Intuitive Understanding of Quantum Computation and Post-Quantum Cryptography (Chapter 5)

Nguyen Thoi Minh Quan *

5 McEliece and Niederreiter's code-based cryptosystem

McEliece code-based cryptosystem [1], [2], [3], [4] has resisted 4 decades of cryptanalysis. Its security is based on the hardness of certain error-correcting problems which we will study in the following section. It's not popular because its public key is large, but its encryption and decryption are fast. Depending on the context of cryptographic protocols, this limitation may or may not be an issue.

5.1 Error correcting code

Let's say we want to transfer a message m of $k = 1$ bit length over the wire. The message is 0 or 1. However, there may be an error in the transmission channel which might cause the bit to be flipped. The problem is how to detect whether there was an error? Another related problem is how can we correct the error?

If we only send the message then there is no way to solve the above problems. We have to send extra bits to help detect the error or to correct it. Instead of sending the bit 0 (correspondingly 1), we send 00 (correspondingly 11) and if the transmission channel only has at most $t = 1$ bit of error then we can detect it. Why? If the sender sends 00 and $t = 1$ bit of error happens then receiver will receive 10 (if the 1st bit is flipped) or 01 (if the 2nd bit is flipped). Similar situation happens if the sender sends 11. I.e., if the receiver receives 01 or 10, the receivers knows that there was an error and if the receiver receives 00 or 11 it knows that there was no error. This solves the problem of error detection, but it doesn't solve the problem of error correction problem. It's not hard to convince ourselves that if we encode 0 as 000 and 1 as 111 and if the transmission channel has at most $t = 1$ bit errors then we can correct the flipped bit.

Let's pause for a moment to introduce the terminologies. The sender wants to send a message m of length k (e.g. 0 or 1). The sender encodes the message into a different form called codeword c of length $n, n > k$ (e.g. 000 or 111). The transmission channel may introduce t bit error e into c, i.e., the receiver receives $\hat{c} = c + e$ where e has at most t bits 1. The receiver decodes t bit error from $\hat{c}$ to get c and deduce m from c. If the receiver can correctly fix up to t bit errors then we say we have t-error correcting code.

In this article, we're only concerned with binary (aka F_2) linear code, i.e., we have a generator matrix of size $k \times n$, each matrix entry is in F_2 and the codeword is $c = mG$. A binary message m of size $1 \times k$ will produce a codeword of size $1 \times n$. Why is it called linear code? It's because mG is the linear combination of rows of matrix G. In this case, we call the set C of all codewords c a (n, k)-code. A matrix H of size $(n - k) \times n$ whose null space is C (i.e., $Hc^t = 0$) is called parity-check matrix. Note that the kernel equation $Hc^t = 0$ is an alternative way to define codes besides using generator matrix. Let's see why H is called check matrix. We have $H\hat{c}^t = H(c+e)^t = Hc^t + He^t = 0 + He^t = He^t$, i.e., if there was no error ($e = 0$) then $H\hat{c}^t = 0$. The value $H\hat{c}^t = He^t$ is called the syndrome of $\hat{c}$.

*https://www.linkedin.com/in/quan-nguyen-a3209817, https://scholar.google.com/citations?user=9uUqJ9IAAAAJ, https://github.com/cryptosubtlety, msuntmquan@gmail.com

5.2 Goppa codes

McEliece original cryptosystem uses binary Goppa codes [5] and it has resisted cryptanalysis for 4 decades. While it's possible to replace binary Goppa codes with other error-correcting codes, a few proposals using alternative codes have been broken over time. Therefore, in this section, we briefly introduce binary Goppa codes.

We will work in finite field $F_n, n = 2^m$. Fix a list of n elements $a_1, a_2, \cdots, a_n$ in F_n. Choose a degree-t irreducible polynomial $g(x) \in F_n[x]$. As $g(x)$ is irreducible, $F_n[x]/g(x)$ (i.e. polynomial where coefficients are in F_n and all operations are $\mod g(x)$) forms a finite field and hence every element has an inverse. The binary Goppa code is defined as follow using kernel equation

$$\Gamma(a_1, a_2, \cdots, a_n, g) = \{c \in F_n^2 : \sum_i \frac{c_i}{x - a_i} = 0 \mod g(x)\}$$

I guess you're confused because in the previous section, we use matrix notation to define error-correcting code, but here we use polynomial. What is the relationship between polynomial form and matrix form?

