5-2013

Secret Sharing and Network Coding

Fiona Knoll
Clemson University, fknol309@gmail.com

Secret Sharing and Network Coding

A Masters Thesis
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science
Mathematical Sciences

---

by
Fiona May Knoll
May 2013

Accepted by:
Dr. Shuhong Gao, Committee Chair
Dr. Gretchen Matthews
Dr. Michael Burr
Dr. Neil Calkin

Abstract

In this thesis, we consider secret sharing schemes and network coding. Both of these fields are vital in today's age as secret sharing schemes are currently being implemented by government agencies and private companies, and as network coding is continuously being used for IP networks. We begin with a brief overview of linear codes. Next, we examine van Dijk's approach to realize an access structure using a linear secret sharing scheme; then we focus on a much simpler approach by Tang, Gao, and Chen. We show how this method can be used to find an optimal linear secret sharing scheme for an access structure with six participants. In the last chapter, we examine network coding and point out some similarities between secret sharing schemes and network coding. We present results from a paper by Silva and Kschischang; in particular, we present the concept of universal security and their coset coding scheme to achieve universal security.

Contents

Chapter 1

Linear Codes

1.1 Linear Codes

Codes are implemented in communication for security and error correction purposes. More specifically, private companies and federal agencies use codes to encode sensitive material in order to prevent the information from falling into the wrong hands. A branch of codes is linear codes. Linear codes provide a structural foundation for secret sharing and network coding. In this section, we briefly discuss linear codes to better understand secret sharing and network coding.

Let $\mathbb{F}_q$ be the field of q elements. An $[n, k]$ *linear code* over $\mathbb{F}_q$ is a linear subspace of $\mathbb{F}_q^n$ of dimension k and can be defined by a matrix $G \in \mathbb{F}_q^{k\times n}$ of rank k:

$$C = \{xG |\, x \in \mathbb{F}_q^k\}.$$

Such a matrix G is called a *generator matrix* for C. A *parity-check matrix* for C is any matrix $H \in \mathbb{F}_q^{(n-k)\times n}$ with rank $n - k$ so that $GH^\top = 0$, i.e.

$$c \in C \Leftrightarrow cH^\top = 0.$$

Hence, H can be used to determine whether or not a codeword is in C. The *dual code* $C^\top$

of C is defined as

$$C^\top = \{v \in \mathbb{F}_q^n |\, cv^\top = 0 \, \forall c \in C\} = \{uH^\top |\, u \in \mathbb{F}_q^{n-k}\}.$$

Thus, H is the generator matrix for the dual code. It can be easily shown that $(C^\perp)^\perp = C$.

Let $x, y \in \mathbb{F}_q^n$. Then the *Hamming distance* $d(x, y)$ between x and y is defined as the number of coordinates in which x and y differ. The *distance* of C is the minimum distance between two distinct codewords $x, y \in C$, i.e.

$$d(C) = \min_{\substack{x,y \in C \\ x \neq y}} d(x, y).$$

The well-known *Singleton bound* relates the quantities n, k, and d: $d \leq n - k + 1$. The codes that achieve this bound, i.e., $d(C) = n - k + 1$, are called *maximum distance separable (MDS) codes*. MDS codes are mainly used in error-correction. They maximize the distance between any two codewords, and thus, on receiving a codeword with slight perturbation, one may determine the correct codeword. An example of an MDS code is the Reed-Solomon code, which will be discussed in Section 1.3.

1.2 Shannon Entropy

We introduce the concept of Shannon entropy, which will be needed for the security analysis of secret sharing and network coding. Let S be a (discrete) random variable with probability distribution:

$$P(S = s_i) = q_i, \; 1 \leq i \leq t$$

where $s_1, s_2, \ldots, s_t$ are all the possible values of S. Then the Shannon entropy of S is defined as

$$H(S) = \sum_{i=1}^{t} q_i \log_2 \left(\frac{1}{q_i}\right) = -\sum_{i=1}^{t} q_i \log_2 q_i.$$

The entropy of S, $H(S)$, measures the uncertainty of S. When the probability distribution is uniform, the entropy of S is maximized. On the other hand, if there exists a q_i such that $q_i = 1$, then $H(S) = 0$, i.e., S is easily predictable. Let W be another random variable which takes values w_j, $1 \leq j \leq m$. The *conditional entropy* is defined as

$$H(S|W = w_j) = -\sum_i P(s_i|w_j) \log_2 P(s_i|w_j).$$

The proportion of unknown information of S when W is known is given by

$$H(S|W) = \sum_j P(w_j) H(S|w_j).$$

The *mutual information* of S and W is defined as

$$I(S, W) = H(S) - H(S|W).$$

If W contains information pertaining to S, then $H(S|W) < H(S)$ and $I(S, W) > 0$; otherwise, $H(S|W) = H(S)$ and $I(S, W) = 0$.

1.3 Reed-Solomon Codes

One of the most important coding schemes utilized in electronic media, such as CDs, DVDs, and QR codes, is the Reed-Solomon code, a cyclic error-correcting code, invented in 1960, by Irving Reed and Gustave Solomon.

A *Reed-Solomon code* C is an $[n, k]$ linear code over $\mathbb{F}_q$ where $k \leq n \leq q$. Let $a_1, a_2, \ldots, a_n \in \mathbb{F}_q$ be fixed distinct numbers belonging to the corresponding players, and let $s = (s_1, s_2, \ldots, s_k) \in \mathbb{F}_q^k$ be a message. To encode s:

1. Form the polynomial $p(x) = \sum_{i=1}^{k} s_i x^{i-1}$.

2. Compute $p(a_i)$, $1 \leq i \leq n$.

Then the codeword for S is $(p(a_1), p(a_2), \ldots, p(a_n))$, and the code is

$$C = \{(p(a_1), p(a_2), \ldots, p(a_n)) \mid p \in \mathbb{F}_q^k[x] \text{ of deg} \leq k-1\}.$$

The code C can be generated by the following Vandermonde matrix:

$$G = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ a_1 & a_2 & \ldots & a_n \\ a_1^2 & a_2^2 & \ldots & a_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ a_1^{k-1} & a_2^{k-1} & \ldots & a_n^{k-1} \end{bmatrix}.$$

Note that a polynomial of deg$< k$ over a field has at most $k-1$ zeros. The distance between any two codewords for C is at least $d = n - (k-1) = n - k + 1$. Also, there is a polynomial of deg$< k$ (e.g., $p(x) = \Pi_{i=1}^{k-1}(x - a_i)$) that has exactly $k-1$ zeros. Therefore, the minimum distance of C is $d = n - k + 1$. So, Reed-Solomon codes are MDS codes.

Chapter 2

Secret Sharing Schemes

2.1 Introduction

Secret sharing consists of a set of players, a dealer, and a distribution scheme. In 1979, Blakely [3] and Shamir [26] independently proposed a secret sharing scheme (SSS) where n participants each receive a share and any m or more of them can together reconstruct the secret with their appointed shares. The construction prohibits fewer than the designated number m to obtain the secret. Such a scheme is called an (m, n)-threshold scheme. More generally, the desired result of secret sharing schemes is that the authorized subsets of players, called an access structure, are the only groups to be able to reveal the secret. Further, using the shares of an unauthorized group will not disclose the secret, or any information pertaining to the secret. The objectives of an SSS are: 1) any subset of the access structure may obtain the message, and 2) any subset in the unauthorized set may not retrieve any information concerning the message. Following in Blakely's and Shamir's footsteps, other mathematicians have developed schemes of their own and have also pointed out the correlation between secret sharing schemes and coding. One such correlation, discovered by Massey, is the construction of secret sharing schemes employing linear error correcting codes.

Massey utilized linear codes for secret sharing schemes and pointed out the relation-

ship between the access structure and the codewords. Two questions then naturally arise: which access structures can be realized by linear codes, and how does one construct a code that realizes an access structure? In his paper, van Dijk [37] proposed a few solutions to these problems by focusing his attention on linear codes. As a response to van Dijk's approach, other means of characterizing the access structure realized by linear codes entered the picture.

The argument presented in this thesis is that there exists a linear code for a given access structure if and only if the system of quadratic equations $GH^\top = 0$ is consistent, where the matrices G and H are constructed from the access and adversary structures. As we shall see, there exists a monotone span program computing the access structure if and only if the system of quadratic equations has a solution. Utilizing these techniques, we discuss characteristics of such an access structure and briefly discuss the upper and lower bounds of the number of shares required.

2.2 Background

Recall that a secret sharing scheme consists of a dealer, a set of participants, and a distribution scheme so that certain subsets of the set of participants, called *authorized sets*, can reconstruct the secret, but any other subsets of participants, called *unauthorized sets*, cannot reconstruct the secret. The authorized sets form an *access structure*. More precisely, let $P = \{1, 2, \ldots, n\}$ be the set of participants. An *access structure* Γ on P is a collection of subsets of P that is monotone increasing, i.e., every subset B of P that contains a subset $A \in \Gamma$ must be in Γ. The complement of Γ, denoted by $\mathcal{R}$, is called the *adversary structure*, i.e., the union of Γ and $\mathcal{R}$ consists of all of the subsets of P. Then $\mathcal{R}$ is a collection of all the unauthorized sets that is monotone decreasing. (Monotone decreasing means that if $A \subset B$ and $B \in \mathcal{R}$, then $A \in \mathcal{R}$.)

Example 1 *Let* $P = \{1, 2, 3, 4\}$ *and*

$$\Gamma = \{(1,2), (1,3), (1,4), (1,2,3), (1,2,4), (1,3,4), (2,3,4), (1,2,3,4)\}.$$

Then the complement of Γ *is* $\mathcal{R} = \{(1), (2), (3), (4), (2,3), (2,4), (3,4)\}$. *One may note that* Γ *is monotone increasing and* $\mathcal{R}$ *is monotone decreasing. For example,* $(1,2) \in \Gamma$ *implies* $(1,2,3) \in \Gamma$ *and* $(2,3) \in \mathcal{R}$ *implies* $(2) \in \mathcal{R}$.

The access structure Γ can be completely determined by the minimal subsets, i.e., the subsets that no longer have access to the secret s if any member is removed from the group. We define

$$\Gamma^- = \{X |\, X \text{ is minimal in } \Gamma\}.$$

Similarly, $\mathcal{R}$ can be completely determined by the maximal subsets, i.e., the subsets Y that can reconstruct the secret when any player $i \notin Y$ is added to the group. We define

$$\mathcal{R}^+ = \{Y |\, Y \text{ is maximal in } \mathcal{R}\}.$$

For Example 1 above,

$$\Gamma^- = \{(1,2), (1,3), (1,4), (2,3,4)\} \text{ and}$$

$$\mathcal{R}^+ = \{(1), (2,3), (2,4), (3,4)\}.$$

Let Γ be any access structure on $P = \{1, 2, \ldots, n\}$. Let S be a space of the messages to be shared and S_i be the space of shares for participant i, $1 \leq i \leq n$. Also, let V be a finite set used for randomization purposes. Let τ be any map

$$\tau : S \times V \to S_1 \times S_2 \times \cdots \times S_n,$$

called a *distribution scheme*. A dealer can then use τ to distribute a message $s \in S$ as follows:

1. Pick a random vector $a \in V$.
2. Compute $\tau(s, a) = (k_1, k_2, \ldots, k_n)$.
3. Securely distribute the share k_i to player i, $1 \leq i \leq n$.

We use Shannon's entropy to define security for a distribution scheme. For this purpose, we view s, a, and k_i as random variables. More precisely, assume the message space S has an arbitrary probabilistic distribution, not necessarily uniform, so s can be viewed as a random variable on S under this distribution. In practice, the dealer picks a vector $a \in V$ according to a uniform random distribution independent of s. Through τ, this induces a random variable $k = (k_1, k_2, \ldots, k_n)$ on $S_1 \times S_2 \times \cdots \times S_n$. A distribution map τ gives a *secret sharing scheme* (SSS) for Γ if the following conditions are satisfied:

1. *Correctness* (zero-error communication): $H(s|k_X) = 0$ if $X \in \Gamma$, and
2. *Privacy*: $H(s|k_X) > 0$ if $X \notin \Gamma$,

where $k_X = (k_i)_{i \in X}$ denotes the collection of shares belonging to the group X. In the first condition, $H(s|k_X) = 0$ implies that the group X can determine the secret s using only the shares k_i, $i \in X$. The second condition means that only using the shares k_i, $i \in X$, one cannot completely determine s (no matter how much computing power one may have), even though one may obtain some partial information on s. A more specific type of privacy is perfect secrecy: for *perfect secrecy*, $H(s|k_X) = H(s)$ for all $s \in S$ and for $X \notin \Gamma$, i.e., the mutual information $I(s, k_X) = 0$. In other words, the group X *cannot obtain any information* pertaining to the secret. A secret sharing scheme is called a *perfect secret sharing scheme* (PSSS) if it achieves perfect secrecy: $H(s|k_X) = H(s)$ for all $X \notin \Gamma$, i.e., the participants in an unauthorized set X cannot get any information on s.

