Annals of the University of Craiova, Mathematics and Computer Science Series
Volume 45(1), 2018, Pages 137–150
ISSN: 1223-6934

LoRi-TTM cryptosystem

Miroslav Kureš, Thibault Decome, and Ghislain Drecourt

Abstract. The new public key cryptosystem is proposed: the tame transformation method TTM is in it innovatively considered over local rings. The paper includes an unsolved example as a challenge for experts in cryptoanalytics.

2010 Mathematics Subject Classification. 60J10, 15B51, 60E05, 82B20.
Key words and phrases. public key cryptography, multivariate cryptography, local ring, TTM scheme.

1. Introduction

In this paper, a ring R will be always commutative and containing an identity $1_R \neq 0_R$, $\mathrm{Z}(R)$ denotes the set of all zero divisors of R (including 0_R) and $\mathrm{U}(R)$ denotes the set of all units of R, which is a group with respect to the ring multiplication. Further, $M_{m\times n}(R)$ denotes the set of all $m \times n$ matrices with entries in R.

The purpose of this paper is to introduce a new public key cryptosystem. The presented cryptosystem belongs to so-called multivariate public key cryptography systems: it is well known that multivariate quadratic polynomial maps over a finite field are widely studied, mainly for their applications in cryptography. The theory comes under commutative algebra, polynomial ideals and algebraic geometry. The development of methods of multivariate public key cryptography (MPKC) comes in 20th century; in contrast, the security of RSA-type cryptosystems relies on the complexity of integer factorization and is based on results in number theory developed in the 17th and 18th centuries. Elliptic curve cryptosystems employ the use of mathematics from the 19th century, [1].

Our main contribution is the use of multivariate quadratic polynomial maps which are not over a finite field (and, of course, not over a finite domain which is automatically a finite field thanks to Wedderburn's little theorem) but over a ring; in particular, we use polynomials over local rings and apply the tame transformation method TTM ([4]) proposed by T. T. Moh. We remark that in Section 2 of Moh's paper [4] from 1999, there is a note that history of TTM starts in the middle 90's where the first project of this cryptosystem was based on the composition $\pi = \lambda_2 \circ \tau \circ \lambda_1$ and was attacked by Montgomery and Sathay already in 1996. Nevertheless, we use the scheme $\pi = \lambda_2 \circ \tau \circ \lambda_1$, too: it is the simplest TTM-scheme and the question whether the attack will be successfull also for unerlying locar ring seems to be interesting for

The first author was supported by Brno University of Technology, the specific research plan No. FSI-S-17-4464.

us. We will call it the Tame Transformation Method over Local Rings or shortly the *LoRi-TTM cryptosystem*.

We remark that there are only very rare examples of the use of local rings in cryptography, we mention e. g. Zu-hua's paper [6] where special matrices over the local ring $\mathbb{Z}/\langle 2^m\rangle$, $m \geq 64$ are used.

2. Polynomials over local rings and their automorphisms

Let F be an arbitrary field and $F[u_1, \ldots, u_\lambda]$ the ring of polynomials in λ indeterminates $u_1, \ldots, u_\lambda$ over F. The maximal ideal of $F[u_1, \ldots, u_\lambda]$ is $\langle u_1, \ldots, u_\lambda\rangle$, i.e. the ideal generated by $u_1, \ldots, u_\lambda$.

Let us consider a factor ring of a form

$$R = F[u_1, \ldots, u_\lambda]/\mathfrak{i} + \mathfrak{m}^{r+1},$$

where the ideal $\mathfrak{i}$ satisfies $\mathfrak{m}^{r+1} \subsetneqq \mathfrak{i} \subseteq \mathfrak{m}^2$ and is generated by a finite number of polynomials.

Then R is an F-algebra thanks to the ring homomorphism mapping elements $a \in F$ to equivalence classes $a + \mathfrak{i} + \mathfrak{m}^{r+1}$. Thus, R is a vector space over F. Moreover, equivalence classes $u_1 + \mathfrak{i} + \mathfrak{m}^{r+1}$, $\ldots$, $u_\lambda + \mathfrak{i} + \mathfrak{m}^{r+1}$ form the nilradical (nilpotent ideal) $\mathfrak{n}_R$ having a finite dimension (as a vector space) and $R/\mathfrak{n}_R = F$. (We will write here also shortly $u_1, \ldots, u_\lambda$ instead $u_1 + \mathfrak{i} + \mathfrak{m}^{r+1}$, $\ldots$, $u_\lambda + \mathfrak{i} + \mathfrak{m}^{r+1}$.) Of course, elements of $\mathfrak{n}_R$ are zero divisors of R.

The nilradical $\mathfrak{n}_R$ is a unique maximal ideal of R (hence the Jacobson radical of R). Now, we call the *order* of R the minimum $\mathrm{ord}(R)$ of the integers r satisfying $\mathfrak{n}_R^{r+1} = 0$ and the *width* $\mathrm{w}(R)$ of R the dimension $\dim_F(\mathfrak{n}_R/\mathfrak{n}_R^2)$. Thanks to our special form if the ideal, we have $\mathrm{ord}(R) = r$, $\mathrm{w}(R) = \lambda$. Units $\mathrm{U}(R)$ of R are non-zero elements of F.

We will call R shortly the *local ring*. The purpose of this paper is to introduce a multivariate public key cryptosystem over R. For a clear understanding, we first present an example of R.

Example 1. We take $F = \mathbb{F}_{19}$, $\lambda = 2$, $r = 2$, $\mathfrak{i} = \langle uv\rangle$ (where we write u, v instead u_1, u_2). We have

$$R = \mathbb{F}_{19}[u, v]/\langle uv\rangle + \mathfrak{m}^3,$$

$\mathrm{ord}(R) = 2$, $\mathrm{w}(R) = 2$ and elements of R can be expressed by

$$A + Bu + Cv + Du^2 + Ev^2 \quad (A, B, C, D, E \in \mathbb{F}_{19})$$

where the multiplication respect that uv and all monomials of the order 3 vanish.

