sciendo

TATRA MOUNTAINS
Mathematical Publications

DOI: 10.2478/tmmp-2020-0013
Tatra Mt. Math. Publ. **77** (2020), 139–162

A PUBLIC KEY CRYPTOSYSTEM USING A GROUP OF PERMUTATION POLYNOMIALS

Rajesh P. Singh[1] — Bhaba K. Sarma[2] — Anupam Saikia[2]

[1]Central University of South Bihar, INDIA

[2]Indian Institute of Technology Guwahati, INDIA

ABSTRACT. In this paper we propose an efficient multivariate encryption scheme based on permutation polynomials over finite fields. We single out a commutative group $\mathfrak{L}(q,m)$ of permutation polynomials over the finite field F_{q^m}. We construct a trapdoor function for the cryptosystem using polynomials in $\mathfrak{L}(2,m)$, where $m=2^k$ for some $k\geq 0$. The complexity of encryption in our public key cryptosystem is $O(m^3)$ multiplications which is equivalent to other multivariate public key cryptosystems. For decryption only left cyclic shifts, permutation of bits and xor operations are used. It uses at most $5m^2+3m-4$ left cyclic shifts, $5m^2+3m+4$ xor operations and 7 permutations on bits for decryption.

1. Introduction

Public key cryptography is used in e-commerce for authentication and secure communication. The most widely used cryptosystems RSA and ECC (elliptic curve cryptosystems) are based on the problem of integer factorization and discrete logarithm, respectively.

Multivariate public key cryptosystems are based on the problem of solving a system of nonlinear equations over a finite field, which is proven to be NP--complete. MIC*, the first practical public key cryptosystem based on this problem was proposed in 1988 by T. Matsumoto and H. Imai [3]. The MIC* public key cryptosystem was based on the idea of hiding a permutation monomial x^{2^k}, for $k=2^t+1$ by two invertible linear transformations. Though MIC* was more efficient than RSA and ECC, it was broken by Patarin in 1995 [5]. In 1996,

2010 Mathematics Subject Classification: 11T71, 11T06.
Keywords: multivariate cryptography; permutation polynomials; linearized polynomials.
[1] Corresponding author.

Patarin [6] gave a generalization of MIC* called HFE. However, the HFE cryptosystem was not as efficient as the original MIC*. The basic instance of HFE was broken in 1999 [11]. The attack was based on the fact that every system of homogeneous quadratic polynomials has a matrix representation. Using this matrix representation a highly overdefined system of equations can be obtained which can be solved by a new technique called relinearization [11]. Other possible attacks on the HFE scheme can be found in [15], [20] and [21]. Patarin [7] investigated whether it is possible to repair MIC* with the same kind of computation efficiency. He designed some multivariate cryptosystems known as Little Dragon and Big Dragon with public key of multivariate polynomials of total degree 2 and 3 respectively, and with efficiency comparable to MIC*. In Dragon cryptosystems, the public key was of mixed type which is quadratic in plaintext variables and linear in ciphertext variables. However, Patarin found [7] that Dragon scheme with one hidden monomial is insecure. In 2010 [31] and in 2011 [30], Singh *et al.* proposed efficient Dragon type multivariate public key cryptosystems using permutation polynomials over finite fields. A public key scheme based on the composition of tame transformation methods (TTM) was proposed in 1999 [10]. This scheme has been broken in 2000 [13], where the cryptanalysis is reduced to an instance of the Min-Rank problem that can be solved within a reasonable time. In 2004 Ding [22] proposed a perturbed variant of MIC* called PMI. The PMI system attempts to increase the complexity of the secret key computations in order to increase security, using a system of r arbitrary quadratic equations over a finite field with the assumption that $r \ll n$, where n is the bitsize. PMI was broken by Fouque, Granboulan and Stern [23]. The trick of the attack on PMI is to use differential cryptanalysis to reduce the PMI system to the MIC* system. Another cryptosystem called Medium Field Equation (MFE) was proposed in 2006 [26] which was broken by Ding in 2007 [27] using high order linearization equation attack. A multivariate public key cryptosystem called the Simple matrix scheme (or ABC) was proposed by Tao *et al.* in 2013 [32]. This cryptosystem uses matrices of plaintext variables. In 2015 [35], Tao *et al.* presented a modified version of this cryptosystem which solves the decryption failure problem of the original Simple matrix scheme. In [38], [37], D. Moody *et al.* and Gu Chunsheng presented some attacks on Simple matrix scheme. Another multivariate public key cryptosystem called SRP was proposed by Yasude *et al.* in 2015 [36]. This cryptosystem combines several multivariate public key cryptosystems into one which seems to prevent some of the known attacks on multivariate cryptosystems. But, it was broken by R. Perlner *et al.* in 2017 [39]. One more multivariate encryption scheme called ZHFE was proposed by Jaiberth Porras *et al.* in 2017 [33]. This cryptosystem uses two high rank HFE polynomials to construct the public key. Unfortunately, this cryptosystem was also broken by Cabarcas *et al.* in 2017 [40] using the rank

attack. It is well known that multivariate cryptography is one of the directions of post quantum cryptography and we see that the most of the multivariate encryption schemes are broken. So, there is a strong motivation to develop new practical multivariate encryption schemes. In contrast to multivariate encryption schemes, there are several practical multivariate signature schemes which are secure and fast, (see [24], [12], [16], [34]). For a detailed introduction of multivariate public key cryptography, we refer the interested readers to [25]. An interesting introduction of hidden monomial cryptosystems can be found in [9].

Most of the multivariate public key cryptosystems use a quadratic function F hidden by two secret invertible linear transformations s and t. The structure is generally described as $y = t(F(s(x)))$, where x is the plaintext variable. Hiding $F(x)$ by two invertible linear transformations s and t is not working effectively. A close observation of the weaknesses in the existing multivariate public key cryptosystems based on this structure suggests that apart from exploring the use of new classes of higher degree functions it is appropriate to make modifications to the structure itself to build new multivariate cryptosystems. A possible alternative structure is

$$F_1(S_1(y)) = T(F(S(x))) \tag{1}$$

where S_1, T, S are invertible linear (or affine) transformations. The secret function $F(x)$ is quadratic and invertible (or injective) in the plaintext variable x. The function $F_1(y)$ is high degree and invertible in the ciphertext variable y. The structure (1) is a generalization of the existing structure in the sense that if S_1 and F_1 are taken to be the identity mappings, then the new structure is equivalent to the existing structure. The aim of this paper is to propose a cryptosystem with structure (1) and based on a new class of permutation polynomials. This is a highly revised and updated version of our multivariate cryptosystem from cryptology e-print archive [29]. In Section 2, we single out a commutative group $\mathfrak{L}(q, m)$ of permutation polynomials over the finite field $\mathbb{F}_{q^m}$, where q is a prime power. In Section 3, we use this group of permutation polynomials to construct a trapdoor function and propose a new cryptosystem with structure (1) based on the problem of solving a system of nonlinear equations over a finite field.

Like Dragon cryptosystems, the public key in our cryptosystem is of mixed type. The public key consists of two sets of quadratic multivariate polynomials. We make some security analysis in Section 4 and see that the proposed cryptosystem is secure against usual known attacks on existing multivariate public key cryptosystems. Computation with polynomials in the group $\mathfrak{L}(2, m)$ is fast, which makes the proposed cryptosystem efficient. The efficiency of the proposed cryptosystem is analyzed in Section 5 and some comparisons with HFE and

ZHFE cryptosystems are made in Section 6. A toy example is provided in Appendix A to give an idea how key generation, encryption and decryption work in the proposed cryptosystem.

2. Preliminaries

Let q be a prime power and $\mathbb{F}_q$ be the finite field of order q. An element ϑ in $\mathbb{F}_{q^m}$, the extension of $\mathbb{F}_q$ of degree m, is *normal* over $\mathbb{F}_q$ if the elements $\vartheta, \vartheta^q, \ldots, \vartheta^{q^{m-1}}$ form a basis (called a *normal basis*) of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. Suppose $\mathbb{B} = \{\vartheta, \vartheta^q, \ldots, \vartheta^{q^{m-1}}\}$ is a fixed normal basis of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. We identify the element $x = \sum_{i=0}^{m-1} x_i \vartheta^{q^i}$ of $\mathbb{F}_{q^m}$ with $(x_0, x_1, \ldots x_{m-1})$ and thereby $\mathbb{F}_{q^m}$ with $\mathbb{F}_q^m$, the set of all m-tuples over $\mathbb{F}_q$. When $q = 2$, we denote the Hamming weight of x by $w(x)$.

It is easy to see that the operation $x \mapsto x^q$ maps $x = (x_0, x_1, \ldots, x_{m-1})$ to $(x_{m-1}, x_0, \ldots, x_{m-2})$ which is just one cyclic shift of x. Hence the cost of exponentiating by q is negligible.

A polynomial $f(x)$ over $\mathbb{F}_q$ is called a *permutation polynomial* of $\mathbb{F}_q$ if it induces a one-to-one map on $\mathbb{F}_q$ onto itself. Permutation polynomials have been a subject of study for almost one and a half century and have applications in many fields including Cryptography (see [2], [4] and [1, Chapter 7]). Permutation polynomials have been used to design efficient multivariate public key cryptosystems [31], [30]. A polynomial $L(x)$ over $\mathbb{F}_{q^m}$ is called a *linearized polynomial* (or a *p-polynomial*) over $\mathbb{F}_q$, if

$$L(x) = \sum_{i=0}^{k} \alpha_i x^{q^i} \tag{2}$$

for some $\alpha_i \in \mathbb{F}_{q^m}$. A linearized polynomial $L(x)$ satisfies the following:

$$L(\beta+\gamma) = L(\beta) + L(\gamma) \quad \text{and} \quad L(a\beta) = aL(\beta) \ \text{ for all } \ \beta, \gamma \in \mathbb{F}_{q^m} \quad \text{and} \quad a \in \mathbb{F}_q.$$

Thus, $L : x \mapsto L(x)$ is a linear operator of the vector space $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. Consequently, $L(x)$ is a permutation polynomial of $\mathbb{F}_{q^m}$ if and only if 0 is the only root of $L(x)$ in $\mathbb{F}_{q^m}$.