In practice, we take S, S_i and V to be linear spaces over $\mathbb{F}_q$, and τ to be a linear map. In this case, τ can be described more explicitly. Specifically, suppose $S = \mathbb{F}_q^k$, $V = \mathbb{F}_q^{\ell-k}$,

and $S_i = \mathbb{F}_q^{p_i}$, where p_i is the number of shares given to player i, $1 \leq i \leq n$ and $\ell \geq k$. For $1 \leq i \leq n$, let G_i be an $\ell \times p_i$ matrix over $\mathbb{F}_q$ of player i, which can be made public. We represent τ by a block matrix $G = (G_1, G_2, \ldots, G_n)$ so that with a message s and a randomly chosen vector a, the dealer computes

$$\tau(s, a) = (s, a)G = (k_1, k_2, \ldots, k_n)$$

and then distributes k_i secretly to player i. In this scenario, the SSS is called a *linear secret sharing scheme* (LSSS).

This scheme can be viewed in the context of linear codes. In fact, $G = [G_1, G_2, \ldots, G_n]$ is an $\ell \times N$ matrix over $\mathbb{F}_q$ where $N = \sum_{i=1}^{n} p_i$ and G generates a linear code C over $\mathbb{F}_q$, i.e.,

$$C = \{xG |\, x \in \mathbb{F}_q^{\ell}\} \subseteq \mathbb{F}_q^{N},$$

where each player i receives the p_i corresponding coordinate(s) of a codeword $c \in C$. While not the focus of this thesis, we direct the reader to an interesting use of algebraic geometric codes for SSS that achieves a low number of distributed shares found in [5].

A desirable goal when constructing a secret sharing scheme is to minimize the number of distributed shares. More concretely, an *ideal secret sharing scheme* occurs when $|S| = |S_i|$ for $1 \leq i \leq n$, i.e., each player receives the same number of shares as the length of the message s. However, not all access structures can be realized by an ideal scheme; therefore, we would like to minimize the number of shares distributed. A concept closely tied to the number of distributed shares is the information rate. If the number of shares given to each player is equal, the information rate of an SSS is defined as

$$\rho = \frac{\log_2 |S|}{\log_2 |S_i|}.$$

However, if one player receives more shares than another, then the information rate is defined as

$$\rho = \frac{\log_2 |S|}{\max_{i \in P}(\log_2 |S_i|)},$$

i.e., ρ represents the inbalance in the size of the message and the sizes of the distributed shares. Notice that in a perfect secret sharing scheme, $\rho \leq 1$, and in the ideal SSS, $\rho = 1$. Hence, we would like to create a scheme realizing Γ with information rate ρ as close to 1 as possible.

An important question then arises: given any access structure Γ, how does one find an LSSS that realizes Γ with the optimal information rate, i.e., a scheme which minimizes the number of shares distributed?

2.3 Explicit Constructions

In order to answer this question in depth, we need to better understand the concept of linear secret sharing schemes. We first examine established constructions of LSSSs. The first and foremost is Shamir's SSS, which is based on a Reed-Solomon code.

2.3.1 Threshold schemes via Reed-Solomon Codes

In 1979, Shamir [26] constructed one of the most famous LSSSs. In Shamir's scheme for n players, any set of m players makes up an authorized subset. To use his scheme:

1. Let $c_i \in \mathbb{F}_q$ represent a publicly known number of player i, $1 \leq i \leq n$, where each c_i is distinct.

2. Let $s \in S = \mathbb{F}_q$ be any message, where q exceeds the number of players n.

3. Randomly choose $m-1$ elements $a_1, a_2, \ldots, a_{m-1} \in \mathbb{F}_q$ where $m < n$.

4. Construct a degree $(m-1)$ polynomial $f(x) \in \mathbb{F}_q[x]$ such that

$$f(x) = s + a_1 x + a_2 x^2 + \ldots + a_{m-1} x^{m-1}.$$

5. Compute the shares $f(c_i)$, $1 \leq i \leq n$.

6. Distribute the share $f(c_i)$ to participant i, $1 \leq i \leq n$.

Through the process of Lagrange interpolation, any m holders of these shares may reconstruct $f(x)$ and thus find $s = f(0)$. This scheme is an (m, n)-*threshold scheme*: the dealer creates n shares from the secret and only m of them are needed to retrieve the message.

Shamir's scheme only permits one share per participant, making it an ideal SSS. Nonetheless, this limits the number of access structures the scheme can realize. Ito et al. [11] constructed the multiple assignment (m, n)-threshold, which allows more shares per participant. Let $P = \{1, 2, \ldots, t\}$ and $G_i = \{x_1, x_2, \ldots, x_{p_i}\}$ be the publicly known vector of participant i, where p_i is the number of shares participant i acquires. In addition, let $f(x)$ be a constructed polynomial as described in Shamir's scheme. Then each player i receives a set of shares:

$$k_i = \{f(x_1), f(x_2), \ldots, f(x_{p_i})\},$$

where $x_j \in G_i$ for $1 \leq j \leq p_i$, $1 \leq i \leq t$, and $\sum_{i=1}^{t} p_i = n$.

It follows that a group $X \subseteq P = \{1, 2, \ldots, t\}$ may reconstruct the secret if and only if $\sum_{i \in X} p_i \geq m$. An access structure may be defined as:

$$\Gamma = \{X \subseteq P : \sum_{i \in X} p_i \geq m\}.$$

Example 2 *Let* $P = \{1, 2, 3, 4\}$ *and let* $\Gamma^- = \{(1, 2), (1, 3), (1, 4), (2, 3, 4)\}$. *Then it follows that on giving player 1 two shares and the others only one share, the (3,5)-threshold scheme computes the desired authorized subsets.*

However, neither of these schemes can realize an arbitrary access structure; thus, we turn to a more general construction.

2.3.2 Schemes for General Access Structures

Ito et al. [11] constructed an extremely general (though impractical) secret sharing scheme which realizes any given access structure Γ^-. Let $P = \{1, 2, \ldots, n\}$ and Γ^- be any access structure. Let the message $s \in S = \mathbb{F}_q$. Then the dealer does the following:

1. For each $X \in \Gamma^-$, the dealer randomly and independently picks $|X|$ shares $r(X, i) \in \mathbb{F}_q$ for $i \in X$, so that
$$\sum_{i \in X} r(X, i) = s,$$
(Note: only $|X| - 1$ of the shares are chosen randomly with the last share being determined by the above relation).

2. The dealer distributes the vector $(r(X, i))_{X \in \Gamma}$ to participant i, $1 \leq i \leq n$.

For any authorized set $X \subseteq P$, the participants in X can reconstruct the secret as follows:

$$s = \sum_{i \in X} r(X, i).$$

Notice that an unauthorized set is just a subset of an authorized set. As a result, the shares belonging to an unauthorized set are randomly and independently chosen. So, one can show that, for any $X \in \mathcal{R}$, the unauthorized set will not be able to reconstruct the secret s nor glean any information pertaining to s.

Though this scheme can compute any given access structure, the number of shares required is large. For example, in the (m, n)-threshold scheme, the number of shares for i using the described scheme for Γ^- is

$$\begin{pmatrix} n-1 \\ m-1 \end{pmatrix},$$

which is exponential in n when $m \approx n/2$. In comparison, using linear codes in Shamir's scheme, each participant i receives only one share. Thus, the question still remains: how

does one construct an LSSS that minimizes the number of distributed shares? van Dijk attempts to answer this question by giving a method for finding a matrix G that realizes Γ, i.e., a matrix that defines a SSS for Γ.

2.4 van Dijk's Approach

Before we look at van Dijk's scheme, we need to first revisit the structure of an LSSS. Let $S = \mathbb{F}_q^k$ be the message space, $V = \mathbb{F}_q^{\ell-k}$ be a finite set, and $S_i = \mathbb{F}_q^{p_i}$ be the share space for participant i, $1 \leq i \leq n$, where $\ell \geq k$. Recall the linear map

$$\tau : S \times V \to S_1 \times S_2 \times \ldots \times S_n$$

that transforms messages to shares. We can represent τ as a matrix form by a block matrix $G = [G_1, G_2, \ldots, G_n]$ so that for message $s \in S$ and vectors $a \in V$, the shares for player i are given by $(s, a)G_i$ where G_i is a publicly known matrix of player $i \in P = \{1, 2, \ldots, n\}$. Let Γ^- be any access structure and $X \subseteq P$. Let

$$G_X = (G_{j_1}, G_{j_2}, \ldots, G_{j_m}) = \begin{bmatrix} G_X^{(1)} \\ G_X^{(2)} \end{bmatrix}$$

represent the concatenation of the publicly known matrices G_i for $i \in X$ where $G_X^{(1)} \in \mathbb{F}_q^{k \times p_{\{X\}}}$, $G_X^{(2)} \in \mathbb{F}_q^{(\ell-k) \times p_{\{X\}}}$, and $p_{\{X\}} = \sum_{i \in X} p_i$. (Note that $G_X^{(1)}$ has k rows, which is the size of the secret.)

Lemma 1 *([37], Theorem 1) The construction based on the matrices G_i, $i \in P$, defines a perfect secret sharing scheme for access structure Γ on P and space of possible secrets S if and only if*

1. *for every $X \in \Gamma$ there exists a $B \in \mathbb{F}_q^{p_{\{X\}} \times k}$ such that $(I_k, 0)^\top = G_X B$, and*
2. *for every $X \notin \Gamma$, rank$(G_X^{(2)})$ = rank(G_X).*

A set of matrices G_i that satisfies the above properties is called suitable.

Proof: The first direction is obvious due to the definiton of a PSSS. We will then move on to the converse. Let Γ^- be an access structure on a set P. Let $s \in S$ be a message to be distributed by the dealer and G_i be the publicly known matrices of players $i \in P$. Suppose the above two conditions hold. Then with the encoded shares, the process of revealing the secret is two-fold:

1. The authorized subset X finds the unique and obtainable matrix B such that

$$\begin{bmatrix} I_k \\ 0 \end{bmatrix} = G_X B.$$

2. With the acquired matrix B, the participants in X may compute

$$(s, a)G_X B = (s, a) \begin{bmatrix} I_k \\ 0 \end{bmatrix} = s$$

   to reveal the secret $s \in S$.

Consequently, $H(s|k_X) = 0$ and the scheme achieves correctness.

Another facet to examine is perfect secrecy. Suppose $X \notin \Gamma^-$. Then by the second condition, $\text{rank}(G_X^{(2)}) = \text{rank}(G_X)$ implies

$$G_X B = \begin{bmatrix} Q & 0 \\ D & 0 \end{bmatrix}$$

where D is of rank m for some $m \leq \ell - k$ and $Q \in \mathbb{F}_q^{k \times k}$. Hence,

$$(s, a)G_X B = sQ + aD$$

is a uniform random vector for any $s \in S$ since D has m linearly independent columns and the vector a is chosen randomly. Thus, perfect secrecy is attained. $\square$

Example 3 *Let* $P = \{1, 2, 3, 4\}$ *and the access structure* $\Gamma^- = \{(1, 2), (1, 3), (1, 4), (2, 3, 4)\}$. *Let* $S = (s_1, s_2)$ *and choose* $a = (a_1, a_2)$. *Then the shares are the following:*

1. $(s, a)G_1 = (a_1, a_2)$ *with* $p_1 = 2$
2. $(s, a)G_2 = (s_1 + a_2, s_2 + a_1)$ *with* $p_2 = 2$
3. $(s, a)G_3 = (s_2 + a_2, s_1 + a_1)$ *with* $p_3 = 2$
4. $(s, a)G_4 = (s_1 + a_1 + a_2, s_2 + 2a_2)$ *with* $p_4 = 2$

where $G = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 2 \end{bmatrix}$

It is easy to verify that for every subset of Γ^-, *the secret* $s = (s_1, s_2)$ *can be recovered. Also, since the length of the secret is two and each player receives two shares, the scheme is an ideal secret sharing scheme for* Γ^-.

