A McEliece-type cryptosystem based on Convolutional Codes

P. Almeida and D. Napp[2]

***Abstract*— In this paper we present a new variant of the McEliece cryptosystem. In contrast to the typical approach, where block codes are used, we propose the use of a convolutional encoder to be part of the public key. In this setting the message is a sequence of messages instead of a single block message and the errors are added randomly throughout the sequence. We point out several advantages of such an approach and indicate interesting lines for future research.**

***Index Terms*— McEliece cryptosystem, convolutional codes, information set decoding.**

***AMS subject classifications* — 94B10, 68P30, 11T71.**

I. INTRODUCTION

There has been a recent interest in post-quantum cryptography due to the fact that the appearance of quantum computers would break most of the public key cryptosystems (PKC) used in practice, more concretely, all cryptosystems based on factorization and discrete logarithm problems. Thus, there is currently a great interest in working on the McEliece cryptosystem as it is one of the most promising PKC able to resist attacks based on quantum computers, since it relies on the hardness of decoding a linear block code without any visible structure.

Another important advantage of the McEliece cryptosystem is its fast encryption and decryption procedures which require a significantly lower number of operations with respect to alternative solutions (like RSA). However, the original McEliece cryptosystem has two main disadvantages: low encryption rate and large key size, both due to the Goppa codes it is based on.

Motivated by this, there have been several attempts to substitute the underlying Goppa codes by other classes of block codes, using Generalized Reed-Solomon (GRS), Low-DensityParity-Check(LDPC), Quasi-Cyclic, among others. Unfortunately, these alternatives have exposed the system to security threats. A new idea was recently presented in [1] where Baldi et al. proposed to replace the permutation matrix used in the original McEliece scheme by a more general transformation. This new variant aimed at opening the possibility of trying to use again different classes of codes (*e.g.*GRS) that were unsuccessfully proposed earlier. Although new variants of this idea are currently under investigation, the proposed system in [1] was broken in [2].

Another new and interesting scheme was proposed in [3] where the secret code is a convolutional code. One of the most appealing feature of this system is the fact that its secret generator matrix contains large parts which are generated completely at random and has no algebraic structure. However, the scheme consider fixed block lengths (with the block Toeplitz structure typical of convolutional codes) and therefore had many similarities with block codes. This allowed the adaptation of existing attacks and the scheme was broken, see [4].

In this paper, we investigate further the possibility of using convolutional codes instead of block codes in order to overcome the above mentioned disadvantages. In the proposed scheme, the message is not a block vector but a stream of vectors sent in a sequential fashion. Again the security relies in the difficulty of decoding a general convolutional code (specially hard when the degree of the code is large). Our construction uses large parts of randomly generated matrices in order to mask the secret key. Moreover, the truncated sliding matrix of the encoders are not full row rank (i.e., is not a generator matrix of a block code). Therefore, the existing attacks to this type of PKC do not seem to be effective. We outline the idea of this novel scheme and provide preliminaries comments on its security.

II. PRELIMINARIES

This section contains the background needed for the development of our results. We introduce the McEliece PKC and the type of convolutional codes considered in this work.

Let $\mathbb{F} = \mathbb{F}_q$ be a finite field of size q, $\mathbb{F}((D))$ be the field of formal Laurent polynomials with coefficients in $\mathbb{F}$, $\mathbb{F}(D)$ be the field of rational polynomials with coefficients in $\mathbb{F}$ and $\mathbb{F}[D]$ be the ring of polynomials with coefficients in $\mathbb{F}$.

A. The original McEliece cryptosystem

Let $G \in \mathbb{F}^{k\times n}$ be an encoder of an $[n, k]$-block code $\mathcal{C}$ with distance d capable of correcting $t = \lfloor \frac{d-1}{2} \rfloor$ errors, $S \in \mathbb{F}^{k\times k}$ an invertible matrix and $P \in \mathbb{F}^{n\times n}$ a permutation matrix. In the classical McEliece cryptosystem G, S and P are kept secret and

$$G' = SGP \tag{1}$$

and t are public. Bob publishes G' and Alice encrypts the cleartext message $\mathbf{u} \in \mathbb{F}^k$ to produce $\mathbf{v} = \mathbf{u}G'$, chooses a random error $\mathbf{e} \in \mathbb{F}^n$ with weight $\mathrm{wt}(\mathbf{e}) \leq t$ and sends the ciphertext

$$\mathbf{y} = \mathbf{v} + \mathbf{e} = \mathbf{u}G' + \mathbf{e} = \mathbf{u}SGP + \mathbf{e}.$$