Note that $\frac{1}{x-a_i}$ is a shortcut notation to denote the inverse of $x - a_i$ in the finite field $F_n[x]/g(x)$. There is an efficient algorithm to compute the inverse of $x - a_i$ and let's denote $g_i = \sum_{j=0}^{t-1} g_{i,j} x^j$ the inverse of $x - a_i$. Now, rewrite the kernel equation $\sum_i \frac{c_i}{x-a_i} = 0 \mod g(x)$ as follow

$$\begin{aligned}
\sum_{i=1}^{n} c_i \frac{1}{x - a_i} &= \sum_{i=1}^{n} c_i \sum_{j=0}^{t-1} g_{i,j} x^j \\
&= \sum_{j=0}^{t-1} \Big(\sum_{i=1}^{n} g_{i,j} c_i \Big) x^j \\
&= 0 \mod g(x)
\end{aligned}$$

The last equation means that x^j's coefficient $\sum_{i=1}^{n} g_{i,j} c_i$ must be zero for all $j = 0, \cdots, t-1$. I.e., we have a system of equations $\sum_{i=1}^{n} g_{i,j} c_i = 0, j = 0, \cdots, t-1$ where $c = (c_1, c_2, \cdots, c_n)$ is our codeword and that is our familiar matrix form of kernel equation.

5.3 McEliece's cryptosystem

McEliece's cryptosystem is based on the following observation. In binary Goppa codes, if the receiver/decoder knows the generator matrix G then it's easy to decode and correct errors in the transmission channel. However, it's difficult to solve the decoding problem for arbitrary (n, k) linear code. Therefore, McEliece cryptosystem generates a generator matrix, keeps it as private key and scrambles it to make it look random and use the scrambled version as the public key.

In details, the sender chooses $k \times n$ Goppa generator matrix G that can correct up to t errors, $k \times k$ binary non-singular matrix S, $n \times n$ permutation matrix P. The matrices S, P are used to hide the generator matrix G.

The public key is $G' = SGP$, the private key is (S, G, P).

To encrypt a message m of length k, choose a random error vector e that has t bits 1, compute the ciphertext $c = mG' + e$.

To decrypt c, compute $cP^{-1} = (mG' + e)P^{-1} = (mSGP + e)P^{-1} = (mS)G + eP^{-1}$. As eP^{-1} is just a permutation of e, it has t bits 1. Therefore, with the knowledge of G, the receiver can use an efficient decoding algorithm to deduce mS. To recover m, compute $(mS)S^{-1} = m$.

5.4 Niederreiter's cryptosystem

Niederreiter's [6], [4] cryptosystem is a variant of McEliece cryptosystem where it has the same security as McEliece cryptosystem. Recall that to define error-correcting code, we can either use generator matrix or use parity-check matrix. McEliece cryptosystem uses generator matrix while Niederreiter cryptosystem uses parity-check matrix.

In details, the sender chooses a $(n-k) \times n$ parity-check matrix H of Goppa codes that can correct up to t errors, a $(n-k) \times (n-k)$ binary non-singular matrix S, $n \times n$ permutation matrix P. The matrices S, P are used to hide the parity check matrix H.

The public key is $K = SHP$, the private key is (S, H, P).

To encrypt a message m of length n and has t bits 1, compute the ciphertext $c = Km^t$.

To decrypt, compute $S^{-1}c = S^{-1}SHPm^t = HPm^t = H(Pm^t)$. Pm^t is just a permutation of m so it has t bits 1. With the knowledge of parity check matrix H, the receiver can use an efficient decoding algorithm to compute Pm^t and hence $m^t = P^{-1}Pm^t$.

References

[1] Robert J McEliece. A public-key cryptosystem based on algebraic coding theory.

[2] Raphael Overbeck and Nicolas Sendrier. Code-based cryptography.

[3] Tanja Lange. Code-based cryptography. https://www.youtube.com/watch?v=EqRsel-rXac.

[4] Daniel J. Bernstein, Tanja Lange, and Christiane Peters. Attacking and defending the mceliece cryptosystem.

[5] V.D.Goppa. A new class of linear correcting code.

[6] H. Niederreiter. Knapsack-type cryptosystems and algebraic coding theory.