As mentioned earlier, an LSSS can be described in terms of a code C. van Dijk uses this approach to introduce and prove his major theorem. Let C be a linear code over $\mathbb{F}_q$ and let the parity-check matrix for C be

$$H = \left(\begin{array}{c|c} I_k & G \\ 0 & \end{array}\right) \in \mathbb{F}_q^{\ell \times (k+N)}, \tag{2.1}$$

where $G = [G_1, G_2, \ldots, G_n]$ is the concatenation of the publicly known matrices and $N = \sum_{i=1}^{n} p_i$ is total number of shares distributed. The code C can then be defined as follows:

$$C = \{c \in \mathbb{F}_q^{(k+N)} \mid cH^\top = 0\}.$$

One can view the last N coordinates of a codeword $c \in C$ as a representation of an authorized subgroup $X \in \Gamma^-$ when using the support of c. With this thought in mind, we define the

support of c.

Definition 1 *([37]) Let $c = (c_1, c_2, \ldots, c_n) \in \mathbb{F}_q^N$ where $c_i \in \mathbb{F}_q^{p_i}$ represents the coordinates of the vector belonging to player i. The support of c, $Supp(c)$, is defined as the set of coordinates i, $1 \leq i \leq n$, for which $c_i \neq 0$, i.e.,*

$$Supp(c) = \{i : c_i \neq 0\}.$$

The support of the codeword c can then be viewed as the set of participants i which obtain at least one share. Similar to the concept of suitable matrices, the set of vectors C can also define a PSSS.

Definition 2 *([37]) Let $\Gamma^- = \{X_1, X_2, \ldots, X_r\}$ be a minimal access structure and let $c^{(i)} = (c_1, c_2, \ldots, c_n) \in \mathbb{F}_q^N$. Then the set of vectors $K = \{c^{(1)}, c^{(2)}, \ldots, c^{(r)}\} \subseteq \mathbb{F}_q^N$ is said to be* suitable *for the access structure Γ if K satisfies the following properties:*

1. *$Supp(c^{(j)}) = X_j$ for $1 \leq j \leq r$.*
2. *For any vector $(\mu_1, \ldots, \mu_r) \in \mathbb{F}_q^r$, such that $\sum_{j=1}^{r} \mu_i \neq 0$, there exists a set $X \in \Gamma^-$ satisfying $X \subseteq Supp(\sum_{j=1}^{r} \mu_j c^{(j)})$.*

(Note that the last condition helps define the monotonicity of an access structure.) Both a set of matrices and a set of vectors can then define access structures. A few questions then arise: Is there any correlation between a suitable set of matrices and a suitable set of vectors for an access structure Γ? If there exists a suitable set of matrices for Γ^-, does there exist a suitable set of vectors, and vice versa? van Dijk answers these questions with the following theorem:

Theorem 1 *([37], Theorem 5) Let $\Gamma = \{X_1, ..., X_r\}$. Let $G_i, i \in P$, be an $l \times p_i$ matrix over $\mathbb{F}_q$ such that the set $\{G_1, G_2, \ldots, G_n\}$ is suitable. Define H as in Equation 2.1 and $\mathcal{I}$ as the set $\{(i, j) : 1 \leq i \leq r,\ 1 \leq j \leq k\}$. Then there exists a suitable set of vectors*

$\{c^{(i,j)} \in \mathbb{F}_q^N : (i,j) \in \mathcal{I}\}$ *such that* $G'H^\top = 0$, *where* G' *is a generator matrix of the code defined by the linear span of the vectors* $(e^j, c^{(i,j)})$, $(i,j) \in \mathcal{I}$.

A partial converse is as follows:

Let the vectors $c^{i,j} \in \mathbb{F}_q^N$, $(i,j) \in \mathcal{I}$, *define a suitable set of vectors for* Γ. *Let* H *be a parity-check matrix of the code defined by the linear span of the vectors* $(e^j, c^{(i,j)})$, $(i,j) \in \mathcal{I}$, *i.e.,* H *is of the form*

$$\left(\begin{array}{c|c} I_k & G \\ 0 & \end{array}\right). \text{ where } G = [G_1, G_2, \ldots, G_n]$$

Then the set of matrices $\{G_1, G_2, \ldots, G_n\}$ *is suitable for* Γ.

Hence, we can construct a suitable set of vectors given suitable matrices for Γ, and given a suitable set of vectors for Γ, we can easily construct a suitable set of matrices.

2.4.1 Direct Approach

van Dijk's approach is complex due to the difficulty of finding the suitable vectors or matrices. In the following, we present a simpler method for finding G and H when the size of the secret is $k = 1$.

Let the block matrix $G = [G_1, G_2, \ldots, G_n]$ be the concatenation of the publicly known matrices of the participants. Let $s \in S$ be a message. Recall the linear map τ:

$$\tau(s, a) = (s, a)G = (k_1, k_2, \ldots, k_n).$$

For the direct approach, the code $\tilde{C}$ is generated by

$$[g_0, G_1, G_2, \ldots, G_n].$$

where $g_0 = [1, 0, \ldots, 0]^\top$. Suppose $\tilde{C}$ defines an SSS for an access structure Γ. Then $X \in \Gamma$ if and only if $g_0 \in \text{span}\{G_i |\, i \in X\}$; otherwise, $X \notin \Gamma$. Hence, the access structure can be

defined as:

$$\Gamma = \{X \subseteq P \mid g_0 \in \text{span}(G_i \mid i \in X)\}. \tag{2.2}$$

One may also define an authorized set by the support of a dual codeword.

Lemma 2 *Let $\tilde{C}$ with generator matrix $\tilde{G}$ define an SSS for an access structure Γ. Let $X \subseteq P$. Then $X \in \Gamma$ if and only if there exists $h = (h^0, h^1, \ldots h^n) \in \tilde{C}^{\perp}$ such that $h^0 = 1$ and $Supp(h \setminus \{h^0\}) = X$.*

Another way of defining an unauthorized or an authorized set is the following:

Lemma 3 *([6]) For $Y \subseteq P$, $Y \in \mathcal{R}$ if and only if there exists a codeword $c = (c^0, c^1, \ldots, c^n) \in \tilde{C}$ such that $c^0 = 1$ and $c^i = 0$ for all $i \in Y$, i.e., $Supp(c) \cap Y = \emptyset$.*

With the access and adversary structures, we show a method to decide if there is a code $\tilde{C}$ that realizes the access structure Γ. Let $P = \{1, 2, \ldots, n\}$ be the set of participants, $\Gamma^- = \{X_1, X_2, \ldots, X_r\}$ be any access structure, and $\mathcal{R}^+ = \{Y_1, Y_2, \ldots, Y_s\}$ be the corresponding adversary structure. Then we construct $\tilde{G}$ from $\mathcal{R}^+$ and H from Γ^-.

First, let $s \times (1 + N)$ matrix $\tilde{G}$ generate code $\tilde{C}$. Then $\tilde{G}$ must have the following form in order for the code $\tilde{C}$ to realize an access structure Γ:

$$\tilde{G} = [\mathbf{1}, \tilde{G}_1, \tilde{G}_2, \ldots, \tilde{G}_n], \text{ where } \tilde{G}_\ell = \begin{bmatrix} g^\ell_{1,1} & \cdots & g^\ell_{1,p_\ell} \\ \vdots & \ddots & \vdots \\ g^\ell_{s,1} & \cdots & g^\ell_{s,p_\ell} \end{bmatrix}.$$

Letting $(g_i)^\ell = (g^\ell_{i,1}, \ldots g^\ell_{i,p_\ell})$ denote the i^{th} *row* of $\tilde{G}_\ell$, which belongs to player ℓ, we require $(g_i)^\ell = 0$ if and only if $\ell \in Y_i$. In this construction, the columns $(g^1)^\ell, (g^2)^\ell, \ldots, (g^{p_\ell})^\ell$ of $\tilde{G}_\ell$ belong to player ℓ and each entry $g_{i,j} \neq 0$ of $\tilde{G}$ is unknown at the moment. Additionally, the number of shares, p_ℓ, distributed to player ℓ, is chosen to minimize the total number of shares distributed. Note that $\sum_{\ell=1}^{n} p_\ell = N$.

Similarly,

$$H = [\mathbf{1}, H_1, H_2, \ldots, H_n], \text{ where } H_\ell = \begin{bmatrix} h^\ell_{1,1} & \ldots & h^\ell_{1,p_\ell} \\ \vdots & \ddots & \vdots \\ h^\ell_{r,1} & \ldots & h^\ell_{r,p_\ell} \end{bmatrix}$$

Letting $(h_i)^\ell$ denote the i^{th} row of H_ℓ that belongs to player ℓ, we require $(h_i)^\ell \neq 0$ if and only if $\ell \in X_i$. The entries $h_{i,j} \neq 0$ of H are also unknown.

Using the above constructed matrices from Γ and $\mathcal{R}$, we create a linear secret sharing scheme that realizes the access structure Γ.

Theorem 2 *([6]) Given access structure Γ and the number of distributed shares p_i to player i, there exists a linear code $\tilde{C}$ over $\mathbb{F}_q$ if and only if $\tilde{G}H^\top = 0$ has a solution over $\mathbb{F}_q$, where $\tilde{G}$ and H are constructed from Γ^- and $\mathcal{R}^+$ respectively.*

Using Lemmas 2 and 3 and Equation 2.2, we now prove Theorem 2.

Proof of Theorem 2:

Suppose $\tilde{C}$ is a linear code with generator matrix $\tilde{G} = [g_0, \tilde{G}_1, \tilde{G}_2, \ldots, \tilde{G}_n]$, which computes an access structure Γ. Then by Lemma 3, for all $Y_i \in \mathcal{R} = \{Y_1, \ldots, Y_t\}$ there exists a codeword $g_i = (g^0, g^1, \ldots, g^n) \in \tilde{C}$ such that $g^0 = 1$ and $g^j = 0$ for all $j \in Y_i$, i.e., $\text{Supp}(g_i) \cap Y_i = \emptyset$. By Lemma 2, there exists $h_1, h_2, \ldots, h_r \in \tilde{C}^\perp$ such that $h^0_j = 1$ and $\text{Supp}(h) = X$ for $X \in \Gamma$. Hence, we can construct matrices G and H such that $GH^\top = 0$

Suppose $\tilde{G}H^\top = 0$ for the constructed matrices $\tilde{G}$ and H. Then we know

$$\text{Supp}(g_i \setminus \{g^0_i\}) \subseteq P \setminus \{Y_i\} \text{ and } \text{Supp}(h_i \setminus \{h^0_i\}) = X_i$$

by construction. Let $\tilde{C}$ be a linear code with $\tilde{G}$ as the generator matrix. Note that $< g, h >= 0$ implies $1 + g^1h^1 + \ldots g^nh^n = 0$. Obviously, $g^0 = 1 \in \text{span}\{g^j \setminus \{g^0\}\}$ for $j \in \text{Supp}(h \setminus \{h^0\})$. Recall Equation 2.2

$$\Gamma = \{X \subseteq P \mid g_0 \in \text{span}(\tilde{G}_i \mid i \in X)\}.$$

Then $\text{Supp}(h_i \setminus \{h_i^0\}) = X_i \in \Gamma$. It follows that $P \setminus \{\text{Supp}(g_i)\} = Y_i \in \mathcal{R}$. $\square$

Example 4 *Let* $\Gamma^- = \{(1,2),(1,3)\}$ *and* $\mathcal{R}^+ = \{(1),(2,3)\}$. *Suppose each participant receives only one share, i.e., each player owns one column of* $\tilde{G}$ *and* H. *Then*

$$\tilde{G} = \begin{bmatrix} 1 & 0 & g_{1,2}^2 & g_{1,3}^3 \\ 1 & g_{2,1}^1 & 0 & 0 \end{bmatrix} \text{ and } H = \begin{bmatrix} 1 & h_{1,1}^1 & h_{1,2}^2 & 0 \\ 1 & h_{2,1}^1 & 0 & h_{2,3}^3 \end{bmatrix}.$$

It follows that

$$\tilde{G}H^\top = \begin{bmatrix} 1 + g_{1,2}^2 h_{1,2}^2 & 1 + g_{1,3}^3 h_{2,3}^3 \\ 1 + g_{2,1}^1 h_{1,1}^1 & 1 + g_{2,1}^1 h_{2,1}^1 \end{bmatrix}$$

