IEICE TRANS. FUNDAMENTALS, VOL.E102–A, NO.9 SEPTEMBER 2019

PAPER *Special Section on Discrete Mathematics and Its Applications*

The Secure Parameters and Efficient Decryption Algorithm for Multivariate Public Key Cryptosystem EFC*

Yacheng WANG[†a)], ***Nonmember***, **Yasuhiko IKEMATSU**[†b)], ***Member***, **Dung Hoang DUONG**[††c)], ***Nonmember***, ***and*** **Tsuyoshi TAKAGI**[†d)], ***Member***

SUMMARY At PQCrypto 2016, Szepieniec et al. proposed a new type of trapdoor called Extension Field Cancellation (EFC) for constructing secure multivariate encryption cryptosystems. They also specifically suggested two schemes EFC^-_p and $\mathrm{EFC}^-_{pt^2}$ that apply this trapdoor and some modifiers. Although both of them seem to avoid all attacks used for cryptanalysis on multivariate cryptography, their decryption efficiency has room for improvement. On the other hand, their security was analyzed mainly through an algebraic attack of computing the Gröbner basis of the public key, and there possibly exists more effective attacks. In this paper, we introduce a more efficient decryption approach for EFC^-_p and $\mathrm{EFC}^-_{pt^2}$, which manages to avoid all redundant computation involved in the original decryption algorithms without altering their public key. In addition, we estimate the secure parameters for EFC^-_p and $\mathrm{EFC}^-_{pt^2}$ through a hybrid attack of algebraic attack and exhaustive search.

key words: *multivariate cryptography, extension field cancellation, decryption algorithm, hybrid attack*

1. Introduction

Ever since Shor [20] introduced a polynomial-time algorithm in 1994 for solving the integer factorization problem and the discrete logarithm problem on quantum computers, on which currently used public key cryptosystems such as RSA and ECC are based, cryptology community has been researching on finding alternative cryptosystems that are quantum resistant (Post-quantum cryptography). Specially, the National Institute of Standards and Technology (NIST) in the United States is calling for post-quantum cryptosystems (PQC) proposals to be standardized, the National Security Agency (NSA) also announced their plan for switching to quantum resistant public key cryptosystems in the future.

Multivariate cryptography is considered as one of the main candidates for post-quantum cryptography because the security of multivariate cryptography is based on the hardness of the problem of solving a set of multivariate quadratic polynomials, which was proven to be NP-complete, and multivariate cryptography is in general very efficient and requires very modest computational resources. In more than 30 years of research on multivariate cryptography, results on constructing signature schemes seem to be more fruitful, UOV [15] and Rainbow [7] remains secure after many years of attack attempts. On the other hand, history on multivariate encryption is more turbulent. Many multivariate encryption schemes have been proposed, such as MI [16], HFE [18], ABC [22], ZHFE [19], SRP [24], EFC [21], HFERP [13] and EFLASH [3]. Most of them were proven to be insecure under various attacks, such as MinRank [12], HighRank [4], Linearization [17]. Nevertheless, ABC, EFC, HFERP and EFLASH still remain secure. At PQCrypto 2016, Szepieniec et al. [21] proposed multivariate encryption schemes EFC^-_p and $\mathrm{EFC}^-_{pt^2}$, which use matrix multiplications as in ABC [22], and extension field structure as in MI [16], HFE [18] and ZHFE [19].

In this paper, we introduce a more efficient decryption approach for EFC^-_p and $\mathrm{EFC}^-_{pt^2}$. The decryption algorithms for EFC^-_p and $\mathrm{EFC}^-_{pt^2}$ rely on the bilinear relation between the plaintext and an augmented ciphertext, that is the concatenation of a ciphertext and the values of the deleted polynomials by the minus modifier. This bilinear relation is used for constructing linear systems in the decryption process of EFC^-_p and $\mathrm{EFC}^-_{pt^2}$. Our proposed decryption algorithms aim to separate the computation of constructing the linear system into two kinds of computations. One is the computation involving the plaintext and the ciphertext. The other one is computation involving the plaintext and the guessed values. In addition, we experimentally investigate the security of EFC^-_p and $\mathrm{EFC}^-_{pt^2}$ through hybrid attack [1], which is a combination of algebraic attack and exhaustive search.

This paper is structured as follows. In Sect. 2, we recall multivariate cryptography. In Sect. 3, we recall the construction of EFC^-_p and $\mathrm{EFC}^-_{pt^2}$, and their decryption algorithms in [21]. In Sect. 4, we introduce our proposed new decryption algorithms for EFC^-_p and $\mathrm{EFC}^-_{pt^2}$. In Sect. 5, we apply hybrid algebraic attack on EFC^-_p and $\mathrm{EFC}^-_{pt^2}$ and estimate new secure parameters for them. Finally, we conclude the paper in Sect. 6.

2. Multivariate Cryptography

In this section, we give a short introduction to multivariate

Manuscript received September 25, 2018.
Manuscript revised January 25, 2019.

[†]The authors are with the Graduate School of Information Science and Technology, The University of Tokyo, Tokyo, 113-8656 Japan.
[††]The author is with the School of Computing and Information Technology, University of Wollongong, NSW 2522, Australia.

*The preliminary version of this paper was presented at 23rd Australasian Conference on Information Security and Privacy [23].

a) E-mail: yacheng_wang@mist.i.u-tokyo.ac.jp
b) E-mail: ikematsu@mist.i.u-tokyo.ac.jp
c) E-mail: hduong@uow.edu.au
d) E-mail: takagi@mist.i.u-tokyo.ac.jp

DOI: 10.1587/transfun.E102.A.1028

cryptography. Let $\mathbb{F}$ denote a finite field with q elements, n, m be two positive integers and denote the polynomial ring in variables $x_1, \ldots, x_n$ over $\mathbb{F}$ by $\mathbb{F}[x_1, \ldots, x_n]$.

2.1 Quadratic Maps and MQ Problem

In this subsection, we introduce the notion of quadratic maps and MQ-problem.

Given quadratic polynomials $f_1, \ldots, f_m \in \mathbb{F}[x_1, \ldots, x_n]$,

$$f_k = \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}x_i x_j + \sum_{i=1}^{n} b_i x_i + c, \ (1 \le k \le m)$$

where $a_{ij}, b_i, c \in \mathbb{F}$, then $F(x_1, \ldots, x_n) = (f_1, \ldots, f_m) : \mathbb{F}^n \to \mathbb{F}^m$ is called a *quadratic map*.

Multivariate cryptography refers to the study of public key cryptosystems whose public keys are quadratic maps. The security of multivariate cryptography is based on *MQ problem*, that is, given m quadratic polynomials $p_1, \ldots, p_m \in \mathbb{F}[x_1, \ldots, x_n]$, find $\mathbf{z} \in \mathbb{F}^n$ such that $p_1(\mathbf{z}) = \cdots = p_m(\mathbf{z}) = 0$. MQ problem is proven to be NP-complete even for the simplest case of multivariate quadratic polynomials over GF(2). Therefore, multivariate cryptography is considered to be a candidate for post-quantum cryptography.

2.2 Construction of Multivariate Encryption Schemes

To construct a multivariate encryption scheme, we need to design its private key, public key, encryption and decryption processes.

– **Private Key**

We start with choosing an easy-to-invert quadratic map $F(x_1, \ldots, x_n) = (f_1, \ldots, f_m)$. "Easy-to-invert" means given the value $(y_1, \ldots, y_m) \in \mathbb{F}^m$, solving the system $f_i = y_i \ (1 \le i \le m)$ is easy. Such map F is also called a *central map*. Then we choose two invertible linear maps $S : \mathbb{F}^n \to \mathbb{F}^n$ and $T : \mathbb{F}^m \to \mathbb{F}^m$. The set $\{F, S, T\}$ is a private key.

– **Public Key**

Given a private key $\{F, S, T\}$, a public key P can be generated from the composition of F, S, T, i.e. $P = T \circ F \circ S$.