So, elements of the nilradical $\mathfrak{n}_R$ have the form

$$Bu + Cv + Du^2 + Ev^2 \quad (B, C, D, E \in \mathbb{F}_{19}).$$

We will consider polynomials over local rings in question and denote the polynomials in n intereminates by

$$R[X_0, \ldots, X_{n-1}].$$

Tame automorphisms of $R[X_0, \ldots, X_{n-1}]$ are generated by *affine automorphisms*

$$\begin{aligned} \lambda(X_0) &= a_{00}X_0 + \cdots + a_{0n-1}X_n - 1 + b_0 \\ &\cdots \\ \lambda(X_{n-1}) &= a_{n-10}X_0 + \cdots + a_{n-1n-1}X_n - 1 + b_{n-1} \end{aligned}$$

(with non-singular matrix of the linear part) and by *triangular (de Jonquières) automorphisms*

$$\begin{aligned} \tau(X_0) &= u_0X_0 + P_0(X_1, \ldots X_{n-1}) \\ &\cdots \\ \tau(X_{n-2}) &= u_{n-2}X_{n-2} + P_{n-2}(X_{n-1}) \\ \tau(X_{n-1}) &= u_{n-1}X_{n-1} + P_{n-1} \end{aligned}$$

(where u_j are units in R and P_j polynomials, $j = 0, \ldots, n-1$).

3. TTM scheme

As we noted above, we use the tame polynomial automorphism $\pi = \lambda_2 \circ \tau \circ \lambda_1$, where λ_1 and λ_2 are affine automorphisms and τ a triangular automorphism. We consider only quadratic P_j in τ. It is easy to find inversions for λ_1, λ_2 and τ, but, in general, computationally difficult for π if we do not know its decomposition to λ_1, λ_2 and τ.

For a creation of linear automorphisms (linear part of affine automorphisms), one can apply elementary transvections and elementary dilations to the identity matrix.

By an *elementary transvection* $V_{ij}(r)$ (of order n), $1 \le i \ne j \le n$, $r \in R$, we mean a matrix $[a_{\nu\mu}]$ from $M_{n\times n}(R)$ with

$$a_{\nu\mu} = \begin{cases} 1_R & \text{for } \nu = \mu \\ r & \text{for } \nu = i, \mu = j \\ 0_R & \text{otherwise} \end{cases}$$

and by an *elementary dilation* $S_i(r)$ (of order n), $1 \le i \le n$, $r \in \mathrm{U}(R)$, we mean a matrix $[a_{\nu\mu}]$ from $M_{n\times n}(R)$ with

$$a_{\nu\mu} = \begin{cases} 1_R & \text{for } \nu = \mu \ne i \\ r & \text{for } \nu = \mu = i \\ 0_R & \text{otherwise.} \end{cases}$$

Finite products of elementary transvections and elementary dilations (of order m) are called (m-th order) *elementary matrices*. Evidently, elementary matrices (elementary transvections, elementary dilations) are invertible and their inversions are elementary matrices (elementary transvections, elementary dilations, respectively).

4. The polymorphic alphabet

In subsequent sections, we will explain the principles of our system on a particular example. For its programming and practical verification was used a free open-source mathematics software system Sage. However, we are sure that the general characteristics of our system remain well understood.

In our LoRi-TTM cryptosystem, we use two alphabets. The first corresponds to the characters allowed by the cryptosystem for the initial message. The second one corresponds to the characters used by the cryptosystem to cipher the message.

4.1. Alphabet of the initial message. Any initial message must be written in the following alphabet of 64 characters:

a b c d e f g h i j k l m n o p q r s t u v w x y z 0 1 2 3 4 5 6 7 8 9
, . ; : ! ? @ [ ] ( ) ^ { } = + * - ' A B C D E F G H
there is also the blank character in this alphabet.

4.2. Alphabet of the ciphering message. This alphabet is based on the previous alphabet but it needed to have a length of 256 letters for a question of bijection in the ciphering process.
As a result, we decide to use 256 random letters of 2 characters taken in the first alphabet.

Example of letters:
a] {u 5k @r

5. On a choice of LoRi-TTM parameters

5.1. Constant part of the cryptosystem. At the same time we implement our cryptosystem, we fix some parameters that be a base for the working of it. In this way, the user is not able to change them.
We fix the following parameters:
(1) The underlying ring R.
(2) The key generator.
(3) The transformation of the message into tuple of ring elements.

5.1.1. *The underlying ring.* Following the previous chapters, we impose the form of our ring of work:

$$R = \mathbb{F}_p[u,v]/\langle uv\rangle + \mathfrak{m}^3 \text{ with } \mathfrak{m}^3 = \langle u^3, uv^2, vu^2, v^3\rangle.$$

All our implementation is based on this ring. Even if Sage compute automatically some modifications due to our ring, we implement functions that create elements for this ring only. If we change the ring of work, we have to change our code.

For example, if we take the function creating a random matrix of elements of our ring, function used in the creation of automorphisms λ and τ, we have:

```
sage: def mat(U, l, c):
for i in range(0, l):
 t[i] = random_matrix(R, c, 5)
r = [k for k in range(0, l)]
for j in range(0, l):
 r[j] = t[j] * U
T = matrix([r[f] for f in range(0, l)])
```

return T

In this function we use the vector $U = (1, u, v, u^2, v^2)$ corresponding to the element of our ring. If the user changes the ring, we would not have the matrix's elements that are ring elements.

5.1.2. *The π generator.* Each automorphism λ and the automorphism τ are created from random matrices of element of our ring. As a consequence, the function π, automatically generated, change every time the user restarts the process. Then, the user can have a new pair of public and private keys by this way. He cannot fix these algorithms because there are automatically generated.

5.1.3. *The length of the alphabet.* In our cyptosystem, we fix the length of the first alphabet. In fact, the length of this alphabet has to be equal to 64. Our ciphering method deals only with this case. When we want to transform a ring element into a part of the message, we consider that the first alphabet has a length of 64 words. Then, we can not change this length without change our code.

5.2. The modifiable parts. There is some parts in our cryptosystem that the user can define as he wants. The modification of these parameters change the format of the ciphered message and the functioning of our cryptosystem.

5.2.1. *The p and the n.* The first parameters that the user has to fix are the prime number p and the integer n which represents the number of indeterminates in

$R[X_0, \ldots, X_{n-1}]$. When the user fixes the used n, he fixes the number of undeterminates used by the cryptosystem. In fact, we change then the number of ring element in the used tuples.
As a consequence, this change modifies also the length of the inital blocks.
Examples:

- For $p = 5$ and $n = 3$, we have to truncate the message in blocks of length 3.
- For $p = 5$ and $n = 5$, we have to truncate the message in blocks of length 5.

