The report in course MTAT.07.022 Research Seminar in Cryptography, Spring 2017

Attacks and countermeasures for LDPC-based McEliece cryptosystem

Sofiya Demchuk

Supervised by Vitaly Skachek

Abstract

The McEliece cryptosystem is a public-key cryptosystem, which is based on a use of error-correcting codes. Goppa codes were originally used in the McEliece cryptosystem. Considering the fact that McEliece cryptosystem has a large size of the public key, it is not used in practice. Due to low transmission rates, it was suggested to use LDPC codes instead of Goppa codes. This work gives a brief overview of the McEliece system generally, including error-correcting codes, Goppa codes, turning to the topic of LDPC-based cryptosystem, existing attacks and countermeasures. The interest of researchers in McEliece cryptosystem increased with the advent of quantum computing as it could potentially resist adversaries with access to such computers.

Introduction

The McEliece cryptosystem is a public key cryptosystem, introduced by Robert McEliece in 1978 [1], which is similar to the Merkle-Hellman Knapsack cryptosystem. It takes an easy case of a non-deterministic polynomial-time hard problem and disguises it to look like the hard instance of the problem. The main characteristic features of this cryptosystem are:
• It has a large public key, but the cryptosystem has a good security reduction and low complexity algorithms for encryption and decryption.
• There is a trade-off in the McEliece cryptosystem between security and efficiency.

Traditionally, in order to describe the cryptosystem we use the example of communication of two friends, Bob and Alice. There is a public key known to everyone, moreover, the public key is based on the private key, but the last one should be impossible to recover and is only held by the recipient of the message. Hence, if Alice wants to send a secret message to Bob, Bob should first show his public key. After this Alice takes her public key and encrypts her message making it a cyphertext that is sent to Bob. Bob can decrypt this cyphertext with his private key.

In the McEliece cryptosystem, the problem is drawn from the theory of error-correcting codes. The basic idea of the McEliece system is to take one of the linear codes and disguise it so that the adversary (further Tom), when trying to decrypt a message, is forced to use syndrome decoding. Syndrome decoding is an efficient method of decoding a linear code over a noisy channel. Maximum-likelihood decoding of a general linear code is known to be an NP-complete problem. However, there are classes of linear codes which have very fast decoding algorithm that correct any number of errors up to a certain threshold. For example, Bob, who sets up the system, can remove the disguise and use the fast decoding algorithm.

McEliece suggested to use binary Goppa Codes, which are linear codes with a fast decoding algorithm, but any linear code with a good decoding algorithm can be used instead.

To explain how the McEliece system works, we will start with the example [6].

Let $C$ be a linear $(n, k)$-code with a fast decoding algorithm that can correct $t$ or fewer errors. Let $G$ be a generator matrix for $C$. To create the disguise, let $S$ be a $k \times k$ invertible matrix (the scrambler) and let $P$ be an $n \times n$ permutation matrix (i.e., having a single 1 in each row and column and 0's everywhere else). The matrix,

$$G' = SGP$$

is made public while $S, G$ and $P$ are kept secret by Bob. Alice divides her message into binary vectors of length $k$. If $x$ is such one block, she randomly constructs a binary vector of length $n$ and of weight $t$ (that is, she randomly places $t$ 1's in a zero vector of length $n$), call it $e$ and then she sends to Bob the vector

$$y = xG' + e.$$

Tom, upon intercepting this message, would have to find the nearest codeword to $y$ of the code generated by $G'$. This would involve calculating the syndrome of $y$ and comparing it to the syndromes of all the error vectors of weight $t$. As there are $\binom{n}{t}$ of these error vectors, good choices of $n$ and $t$ will make this computation infeasible. Bob, on the other hand, would calculate

$$yP^{-1} = (xG' + e)P^{-1} = xSG + eP^{-1} = xSG + e'$$

where $e'$ is a vector of weight $t$ (since $P^{-1}$ is also a permutation matrix). Bob now applies a fast decoding algorithm to strip off the error vector $e'$ and to obtain the codeword $(xS)G$. Now the vector $xS$ can be obtained by multiplying by $G^{-1}$ on the right (moreover, Bob can write $G$ in a standard form $[I_k A]$, and then $xS$ would just be the first $k$ positions of $xSG$ and this multiplication is not needed). Finally, Bob estimates $x$ by multiplying $xS$ on the right by $S^{-1}$. 

In McEliece's original paper it was suggested to use \( n = 1024, k = 524, t = 50 \), as it is the minimal value for security reason, which gives Tom more than \( 10^{80} \) syndromes to calculate.

