Some Applications of Coding Theory in Cryptography

EE 605: Endsem Project

Arush Tadikonda, Ishan Kapnadak

IIT Bombay

Table of contents

Introduction

(S,T) Threshold Schemes

We begin by introducing a scheme for secret sharing known as the (S, T) threshold scheme.

Definition 1.1

An (S, T) *threshold scheme* for secret sharing is a method in which a secret is transformed into S shares such that

1. the knowledge of any T shares reveals the secret, but
2. the knowledge of any $T - 1$ or fewer shares reveals no information about the secret.

Here by knowledge of a share, we mean knowledge of its value as well as its position in the list of shares.

(S,T) Threshold Schemes using polynomial interpolation

An elegant formulation of the (S, T) threshold schemes is via polynomial interpolation. We may assume that the secret is an integer, D. Suppose we wish to design a (n, k) threshold scheme. We pick a prime p that is larger than both D and n. We now pick a random $k-1$-degree polynomial $q(x) \in \mathbb{Z}_p[x]$, defined as

$$q(x) = a_0 + a_1 x + \cdots + a_{k-1} x^{k-1}$$

where $a_0 = D$. We further evaluate $D_i = q(i)$ for all $i = 1, \ldots, n$. The list of shares is now $\{D_1, \ldots, D_n\}$. Given any subset of k shares, we can determine the polynomial $q(x)$ exactly using polynomial interpolation, and thus D can be evaluated as $q(0)$. The knowledge of $k-1$ or fewer shares however does not reveal any information since we need at least k shares to interpolate.

(S,Γ) Access Structure Schemes

In some cases, we want some users to have a greater privilege of access than others. This is most commonly formulated using an (S, Γ) access structure scheme, as defined below.

Definition 1.2

An (S, Γ) *access structure scheme* for secret sharing is a method in which a secret is transformed into S shares such that

1. the knowledge of the shares in any set in Γ reveals the secret, but
2. the knowledge of the shares of a set having no subset (not necessarily proper) in Γ reveals no information about the secret.

(S,Γ) Access Structure Schemes - An Example

As an example, consider that a bank has 4 employees, A, B, C, and D. A is the president whereas B,C,D are all vice-presidents. A possible access structure scheme for such a case may be defined as $S = \{A, B, C, D\}$ and $\Gamma = \{\{A, B\}, \{A, C\}, \{A, D\}\}$. Intuitively, such a scheme gives high privilege to the president since the president can access the secret along with any one other vice-president, but the vice-presidents, even together, cannot access the secret without knowledge of the president's share.

To analyse what access structures can be realised using q-ary (n, k) codes, we first look at the concept of minimal codewords.

Minimal Codewords

Minimal Codewords

A q-ary n-tuple x is said to cover a q-ary n-tuple x' if in every coordinate where x' is non-zero, x is also non-zero. For example, the 3-ary 4-tuple $(0, 1, 2, 1)$ covers $(0, 2, 1, 0)$

Definition 2.1 (Minimal Codeword)

A codeword x in a q-ary (n, k) code is said to be *minimal* if

1. x is a non-zero codeword whose leftmost non-zero component is a 1, and
2. x covers no other codeword whose leftmost non-zero component is a 1.

We have the following interesting property of minimal codewords.

Proposition 2.2

Every non-zero codeword in a q-ary (n, k) linear code can be written as a linear combination of those minimal codewords that it covers.

Proof of Proposition 2.2

Proof.

Let x be a non-zero codeword. If x is minimal, we are done. Else, x covers a minimal codeword (say x_1), and thus, there is a constant c_1 such that $x - c_1x_1$ has Hamming weight strictly less than x. If $x - c_1x_1$ is 0 or a minimal codeword, we are again done. Else, $x - c_1x_1$ covers another minimal codeword (say x_2), and thus $x - c_1x_1 - c_2x_2$ has Hamming weight strictly less than $x - c_1x_1$. Continuing this way, we must eventually arrive at the zero codeword since we decrease the Hamming weight at each step. As a result, we have

$$x = c_1x_1 + \cdots + c_nx_n$$

for some constants $c_1, \ldots, c_n$. Moreover, $x_1, \ldots, x_n$ are exactly those minimal codewords that x covers. □

Access Structure Scheme using Minimal Codewords

Every q-ary (n, k) linear code C with no idle components (i.e. no components that are 0 in all codewords) forms an (S, Γ) access structure scheme with $S = n - 1$. The secret is taken as the first digit of the codeword, x_1 and the digits in some other specified $k - 1$ components are independently chosen at random. These k components together form an *information set* and thus we are able to compute the entire codeword $x = (x_1 \cdots x_n)$. The share list as taken as $\{x_2, \ldots, x_n\}$, which gives us $S = n - 1$.