– **General Workflow**

Given a public key P and a plaintext $\mathbf{z} \in \mathbb{F}^n$, the ciphertext $\mathbf{c} \in \mathbb{F}^m$ of $\mathbf{z}$ can be obtained by performing $\mathbf{c} = P(\mathbf{z})$.

Conversely, given a private key $\{F, S, T\}$ and a ciphertext $\mathbf{c} \in \mathbb{F}^m$, a multivariate encryption scheme decrypts $\mathbf{c}$ by computing the inverse of T, F and S individually.

2.3 Algebraic Attack

Algebraic attack directly solves the system:

$$p_1(x_1, \ldots, x_n) = c_1, \cdots, p_m(x_1, \ldots, x_n) = c_m, \qquad (1)$$

where p_i's are public key polynomials, and $\mathbf{c} = (c_1, \ldots, c_m)$ is a ciphertext. Usually, Gröbner basis method is used in the solving process.

Gröbner basis method was introduced by Buchberger, who proposed Buchberger algorithm [2], and was later improved by Faugere with F4/F5 algorithms, see [10], [11]. "Field equations" $x_1^q - x_1 = 0, \cdots, x_n^q - x_n = 0$ are added to the polynomial system (Eq. (1)) to solve this system in $\mathbb{F}$, and it is much faster to compute a Gröbner basis when field equations are added. In multivariate cryptography, let

$$G = \{p_1 - c_1, \ldots, p_m - c_m, x_1^q - x_1, \ldots, x_n^q - x_n\}, \quad (2)$$

we then need to compute the Gröbner basis of the ideal $\mathcal{I}$ generated by G. The complexity of the Gröbner basis method is determined by a so-called *degree of regularity*. There are several different definitions for degree of regularity, through out this paper, we regard the degree of regularity as the degree where a non-trivial syzygy producing a degree fall first occurs, which is also called the *first fall degree* [14]. If F4 or F5 algorithm is used, the complexity of computing the Gröbner basis of $\mathcal{I}$ is

$$\text{Complexity}_{F4/F5} = O\left(\binom{n + d_{reg} - 1}{d_{reg}}^2 \binom{n}{2}\right), \qquad (3)$$

where d_{reg} is the degree of regularity of $\mathcal{I}$.

In [1], Bettale et al. proposed a hybrid algebraic method for solving multivariate systems over finite fields, which can speed up the computation of Gröbner basis with F4/F5 algorithms. Hybrid algebraic method is a combination of algebraic attack and exhaustive search. Specifically, given $e \in \mathbb{N}$, hybrid algebraic method first guesses the value of variables $x_1, \ldots, x_e$, evaluate them in G with the guessing values, and then apply algebraic attack on the remaining polynomial system. The guessing processes terminates when a solution is found.

3. Extension Field Cancellation (EFC)

In this section, we recall the constructions of EFC_p^- and $\text{EFC}_{pt^2}^-$ [21], and the original decryption algorithms designed for them.

3.1 Notations

Let $\mathbb{F}$ be a finite field of 2 elements. Given a positive integer n, $x_1, \ldots, x_n$ are n variables over $\mathbb{F}$, and define $\mathbf{x} = (x_1, \ldots, x_n)$. $\mathbb{E}$ denotes a degree n extension field of $\mathbb{F}$. Denote the set of all $n \times m$ matrices by $\mathbb{F}^{n\times m}$. Matrices are denoted by capital letters, vectors are denoted by bold lowercase letters, and all vectors are treated as row vectors. The i-th entry of a vector $\mathbf{v}$ is denoted by v_i, the i-th row of a matrix M is denoted by M_i. For $N, M \in \mathbb{F}^{n\times n}$, $(N||M) \in \mathbb{F}^{n\times 2n}$ denotes the horizontal join of N and M.

Choose a basis $\{\theta_1, \ldots, \theta_n\}$ of $\mathbb{E}/\mathbb{F}$, and define the isomorphism $\varphi : \mathbb{F}^n \ni \mathbf{v} \mapsto \mathbf{v}\mathbf{b}^\top \in \mathbb{E}$, where $\mathbf{b} = (\theta_1, \ldots, \theta_n) \in \mathbb{E}^n$. For $A \in \mathbb{F}^{n\times n}$, and $\mathbf{v} = (v_1, \ldots, v_n) \in \mathbb{F}^n$, define $\alpha(\mathbf{v}) = \varphi(\mathbf{v}A) \in \mathbb{E}$. The matrix associated with the linear map $\mathbb{E} \ni X \mapsto \alpha(\mathbf{v})X \in \mathbb{E}$ is denoted by $\alpha_m(\mathbf{v}) \in \mathbb{F}^{n\times n}$.

For a matrix $B \in \mathbb{F}^{n\times n}$ and $\mathbf{v} \in \mathbb{F}^n$, we define $\beta(\mathbf{v})$ and $\beta_m(\mathbf{v})$ in the same way as $\alpha(\mathbf{v})$ and $\alpha_m(\mathbf{v})$. For a positive integer a, π_a stands for the following projection:

$$\pi_a : \mathbb{F}^{2n} \ni (v_1, \cdots, v_{2n}) \mapsto (v_1, \cdots, v_{2n-a}) \in \mathbb{F}^{2n-a}.$$

3.2 Construction of the EFC$_p^-$ Scheme

– Key Generation

Given a prime number n, randomly choose $A, B \in \mathbb{F}^{n\times n}$ of rank $n-1$ such that the intersection of the kernel spaces of A and B is the zero subspace. Randomly choose two invertible linear maps $S : \mathbb{F}^n \to \mathbb{F}^n$ and $T : \mathbb{F}^{2n} \to \mathbb{F}^{2n}$, we identify these linear maps with matrices $S \in \mathbb{F}^{n\times n}$, $T \in \mathbb{F}^{2n\times 2n}$. The central map F for EFC$_p^-$ is

$$F : \mathbb{F}^n \ni \mathbf{x} \mapsto (\mathbf{x}\cdot\alpha_m(\mathbf{x}),\ \mathbf{x}\cdot\beta_m(\mathbf{x})) \in \mathbb{F}^{2n}.$$

The public key for EFC$_p^-$ is given by

$$P = (p_1, \cdots, p_{2n-a}) = \pi_a \circ T \circ F \circ S : \mathbb{F}^n \to \mathbb{F}^{2n-a},$$

where p_i $(1 \le i \le 2n-a)$ are quadratic polynomials in $x_1, \ldots, x_n$ over $\mathbb{F}$.

Next we take a look at the explicit form of the central map F. For any $\mathbf{x} \in \mathbb{F}^n$, $\alpha(\mathbf{x}) \in \mathbb{E}$ can be represented with basis $\{\theta_1, \ldots, \theta_n\}$, i.e. $\alpha(\mathbf{x}) = \mathbf{x}A\mathbf{b}^\top$. Let $\alpha_i = A_i\mathbf{b}^\top \in \mathbb{E}$ for $1 \le i \le n$, then we have $\alpha(\mathbf{x}) = \sum_{i=1}^n x_i\alpha_i$. Define matrices $C^{(i)} \in \mathbb{F}^{n\times n}$ by $(C^{(i)})_j^\top = \varphi^{-1}(\alpha_i\theta_j)$ for $1 \le i, j \le n$. It is easy to check that $C^{(i)}$ satisfies $\mathbf{b}C^{(i)} = \alpha_i\mathbf{b}$ for $1 \le i \le n$, which indicates $\alpha_m(\mathbf{x}) = \sum_{i=1}^n x_iC^{(i)}$. Similarly, we define matrices $D^{(i)} \in \mathbb{F}^{n\times n}$ for $1 \le i \le n$ and they satisfy $\beta_m(\mathbf{x}) = \sum_{i=1}^n x_iD^{(i)}$. Therefore, the explicit form of F is

$$F : \mathbb{F}^n \ni \mathbf{x} \mapsto \left(\mathbf{x}\cdot\left(\sum_{i=1}^n C^{(i)}x_i\right),\ \mathbf{x}\cdot\left(\sum_{i=1}^n D^{(i)}x_i\right)\right) \in \mathbb{F}^{2n}.$$

– Encryption

Given a public key P and a plaintext $\mathbf{z} \in \mathbb{F}^n$, its ciphertext is $\mathbf{c} = P(\mathbf{z}) \in \mathbb{F}^{2n-a}$.