**Background**

In order to protect data from random errors it is possible to add redundant symbols to the transmitted data vector according to certain rules. Thus, the receiver can recognize a limited number of errors and then correct these errors. More formally, we say that \( t \)-error correcting code is a code where any pattern of \( t \) or less errors can be corrected:

**Definition 1.** A linear \((n, k)\) code \( C \) over a field \( F \) is a vector subspace of \( F^n \) of dimension \( k \), where \( n \) is called the length of the code \( C \). The numbers \( n, k, d \) are called the parameters of the code \( C \).

The minimum distance of a code \( C \) is

\[
d(C) = \min \{ d(x, y) : x, y \in C, x \neq y \},
\]

where \( d(x, y) \) is the Hamming distance that is the number of positions where vectors \( x \) and \( y \) differ.

The most efficient codes will have the values of \( k \) and \( d \) as large as possible at the same time. Since there are \( n - k \) checks because codeword has \( n \) bits with \( k \) information, so \( k \) should be large relatively to \( n \). Similarly, it applies for the minimum distance \( d \), because we can correct up to \( \frac{d-1}{2} \) errors, so \( d \) should be large.

**Goppa codes**

We can briefly mention the original construction of the McEliece cryptosystem. The original variant of the cryptosystem is based on binary Goppa codes. Robert McEliece firstly used them with the parameters \( n = 1024, k = 524, \) and minimum distance \( d = 101 \) (at least), and thus the cryptosystem was able to correct \( t = 50 \) errors. Here are briefly described several statements for better understanding of the Goppa codes:

**Definition 2.** Let \( p \) be a prime number and \( k \in \mathbb{Z} \), where \( k > 0 \). \( GF(p^m) \) - is called the extension Galois field \( GF(p) \) of degree \( m \), where the Galois field of order \( q = p^k \) - number of elements in the field.

**Definition 3.** A polynomial over \( GF(p^m) \) is irreducible if it can’t be divided by polynomial over \( GF(p^m) \) with a smaller non-zero degree.

Goppa codes are determined by two objects:

- Goppa polynomial is a polynomial over \( GF(p^m) \) that
  \[
g(x) = g_0 + g_1x + \cdots + g_tx^t = \sum_{i=0}^{t} g_ix^i.
\]
and any \( g_i \in GF(p^m) \).

- And let \( L \) be a finite subset of the \( GF(p^m) \), and

\[
L = \{ \alpha_1, \ldots, \alpha_n \} \subseteq GF(p^m) \text{ and } g(\alpha_i) \neq 0 \text{ for all } \alpha_i \in L.
\]

**Definition 4.** A Goppa code \( \Gamma(L, g(x)) \) is constructed of all code vectors \( c \), where

\[
R_c(x) \equiv 0 \pmod{g(x)},
\]

where \( c \) – given code word vector, \( R_c(x) = \sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \) and \( \frac{1}{x - \alpha_i} \) is the unique polynomial of degree \( \leq t - 1 \). So, the polynomial \( g(x) \) divides \( R_c(x) \).

- A binary Goppa code is constructed from the polynomial \( g(x) \) over \( GF(2^m) \) of degree \( t \).
- Irreducible binary Goppa codes (when Goppa polynomial is irreducible) enable us to create an efficiently error-correcting algorithm and define more precisely the lower bound of the Hamming distance. Irreducible means we choose \( g(x) \) to be an irreducible Goppa polynomial.

Figure 1 [4] shows how the original McEliece cryptosystem works:

![Figure 1. The original McEliece cryptosystem.](image)

There are many variants of the McEliece cryptosystem, where the Goppa codes are replaced by other codes which have polynomial-time bounded distance decoding such as Reed-Muller codes, Gabidulin codes or generalized Reed-Solomon codes [3]. Nevertheless, most of
these variants are insecure. Finally, we can turn to a relatively new approach based on LDPC codes that have not yet been successfully attacked.

**LDPC codes**

The most important drawback of the original McEliece cryptosystem is the key sizes, that is why we can reduce them significantly by using low-density parity-check codes without compromising the security.

**Definition 5.** Let \( C \) be an \((n,k)\) LDPC code:

\[
C = \{ c \in GF(2)^n : H \cdot c^T = 0 \},
\]

where the matrix \( H \) is called a parity-check matrix of the code \( C \) [3].

In order to obtain good performance, the parity-check matrix \( H \) should have a low density of 1 symbols, where usually the number of 1’s in every row and every column is bounded from above by a small constant. Also, it is important to avoid the short cycles in the associated Tanner graph to achieve the better performance. This graph is bipartite and is used to compose longer codes from smaller one. The codes are not structured because the positions of 1 symbols in every row of the \( H \) matrix are placed irregular, which makes encoding and decoding more difficult.