When the user fixes the used p, he fixes differents criterias of the cryptosystem. First, the coefficients of each ring element are taken between 0 and $p - 1$. This point has for consequences the limitation of the length in bits of each fundamental block.
Examples:

- For $p = 5$, the maximal length is $\underline{12}$.
- For $p = 400009$, the maximal length is $\underline{94}$.

Another limitation is the length in bits of these fundamental blocks. In fact, when we change p, we change the maximal number of elements supported by our ring and then, the maximal length has to be lower than this number.
Example:

- For $p = 5$, the maximal length is 1.
- For $p = 400009$, the maximal length is 11.

5.2.2. *Modification of the used alphabet.* One of the parameters of our cryptosystem is the used alphabet. It defines the format of the original message.
We saw in a previous section that we can not change the length of the alphabet but we can change the words in this alphabet. As long as its length does not change, the

user can define the words he wants. The second alphabet would change automatically in this case.

5.2.3. *Modification of the affine automorphism constructor.* In order to create affine automorphisms, we create for its linear part random elementary transvections and dilations; we iterate the multiplications of these matrices. In our cryptosystem, we implicitly set this number of iteration as 50 but the user can change this number.
In fact, when you change this number you randomize more or less your affine automorphism.

6. Possible attacks

The problem of developing new public key cryptosystem had occupied the cryptographic research fields for the last decades and several recent public key systems use multivariate polynomial systems of equations, particularly quadratic polynomials, instead of number-theoretic constructions (we recall the Matsumoto-Imai method, the hidden graph method or TTM). In the development of multivariate public key cryptosystems, algebraic attack is an important area of research, which comes from the linearization equation attack by Patarin [5]. This attack method refers to any technique that ends with a solving system. A linearization equation is an equation of such form $\sum_{i,j=1}^{n} a_{ij}x_iy_j + \sum_{i=1}^{n} b_ix_i + \sum_{j=1}^{n} c_jy_j + d = 0$, where x_i are plaintext variables and y_j are ciphertext variables. Another generalization of linearization equation has a form $\sum_{i,j,k=1}^{n} a_{ijk}x_iy_jy_k + \sum_{i,j=1}^{n} b_{ij}x_iy_j + \sum_{i=1}^{n} c_ix_i + \sum_{i,j=1}^{n} d_{jk}y_jy_k + \sum_{j=1}^{n} e_jy_j + f = 0$. Such equations are sometimes called *high order linearization equations (HOLE)*; the total degree of the highest order of the ciphertext variables is called the *order* of the HOLE; thus, the first equation can be called the *first order linearization equation (FOLE)* and the second one the *second order linearization equation (SOLE)*.

7. Example: ciphered message

7.1. Cryptosystem principles. Our cryptosystem is based on a quotient ring R given in a following section.
At the beginning we have a message of any length written in a particular alphabet. We apply the following steps:

(1) **Transformation of the message in a tuple of ring element:** we transform the message in a tuple of n elements of R throught binary strings.
(2) **Using of a TTM automorphism π:** we apply the automorphism π on the tuple in order to obtain a new tuple of n elements of our ring.
(3) **Transformation of the new tuple in the ciphered message:** during this step, we transform the tuple throught binary strings. However, the ciphered message is written over a random alphabet of 2-length letters.

7.2. Used ring. The implemented cryptosystem is based on the following ring:

$A = R[X_0, \cdots, X_{n-1}]$, $R = \mathbb{F}_p[u,v]/\langle uv \rangle + \mathfrak{m}^3$ a quotient ring with n a natural integer, p a prime number and $\mathfrak{m}^3 = \langle u^3, uv^2, vu^2, v^3 \rangle$.

7.3. A resolved example: $p = 19$, $n = 3$. For a better understanding from the reader, we offer the process of the ciphering of a message by our cryptosystem and its deciphering.

7.3.1. *Ciphering of a message.* We want to cipher the following message with our cryptosystem:

"this is the message we want to cipher"

In this section, we explain the different steps in the ciphering process that conduct us to the ciphered message.
About the length of the message. We have to divide our message in blocks of the same length that is a multiple of n. In our example, we have to divide our message in 7 blocks of six caracters. However, the length of our message is 37 characters. We need to add 5 more blank characters at the end.
We obtain the following list:

tronc = ['this i', 's the ', 'messag', 'e we w', 'ant to', ' ciphe', 'r ']

Transformation of the message. Our cryptosystem use a special TTM automorphism that need an input of a tuple of n elements of our ring ($n = 3$ in this example). Therefore, we need to transform our message into this tuple as the first step.

We consider the first element of our list *tronc*. We divide it in 3 blocks of 2 characters and we obtain the list:

tronc1 = ['th', 'is', ' i']

Now we transform each element of *tronc1* in a binary string regarding the rank of each character in our first alphabet. Then we have:

['0001001100000111', '0000100000010010', '0010010000001000']

In the following step we transform these binary strings into their related integer:

[4871, 2066, 9224]

We tranform these integers into vectors of 5 elements corresponding to their representation in the base p:

[(0, 0, 13, 9, 7), (0, 0, 5, 13, 14), (0, 1, 6, 10, 9)]

Finally we obtain our tuple of elements from the vectors: for a tuple (a, b, c, d, e), the element is $du^2 + ev^2 + bu + cv + a$

$$tup=[9u^2 + 7v^2 - 6v, -6u^2 - 5v^2 + 5v, -9u^2 + 9v^2 + u + 6v]$$

Using of the π function. One of the characteristics of this cryptosystem is the using of a TTM automorphism π during its process. We present in a next subsection what is our π function in this example and how we obtain it.
We apply this function over our tuple of elements *tup* and we have at the output:

$$[-6u^2 - 2v^2 - 8u - 8v - 5, -3u^2 - v^2 - 2u + 6v - 2, 5u^2 - v^2 + 4u - 3v - 6]$$

Transformation of the tuple of ring's elements. The aim of this code is getting a ciphered message at the output. We have to transform our new tuple of elements into a new message.
The first step is to get back the coefficients of each element: for an element $au^2 + bv^2 + cu + dv + e$, the related tuple is (a, b, c, d, e)

[[13, 17, 11, 11, 14], [16, 18, 17, 6, 17], [5, 18, 4, 16, 13]]