Letting the field be $\mathbb{F}_5$, *a solution to the quadratic system of equations* $\tilde{G}H^\top = 0$ *will be*

$$\tilde{G} = \begin{bmatrix} 1 & 0 & 2 & 2 \\ 1 & 2 & 0 & 0 \end{bmatrix} \text{ and } H = \begin{bmatrix} 1 & 2 & 2 & 0 \\ 1 & 2 & 0 & 2 \end{bmatrix}.$$

Hence, there is a linear code for the given access structure Γ^- *and* $N = 3$ *shares.*

Example 5 *Imagine that a research facility contains a top-secret research lab and in that lab are deadly diseases, which may only be accessed when the following groups are present: the president and vice-president; the president and two committee members; the vice-president and two committee members; or all four committee members of the research facility. In this scenario, we have the set* $P = \{1,2,3,4,5,6\}$, *an access structure*

$$\begin{aligned} \Gamma^- = &\{(1,2),(1,3,4),(1,3,5),(1,3,6),(1,4,5),(1,4,6),(1,5,6),(2,3,4),(2,3,5), \\ &(2,3,6),(2,4,5),(2,4,6),(2,5,6),(3,4,5,6)\}, \end{aligned}$$

and an adversary structure

$$\mathcal{R}^+ = \{(1,3),(1,4),(1,5),(1,6),(2,3),(2,4),(2,5),(2,6),(3,4,5),(3,4,6),(3,5,6),$$
$$(4,5,6)\}.$$

Concerning this example, finding the solution to the system $GH^\top = 0$ is difficult and tedious. However, one may find that the system is inconsistent for $N = 6$ and for $N = 7$, where N is the total number of shares distributed. On the other hand, when $N = 8 = n+2$, the access structure can be realized by a $(4, 8)$-threshold scheme where the president (player 1) and the vice-president (player 2) each receive two shares and the committee members each receive one. Hence, by Theorem 2, $GH^\top = 0$ has a solution over $\mathbb{F}_q$ for some prime power q. The specifics may be found in Section 4.1 along with the code constructed to narrow down variables and equations.

2.4.2 Monotone Span Programs

Theorem 2 guarantees that a linear secret sharing scheme exists if $GH^\top = 0$ is consistent. Once we know there is an LSSS realizing Γ, a natural inclination is to find such an LSSS.

A *monotone span program* (MSP) is another approach to computing access structures, and it consists of four parameters: a field, a matrix M, a labeling function, and a target vector. More specifically,

Definition 3 *([21]) A monotone span program M is given by a quadruple $(\mathbb{F}, M, \phi, \varepsilon)$, where $\mathbb{F}$ is a field, M is a $(m \times d)$ matrix over $\mathbb{F}$ with a labeling function*

$$\phi : \{1, \ldots, m\} \to \{1, \ldots, n\},$$

which assigns to every row of M a participant in P, and ε is a fixed non-zero vector, called the target vector.

Similar to linear secret sharing schemes, we can utilize an MSP to realize any given access structure. Let the set of players be denoted as $P = \{1, 2, \ldots, n\}$. Let $A \subseteq P$ be an arbitrary set and $M_A \in \mathbb{F}^{t \times d}$ represent the submatrix of M formed by the rows of the participants in A. Then an MSP *accepts* $A \subseteq P$ if and only if the target vector $\varepsilon \in \text{span}(M_A)$ where M_{A_i} denotes the i^{th} row of M_A. Otherwise, the MSP *rejects* A. Without loss of generality, we let ε be the concatenation of $\mathbf{1} \in \mathbb{F}^k$ and $\mathbf{0} \in \mathbb{F}^{\ell-k}$.

If A is accepted, there exists a *recombination vector* λ such that $M_A^\top \lambda = \varepsilon$. Similar to LSSSs, the players in possession of the recombination vector are able to decipher the secret:

$$\langle \lambda, M_A(s, a)^\top \rangle = \langle M_A^\top \lambda, (s, a)^\top \rangle = \langle \varepsilon, (s, a)^\top \rangle = s$$

It follows that in order to utilize an MSP as an SSS for any access structure Γ, one creates a matrix M that accepts each authorized set X and rejects each unauthorized set Y.

Example 6 *Let the monotone span program be* $(\mathbb{F}_5, M, \rho, \varepsilon)$ *where*

$$M = \begin{bmatrix} 1 & 2 & 3 \\ 1 & 4 & 1 \\ 3 & 1 & 2 \\ 1 & 1 & 1 \end{bmatrix}$$

and

$$\rho(1) = \rho(2) = 1,\ \rho(3) = 2,\ \textit{and}\ \rho(4) = 3.$$

Note that $d_1 = 2$, $d_2 = d_3 = 1$.

There exists a vector $v \in \mathbb{F}_5^{1 \times 3}$ *such that* $vM_{\{1,2\}} = e_1$*; specifically* $v = \begin{bmatrix} 2 & 4 & 0 \end{bmatrix}^\top$. *In addition,* $vM_{\{1,3\}} = e_1$ *where* $v = \begin{bmatrix} 2 & 4 & 0 \end{bmatrix}^\top$. *Thus, the MSP* M *accepts the sets* $\{1, 2\}$ *and* $\{1, 3\}$. *On the other hand, there is no linear combination of the rows of either of the submatrices* $M_{\{1\}}$ *and* $M_{\{2,3\}}$ *that produces* e_1. *Thus, the MSP* M *rejects the sets* $\{1\}$

and $\{2,3\}$. *In conclusion, the MSP* M *computes the access structure* $\Gamma^- = \{(1,2),(1,3)\}$.

Now, we have the means to know when an MSP realizes an access structure Γ. Let G be the matrix constructed from the adversary structure $\mathcal{R}^+$ and $M = (G \setminus \{g^0\})^\top$. Then we have the equivalence to Theorem 2:

> There is an MSP with matrix $M = (G \setminus \{g^0\})^\top$ for Γ if and only if $GH^\top = 0$ is consistent over $\mathbb{F}$, where G and H are constructed from Γ^- and $\mathcal{R}^+$ respectively.

Example 7 *Recall Example 4. Setting* $M = (G \setminus \{g^0\})^\top$, *we obtain an MSP with matrix*

$$M = \begin{bmatrix} 0 & 2 & 2 \\ 2 & 0 & 0 \end{bmatrix}$$

which realizes the access structure Γ^-.

2.5 Lower and Upper Bounds

An open question concerning secret sharing schemes concerns the efficiency of a scheme, i.e., the number of shares distributed and the information rate. At the moment, the available optimal schemes have information ratio (the inverse of information rate) of $2^{O(n)}$, where n is the cardinality of the access structure. Utilizing Shannon inequalities, Csirmaz produced and proved a nearly optimal lower bound: "For every n there exists an n-party access structure Γ_n such that every secret sharing scheme realizing it has information ratio $\Omega(n/\log n)$" [2]. Before this result, Karnin [13] made an observation that the number of shares distributed to an authorized group must equal or exceed the secret's size.

Additionally, Karchmer and Wigderson [12] used the concept of an MSP to find a lower bound for the information ratio. An underlying feature of MSPs is the fact that the rank of a matrix represents both the communicational complexity in an LSSS and the computational complexity for reconstructing keys. Consequently, the rank of the MSP matrix, otherwise known as the MSP complexity, provides the best means of determining

the efficiency of an LSSS and as a result, it is a focal point in research of secret sharing schemes. More specifically, in [12], Karchmer and Wigderson proved that if there is an MSP of size N for some function, then there exists a scheme for the corresponding secret sharing problem in which the sum of the lengths of the shares of all the parties is N. Thus, every lower bound on the total size of shares in a secret sharing scheme is also a bound on the size of MSPs for the same function. As a result, finding the best lower bound of an MSP correlates to finding the best lower bound of the corresponding secret sharing scheme.

Additionally, graph theory, specifically polymatroids, comes into play.

Definition 4 *([23]) A polymatroid is a pair (Q, f), where Q is a finite set, and f is a map $f : P(Q) \to \mathbb{R}$ satisfying the following properties.*

1. *$f(\emptyset) = 0$.*
2. *f is monotone increasing: if $A \subseteq B \subseteq Q$, then $f(A) \leq f(B)$.*
3. *f is submodular: $f(A \cup B) + f(A \cap B) \leq f(A) + f(B)$ for all A, $B \subseteq Q$.*

Employing polymatroids, the exact optimal information ratio has been found for all tree defined access structures. Furthermore, the optimal information ratio of the majority of the access structures with fewer than six players, and of the "access structures defined by graphs with at most six vertices," has been proven [23]. More information concerning polymatroids and optimal bounds may be found in [23].

2.6 Conclusion

For any given access structure Γ, there is a secret sharing scheme realizing Γ. The schemes discussed have been the traditional schemes, Reed-Solomon based schemes, and monotone span programs. The question of optimality enters each scenario and has been shown to be nearly $\Omega(n/logn)$ by Csirmaz. However, moving away from the established Shannon's inequalities, a better optimal information rate is yet to be shown. Thus, we have the following open problems:

- What are the characteristics of ideal structures?
- Can the current lower bound $\Omega(n/logn)$ be improved?
- Can we create an access structure where the number of shares linearly depends on the number of participants?

Chapter 3

Network Coding

3.1 Introduction

Recently, communication has rapidly moved to the area of IP networks. Messages can be sent to any location in the world as long as one has connection to the internet. This progression of technology has greatly changed the field of coding and brought about the invention of network coding. Networking is generally pictured as a directed graph with the sender represented as the initial (source) node, the vehicle as the edges, the intermediate devices as the intermediate nodes, and the recipient as a receiver node. The sender initiates the coding process by splitting a message into n packets and sending them to multiple destinations. The packets are then transmitted from point to point throughout the network until they reach their final destination. In the beginning of network coding, the packets were unchanged during the transmission process and thus, collected little or no error. However, the described process is highly susceptible to eavesdroppers, giving rise to the necessity of encoding the packets. One solution is to let each intermediate node encode each of its outgoing packets by sending a fixed linear combination of the packets received rather than the uncoded packets. Consequently, information may be lost, and thus there is a need for error-correcting codes. Cai and Yeung [4] took this challenge and found a means of using the Hamming distance to produce an error-correcting code for a network code.

At the turn of the 21st century, noncoherent networks became a possibility. In this case, the intermediate nodes no longer have any knowledge of the underlying system, i.e., they are blind to the structure of the network and, as a result, are not able to retrieve the message sent over the network. As one can imagine, this result has its benefits in the internet world. In the noncoherent network, each intermediate node creates a random linear combination of its received packets and sends those to the next destination.

Koetter and Kschischang [16] furthered the work on error control for a noncoherent network, followed by Silva and Kschischang [33]. The remainder of this thesis will be dedicated to Silva's method, coset coding. Coset coding utilizes the rank metric and achieves both correctness and privacy. One of the key factors of coset coding is the characteristic of universal security when the eavesdropper is not allowed to tap a link more than once. In [27], Shioji *et al.* prove that a network may not satisfy the condition of universal security if the eavesdropper is permitted to tap a link on multiple occasions.

3.2 Rank Metric Codes

In this thesis, we examine network codes, specifically coset coding. A couple of building blocks of network codes are the rank distance and the rank metric code. Let $X, Y \in \mathbb{F}_q^{n\times m}$. The *rank distance* between X and Y is defined as

$$d_R(X,Y) = \text{rank}(Y - X).$$

A *rank metric code* is a matrix code $C \subseteq \mathbb{F}_q^{n\times m}$, which utilizes the rank metric. The distance of C is the minimum distance between two distinct codewords $X, Y \in C$, i.e.,

$$d_R(C) = \min_{\substack{X,Y\in C \\ X\neq Y}} (d_R(X,Y)).$$

To ensure universal security (discussed in Section 3.4), we assume $m \geq n$ for the duration of this thesis. Also, we only consider linear matrix codes in this thesis. Hence, the Singleton

bound applies. Let $C \subseteq \mathbb{F}_q^{n \times m}$ be a matrix code with dimension k. Then

$$d_R(C) \leq n - k + 1.$$

Equivalently,

$$|C| = q^k \leq q^{n-d+1}.$$

If $d_R(C) = n - k + 1$, C is called a *maximum rank distance (MRD) code*. Note that when $m \geq n$, MRD codes are MDS codes. Three examples of an MRD code are the Gabidulin code, the generalized Gabidulin, and the Cartesian product of an MRD code as described in [8]. The Gabidulin code will be referenced later in Section 3.6.