Corresponding to an element $\alpha = (\alpha_0, \alpha_1, \ldots, \alpha_{m-1})$ of $\mathbb{F}_{q^m}$, we define a linearized polynomial $L_\alpha(x)$ on $\mathbb{F}_{q^m}$ by

$$L_\alpha(x) = \sum_{i=0}^{m-1} \alpha_i x^{q^i}. \tag{3}$$

It is known that each function on $\mathbb{F}_{q^m}$ is given by a unique polynomial of degree at most $q^m - 1$ (see Chapter 7 of [1]). Since the polynomial $L_\alpha(x)$ is of degree at most q^{m-1}, the polynomials $L_\alpha(x)$ generated from different α are distinct as functions on $\mathbb{F}_{q^m}$.

Let

$$\alpha = (\alpha_0, \alpha_1, \ldots, \alpha_{m-1}) \quad \text{and} \quad \beta = (\beta_0, \beta_1, \ldots, \beta_{m-1}) \quad \text{be elements of } \mathbb{F}_{q^m}.$$

The *convolution* $\alpha * \beta$ of α and β is defined by

$$\alpha * \beta = (\gamma_0, \gamma_1, \ldots, \gamma_{m-1}), \quad \text{where} \quad \gamma_r = \sum_{i=0}^{m-1} \alpha_i \beta_{(r-i) \bmod m}.$$

Clearly, $\alpha * \beta = \beta * \alpha$. It can be easily verified that convolution is an associative binary operation. Moreover, we have

$$L_\alpha(\beta) = \sum_{i=0}^{m-1} \alpha_i \beta^{q^i} = \alpha_0(\beta_0, \ldots, \beta_{m-1}) + \alpha_1(\beta_{m-1}, \beta_0, \ldots, \beta_{m-2}) + \cdots + \alpha_{m-1}(\beta_1, \beta_2, \ldots, \beta_0) = \alpha * \beta.$$

In particular, $L_\alpha(\beta) = L_\beta(\alpha)$. We denote by $L_\alpha \circ L_\beta$ the composition of the functions L_α and L_β. Then, for any $\gamma \in \mathbb{F}_{q^m}$ we have

$$L_\alpha \circ L_\beta(\gamma) = L_\alpha(\beta * \gamma) = \alpha * \beta * \gamma = L_{\alpha * \beta}(\gamma),$$

that is,

$$L_\alpha \circ L_\beta = L_{\alpha * \beta}.$$

With composition as multiplication and with usual addition, the polynomials L_α form a commutative ring $\mathfrak{L}_{\mathcal{R}}(q, m)$.

The ring $\mathfrak{L}_{\mathcal{R}}(q, m)$ is isomorphic to the commutative ring $F_q[U]/(U^m - 1)$ with unity. This is seen as follows: for $\alpha = (\alpha_0, \ldots, \alpha_{m-1})$ define

$$\psi L_\alpha = \sum_{i=0}^{m-1} \alpha_i U^i \in F_q[U]/(U^m - 1).$$

Then ψ is an isomorphism of $\mathfrak{L}_{\mathcal{R}}(q, m)$ onto $F_q[U]/(U^m - 1)$. Indeed, if

$$\alpha = (\alpha_0, \ldots, \alpha_{m-1}) \quad \text{and} \quad \beta = (\beta_0, \ldots, \beta_{m-1}) \in \mathbb{F}_{q^m},$$

then

$$\begin{aligned} (\psi L_\alpha)(\psi L_\beta) &= \left(\sum_{i=0}^{m-1} \alpha_i U^i \right) \left(\sum_{i=0}^{m-1} \beta_i U^i \right) \\ &= \sum_{r=0}^{m-1} U^r \left(\sum_{i+j \,=\, r \bmod m} \alpha_i \beta_j \right) \\ &= \psi(L_{\alpha * \beta}) = \psi(L_\alpha \circ L_\beta). \end{aligned}$$

Let $\mathfrak{L}(q, m)$ denote the group of units of $\mathfrak{L}_{\mathcal{R}}(q, m)$, that is, the group of all invertible linearized polynomials L_α.

We consider the case $q = 2, m = 2^k, k \geq 0$. An element $u = \sum_{i=0}^{m-1} \alpha_i U^i \in F_2[U]/(U^m - 1)$ is invertible if and only if $(\alpha_0, \ldots, \alpha_{m-1})$ has odd weight.

Thus, $L_\alpha \in \mathfrak{L}(2,m)$ if and only if $w(\alpha)$ is odd, and therefore $\mathfrak{L}(2,m)$ has order 2^{m-1}. Because $u^m = 1$ in $F_2[U]/(U^m - 1)$, the m times composition L_α^m of L_α with itself is L_ϑ, the identity of $\mathfrak{L}(2,m)$. In particular, order of L_α in $\mathfrak{L}(2,m)$ is a divisor of m and L_α^{m-1} is the inverse of L_α. Suppose α, β are elements in $\mathbb{F}_{2^{2^k}}$ of odd weight. Then $L_\alpha \circ L_\beta = L_{\alpha * \beta} \in \mathfrak{L}(2,m)$. This implies $L_\alpha(\beta) = \alpha * \beta$ is of odd weight. In particular, L_α permutes the elements of $\mathbb{F}_{2^{2^k}}$ which are of odd weight.

We summarize the previous discussion in the following proposition.

Proposition 2.1.

(a) *Let α and β be elements of $\mathbb{F}_{q^m}$. Then*

$$L_\alpha(\beta) = L_\beta(\alpha) = \alpha * \beta \quad \text{and} \quad L_\alpha \circ L_\beta = L_{\alpha*\beta}.$$

(b) *The invertible linearized permutation polynomials over $\mathbb{F}_{q^m}$ form a commutative group $\mathfrak{L}(q,m)$.*

(c) *Let $q = 2$ and $m = 2^k$, for some k. Then L_α is a permutation polynomial if and only if $w(\alpha)$ is odd. In particular, the group $\mathfrak{L}(2,m)$ has order 2^{m-1}.*

(d) *If $\alpha \in \mathbb{F}_{2^m}$ is of odd weight, then L_α permutes the elements in $\mathbb{F}_{2^m}$ which are of odd weight.*

3. The proposed public key cryptosystem

In this section we present a multivariate public key cryptosystem with structure (1) using the group $\mathfrak{L}(2,m)$, where $m = 2^k$ for some positive integer k. To obtain the quadratic polynomials we use convolution of bits. We have seen that the convolution of two binary strings is equivalent to the composition of corresponding linearized polynomials and that the convolution of two binary strings of odd weight is a binary string of odd weight. For $x \in \mathbb{F}_{2^m}$ and a positive integer ℓ let $(x)^\ell$ denote the ℓ times convolution of x with itself. We denote by $O\mathbb{F}_2^m$ the set of all elements of $\mathbb{F}_2^m$ of odd weight. For the rest of this article, $m = 2^k$ for some positive integer k.

Before presenting the cryptosystem, we consider the following results which will be required in the sequel.

Lemma 3.1. *Let $x = (x_0, x_1, \ldots, x_{m-1}) \in \mathbb{F}_2^m$ and ℓ be a positive integer less than 2^m. If $(x)^\ell = (h_0, h_1, \ldots, h_{m-1})$, then the coordinates h_i are polynomials in x_i of degree $w(\ell)$, where $w(\ell)$ denotes the Hamming weight of ℓ in the base-2 number system.*

P r o o f. We have

$$(x)^2 = (c_0, c_1, \ldots, c_{m-1}), \quad \text{where} \quad c_{2i+1} = 0 \quad \text{and} \quad c_{2i} = x_i + x_{\frac{m}{2}+i}.$$

Thus, the coordinates c_j of $(x)^2$ are linear functions of x_i. Consequently, the coordinates of $(x)^{2^u}$ are linear functions of x_i for any positive integer u. On the other hand, if $y = (y_0, y_1, \ldots, y_{m-1}) \neq x$ and each of y_j is a polynomial in x_i of degree $d < m$, then the coordinates of $x * y$ are polynomials in x_i of degree $d+1$. Therefore, the result follows by induction on the Hamming weight of ℓ. □

Lemma 3.2. *If ℓ is a positive integer co-prime to m, then the function $h_\ell : O\mathbb{F}_2^m \to O\mathbb{F}_2^m$ defined by $h_\ell(x) = (x)^\ell$ is a bijection.*

P r o o f. Since ℓ and m are co-prime, there exist integers r and s such that $\ell r = 1 + sm$. If $y = h_\ell(x) = (x)^\ell$, then $L_{(y)^r} = L_{(x)^{\ell r}} = L_x^{\ell r} = L_x^{1+sm} = L_x$, since $L_x^m = L_\vartheta$, the identity of $\mathfrak{L}(2, m)$. Consequently, $x = (y)^r$, and therefore, h_ℓ is invertible. □

Corollary 3.3. *If $a \in \mathbb{F}_2^m$ is of even weight and $b \in \mathbb{F}_2^m$ is of odd weight, then the function $x \mapsto a + b * (x)^{2m-1}$ is a bijection of $O\mathbb{F}_2^m$ onto itself.*

P r o o f. We have $b * (x)^{2m-1} = L_b((x)^{2m-1}) = L_b \circ h_{2m-1}(x)$ and $L_b \circ h_{2m-1}$ is a bijection of $O\mathbb{F}_2^m$ onto itself. Since the xor of a binary string of even weight and a binary string of odd weight is of odd weight, the given function is a bijection of $O\mathbb{F}_2^m$ onto itself. □

3.1. Public key generation

We consider a message to be a binary string $(x_0, x_1, \ldots, x_{m-2})$. We adjoin an additional bit x_{m-1} so that $X = (x_0, x_1, \ldots, x_{m-1})$ is a binary string of odd weight. After decryption one can just ignore the last bit x_{m-1}. Thus, the plaintext variable X is a generic element of $O\mathbb{F}_{2^m}$, and $x_i,\ 0 \leq i \leq m-1$, are the plaintext variables. In conformity with the structure (1), the ciphertext would be an element $Y = (y_0, \ldots, y_{2m-1})$ in $\mathbb{F}_2^{2m}$ which is to be computed by solving a system of linear equations in y_i. For generating the public key, which will be a set of degree three nonlinear equations, linear in y_i and quadratic in x_i, we use a set consisting of eight linearized permutation polynomials from the groups $\mathfrak{L}(2, m)$ and $\mathfrak{L}(2, 2m)$, eight permutations of bits and ten field elements.