Then we transform these coefficients into their related integers based on this relation: for a tuple (a,b,c,d,e), the related integer is $ep^4 + cp^3 + dp^2 + ap + b$:

[1904178, 2334548, 1727498]

We convert these numbers into their binary strings. Then we have:

['11101000011100011 0010', '10001110011111010 10100', '11010010111000000 1010']

Finally we transform these binary strings in their related letters in our second alphabet:

[']4y!yC', 'vCso;3', '4B16t:']

By combining each element, we can write that the ciphered message of *'this i'* is

]4y!yCvCso;34B16t:

If we iterate the same process for each element of our list *tronc*, the final chiphered message is:

]4y!yCvCso;34B16t:8ky!zE_gr=[B4Bo)0 w8k_g6x_g,sf[4BC@G?]4uAB4vC8-C4B
(2nl8kvCi[_go)C@4Bm]'w8k?kv?_g+^0B4B8k{+8kB4m;vC8-4B;]ywm;
where _ represents the blank character

Creation of the π automorphism. We saw the need of a π automorphism in our cryptosystem. Here are the differents functions used to created our TTM automorphism. The two affine automorphisms are:

$$\lambda_1 = \begin{cases} X_0 \to (u^2 + 6v^2 + 5u + 7v - 4)X_0 + (-8u^2 + 7v^2 - 9u - 6v - 4)X_1 \\ +(u^2 + 4v^2 - u - v + 4)X_2 + 4u^2 + v^2 - 3u + 2v - 3 \\ \\ X_1 \to (-2u^2 + 5v^2 + 8u - 3v - 3)X_0 + (5u^2 - 5v^2 + 8u - v + 7)X_1 \\ +(2u^2 - 4v^2 + 3u + 7v + 8)X_2 - 2u^2 - 6v^2 - v - 7 \\ \\ X_2 \to (-8u^2 - 7v^2 + 6u - 8v - 8)X_0 + (6u^2 - 5v^2 - 5u + 2v + 8)X_1 \\ +(8u^2 + 4v^2 - u - 2v + 3)X_2 + 9u^2 - 2v^2 - u + 8 \end{cases}$$

$$\lambda_2 = \begin{cases} X_0 \to (2u^2 - 3v^2 + 3u - 9v - 4)X_0 + (4u^2 - v^2 - 8u + 8v + 4)X_1 \\ +(-5u^2 - v^2 + 2u + v + 7)X_2 - 3u^2 - 9v - 7 \\ \\ X_1 \to (-8u^2 - 4v^2 - u - v - 5)X_0 + (3u^2 - 5v^2 + 9u - 2v)X_1 \\ +(6u^2 + 3v^2 + 6u - 2v)X_2 - 3v^2 + 3u - 7v - 5 \\ \\ X_2 \to (-9u^2 - 8v^2 + u - 9v - 1)X_0 + (-6v^2 - 9u + 5v + 5)X_1 \\ +(-5u^2 + 4v^2 + 9u + 2v + 5)X_2 - 6u^2 - 5v^2 + 9u - v \end{cases}$$

The triangular automorphism is:

$$\tau = \begin{cases} X_0 \to (-5u^2+2v^2-5u+8v)X_1^2+(-7u^2+v^2+4u-4v+2)X_2^2-X_0 \\ +(-u^2-7v^2+4u-9v-4)X_1+(-u^2+2v^2+2u-3v+2)X_2 \\ +v^2-4u+1 \\ \\ X_1 \to (-2u^2+8v^2-5u-9v+5)X_2^2+8X_1+(-7u^2+8v^2-u-4v+5)X_2 \\ -9u^2-v^2+2u+v-8 \\ \\ X_2 \to 7X_2+3u^2-4u-3v+8 \end{cases}$$

With the composition of these three automorphisms, we obtain our π function:

$$\pi = \begin{cases} X_0 \to (-9u^2+8v^2-7u+5v-9)X_0^2+(-8u^2+4v^2+u+8v-5)X_0X_1 \\ +(7u^2-3u-6v+3)X_1^2+(2u^2+9v^2+2u-8v-5)X_0X_2 \\ +(4u^2+8u-8v+6)X_1X_2+(7u^2-5v^2-8u-2v+3)X_2^2 \\ +(6u^2-v^2-9u-9v+7)X_0+(-6u^2+2v^2-u-v-3)X_1 \\ +(u^2+7v^2-4u+7v+9)X_2-5u^2+v^2+2u-5v-5 \\ \\ X_1 \to (2u^2+6v^2+3u-7v-9)X_0^2+(-3u^2-3v^2-5)X_0X_1 \\ +(6u^2+v^2+9u-5v+3)X_1^2+(-6u^2+3v^2+u+3v-5)X_0X_2 \\ +(u^2-3v^2-6u-6v+6)X_1X_2+(5u^2-9v^2+4u-v+3)X_2^2 \\ +(u^2-5v^2+8u-3v)X_0+(5u^2+2v^2+4u-3v)X_1 \\ +(-5v^2+6u-4v+9)X_2-7u^2+7v^2+8u+9v-2 \\ \\ X_2 \to (-8u^2-7v^2-7u+2v+5)X_0^2+(-7u^2-3v^2+2u-2v+7)X_0X_1 \\ +(9u^2-9v^2+9u-7v-8)X_1^2+(-u^2+5v^2-7u+9v+7)X_0X_2 \\ +(5u^2+2v^2+6u+7v+3)X_1X_2+(u^2-v^2-3u-5v-8)X_2^2 \\ +(6u^2+9v^2-u+3v-6)X_0+(3u^2-v^2-7u-4v-9)X_1 \\ +(-6u^2+4v^2+7u-4)X_2+8u^2+9v^2+8u-8v-6 \end{cases}$$

7.3.2. *Deciphering of a message.* In this section, we explain the different steps of our cryptosystem to decipher a message in order to get back the initial message. The process is likely the same as previously but we want to provide to the reader a complete solved example for seeing the similar steps in each process.
The first step is to divide our ciphered message in blocks of 18 characters in this example. This division gives us the list:

tronc2 = [']4y!yCvCso;34B16t:', '8ky!zE gr=[B4Bo)0w',
'8k g6x g,sf[4BC@G?', ']4uAB4vC8-C4B(2nl',
'8kvCi[go)C@4Bm]'w', '8k?kv? g+^0B4B8k{+', '8kB4m;vC8-4B;]ywm;']

