsilon+1})^{\epsilon} && \text{(A.52)}\\
& && \text{(A.53)}
\end{aligned}$$

where $m = n^2$. □

A.11 Proof of Theorem 19

Proof. For $i = 1, j = 1$:

Denote $M_1, M_2, \ldots, M_n$ as row vector of length n of maxtrix M. These vectors M_i need to satisfy:

- Have a non-zero vector, say M_k such that $w(M_k) > 1$.
- Have at least one vector M_l with $l \neq k$ such that $M_l = M_k$.
- Any other vectors M_i with $i \notin \{k, l\}$ can be either zero vector or equal M_k.

For a fixed M_k, the number of n ordered vectors satisfying the above properties is $2^n - n - 1$.

The possible choices for M_k is: $2^n - n - 1$. (Total number of length n vectors minus the number of vectors with weight of 0 and 1).

For $i = 2, j = 1$:

The vectors M_i need to satisfy:

- Have a pair of linearly independent vectors (M_1, M_2) as above.
- At least one vector $M_3 = M_1 + M_2$ or another pair $M_3 = M_1; M_4 = M_2$
- The others can be either $\underline{0}, M_1, M_2, (M_1 + M_2)$

Now, let compute the number of matrices that we can formulate:

- Let t_1 be the number of M_1
- Let t_2 be the number of M_2
- Let t_3 be the number of M_3
- Let t_4 be the number of $\underline{0}$

We require one of these two cases:

$$\begin{cases} t_1 \geq 1; t_2 \geq 1; t_3 \geq 1 \rightarrow \binom{n}{3} \\ t_1 \geq 2; t_2 \geq 2; t_3 = 0 \rightarrow \binom{n-2}{2} \end{cases}$$

and we have:

$$|\mathcal{X}_2^1| = \sum_{t_1+t_2+t_3+t_4=n} \frac{n!}{t_1!t_2!t_3!t_4!}$$

□

A.12 Proof of Theorem 20

Proof. In case $w = r = a$: we have a entries with different rows and different column. In total, there are $\binom{n}{a}$ possible choices of rows for a entries and another $\binom{n}{a}$ possible choices of columns. Also, to match row and column, there are $a!$ possible choices. Hence, take the product of them, we get the total number.

In case $w = a + 1; r = a$: we have the same possible of choices for the first a entries. Now, consider the last non-zero entry, it cannot have both different row and different column as the first a entries. Hence, we have two small case.

- First, its rows and columns are both belong to the set of rows and columns of the first a entries. Hence, we have $a^2 - a$ possible choices for this cases.
- Second, only rows or columns belong to the set of rows and columns of the first a entries and columns or rows, respectively would be different than the first a entries. Hence, we have $2(n-a)a$ possible choices. However, there would be overlapped to choose the last entry in this area. Since the last entry can replace the entry with the same row (or column) to form the first a entries. Hence, we need to divide the possible choices in this case by two.

In total, the number of matrices is

$$(a^2 - a + (n-a)a)a!\binom{n}{a}^2 = (n-1)a \times a!\binom{n}{a}^2$$

□

A.13 Proof of Proposition 22

Proof. Since the matrix H is of rank k in $\mathbf{GP}(q)$ and rows $u_1, u_2, \ldots, u_k$ are linearly independent, the other $m-k$ rows $v_1, v_2, \ldots, v_{m-k}$ could be represented as linear com-

binations of $u_1, u_2, \ldots, u_k$. In other words, for any row v_i, we have:

$$v_i = \sum_{q=1}^{k} c_s u_s, \tag{A.54}$$

where $c_s \in \{0, 1, 2, \ldots, q-1\}$ and at least one of the coefficient c_s's is different from 0 since H contains no row with all zero entries. Let denote that non-zero coefficient as $c_{s'}$. Now we just need to pick $u_{s'}$ to be replaced by v_i and still obtain a set of linearly independent rows. We will prove this by contradiction.