Proposition 2.3

For the above q-ary (n, k) linear code, the access set Γ consists of precisely those share sets corresponding to those minimal codewords of $C^{\perp}$ that have 1 as their first component. The share set specified by such a minimal codeword contains the share x_i $(2 \leq i \leq n)$ iff the i^{th} component of the minimal codeword is non-zero.

An Example

The proof for Proposition 2.3 follows directly from 2.2, and hence we omit the proof here. As an illustrative example, consider a code C over $\mathbb{F}_2$ having the following parity check matrix (which is also the generator matrix for the dual code $C^{\perp}$)

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

It is easy to see that the only minimal codewords of $C^{\perp}$ are 11100, 11010, 11001, 00110, 00101, and 00011. The minimal codewords with 1 as their first component are 11100, 11010, and 11001. This corresponds to $\Gamma = \{\{x_2, x_3\}, \{x_2, x_4\}, \{x_2, x_5\}\}$. Letting x_2, x_3, x_4, x_5 be the shares of A, B, C, D respectively, we see that we are able to emulate the president and vice-president example discussed previously.

Orthogonal Arrays

Orthogonal Arrays

We collect some definitions and results about orthogonal arrays that will help us in further sections.

Lemma 3.1

If the $k \times n$ q-ary matrix G is the generator matrix of a q-ary linear (n, k) code C, then the minimum distance d of C is the smallest number of columns that can be removed from G to yield a matrix with rank less than k.

Lemma 3.2

If the $k \times n$ q-ary matrix G is the generator matrix of a q-ary linear (n, k) code C, then the minimum distance $d^\perp$ of the dual code $C^\perp$ is the smallest number t of columns of G that form a $k \times t$ matrix with rank less than t.

We omit the proofs here due to a paucity of time.

Orthogonal Arrays

Definition 3.3

An *orthogonal array* $OA_\lambda(t, n, q)$ is a $\lambda q^t \times n$ array such that in every choice of t columns ($t \geq 1$), each of the q^t possible q-ary t-tuples appears in exactly λ rows. Orthogonal arrays with distinct rows (or equivalently, $\lambda = 1$) are said to be *simple*.

Note that for $t \geq 2$, an $OA_\lambda(t, n, q)$ is also an $OA_{\lambda q}(t-1, n, q)$, and thus, the parameter of interest is the maximum t for which a given array is orthogonal. The following is an example of a simple orthogonal array $OA_1(2, 3, 2)$.

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

Proposition 3.4

The maximum t for which a $q^k \times n$ q-ary array, whose rows are codewords of a q-ary linear (n, k) code C with $k < n$ and $d^\perp > 1$ is an orthogonal array $OA_\lambda(t, n, q)$ (necessarily simple), is $t = d^\perp - 1$.

Proof.

Let $\tilde{G}$ be the submatrix of G formed by some choice of t columns. Then, $\tilde{x} = u\tilde{G}$ is a mapping of the information k-tuple u to those t components of the codeword that correspond to the chosen columns. This mapping is surjective if and only if $\tilde{G}$ has rank t. Assuming the map is surjective, linearity guarantees that the equation $\tilde{x} = u\tilde{G}$ has the same number of solutions for each possible t-tuple, $\tilde{x}$. (Note that if $\tilde{x}_1 = u_1\tilde{G}$ and $\tilde{x}_2 = u_2\tilde{G}$, then $\tilde{x}_1 + \tilde{x}_2 = (u_1 + u_2)\tilde{G}$). The result now follows from Lemma 3.2. □

Large Set of Simple Orthogonal Arrays

Definition 3.5

A *large set of simple orthogonal arrays* $OA_\lambda(t, n, q)$ is a collection of $\frac{q^{n-t}}{\lambda}$ such arrays with the property that every q-ary n-tuple appears as a row in exactly one of the arrays of the collection.