We focus our explaination on the first element of this list but the process is the same for the others.
Transformation of the message. The first step is to transform our ciphered message in a tuple of element of our ring. We divide our element in 3 blocks of 6 characters corresponding to 3 letters of our second alphabet. Then we get the following list:

[']4y!yC', 'vCso;3', '4B16t:']

The next step is to transform each element of this list in a binary string, regarding the rank of each letter of 2 characters in our second alphabet. We obtain the same binary strings than in the precedent subsection:

['11101000011100011 0010', '1000111001111101010100', '1101001011100000010 10']

We transform each binary string in its related integer:

[1904178, 2334548, 1727498]

In the following step we convert these integers into vectors of 5 elements corresponding to their representation in the base p:

[(14, 11, 11, 13, 17), (17, 17, 6, 16, 18), (13, 4, 16, 5, 18)]

Finally, we obtained our tuple of elements of our ring:

$$\text{tup2} = [-6u^2 - 2v^2 - 8u - 8v - 5, -3u^2 - v^2 - 2u + 6v - 2, 5u^2 - v^2 + 4u - 3v - 6]$$

Using of the inverse function of π. We use a TTM automorphism when we cipher the initial message. In order to decipher it, we have to use the inverse function of π. After apply this fuction to our *tup2*, we obtain:

$$[9u^2 + 7v^2 - 6v, -6u^2 - 5v^2 + 5v, -9u^2 + 9v^2 + u + 6v]$$

Transformation of the tuple of ring's elements. In this step, we want to convert our new tuple into the original message.
We start by getting back the coefficient of each element:

[[9, 7, 0, 13, 0], [13, 14, 0, 5, 0], [10, 9, 1, 6, 0]]

We convert these coefficients in their related integers based on the same relation written in the previous subsection:

[4871, 2066, 9224]

The next step is the conversion of each number into its binary representation:

['1001100000111', '100000010010', '10010000001000']

The final step is the conversion of these binary strings into their related letters in our first alphabet:

['th', 'is', ' i']

By combining each element we obtain:

this i

We iterate the same process for each element of *tronc2* and we get the original message:

this is the message we want to cipher

Creation of the inverse function of π. This deciphering use the inverse function of π. To create this function, we need to compose λ_1^{-1}, λ_2^{-1} and τ^{-1} in the following order:

$$\pi^{-1} \to \lambda_2^{-1} \circ \tau^{-1} \circ \lambda_1^{-1}$$

In this example, we use the following functions:

$$\lambda_1^{-1} = \begin{cases} X_0 \to (2u^2+5v^2+2u+9v+2)X_0 + (-7u^2-v^2+2u-4v+9)X_1 \\ +(-6u^2+3v^2-2u+7v+5)X_2 - 7u^2+5v^2+9u+5v-9 \\ \\ X_1 \to (2u^2+9v^2+5u+5v+3)X_0 + (-u^2+3v^2+9u-6v-8)X_1 \\ +(6u^2-v^2+u-8v-8)X_2 - 8u^2-8v^2-5u+4v-2 \\ \\ X_2 \to (-u^2-v^2+9u+8v-9)X_0 + (u^2+2v^2+u-9v+1)X_1 \\ +(-4u^2+3v^2-7u+6v-3)X_2 + 7u^2+9v^2+3u+8v+4 \end{cases}$$

$$\lambda_2^{-1} = \begin{cases} X_0 \to (6u^2+4v^2-4u)X_0 + (-9u^2+7v^2+8u-8v-4)X_1 \\ +(-6u^2-5v^2-u+6v)X_2 + u^2+v^2+5u+8v-1 \\ \\ X_1 \to (5u^2+6u-2v+6)X_0 + (-u^2-4v^2-7u-8v+6)X_1 \\ +(-8u^2+8v^2+7u-v+3)X_2 + 5u^2-5v^2+7v-4 \\ \\ X_2 \to (u^2+v^2-8u+6v-6)X_0 + (7u^2+9v^2-u+2v-3)X_1 \\ +(2u^2+4u-5v+1)X_2 - 3u^2-3v^2-4u-3v \end{cases}$$

$$\tau^{-1} = \begin{cases} X_0 \to (u^2+3v^2-u-6v)X_2^4 + (-9v^2-7u-4v)X_1X_2^2 \\ +(8u^2-v^2-u-6v)X_2^3 + (2u^2+3v^2+2u-7v)X_1^2 \\ +(-4u^2-9v^2+6u-2v)X_1X_2 + (4u^2-9v^2-8u+3)X_2^2 \\ -X_0 + (-9v^2+5u-5v+9)X_1 + (7u^2-v^2-6u+5v-8)X_2 \\ +6u^2+v^2+4u-3v+2 \\ \\ X_1 \to (-3u^2-7v^2+2u-4v-2)X_2^2 + (-7)X_1 \\ +(-7u^2+v^2+8u-9v-1)X_2 + u^2+4v^2-6u+8v+4 \\ \\ X_2 \to (-8)X_2 + 5u^2+6u-5v+7 \end{cases}$$

With the composition of this three automorphisms, we obtain the function π^{-1}:

$$\pi^{-1} = \begin{cases} X_0 \to 5u^2X_0^4 + 2u^2X_0^3X_1 + 6u^2X_0^2X_1^2 + 8u^2X_0X_1^3 + 4u^2X_1^4 + (-6u^2)X_0^3X_2 \\ +2u^2X_0^2X_1X_2 + 4u^2X_0X_1^2X_2 + 9u^2X_1^3X_2 + (-3u^2)X_0^2X_2^2 + 7u^2X_0X_1X_2^2 + 7u^2X_1^2X_2^2 \\ +(-7u^2)X_0X_2^3 + 5u^2X_1X_2^3 + u^2X_2^4 + u^2X_0^3 + 3u^2X_0^2X_1 + (-4u^2)X_1^3 \\ +(-5u^2)X_0^2X_2 + (-3u^2)X_0X_1X_2 + (-5u^2)X_1^2X_2 + 9u^2X_0X_2^2 + 5u^2X_1X_2^2 + 5u^2X_2^3 \\ +(-3u^2+5v^2+6u-4v+2)X_0^2 + (-6u^2+4v^2+2u-7v+8)X_0X_1 \\ +(3u^2-9v^2-u+2v+8)X_1^2 + (4u^2+6v^2+5v-5)X_0X_2 \\ +(-8u^2-3v^2-u-6v+9)X_1X_2 + (8u^2+8v^2-7u-4)X_2^2 \\ +(-u^2-5v^2+3u-3v+4)X_0 + (8u^2+v^2-3u+7v-4)X_1 \\ +(3u^2+7v^2+2u+2v+9)X_2 - 4u^2+4v^2+5u-5 \\ \\ X_1 \to (-8u^2-2v^2+2u-7v)X_0^4 + (6u^2-3v^2-3u+v)X_0^3X_1 \\ +(-8v^2-9u+3v)X_0^2X_1^2 + (-5u^2-3v^2+7u+4v)X_0X_1^3 \\ +(-5u^2-4v^2-6u+2v)X_1^4 + (-6u^2+7v^2+9u-3v)X_0^3X_2 \\ +(-4u^2-2v^2-3u+v)X_0^2X_1X_2 + (-u^2+3v^2-6u+2v)X_0X_1^2X_2 \\ +(4u^2-6v^2-4u-5v)X_1^3X_2 + (-7u^2+2v^2-5u+8v)X_0^2X_2^2 \\ +(8u^2+6v^2-u-6v)X_0X_1X_2^2 + (6u^2+4v^2-u-6v)X_1^2X_2^2 \\ +(6u^2-2v^2+u+6v)X_0X_2^3 + (-3u^2+2u-7v)X_1X_2^3 \\ +(5u^2-6v^2+8u-9v)X_2^4 + (4u^2-2v^2+8u-9v)X_0^3 \\ +(5u^2+v^2+5u-8v)X_0^2X_1 + (8u^2+7v^2)X_0X_1^2 \\ +(9u^2-6v^2+6u-2v)X_1^3 + (-6u^2-6v^2-2u+7v)X_0^2X_2 \\ +(u^2+5v^2-5u+8v)X_0X_1X_2 + (8v^2-2u+7v)X_1^2X_2 \\ +(2u^2+v^2-4u-5v)X_0X_2^2 + (6u^2+6v^2+2u-7v)X_1X_2^2 \end{cases}$$

$$\pi^{-1} = \begin{cases} +(5u^2-4v^2+2u-7v)X_2^3+(8u^2-v^2+7u+6v-8)X_0^2 \\ +(-5u^2+3v^2-8u+4v+6)X_0X_1+(8u^2-4v^2+9u-2v+6)X_1^2 \\ +(-7u^2-4v^2+7u-v+1)X_0X_2+(9u^2+9v^2+5u+3v+2)X_1X_2 \\ +(9u^2-2v^2-6u-2v-3)X_2^2+(3u^2-2u+9)X_0 \\ +(-3u^2-9v^2+9u+9v-9)X_1+(2u^2+8v^2+6u+8v+2)X_2 \\ +5u^2+5v^2-u-7v \\ \\ X_2 \to (u^2-9v^2-2u+7v)X_0^4+(-5u^2-9v^2+3u-v)X_0^3X_1 \\ +(3u^2-9v^2+9u-3v)X_0^2X_1^2+(9u^2-7v^2-7u-4v)X_0X_1^3 \\ +(7u^2-v^2+6u-2v)X_1^4+(3u^2-9v^2-9u+3v)X_0^3X_2 \\ +(5u^2+9v^2+3u-v)X_0^2X_1X_2+(3u^2-8v^2+6u-2v)X_0X_1^2X_2 \\ +(-9u^2+9v^2+4u+5v)X_1^3X_2+(-4u^2-3v^2+5u-8v)X_0^2X_2^2 \\ +(5u^2+9v^2+u+6v)X_0X_1X_2^2+(7u^2-8v^2+u+6v)X_1^2X_2^2 \\ +(6v^2-u-6v)X_0X_2^3+(-4u^2+8v^2-2u+7v)X_1X_2^3 \\ +(5u^2-8u+9v)X_2^4+(6u^2-4v^2-8u+9v)X_0^3 \\ +(6u^2-5u+8v)X_0^2X_1+(-8u^2-7v^2)X_0X_1^2 \\ +(8u^2-8v^2-6u+2v)X_1^3+(-6u^2-2v^2+2u-7v)X_0^2X_2 \\ +(7u^2-6v^2+5u-8v)X_0X_1X_2+(7u^2+3v^2+2u-7v)X_1^2X_2 \\ +(-7u^2+2v^2+4u+5v)X_0X_2^2+(6u^2+2v^2-2u+7v)X_1X_2^2 \\ +(7u^2-7v^2-2u+7v)X_2^3+(-8u^2+9v^2+6v-3)X_0^2 \\ +(9u^2-6v^2+5u+4v+7)X_0X_1+(9u^2-5u+8v+7)X_1^2 \\ +(-7u^2+3v^2+7u+8v-2)X_0X_2+(7u^2-5v^2+4v-4)X_1X_2 \\ +(6u^2+v^2-7u+3v+6)X_2^2+(-6u^2+6v^2-3u-4v-9)X_0 \\ +(v^2-2u+9v-1)X_1+(u^2-5v^2+9u+2v-3)X_2 \\ +7u^2-7v^2-3u+2v-6 \end{cases}$$

7.4. Unsolved example: $p = 400009$**,** $n = 5$**.** We propose to the reader to break our cryptosystem based on his understanding of our resolved example. The only elements given to the reader are the ciphered message and the automorphism π

7.4.1. *Ciphered message.* The ciphered message obtained with the cryptosystem is the following:

m310+a@axb:d;toa4)o!.cy3vw+w;nn4ds,iy9vF]{4G+u:ryjral)=_t?3nEG0G0G
ah@aisryswod!;CEG,u)'(xg[(3r(kCh_mje0tubq;Cl]H6{ejw4t,(_t,u+w).1oAg=fFA}G2wpjy
,w+2ah;5_6).uhGeH},()E9vn4yaoG{8}]?;(x)caA4AvgtD{e;Cjwxsm7bq?l_^['(3HEr(c_xs+
2vdtys,G2H6oGD}]wdFqryG2sqo:Fn:ntu;!ogvwF]rH!4q2g(E}m3i-mBHEg=alFjp_H
Eiso.g9=tBo4tjvc'i-D}B!oa.]cy,(?lfqvdau)Ei-G+C9B!7!.c)jw).xs=5vg

The character _ represent the blank character in this message

7.4.2. *TTM automorphism* π*.* We use the following TTM automorphism in this cryptosystem:

$$\pi = \begin{cases} X_0 \to (-140948u^2-52638v^2+61979u-60408v+46812)X_0^2 \\ +(-132380u^2+112517v^2+162813u+64389v+16914)X_0X_1 \\ +(-88311u^2+186065v^2+9058u-152291v-50514)X_1^2 \\ +(194882u^2-110784v^2-102548u-160323v-167276)X_0X_2 \\ +(-1771u^2+104665v^2-3230u-198435v+54327)X_1X_2 \\ +(79094u^2-152236v^2+147556u+87447v+3012)X_2^2 \\ +(69746u^2+110241v^2+27107u+126669v+198819)X_0X_3 \\ +(-4196u^2-117255v^2+181663u+124903v+101572)X_1X_3 \\ +(32146u^2-43910v^2+133537u+2899v-9599)X_2X_3 \\ +(-49330u^2-3246v^2-118083u+146061v-175122)X_3^2 \\ +(-69299u^2+146059v^2+170296u-185991v-150845)X_0X_4 \end{cases}$$

$$
\pi = \left\{ \begin{array}{l}
+(172937u^2 - 89544v^2 + 173397u - 83112v + 126904)X_1X_4 \\
+(740u^2 - 71367v^2 + 537u + 22289v - 6028)X_2X_4 \\
+(-66111u^2 + 148171v^2 - 9461u + 98984v + 163418)X_3X_4 \\
+(-22312u^2 + 149848v^2 + 80963u - 164954v + 170985)X_4^2 \\
+(156104u^2 - 116937v^2 - 105093u - 192789v - 120996)X_0 \\
+(-165691u^2 + 115379v^2 - 165654u - 12622v + 69664)X_1 \\
+(175408u^2 + 72950v^2 + 98227u + 62128v + 130385)X_2 \\
+(93808u^2 - 91986v^2 - 159580u + 154981v - 72347)X_3 \\
+(-49811u^2 - 101286v^2 - 170417u + 5256v - 184574)X_4 \\
+183841u^2 + 2187v^2 + 20532u - 98646v + 165837 \\
\\
X_1 \to (-9652u^2 - 77225v^2 + 119417u + 6221v + 27762)X_0^2 \\
+(165243u^2 + 37667v^2 + 90871u - 58692v + 192435)X_0X_1 \\
+(127886u^2 + 193379v^2 + 60455u - 142771v - 197548)X_1^2 \\
+(193845u^2 - 104716v^2 + 139226u - 45181v - 47567)X_0X_2 \\
+(-119793u^2 - 191874v^2 + 77933u - 11761v + 32037)X_1X_2 \\
+(129025u^2 + 132753v^2 + 59828u + 135916v - 60922)X_2^2 \\
+(191865u^2 + 130682v^2 - 129933u - 41030v + 111993)X_0X_3 \\
+(39114u^2 + 147952v^2 + 88160u - 189047v - 574)X_1X_3 \\
+(120111u^2 - 77666v^2 - 38443u - 8773v + 79346)X_2X_3 \\
+(85454u^2 - 194051v^2 - 7221u - 182540v + 158281)X_3^2 \\
+(126244u^2 - 184371v^2 + 81294u + 92506v + 176084)X_0X_4 \\
+(-77885u^2 + 130536v^2 - 176846u + 36717v + 45633)X_1X_4 \\
+(-163489u^2 + 145382v^2 + 190310u - 119467v + 260)X_2X_4 \\
+(-25330u^2 - 193020v^2 - 20931u - 24393v + 56035)X_3X_4 \\
+(38350u^2 - 9142v^2 + 112132u - 39742v - 526)X_4^2 \\
+(-66514u^2 - 188549v^2 + 29382u + 42101v + 189975)X_0 \\
+(-17403u^2 - 7013v^2 - 113103u - 100671v - 117315)X_1 \\
+(-37731u^2 + 180247v^2 - 182254u - 12866v + 155629)X_2 \\
+(21562u^2 - 85096v^2 + 185593u - 193837v + 195650)X_3 \\
+(51395u^2 - 33941v^2 + 164868u + 105638v + 104843)X_4 \\
+177671u^2 + 79442v^2 + 151953u - 122158v + 71951 \\
\\
X_2 \to (23571u^2 - 163714v^2 - 52186u - 56085v - 140845)X_0^2 \\
+(-27712u^2 - 55637v^2 + 145534u - 131055v + 7435)X_0X_1 \\
+(-33425u^2 - 51387v^2 + 130440u - 75819v - 87314)X_1^2 \\
+(-53089u^2 - 44855v^2 - 132011u + 77882v + 21822)X_0X_2 \\
+(179033u^2 + 12432v^2 + 50777u - 11695v + 199377)X_1X_2 \\
+(40365u^2 - 83327v^2 - 166164u - 199093v + 131553)X_2^2 \\
+(-41179u^2 + 72546v^2 - 112688u + 115037v + 109282)X_0X_3 \\
+(-101996u^2 - 46689v^2 + 178193u - 135981v + 128221)X_1X_3 \\
+(61462u^2 - 102859v^2 - 5143u - 153154v + 189260)X_2X_3 \\
+(-131271u^2 - 29104v^2 - 4485u + 176792v - 33179)X_3^2 \\
+(94003u^2 - 31651v^2 + 148267u - 124772v - 41618)X_0X_4 \\
+(86129u^2 + 68664v^2 + 167658u - 42586v - 64704)X_1X_4 \\
+(150077u^2 - 30922v^2 - 53257u + 138811v + 165510)X_2X_4 \\
+(118409u^2 - 9v^2 + 142680u - 67580v + 182436)X_3X_4 \\
+(144628u^2 + 133178v^2 + 162663u - 152997v - 43035)X_4^2 \\
+(65306u^2 - 81470v^2 - 17767u - 9190v + 52439)X_0 \\
+(-116598u^2 + 49070v^2 - 141833u - 1318v + 123696)X_1 \\
+(199930u^2 + 158763v^2 - 3971u + 172898v + 112037)X_2 \\
+(50835u^2 - 45351v^2 + 19000u + 195626v + 51147)X_3 \\
+(99799u^2 - 121293v^2 + 169415u + 143649v - 111146)X_4 \\
-4214u^2 + 129576v^2 + 197997u + 84811v + 118505 \\
\\
X_3 \to (190727u^2 + 19664v^2 + 159492u - 114493v - 92106)X_0^2 \\
+(-73162u^2 - 21788v^2 - 137193u + 75000v - 105935)X_0X_1 \\
+(-129213u^2 + 94931v^2 - 130797u - 84435v - 11285)X_1^2 \\
+(-38441u^2 - 109589v^2 + 155361u - 54798v - 26620)X_0X_2 \\
+(-119294u^2 - 181060v^2 + 34986u - 194904v - 150496)X_1X_2 \\
+(33524u^2 + 52885v^2 - 77723u - 51450v - 136813)X_2^2 \\
+(-104021u^2 - 81086v^2 + 196859u - 5816v + 176156)X_0X_3 \\
+(-51272u^2 - 28456v^2 + 42182u - 103824v - 63317)X_1X_3 \\
+(-182090u^2 + 96466v^2 - 58716u - 199215v - 77706)X_2X_3 \\
+(-189566u^2 - 192137v^2 + 11023u - 83591v + 169349)X_3^2 \\
+(102493u^2 - 124837v^2 - 151926u - 163801v - 58974)X_0X_4 \\
+(84519u^2 + 96886v^2 + 11018u + 82093v - 68270)X_1X_4 \\
+(-72684u^2 + 119964v^2 + 156980u + 2265v - 37392)X_2X_4 \\
+(-1849u^2 + 116425v^2 - 65209u + 60278v - 26331)X_3X_4 \\
+(-82991u^2 + 48833v^2 + 145015u - 5487v + 192811)X_4^2 \\
+(119042u^2 - 87231v^2 + 167541u - 137087v + 124701)X_0 \\
+(-114321u^2 - 147061v^2 + 123102u + 65083v + 27621)X_1
\end{array} \right.
$$