There are also structured LDPC codes. One of such families of codes are QC-LDPC codes (quasi-cyclic). Those codes are constructed such that length \( n \) and dimension \( k \) are multiple of an integer \( p \). What is also significant is that each cyclic shift of a codeword by \( n_0 \) positions produces a valid codeword. This influences the parity-check matrices and form circulant blocks. A \( p \times p \) circulant matrix \( A \) over \( GF(2) \) looks like this:

\[
\begin{bmatrix}
  a_0 & a_1 & a_2 & \cdots & a_{p-1} \\
  a_{p-1} & a_0 & a_1 & \cdots & a_{p-2} \\
  a_{p-2} & a_{p-1} & a_0 & \cdots & a_{p-3} \\
  \vdots & \ddots & \ddots & \ddots & \vdots \\
  a_1 & a_2 & a_3 & \cdots & a_0
\end{bmatrix}
\]

where \( a_i \in GF(2), i = 0 \ldots p - 1 \).

As any other type of linear block codes, QC-LDPC codes are vulnerable to the same attacks. Moreover, it is not enough to hide the secret code through a permutation, but it must be ensured that the public code does not admit sparse characteristic matrices [3]. That is why it is better to replace the permutation matrix with different transformation matrix [9].

\( Q \) is a sparse \( n \times n \) matrix, with Hamming weight \( m > 1 \) for columns and rows. That makes the LDPC matrix of the secret code \((H)\) mapped into a new parity-check matrix that is valid for the public code [9]:

---

\[
\text{5}
\]
\[ H' = H \cdot Q^T. \]

LDPC codes are used in conjunction with iterative (message-passing) decoding algorithms. One algorithm is the belief propagation algorithm. If we have a graph with message nodes and check nodes, the messages are passed from the first set nodes to the second set of nodes, and vice versa. Those messages are probabilities, or beliefs. Since the graph is sparse, the number of edges need to be traversed is small, and that is why the decoding complexity is low.

**Attacks and countermeasures**

References [3,4] propose several attacks that could potentially be efficient against the McEliece cryptosystem:

- **Attacks to the dual code**
  This attack uses the idea that the dual of the public key taken from \( H \) matrix can contain low weight codewords. Those codewords can be found by probabilistic algorithms. There are \( A_w \geq (n - k) \) codewords with low weight that is less or equal than \( d_c \), where \( d_c \) defined as the row weight of \( H \). Systems that were proposed in papers [1] and [2] based on QC-LDPC codes were tested for work factor (the effort or time needed to overcome a protective measures), and proved that LDPC based cryptosystem can be considered secure against attacks to the dual code.

- **OTD attacks**
  OTD attack is formulated by Otmani, Tillich and Dallot [8]. The main idea of this attack is that as matrices \( S \) and \( Q \) are sparse, with non-null blocks and their rows have weight \( d_c \), and \( Q \) matrix has 0 symbols everywhere except the main diagonal. This attack is more successful because adversary can obtain \( Q_iS_{i,j} \) that allows to obtain the polynomial \( g_{i,j}(x) = q_i(x)s_{i,j}(x)mod(x^p + 1) \).

  However, with dense \( S \) matrices, even if we know \( Q_iS_{i,j} \), the probability to obtain \( Q_i \) and \( S_{i,j} \) is really small. The probability that there will be at least one shift of \( q_i(x) \) is also small. It is still important to have \( Q \) with rows/columns of weight \( m \) to preserve the correcting of all intentional errors.

- **Decoding attacks**
  Those attacks are the most popular against the McEliece cryptosystem as they allow to find vector \( e \) that is used for encrypting a cyphertext. In the QC-LDPC based cryptosystem each block is “cyclically shifted version of the cyphertext \( x \) is still a valid cyphertext” [4]. Thus, the adversary can extend \( G \) by adding shifted variants of \( x \):

\[ G' = \begin{bmatrix} G \\ x \end{bmatrix} \]

Due to complexity of such decoding process, the system is still secure enough against this attack.
**Conclusion**

In this report, the main drawbacks of the original McEliece cryptosystem were briefly overviewed, namely large key and low transmission rate. Additionally, the most popular binary Goppa codes used by McEliece were discussed. It was suggested in the literature that adopting LDPC codes in such cryptosystem could provide a good level of security against several types of attacks. Nevertheless, LDPC based variants of the McEliece system can be vulnerable to new types of attacks due to the sparsity of the parity-check matrix.

To summarize, [3] and [4] show that adopted QC-LDPC McEliece cryptosystem could be considered as one of the best solutions in order to overcome weaknesses of the original McEliece cryptosystem and to resist popular attacks.
References:


