Piece In Hand Concept for Enhancing the Security of Multivariate Type Public Key Cryptosystems: Public Key Without Containing All the Information of Secret Key

Shigeo Tsujii[†] Kohtaro Tadaki[‡] Ryou Fujita[§]

[†] Institute of Information Security
2–14–1 Tsuruya-cho, Kanagawa-ku, Yokohama-shi, 221–0835 Japan
[‡] 21st Century Center OF Excellence Program, Chuo University
1–13–27 Kasuga, Bunkyo-ku, Tokyo, 112–8551 Japan
[§] Division of Management Science, Graduate School of Engineering
Tokyo University of Science, 1–3 Kagurazaka, Shinjuku-ku, Tokyo, 162–8601 Japan

Abstract. We propose a new concept, named *piece in hand*, which can be applicable to a wide class of multivariate type public key cryptosystems to enhance their security. The piece in hand provides such cryptosystems with a new type of trapdoor which is compatible with the trapdoor originally equipped in them. The piece in hand concept is based on a new paradigm for public key cryptosystem in general. On the one hand, in most traditional public key cryptosystems such as the RSA and ElGamal schemes, the public key contains all the information of the secret key. On the other hand, in our scheme, the piece in hand, which is a part of the secret key, is not contained in the public key but is taken from outside of the public key to plug in during the decryption. In this paper, we illustrate how to apply the piece in hand concept to enhance the security of multivariate type public key cryptosystems, by presenting the general theory for the use of the concept.

Key words: public key cryptosystem, multivariate polynomial, multivariate type public key cryptosystem, piece in hand concept

1 Introduction

In the field of algorithmic research for the construction of new public key cryptosystems, various methods employing multivariate polynomials have been made actively. One of the backgrounds of this trend seems to be formed by the sense of emergency against the possibility of advent of quantum computers in the future. Although public key cryptosystems based on the intractability of prime factorization or discrete logarithm problem are presently assumed to be secure, there is the growing concern that such security would not be guaranteed in the quantum computer age. However, the early attempts in 1980s such as [7, 11] to construct multivariate type public key cryptosystems are made before such a threat posed by quantum computer are known. The aim of them is to construct a more efficient public key cryptosystem than the RSA public key cryptosystem. In fact, another motive of devising new public key cryptosystems in recent years is to implement more fast encryption algorithm compared with RSA (based on prime factorization), or elliptic curve cryptosystem (based on discrete logarithm problem). With having the same critical minds, a variety of researches on multivariate type public key cryptosystems, such as [3, 8, 12, 10, 6, 9, 1, 2, 4, 5, 14], have been conducted so far.

1.1 Piece in Hand Concept for Public Key Cryptosystem

This paper presents useful concept of general framework to enhance any type of multivariate type public key cryptosystems where the public key is expressed by multivariate polynomials. Our concept

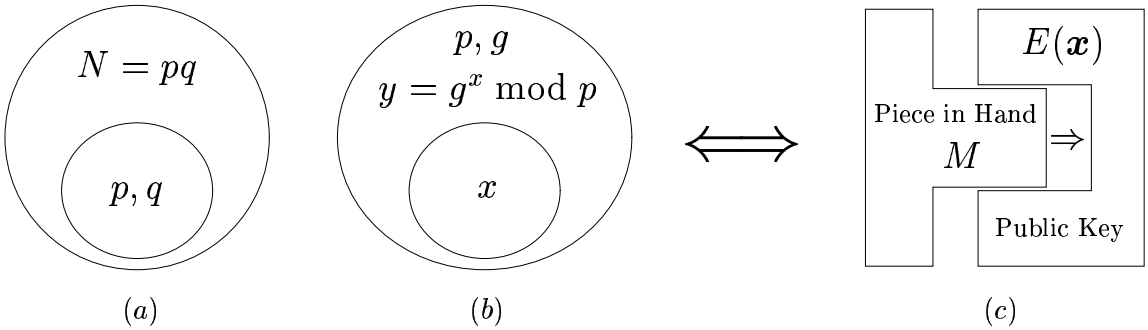

Figure 1: The difference between the traditional and our public key cryptosystem.

is based on a new paradigm for public key cryptosystem in general. Shown in Figure 1 is the difference between the traditional paradigm and our new paradigm. In most traditional public key cryptosystems, the public key contains all the information of secret key, i.e., if unlimited computational power is available, anyone can compute the secret key from the public key. For example, in the RSA scheme, (a) of Figure 1, the public key N contains the secret key (p, q) via $N = pq$, and in the ElGamal scheme, (b) of Figure 1, the public key (p, g, y) contains the secret key x as the discrete logarithm of y, where g is a primitive root of the multiplicative group of residues modulo a prime p, $(\mathbb{Z}/p\mathbb{Z})^*$, and $y = g^x \bmod p$. In contrast to these traditional schemes, we propose a scheme for public key cryptosystem where a certain part of the secret key, called *piece in hand*,[1] is not contained in the public key but is taken from outside of the public key to plug in during the decryption. In the public key cryptosystems considered in the present paper, the public key is expressed by a system of multivariate polynomials $E(\boldsymbol{x})$, and a piece in hand is represented by a certain matrix M. Thus, in our scheme, the piece in hand matrix M is not contained in the public key $E(\boldsymbol{x})$ but is plugged in during the decryption, as illustrated in (c) of Figure 1.

1.2 Schemes of Multivariate Type Public Key Cryptosystems

A multivariate type public key cryptosystem such as in [8, 12, 10, 9, 5] can be considered to comply with the following scheme: Let $\mathbf{F}_q$ be a finite field which has q elements. A plain text is represented by a column vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_k)^T$, and a cipher text is represented by a column vector $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)^T$, where the components x_i and y_i are in $\mathbf{F}_q$, and T denotes the transpose of vector. Then the encryption process is given by the following transformation from $\boldsymbol{x}$ to $\boldsymbol{y}$.

$$\boldsymbol{y} = E(\boldsymbol{x}) = (B \circ F_0 \circ A)(\boldsymbol{x}). \tag{1}$$

Here A and B are invertible linear transformations on ${\mathbf{F}_q}^k$ and ${\mathbf{F}_q}^n$, respectively. Thus we can assume that A is an invertible $k \times k$ matrix and B is an invertible $n \times n$ matrix, where the entries of both A and B are in $\mathbf{F}_q$. F_0 is a nonlinear function from ${\mathbf{F}_q}^k$ to ${\mathbf{F}_q}^n$ such that the components in $F_0(\boldsymbol{u})$ are polynomials in $\mathbf{F}_q[\boldsymbol{u}]$, where $\mathbf{F}_q[\boldsymbol{u}]$ is the set of all polynomials in variables $u_1, u_2, \ldots, u_k$ with coefficients in $\mathbf{F}_q$, and a vector $\boldsymbol{u} = (u_1, u_2, \ldots, u_k)^T$ is related to $\boldsymbol{x}$ by $\boldsymbol{u} = A\boldsymbol{x}$. In this scheme, q and $E(\boldsymbol{x})$ form the public key, and A, B and F_0 form the secret key. $E(\boldsymbol{x})$ is assumed to be constructed

[1] In Japanese chess, *Shogi*, a captured piece is called a *piece in hand*. Unlike in the case of chess, in Shogi a player can get in a piece in hand on the board instead of moving a piece on it. In our paradigm, a piece in hand is taken from the outside of the public key, and is brought into the decryption. However, there is a difference between a piece in hand in Shogi and our concept: Pieces in hand are a public information in Shogi, whereas a piece in hand is secret in our paradigm.

so that, without the knowledge about the secret key, it is difficult to decipher $\boldsymbol{x}$ from $\boldsymbol{y} = E(\boldsymbol{x})$ in polynomial-time.

Let us consider the situation that Bob has the secret key and Alice transmits her cipher text $\boldsymbol{y} = E(\boldsymbol{x})$ to Bob. When Bob receives the cipher text, using the secret key he can efficiently decipher it to obtain the plain text $\boldsymbol{x}$. On the other hand, it is intractable for an eavesdropper, Catherine, to recover $\boldsymbol{x}$ from $\boldsymbol{y}$, since she has no knowledge about the secret key and she has to solve the equation $E(\boldsymbol{x}) = \boldsymbol{y}$ on $\boldsymbol{x}$ directly. The form of the nonlinear function F_0 determines the security of this type of public key cryptosystem, and various methods of constructing F_0 have been proposed so far. In a certain choice of the form of F_0 [11, 12], the sequential solution method can be available to Bob in the decryption. The method is explained in Subsection 1.4 below. In 1986, based on the sequential solution method, a new public key cryptosystem was proposed by [11], and then the cryptosystem was broken for the special case where rational functions are used. Later on, in 1989, [12] proposed the revised version of [11], where birational transformation, named *core transformation*, was employed.[2] The core transformation is a certain type of trapdoor equipped in F_0, and make it difficult to invert the public key $E(\boldsymbol{x})$ in combination with the sequential solution method. No attack to this revised version has been succeeded so far.

In this paper, we propose a new concept, piece in hand (PH, for short), which can be applicable to any type of public key cryptosystem with the form of (1) in order to enhance its security. Our PH method is of great generality on the scope of its application. For understandability, however, we first illustrate a new paradigm introduced by the PH concept, especially based on a sequential solution type public key cryptosystem such as in [11] and [12]. We describe the way of application of our PH method in the most general form later.

1.3 Organization

The paper is organized as follows. In Section 2, based on a sequential solution type public key cryptosystem, we illustrate one form of our PH method along with some example with small values of parameters. The general procedure to design the secret key of this illustrative cryptosystem is described in Section 3. Section 4 is devoted to describing how to apply our PH method to any type of public key cryptosystem with the general form of (1). We then consider the attack by computing a Gröbner basis of the public key, which may be a threat to any type of proposed multivariate type public key cryptosystem, and we present countermeasures against the attack to keep our PH method effective in Section 5. We conclude this paper with a discussion about the future direction of our work in Section 6.

1.4 Sequential Solution Method

As a preliminary to the next section, we here recall the idea of the sequential solution method briefly by considering the following simple example. In this method, multivariate equations $\boldsymbol{w} = F_0(\boldsymbol{u})$ on $\boldsymbol{u}$ are constructed so that they can be solved easily in a sequential manner. For example, the following equations have a typical form to which the method can be applied.

$$\begin{pmatrix} w_3 \\ w_2 \\ w_1 \end{pmatrix} = F_0 \begin{pmatrix} u_3 \\ u_2 \\ u_1 \end{pmatrix} = \begin{pmatrix} 6{u_2}^2 & 4u_1u_2 & 2 \\ 0 & 2{u_1}^2 & 5 \\ 0 & 0 & 3 \end{pmatrix} \begin{pmatrix} u_3 \\ u_2 \\ u_1 \end{pmatrix} \bmod 7. \tag{2}$$

Note that we here work on the finite field $\mathbf{F}_7$. As in this example, in the sequential solution type public key cryptosystems [11, 12] the nonlinear function F_0 is chosen to be represented by an upper

[2] The paper [12] was originally written in Japanese. We include an English translation of [12] in Appendix. See Appendix for the detail of the work [12].

triangular matrix whose entries are polynomial or rational functions of the argument of F_0. By the form of F_0, given $w_1, w_2, w_3 \in \mathbf{F}_7$, we can sequentially calculate the values of u_1, u_2 and u_3 in $\mathbf{F}_7$ in this order. The calculation proceeds as follows: First, by (2) we obtain three equations

$$u_1 = \frac{w_1}{3} \bmod 7 \tag{3}$$

$$u_2 = \frac{w_2 - 5u_1}{2{u_1}^2} \bmod 7 \tag{4}$$

$$u_3 = \frac{w_3 - 4u_1{u_2}^2 - 2u_1}{6{u_2}^2} \bmod 7. \tag{5}$$

The value of u_1 is already obtained in (3). By substituting this value into (4), we obtain the value of u_2. Then, by substituting the values of u_1 and u_2 into (5), we obtain the value of u_3. Thus the calculation is completed and we obtain the solution of the equations (2) on (u_1, u_2, u_3).

2 Illustration of PH method Based on Sequential Solution Method

In this section, we outline our illustrative multivariate type public key cryptosystem which is designed to demonstrate the procedure for enhancing the security by the PH method. This cryptosystem is a sequential solution type public key cryptosystem.

2.1 Scheme of the Illustrative Cryptosystem

In our illustrative cryptosystem, the nonlinear function F_0 is chosen as follows: Let $Q = (q_{i,j})$ be an $n \times k$ matrix of rank k whose entries are in $\mathbf{F}_q$, and let $F = (f_{i,j})$ be an $n \times n$ matrix such that each entries $f_{i,j}$ in F is in $\mathbf{F}_q[\boldsymbol{v}]$, where a vector $\boldsymbol{v} = (v_1, v_2, \ldots, v_n)^T$ is related to $\boldsymbol{u}$ by $\boldsymbol{v} = Q\boldsymbol{u}$. We call F a *nonlinear matrix*. Then the encryption of our public key cryptosystem is given by the transformation (1) with $F_0(\boldsymbol{u}) = FQ\boldsymbol{u}$, i.e., $\boldsymbol{y} = E(\boldsymbol{x}) = BFQA\boldsymbol{x}$. We assume that $n > k$. Since Q is an $n \times k$ matrix with $n > k$, the expression $\boldsymbol{v}$ is redundant in representing the content of $\boldsymbol{u} = A\boldsymbol{x}$. Thus we call Q a *redundantization matrix* since Q redundantizes any vector on which Q operates. Then (A, B, Q, F) constitutes a part of the whole secret key. F should be designed in such a way that anyone who only knows the part (A, B, Q, F) of the secret key cannot efficiently recover a plain text from the corresponding cipher text.

Now we introduce a PH matrix, which is the last component of the secret key. Let $M = (m_{i,j})$ be an $h \times n$ matrix whose entries $m_{i,j}$ are in $\mathbf{F}_q$. M has the function for simplifying F by multiplying itself to F from the left. We denote by H the result MF of the multiplication. As a result, Bob can efficiently recover the plain text $\boldsymbol{x}$ from the cipher text $\boldsymbol{y}$ by solving the following equation on $\boldsymbol{x}$:

$$HQA\boldsymbol{x} = MB^{-1}\boldsymbol{y}, \tag{6}$$

which follows from (1). We call M a *PH matrix*. The detail of the construction of M will be explained in Section 3

The public key is the pair $(q, E(\boldsymbol{x}))$, where $E(\boldsymbol{x})$ is the system of multivariate polynomials obtained by expanding the products $BFQA\boldsymbol{x}$ and then trimming them. Using the public key, Alice encrypts any plain text vector $\boldsymbol{x}$ to generate the corresponding cipher text vector $\boldsymbol{y}$, where $\boldsymbol{y} = E(\boldsymbol{x}) = BFQA\boldsymbol{x}$, and then sends $\boldsymbol{y}$ to Bob.

The secret key is the quintuple (A, B, Q, F, M). On receiving $\boldsymbol{y}$, Bob first multiplies B^{-1} to $\boldsymbol{y}$ to obtain $\boldsymbol{w} = B^{-1}\boldsymbol{y}$. Bob then multiplies M to $\boldsymbol{w}$, and the following equations on $\boldsymbol{u}$ are obtained by (6).

$$M\boldsymbol{w} = H\boldsymbol{v} \tag{7}$$

$$\boldsymbol{v} = Q\boldsymbol{u}. \tag{8}$$

Due to the effect of PH matrix M, the above system of multivariate equations on $\boldsymbol{u}$ can be solved efficiently by the predetermined method such as the sequential solution method. Finally, by solving $\boldsymbol{u} = A\boldsymbol{x}$ on $\boldsymbol{x}$, Bob obtains the plain text vector $\boldsymbol{x}$.

The encryption and decryption processes are schematically represented in Figure 2.

- Encryption

$$\boxed{\boldsymbol{y}} = \boxed{B} \times \boxed{F} \times \boxed{\mathrm{Q}} \times \boxed{A} \times \boxed{\boldsymbol{x}}$$

- Decryption

$$\begin{aligned} \boxed{B^{-1}} \times \boxed{\boldsymbol{y}} &= \boxed{F} \times \boxed{\mathrm{Q}} \times \boxed{A} \times \boxed{\boldsymbol{x}} \\ &\downarrow \\ \boxed{M} \times \boxed{B^{-1}} \times \boxed{\boldsymbol{y}} &= \boxed{M} \times \boxed{F} \times \boxed{\mathrm{Q}} \times \boxed{A} \times \boxed{\boldsymbol{x}} \\ &= \boxed{H} \times \boxed{Q} \times \boxed{A} \times \boxed{\boldsymbol{x}} \end{aligned}$$