$$\pi = \begin{cases} +(-59463u^2 + 48050v^2 - 53120u + 134568v + 95104)X_2 \\ +(-25386u^2 + 26817v^2 + 191759u - 123418v + 43888)X_3 \\ +(-66851u^2 - 101406v^2 - 65885u - 22733v - 51366)X_4 \\ -166399u^2 + 53240v^2 - 148616u + 140738v - 160420 \\ \\ X_4 \to (117318u^2 - 176462v^2 + 95872u - 54347v - 50877)X_0^2 \\ +(-118736u^2 - 4210v^2 + 124969u + 22541v + 118107)X_0X_1 \\ +(-155056u^2 + 22217v^2 + 63247u - 86718v - 88796)X_1^2 \\ +(-59718u^2 + 4943v^2 + 130746u - 146453v - 57433)X_0X_2 \\ +(32755u^2 + 46582v^2 - 7733u - 142341v - 26446)X_1X_2 \\ +(106607u^2 - 64163v^2 - 116078u + 25673v - 114249)X_2^2 \\ +(-173935u^2 - 93616v^2 - 123032u - 183620v + 82153)X_0X_3 \\ +(-169382u^2 + 6494v^2 + 44325u + 130795v - 137688)X_1X_3 \\ +(-124606u^2 + 52108v^2 - 13635u - 69159v - 136297)X_2X_3 \\ +(-127940u^2 - 152344v^2 + 23227u - 99845v - 24590)X_3^2 \\ +(-187076u^2 - 115218v^2 + 137787u + 189772v - 67036)X_0X_4 \\ +(-22511u^2 - 46560v^2 + 179621u + 197531v + 46960)X_1X_4 \\ +(16799u^2 - 67232v^2 + 75304u + 60644v - 24681)X_2X_4 \\ +(84400u^2 - 77140v^2 - 178704u + 60919v - 179979)X_3X_4 \\ +(-64607u^2 + 141710v^2 - 81277u - 184401v - 33966)X_4^2 \\ +(-88086u^2 + 123870v^2 + 10448u + 96831v + 189828)X_0 \\ +(-7894u^2 - 46470v^2 + 58659u - 158303v + 61476)X_1 \\ +(81196u^2 + 87859v^2 - 139061u + 70790v + 129097)X_2 \\ +(198703u^2 - 88156v^2 + 70991u - 145701v + 41927)X_3 \\ +(58752u^2 + 153356v^2 - 29394u + 60609v - 182789)X_4 \\ -62344u^2 - 149500v^2 + 60295u + 13348v + 18053 \end{cases}$$

References

1. J. Ding, B. Yang, Multivariate Public Key Cryptography, In: (Bernstein, D.J., Buchmann, J., Dahmen, E. eds.) *Post-Quantum Cryptography*, Springer, 2009, 193–241.
2. J. Hrdina, M. Kureš, P. Vašík, A note on tame polynomial automorphisms and the security of TTM cryptosystem, *Appl. Comput. Math.* **9** (2010), no. 2, 226–233.
3. M. Kureš, L. Skula, Smith normal form of a matrix of generalized polynomials with rational exponents, *Annal. Univ. Mar. Curie Sklodow.* **62** (2008), no. 1, 81–90.
4. T. Moh, On tame transformation method (TTM), Lecture on the *Midwest Arithmetical Geometry in Cryptography Workshop*, at University of Illinois in November 1999.
5. J. Patarin, Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt '88, In: Coppersmith, Don (ed.), *Advances in cryptology - CRYPTO '95*; 15th annual International Cryptology Conference, Santa Barbara, CA, USA, August 27-31, 1995; Proceedings; Berlin: Springer-Verlag, Lect. Notes Comput. Sci. **963** (1995), 248–261.
6. S. Zu-hua, Public key cryptosystem and digital-signature schemes based on linear algebra over a local ring, *Compute and Digital Techniques, IEE Proceedings E* **134** (1987), no. 5, 254–256.

(Miroslav Kureš) Institute of Mathematics, Brno University of Technology, Technická 2, 61669 Brno, Czech Republic
E-mail address: kures@fme.vutbr.cz

(Thibault Decome) École Spéciale Militaire de Saint-Cyr, France
E-mail address: thibault.decome@wanadoo.fr

(Ghislain Drecourt) École Spéciale Militaire de Saint-Cyr, France
E-mail address: nioulp@gmail.com