3.3 Network Coding Layout

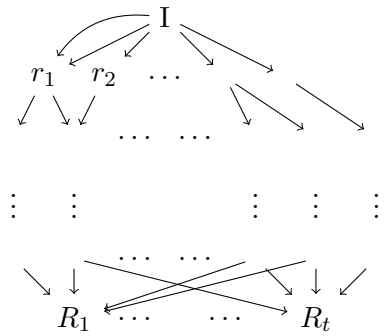
In this section, we describe the process of a network code. Suppose we want to send n packets of size m from the source node I to designated receivers $R_1, \ldots, R_t$. Let G be a directed graph composed of the source node, the receivers, the intermediate nodes, and the unit capacity edges, which must have min-cut of at least n, i.e., the max flow rate is at least n. Let $s \in S$ be a message to be transmitted to the receivers. First, the source node I splits the message s into n packets, each of length m. Let

$$X = \begin{bmatrix} -X_1- \\ -X_2- \\ \vdots \\ -X_n- \end{bmatrix} \in \mathbb{F}_q^{n \times m}$$

be the row concatenation of the n packets.

Then I sends linear combinations of $X_1, X_2, \ldots, X_n$ to the receivers through the directed graph G. More specifically, for each adjacent edge e, the source node I does the following:

Figure 3.1: General Network Layout



1. Computes a random linear combination

$$P_e = c_e X$$

   where c_e represents the corresponding linear combination and is called the *global coding vector* for edge e.

2. Transmits the packet P_e over edge e to the corresponding intermediate node r.

On obtaining the incoming packets, an intermediate node r:

1. Computes a random linear combination $P_e = c_e X$ of its received packets for each outgoing edge e.

2. Transmits P_e over the corresponding edge e.

For simplicity, we denote the collection of receivers as $\mathcal{R} = \{R_1, R_2, \ldots, R_n\}$, the collection of edges as $\epsilon = \{e_1, e_2, \ldots, e_n\}$ and we let

$$\tilde{C} = \begin{bmatrix} -c_{e_1}- \\ -c_{e_2}- \\ \vdots \\ -c_{e_n}- \end{bmatrix}$$

be a row concatenation of the global coding vectors $c_e \in \mathbb{F}_q^n$. Each receiver R can be uniquely identified by the set R_e of its incoming edges. Let $\tilde{C}_R$ be the submatrix of $\tilde{C}$, composed of the global coding vectors c_e such that $e \in R_e$. Then the combination of the received packets of R is defined as

$$Y(R) = \tilde{C}_R X.$$

However, the linear combinations that a node receives may contain error. For example, if $\tilde{C}_R$ does not have full rank, information pertaining to a packet X_i is lost. In this case, the code C is called *rank-deficient* for R; otherwise, the code is called feasible. The rank deficiency ρ of the network is defined as

$$\rho = n - \min_{R \in \mathcal{R}} \text{rank}(\tilde{C}_R)$$

where $n \leq s$ and n is the number of columns of $\tilde{C}_R$.

3.4 Coset Coding

We now consider the scheme *coset coding*, which was constructed by Ozarow and Wyner, and remodified by Silva and Kschischang [33]. A *coset code* C is an $[n, n-k]$ linear code over the field $\mathbb{F}_q^m$ with a parity-check matrix $H \in \mathbb{F}_q^{k \times n}$. Let $\hat{S} = \mathbb{F}_{q^m}^k$ be the message space for C. Then for each message $S \in \hat{S}$, the sender (source node) does the following:

1. Chooses an arbitrary matrix $X \in \mathbb{F}_{q^m}^n$ such that

$$S = HX.$$

2. Transmits X through a network to designated receivers $R \in \mathcal{R}$ as described in Section 3.3.

In a noiseless network, the receiver R reconstructs S by simply evaluating HX, i.e., the network achieves the condition correctness and $H(S|Y(R)) = 0$. In this section, we assume the network is noiseless and thus, correctness holds.

Though we assume correctness, the network might not be private. Let μ denote the maximum number of edges that an eavesdropper may intercept. Without loss of generality, we assume the eavesdropper intercepts exactly μ edges. Let the observation of the eavesdropper be defined as

$$W = BX$$

where $B \in \mathbb{F}_q^{\mu \times n}$ is a matrix composed of μ altered, intercepted linear combinations c_e. As a result, the eavesdropper obtains μ linear combinations of the n sent packets, $X_1, X_2, \ldots, X_n$.

However, note that X, H, and S contain entries in the extension field $\mathbb{F}_{q^m}$; whereas, the entries of B are in $\mathbb{F}_q$. This feature allows us to fix a parity-check matrix H such that $H(S|W) = H(S)$, i.e., the observation W does not provide the eavesdropper with any information pertaining to S.

Definition 5 *([33]) A coding scheme is universally secure under μ observations if $H(S|W) = H(S)$ for each eavesdropper observation $W = BX$, for all $B \in \mathbb{F}_q^{\mu \times n}$*

If $m = 1$, then the entries of B lie in the same field as the entries of H, X, and S. Hence, it is impossible to create a universally secure coding scheme when $m = 1$. Suppose $m > 1$. In [33], Silva and Kschischang prove there exists a universally secure coding scheme and provide a means of construction. Before we provide a proof of the existence of a universally secure coding scheme, we introduce a few theorems.

Theorem 3 *(Theorem 4, [33]) If $I(S;W) = 0$, then $H(S) \leq n - \mu$. Moreover, if $H(S) = k = n - \mu$, then*

$$I(S;W) = 0 \Leftrightarrow \langle H \rangle \cap \langle B \rangle = 0.$$

Recall $S = HX$ and $W = BX$. If H and B have any rows in common then some information pertaining to S has been revealed. So, if $\langle H \rangle \cap \langle B \rangle = 0$, then the rows are linearly independent. We then obtain the following:

Theorem 4 *(Theorem 5, [33]) Let C be an [n,n-k]-linear code over $\mathbb{F}_{q^m}$ with parity-check matrix $H \in \mathbb{F}_{q^m}^{k\times n}$. If $d_R(C) = k+1$ and $\mu \leq n-k$, then*

$$rank \begin{bmatrix} H \\ B \end{bmatrix} = rank(H) + rank(B), \forall B \in \mathbb{F}_q^{\mu\times n}$$

Conversely, if $\mu = n-k$, then the above holds only if $d_R(C) = k+1$.

Proof: [33]

Let C be an $[n, n-k]$-linear code over $\mathbb{F}_{q^m}$ with parity-check matrix $H \in \mathbb{F}_{q^m}^{k\times n}$. We are going to first prove the forward direction by contradiction. Suppose $d_R(C) = k+1$ and suppose there exists $B \in \mathbb{F}_q^{\mu\times n}$ for $\mu \leq n-k$ such that

$$\text{rank} \begin{bmatrix} H \\ B \end{bmatrix} < \text{rank}(H) + \text{rank}(B).$$

Let $r = \text{rank}(B)$. Let $T \in \mathbb{F}_q^{r\times\mu}$ where T is full-rank such that

$$\text{rank}(TB) = r.$$

In addition, let $D \in \mathbb{F}_q^{(n-k-r)\times n}$ be of full rank such that $\langle D\rangle \cap \langle TB\rangle = 0$, i.e.,

$$\begin{bmatrix} TB \\ D \end{bmatrix} \in \mathbb{F}_q^{(n-k)\times n}$$

is full rank. Let

$$B' = \begin{bmatrix} TB \\ D \end{bmatrix}.$$

Let the matrix

$$M = \begin{bmatrix} H \\ B' \end{bmatrix}.$$

Then it follows that

$$\begin{aligned}\operatorname{rank}(M) = \operatorname{rank}\begin{bmatrix} H \\ TB \\ D \end{bmatrix} &\leq \operatorname{rank}\begin{bmatrix} H \\ B \end{bmatrix} + \operatorname{rank}(D) \\ &< \operatorname{rank}(H) + \operatorname{rank}(B) + \operatorname{rank}(D) \\ &= n.\end{aligned}$$

Hence, $\operatorname{rank}(M) < n$, i.e., there exists an $x \in \mathbb{F}_q^n \setminus \{0\}$ such that $Mx = 0$. It follows that

$$\begin{bmatrix} H \\ B' \end{bmatrix} x = 0$$

giving us $Hx = 0$. (Recall, H is a parity-check matrix for C, resulting in $x \in C$.) In addition, $B'x = 0$, which implies that some information has been retrieved and thus, $d_R(C) \leq k$. This contradicts our first assumption that $d_R(C) = k + 1$. Therefore, if $d_R(C) = k + 1$ and $\mu \leq n - k$, then

$$rank\begin{bmatrix} H \\ B \end{bmatrix} = rank(H) + rank(B), \forall B \in \mathbb{F}_q^{\mu \times n}.$$

To prove the converse, let $\mu = n - k$ and suppose $d_R(C) \leq k$. Then there exists a codeword $x \in C \setminus 0$ such that the $\operatorname{rank}(x) \leq k$, which implies that there exists a matrix $B \in \mathbb{F}_q^{(n-k)\times n}$ with full rank such that $Bx = 0$. Clearly, $Hx = 0$ since $x \in C$ and H is the parity-check matrix. So, $CS(B) \cap CS(H) \neq \emptyset$ where CS denotes the column space. This implies that

$$\operatorname{rank}\begin{bmatrix} H \\ B \end{bmatrix} < n = \operatorname{rank}(H) + \operatorname{rank}(B).$$

Therefore, if $\mu = n - k$ and

$$\text{rank} \begin{bmatrix} H \\ B \end{bmatrix} = \text{rank}(H) + \text{rank}(B),$$

then $d_R(C) = k$. □

It then follows that we may construct a universally secure network.

Theorem 5 *(Theorem 7, [33]) Consider an $(n \times m, n)_q$ linear coded network. Let C be an [n,n-k]-linear code over $\mathbb{F}_{q^m}$ with parity-check matrix $H \in \mathbb{F}_{q^m}^{k \times n}$. A coset coding scheme based on H is universally secure under μ observations if $k \leq n - \mu$, $m \geq n$, and C is MRD. Conversely, in the case of a uniformly distributed message, the scheme is universally secure under $n - k$ observations only if C is an MRD code with $m \geq n$.*

In order to prove the above theorem, we will need the following lemma:

Lemma 4 *(Lemma 6, [33]) Let $H \in \mathbb{F}_{q^m}^{k \times n}$ and $B \in \mathbb{F}_q^{\mu \times n}$. Let $X \in \mathbb{F}_{q^m}^n$ be random, and let $S = HX$ and $W = BX$. Let*

$$\hat{S} = \{Hx : x \in \mathbb{F}_{q^m}^n\}$$

and, for all $s \in \hat{S}$, let

$$\hat{X}_s = \{x \in \mathbb{F}_{q^m}^n : s = Hx\}.$$

1.) If X is uniform over $\hat{X}_s$, then

$$I(S;W) \leq rank(H) + rank(B) - rank \begin{bmatrix} H \\ B \end{bmatrix}.$$

2.) If S is uniform over $\hat{S}$, then

$$I(S;W) \geq rank(H) + rank(B) - rank \begin{bmatrix} H \\ B \end{bmatrix}.$$

Proof of Theorem 5:

Let $k \leq n - \mu$, $m \geq n$ and C be MRD. Assuming X is uniform, then by Lemma 1 part 1,

$$I(S;W) \leq rank(H) + rank(B) - rank \begin{bmatrix} H \\ B \end{bmatrix}.$$

In addition, since C is MRD,

$$rank \begin{bmatrix} H \\ B \end{bmatrix} = rank(H) + rank(B)$$

for all $B \in \mathbb{F}_q^{\mu \times n}$ by Theorem 2. As a consequence, $I(S;W) = 0$ and we have a universally secure coset coding scheme based on H under μ observations. Concerning the converse, let S be uniformly distributed and $I(S;W) = 0$ with $n - k$ observations. Then by Lemma 1 part 2,

$$0 = I(S;W) \geq rank(H) + rank(B) - rank \begin{bmatrix} H \\ B \end{bmatrix} \geq 0$$

which implies

$$rank(H) + rank(B) = rank \begin{bmatrix} H \\ B \end{bmatrix}.$$

By the converse of the previous theorem, $d_R(C) = k + 1$ and thus, C is MRD and $m \geq n$. □

3.5 Similarities between Coset Coding and Secret Sharing

We will now briefly point out some similarities between coset coding and secret sharing schemes. Recall that in a linear secret sharing scheme, we have the following: Let $S = \mathbb{F}_q^k$ be the message space, $V = \mathbb{F}_q^{\ell - k}$ be a finite field, and $S_i = \mathbb{F}_q^{p_i}$ be the share space of participant i, where $1 \leq i \leq n$ and $\ell \geq k$. For $1 \leq i \leq n$, let G_i be a publicly known $\ell \times p_i$ matrix over $\mathbb{F}_q$ of player i. We represent τ by a block matrix $G = [G_1, G_2, \ldots, G_n]$ so that

with a message s and a randomly chosen vector a, the dealer computes

$$\tau(s, a) = (s, a)G = (k_1, k_2, \ldots, k_n)$$

and then distributes k_i secretly to player i.