– Decryption

Given the private key $\{A, B, S, T\}$ and a ciphertext $\mathbf{c} \in \mathbb{F}^{2n-a}$, decryption process is to find the plaintext $\mathbf{z} \in \mathbb{F}^n$ such that $P(\mathbf{z}) = \mathbf{c}$. First, we need to guess the value $\mathbf{v}$ from $\mathbb{F}^a$ for the deleted polynomials by π_a. Second, we compute $\mathbb{F}^n \times \mathbb{F}^n \ni (\mathbf{d}_1, \mathbf{d}_2) = \mathbf{d} = T^{-1}(\mathbf{c}, \mathbf{v})$. Next we invert the map F by solving the linear system

$$\mathbf{d}_2\alpha_m(\mathbf{x}) = \mathbf{d}_1\beta_m(\mathbf{x}), \tag{4}$$

and obtain a solution $\mathbf{h} \in \mathbb{F}^n$. Finally, if $F(\mathbf{h}) = (\mathbf{d}_1, \mathbf{d}_2)$, then we obtain the plaintext by $\mathbf{z} = S^{-1}(\mathbf{h})$. The loop of guessing the value $\mathbf{v}$ from $\mathbb{F}^a$ terminates when the correct plaintext $\mathbf{z}$ is found. The details are shown in Algorithm 1.

Regrading the complexity of this decryption algorithm, we have the following proposition:

Proposition 1. *The number of field operations involved in the decryption algorithm for EFC$_p^-$ is*

$$8n^4+9n^3+\frac{1}{2}n^2-\frac{7}{2}n+2^{(a-1)}\left(\frac{26}{3}n^3+\frac{21}{2}n^2-\frac{31}{6}n\right). \tag{5}$$

Algorithm 1: Decryption algorithm for EFC$_p^-$ [21]

Input : A ciphertext $\mathbf{c} \in \mathbb{F}^{2n-a}$,
The private key $A, B, S \in \mathbb{F}^{n\times n}$ and $T \in \mathbb{F}^{2n\times 2n}$.
Output: The plaintext $\mathbf{z} \in \mathbb{F}^n$.

```
S_inv ← S^{-1}, T_inv ← T^{-1}
Generate α_m(x), β_m(x) and F from A, B
for v ∈ F^a do
    F^n × F^n ∋ (d_1, d_2) = d ← (c, v) · T_inv
    construct a linear system d_2 · α_m(x) − d_2 · β_m(x) = 0
    x = h ← solve d_2 · α_m(x) − d_2 · β_m(x) = 0
    if F(h) = d then
        break
F^n ∋ z ← h · S_inv
Return z.
```

Proof. Let $[+]_\mathbb{F}$ denotes $\mathbb{F}$-addition, and $[\times]_\mathbb{F}$ denotes $\mathbb{F}$-multiplication of $\mathbb{F}$. We recall the complexity of Gaussian Elimination, and multiplication in $\mathbb{E}$. For an input of $n \times m$ $(m \ge n)$ matrix over $\mathbb{F}$, Gaussian Elimination requires $\sum_{i=1}^{n-1}(n-i)(m-i)$ $[+]_\mathbb{F}$ and $\sum_{i=1}^{n-1}(n-i)(m-i)+\sum_{i=1}^{n-1}(n-i)$ $[\times]_\mathbb{F}$. For any $a, b \in \mathbb{E}$, that are represented in basis $\{\theta_1, \ldots, \theta_n\}$, the multiplication $a \cdot b$ requires $(n-1)(2n-1)$ $[+]_\mathbb{F}$ and $2n^2$ $[\times]_\mathbb{F}$.

Now we analyze the complexity based on the Algorithm 1.

In step 1, computing T^{-1} requires $\frac{n(20n^2-12n+1)}{3}$ $[+]_\mathbb{F}$ and $\frac{2n(10n^2-3n-1)}{3}$ $[\times]_\mathbb{F}$, and computing S^{-1} requires $\frac{n(5n^2-6n+1)}{6}$ $[+]_\mathbb{F}$ and $\frac{n(5n^2-3n-2)}{6}$ $[\times]_\mathbb{F}$. In step 2, to obtain $\alpha_m(\mathbf{x})$, we need to compute $\alpha(\mathbf{x}) = \sum_{i=1}^n x_i\alpha_i$, where $\alpha_i = A_i\mathbf{b}^\top$ $(1 \le i \le n)$, and this requires $n(n-1)$ $[+]_\mathbb{F}$ and n^2 $[\times]_\mathbb{F}$. Then we need to compute $\alpha_i\mathbf{b}$ for $1 \le i \le n$, which indicates n^2 $[\times]_\mathbb{E}$, and it requires $n^2(n-1)(2n-1)$ $[+]_\mathbb{F}$ and $2n^4$ $[\times]_\mathbb{F}$. Same complexity holds for obtaining $\beta_m(\mathbf{x})$.

From step 3 to step 8, we enter a loop of size 2^a. In step 4, $(\mathbf{c}, \mathbf{v}) \cdot T_{inv}$ requires $2n(2n-1)$ $[+]_\mathbb{F}$ and $4n^2$ $[\times]_\mathbb{F}$. In step 5, constructing the linear system needs $2n^3 - n^2$ $[+]_\mathbb{F}$ and $2n^3$ $[\times]_\mathbb{F}$. In step 6, solving the linear system with Gaussian Elimination requires $\frac{n(n-1)(2n+5)}{6}$ $[+]_\mathbb{F}$ and $\frac{n(n^2+3n-1)}{3}$ $[\times]_\mathbb{F}$. In step 7, verifying whether $F(\mathbf{h}) = \mathbf{d}$ holds costs $2n(n^2-1)$ $[+]_\mathbb{F}$ and $2n^2(n+1)$ $[\times]_\mathbb{F}$. The loop terminates in step 8 after an average of $2^{(a-1)}$ times. Therefore, the loop costs $2^{(a-1)}(\frac{13}{3}n^3+\frac{7}{2}n^2-\frac{29}{6}n)$ $[+]_\mathbb{F}$ and $2^{(a-1)}(\frac{13}{3}n^3+7n^2-\frac{1}{3}n)$ $[\times]_\mathbb{F}$ in average.

In step 9, computing $\mathbf{h} \cdot S_{inv}$ needs $n(n-1)$ $[+]_\mathbb{F}$ and n^2 $[\times]_\mathbb{F}$.

Since step 1, step 2 and step 9 together costs $4n^4 + \frac{3}{2}n^3 - \frac{5}{2}n$ $[+]_{\mathbb{F}}$ and $4n^4 + \frac{15}{2}n^3 + \frac{1}{2}n^2 - n$ $[\times]_{\mathbb{F}}$, the total cost of this decryption algorithm is Eq. (5). This completes the proof. □

3.3 Construction of the $\text{EFC}^-_{pt^2}$ Scheme

– Key Generation

Choose the secret key A, B and S, T as in EFC^-_p. The central map F for $\text{EFC}^-_{pt^2}$ is

$$F : \mathbb{F}^n \to \mathbb{F}^{2n} : \mathbf{x} \mapsto \left(\mathbf{x}\alpha_m(\mathbf{x}) + \varphi^{-1}(\beta(\mathbf{x})^3),\ \mathbf{x}\beta_m(\mathbf{x}) + \varphi^{-1}(\alpha(\mathbf{x})^3)\right). \tag{6}$$

The public key for $\text{EFC}^-_{pt^2}$ is $P = (p_1, \ldots, p_{2n-a}) = \pi_a \circ T \circ F \circ S : \mathbb{F}^n \to \mathbb{F}^{2n-a}$. The private key consists of A, B and S, T.

We take a look at the explicit structure of (6) using $\mathbf{b} = (\theta_1, \ldots, \theta_n)$. Since $\mathbf{x}\cdot\alpha_m(\mathbf{x})$ and $\mathbf{x}\cdot\beta_m(\mathbf{x})$ can be represented in the same way as in Sect. 3.2, we show the explicit form of $\varphi^{-1}(\alpha(\mathbf{x}))$ and $\varphi^{-1}(\beta(\mathbf{x}))$ here. Let $\Theta = \mathbf{b}^\top \cdot \mathbf{b} \in \mathbb{E}^{n\times n}$ and $\varphi^{-1}(\Theta) = (\Theta_1, \ldots, \Theta_n) \in (\mathbb{F}^{n\times n})^n$. Define a matrix $\Delta \in \mathbb{F}^{n\times n}$ by its i-th row $\Delta_i = \varphi^{-1}(\theta_i^2)$. Then $\alpha(\mathbf{x})^3$ can be represented as

$$\begin{aligned}\alpha(\mathbf{x})^3 &= \alpha(\mathbf{x})^2 \cdot \alpha(\mathbf{x}) = \mathbf{x}A\left(\theta_1^2, \cdots, \theta_n^2\right)^\top \cdot \mathbf{b}(\mathbf{x}A)^\top \\ &= \mathbf{x}A\Delta\Theta(\mathbf{x}A)^\top = \sum_{i=1}^{n} \theta_i \cdot \mathbf{x}A\Delta\Theta_i(\mathbf{x}A)^\top.\end{aligned}$$

$\beta(\mathbf{x})^3$ can be represented in the same way. Therefore,

$$\begin{aligned}\varphi^{-1}(\alpha(\mathbf{x})^3) &= (\mathbf{x}A\Delta\Theta_1(\mathbf{x}A)^\top, \ldots, \mathbf{x}A\Delta\Theta_n(\mathbf{x}A)^\top),\\ \varphi^{-1}(\beta(\mathbf{x})^3) &= (\mathbf{x}B\Delta\Theta_1(\mathbf{x}B)^\top, \ldots, \mathbf{x}B\Delta\Theta_n(\mathbf{x}B)^\top).\end{aligned}$$

– Encryption

Given a public key P and a plaintext $\mathbf{z} \in \mathbb{F}^n$, the ciphertext is $\mathbf{c} = P(\mathbf{z}) \in \mathbb{F}^{2n-a}$.

– Decryption

We take a look at how to invert the central map F. It requires solving the system $F(\mathbf{x}) = \mathbf{d} \in \mathbb{F}^{2n}$, i.e.

$$\mathbf{x}\cdot\alpha_m(\mathbf{x}) + \varphi^{-1}(\beta(\mathbf{x})^3) = \mathbf{d}_1,\ \mathbf{x}\cdot\beta_m(\mathbf{x}) + \varphi^{-1}(\alpha(\mathbf{x})^3) = \mathbf{d}_2, \tag{7}$$

where $\mathbf{d} = (\mathbf{d}_1, \mathbf{d}_2) \in \mathbb{F}^n \times \mathbb{F}^n$. By definition of $\alpha_m(\mathbf{x})$ in Sect. 3.1, the equation $\varphi(\mathbf{x}\cdot\alpha_m(\mathbf{x})) = \varphi(\mathbf{x})\alpha(\mathbf{x})$ holds. Thus (7) is equivalent to

$$\varphi(\mathbf{x})\alpha(\mathbf{x}) + \beta(\mathbf{x})^3 = \varphi(\mathbf{d}_1),\ \varphi(\mathbf{x})\beta(\mathbf{x}) + \alpha(\mathbf{x})^3 = \varphi(\mathbf{d}_2),$$

from which the following system can be constructed:

$$\mathbf{d}_2\alpha_m(\mathbf{x}) - \mathbf{d}_1\beta_m(\mathbf{x}) = \varphi^{-1}(\alpha(\mathbf{x})^4 - \beta(\mathbf{x})^4). \tag{8}$$

Define a matrix $\Lambda \in \mathbb{F}^{n\times n}$ by $\Lambda_i = \varphi^{-1}(\theta_i^4)$ for $1 \le i \le n$, and apply it to (8). Then (8) turns into

$$\mathbf{d}_2\alpha_m(\mathbf{x}) - \mathbf{d}_1\beta_m(\mathbf{x}) = \mathbf{x}(A - B)\Lambda, \tag{9}$$

which is a linear system in $\mathbf{x}$. The rest of the procedures of decryption is similar to that of EFC^-_p, details are shown in Algorithm 2.

Algorithm 2: Decryption algorithm for $\text{EFC}^-_{pt^2}$ [21]

Input : $\mathbf{b} = (\theta_1, \ldots, \theta_n) \in \mathbb{E}^n$, a ciphertext $\mathbf{c} \in \mathbb{F}^{2n-a}$, the private key $A, B, S \in \mathbb{F}^{n\times n}$ and $T \in \mathbb{F}^{2n\times 2n}$.
Output: The plaintext $\mathbf{z} \in \mathbb{F}^n$.

```
S_inv ← S^{-1}, T_inv ← T^{-1}
Define Λ ∈ F^{n×n} by Λ_i = φ^{-1}(θ_i^4)
Generate α_m(x), β_m(x) and F from A, B
for v ∈ F^a do
    F^n × F^n ∋ (d_1, d_2) = d ← (c, v) · T_inv
    construct a linear system d_2 α_m(x) − d_1 β_m(x) = x(A − B)Λ
    x = h ← solve d_2 α_m(x) − d_1 β_m(x) = x(A − B)Λ
    if F(h) = d then
        break
F^n ∋ z ← h · S_inv
Return z.
```

We analyze the complexity of the decryption algorithm for $\text{EFC}^-_{pt^2}$ adopting the same approach as in the proof of Proposition 1, and obtain the number of field operations involved in the decryption algorithm for $\text{EFC}^-_{pt^2}$ as

$$8n^4 + 17n^3 - \frac{11}{2}n^2 - \frac{3}{2}n + 2^{(a-1)}\left(\frac{32}{3}n^3 + \frac{19}{2}n^2 - \frac{31}{6}n\right). \tag{10}$$

4. Our Proposed Efficient Decryption Algorithms for EFC^-_p and $\text{EFC}^-_{pt^2}$

In this section, we introduce our new decryption algorithms for EFC^-_p and $\text{EFC}^-_{pt^2}$.

4.1 New Decryption Algorithm for EFC^-_p

The new decryption algorithm is derived from linearization equations, which represent a relation between the plaintext and ciphertext.

We begin with deriving linearization equations related to the central map of EFC^-_p. Recall that the linear system (4) for inverting its central map is

$$\mathbf{d}_2\alpha_m(\mathbf{x}) - \mathbf{d}_1\beta_m(\mathbf{x}) = 0,$$

which can also be written as

$$\alpha(\mathbf{x})\varphi(\mathbf{d}_2) - \beta(\mathbf{x})\varphi(\mathbf{d}_1) = \mathbf{x}A\mathbf{b}^\top \cdot \mathbf{b}\mathbf{d}_2^\top - \mathbf{x}B\mathbf{b}^\top \cdot \mathbf{b}\mathbf{d}_1^\top = 0.$$

Let $\Theta = \mathbf{b}^\top \cdot \mathbf{b}$ and $(\Theta_1, \ldots, \Theta_n) = \varphi^{-1}(\Theta)$, then from this equation, we can obtain linearization equations corresponding to the central map of EFC^-_p as follows:

$$\mathbf{x}(B\Theta_i||A\Theta_i)\mathbf{d}^\top = 0,\ (1 \le i \le n), \tag{11}$$

where $\mathbf{d} = (\mathbf{d}_1, \mathbf{d}_2)$.

Subsequently, we apply S and T to the linearization equation related to the central map. Let $\mathbf{c} \in \mathbb{F}^{2n}$ be a ciphertext of EFC_p^- without minus modifier, then $\mathbf{c} = T(\mathbf{d})$. Apply the linear maps S and T to Eq. (11), and we obtain the linearization equations between a plaintext $\mathbf{x}$ and $\mathbf{c}$ as

$$\mathbf{x}S(B\Theta_i||A\Theta_i)(\mathbf{c}T^{-1})^\top = 0. \tag{12}$$

For a ciphertext $\mathbf{c}$ of EFC_p^- without minus modifier, its corresponding plaintext can be found by solving Eq. (12).