Indeed, suppose that u_1, u_2, ..., $u_{s'-1}$, v_i, $u_{s'+1}$, ..., u_k are not linearly independent. As a result, since u_1, u_2, ..., $u_{s'-1}$, $u_{s'+1}$, ..., u_k are linearly independent, v_i could be represented by a linear combination of u_1, u_2, ..., $u_{s'-1}$, $u_{s'+1}$, ..., u_k. In other words,

$$v_i = \sum_{s \neq s'} c'_s u_s. \tag{A.55}$$

From (A.54) and (A.55),

$$\begin{aligned} & \textstyle\sum_{s=1}^{k} c_s u_s = \sum_{s \neq s'} c'_s u_s, \\ \Leftrightarrow \quad & c_{s'} u_{s'} = \textstyle\sum_{s \neq s'} (c_s - c'_s) u_s, \end{aligned}$$

or $u_1, u_2, \ldots, u_k$ are linearly dependent (contradiction). □

A.14 Proof of Proposition 23

Proof. The average number of bits that the systems can transmit at a time (transmission rate) using LAC is computed as follows.

$$\mathbf{E}[R_{LAC}] = \mathbf{E}[rank(H)] = \sum_{k=1}^{n} \mathbf{P}(rank(H) = k)k$$

From [9], the expected number of linear dependencies of the rows of H in $\mathbf{GF}(q)$ is:

$$\mathbf{E}[l(H)] = \sum_{k=1}^{n} \binom{n}{k} \gamma^k (1-\gamma)^{n-k} [1 + (q-1)(1 - p/\gamma)^k]^n$$

where $\gamma = 1 - 1/q$ with $q = 2$ in this case. Since $l(H) = q^{n-rank(H)} - 1$ then one can compute the approximation of expected rank of H which is the average transmission rate (number of bits transmitted at a time) using LAC:

$$\mathbf{E}[rank(H)] \approx n - \log_q (\mathbf{E}[l(H)] + 1)$$

□

A.15 Proof of Proposition 24

Proof. First, for LAC, we can transfer k bits at a time if the topology matrix H has rank of k. Denote $C(k)$ as the number of $n \times n$ matrix of rank k in $\mathbf{GF}(2)$. According to [70], we have:

$$C(k) = \prod_{i=0}^{k-1} \frac{(2^n - 2^i)^2}{2^k - 2^i} \tag{A.56}$$

Also, the total number of $n \times n$ matrix in $\mathbf{GF}(2)$ is 2^{n^2} then:

$$\mathbf{P}(rank(H) = k) = \frac{C(k)}{2^{n^2}} = \frac{1}{2^{n^2}} \prod_{i=0}^{k-1} \frac{(2^n - 2^i)^2}{2^k - 2^i} \tag{A.57}$$

Therefore, the average rate (average number of bits transmitted at a time) for LAC is shown as follows.

$$\begin{aligned} \mathbf{E}[R_{LAC}] = \mathbf{E}[rank(H)] &= \sum_{k=1}^{n} k \times \mathbf{P}(rank(H = k)) \\ &= \frac{1}{2^{n^2}} \sum_{k=1}^{n} k \prod_{i=0}^{k-1} \frac{(2^n - 2^i)^2}{2^k - 2^i} \end{aligned}$$

Now, the probability that the matrix H is invertible [67] is:

$$\begin{aligned} \mathbf{P}(rank(H) = n) &= (1 - 2^{-n}) \dots (1 - 2^{-2})(1 - 2^{-1}) \\ &= \prod_{i=1}^{n} (1 - \frac{1}{2^i}) \end{aligned} \tag{A.58}$$

Interestingly, in [67], one can show that

$$\lim_{n\to\infty} \mathbf{P}(rank(H) = n) \approx 0.289 \tag{A.59}$$

□

A.16 Proof of Proposition 25

Proof. Now, we compute the average rate for BC. The system can transmit k bits if there are k receivers located in k non-overlapped regions and in each of these k regions there is only one receiver. As a result, the topology matrix H would have k "1" entries such that each entry of these k entries is the only non-zero entry in its row and its column. Denote $D(k)$ as the number of $n \times n$ matrices in $\mathbf{GF}(2)$ that have at least k "1" entries satisfying the condition. Hence, we have:

$$D(k) = k! \binom{n}{k}^2 2^{(n-k)^2} \tag{A.60}$$

for $k = 1, \ldots, n$ and $D(n+1) = 0$. Then, the number of matrices in $\mathbf{GF}(2)$ that have exactly k entries satisfying the condition is $D(k) - D(k+1)$. Therefore, the average rate could be computed as

$$\mathbf{E}[R_{BC}] = \frac{1}{2^{n^2}} \sum_{k=1}^{n} (D(k) - D(k+1))k = \frac{1}{2^{n^2}} \sum_{k=1}^{n} D(k) \tag{A.61}$$