Figure 2: Encryption and decryption in PH matrix method

2.2 Example with Small Values of Parameters

Before giving the general prescription for choosing Q, F, and M in our PH matrix method, we give below an example with small values of parameters for our illustrative cryptosystem, where these matrices are appropriately chosen. We here set $q = 7$ and therefore we work on the finite field $\mathbf{F}_7$. We then set $k = h = 3$ and $n = 5$. Thus the plain text vectors and the cipher text vectors are in ${\mathbf{F}_7}^3$ and ${\mathbf{F}_7}^5$, respectively.

We choose the secret key (A, B, Q, F, M) as follows. The square matrices A and B are chosen to be the identity matrix. The redundantization matrix Q is chosen as

$$Q = \begin{pmatrix} 1 & 3 & 5 \\ 0 & 4 & 2 \\ 0 & 1 & 1 \\ 0 & 0 & 6 \\ 0 & 0 & 1 \end{pmatrix}. \tag{9}$$

The nonlinear matrix F is chosen as

$$F = \begin{pmatrix} f_1 & 3f_2 & 0 & 3f_4 & 1 \\ 0 & 2g_1 & f_3 & 5f_4 & 1 \\ 0 & 2f_2 & 4f_3 & f_4 & 2+3g_2 \\ 2f_1 & 6g_1 & 2f_3 & 2f_4 & 3+5g_2 \\ f_1 & 3f_2 & 2f_3 & 2f_4 & 5 \end{pmatrix}, \tag{10}$$

where the entries f_1, f_2, f_3, f_4, g_1 and g_2 are polynomial functions of the redundant vector $\boldsymbol{v}$ to which $\boldsymbol{u}$ is made redundant by the redundantization matrix Q. Note here that $2g_1, 6g_1, 3g_2$ and $5g_2$ will be

removed in the decryption process due to the effect of the PH matrix M which is chosen to be

$$M = \begin{pmatrix} 1 & 6 & 1 & 5 & 0 \\ 5 & 0 & 0 & 0 & 2 \\ 0 & 4 & 3 & 1 & 5 \end{pmatrix}. \tag{11}$$

This can be checked by multiplying M to F from the left, i.e., we see that

$$H = MF = \begin{pmatrix} 4f_1 & 5f_2 & 6f_3 & 2f_4 & 3 \\ 0 & 0 & 4f_3 & 5f_4 & 1 \\ 0 & 0 & 0 & 0 & 3 \end{pmatrix},$$

where g_1 and g_2 are certainly removed. The g_i's play a role in randomizing F and therefore the public key $E(\boldsymbol{x})$. Hence the g_i's are called *randomizing polynomials*. Thus we have

$$H\boldsymbol{v} = \begin{pmatrix} 4f_1v_1 + 5f_2v_2 + 6f_3v_3 + 2f_4v_4 + 3v_5 \\ 4f_3v_3 + 5f_4v_4 + v_5 \\ 3v_5 \end{pmatrix}. \tag{12}$$

On the other hand, by $\boldsymbol{v} = Q\boldsymbol{u}$, we have the relation between $\boldsymbol{v}$ and $\boldsymbol{u}$ as follows.

$$\begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{pmatrix} = \begin{pmatrix} u_1 + 3u_2 + 5u_3 \\ 4u_2 + 2u_3 \\ u_2 + u_3 \\ 6u_3 \\ u_3 \end{pmatrix}. \tag{13}$$

Thus, using the sequential solution method, Bob can efficiently solve the equations (7) and (8) to obtain $\boldsymbol{x}$ from $\boldsymbol{y}$, as seen below.

Note that, in order to make the elimination of g_1 and g_2 in the matrix H possible, M has to have rank less than n. On the other hand, in order to uniquely recover the plain text vector $\boldsymbol{x}$ from the cipher text vector $\boldsymbol{y}$, M has to have rank at least k. Thus $k < n$ has to hold. This is the reason why the vector $\boldsymbol{u} = A\boldsymbol{x}$ needs to be made redundant in advance by Q and transformed to the vector $\boldsymbol{v} = Q\boldsymbol{u}$.

We here choose f_1, f_2, f_3, f_4, g_1 and g_2 in the nonlinear matrix F as

$$\begin{cases} f_1 = & & 2v_2 & +2v_3 & +v_4 & +5v_5 & +3 \\ f_2 = & & & 3v_3 & +2v_4 & +4v_5 & +1 \\ f_3 = & & & & 4v_4 & +3v_5 & +6 \\ f_4 = & & & & & v_5 & +4 \\ g_1 = & 2v_1 & +3v_2 & +v_3 & +4v_4 & +5v_5 & +2 \\ g_2 = & v_1 & +4v_2 & +3v_3 & +2v_4 & +2v_5 & +5 \end{cases}$$

in the concrete. Then the public key is the pair $(7, E(\boldsymbol{x}))$ where $E(\boldsymbol{x})$ is the system of trimmed multivariate polynomials obtained by simplifying the products $BFQA\boldsymbol{x}$. We see that $E(\boldsymbol{x})$ is given by

$$E(\boldsymbol{x}) = \begin{pmatrix} 3x_1x_2 + 3x_1x_3 + 3x_2^2 + 4x_2x_3 + 3x_1 + 3x_3 \\ 2x_1x_2 + x_1x_3 + 5x_2^2 + 2x_2x_3 + 3x_3^2 + x_2 + 2x_3 \\ 3x_1x_3 + 3x_2^2 + 2x_2x_3 + 4x_2 + 6x_3 \\ 5x_1x_2 + 5x_2^2 + 4x_2x_3 + 6x_1 + x_2 + 2x_3 \\ 3x_1x_2 + 3x_1x_3 + 3x_2^2 + 2x_2x_3 + 6x_3^2 + 3x_1 + 5x_2 + 2x_3 \end{pmatrix}.$$

Assume that Alice wants to send Bob the plain text vector $\boldsymbol{x} = \begin{pmatrix}5 & 1 & 4\end{pmatrix}^T$. Then, using the public key $(7, E(\boldsymbol{x}))$, Alice calculates the cipher text vector $\boldsymbol{y}$ as $\boldsymbol{y} = E(\boldsymbol{x}) = \begin{pmatrix}2 & 2 & 1 & 1 & 0\end{pmatrix}^T$. Alice then sends Bob this $\boldsymbol{y}$. On receiving the cipher text vector $\boldsymbol{y}$, Bob first multiplies B^{-1} to it to obtain $\boldsymbol{w} = B^{-1}\boldsymbol{y} = \boldsymbol{y} = \begin{pmatrix}2 & 2 & 1 & 1 & 0\end{pmatrix}^T$. Note that, in this example, B is assumed to be the identity matrix. Bob then multiplies the PH matrix M to $\boldsymbol{w}$ to get $M\boldsymbol{w} = \begin{pmatrix}6 & 3 & 5\end{pmatrix}^T$. Thus by (7) and (12) Bob has

$$\left\{\begin{array}{rrrrrl} 4f_1v_1 & +5f_2v_2 & +6f_3v_3 & +2f_4v_4 & +3v_5 & = 6 \\ & & 4f_3v_3 & +5f_4v_4 & +v_5 & = 3 \\ & & & & 3v_5 & = 5. \end{array}\right. \tag{14}$$

Note that the randomizing polynomials g_1 and g_2 which melt away into the public key $E(\boldsymbol{x})$ have been removed in the above system of equations. Using (13) and (14), Bob proceeds through the sequential solving process as follows:

$$\left.\begin{array}{l} 5 = 3v_5 \\ \Rightarrow v_5 = 4 \end{array}\right\} \longmapsto \left.\begin{array}{l} u_3 = v_5 \\ \Rightarrow u_3 = 4 \end{array}\right\} \longmapsto \left.\begin{array}{l} v_4 = 6u_3 \\ \Rightarrow v_4 = 3 \end{array}\right\} \longmapsto \left.\begin{array}{c} 3 = 4f_3v_3 + 5f_4v_4 + v_5 \\ = v_3 + 1 + 4 \\ \Rightarrow v_3 = 5 \end{array}\right\} \longmapsto \left.\begin{array}{l} u_2 + u_3 = v_3 \\ \Rightarrow u_2 = 1 \end{array}\right\}$$