Next we show how to represent Eq. (12) into one simple equation. Let $N^{(i)} = T^{-1}(SB\Theta_i||SA\Theta_i)^\top \in \mathbb{F}^{2n\times n}$, and define matrices $U^{(j)}$ by $U_i^{(j)} = N_j^{(i)}$ for $1 \le j \le 2n$ and $1 \le i \le n$. Then Eq. (12) turns into one simple equation

$$(c_1U^{(1)} + \cdots + c_{2n}U^{(2n)}) \cdot \mathbf{x}^\top = 0. \tag{13}$$

This equation indicates that as long as we have the set $\Psi = (U^{(1)}, \ldots, U^{(2n)})$, the decryption process of EFC_p^- without minus modifier can be reduced into the computation of the right kernel space of $c_1U^{(1)} + \ldots + c_{2n}U^{(2n)}$.

Since in our new decryption algorithm, only the ciphertext $\mathbf{c}$ and $U^{(1)}, \ldots, U^{(2n)}$ are necessary, we intend to save $\Psi = (U^{(1)}, \ldots, U^{(2n)})$ as the new private key for EFC_p^-, which is $2n/7$ times larger than the original private key.

Now we explain our proposed decryption algorithm for EFC_p^-. First, we compute $L = \sum_{i=1}^{2n-a} c_iU^{(i)}$. Second, we guess the values for the deleted polynomials by π_a from $\mathbb{F}^a$, and denote these values by $\mathbf{v} = (v_1, \cdots, v_a)$. Next, we compute the right kernel space $\mathbf{ker} = \ker(L + \sum_{i=1}^{a} v_iU^{(2n-a+i)})$. Finally, we check if there exists $\mathbf{z} \in \mathbf{ker}$ such that $P(\mathbf{z}) = \mathbf{c}$ holds. If so, then $\mathbf{z}$ is the plaintext, otherwise, go back to the guessing step and start over. The details of the procedures of generating Ψ and the decryption process are shown in Algorithm 3.

Remark 1. *The iterative computation complexity of $\sum_{i=1}^{a} v_iU^{(2n-a+i)}$ can be further reduced. Assume we have $\mathbf{v}^{(1)}, \mathbf{v}^{(2)} \in \mathbb{F}_2^a$, and we need to compute*

$$L^{(1)} = \sum_{i=1}^{a} v_i^{(1)}U^{(2n-a+i)},\ L^{(2)} = \sum_{i=1}^{a} v_i^{(2)}U^{(2n-a+i)}.$$

If we know the difference between $\mathbf{v}^{(1)}$ and $\mathbf{v}^{(2)}$, we will be able to compute $L^{(2)}$ from $L^{(1)}$ by subtracting a few matrices. Since Gray code has the property that two successive binary numeral values differ in only one bit, we use Gray code to represent $\mathbb{F}_2^a$. Assume $\mathbf{v}^{(1)}$ and $\mathbf{v}^{(2)}$ are two successive codes, and they differ at j-th bit, then we can compute $L^{(2)}$ by

$$L^{(2)} = L^{(1)} + U^{(2n-a+j)}.$$

Therefore, this technique reduces the number of operations of matrix addition involved in recursive computation of $\sum_{i=1}^{a} v_iU^{(2n-a+i)}$. We consider this technique when we evaluate the complexity of our new decryption.

Algorithm 3: New decryption algorithm for EFC_p^-

Input : $\mathbf{b} = (\theta_1, \cdots, \theta_n)$, the private key $A, B, S \in \mathbb{F}^{n\times n}$ and $T \in \mathbb{F}^{2n\times 2n}$, a ciphertext $\mathbf{c} \in \mathbb{F}^{2n-a}$.
Output: The plaintext $\mathbf{z} \in \mathbb{F}^n$ s.t. $P(\mathbf{z}) = \mathbf{c}$.

1 $\Theta \leftarrow \mathbf{b}^\top \cdot \mathbf{b}$, $(\Theta_1, \ldots, \Theta_n) \leftarrow \varphi^{-1}(\Theta)$
2 **for** $i \leftarrow 1$ **to** n **do**
3 $N^{(i)} \leftarrow T^{-1}(SB\Theta_i||SA\Theta_i)^\top \in \mathbb{F}^{2n\times n}$
4 **for** $j \leftarrow 1$ **to** $2n$ *and* $i \leftarrow 1$ **to** n **do**
5 $U_i^{(j)} \leftarrow N_j^{(i)}$
6 $L \leftarrow \sum_{i=1}^{2n-a} c_iU^{(i)}$
7 **for** $\mathbf{v} = (v_1, \ldots, v_a) \in \mathbb{F}^a$ **do**
8 $H \leftarrow L + \sum_{i=1}^{a} v_iU^{(2n-a+i)}$
9 $\mathbf{ker} \leftarrow \mathrm{RightKer}(H)$
10 **for** $\mathbf{z} \in \mathbf{ker}$ **do**
11 **if** $P(\mathbf{z}) = \mathbf{c}$ **then**
12 **Return** $\mathbf{z}$
13 break

Regarding the complexity of the new decryption algorithm for EFC_p^-, we have the following proposition.

Proposition 2. *The number of field operations involved in the new decryption for EFC_p^- is*

$$2^{(a-1)}\left(\frac{14}{3}n^3 + (\frac{11}{2} - 2a)n^2 - (a + \frac{19}{6})n + a\right) + 4n^3 - (2a+1)n^2 \tag{14}$$

Proof. Let $[+]_{\mathbb{F}}$ denote $\mathbb{F}$-addition, and $[\times]_{\mathbb{F}}$ denote the $\mathbb{F}$-multiplication. We analyze the complexity based on Algorithm 3. Note that we analyze the complexity starting from step 6 since we save Ψ as the new private key.

In step 6, $\sum_{i=1}^{2n-a} c_iU^{(i)}$ requires $n^2(2n-a-1)$ $[+]_{\mathbb{F}}$ and $n^2(2n-a)$ $[\times]_{\mathbb{F}}$.

From step 7 to 13, we enter a loop of size 2^a. In step 8, $L + \sum_{i=1}^{a} v_iU^{(2n-a+i)}$ costs $2n^2$ $[+]_{\mathbb{F}}$ using Gray code technique in Remark 1. In step 9, finding the right kernel of H requires $\frac{n(n-1)(2n+5)}{6}$ $[+]_{\mathbb{F}}$ and $\frac{n(n^2+3n-1)}{3}$ $[\times]_{\mathbb{F}}$. In step 11, verifying the solution requires $(2n-a)(n^2-1)$ $[+]_{\mathbb{F}}$ and $(2n-a)(n^2+n)$ $[\times]_{\mathbb{F}}$. In step 13, the loop terminates after an average of $2^{(a-1)}$ times. Therefore, the loop requires $2^{(a-1)}(\frac{7}{3}n^3 + (\frac{5}{2} - a)n^2 - \frac{17}{6}n + a)$ $[+]_{\mathbb{F}}$ and $2^{(a-1)}(\frac{7}{3}n^3 + (3-a)n^2 - (a + \frac{1}{3})n)$ $[\times]_{\mathbb{F}}$ in average.

Therefore, the total cost of this decryption algorithm is Eq. (14). This completes the proof. □

4.2 New Decryption Algorithm for $\mathrm{EFC}_{pt^2}^-$

Same as EFC_p^-, the new decryption algorithm for $\mathrm{EFC}_{pt^2}^-$ also derives from linearization equations.

We first consider linearization equations related to the central map of $\mathrm{EFC}_{pt^2}^-$. Recall in Sect. 3.3, inverting the central map of $\mathrm{EFC}_{pt^2}^-$ requires solving the linear system

$$\mathbf{d}_2\alpha_m(\mathbf{x}) - \mathbf{d}_1\beta_m(\mathbf{x}) = \mathbf{x}(A-B)\Lambda, \tag{15}$$

where $\Lambda \in \mathbb{F}^{n\times n}$, $\Lambda_i = \varphi^{-1}(\theta_i^4)$ for $1 \le i \le n$. This equation can also be written as

$$\varphi(\mathbf{d}_2)\alpha(\mathbf{x}) - \varphi(\mathbf{d}_1)\beta(\mathbf{x}) = \varphi(\mathbf{x}(A-B)\Lambda). \tag{16}$$

Applying φ^{-1} on both sides of Eq. (16) gives us

$$\mathbf{x}(B\Theta_i||A\Theta_i)\mathbf{d}^\top - (\mathbf{x}(A-B)\Lambda)_i = 0,\ (1 \le i \le n), \tag{17}$$

which are the linearization equations related to the central map of $\mathrm{EFC}^-_{pt^2}$.