The generator matrix G is selected in such a way that allows an easy decoding so that when Bob receives the vector $\mathbf{y}$, multiplies from the right by the inverse of P and recovers

*This work was not supported by the Portuguese Foundation for Science and Technology (FCT-Fundação para a Ciência e a Tecnologia), through CIDMA - Center for Research and Development in Mathematics and Applications, within project UID/MAT/04106/2013

[1] The authors are with the Department of Mathematics, University of Aveiro, Portugal palmeida@ua.pt, diego@ua.pt

$\mathbf{u}S$ by decoding $(\mathbf{u}S)G + \mathbf{e}P^{-1}$ as $wt(\mathbf{e}P^{-1}) \leq t$. Finally, Bob multiplies on the right $\mathbf{u}S$ by the matrix S^{-1} to obtain $\mathbf{u}$.

It is important to remark that the security of this cryptosystem lies in the difficulty of decoding a random encoder, known to be an NP hard problem. Hence, G needs to admit an easy decoding algorithm but G' has to look as random as possible.

B. Convolutional Codes

Unlike linear block codes, there exist several approaches defining convolutional codes. In this subsection we introduce convolutional codes using the generator matrix approach. As opposed to block codes, convolutional codes process a continuous sequence of data instead of blocks of fixed vectors. If we introduce a variable D, usually called the *delay operator*, to indicate the time instant in which each information arrived or each codeword was transmitted, then we can represent the sequence message $(\mathbf{u}_0, \mathbf{u}_1, \cdots), \mathbf{u}_i \in \mathbb{F}^k$ as a polynomial sequence $\mathbf{u}(D) = \mathbf{u}_0 + \mathbf{u}_1 D + \cdots \in \mathbb{F}^k((D))$. In this representation the encoding process of convolutional codes, and therefore the notion of convolutional code, can be presented as follows.

A *convolutional code* $\mathcal{C}$ (see definition [5, Definition 2.3]) of rate k/n is an $\mathbb{F}((D))$-subspace of $\mathbb{F}((D))^n$ of dimension k given by a rational *encoder matrix*

$$G(D) = \sum_{i=0}^{m} G_i D^i \in \mathbb{F}(D)^{k\times n},$$

i.e.

$$\mathcal{C} = \text{ Im }_{\mathbb{F}((D))} G(D) = \left\{ \mathbf{u}(D)G(D) : \mathbf{u}(D) \in \mathbb{F}^k((D)) \right\},$$

where m is called the *memory* of $G(D)$.

III. A new variant of the McEliece PKC based on convolutional encoders

Here we propose a new scheme of the McEliece PKC where a secret encoder of a block code is masked by polynomial matrices yielding a convolutional polynomial encoder, which constitutes the public key. More precisely, the ingredients of the keys are the following:

- $G \in \mathbb{F}^{k\times n}$ an encoder of a $(n.k)$-block code with error-correcting capability t and that admits an easy decoding algorithm.
- $T(D, D^{-1}) = \sum_{i=-1}^{1} T_i D^i \in \mathbb{F}((D))^{n\times n}$ invertible (in $\mathbb{F}((D))$) such that the positions of the nonzero columns of T_i form a partition of n and each row of T_i has at most one nonzero element, for $i = -1, 0, 1$.
- $P(D, D^{-1}) := T^{-1}(D, D^{-1}) = \sum_{i=-1}^{1} P_i D^i \in \mathbb{F}((D))^{n\times n}$.
- $S(D) = \sum_{i=1}^{\nu} S_i D^i \in \mathbb{F}((D))^{k\times k}$ with S_1 invertible.
- $\mathbf{e}(D) = \sum_{i\geq 0} e_i D^i \in \mathbb{F}[D]^n$ a random error vector satisfying
$$\text{wt}((\mathbf{e}_i, \mathbf{e}_{i+1}, \mathbf{e}_{i+2})) \leq t \tag{2}$$
for all $i \geq 0$ and $\mathbf{e}_j = 0$ for $j < 0$.