Choose $L_{\alpha_i},\ 1 \leq i \leq 6$, in the group $\mathfrak{L}(2, m)$ and $L_{\beta_i},\ i = 1, 2,$, in the group $\mathfrak{L}(2, 2m)$. Let $\pi_i,\ 1 \leq i \leq 6$, be random permutations of the bits in $(t_0, \ldots, t_{m-1})$, and $\eta_i,\ i = 1, 2$, random permutations of the bits in $(t_0, \ldots, t_{2m-1})$. Choose nonzero elements $\sigma_i,\ 1 \leq i \leq 6$, of even weight in $\mathbb{F}_{2^m}$, and $\delta_i,\ i = 1, 2$ of even weight in $\mathbb{F}_{2^{2m}}$. Further, choose $\gamma_1 \in \mathbb{F}_{2^{2m}}$ of even weight and $\gamma_2 \in \mathbb{F}_{2^{2m}}$ of odd weight.

Define $T_i' = L_{\alpha_i} \circ \pi_i,\ 1 \le i \le 6$, and $S_i' = L_{\beta_i} \circ \eta_i,\ i = 1, 2$ where "$\circ$" denotes composition of mappings. Next, define the affine transformations

$$T_i = T_i' + \sigma_i,\ \ 1 \le i \le 6, \quad \text{and} \quad S_i = S_i' + \delta_i,\ \ i = 1, 2.$$

Clearly, T_i are bijections from $O\mathbb{F}_2^m$ onto itself, and S_i are bijections of $O\mathbb{F}_2^{2m}$ onto itself.

First, we transform the plaintext $X = (x_0, \ldots, x_{m-1})$ with T_1 and T_2 and put

$$X^{(1)} = T_1(X), \tag{4}$$

$$X^{(2)} = T_2(X). \tag{5}$$

Since T_1 and T_2 are affine transformations, the coordinates of $X^{(1)}$ and $X^{(2)}$ are linear polynomials in the plaintext variables x_i.

Next, we compute

$$X^{(3)} = T_3\left(\left(X^{(1)}\right)^2 * X^{(2)}\right) \tag{6}$$

and

$$X^{(4)} = T_4\left(X^{(1)} * X^{(2)}\right) + T_5\left(\left(X^{(1)}\right)^2 * X^{(2)}\right). \tag{7}$$

In the sequel, we will consider the function $F(X) = (X^{(3)}, X^{(4)})$ which maps an element X of $O\mathbb{F}_2^m$ into $\mathbb{F}_2^{2m}$. Suppose

$$X^{(3)} = (f_0, \ldots, f_{m-1}) \text{ and } X^{(4)} = (f_m, \ldots, f_{2m-1}). \tag{8}$$

In view of Lemma 3.1, the coordinates of $(X^{(1)})^2$ are linear expressions in the plaintext variables x_i, and therefore the coordinates $f_i, 0 \le i \le m-1$, of $X^{(3)}$ are quadratic polynomials in x_i. Similarly, the coordinates $f_i, m \le i \le 2m-1$, of $X^{(4)}$ are quadratic polynomials in x_i.

Since T_3 and T_4 are bijections of $O\mathbb{F}_2^m$ onto itself, it follows from (6) and (7) that for any X of odd weight $X^{(3)}$ is of odd weight and $X^{(4)}$ is of even weight. Consequently, $F : X \mapsto (f_0, f_1, \ldots, f_{2m-1})$ is a function from $O\mathbb{F}_2^m$ to $O\mathbb{F}_2^{2m}$.

Next, consider the functions $Y \mapsto Z = S_1(Y)$ and $Z \mapsto \gamma_1 + \gamma_2 * (Z)^{2m-1}$ from $O\mathbb{F}_2^{2m}$ into itself. Because S_1 is a bijection, Corollary 3.3 yields that $Y \mapsto \gamma_1 + \gamma_2 * (Z)^{2m-1}$ is a bijection of $O\mathbb{F}_2^{2m}$. Clearly, the coordinates of $\gamma_1 + \gamma_2 * (Z)^{2m-1}$ are polynomials in the coordinates y_i of Y.

We impose the following relation that is to be satisfied by the plaintext and ciphertext variables

$$S_2\left(F(T_6(X))\right) = \gamma_1 + \gamma_2 * (Z)^{2m-1}. \tag{9}$$

We have $\gamma_2 * (Z)^{2m} = L_Z^{2m}(\gamma_2) = L_\vartheta(\gamma_2) = \gamma_2$, where L_ϑ is the identity of $\mathfrak{L}(2, 2m)$. Taking convolution of each side of (9) with Z, we have the following relation between the plaintext and the ciphertext variables:

$$S_2(F(T_6(X))) * Z + \gamma_1 * Z + \gamma_2 = 0. \tag{10}$$

Since S_1 and S_2 are affine transformations, the coordinates of $Z = S_1(Y)$ are linear expressions in the ciphertext variables y_i and the coordinates of $S_2(F(T_6(X)))$ are quadratic polynomials in the plaintext variables $x_i, 0 \leq i \leq m-1$. Equating the coordinates of the left side of (10) to zero, we get a system of $2m$ polynomial equations, each of total degree 3 in the variables x_i and y_j, with degree 1 in the variables y_j. The system is of the form

$$\sum a_{rijl}x_ix_jy_l + \sum b_{ril}x_iy_l + \sum c_{rij}x_ix_j + \sum d_{rl}y_l + \sum e_{ri}x_i + f_r = 0 \quad (0 \leq r \leq 2m-1). \quad (11)$$

The system (11) is the required public key of the cryptosystem. Each of the equations in (11) will have some terms $x_ix_jy_l$ of degree three with nonzero coefficients. Each of them will also have terms x_iy_l of degree two and linear terms in x_i and y_l with nonzero coefficients, since σ and δ's are nonzero. Thus, given a ciphertext Y, i.e., a set of values of the variables y_l, the public equations (11) are nonlinear and non-homogeneous in x_i.

Each of equations in (11) is of degree three and so will have $O(m^3)$ terms. As there are $2m$ equations, total public key size will be of $O(m^4)$ terms. Though this seems to be large, it is possible to reduce the size to $O(m^3)$ terms which can be seen as follows:

Consider the coefficients $h_{rl} = \sum a_{rijl}x_ix_j$ of y_l in the r-th equation of (11). It follows from (10) that each h_{rl} is a linear combination of the second degree homogeneous parts of $f_0, f_1, \ldots, f_{2m-1}$. Therefore at most $2m$ quadratic polynomials h_{rl} can be independent. Writing each h_{rl} as a linear combination in λ quadratic polynomials $g_1, \ldots, g_\lambda$, $\lambda \leq 2m$, the public key can be expressed as two sets of quadratic equations

$$g_s = \sum h_{sij}x_ix_j,$$

and

$$\sum a'_{rs\,l}g_sy_l + \sum b_{ril}x_iy_l + \sum c_{rij}x_ix_j + \sum d_{rl}y_l + \sum e_{ri}x_i + f_r = 0,$$

where

$$1 \leq s \leq \lambda,\ 0 \leq r \leq 2m-1.$$

Since the reduction of the public key size to $O(m^3)$ terms can be done in polynomial time, it does not change the security of the cryptosystem.

3.2. Secret Key

The affine transformations $(T_1, T_2, T_3, T_4, T_5, T_6, S_1, S_2)$ and the field elements (γ_1, γ_2) form the required secret key.

3.3. Encryption

The encryption algorithm consists of the following two steps:

(1) First, substitute the plaintext $(x_0, \ldots, x_{m-1})$ in the $2m$ equations in (11) and get the $2m$ linear equations in ciphertext variables $y_i, 0 \leq 2m-1$.

(2) Next, using Gaussian elimination solve the system of linear equations to obtain the ciphertext $(y_0, \ldots, y_{2m-1})$.

It follows from the equation (9) that

$$S_2(F(T_6(X))) + \gamma_1 = \gamma_2 * \big(S_1(Y)\big)^{2m-1}$$

or

$$\big(S_2(F(T_6(X))) + \gamma_1\big) * (\gamma_2)^{2m-1} = \big(S_1(Y)\big)^{2m-1}.$$

For $\alpha \in O\mathbb{F}_{2^{2m}}$ one gets $L_\alpha = L_\alpha^{(2m-1)(2m-1)} = L_{(\alpha)^{(2m-1)(2m-1)}}$, and therefore

$$(\alpha)^{(2m-1)(2m-1)} = \alpha.$$

Thus, the plaintext variable X and the ciphertext variable Y satisfy the relation

$$S_1(Y) = \Big(\big(S_2(F(T_6(X))) + \gamma_1\big) * \gamma_2^{2m-1}\Big)^{2m-1}, \tag{12}$$

which yields the encryption function E as

$$\begin{aligned} E(X) = Y &= S_1^{-1}\left[\Big(\big(S_2(F(T_6(X))) + \gamma_1\big) * \gamma_2^{2m-1}\Big)^{2m-1}\right] \\ &= S_1^{-1}\Big[\big(S_2(F(T_6(X))) + \gamma_1\big)^{2m-1} * \gamma_2\Big]. \end{aligned} \tag{13}$$

Since $F(X)$ and γ_2 are of odd weight and γ_1 is of even weight, for a given $X \in O\mathbb{F}_{2^m}$ we note that $\big(S_2(F(T_6(X))) + \gamma_1\big)^{2m-1} * \gamma_2 \in O\mathbb{F}_{2^{2m}}$, and therefore the encryption function E from $O\mathbb{F}_{2^m}$ to $O\mathbb{F}_{2^{2m}}$ is well-defined.