In the case of coset coding, let $\hat{S} \in \mathbb{F}_{q^m}^k$ be the message space, $S \in \hat{S}$ be the message, and $H \in \mathbb{F}_{q^m}^{k \times n}$ be the parity-check matrix. Then the sender chooses an arbitrary $X \in \mathbb{F}_{q^m}^n$ such that $S = HX$.

Now, let X be chosen such that $X^\top = S^\top Y$ for some fixed Y. Then $X^\top = [X_1^\top, X_2^\top, \ldots, X_n^\top]$ can be viewed as the shares $(k_1, k_2, \ldots, k_n)$ to be transmitted.

Additionally, in coset coding, if the eavesdropper has a linear combination of μ or less shares, he is not able to obtain any information pertaining to the secret S. This closely relates to the mutlitple sharing (μ, n)-threshold scheme.

3.6 Characteristics of Parity-Check Matrix H

The question that still remains is how to construct a universally secure scheme. In this section, our goal is to construct a parity-check matrix $H \in \mathbb{F}_{q^m}^{k \times n}$ with rank k such that the coset coding scheme is universally secure. We utilize theorems from Section 3.4 and ideas from other works to construct a matrix H such that $H(S|W) = H(S)$.

Let C be an $[n, n-k]$ linear code unless otherwise specified. Let μ represent the number of observations and $W = BX$ be the information gleaned from the observations made by the eavesdropper. In the following, we present a few approaches in constructing a universally secure network.

1. Let $\mu \leq n - k$ and $m \geq n$. Let H be a parity-check matrix for an MRD code over

$\mathbb{F}_{q^m}$. Then by Theorem 5, C is universally secure. For example, let

$$H = \begin{bmatrix} g_1 & \cdots & g_n \\ g_1^q & \cdots & g_n^q \\ \vdots & \ddots & \vdots \\ g_1^{q^{k-1}} & \cdots & g_n^{q^{k-1}} \end{bmatrix}$$

where $g_i \in F_{q^m}$. The Vandermonde matrix H is the parity-check matrix of the Gabidulin code, which is an MRD code. Hence, the coset coding scheme is universally secure.

2. Let C be an MDS code with the parity-check matrix $H \in \mathbb{F}_{q^m}^{k \times n}$ such that any linear combination of μ or less global coding vectors $c_i \in C$ will not be in $\langle H \rangle$. If this holds true, then $\langle H \rangle \cap \langle B \rangle = \emptyset$ since B is composed of the rows of linear combinations of the global coding vectors. Letting $\mu \leq n - k$, then by Theorem 3, the coding scheme is universally secure [25].

3. Another characteristic that arises in [25] is quite similar. Let K be a generator matrix of a linear α-error-correcting code of dimension $(n - 2\alpha)$. Let C_w be a matrix of rank μ with rows composed from the global coding vectors corresponding to the μ observed edges. Let $H \in \mathbb{F}_{q^m}^{k \times n}$ such that

$$\text{rank} \begin{bmatrix} H \\ C_w K \end{bmatrix} = k + \mu$$

for all C_w. As a result,

$$\text{rank} \begin{bmatrix} H \\ B \end{bmatrix} = k + \mu = \text{rank}(H) + \text{rank}(B)$$

for all B. Then by Part 2) from above, C is universally secure.

4. Yet another strategy constructed was by Feldman *et al.* [7]. They concluded that a universally secure code is formed by the following procedure:

   Let T be an invertible $n \times n$ matrix over the field $\mathbb{F}_q$. Let $H \in \mathbb{F}_q^{k \times n}$ be the matrix composed of the first k rows of T^{-1}. Then security "holds if and only if any set of vectors consisting of:

   1.) at most μ linearly independent edge coding vectors and/or

   2.) any number of vectors from the first k rows of T^{-1}

   is linearly independent" [25].

3.7 The Problematic Eavesdropper

However, the coset coding scheme derived by Silva and Kschischang [33] did not take into account the possibility of the eavesdropper tapping into the same link multiple times. In [27], the authors point out the flaws of the above described network codes. More specifically, they point out the fact that "each symbol is transmitted over multiple time slots" [27]. In other words, X is split and the vector $(X_1^{(t)}, X_1^{(t)}, \ldots, X_1^{(t)})^\top$ is sent over the network through m time slots where $1 \leq t \leq m$.

In [27], Shioji *et al.* provide the following example to disprove universal security when when an eavesdropper is permitted to tap into a link multiple times.

Let $q = 2$, $k = 1$, $n = 2$, and $m = 2$ over the field $\mathbb{F}_4 = \mathbb{F}_2(\alpha)$ where α is a root of the irreducible polynomial $f(x) = x^2 + x + 1$. Let H = [1,α] be the parity-check matrix. Then we obtain a [2,1] MRD code with $S = X_1 + \alpha X_2$. Since the length is 2, we have the following global coding vectors:

$$(0,1)^\top, (1,0)^\top, (1,1)^\top$$

X_1	X_2	$X_1 + X_2$	S
$0 = (0,0)$	$0 = (0,0)$	$0 = (0,0)$	0
$0 = (0,0)$	$\alpha^0 = (0,1)$	$\alpha^0 = (0,1)$	$\underline{\alpha^1}$
$0 = (0,0)$	$\alpha^1 = (1,0)$	$\alpha^1 = (1,0)$	α^2
$0 = (0,0)$	$\alpha^2 = (1,1)$	$\alpha^2 = (1,1)$	$\underline{\alpha^0}$
$\alpha^0 = (0,1)$	$0 = (0,0)$	$\alpha^0 = (0,1)$	$\underline{\alpha^0}$
$\alpha^0 = (0,1)$	$\alpha^0 = (0,1)$	$0 = (0,0)$	α^2
$\alpha^0 = (0,1)$	$\alpha^1 = (1,0)$	$\alpha^2 = (1,1)$	$\underline{\alpha^1}$
$\alpha^0 = (0,1)$	$\alpha^2 = (1,1)$	$\alpha^1 = (1,0)$	0
$\alpha^1 = (1,0)$	$0 = (0,0)$	$\alpha^1 = (1,0)$	α^1
$\alpha^1 = (1,0)$	$\alpha^0 = (0,1)$	$\alpha^2 = (1,1)$	0
$\alpha^1 = (1,0)$	$\alpha^1 = (1,0)$	$0 = (0,0)$	α^0
$\alpha^1 = (1,0)$	$\alpha^2 = (1,1)$	$\alpha^0 = (0,1)$	α^2
$\alpha^2 = (1,1)$	$0 = (0,0)$	$\alpha^2 = (1,1)$	α^2
$\alpha^2 = (1,1)$	$\alpha^0 = (0,1)$	$\alpha^1 = (1,0)$	α^0
$\alpha^2 = (1,1)$	$\alpha^1 = (1,0)$	$\alpha^0 = (0,1)$	0
$\alpha^2 = (1,1)$	$\alpha^2 = (1,1)$	$0 = (0,0)$	α^1

Figure 3.2: Chart for possible messages given the received values X_1 and X_2. Underlined messages are the possible messages given the values $X_1^{(1)}$ and $(X_1 + X_2)^{(2)}$.

and the possible outgoing packets:

$$X_1, X_2, \text{ and } X_1 + X_2.$$

Assume the eavesdropper obtains the following observation:

$$(X_1^{(1)}, (X_1 + X_2)^{(2)}) = (0,1).$$

Then the only possible values for s are α^0 and α, implying the message α^2 was eliminated from the possible choices (see Table 3.2). Consequently,

$$H(S|(X_1^{(1)}, (X_1 + X_2)^{(2)})) \neq H(S),$$

eliminating the possibility of universal security.

In [27], a more general proof is provided that demonstrates the impossibility of

obtaining a universally secure network coding scheme where the eavesdropper has the power to re-select a link. In addition, Shioji *et al.* [27] give a scheme that is universally secure under the more powerful eavesdropper.

3.8 Noisy Networks

We are now going to briefly discuss coset coding in a noisy network. Let $\mathcal{R} = \{R_1, R_2, \ldots, R_t\}$ denote the designated receivers and $\varepsilon = \{e_1, e_2, \ldots, e_{|\varepsilon|}\}$ denote the edges of the network. Let $\tilde{C}$ be the global coding matrix. Let X be the matrix of the n packets created by the initial node I.

Recall that in a noiseless network, the received packets of R were represented by $Y(R) = \tilde{C}_R X$ where $\tilde{C}_R$ was the route of the obtained packets. In the scenario of a noisy network, some of the packets may become distorted or deleted. When either of these occur, error has been added to the packet. As a result, the received packets of R are denoted as

$$Y(R) = \tilde{C}_R X + F_R Z,$$

where $Z \in \mathbb{F}_q^{|\varepsilon| \times m}$ consists of the injected error packets, and $F_R \in \mathbb{F}_q^{|\varepsilon| \times |\varepsilon|}$ represents the route to the node R of those injected error packets. So, the number of non-zero rows in Z may be viewed as the number of erroneous packets. Let t be the maximum number of erroneous packets injected into the network by either a malicious node or an eavesdropper, and let ρ be the number of packets deleted.

In the following, we are going to construct a coset coding scheme with universal security and correctness. First, let us consider universal security. Let C be an $(n \times m, n - \rho)_q$ linear code with t errors and μ observations where $m \geq n$. Note that $(n \times m, n - \rho)_q$ implies that C has ρ rank deficiency. Let $S' \in \mathbb{F}_{q^m}^{(n-\mu)}$ and $T \in \mathbb{F}_{q^m}^{n \times n}$ be nonsingular. Let

$$X = T \begin{bmatrix} S' \\ V \end{bmatrix},$$

where $V \in \mathbb{F}_{q^m}^{\mu}$ is picked randomly and independently from S'. Let G be the last μ rows of $T^\top$. Then we have the following:

Proposition 1 *(Proposition 9, [33]) The above is universally secure under $\mu \leq n-k$ observations if the last $n-k$ rows of $T^\top$ form a generator matrix of an $[n, n-k]$ linear MRD code over $\mathbb{F}_{q^m}$ with $m \geq n$.*

We now have two options to consider when creating a universally secure code:

1. If G is a generator matrix for an $[n, \mu]$-linear MRD code over $\mathbb{F}_{q^m}$, then by Proposition 1 we have universal security for μ eavesdropped links.
2. If $S' = \begin{bmatrix} 0 & S \end{bmatrix}^\top$ and $k \leq n - \mu$, then the resulting code is universally secure by Theorem 5.

Now, let us consider correctness, i.e., $H(S|R_i) = 0$ for $1 \leq i \leq t$. To be able to achieve correctness, we introduce the term universally t-error-ρ-erasure-correcting codes and a theorem concerning it.

Definition 6 *([33]) A coding scheme for an $(n \times m)_q$ linear coded network is universally t-error-ρ-erasure-correcting if it is zero-error under the fan-out set*

$$Y_x = \{(A, y) \in \mathbb{F}_q^{n \times n} \times \mathbb{F}_q^{m \times n} | y = Ax + Z,\ rank(A) \geq n - \rho,\ rank(Z) \leq t,\ Z \in \mathbb{F}_q^{n \times m}\}$$

The following theorem and its proof can be found in [33].

Theorem 6 *(Theorem 2, [33]) Consider a deterministic encoder $X = E(S)$, where $E : \hat{S} \to \hat{X}$, and let $C = E(S) : S \in \hat{S}$. Then the encoder is universally t-error-ρ-erasure-correcting if and only if $d_R(C) \geq 2t + \rho$.*

Let $U = \begin{bmatrix} S \\ V \end{bmatrix}$. Then let $E(S) = G^\top U = X$ where G is the matrix composed of the last $k + \mu$ rows of $T^\top$. By Theorem 6, the code achieves correctness if $d_R(C) \geq 2t + \rho$.

Hence, we have a method of creating a code which achieves correctness and is universally secure.