Next we apply S and T to the linearization equations we obtained. Let $\mathbf{c} \in \mathbb{F}^{2n}$ be a ciphertext of $\mathrm{EFC}^-_{pt^2}$ without minus modifier, then $\mathbf{c} = T(\mathbf{d})$. Applying linear maps S and T on (17) gives us the linearization equations of a plaintext $\mathbf{x}$ and a ciphertext $\mathbf{c}$:

$$\mathbf{x}(SB\Theta_i||SA\Theta_i)(\mathbf{c}T^{-1})^\top - (\mathbf{x}S(A-B)\Lambda)_i = 0. \tag{18}$$

Next we show how to represent Eq. (18) into one simple equation. Let $M = S(A-B)\Lambda \in \mathbb{F}^{n\times n}$, $N^{(i)} = T^{-1}(SB\Theta_i||SA\Theta_i)^\top \in \mathbb{F}^{2n\times n}$, and define matrices $U^{(j)}$ by $U_i^{(j)} = N_j^{(i)}$ for $1 \le j \le 2n$ and $1 \le i \le n$. Then (18) can be rearranged into

$$(c_1U^{(1)} + \cdots + c_{2n}U^{(2n)} - M^\top)\cdot\mathbf{x}^\top = 0. \tag{19}$$

This equation indicates that the decryption of $\mathrm{EFC}^-_{pt^2}$ without minus modifier can be reduced to the computation of the right kernel space of $c_1U^{(1)} + \cdots + c_{2n}U^{(2n)} - M^\top$.

Similar to EFC^-_p, we save $\Psi = (U^{(1)}, \ldots, U^{(2n)}, M)$ as the new private key for $\mathrm{EFC}^-_{pt^2}$, which is $(2n+1)/7$ times larger than the original private key. New decryption algorithm for $\mathrm{EFC}^-_{pt^2}$ works similarly to that of EFC^-_p, detailed procedures are shown in Algorithm 4.

We can analyze the complexity of the decryption algorithm for EFC_{pt^2} using the same approach as in the proof of Proposition 2. The number of field operations involved in the new decryption algorithm for $\mathrm{EFC}^-_{pt^2}$ is

$$2^{(a-1)}\left(\frac{14}{3}n^3 + (\frac{11}{2} - 2a)n^2 - (a + \frac{19}{6})n + a\right) + 4n^3 - 2an^2. \tag{20}$$

4.3 Implementation

We verify the effectiveness of our decryption method through implementation operated on a 2.10 GHz Intel® Xero® Gold 6130 CPU with Magma V2.23-10 under originally claimed 80-bit security parameters, see [21], and then compare the results with complexity given in (5), (10), (14) and (20). The implementation results are given in Table 1.

From Table 1, we know, under 80-bit security parameter given in [21], theoretically our new decryption algorithms are 2.24 times faster for EFC^-_p and 3.58 times faster for $\mathrm{EFC}^-_{pt^2}$ than the original decryption algorithms, and the speed-up obtained in our implementation are 8.92 and 9.33 times for EFC^-_p and $\mathrm{EFC}^-_{pt^2}$, respectively.

Since our proposed decryption algorithms do not alter public keys for EFC^-_p and $\mathrm{EFC}^-_{pt^2}$, their security does not change. As for the private key, to match with our proposed decryption algorithms, we use new private keys, which is $2n/7$ times larger for EFC^-_p, and $(2n+1)/7$ times larger for $\mathrm{EFC}^-_{pt^2}$ compared to the original private keys.

Algorithm 4: New decryption algorithm for $\mathrm{EFC}^-_{pt^2}$

Input : $\mathbf{b} = (\theta_1, \cdots, \theta_n)$, the private key $A, B, S \in \mathbb{F}^{n\times n}$ and $T \in \mathbb{F}^{2n\times 2n}$.
Output: The plaintext $\mathbf{z} \in \mathbb{F}^n$ s.t. $P(\mathbf{z}) = \mathbf{c}$.

$\Theta \leftarrow \mathbf{b}^\top \cdot \mathbf{b}$, $(\Theta_1, \ldots, \Theta_n) \leftarrow \varphi^{-1}(\Theta)$
Define $\Lambda \in \mathbb{F}^{n\times n}$, where $\Lambda_i = \varphi^{-1}(\theta_i^4)$
$M \leftarrow S(A-B)\Lambda$
for $i \leftarrow 1$ **to** n **do**
  $N^{(i)} \leftarrow T^{-1}(SB\Theta_i||SA\Theta_i)^\top \in \mathbb{F}^{2n\times n}$
for $j \leftarrow 1$ **to** $2n$ *and* $i \leftarrow 1$ **to** n **do**
  $U_i^{(j)} \leftarrow N_j^{(i)}$
$L \leftarrow \sum_{i=1}^{2n-a} c_i U^{(i)} - M^\top$
for $\mathbf{v} = (v_1, \ldots, v_a) \in \mathbb{F}^a$ **do**
  $H \leftarrow L + \sum_{i=1}^{a} v_i U^{(2n-a+i)}$;
  ker $\leftarrow$ RightKer(H)
  for $\mathbf{z} \in$ **ker do**
    if $P(\mathbf{z}) = \mathbf{c}$ **then**
      Return z
      break

Table 1 Timing comparison between old EFC^-_p, $\mathrm{EFC}^-_{pt^2}$ with new EFC^-_p, $\mathrm{EFC}^-_{pt^2}$ under 80-bit security parameter given in [21], [+, ×] represents the number of involved field operations†.

	Scheme (n, a)	KeyGen.[s]	Enc.[s]	Dec.[s]	$[+, \times]_{\mathbb{F}}$
Old	$\mathrm{EFC}^-_p(83, 10)$	0.022	0.00023	0.116	2.96×10^9
	$\mathrm{EFC}^-_{pt^2}(83, 8)$	0.023	0.00023	0.028	1.18×10^9
New	$\mathrm{EFC}^-_p(83, 10)$	0.022	0.00023	0.013	1.32×10^9
	$\mathrm{EFC}^-_{pt^2}(83, 8)$	0.023	0.00023	0.003	0.33×10^9

5. Hybrid Attack Against EFC^-_p and $\mathrm{EFC}^-_{pt^2}$

Because of the minus modifier, the most efficient attack against EFC^-_p and $\mathrm{EFC}^-_{pt^2}$ is expected to be the algebraic attack, see Sect. 2.3, which computes the Gröbner basis of the ideal generated by the public key and the field equations. Normally we use F4/F5 [10], [11] algorithms, which has complexity given in Eq. (3). In [21], upper bounds for the degree of regularity of EFC^-_p and $\mathrm{EFC}^-_{pt^2}$ are given:

†Note that only 80-bit security level parameters are proposed in [21].

$$d_{reg} \leq \frac{r}{2} + 2, \quad r = \begin{cases} 2 + a, & \text{EFC}_p^-, \\ 4 + a, & \text{EFC}_{pt^2}^-. \end{cases} \tag{21}$$

And [21] also claimed the degree of regularity of EFC_p^- and $\text{EFC}_{pt^2}^-$ lie close to these upper bounds. Under the assumption that these upper bounds are identical to the degree of regularity of EFC_p^- and $\text{EFC}_{pt^2}^-$, 80-bit and 128-bit security parameters are estimated in [21] and [23], respectively.

In this section, we apply hybrid algebraic attack on EFC_p^- and $\text{EFC}_{pt^2}^-$ to learn how parameters n and a affect their degree of regularity. All of our experiments are conducted on a 2.10 GHz Intel® Xero® Gold 6130 Processor with Magma V2.23-10, where F4 algorithm is implemented.