If S is the set of rows of a simple orthogonal array $OA_\lambda(t, n, q)$ and x is a q-ary n-tuple, then it is easy to see that the translated set $x + S$ is also a simple orthogonal array. Moreover, if S corresponds to a linear code C, then $x + S$ is a coset of C. Since cosets are either identical or disjoint, we may form a large set of simple orthogonal arrays by vertically stacking S along with all its $q^{n-k} - 1$ non-zero translates (or equivalently, by vertically stacking all q^{n-k} cosets of C).

Resilient Functions

Resilient Functions

A common objective of cryptographers is to ensure that local constraints on the input result in no constraints on the output. This idea is tied with the notion of resilient functions. Intuitively, resilient functions are functions whose output appears random even if some portion of the input is revealed or fixed. We define resilient functions more formally below.

Definition 4.1

A function $f\colon \mathbb{F}_q^n \to \mathbb{F}_q^{n-k}$ $(1 < k < n)$ is said to be *t-resilient* if, for every choice of t of the input digits, when the value of these input digits are fixed and the values of the remaining $n - t$ input digits are chosen uniformly at random, then all $n - k$ of the output digits are uniformly random.

Resilient Functions and Orthogonal Arrays

Lemma 4.2

A function $f\colon \mathbb{F}_q^n \to \mathbb{F}_q^{n-k}$ is t-resilient if and only if the collection of q^{n-k} sets $f^{-1}(y_1, \ldots, y_{n-k})$ determined by all possible outputs $[y_1, \ldots, y_{n-k}]$ is a large set of simple orthogonal arrays $OA_\lambda(t, n, q)$.

Interestingly, we have already discussed an example of a t-resilient function, while discussing syndrome decoding. We formalise this now.

Proposition 4.3

If H is an $(n-k) \times n$ parity check matrix of a q-ary (n, k) linear code C, then the maximum t such that the linear function $x \mapsto xH^\intercal$ is t-resilient is $t = d^\perp - 1$, where $d^\perp$ is the minimum distance of the dual code $C^\perp$.

How It All Connects

We now look at a syndrome decoding example to illustrate how beautifully the concepts of orthogonal arrays, resilient functions, and syndromes connect. We consider Q8 of HW1 which talks about a $(6, 3)$ binary linear code that has the following parity check matrix

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

We calculate the generator matrix as

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

How It All Connects (2)

For the previous code, we have the following coset decomposition.

000000	100111	010101	001011	110010	101100	011110	111001
000001	100110	010100	001010	110011	101101	011111	111000
000010	100101	010111	001001	110000	101110	011100	111011
000100	100011	010001	001111	110110	101000	011010	111101
001000	101111	011101	000011	111010	100100	010110	110001
010000	110111	000101	011011	100010	111100	001110	101001
100000	000111	110101	101011	010010	001100	111110	011001
000110	100001	010011	001101	110100	101010	011000	111111

The first row consists of all the codewords of the code. We leave it as a simple exercise to verify that this indeed forms an orthogonal array. Moreover, each row is the first row translated by the error vector (which is the coset leader). In line with the discussion on orthogonal arrays, we have that each row is a translate of an orthogonal array and thus, is an orthogonal array itself. Thus, the above collection forms a large set of simple orthogonal arrays.

How It All Connects (3)

Now, the syndrome decoding table is given by

Coset Leader	Syndrome
000000	000
000001	001
000010	011
000100	111
001000	010
010000	110
100000	101
000110	100

Thus, the inverse image of a given syndrome consists of all the codewords of the code translated by the corresponding coset leader. Equivalently, this is the row in the coset decomposition corresponding to the coset leader for the given syndrome. Moreover, since this coset decomposition forms a large set of simple orthogonal arrays, the syndrome map $f\colon \mathbb{F}_2^6 \to \mathbb{F}_2^3$ is resilient.

Summary

Summary

Our discussion can essentially be divided into two components. We first talked about secret sharing schemes and how they connect with minimal codewords. Then, we talked about orthogonal arrays and resilient functions. As an illustrative example, we discussed how these concepts come into play in the familiar territory of syndrome decoding that we have already encountered before in the course. There were two sections from the main paper that we could not cover due to paucity of time. These include local randomisation, and quantifying non-linearity of Boolean functions. We encourage interested peers to refer to the original paper to look up these sections. Thank you!