3.9 Conclusion

In this section, we have shown that a universal secure network is achievable for no more than μ observations of distinct links. We have proven that coset coding is universally secure when an eavesdropper is only allowed to tap a link once. Also, we have shown how one can construct a parity-check matrix H to make the scheme perfectly secure for every set of μ or less observations. Some open problems and future work pertaining to network coding, specifically coset coding, are the following:

- Can one generalize the described results beyond multicast networking?
- How can one improve the scheme proposed in [27], which allows the eavesdropper to tap links multiple times?
- Can the coset coding scheme be improved?

Chapter 4

Appendix

4.1 Secret Sharing Example

Recall the secret sharing scenario in example 2 (pg 20): Imagine that a research facility contains a top-secret research lab and in that lab are deadly diseases, which may only be accessed when the following groups are present: the president and vice-president; the president and two committee members; the vice president and two committee members; or all four committee members of the research facility.

Given the above framework, we are able to then construct the access structure

$$\begin{aligned}\Gamma^{-} =&\{(1,2),(1,3,4),(1,3,5),(1,3,6),(1,4,5),(1,4,6),(1,5,6),(2,3,4),(2,3,5),\\ &(2,3,6),(2,4,5),(2,4,6),(2,5,6),(3,4,5,6)\},\end{aligned}$$

and adversary structure

$$\begin{aligned}R^{+} =&\{(1,3),(1,4),(1,5),(1,6),(2,3),(2,4),(2,5),(2,6),(3,4,5),(3,4,6),(3,5,6),\\ &(4,5,6)\}.\end{aligned}$$

Initially, we would like to determine whether or not there is an LSSS which will realize the access structure where $N = 6$, i.e., each person receives one share.

With $N = n = 6$, we obtain the corresponding Γ^- matrix

$$\begin{bmatrix}
1 & h_{1,1} & h_{1,2} & 0 & 0 & 0 & 0 \\
1 & h_{2,1} & 0 & h_{2,3} & h_{2,4} & 0 & 0 \\
1 & h_{3,1} & 0 & h_{3,3} & 0 & h_{3,5} & 0 \\
1 & h_{4,1} & 0 & h_{4,2} & 0 & 0 & h_{4,6} \\
1 & h_{5,1} & 0 & 0 & h_{5,4} & h_{5,5} & 0 \\
1 & h_{6,1} & 0 & 0 & h_{6,4} & 0 & h_{6,6} \\
1 & h_{7,1} & 0 & 0 & 0 & h_{7,5} & h_{7,6} \\
1 & 0 & h_{8,2} & h_{8,3} & h_{8,4} & 0 & 0 \\
1 & 0 & h_{9,2} & h_{9,3} & 0 & h_{9,5} & 0 \\
1 & 0 & h_{10,2} & h_{10,3} & 0 & 0 & h_{10,6} \\
1 & 0 & h_{11,2} & 0 & h_{11,4} & h_{11,5} & 0 \\
1 & 0 & h_{12,2} & 0 & h_{12,4} & 0 & h_{12,6} \\
1 & 0 & h_{13,2} & 0 & 0 & h_{13,5} & h_{13,6} \\
1 & 0 & 0 & h_{14,3} & h_{14,4} & h_{14,5} & h_{14,6}
\end{bmatrix}$$

and the associated R^+ matrix

$$\begin{bmatrix}
1 & 0 & g_{1,2} & 0 & g_{1,4} & g_{1,5} & g_{1,6} \\
1 & 0 & g_{2,2} & g_{2,3} & 0 & g_{3,5} & g_{2,6} \\
1 & 0 & g_{3,2} & g_{3,3} & g_{3,4} & 0 & g_{3,6} \\
1 & 0 & g_{4,2} & g_{4,3} & g_{4,4} & g_{4,5} & 0 \\
1 & g_{5,1} & 0 & 0 & g_{5,4} & g_{5,5} & g_{5,6} \\
1 & g_{6,1} & 0 & g_{6,3} & 0 & g_{6,5} & g_{6,6} \\
1 & g_{7,1} & 0 & g_{7,3} & g_{7,4} & 0 & g_{7,6} \\
1 & g_{8,1} & 0 & g_{8,3} & g_{8,4} & g_{8,5} & 0 \\
1 & g_{9,1} & g_{9,2} & 0 & 0 & 0 & g_{9,6} \\
1 & g_{10,1} & g_{10,2} & 0 & 0 & g_{10,5} & 0 \\
1 & g_{11,1} & g_{11,2} & 0 & g_{11,4} & 0 & 0 \\
1 & g_{12,1} & g_{12,2} & g_{12,3} & 0 & 0 & 0
\end{bmatrix}$$

The system of quadratic equation $GH^\top{=}0$ is found to be inconsistent when one runs it through the program (further on in the appendix). The code determines that the variables such as $h_{6,6}$ must be zero, which is a contradiction to the construction of the matrices.

In the substitution process, the quadratic equation $g_{4,4}h_{6,4}+1 = 0$ becomes $h_{6,6} = 0$ through the following steps:

- Substitute 1: $h_{6,4} = g_{1.6}h_{2,4}h_{6,6} + h_{2,4}$
- Result: $g_{4,4}g_{1,6}h_{2,4}h_{6,6} + g_{4,4}h_{2,4} + 1 = 0$

- Substitute 2: $g_{1,6} = -1/h_{4,6}$
- Result: $-g_{4,4}h_{2,4}h_{6,6} + g_{4,4}h_{2,4}h_{4,6} + h_{4,6} = 0$
- Substitute 3: $h_{4,6} = h_{6,6}$
- Result: $h_{6,6} = 0$

As a result, there is no linear secret sharing scheme which realizes the given access structure for $N = n = 6$. Thus, we move on to the next system and set $N = n + 1 = 7$. In this case, we have two different scenarios: the president (vice-president) receives two shares or a committee member receives two shares. Hence, we obtain the following when the president receives two shares:

$$\begin{bmatrix}
1 & h_{1,1}a & h_{1,1}b & h_{1,2} & 0 & 0 & 0 & 0 \\
1, & h_{2,1}a & h_{2,1}b & 0 & h_{2,3} & h_{2,4} & 0 & 0 \\
1 & h_{3,1}a & h_{3,1}b & 0 & h_{3,3} & 0 & h_{3,5} & 0 \\
1 & h_{4,1}a & h_{4,1}b & 0 & h_{4,2} & 0 & 0 & h_{4,6} \\
1 & h_{5,1}a & h_{5,1}b & 0 & 0 & h_{5,4} & h_{5,5} & 0 \\
1 & h_{6,1}a & h_{6,1}b & 0 & 0 & h_{6,4} & 0 & h_{6,6} \\
1 & h_{7,1}a & h_{7,1}b & 0 & 0 & 0 & h_{7,5} & h_{7,6} \\
1 & 0 & 0 & h_{8,2} & h_{8,3} & h_{8,4} & 0 & 0 \\
1 & 0 & 0 & h_{9,2} & h_{9,3} & 0 & h_{9,5} & 0 \\
1 & 0 & 0 & h_{10,2} & h_{10,3} & 0 & 0 & h_{10,6} \\
1 & 0 & 0 & h_{11,2} & 0 & h_{11,4} & h_{11,5} & 0 \\
1 & 0 & 0 & h_{12,2} & 0 & h_{12,4} & 0 & h_{12,6} \\
1 & 0 & 0 & h_{13,2} & 0 & 0 & h_{13,5} & h_{13,6} \\
1 & 0 & 0 & 0 & h_{14,3} & h_{14,4} & h_{14,5} & h_{14,6}
\end{bmatrix}$$

$$\begin{bmatrix}
1 & 0 & 0 & g_{1,2} & 0 & g_{1,4} & g_{1,5} & g_{1,6} \\
1 & 0 & 0 & g_{2,2} & g_{2,3} & 0 & g_{3,5} & g_{2,6} \\
1 & 0 & 0 & g_{3,2} & g_{3,3} & g_{3,4} & 0 & g_{3,6} \\
1 & 0 & 0 & g_{4,2} & g_{4,3} & g_{4,4} & g_{4,5} & 0 \\
1 & g_{5,1}a & g_{5,1}b & 0 & 0 & g_{5,4} & g_{5,5} & g_{5,6} \\
1 & g_{6,1}a & g_{6,1}b & 0 & g_{6,3} & 0 & g_{6,5} & g_{6,6} \\
1 & g_{7,1}a & g_{7,1}b & 0 & g_{7,3} & g_{7,4} & 0 & g_{7,6} \\
1 & g_{8,1}a & g_{8,1}b & 0 & g_{8,3} & g_{8,4} & g_{8,5} & 0 \\
1 & g_{9,1}a & g_{9,1}b & g_{9,2} & 0 & 0 & 0 & g_{9,6} \\
1 & g_{10,1}a & g_{10,1}b & g_{10,2} & 0 & 0 & g_{10,5} & 0 \\
1 & g_{11,1}a & g_{11,1}b & g_{11,2} & 0 & g_{11,4} & 0 & 0 \\
1 & g_{12,1}a & g_{12,1}b & g_{12,2} & g_{12,3} & 0 & 0 & 0
\end{bmatrix}$$

We similarly can construct G and H for the scenario of when a committee member receives two shares. Concerning both of these cases, one finds that certain variables must equal zero in order for the system $GH^\top = 0$ to hold. Thus, there is no solution over any field, resulting in the fact that there is no linear secret sharing scheme which computes the access structure for $N = n + 1 = 7$.

Again, we add one to the number of shares dealt and obtain $N = n + 2 = 8$. We have already briefly discussed the result in the content of this paper, but now we can go into the details.

Let $N = n + 2 = 8$. Then we have four choices:

1. The vice-president or president receives 3 shares.
2. The vice-president and president each receive an extra share.
3. Two of the chairmen receive an extra share.
4. A chairmen receives 2 shares.

Using the concept of the weighting system, we find 2.) to be more likely as it resembles the multiple assignment scheme. Using option 2, we find that there is a solution for a big enough field $\mathbb{F}_q$. However, due to the complexity of the system, we revert to the multiple assignment scheme. Let players 1 and 2 each receive two shares while the rest of the players (3-6) each receive only one share.

Let the number of shares required to obtain the secret be 4. Then we have a (4,8)-threshold. We can easily see that with the chosen weights, one obtains the access structure:

$$\Gamma^- = \{(1,2),(1,3,4),(1,3,5),(1,3,6),(1,4,5),(1,4,6),(1,5,6),(2,3,4),(2,3,5),$$
$$(2,3,6),(2,4,5),(2,4,6),(2,5,6),(3,4,5,6)\}$$

which is the desired access structure. Also, note that the adversary structure using the

(4,8)-threshold would give us

$$R^+ = \{(1,3),(1,4),(1,5),(1,6),(2,3),(2,4),(2,5),(2,6),(3,4,5),(3,4,6),(3,5,6),$$
$$(4,5,6)\}.$$

Thus, we may conclude that the linear secret sharing scheme (4,8)-threshold computes the appropriate access structure for the given scenario and therefore, the quadratic system $GH^\top = 0$ has a solution over a big enough field $\mathbb{F}_q$.

If one returns to the four choices presented earlier, it may be found that the other three choices are not realizable by an LSSS due to similar results of the previous scenarios where $N = n$ and $N = n + 1$.