Theorem 3.1. *The encryption function E is a bijection from $O\mathbb{F}_{2^m}$ to the set $E(O\mathbb{F}_{2^m})$ of all valid ciphertexts in $O\mathbb{F}_{2^{2m}}$.*

Proof. Suppose $X_1, X_2 \in O\mathbb{F}_{2^m}$ such that $E(X_1) = E(X_2)$. Then, we have

$$\big(S_2(F(T_6(X_1))) + \gamma_1\big)^{2m-1} * \gamma_2 = \big(S_2(F(T_6(X_2))) + \gamma_1\big)^{2m-1} * \gamma_2.$$

Let $T_6(X_1) = W_1$, $T_6(X_2) = W_2$. Taking $2m-1$ times convolutions of both sides we get

$$\big(S_2(F(W_1)) + \gamma_1\big) * (\gamma_2)^{2m-1} = \big(S_2(F(W_2)) + \gamma_1\big) * (\gamma_2)^{2m-1}.$$

Again taking the convolution of both sides with γ_2 and noting that $(\gamma_2)^{2m} = \vartheta'$, the identity of convolution, and that the linear transformation S_2 is invertible,

we have $F(W_1) = F(W_2)$, i.e.,

$$T_3\left(\left(W_1^{(1)}\right)^2 * W_1^{(2)}\right) = T_3\left(\left(W_2^{(1)}\right)^2 * W_2^{(2)}\right)$$

and

$$T_4\left(W_1^{(1)} * W_1^{(2)}\right) + T_5\left(\left(W_1^{(1)}\right)^2 * W_1^{(2)}\right) = T_4\left(W_2^{(1)} * W_2^{(2)}\right) + T_5\left(\left(W_2^{(1)}\right)^2 * W_2^{(2)}\right).$$

From these two relations we have

$$\left(W_1^{(1)}\right)^2 * W_1^{(2)} = \left(W_2^{(1)}\right)^2 * W_2^{(2)} \quad \text{and} \quad W_1^{(1)} * W_1^{(2)} = W_2^{(1)} * W_2^{(2)},$$

i.e.,

$$W_1^{(1)} * \left(W_2^{(1)} * W_2^{(2)}\right) = W_2^{(1)} * \left(W_2^{(1)} * W_2^{(2)}\right).$$

This implies $W_1^{(1)} = W_2^{(1)}$ or equivalently $T_1(W_1) = T_1(W_2)$ and therefore $X_1 = X_2$. □

3.4. Decryption

The decryption algorithm for the cryptosystem is as follows.

Input: Ciphertext $Y = (y_0, \ldots, y_{2m-1})$ and secret parameters $(T_1, T_2, T_3, T_4, T_5, T_6, S_1, S_2, \gamma_1, \gamma_2)$.

Output: Message X

1: $Z \leftarrow S_1(Y)$.
2: $Z_1 \leftarrow L_Z^{2m-2}(Z)$.
3: $Z_2 \leftarrow \gamma_1 + L_{\gamma_2}(Z_1)$.
4: $\Delta \leftarrow S_2^{-1}(Z_2)$.
5: $(t_0, \ldots, t_{2m-1}) \leftarrow \Delta$.
6: $\Delta_1 \leftarrow (t_0, \ldots, t_{m-1})$ and $\Delta_2 \leftarrow (t_m, \ldots, t_{2m-1})$.
7: $\Delta_3 \leftarrow T_3^{-1}(\Delta_1)$.
8: $\Delta_4 \leftarrow T_5(\Delta_3)$.
9: $\Delta_5 \leftarrow \Delta_2 + \Delta_4$.
10: $\Delta_6 \leftarrow T_4^{-1}(\Delta_5)$.
11: $\Delta_7 \leftarrow L_{\Delta_6}^{m-1}(\Delta_3)$.
12: $\Delta_8 \leftarrow T_1^{-1}(\Delta_7)$.
13: $X \leftarrow T_6^{-1}(\Delta_8)$.
14: Return X.

We now prove that the above algorithm gives the valid plaintext X as the output for a ciphertext Y.

Theorem 3.2. *Given ciphertext Y, the output X given by the decryption algorithm is the valid plaintext.*

P r o o f. First, note that $L_Z^{2m-2}(Z) = L_{(Z)^{2m-2}}(Z) = (Z)^{2m-2} * (Z) = (Z)^{2m-1}$, where $Z = S_1(Y)$ and $\gamma_1 + L_{\gamma_2}\left((Z)^{2m-1}\right) = \gamma_1 + \gamma_2 * (Z)^{2m-1}$. The relation between plaintext and ciphertext is $S_2\left(F(T_6(X))\right) = \gamma_1 + \gamma_2 * (Z)^{2m-1}$, i.e., $\left(f_0^{'}, \ldots, f_{2m-1}^{'}\right) = S_2^{-1}\left(\gamma_1 + \gamma_2 * (Z)^{2m-1}\right)$, where $F(T_6(X)) = \left(f_0^{'}, \ldots, f_{2m-1}^{'}\right)$.

In the first four steps, the decryption algorithm computes

$$S_2^{-1}\left(\gamma_1 + \gamma_2 * (Z)^{2m-1}\right).$$

Now suppose that Δ_1, Δ_2 denote the first m bits and last m bits of

$$S_2^{-1}\left(\gamma_1 + \gamma_2 * (Z)^{2m-1}\right), \quad \text{respectively.}$$

Then, we have

$$\left(f_0^{'}, \ldots, f_{m-1}^{'}\right) = \Delta_1, \text{ and } \left(f_m^{'}, \ldots, f_{2m-1}^{'}\right) = \Delta_2,$$

i.e.,

$$T_3\left(\left(W^{(1)}\right)^2 * W^{(2)}\right) = \Delta_1 \text{ and } T_4\left(W^{(1)} * W^{(2)}\right) + T_5\left(\left(W^{(1)}\right)^2 * W^{(2)}\right) = \Delta_2,$$

where $W = T_6(X)$. From these two relations, we have

$$\left(W^{(1)}\right)^2 * W^{(2)} = T_3^{-1}(\Delta_1) \text{ and } W^{(1)} * W^{(2)} = T_4^{-1}\Big(\Delta_2 + T_5\big(T_3^{-1}(\Delta_1)\big)\Big).$$

Suppose,

$$\Delta_6 = T_4^{-1}\Big(\Delta_2 + T_5\big(T_3^{-1}(\Delta_1)\big)\Big) \text{ and } \Delta_3 = T_3^{-1}(\Delta_1).$$

Then we have $W^{(1)} * \Delta_6 = \Delta_3$ or, equivalently, $L_{\Delta_6}\left(W^{(1)}\right) = \Delta_3$. Therefore,

$$W^{(1)} = L_{\Delta_6}^{-1}(\Delta_3) = L_{\Delta_6}^{m-1}(\Delta_3), \quad \text{i.e.,} \quad W = T_1^{-1}\big(L_{\Delta_6}^{m-1}(\Delta_3)\big).$$

Finally, $X = T_6^{-1}\Big(T_1^{-1}\big(L_{\Delta_6}^{m-1}(\Delta_3)\big)\Big)$. □

3.5. Parameters

We suggest the message to be a binary string of length 127. We adjoin one extra bit to make the weight odd. Thus the message is an element of odd weight of finite field $\mathbb{F}_{2^m}$, $m = 128$. For $m = 128$, the public key consists of 256 equations of degree three which are quadratic in the plaintext variables x_i and linear in the ciphertext variables y_j. As mentioned, the public size can be reduced by writing it as two sets of $2m$ quadratic equations in the plaintext and ciphertext variables. So, the public key consists of 512 quadratic equations in the plaintext and ciphertext variables. For the secret keys generation, we have to select random even weight and odd weight elements of finite fields $\mathbb{F}_{2^{128}}$ and $\mathbb{F}_{2^{256}}$ and random permutations on 128 bits and 256 bits.

4. Security of the proposed Cryptosystem

Similar to other families of post-quantum cryptosystems, the security of multivariate cryptosystems is still not completely understood. In terms of provable security, there exist hardly any rigorous proofs which reduce the security of multivariate schemes to hard mathematical problems [41]. In this section, we investigate the security of the proposed cryptosystem against all known attacks.

Most of the multivariate public key cryptosystems use the structure $t(f(s(x)))$. However, hiding $f(x)$ by two linear transformations s and t does not work very effectively (see the attack of Kipnis and Shamir on HFE [11]). In our cryptosystem the ciphertext variable Y and the plaintext variable X are connected by the relation

$$F_1\big(S_1(Y)\big) = S_2\big(F(T_6(X))\big),$$

where

$$F_1(Y) = \gamma_1 + \gamma_2 * (Y)^{2m-1} \quad \text{and} \quad F(X) = \left(X^{(3)}, X^{(4)}\right).$$

We recall that

$$X^{(3)} = T_3\left(\left(X^{(1)}\right)^2 * X^{(2)}\right)$$

and

$$X^{(4)} = T_4\left(X^{(1)} * X^{(2)}\right) + T_5\left(\left(X^{(1)}\right)^2 * X^{(2)}\right),$$

where $X^{(1)} = T_1(X)$, $X^{(2)} = T_2(X)$. In our cryptosystem the linear transformations $T_1, T_2, T_3, T_4,$ and T_5 are secrets. So the quadratic function $F(X)$ is secret. It is evident that if we take F_1 and S_1 as identity functions, the relation between the plaintext variable X and the ciphertext variable Y becomes $Y = S_2(F(T_6(X)))$. Thus, if it is possible to attack our structure, then it is also possible to attack $t(f(s(x)))$ structure. This proves that our structure is at least as secure as the commonly used structure in multivariate cryptography, that is $t(f(s(x)))$.