$$\longmapsto \left.\begin{array}{l} v_2 = 4u_2 + 2u_3 \\ \Rightarrow v_2 = 5 \end{array}\right\} \longmapsto \left.\begin{array}{c} 6 = 4f_1v_1 + 5f_2v_2 + 6f_3v_3 + 2f_4v_4 + 3v_5 \\ = 2v_1 + 5 + 4 + 6 + 5 \\ \Rightarrow v_1 = 0 \end{array}\right\} \longmapsto \left.\begin{array}{l} v_1 = u_1 + 3u_2 + 5u_3 \\ \Rightarrow u_1 = 5 \end{array}\right\}.$$

Thus Bob obtains $\boldsymbol{u} = \begin{pmatrix}5 & 1 & 4\end{pmatrix}^T$. The sequential solving process is summarized as: $v_5 \to u_3 \to v_4 \to v_3 \to u_2 \to v_2 \to v_1 \to u_1$. Since $\boldsymbol{x} = \boldsymbol{u}$ in this example, Bob finally obtains the plain text vector $\boldsymbol{x} = \begin{pmatrix}5 & 1 & 4\end{pmatrix}^T$.

3 How to Design PH Matrix and Other Related Matrices

3.1 General Procedure for Designing Q, F and M

In this section, we describe the detail of the design of the matrices Q, F and M.

To begin with, the $n \times k$ redundantization matrix $Q = (q_{i,j})$ is designed, where $n > k$. For each $j = 1, \ldots, k$, let l_j be the row number of the nonzero entry in the j-th column of Q such that all entries at the lower positions are zero, i.e., $l_j = \max\{\, i \mid 1 \le i \le n \;\&\; q_{i,j} \neq 0 \,\}$. Then the redundantization matrix Q is chosen to satisfy the following condition.

Condition 1. $\operatorname{rank} Q = k$ *(i.e., Q has full column rank) and* $1 \le l_1 < l_2 < \cdots < l_{k-1} < l_k = n$. □

For example, $k = 3$ and $(l_1, l_2, l_3) = (1, 3, 5)$ for the matrix Q given by (9), and therefore this matrix Q is certainly competent to be a redundantization matrix.

The nonlinear matrix F is designed as the sum of two particular kind of matrices $S(\boldsymbol{v})$ and $N(\boldsymbol{v})$:

$$F = S(\boldsymbol{v}) + N(\boldsymbol{v}). \tag{15}$$

Here $S(\boldsymbol{v})$ is designed to have the following form:

$$S(\boldsymbol{v}) = T\,\mathrm{diag}(f_1(v_2, \ldots, v_n), \ldots, f_{n-1}(v_n), 1), \tag{16}$$

where T is an $n \times n$ matrix whose entries are in $\mathbf{F}_q$, $f_1(v_2, \ldots, v_n), \ldots, f_{n-1}(v_n)$ are in $\mathbf{F}_q[\boldsymbol{v}]$, and $\mathrm{diag}(a_1, \ldots, a_n)$ is the diagonal matrix whose (i,i)-entry is a_i. On the other hand, $N(\boldsymbol{v})$ is designed to have the following form:

$$N(\boldsymbol{v}) = R \cdot G(\boldsymbol{v}), \tag{17}$$

where R is an $n \times n$ matrix whose entries are in $\mathbf{F}_q$, and $G(\boldsymbol{v})$ is an $n \times n$ matrix whose entries $g_{i,j}(\boldsymbol{v})$'s are in $\mathbf{F}_q[\boldsymbol{v}]$. Particularly, the entries of $G(\boldsymbol{v})$ are randomizing polynomials which randomize F. For example, in the case where $n = 5$, we can choose $N(\boldsymbol{v})$ by

$$N(\boldsymbol{v}) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 2g_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3g_2 \\ 0 & 6g_1 & 0 & 0 & 5g_2 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{18}$$

where R and $G(\boldsymbol{v})$ are chosen by

$$R = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 \\ 0 & 6 & 0 & 0 & 5 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{19}$$