4.2 Code for Narrowing Choices (Sage)

The following code was compiled for the situation where $N = n + 2$ and players 1 and 2 receive two shares each. There were no contradictions found. However, due to the vast number of fields and extension fields, it did not yield any concrete solutions.

```
from sage.all import *
import operator
varstr = (reduce(operator.add, ['greduce(operator.add, ['h
RR = PolynomialRing(QQ, 116, varstr)
Rvars = RR.gens()
class conv(object):
        def __init__(self):
        for i in range(116):
                self.__dict__["_"+str(i)] = Rvars[i]
R = conv()
```

$$G = matrix([[1, 0, 0, R.\_0, R.\_1, 0, R.\_2, R.\_3, R.\_4],$$
$$[1, 0, 0, R.\_5, R.\_6, R.\_7, 0, R.\_8, R.\_9],$$
$$[1, 0, 0, R.\_10, R.\_11, R.\_12, R.\_13, 0, R.\_14],$$
$$[1, 0, 0, R.\_15, R.\_16, R.\_17, R.\_18, R.\_19, 0],$$
$$[1, R.\_20, R.\_21, 0, 0, 0, R.\_22, R.\_23, R.\_24],$$
$$[1, R.\_25, R.\_26, 0, 0, R.\_27, 0, R.\_28, R.\_29],$$
$$[1, R.\_30, R.\_31, 0, 0, R.\_32, R.\_33, 0, R.\_34],$$
$$[1, R.\_35, R.\_36, 0, 0, R.\_37, R.\_38, R.\_39, 0],$$
$$[1, R.\_40, R.\_41, R.\_42, R.\_43, 0, 0, 0, R.\_44],$$
$$[1, R.\_45, R.\_46, R.\_47, R.\_48, 0, R.\_49, 0, 0],$$
$$[1, R.\_50, R.\_51, R.\_52, R.\_53, R.\_54, 0, 0, 0],$$
$$[1, R.\_55, R.\_56, R.\_57, R.\_58, 0, 0, R.\_59, 0]]);$$

$$H = matrix([[1, R.\_60, R.\_61, 0, 0, R.\_62, R.\_63, 0, 0],$$
$$[1, R.\_64, R.\_65, 0, 0, R.\_66, 0, R.\_67, 0],$$
$$[1, R.\_68, R.\_69, 0, 0, R.\_70, 0, 0, R.\_71],$$
$$[1, R.\_72, R.\_73, 0, 0, 0, R.\_74, R.\_75, 0],$$
$$[1, R.\_76, R.\_77, 0, 0, 0, R.\_78, 0, R.\_79],$$
$$[1, R.\_80, R.\_81, 0, 0, 0, 0, R.\_82, R.\_83],$$
$$[1, 0, 0, R.\_84, R.\_85, R.\_86, R.\_87, 0, 0],$$
$$[1, 0, 0, R.\_88, R.\_89, R.\_90, 0, R.\_91, 0],$$
$$[1, 0, 0, R.\_92, R.\_93, R.\_94, 0, 0, R.\_95],$$
$$[1, 0, 0, R.\_96, R.\_97, 0, R.\_98, 0, R.\_99],$$
$$[1, 0, 0, R.\_100, R.\_101, 0, R.\_102, R.\_103, 0],$$
$$[1, 0, 0, R.\_104, R.\_105, 0, 0, R.\_106, R.\_107],$$
$$[1, 0, 0, 0, 0, R.\_108, R.\_109, R.\_110, R.\_111],$$
$$[1, R.\_112, R.\_113, R.\_114, R.\_115, 0, 0, 0, 0]])$$

```
K = G*H.transpose()

def find_linear_monomial(eqns, Rvars):
        (var,val, i, eq) = (None,None,None,None)
        for i in xrange(len(eqns)):
                if eqns[i] == 0:
                        continue
                coeffs = [ eqns[i].coefficient(Rvars[j]) for j in range(len(Rvars)) ]
                degs = [c.degree() for c in coeffs]
```

```
                try:
                        ind = degs.index(0)
                        var = Rvars[ind]
                        other = (eqns[i] - var*coeffs[ind])
                        val = -1*coeffs[ind]*other
                        eq = eqns[i]
                        break
                except:
                        pass
                if var: print 'found LINEAR eq                    return (var, val, i, eq)

def find_k_monomial(eqns, k):
        (var, val, ind, eq) = (None, None, None, None)
        for i in xrange(len(eqns)):
                if var: break;
                if len(eqns[i].monomials()) > k: continue;
                for v in eqns[i].variables():
                        if eqns[i].degree(v) ==1:
                                var = v
                                coeff = eqns[i].coefficient(var:1)
                                # make sure we only divide by a single monomial
                                if len(coeff.monomials()) !=1:
                                        var = None
                                        continue
                                val = -(eqns[i]-var*coeff)/coeff
                                ind = i
                                eq = eqns[i]
                                if val ==0:
                                        print ind
                                        (var,val, ind, eq) = find_k_monomial([coeff],k)
                                        (ind, eq) = (i, eqns[i])
                                if var:
                                        break
        if var: print 'found eq          return (var, val, ind, eq)

if__name__ == '__main__':
        eqns = reduce(operator.add, [list(row) for row in K])
        subs_count = 0
        k=2
        while k < 15:
                var = None
                # first find a variable that appears only in a linear monomial
                (var, val, i, eq) = find_linear_monomial(eqns, Rvars)
                # no linear monomial – look for a degree 1 appearing in fewer than k terms
                if not var:
                        (var, val, i, eq) = find_k_monomial(eqns, k)
```

```
if var:
        subs_count += 1
        for j in xrange(len(eqns)):
                if eqns[j] == 0:
                        continue
                #print i,j
                eqns[j] = eqns[j].substitute(var:val).numerator().change_ring(QQ)
        else:
                print "vars left:%d" % (116-subs_count)
                k += 1
```

Bibliography

[1] A. Beimel, A. Gal, and M. Paterson. Lower bounds for monotone span programs. *Computational Complexity 6*, pp. 29-45, 1996. Preliminary version appeared in *Proc. 36th IEEE FOCS*, pp. 674-681, 1995.

[2] A. Beimel. Secret-sharing schemes: a survey. In *Chee, Y.M., Guo, Z., Ling, S., Shao, F., Tang, Y., Wang, H., Xing, C. (eds.) IWCC 2011. LNCS, vol. 6639*, pp. 11-46. Springer, Heidelberg, 2011.

[3] G. Blakley. Safeguarding Cryptographic Keys. In *Proc. AFIPS 1979 National Computer Conference, New York, NY*, pp. 313-317, June 1979.

[4] N. Cai and R. W. Yeung. Network coding and error correction. In *Proc. 2002 IEEE Inform. Theory Workshop, Bangalore, India*, pp. 119-122, Oct. 20-25, 2002.

[5] H. Chen, R. Cramer, S. Goldwasser, R. de Haan, and V. Vaikuntanathan. Secure computation from random error correcting codes. *Advances in cryptologyEUROCRYPT 2007, Lecture Notes in Comput. Sci., 4515, Springer, Berlin*, pp. 291-310, 2007.

[6] Y. Chen, C. Tang, and S. Gao. Several Properties of Monotone Span Programs. Preprint. 2013.

[7] J. Feldman, T. Malkin, C. Stein, and R. A. Servedio. On the capacity of secure network coding. In *Proc. 42nd Annual Allerton Conference on Commun., Control, and Comput.*, 2004.

[8] M. Gadouleau and Z. Yan. Properties of codes with the rank metric. In *Proc. IEEE Globecom 2006, San Francisco, CA*, 2006.

[9] E. Gabidulin. Rank-metric codes and applications. *Moscow Institute of Physics and Technology (State University).*

[10] A. Gal and P. Pudlak. Monotone complexity and the rank of matrices. *Inform. Process. Lett.*, 87:321-326, 2003.

[11] M. Ito, A. Saito, and T. Nishizeki. Secret sharing schemes realizing general access structure. In *Proc. of the IEEE Global Telecommunication Conf., Globecom 87*, pages 99-102, 1987. Journal version: Multiple assignment scheme for sharing secret. *J. of Cryptology*, 6(1):15-20, 1993.

[12] M. Karchmer and A. Wigderson. On span programs. In *Proc. of 8th IEEE Symp. Structure in Complexity Theory*, pp.102-111, 1993.

[13] E. D. Karnin, J. W. Greene and M. E. Hellman. On secret sharing systems. In *IEEE Transactions on Information Theory 29*, pp. 35-41, 1983.

[14] A. Khaleghi, D. Silva, and F. R. Kschischang. Subspace codes. In *proceedings of 12th IMA International Conference on Cryptography and Coding*, pp. 1-21, 2009.

[15] A. Khisti, D. Silva, and F. Kschischang. Secure-broadcast codes over linear-determintic channels. In *IEEE International Symposium on Information Theory*, May 2010.

[16] R. Koetter and F. R. Kschischang. Coding for errors and erasures in random network coding. *IEEE Trans. Inf. Theory, vol. 54, no. 8*, pp. 3579-3591, Aug. 2008.

[17] J. Kurihara, T. Uyematsu, and R. Matsumoto. New parameters of linear codes expressing security performance of universal secure network coding. In *Proc. Proc. 50th Annu. Allerton Conf. Communication, Control, and Computing, Monticello, IL, USA, Oct. 2012*. [Online]. Available: arXiv:1207.1936

[18] R. Matsumoto and M. Hayashi. Secure multiplex network coding. In *Proc. 2011 Int. Symp. Network Coding, Beijing, China, Jul. 2011*, pp. 16. [Online]. Available: arXiv:1102.3002.

[19] R. Matsumoto and M. Hayashi. Universal strongly secure network coding with dependent and non-uniform messages. arXiv:1111.4174, 2011.

[20] K. M. Martin. Challenging the adversary model in secret sharing schemes. In *Proc. of the Contact Forum Coding Theory and Cryptography II, September 21, 2007, at The Royal Flemish Academy of Belgium for Science and the Arts, Brussels, Belgium*, pp. 45-64, 2008.

[21] V. Nikov, S. Nikova, and B. Preneel. On the size of monotone span programs. In *Proc. SCN 2004, LNCS vol. 3352*, pp. 249-262, 2005.

[22] H. Ozadam, F. Ozbudak, and Z. Saygi. Secret sharing schemes and linear codes. In *Information Secrity and Cryptology Confernece with International Participation*, pp. 101-106, 2007.

[23] C. Padro. Lecture Notes in Secret Sharing. In *Cryptology ePrint Archive, Report 2012/674*, 2012. [Online]. Available: http://eprint.iacr.org/2012/674.pdf

[24] S. Y. E. Rouayheb and E. Soljanin. On wiretap networks II. In *Proc. IEEE Int. Symp. Information Theory, Nice, France, Jun. 24-29, 2007*, pp. 551-55, 2007.

[25] S. Y. El Rouayheb, E. Soljanin, and A. Sprintson. Secure network coding for wiretap networks of type II. In *IEEE Trans. Inf. Theory, vol. 58, no. 3*, pp. 1361-1371, Mar. 2012. [Online]. Available: arXiv:0907.3493 [cs.IT]

[26] A. Shamir. How to share a secret. In *Communications of the ACM 22*, pp. 612-613, 1979.

[27] E. Shioji, R. Matsumoto, and T. Uyematsu. Vulnerability of MRD-code-based universal secure network coding against stronger eavesdroppers. In *IEICE Trans. Fundamentals, vol. E93-A, no. 11*, pp. 2026-2033, Nov. 2010.

[28] M. Jafari Siavoshani, C. Fragouli, and S. N. Diggavi. Subspace properties of network coding and their applications. In *IEEE Transactions on Information Theory, vol. 58, no. 5*, pp. 2599-2619, May 2012.

[29] D. Silva. Error control for network coding. *Ph.D. dissertation, University of Toronto, Toronto, Canada*, 2009.

[30] D. Silva. Security for wiretap networks via rank-metric codes. In *Proc. IEEE Int. Symp. Information Theory, Toronto, Canada, Jul. 6-11, 2008*, pp. 176-180, 2008.

[31] D. Silva and F. R. Kschischang. On metrics for error correction in network coding. In *IEEE Trans. Inf. Theory, vol. 55, no. 12*, pp. 5479- 5490, 2009.

[32] D. Silva and F. R. Kschischang. Universal secure errorcorrecting schemes for network coding. In *IEEE Int. Symp. Information Theory, 2010.* [Online]. Available: http://arxiv.org/abs/1001.3387

[33] D. Silva and F. R. Kschischang. Universal secure network coding via rank-metric codes. arXiv:0809.3546v1, 2008.

[34] D. Silva and F. R. Kschischang. Universal weakly secure network coding. In *Proc. 2009 IEEE Information Theory Workshop, Volos, Greece, Jun. 2009*, pp. 281-285, 2009.

[35] D. Silva and F. R. Kschischang. Using rank-metric codes for error correction in random network coding. In *Proc. IEEE Int. Symp. Inofrmation Theory, Nice, France, Jun. 24-29*, pp. 796-800, 2007.

[36] D. Silva, F. R. Kschischang, and R. Koetter. A rank-metric approach to error-control in random network coding. In *IEEE Trans. on Inform. Theory.* [Online]. Available: http://arxiv.org/abs/0711.0708, 2007.

[37] M. van Dijk. A linear construction of perfect secret sharing schemes. In *Proc. of Eurocrypt'94, Springer-Verlag, Berlin, LNCS 950*, pp. 23-34, 1995.

[38] H. Yao, D. Silva, S. Jaggi, and M. Langberg. Network codes resilient to jamming and eavesdropping. In *IEEE Int. Symp. Network Coding*, 2010. [Online]. Available: http://arxiv.org/abs/1001.3714

[39] Z. Zhang. Network error correction coding in packetized networks. In *Proc. 2006 IEEE Inform. Theory Workshop, Chengdu, China*, pp. 433-437, Oct. 22-26, 2006.