We discuss now some known attacks developed for multivariate cryptosystems and see whether these attacks are applicable to the proposed cryptosystem.

4.1. Linearization equation attacks

Patarin [5] used the following idea to break MIC*: in case the function applied is $F(X) = X^{q^i+1} = \{f_0, f_1, \ldots, f_{m-1}\}$, then it is possible to obtain quadratic relations between the plaintext variables $(x_0, x_1, \ldots, x_{m-1})$ and the ciphertext variables $(y_0, y_1, \ldots, y_{m-1})$ of the form:

$$\sum_{i=0}^{m-1}\sum_{j=0}^{m-1} a_{ij}x_iy_j + \sum_{i=0}^{m-1} b_ix_i + \sum_{j=0}^{m-1} c_jy_j + d = 0. \tag{14}$$

By taking at least $(m+1)^2$ different plaintext and ciphertext pairs, a linear system of equations can be obtained and solved for the unknown constants a_{ij}, b_i, c_j and d.

We emphasize here that in the proposed cryptosystem, it is possible to find a relation which is linear in the plaintext variables (but nonlinear in the ciphertext variables), in the line of Patarin's linearization equation attack on MIC*.

From the relation (9) we have $F(W) = S_2^{-1}(Z')$, where $Z' = \gamma_1 + \gamma_2 * (Z)^{2m-1}$, $Z = S_1(Y)$ and $W = T_6(X)$. Therefore, $S_2^{-1}(Z')$ will give nonlinear polynomials of degree $w(2m-1)$ in the ciphertext variables. Suppose $S_2^{-1}(Z') = (Z_0, \ldots, Z_{2m-1})$. Then we have the following relations between the plaintext and the ciphertext:

$$T_3\left(\left(W^{(1)}\right)^2 * W^{(2)}\right) = (Z_0, \ldots, Z_{m-1}), \tag{15}$$

and

$$T_4\left(W^{(1)} * W^{(2)}\right) + T_5\left(\left(W^{(1)}\right)^2 * W^{(2)}\right) = (Z_m, \ldots, Z_{2m-1}). \tag{16}$$

Using these two relations we get the following relation:

$$W^{(1)} * T_4^{-1} \circ T_5(Z') + W^{(1)} T_4^{-1}(Z_m, \ldots, Z_{2m-1}) + Z', \tag{17}$$

where $W^{(1)} = T_1(W) = T_1(T_6(X))$, $Z' = T_3^{-1}(Z_0, \ldots, Z_{m-1})$ and T_1, T_2, T_3, T_4, T_5, T_6 and S_1 are linear transformations. Note that the relation (17) is linear in the plaintext but of degree $w(2m-1)$ in the ciphertext. Thus, the total degree of (17) is $w(2m-1)+1$, far from being quadratic. Moreover, the degree is a function of m. Therefore, to attack the cryptosystem with Patarin's tool, we need Gaussian reduction on $O\left(m^{w(2m-1)+1}\right)$ terms which is impractical. For example, for bit size equal to 64, we have $m = 64$ and $w(2m-1)+1 = 8$.

4.2. Attacks with differential cryptanalysis

Differential cryptanalysis has been successfully used earlier to attack the symmetric cryptosystems. In recent years differential cryptanalysis has emerged as a powerful tool to attack the multivariate public key cryptosystems, too. In 2005 [23] Fouque, Granboulan and Sterm used differential cryptanalysis to attack the multivariate cryptosystems. The key point of this attack is that in case of quadratic polynomials the differential of public key is a linear map and its kernel or its rank can be analyzed to get some information on the secret key. For any multivariate quadratic function $G : \mathbb{F}_q^n \to \mathbb{F}_q^m$ the differential operator between any two points $x, k \in \mathbb{F}_q^n$ can be expressed as

$$L_{G,k}G(x+k) - G(x) - G(k) + G(0)$$

and in fact that operator is a bilinear function. By knowing the public key of a given multivariate quadratic scheme and by knowing the information about the nonlinear part (x^{q^i+1}) they showed that for certain parameters it is possible to recover the kernel of $L_{G,k}$. This attack was successfully applied on Ding's cryptosystem [22] and afterwards using the same technique Dubois, Fouque, Shamir

and Sterm in 2007 [28] have completely broken all versions of the SFLASH signature scheme proposed by Patarin, Courtois, and Goubin [17]. In our cryptosystem, the encryption function is

$$E(X) = Y = S_1^{-1}\Big[\big(S_2(F(T_6(X))) + \gamma_1\big)^{2m-1} * \gamma_2\Big]$$

which is not quadratic but of total nonlinear degree $2w(2m-1)$ in plaintext variables $x_0, x_1, \ldots, x_{m-1}$. So in case of our encryption function, the differential operator is not a bilinear function. Thus to attack our cryptosystem by the methods of [23] and [28] is not feasible.

4.3. Attacks using the univariate polynomial representation of multivariate public polynomials

The fact that any function from a finite field into itself can be represented by a univariate polynomial is sometimes used to attack multivariate cryptosystems (see [25] for example). In our case, the encryption function is from the finite field $\mathbb{F}_{2^m}$ to the finite field $\mathbb{F}_{2^{2m}}$, and therefore we cannot represent the encryption function by a polynomial directly. It is possible to have such a representation by introducing dummy variables $x_m, x_{m+1}, \ldots, x_{2m}$. Note that the encryption function $E(X)$ in our Cryptosystem is of total nonlinear degree $2.w(2m-1)$ (see 13). By Lemma 3.3 of [11], the degree of the univariate polynomial representation is not constant but it is a function of m. Thus, the degree and the number of nonzero terms of the univariate polynomial representation of encryption function are both $O(m^m)$. The complexity of root finding algorithms, Berlekamp algorithm for example, is polynomial in the degree of the polynomial. This results in an exponential time algorithm for finding the roots of univariate polynomial. Therefore, this approach is less efficient than the exhaustive search.

4.4. Gröbner basis attacks

After substituting the ciphertext in the public key, one can get $2m$ quadratic equations in m variables and then Gröbner basis techniques can be applied to solve the system. The classical algorithm for solving systems of multivariate equations is Buchberger's algorithm for constructing Gröbner basis (see [8]). Theoretically, it can solve all the multivariate quadratic equations. However, its complexity is exponential in the number of variables, although there is no closed-form formula for it. In the worst case, the Buchberger's algorithm is known to run in double exponential time and on average its running time seems to be single exponential (see [14]). There are some efficient variants F_4 and F_5 of Buchberger's algorithm given by Jean-Charles Faugere (see [18] and [19]). The complexity of computing a Gröbner basis for the public polynomials of the basic HFE scheme is not feasible using Buchberger's algorithm. However, it is completely feasible using the algorithm F_5. The complexities of solving the public polynomials of several instances of the HFE using the algorithm F_5 are provided

in [21]. Moreover, it has been expressed in [21], "A crucial point in the cryptanalysis of HFE is the ability to distinguish a randomly algebraic system from an algebraic system coming from HFE". Instead of using any polynomial of special form we are using convolution operation to construct the public polynomials. Moreover our public key is of mixed type, this means, for different ciphertexts we will get different system of quadratic polynomial equations, so in our public key the quadratic polynomials look random. We have already seen that the degree of the univariate polynomial representation of the encryption function is proportional to m. It is explained in [21] that in this case there does not seem to exist polynomial time algorithm to compute the Gröbner basis. Hence, attack on our cryptosystem by Gröbner basis method is not feasible.

4.5. Relinearization, XL and FXL algorithms

Relinearization, XL or FXL algorithm are some techniques to solve the quadratic equations directly. The relinearization technique is developed in [11] for solving an overdefined system of quadratic equations. However, it is shown in [14] that the relinearization technique is not as efficient as one may expect since many of newly generated equations are dependent. Therefore, a technique called XL (extended relinearization) has been proposed in [14]. It is claimed to be the best algorithm for solving overdefined multivariate equations. However, when the number of equations is $m + r$ for some $1 \leq r \leq m$, then it is proved in [14] that XL has exponential complexity. A variant of the XL algorithm called FXL, was introduced in [14]. In this algorithm some variables are guessed to make the system slightly overdefined. Then the XL algorithm is applied. The main question is how many variables must be guessed. Although more guesses make the system more unbalanced, they add to the complexity of the algorithm. The optimum number of guesses is provided in [14].

In the case of applying XL, $r = m$ in our cryptosystem. Hence, XL algorithm cannot be used directly to attack our cryptosystem, since it has exponential complexity. Even using the optimum value for the number of variables guessed in the nonlinear equation, FXL has the exponential complexity for solving the system of public polynomials in the proposed cryptosystem. Hence, the FXL algorithm is not applicable to our cryptosystem.

5. Complexity and number of operations for encryption and decryption

5.1. Encryption

The public key in our cryptosystem consists of $2m$ equations of the form (11). There are $O(m^2)$ terms of the form $x_i x_j$ in each of the $2m$ equations of the public key so the complexity of evaluating public key at message block $x_0,\ x_1, \ldots, x_{m-1}$

is $O(m^3)$. The next step of encryption is to solve the $2m$ linear equations in $2m$ ciphertext variables $y_0, y_1, \ldots, y_{2m-1}$, which can be done efficiently by Gaussian elimination in $O(m^3)$ complexity. Hence the total complexity of encryption is $O(m^3)$.

5.2. Decryption

Decryption in the cryptosystem is fast, because it uses only permutations, cyclic shifts and xor operation of bits. Though the exact number of operations will depend on the chosen secret keys, we can count here the upper limits of the number of these operations.