Matrices $T(D, D^{-1})$ with more than three coefficients can be also used in this schema. However, in this preliminary work, and for the sake of simplicity, we restrict ourself to this simple case. Before introducing the scheme we present the following straightforward result.

Lemma 3.1: Let $T(D, D^{-1})$ and $\mathbf{e}(D)$ be as described above. Then, all the coefficients of $\mathbf{e}(D)T(D, D^{-1})$ have weight less than or equal to t.

Remark 3.1: Observe that as S_1 is invertible one can retrieve

$$\mathbf{u}(D) = \mathbf{u}_0 + \mathbf{u}_1 D + \cdots + \mathbf{u}_\ell D^\ell$$

knowing $\mathbf{u}(D)S(D) =: \sum_{i=\nu}^{\nu+\ell} \mathbf{w}_i D^i$.

The scheme works as follows:

Secret key: $\{S(D), G, P(D, D^{-1})\}$.

Public key: $\{G'(D) := S(D)GP(D, D^{-1})$, t$\}$.

Encryption: Alice selects an error vector $\mathbf{e}(D)$ and encrypts the message

$$\mathbf{u}(D) = \mathbf{u}_0 + \mathbf{u}_1 D + \mathbf{u}_2 D^2 + \cdots + \mathbf{u}_\ell D^\ell \in \mathbb{F}[D]^k$$

as

$$\mathbf{y}(D) = \mathbf{u}(D)G'(D) + \mathbf{e}(D), \tag{3}$$

to finally send the ciphertext $\mathbf{y}(D) = \sum_{i\geq 0} \mathbf{y}_i D^i$.

Decryption: Bob multiplies (3) from the right by the matrix $T(D, D^{-1})$ to obtain

$$\mathbf{u}(D)S(D)G + \mathbf{e}(D)T(D, D^{-1}). \tag{4}$$

By Lemma 3.1 each coefficient of $\mathbf{e}(D)T(D, D^{-1})$ has weight $\leq$ t and therefore each of the coefficient of $\mathbf{u}(D)S(D) = \sum_{i=\nu}^{\nu+\ell} \mathbf{w}_i D^i$ can be decoded. By Remark (3.1) Bob can recover the message $\mathbf{u}(D)$ from $\mathbf{u}(D)S(D)$.

Remark 3.2: We have

$$G'(D) = G'_0 + G'_1 D + \cdots + G'_{1+\nu} D^{1+\nu},$$

where $G'_0 = S_1 G P_{-1}$, $G'_1 = S_2 G P_{-1} + S_1 G P_0$ and so on. So it is easy to see that the term G'_0 is the most vulnerable to structural attacks. In order to protect the secret key G against this type of attacks, we choose P_{-1} with many null rows. This is achieved if we choose T as described above as we will show in the next section. Note that, in this way, we also guarantee that G'_0 is not full row rank, therefore we are also being protected against information set decoding attacks.

It is important to note that it is not difficult to show that the scheme presented can be implemented *sequentially*, *i.e.*, we can encrypt the data $\mathbf{u}_i$'s as it enters into the encoder G' and decrypt the sequence of $\mathbf{y}_i$'s as they arrive, without waiting for the end of the transmission.

IV. Comments on the security

One general observation about the security of the proposed cryptosystem is about the way the public key is generated. As opposed to previous constructions of the variants of the McEliece PKC, $G'(D) = S(D)GP(D, D^{-1})$ is constructed using large parts randomly generated, and this makes structure attacks more difficult. Effectively, the coefficients of $S(D)$ are totally randomly generated except for the S_1 that is required to be invertible. Therefore, we have

$$\left(q^{k^2}\right)^{\nu-1} \prod_{i=0}^{k-1} (q^k - q^i)$$

possible $S(D)$ matrices. The block code with generator matrix G can be any code with a fast decoding algorithm. For example a GRS, Goppa or LDPC code. Moreover, the set of admissible $P((D, D^{-1}))$ is considerably large, and therefore also the selection of $P((D, D^{-1}))$ introduces additional randomness into the system.