□

A.17 Proof of Proposition 26

Proof.

$$\sum_i b_i = \sum_i (\bar{A}x)_i \tag{A.62}$$

$$= \frac{1}{k} \sum_i \sum_j x_j A_{ij} \tag{A.63}$$

$$= \frac{1}{k} \sum_j x_j \sum_i A_{ij} \tag{A.64}$$

$$= \sum_j x_j \tag{A.65}$$

and since

$$\sum_i b_i = 1$$

This completes the proof. □

A.18 Proof of Proposition 30

Proof. We will prove that from the set of column in matrix $\bar{A}$, we can formulate a vector space of rank d.

First, based on the set $\{a_1, a_2, \ldots, a_d\}$, we can form the set of vector $h_i = e_i - e_{i+1}$ $\forall i = 1, \ldots, d-1$. Now, formulate a matrix B from the set of $d-1$ vectors $\{h_1, h_2, \ldots, h_{d-1}\}$ and one vector a_l in the set $\{a_1, a_2, \ldots, a_d\}$ such that $a_l(m) = 1$ as the row vectors of B, we have:

$$B = \begin{bmatrix} 1 & -1 & 0 & \ldots & \ldots & 0 \\ 0 & 1 & -1 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \ldots & 0 & 1 & -1 & 0 \\ 0 & \ldots & \ldots & 0 & 1 & -1 \\ 1 & \ldots & 0 & \ldots & \ldots & 1 \end{bmatrix}$$

Using Gaussian elimination, all the entries of the last row can be zero out except for the

last entry $B_{d,d}$. Now we have the new matrix:

$$B' = \begin{bmatrix} 1 & -1 & 0 & \dots & \dots & 0 \\ 0 & 1 & -1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & 1 & -1 & 0 \\ 0 & \dots & \dots & 0 & 1 & -1 \\ 0 & \dots & \dots & \dots & 0 & 1 \end{bmatrix}$$

Since matrix B' is in the upper triangular form, we can see that $det(B) = 1$ so B' is a full rank matrix or $rank(B') = d$. On the other hand, matrix B' will have the same rank as matrix B since Gaussian elimination doesn't change the rank of matrix. Hence, $rank(B) = d$. Therefore, $rank(\bar{A}) = d$.

Since $d \geq m$ and $rank(\bar{A}) = m$, the systems of linear equations $\bar{A}x = b$ always exists a solution. □

A.19 Proof of Proposition 31

Proof. Since

$$c = \sum_{x_i \in \mathcal{N}} x_i a_i + \sum_{x_i \in \mathcal{S}} x_i a_i$$

and

$$c = \sum_{i=1}^{m} y_i e_i$$

Then

$$\sum_{i=1}^{m} y_i e_i + \sum_{x_i \in \mathcal{P} \setminus \mathcal{S}} x_i = b$$

Also the target rate distribution is achieved and the average transmission bits per round is:

$$\sum_{i=1}^{m} k y_i + k \sum_{x_i \in \mathcal{P} \setminus \mathcal{S}} x_i = k(\frac{1}{k}(\sum_{x_i \in \mathcal{N}} x_i + \sum_{x_i \in \mathcal{S}} x_i) + \sum_{x_i \in \mathcal{P} \setminus \mathcal{S}} x_i) < k$$

follows from

$$\sum_{x_i \in \mathcal{N}} x_i + \sum_{x_i \in \mathcal{S}} x_i + \sum_{x_i \in \mathcal{P} \setminus \mathcal{S}} x_i = 1$$

and

$$\sum_{i=1}^{m} y_i = \frac{1}{k}(\sum_{x_i \in \mathcal{N}} x_i + \sum_{x_i \in \mathcal{S}} x_i).$$

□