5.1 Notations

Notations used in this section are as follows:

- d_{reg} : degree of regularity (see Sect. 2.3).
- step_{deg} : step degree, a sequence of degrees of polynomials appeared in all the steps of F4 algorithm.
- s_{deg} : solving degree, the degree of the most time-consuming step of F4 algorithm.
- e : the number of variables evaluated in hybrid algebraic attack (see Sect. 2.3).
- time/gb : average time cost of one round of F4 algorithm in hybrid algebraic attack.
- total time (est.) : estimated time cost of hybrid algebraic attack, and it holds the following equation: total time (est.) = $2^{e-1} * (\text{time/gb})$.
- total time : average time cost of hybrid algebraic attack.
- s, h, d, y : second, hour, day, year.

5.2 Hybrid Attack on EFC_p^- and Update Secure Parameters

For EFC_p^-, we first verify if the claimed 80-bit security parameter in [21] indeed has 80-bit security against hybrid attack. Let e be a positive integer. We guess values for variables $(x_1, \ldots, x_e)$ from $\mathbb{F}^e (|\mathbb{F}^e| = 2^e)$, and evaluate the system G, see (2), with those guessing values, then perform algebraic attack on the obtained system. The experiment terminates when a solution can be found. The results are shown in Table 2. From this table, we know when $e = 16$, both of d_{reg} and s_{reg} are 4, and the complexity of the hybrid attack is around $2^{16}\binom{83-16+4-1}{4}^2\binom{83-16}{2} \approx 2^{67}$ by formula (3). Therefore, the originally proposed 80-bit security parameter fails in achieving its claimed security level.

To update secure parameters for EFC_p^-, we need to find a lower bound for the degree of regularity for EFC_p^-. The upper bound given in (21) is given following analysis on HFE based schemes [5], [6], [8], [9], and it is deduced according to the fact that the d_{reg} of a subsystem is equal to or larger than the full polynomial system. Following this approach, we know the d_{reg} of an EFC_p^- system with $a - 1$ polynomial deleted is equal to or smaller than that of an EFC_p^- system with a polynomial deleted. It means a lower bound for d_{reg} of

Table 2 Hybrid algebraic attack on $\text{EFC}_p^-(83, 10)$.

e	step_{deg}	d_{reg}	s_{deg}	time/gb	total time (est.)
15	(2,3,4,4,5,...)	4	≥ 5	–	–
16	(2,3,4,4,4,4)	4	4	2.618d	235.065y
17	(2,3,4,4,4)	4	4	1.896d	340.418y
18	(2,3,4,4,3)	4	4	23.658h	353.983y
19	(2,3,4,4)	4	4	4.900h	146.647y
20	(2,3,4,4)	4	4	4.016h	240.370y
21	(2,3,4,4)	4	4	3.132h	374.864y

Table 3 Behave of d_{reg} of $\text{EFC}_p^-(n = 41, a)$ with $e = 1$.

a	step_{deg}	d_{reg}	s_{deg}	total time
10	(2,3,4,4,5,2,3,4,5,6,7)	4	5	2.181h
11	(2,3,4,5,2,3,4,5,6,7)	5	5	2.550h
12	(2,3,4,5,4,2,3,4,5,6,7)	5	5	2.707h
13	(2,3,4,5,5,2,3,4,6,7)	5	5	6.204h
14	(2,3,4,5,5,2,3,4,5,6,7)	5	5	6.995h
15	(2,3,4,5,5,2,3,4,5,6,7)	5	5	7.911h
16	(2,3,4,5,5,2,3,4,5,6,7)	5	5	8.713h
17	(2,3,4,5,5,2,3,4,5,6,7)	5	5	9.305h
18	(2,3,4,5,6,...)	≥ 6	≥ 6	–

EFC_p^- can be obtained as long as one linear combination of the polynomials deleted by minus modifier can be recovered. However, this indicates the total break of EFC_p^-, and it is a very difficult task. Therefore, we experiment with small parameters to find an experimental lower bound. Parameter n is fixed to be 41, then we experiment with parameter a increasing. The results are shown in Table 3, from which we know there are two turning points for the degree of regularity. One of them is when a rises to 11 from 10, the degree of regularity turns into 5, the other one is when a rises from 17 to 18, the degree of regularity turns into 6. We regard them as lower bounds to update new parameters in Sect. 5.4. Since the d_{reg} of an EFC_p^- system with larger parameter is equal to or larger than the d_{reg} of an EFC_p^- system with smaller parameter, updateding new parameters using those experimental lower bounds should be enough.

5.3 Hybrid Attack on $\text{EFC}_{pt^2}^-$

We apply hybrid attack on $\text{EFC}_{pt^2}^-$ in the same approach as EFC_p^-, the results are shown in Table 4. This table shows hybrid attack cannot work any better than algebraic attack. Therefore, the originally proposed 80-bit security parameter set for $\text{EFC}_{pt^2}^-$ remains secure against hybrid attack. Table 4 also shows the variation of d_{reg} of $\text{EFC}_{pt^2}^-$ is more drastic compared to EFC_p^-, which is also the reason why we were not able to perform hybrid attack with small parameters. Even with $n = 40, a = 8$, which is expected to have $d_{reg} \leq 8$, it takes significantly long time to perform F4 algorithm that we were not able to get the results. We therefore use bound given in (21) to estimate 128-bit security level parameter.

From Table 4, we can see that there is a tendency of hybrid attack not outperforming direct attack for $\text{EFC}_{pt^2}^-$. Because of this, the estimation of secure parameters for $\text{EFC}_{pt^2}^-$

Table 4 Hybrid algebraic attack on $\mathrm{EFC}^-_{pt^2}(83, 8)$.

e	step_{deg}	d_{reg}	s_{deg}	complexity
36	(2,3,4,5,6,...)	≥ 6	≥ 6	$\geq 2^{95}$
37	(2,3,4,5,5)	5	5	2^{89}
44	(2,3,4,5)	5	4	2^{93}
45	(2,3,4,4)	4	4	2^{88}
53	(2,3,4)	4	3	2^{92}
54	(2,3,3)	3	3	2^{87}

Table 5 Updated parameters sets.

	Security	Old parameters	New parameters
EFC^-_p	80-bit	(83, 10)	(241, 11)
	128-bit	(467, 10)	(1523, 18)
$\mathrm{EFC}^-_{pt^2}$	80-bit	(83, 8)	(83, 8)
	128-bit	(467, 8)	(467, 8)

Table 6 Performance comparison between EFC^-_p and $\mathrm{EFC}^-_{pt^2}$ with updated parameters and new decryption algorithms.

	Scheme (n, a)	KeyGen.[s]	Enc.[s]	Dec.[s]
80-bit	$\mathrm{EFC}^-_p(241, 11)$	0.352	0.0013	0.090
	$\mathrm{EFC}^-_{pt^2}(83, 8)$	0.023	0.00023	0.003
128-bit	$\mathrm{EFC}^-_p(1523, 18)$	403.946	0.215	562.435
	$\mathrm{EFC}^-_{pt^2}(467, 8)$	3.734	0.0072	0.065

for high security levels strongly relies on a good theoretical estimation of degree of regularity. Therefore, the tightness of bound given in (21) should be further investigated, and we would like to continue working in this regard in the future.

5.4 Update New Secure Parameters

Taking the experimental results of hybrid algebraic attack on EFC^-_p and $\mathrm{EFC}^-_{pt^2}$ into account, we update their secure parameters, see Table 5. Old 128-bit security parameters are deduced according to (21), and new parameter for EFC^-_p are obtained by considering the experimental lower bound of degree of regularity in Sect. 5.2 and Sect. 5.3.

5.5 Implementation Under New Parameters

New secure parameters for EFC^-_p are updated, we verify its performance by implementation, and compare it with $\mathrm{EFC}^-_{pt^2}$, see Table 6. From this table, we can easily see, $\mathrm{EFC}^-_{pt^2}$ performs better when it comes to efficiency.