For $\alpha, \beta \in \mathbb{F}_{2^m}$, the computation of $\alpha * \beta = L_\alpha(\beta)$ requires at most $m-1$ left cyclic shifts and $m-1$ xor operations. To operate T_i or T_i^{-1} $(1 \leq i \leq 6)$ on a m bit string, we need one permutation on bits, at most $m-1$ left cyclic shifts and at most m xor operations. To operate S_i or S_i^{-1} $(i = 1, 2)$ on a $2m$ bit string, we need one permutation on $2m$ bits, at most $2m-1$ left cyclic shifts and at most $2m$ xor operations. Thus, we need at most $5m^2 + 3m - 4$ left cyclic shifts and $5m^2 + 3m + 4$ xor operations for the decryption, in total. In addition we need exactly 2 permutations of $2m$ bit strings and 5 permutations on m bit strings.

6. Comparison with HFE and ZHFE cryptosystems

In this section, we compare our cryptosystem with HFE [6] and a variant of it called ZHFE [33]. In our cryptosystem the complexity of encryption is $O(m^3)$, i.e., equivalent to that of HFE and ZHFE. But the decryption is faster than HFE and ZHFE. In HFE the decryption is slow because one needs to compute the roots of a polynomial. The decryption complexity of HFE is $O\left(n^4 d^2 log(d)\right)$ where d is the degree of HFE polynomial. Note that for security reasons one cannot take smaller degree. Due to this the decryption process in HFE is slow. In our cryptosystem we are using left cyclic shifts and xor operations resulting in a much faster decryption process. In our cryptosystem we need $O(m^2)$ left cyclic shifts and $O(m^2)$ xor operations to decrypt a message. The efficiency of HFE and ZHFE are equivalent, see [33]. So, the decryption in our cryptosystem is much faster than in HFE and ZHFE cryptosystems. Public key size of HFE and ZHFE is $O(m^3)$ terms. In our cryptosystem, public key size is bigger than HFE and ZHFE but it is also $O(m^3)$ as it is possible to write public key as two sets of quadratic public polynomials. Secret key generation in our public key cryptosystem is faster than HFE and ZHFE because for secret keys we have to select random odd weight and even weight binary strings and random permutations.

7. Conclusion

In this paper we have designed an efficient multivariate public key cryptosystem using a group of Linearized permutation polynomials over finite fields. The complexity of encryption, $O(m^3)$, is equivalent to that of other multivariate cryptosystems. Computation with polynomials in the group $\mathfrak{L}(2, m)$ is fast, which makes the decryption in the proposed cryptosystem fast. We have given the security analysis of our cryptosystem against the known attacks.

Acknowledgement. We are grateful to Prof. Rana Barua, Indian Statistical Institute, Kolkata, India, for his valuable suggestions. We thank the anonymous referees for carefully reading the paper and giving constructive suggestions to improve the paper.

REFERENCES

[1] LIDL, R.—NIEDERREITER, H.: *Finite Fields*. Addision-Wesley, 1983.
[2] LIDL, R.— MULLEN, G. L.: *When does a polynomial over a finite field permute the elements of the field?*, The American Math. Monthly, **95** (1988), no. 3, 243–246.
[3] MATSUMOTO, T.—IMAI, H.: *Public quadratic polynomial-tuples for efficient signature--verification and message-encryption*. In: *Advances in cryptology-Eurocrypt '88 (Davos, 1988)*, *Lecture Notes in Comput. Sci., Vol. 330*, Springer-Verlag, 1988, pp. 419–453.
[4] LIDL, R.— MULLEN, G. L.: *When does a polynomial over a finite field permute the elements of the field? II*, Amer. Math. Monthly, **100** (1993), no. 1, 71–74.
[5] PATARIN, J.: *Cryptanalysis of Matsumoto and Imai public key scheme of Eurocrypt '88*. In: *Advances in Cryptology- Crypto '95*, *Lecture Notes in Comput. Sci., Vol. 963*, Springer--Verlag, Berlin, 1995, pp. 248–261.
[6] PATARIN, J.: *Hidden Field equations (HFE) and isomorphism of polynomials (IP): two new families of asymmetric algorithms*. In: *Advances in cryptology- Eurocrypt '96*, *Lecture Notes in Comput. Sci., Vol. 1070*, Springer-Verlag, Berlin, 1996, pp. 33–48.
[7] PATARIN, J.: *Asymmetric cryptography with a hidden monomial*. In: *Advances in Cryptology-Crypto '96*, *Lecture Notes in Comput. Sci., Vol. 1109*, Springer-Verlag, Berlin, pp. 45–60, 1996.
[8] COX, D.—LITTLE, J.—O'SHEA, D.: *Ideals, Varieties, and Algorithms. An Introduction to Computational Algebraic Geometry and Commutative Algebra (2nd edition)*. Undergraduate Texts in Mathematics, Springer-Verlag, New York, 1997.
[9] KOBLITZ, N.: *Algebraic Aspects of Cryptography*. Springer-Verlag, Berlin, 1998.
[10] MOH, T. T.: *A public key system with signature and master key functions*, Commun. Algebra, **27**(1999), no. 5, 2207–2222.
[11] KIPNIS, A.—SHAMIR, A.: *Cryptanalysis of the HFE public key cryptosystem by relinearization*. In: *Advances in cryptology—CRYPTO '99, Santa Barbara, CA*, *Lecture Notes in Comput. Sci., Vol. 1666*, Springer, Berlin, pp. 19–30, 1999.
[12] KIPNIS, A.—PATARIN, J.—GOUBIN, L.: *Unbalanced oil and vinegar signature scheme*. In: *EUROCRYPT 1999*, *Lecture Notes in Comput. Sci., Vol. 1592*, Springer-Verlag, Berlin, 1999, pp. 206–222.
[13] GOUBIN, L.—COURTOIS, N. T.: *Cryptanalysis of the TTM cryptosystem*. In: *Adv. Cryptol. ASIACRYPT, 2000*, *Lecture Notes in Comput. Sci., Vol. 1976*, Springer-Verlag, Berlin, 2000. pp. 44–57,

[14] COURTOIS, N. T.—KLIMOV, A.—PATARIN, J.—SHAMIR, A.: *Efficient algorithm for solving overdefined system of multivariate polynomial equations*, In: *EUROCRYPT 2000. Lecture Notes in Comput. Sci., Vol. 1807*, Springer-Verlag, Berlin, 2001, pp. 392–407.

[15] COURTOIS, N. T.: *The security of hidden field equations (HFE).* In : *CT-RSA 2001, Lecture Notes in Comput. Sci., Vol. 2020*, Springer-Verlag, Berlin, 2001, pp. 266–281.

[16] PATARIN, J.—COURTOIS, N. T.—GOUBIN, L.: *QUARTZ, 128-bit long digital signatures.* In: *CTRSA 2001, Lecture Notes in Comput. Sci., Vol. 2020*, Springer-Verlag, Berlin, 2001, pp. 282-297.

[17] PATARIN, J.—COURTOIS, N. T.—GOUBIN, L.: *FLASH, a fast multivariate signature algorithm.* In: *CT-RSA01 2001, Lecture Notes in Comput. Sci., Vol. 2020*, Springer-Verlag, Berlin, 2001, pp. 298–307.

[18] FAUGÈRE, J.-C.: *A new efficient algorithm for computing Gröbner bases* (F_4), J. Pure Appl. Algebra, **139**(2002), 61–88.

[19] FAUGÈRE, J.-C.: *A new efficient algorithm for computing Gröbner bases without reduction to zero* (F_5). In: *International Symposium on Symbolic and Algebraic Computation––ISSAC 2002*, ACM Press, New York, pp. 75–83.

[20] COURTOIS, N. T.—DAUM, M.—FELKE, P.: *On the security of HFE, HFEv- and quartz.* In: *PKC 2003, Lecture Notes in Comput. Sci., Vol. 2567*, Springer-Verlag, Berlin, 2003, pp. 337–350.

[21] FAUGERE, J.-C.—JOUX, A.: *Algebraic cryptanalysis of hidden field equation (HFE) cryptosystems using Gröbner basis.* In: *CRYPTO 2003, Lecture Notes in Comput. Sci., Vol. 2729*, Springer-Verlag, Berlin, 2003, pp. 44–60,

[22] DING, J.: *A new variant of the Matsumoto-Imai cryptosystem through perturbation.* In: *PKC04, Lecture Notes in Comput. Sci., Vol. 2947*, Springer-Verlag, Berlin, 2004, pp. 305–318.

[23] FOUQUE, P.-A.—GRANBOULAN, L.—STERM, J.: *Differential cryptanalysis for multivariate schemes.* In: *in EUROCRYPT 2005, Lecture Notes in Comput. Sci., Vol. 3494*, Springer-Verlag, Berlin, 2005, pp. 341–353.

[24] DING, J.—SCHMIDT, D. S.: *Rainbow, a new multivariate polynomial signature scheme.* In: *ACNS 2005, Lecture Notes in Comput. Sci., Vol. 3531*, Springer-Verlag, Berlin, 2005, pp. 164–175.

[25] DING, J.—GOWER, J. E.—SCHMIDT, D. S.: *Multivariate Public Key Cryptosystems*, Springer-Verlag, Berlin, 2006.

[26] WANG, L.-C.—YANG, B.-Y.—HU, Y.-H.—LAI, F.: *A medium-field multivariate public key encryption scheme.* In: *CT-RSA 2006: The Cryptographers Track at the RSA Conference 2006, Lecture Notes in Comput. Sci., Vol. 3860*, Springer-Verlag, Berlin, 2006. pp. 132–149.

[27] DING, J.—HU, L.—NIE, X.—LI, J.—WAGNER, J.: *High order linearization equation (HOLE) attack on multivariate public key cryptosystems.* In: *PKC 2007, Lecture Notes in Comput. Sci., Vol. 4450*, Springer-Verlag, Berlin, 2007. pp. 233–248,

[28] DUBOIS, V.—FOUQUE, P.-A.—SHAMIR, A.—STERN, J.: *Practical cryptanalysis of SFLASH.* In: *Advances in Cryptology-Crypto 2007, Lecture Notes in Comput. Sci., Vol. 4622*, Springer-Verlag, Berlin, 2007, pp. 1–12.