We construct T as $T = \Pi\Delta\,\Gamma$, where Γ is a permutation matrix, Δ is a diagonal matrix and Π is invertible matrix, satisfying certain conditions that imply that $T((D, D^{-1}))$ is invertible, the positions of the nonzero columns of T_i form a partition of n and each row of T_i has at most one nonzero element, for $i = -1, 0, 1$. More specifically, Γ is any permutation matrix of order n,

$$\Delta = \begin{bmatrix} 1/D & & & & & & & & \\ & \ddots & & & & & & & \\ & & 1/D & & & & & & \\ & & & 1 & & & & & \\ & & & & \ddots & & & & \\ & & & & & 1 & & & \\ & & & & & & D & & \\ & & & & & & & \ddots & \\ & & & & & & & & D \end{bmatrix}$$

$$\underbrace{\qquad\qquad\qquad}_{d_1}\quad\underbrace{\qquad\qquad}_{d_2}\quad\underbrace{\qquad\qquad}_{d_3}$$

where $d_1 = d_3$ and $d_1 + d_2 + d_3 = n$, and

$$\Pi = \left[\begin{array}{c|c|c} I_{d_1} & U_{1\,2} & U_{1\,3} \\ \hline U_{2\,1} & I_{d_2} & U_{2\,3} \\ \hline U_{3\,1} & U_{3\,2} & I_{d_3} \end{array}\right]$$

where each $U_{i\,j}$ has at most one nonzero entry in each row, for $i < j$. If $i > j$, the matrices $U_{i\,j}$ should be chosen so that Π is invertible and each row has at most one nonzero element. The simplest case is when $U_{2\,1}, U_{3\,1}$ and $U_{3\,2}$ are null matrices. So, there are $n!$ possible Γ matrices, $\lfloor n/2 \rfloor$ possible Δ matrices and $(q-1)(d_j+1)^{d_i}$ possible $U_{i\,j}$ matrices, when $i < j$.

Remark 4.1: Since

$$P(D, D^{-1}) = T^{-1}(D, D^{-1}) = \Gamma^{-1}\Delta^{-1}\Pi^{-1}$$

it easily follows that the positions of the nonzero rows of P_i form a partition of n. Hence if T_1 have just a few nonzero columns then P_{-1} will have many null rows, which enable us to protect the secret key G in the initial term G'_0 against structural attacks.

A. Plaintext recovery

These attacks try to decode a random linear code without requiring any knowledge of the secret key. However, trying to decode directly a convolutional code, for example with the Viterbi decoding algorithm, seems very difficult.

The plaintext recovery type of attack is typically performed using information set decoding algorithms (ISD). Information set decoding tries to solve the following NP-hard problem: Given a random looking generator matrix $G' \in \mathbb{F}^{k\times n}$ of a linear code C' and a vector $\mathbf{y} = \mathbf{u}G' + \mathbf{e}$, recover $\mathbf{u}$. Roughly speaking, the problem is that of decoding a random linear code. The first step of any ISD algorithm is to find a size-k index set $I \subset \{1, 2, \ldots, n\}$ such that the sub-matrix of G' with the columns indexed by I forms an invertible matrix of order k, or, equivalently, the sub-matrix of the parity check matrix H' (associated with G') with the columns indexed by $I^\star = \{1, 2, \ldots, n\}\backslash I$ forms an invertible matrix of order $n-k$. The set I is called an *Information Set.*

The second step depends of the algorithm we are using, but the basic idea is to guess the I-indexed part e_I of the error vector $\mathbf{e}$ according to a predefined method (that depends on each specific algorithm) and try to obtain the whole $\mathbf{e}$ from these assumptions. For example:

- In Prange algorithm (also called plain ISD algorithm) we guess that $\mathrm{wt}(e_I) = 0$.
- In Lee-Brickell algorithm we guess that $\mathrm{wt}(e_I) = p$, for a fixed value p.
- In Stern algorithm we separate I in two sets I_1 and I_2 with approximately the same size and guess that $\mathrm{wt}(e_{I_1}) = p/2$, $\mathrm{wt}(e_{I_2}) = p/2$, and so $\mathrm{wt}(e_I) = p$. Stern considers also other restritions in the $I^\star$-indexed part of $\mathbf{e}$, namely the first ℓ coordinates of $e_{I^\star}$ have weight zero.

Next, we analyse how these ideas could be adapted to our context.