6. Conclusion

We have shown that EFC^-_p and $\mathrm{EFC}^-_{pt^2}$ both contain redundant computation in their decryption. By removing them, both of their decryption process can be improved. Based on this idea, we proposed our decryption algorithms for EFC^-_p and $\mathrm{EFC}^-_{pt^2}$ without weakening their security. We also showed originally proposed 80-bit security parameter for EFC^-_p failed in achieving the claimed security level through hybrid algebraic attack. However, the originally proposed 80-bit security parameter for $\mathrm{EFC}^-_{pt^2}$ seemed secure enough. We estimated new secure 80-bit and 128-bit security parameter for EFC^-_p and compared its performance with $\mathrm{EFC}^-_{pt^2}$ under the same security level, and conclude that it is recommended to use $\mathrm{EFC}^-_{pt^2}$ since 128-bit EFC^-_p is very inefficient.

Moreover, a thorough investigation on degree of regularity of EFC^-_p and $\mathrm{EFC}^-_{pt^2}$ is inadequate. With a good estimation of degree of regularity, parameter choosing process will become simpler comparing to our method of using experimental results on hybrid attack. We will continue working on this in the future.

Acknowledgments

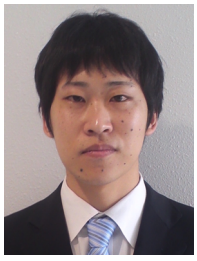
The first author thanks the Japanese Society for the Promotion of Science (JSPS) for financial support under grant KAKENHI 18J20866. The second and fourth authors were supported by JST CREST (Grant Number JPMJCR14D6).

References

[1] L. Bettale, J.-C. Faugère, and L. Perret, "Hybrid approach for solving multivariate systems over finite fields," J. Mathematical Cryptology, vol.3, no.3, pp.177–197, 2009.
[2] B. Buchberger, Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal, Ph.D. thesis, Universitat Innsbruck, 1965.
[3] R. Cartor and D. Smith-Tone, "EFLASH: A new multivariate encryption scheme," ePrint, 2017/1184, 2017.
[4] D. Coppersmith, J. Stern, and S. Vaudenay, "Attacks on the birational permutation signature schemes," Advances in Cryptology - CRYPTO'93, vol.773 of LNCS, pp.435–443, Springer, 1993.
[5] J. Ding and T.J. Hodges, "Inverting HFE system is quasi-polynomial for all fields," Advances in Cryptology – CRYPTO 2011, vol.6841 of LNCS, pp.724–742, Springer, 2011.
[6] J. Ding and T. Kleinjung, "Degree of regularity for HFE-," Cryptology ePrint Archive, Report 2011/570, 2011.
[7] J. Ding and D. Schmidt, "Rainbow, a new multivariate polynomial signature scheme," Applied Cryptography and Network Security – ACNS 2005, vol.3531 of LNCS, pp.164–175, Springer, 2005.
[8] J. Ding and B.-Y. Yang, "Degree of regularity for HFEv and HFEv-," Post-Quantum Cryptography 2013, vol.7932 of LNCS, pp.52–66, Springer, 2013.
[9] V. Dubois and N. Gama, "The degree of regularity of hfe systems," Advances in Cryptology - ASIACRYPT 2010, vol.6477 of LNCS, pp.557–576, Springer, 2010.
[10] J.-C. Faugère, "A new efficient algorithm for computing Gröbner bases (F4)," J. Pure Appl. Algebra, vol.139, no.1, pp.61–88, 1999.
[11] J.C. Faugère, "A new efficient algorithm for computing Gröbner bases without reduction to zero (F5)," ISSAC 2002, pp.75–83, ACM, 2002.
[12] L. Goubin and N.T. Courtois, "Cryptanalysis of the TTM cryptosystem," Advances in Cryptology — ASIACRYPT 2000, vol.1976 of LNCS, pp.44–57, Springer, 2000.
[13] Y. Ikematsu, R. Perlner, D. Smith-Tone, T. Takagi, and J. Vates, "HFERP - A new multivariate encryption scheme," Post-Quantum Cryptography – PQCrypto 2018, vol.10786 of LNCS, pp.396–416, Springer, 2018.
[14] T.J. Hodges, C. Petit, and J. Schlather, "First fall degree and weil descent," Finite Fields and Their Applications, vol.30, pp.155–177, 2014.
[15] A. Kipnis, J. Patarin, and L. Goubin, "Unbalanced oil and vinegar signature schemes," Advances in Cryptology – EUROCRYPT'99,

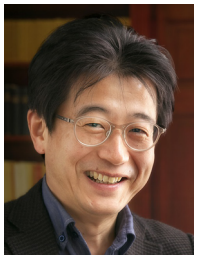
vol.1592 of LNCS, pp.206–222, Springer, 1999.
[16] T. Matsumoto and H. Imai, "Public quadratic polynomial-tuples for efficient signature-verification and message-encryption," Advances in Cryptology – EUROCRYPT'88, vol.330 of LNCS, pp.419–453, Springer, 1988.
[17] J. Patarin, "Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt'88," Advances in Cryptology – CRYPT0'95, vol.963 of LNCS, pp.248–261, Springer, 1995.
[18] J. Patarin, "Hidden fields equations (HFE) and isomorphisms of polynomials (IP): Two new families of asymmetric algorithms," Advances in Cryptology – EUROCRYPT'96, vol.1070 of LNCS, pp.33–48, Springer, 1996.
[19] J. Porras, J. Baena, and J. Ding, "ZHFE, a new multivariate public key encryption scheme," Post-Quantum Cryptography 2014, vol.8772 of LNCS, pp.229–245, Springer, 2014.
[20] P. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," SIAM J, Comput., vol.26, no.5, pp.1484–1509, 1997.
[21] A. Szepieniec, J. Ding, and B. Preneel, "Extension field cancellation: A new central trapdoor for multivariate quadratic systems," Post-Quantum Cryptography 2016, vol.9606 of LNCS, pp.182–196, Springer, 2016.
[22] C. Tao, A. Diene, S. Tang, and J. Ding, "Simple matrix scheme for encryption," Post-Quantum Cryptography 2013, vol.7932 of LNCS, pp.231–242, Springer, 2013.
[23] Y. Wang, Y. Ikematsu, D.H. Duong, and T. Takagi, "Efficient decryption algorithms for extension field cancellation type encryption schemes," Australasian Conference on Information Security and Privacy – ACISP 2018, vol.10946 of LNCS, pp.487–501, Springer, 2018.
[24] T. Yasuda and K. Sakurai, "A multivariate encryption scheme with Rainbow," Information and Communications Security–ICICS 2015, vol.9543 of LNCS, pp.236–251, Springer, 2016.

Yacheng Wang received the M.Sc. degree in mathematics from Kyushu University. He is currently a Ph.D. student in the the Graduate School of Information Science and Technology at University of Tokyo. His research interest is multivariate cryptography.



Yasuhiko Ikematsu received the Ph.D. in Mathematics in 2016 from Kyushu University. He is currently a postdoctoral fellow in the Graduate School of Information Science and Technology at University of Tokyo. His research interests include number theory and multivariate cryptography.

Dung Hoang Duong received the Ph.D. in Mathematics in 2013 from Leiden University. He is currently a lecturer in the School of Computing and Information Technology at University of Wollongong. His research interests include asymptotic group theory, multivariate and lattice cryptography.



Tsuyoshi Takagi received the B.Sc. and M.Sc. degrees in mathematics from Nagoya University in 1993 and 1995, respectively. He was engaged in research on network security at NTT Laboratories from 1995 to 2001. He received the PhD from the Technical University of Darmstadt in 2001. He was an Assistant Professor in the Department of Science at Technical University of Darmstadt until 2005. He was a Professor at Kyushu University until 2018. He is currently a Professor in the Graduate School of Information Science and Technology at University of Tokyo. His current research interests are information security and cryptography. He received DOCOMO Mobile Science Award in 2013, IEICE Achievement Award in 2013, and JSPS Prize in 2014. Dr. Takagi was a Program Chair of the 7th International Conference on Post-Quantum Cryptography, PQCrypto 2016.