[29] SINGH, R. P.—SARMA, B. K.—SAIKIA, A.: *Public key cryptography using permutation P-polynomials over finite fields*, Cryptology eprint archive, 2009/208, `https://eprint.iacr.org/2009/208`

[30] SINGH, R. P.—SAIKIA, A.—SARMA, B. K.: *Little Dragon Two: An efficient multivariate public key cryptosystem*, Int. J. Network Security and Appl, **2** (2010), no. 2, 1–10.

[31] SINGH, R. P.—SAIKIA, A.—SARMA, B. K.: *Poly-dragon: an efficient multivariate public key cryptosystem*, Journal of Mathematical Cryptology, **4** (2011), no. 4, 349–364.

[32] TAO, C.—DIENE, A.—TANG, S.—DING, J.: *Simple-matrix scheme for encryption*. In: *PQCrypto 2013*, *Lecture Notes in Comput. Sci., Vol. 7932*, Springer-Verlag, Berlin, 2013, pp. 231–242.

[33] PORRAS, J.—BAENA, J.—DING, J.: *ZHFE, a new multivariate public key encryption scheme*. In: *PQCrypto 2014*, *Lecture Notes in Comput. Sci., Vol. 8772*, Springer-Verlag, Berlin, 2014, pp. 229–245.

[34] PETZOLDT, A.—CHEN, M.-S.—YANG, B.-Y.—TAO, C.—DING, J.: *Design principles for HFEv- based signature schemes*. In:*ASIACRYPT 2015 (Part I)*, *Lecture Notes in Comput. Sci., Vol. 9452*, Springer-Verlag, Berlin, 2015, pp. 311–334.

[35] TAO, C.—XIANG, H.—PETZOLDT, A.—DING, J.: *Simple Matrix - a multivariate public key cryptosystem (MPKC) for encryption*, Finite Fields Appl. **35** (2015), 352–368.

[36] YASUDA, T.—SAKURAI, K.: *A multivariate encryption scheme with Rainbow*. In: *ICISC 2015*, *Lecture Notes in Comput. Sci., Vol. 9543*, Springer-Verlag, Berlin, 2015, pp. 222–236.

[37] CHUNSHENG, G.: *Cryptanalysis of simple matrix scheme for encryption*, Cryptology eprint archive, 2016/1075, `https://eprint.iacr.org/2016/1075`

[38] MOODY, D.—PERLNER, R.—SMITH-TONE, D.: *Key recovery attack on the cubic ABC simple matrix multivariate encryption scheme*. In: *Selected Areas in Cryptography––SAC 2016*, *Lecture Notes in Comput. Sci., Vol. 10532*, Springer-Verlag, Berlin, 2016, pp. 543–558.

[39] MOODY, D.—PETZOLDT, A.—SMITH-TONE, D.: *Key recovery attack on the cubic ABC simple matrix multivariate encryption scheme*. In: *Selected Areas in Cryptography––SAC 2017*, *Lecture Notes in Comput. Sci., Vol. 10719*, Springer-Verlag, Berlin, 2017, pp. 355–373.

[40] CABARCAS, D.—SMITH-TONE, D.—VERBEL, J. A.: *Key recovery attack for ZHFE*. In: *PQCrypto 2017*, *Lecture Notes in Comput. Sci., Vol. 10346*, Springer-Verlag, Berlin, 2017, pp. 289–308.

[41] DING, J.—PETZOLDT, A.: *Current state of multivariate cryptography*, IEEE Security and Privacy, **15** (2017), 28–36. (DOI: 10.1109/MSP.2017.3151328)

Appendix A. A toy example

We exhibit the public key generation and encryption in the proposed cryptosystem with a simple example. We consider the finite fields $\mathbb{F}_{2^4}$ and $\mathbb{F}_{2^8}$ with some fixed normal basis on each of them. Thus, we take $k = 2,\ m = 2^k = 4$.

Public key generation. We take

$$\alpha_1 = \alpha_2 = \alpha_6 = (1,0,0,0), \quad \alpha_3 = (1,1,1,0),$$
$$\alpha_4 = (0,1,1,1), \quad \alpha_5 = (1,1,0,1) \text{ in } \mathbb{F}_{2^4},$$

so that

$$L_{\alpha_1},\ L_{\alpha_2} \quad \text{and} \quad L_{\alpha_6}$$

are the identity polynomial,

$L_{\alpha_3}(x) = x+x^2+x^4$, $L_{\alpha_4}(x) = x^2+x^4+x^8$ and $L_{\alpha_5}(x) = x+x^2+x^8$ in $\mathfrak{L}(2,4)$.

Again, we take $\beta_1 = \beta_2 = (1,0,0,0,0,0,0,0)$ in $\mathbb{F}_{2^8}$, so that L_{β_1} and L_{β_2} are the identity polynomial in $\mathfrak{L}(2,8)$. We take permutations $\pi_i,\ 1 \le i \le 5$, of degree 4, where

$$\pi_1 = \begin{pmatrix} 0 & 1 & 2 & 3 \\ 2 & 0 & 3 & 1 \end{pmatrix}, \quad \pi_2 = \begin{pmatrix} 0 & 1 & 2 & 3 \\ 3 & 2 & 1 & 0 \end{pmatrix}$$

and π_3, π_4, π_5 and π_6 are the identity permutation. Similarly, we take permutations $\eta_i,\ i = 1, 2$, of degree 8, where

$$\eta_1 = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 3 & 1 & 5 & 0 & 4 & 2 & 6 & 7 \end{pmatrix}, \quad \eta_2 = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 5 & 3 & 7 & 2 & 1 & 0 & 4 & 6 \end{pmatrix}.$$

We take

$$\sigma_i = (0,0,0,0) \in \mathbb{F}_{2^4} \quad \text{for } 1 \le i \le 5$$

and

$$\delta_i = (0,0,0,0,0,0,0,0) \in \mathbb{F}_{2^8} \quad \text{for } i = 1, 2.$$

Finally, we take

$$\gamma_1 = (0,0,0,0,0,0,0,0), \quad \gamma_2 = (1,0,0,0,0,0,0,0) \text{ in } \mathbb{F}_{2^8}.$$

With the above inputs we have: $T_1 = \pi_1,\ T_2 = \pi_2, T_3 = L_{\alpha_3},\ T_4 = L_{\alpha_4},\ T_5 = L_{\alpha_5},\ T_6 =$ identity map and $S_1 = \eta_1,\ S_2 = \eta_2$.

Suppose $X = (x_0, x_1, x_2, x_3)$ denotes the plaintext variables. Then

$$X^{(1)} = T_1(X) = (x_2, x_0, x_3, x_1), \quad X^{(2)} = T_2(X) = (x_3, x_2, x_0, x_1).$$

Therefore,

$$X^{(3)} = T_3\left((X^{(1)})^2 * X^{(2)}\right) = (f_0, f_1, f_2, f_3),$$

where

$$\begin{aligned} f_0 &= x_3x_2 + x_0x_1 + x_3 + x_1, \\ f_1 &= 1 + x_0x_3 + x_1x_2, \\ f_2 &= 1 + x_0x_3 + x_0x_1, \\ f_3 &= x_0 + x_2 + x_3x_2 + x_1x_2. \end{aligned}$$

Similarly,

$$X^{(4)} = T_4\left(X^{(1)} * X^{(2)}\right) + T_5\left(\left(X^{(1)}\right)^2 * X^{(2)}\right) = (f_4, f_5, f_6, f_7),$$

where

$$\begin{aligned} f_4 &= x_0 + x_0x_3 + x_2x_3 + x_1x_3, \\ f_5 &= 1 + x_0 + x_2x_3 + x_0x_3, \\ f_6 &= x_3 + x_0x_2 + x_1x_2 + x_0x_1, \\ f_7 &= 1 + x_0x_1 + x_0x_2 + x_1x_3 + x_2x_3. \end{aligned}$$

Suppose $Y = (y_0, y_1, y_2, y_3, y_4, y_5, y_6, y_7)$ denotes the ciphertext variables. Then

$$Z = S_1(Y) = (y_3, y_1, y_5, y_0, y_4, y_2, y_6, y_7).$$

Since $\gamma_1 = 0, \gamma_2 = (1,0,0,0,0,0,0,0)$, the relation (10) of Section 3 becomes $S_2(f_0, f_1, \ldots, f_7) * Z + (1,0,0,0,0,0,0,0) = 0$. Now taking the coordinate representation, the public key becomes the system of the following eight equations:

$$\begin{aligned}
P_0 &= 1 + y_0(x_0x_1 + x_1x_2) + y_1(x_0x_2) + y_2(x_0x_3 + x_2x_3) + y_3(x_2x_3 + x_0x_3) \\
&\quad + y_4(x_0x_3 + x_1x_3) + y_5(x_2x_3 + x_0x_3 + x_1x_3) + y_6(x_1x_2 + x_2x_3 + x_1x_3) \\
&\quad + y_7(x_0x_1 + x_1x_3) + x_0y_3 + x_1y_3 + x_0y_7 + x_2y_7 + x_2y_6 + x_0y_6 + y_2 + y_4 \\
&\quad + x_1y_0 + x_3y_0 + x_0y_5 + x_1y_1, \\
P_1 &= y_0(x_2x_3 + x_0x_3 + x_1x_3) + y_1(x_0x_3 + x_2x_3) + y_2(x_0x_3 + x_1x_3) \\
&\quad + y_3(x_0x_1 + x_1x_2) + y_4(x_0x_1 + x_1x_2) + y_5(x_0x_2) + y_6(x_0x_3 + x_2x_3) \\
&\quad + y_7(x_2x_3 + x_1x_2 + x_0x_2 + x_1x_3) + x_0y_0 + x_0y_1 + x_0y_3 + x_0y_7 + x_1y_1 \\
&\quad + x_1y_4 + x_1y_5 + x_2y_3 + x_2y_7 + x_3y_4 + y_2 + y_0, \\
P_2 &= y_0(x_0x_2) + y_1(x_0x_1 + x_1x_2) + y_2(x_0x_1 + x_1x_2) + y_3(x_2x_3 + x_1x_3 + x_0x_2 \\
&\quad + x_1x_2) + y_4(x_0x_3 + x_2x_3 + x_1x_3) + y_5(x_2x_3 + x_0x_3) + y_6(x_0x_3 + x_1x_3) \\
&\quad + y_7(x_0x_3 + x_2x_3) + x_0y_1 + x_0y_3 + x_0y_4 + x_0y_5 + x_1y_0 + x_1y_2 + x_1y_5 \\
&\quad + x_2y_1 + x_2y_3 + x_3y_2 + y_6 + y_7, \\
P_3 &= y_0(x_0x_3 + x_2x_3) + y_1(x_0x_2 + x_1x_2 + x_2x_3) + y_2(x_0x_3 + x_2x_3 + x_1x_3) \\
&\quad + y_3(x_2x_3 + x_0x_3) + y_4(x_0x_2) + y_5(x_0x_1 + x_1x_2) + y_6(x_0x_1 + x_1x_2) \\
&\quad + y_7(x_0x_3 + x_1x_3) + x_0y_0 + x_0y_1 + x_0y_2 + x_0y_5 + x_1y_0 + x_1y_4 + x_1y_6 \\
&\quad + x_2y_1 + x_2y_5 + x_3y_6 + y_3 + y_7, \\
P_4 &= y_0(x_0x_1 + x_1x_2) + y_1(x_2x_3 + x_0x_3) + y_2(x_0x_2) + y_3(x_1x_3 + x_0x_3) \\
&\quad + y_4(x_0x_3 + x_2x_3) + y_5(x_2x_3 + x_0x_2 + x_1x_3 + x_1x_2) + y_6(x_1x_3 + x_2x_3 \\
&\quad + x_0x_3) + y_7(x_0x_3 + x_2x_3) + x_0y_0 + x_0y_4 + x_0y_5 + x_1y_4 + x_1y_2 + x_1y_7 \\
&\quad + x_2y_0 + x_0y_6 + x_2y_5 + x_3y_7 + y_1 + y_3, \\
P_5 &= y_0(x_0x_2 + x_1x_2 + x_1x_3 + x_2x_3) + y_1(x_0x_3 + x_1x_3) + y_2(x_0x_3 + x_2x_3) \\
&\quad + y_3(x_0x_1 + x_1x_2) + y_4(x_0x_1 + x_1x_2) + y_5(x_2x_3 + x_0x_3) + y_6(x_0x_2) \\
&\quad + y_7(x_2x_3 + x_0x_3 + x_1x_3) + x_0y_0 + x_0y_2 + x_0y_4 + x_0y_7 + x_1y_2 + x_1y_3 \\
&\quad + x_1y_6 + x_2y_0 + x_2y_4 + x_3y_3 + y_1 + y_5, \\
P_6 &= y_0(x_0x_3 + x_2x_3) + y_1(x_0x_1 + x_1x_2) + y_2(x_0x_1 + x_1x_2) + y_3(x_2x_3 + x_0x_3 \\
&\quad + x_1x_3) + y_4(x_0x_2 + x_1x_3 + x_2x_3 + x_1x_2) + y_5(x_0x_3 + x_1x_3) \\
&\quad + y_6(x_0x_3 + x_2x_3) + y_7(x_0x_2) + x_0y_2 + x_0y_3 + x_0y_4 + x_0y_6 + x_1y_1 \\
&\quad + x_1y_6 + x_1y_7 + x_2y_2 + x_2y_4 + x_3y_1 + y_0 + y_5, \\
P_7 &= y_0(x_0x_3 + x_1x_3) + y_1(x_0x_3 + x_1x_3 + x_2x_3) + y_2(x_0x_2 + x_2x_3 + x_1x_2 \\
&\quad + x_1x_3) + y_3(x_0x_2) + y_4(x_2x_3 + x_0x_3 + x_0x_2) + y_5(x_1x_2 + x_0x_1) \\
&\quad + y_6(x_0x_1 + x_1x_2) + y_7(x_2x_3 + x_0x_3) + x_0y_1 + x_0y_2 + x_0y_6 + x_0y_7 + x_1y_3 \\
&\quad + x_1y_5 + x_1y_7 + x_2y_2 + x_2y_6 + x_3y_5 + y_0 + y_4.
\end{aligned}$$

The size of the public key can be reduced by writing it as two sets of equations containing only quadratic terms as follows:

$$\begin{aligned} g_0 &= x_0x_2, & g_1 &= x_0x_1 + x_1x_2, \\ g_2 &= x_2x_3 + x_1x_2 + x_0x_2 + x_1x_3, & g_3 &= x_2x_3 + x_0x_3 + x_1x_3, \\ g_4 &= x_2x_3 + x_0x_3, & g_5 &= x_0x_3 + x_1x_3, \\ g_6 &= x_2x_3 + x_1x_2 + x_0x_2, & g_7 &= x_2x_3 + x_0x_3 + x_0x_2, \\ b &= g_0 + g_2; \end{aligned}$$

and

$$\begin{aligned} P'_0 &= 1 + y_0g_1 + y_1g_0 + y_2g_4 + y_3g_4 + y_4g_5 + y_5g_3 + y_6b + y_7g_5 + x_0y_3 + x_1y_3 \\ &\quad + x_0y_7 + x_2y_7 + x_2y_6 + x_0y_6 + y_2 + y_4 + x_1y_0 + x_3y_0 + x_0y_5 + x_1y_1, \\ P'_1 &= y_0g_3 + y_1g_4 + y_2g_5 + y_3g_1 + y_4g_1 + y_5g_0 + y_6g_4 + y_7g_2 + x_0y_0 + x_0y_1 \\ &\quad + x_0y_3 + x_0y_7 + x_1y_1 + x_1y_4 + x_1y_5 + x_2y_3 + x_2y_7 + x_3y_4 + y_2 + y_6, \\ P'_2 &= y_0g_0 + y_1g_1 + y_2g_1 + y_3g_2 + y_4g_3 + y_5g_4 + y_6g_5 + y_7g_4 + x_0y_1 + x_0y_3 \\ &\quad + x_0y_4 + x_0y_5 + x_1y_0 + x_1y_2 + x_1y_5 + x_2y_1 + x_2y_3 + x_3y_2 + y_6 + y_7, \\ P'_3 &= y_0g_4 + y_1g_6 + y_2g_3 + y_3g_4 + y_4g_0 + y_5g_1 + y_6g_1 + y_7g_5 + x_0y_0 + x_0y_1 \\ &\quad + x_0y_2 + x_0y_5 + x_1y_0 + x_1y_4 + x_1y_6 + x_2y_1 + x_2y_5 + x_3y_6 + y_3 + y_7, \\ P'_4 &= y_0g_1 + y_1g_4 + y_2g_0 + y_3g_5 + y_4g_4 + y_5g_2 + y_6g_3 + y_7g_4 + x_0y_0 + x_0y_4 \\ &\quad + x_0y_5 + x_0y_6 + x_1y_4 + x_1y_2 + x_1y_7 + x_2y_0 + x_2y_5 + x_3y_7 + y_1 + y_3, \\ P'_5 &= y_0g_2 + y_1g_5 + y_2g_4 + y_3g_1 + y_4g_1 + y_5g_4 + y_6g_0 + y_7g_3 + x_0y_0 + x_0y_2 \\ &\quad + x_0y_4 + x_0y_7 + x_1y_2 + x_1y_3 + x_1y_6 + x_2y_0 + x_2y_4 + x_3y_3 + y_1 + y_5, \\ P'_6 &= y_0g_4 + y_1g_1 + y_2g_1 + y_3g_3 + y_4g_2 + y_5g_5 + y_6g_4 + y_7a_0 + x_0y_2 + x_0y_3 \\ &\quad + x_0y_4 + x_0y_6 + x_1y_1 + x_1y_6 + x_1y_7 + x_2y_2 + x_2y_4 + x_3y_1 + y_0 + y_5, \\ P'_7 &= y_0g_5 + y_1g_3 + y_2g_2 + y_3g_0 + y_4g_7 + y_5g_1 + y_6g_1 + y_7g_4 + x_0y_1 + x_0y_2 \\ &\quad + x_0y_6 + x_0y_7 + x_1y_3 + x_1y_5 + x_1y_7 + x_2y_2 + x_2y_6 + x_3y_5 + y_0 + y_4. \end{aligned}$$

Suppose $M = (0, 0, 0, 1)$ is the plaintext message. Substituting this in above public equations we get linear equations,

$$\begin{gathered} y_2 + y_4 + y_0 = 1, \quad y_2 + y_4 + y_6 = 0, \quad y_2 + y_6 + y_7 = 0, \\ y_3 + y_6 + y_7 = 0, \quad y_1 + y_3 + y_7 = 0, \quad y_1 + y_3 + y_5 = 0, \\ y_0 + y_1 + y_5 = 0, \quad y_0 + y_4 + y_5 = 0. \end{gathered}$$

Solving them by Gaussian elimination we get

$$(y_0, y_1, y_2, y_3, y_4, y_5, y_6, y_7) = (0, 1, 0, 0, 1, 1, 1, 1)$$

which is the required ciphertext.

Received July 15, 2019

Rajesh P. Singh (Corresponding author)
Department of Mathematics
Central University of South Bihar
SH-7, Gaya Panchanpur Road
Village- Karhara, Post- Fatehpur
Gaya-824236 (Bihar)
INDIA
E-mail: rpsingh@cub.ac.in

Bhaba Kumar Sarma
Anupam Saikia
Department of Mathematics
Indian Institute of Technology Guwahati
Guwahati-781039
INDIA
E-mail: bks@iitg.ac.in
a.saikia@iitg.ac.in