The ciphertext is generated as $\mathbf{y}(D) = \mathbf{u}(D)G'(D) + \mathbf{e}(D)$ or equivalently, $\begin{bmatrix} \mathbf{y}_0 & \mathbf{y}_1 & \cdots & \mathbf{y}_l & \cdots & \mathbf{y}_{l+\nu+\mu} \end{bmatrix}$which is equal to the multiplication of

$$\begin{bmatrix} \mathbf{u}_0 & \mathbf{u}_1 & \cdots & \mathbf{u}_\ell \end{bmatrix}$$

with

$$\begin{bmatrix} G'_0 & G'_1 & \cdots & G'_{\nu+\mu} & & & & & \\ & G'_0 & G'_1 & \cdots & G'_{\nu+\mu} & & & & \\ & & \ddots & \ddots & & \ddots & & & \\ & & & G'_0 & G'_1 & \cdots & G'_{\nu+\mu} & & \\ & & & & \ddots & \ddots & & \ddots & \\ & & & & & G'_0 & G'_1 & \cdots & G'_{\nu+\mu} \end{bmatrix}$$

and adding the error vector

$$\begin{bmatrix} \mathbf{e}_0 & \mathbf{e}_1 & \cdots & \mathbf{e}_{\ell+\nu+\mu} \end{bmatrix}.$$

Hence, one may try to attack the first vectors $\mathbf{y}_i$'s, *i.e.*, decode the following interval of data

$$\begin{bmatrix} \mathbf{u}_0 & \mathbf{u}_1 & \cdots & \mathbf{u}_s \end{bmatrix} \underbrace{\begin{bmatrix} G'_0 & G'_1 & \cdots & G'_s \\ & G'_0 & \cdots & G'_{s-1} \\ & & \ddots & \vdots \\ & & & G'_0 \end{bmatrix}}_{=:G'_{truc}(s)} + \qquad (5)$$

$$+ \begin{bmatrix} \mathbf{e}_0 & \mathbf{e}_1 & \cdots & \mathbf{e}_s \end{bmatrix} = \begin{bmatrix} \mathbf{y}_0 & \mathbf{y}_1 & \cdots & \mathbf{y}_s \end{bmatrix},$$

using, for instance, the Stern algorithm or some of its more improved versions. However, note that this is not an standard decoding problem as $G'_{truc}(s)$ is still not a full row rank matrix and one should adapt somehow the ISD techniques to the context of this work.

V. CONCLUSIONS

In this work we have proposed a new variant of the McEliece cryptosystem using convolutional codes instead of block codes. This scheme is in many aspects different from the previous proposed variants as the message is not a block vector anymore but a stream sequence of vectors. Trying to adapt the existing attacks seems not straightforward. A more detail analysis of the key size and the security of this system needs to the done.

ACKNOWLEDGMENT

This work was partially supported by the Portuguese Foundation for Science and Technology (FCT-Fundação para a Ciência e a Tecnologia), through CIDMA - Center for Research and Development in Mathematics and Applications, within project UID/MAT/04106/2013

REFERENCES

[1] M. Baldi, M. Bianchi, F. Chiaraluce, J. Rosenthal, and D. Schipani, "Enhanced public key security for the mceliece cryptosystem," *Journal of Cryptology*, vol. 29, no. 1, pp. 1–27, 2016.

[2] A. Couvreur, A. Otmani, J. Tillich, and V. Gauthier-Umana, "A polynomial-time attack on the BBCRS scheme," *Conference Public Key Cryptography (PKC), 2015 https://arxiv.org/pdf/1501.03736.pdf*, 2015.

[3] C. Löndahl and T. Johansson, "A new version of mceliece pkc based on convolutional codes," in *Information and Communications Security*, T. W. Chim and T. H. Yuen, Eds. Springer Berlin Heidelberg, 2012, pp. 461–470.

[4] G. Landais and J.-P. Tillich, "An efficient attack of a mceliece cryptosystem variant based on convolutional codes," in *Post-Quantum Cryptography*, P. Gaborit, Ed. Springer Berlin Heidelberg, 2013, pp. 102–117.

[5] R. McEliece, "The algebraic theory of convolutional codes," in *Handbook of Coding Theory Vol. 1*, R. B. V.S. Pless, W.C. Huffman, Ed. North-Holland, Amsterdam, 1998.