$$G(\boldsymbol{v}) = \mathrm{diag}(0, g_1, 0, 0, g_2), \tag{20}$$

and $g_1, g_2 \in \mathbf{F}_q[\boldsymbol{v}]$. Thus F is specified by the constituents T, $f_i(\boldsymbol{v})$'s, R, and $G(\boldsymbol{v})$. Reflecting the performance of our cryptosystem such as security, encryption/decryption speed, and so forth, these constituents should be carefully determined. Therefore, in order to obtain the nonlinear matrix F with desirable properties, we design $f_i(\boldsymbol{v})$'s and $G(\boldsymbol{v})$ appropriately. On the other hand, we design R, M, and T in sequence as follows.

First we choose R so as to satisfy the following condition.

Condition 2. $n \geq k + \mathrm{rank}\, R$. □

This condition is necessary to guarantee the existence of the PH matrix M which satisfies Condition 3 and Condition 4 given below. Once R is designed, we can design the PH matrix $M = (m_{i,j})$ as follows. By (15) and (17), in order to eliminate the randomizing polynomials $g_{i,j}(\boldsymbol{v})$'s from the nonlinear matrix F by the multiplication of M, it is sufficient to impose the following condition on M.

Condition 3. $MR = 0$. □

On the other hand, for the unique recovery of the plain text vector $\boldsymbol{x}$ from the cipher text vector $\boldsymbol{y}$, we have to impose an additional condition on M, as shown in the following consideration. In the decryption, Bob solves the equation on $\boldsymbol{x}$:

$$M\boldsymbol{w} = HQA\boldsymbol{x}. \tag{21}$$

The number of the vectors $M\boldsymbol{w}$ at the left-hand side of (21) is at most $q^{\mathrm{rank}\, M}$. On the other hand, the number of plain text vectors $\boldsymbol{x}$ is exactly q^k. For the unique recovery of $\boldsymbol{x}$, the transformation $HQA\boldsymbol{x}$ on $\boldsymbol{x}$ at the right-hand side of (21) has to be injective. This implies that the inequality $q^{\mathrm{rank}\, M} \geq q^k$ has to hold. Thus, it is necessary for M to have rank at least k. We here impose on M the minimal condition which meets this requirement, as follows.

Condition 4. *M is a $k \times n$ matrix, and* $\mathrm{rank}\, M = k$ *(i.e., M has full row rank).* □

Since R is chosen to satisfy Condition 2, by the following proposition, we can certainly choose M so as to satisfy Condition 3 and Condition 4.

Proposition 3.1. *Let A_1 be an $s \times s$ matrix. If* $\operatorname{rank} A_1 \leq s - t$ *with* $1 \leq t \leq s$ *then there exists an $s \times t$ matrix A_2 such that $A_1 A_2 = 0$ and* $\operatorname{rank} A_2 = t$.

The above proposition is an elementary result of linear algebra.

Next we design T as follows. Let $t_{i,j}$ be the (i,j)-entry of T. Then

$$MS(\boldsymbol{v}) = \begin{pmatrix} f_1 \sum_{j=1}^{n} m_{1,j} t_{j,1} & \cdots & \sum_{j=1}^{n} m_{1,j} t_{j,n} \\ f_1 \sum_{j=1}^{n} m_{2,j} t_{j,1} & \cdots & \sum_{j=1}^{n} m_{2,j} t_{j,n} \\ \vdots & & \vdots \\ f_1 \sum_{j=1}^{n} m_{k,j} t_{j,1} & \cdots & \sum_{j=1}^{n} m_{k,j} t_{j,n} \end{pmatrix}.$$

The design of the form of the above $MS(\boldsymbol{v})$ depends on the trapdoor employed in a multivariate type public key cryptosystem. We assume that the sequential solution algorithm is employed. In this case, since each f_i contains only variables $v_{i+1}, \ldots, v_n$ and Condition 1 holds for Q, it is necessary and sufficient to impose the following condition on T in order to make the sequential solution algorithm work properly, as seen in the proof of Theorem 3.3.

Condition 5. *For every $i \in \{1, \ldots, k\}$, the (i, l_i)-entry in MT is the left-most nonzero entry in the i-th row of MT, i.e.,*

$$\begin{cases} \sum_{j=1}^{n} m_{1,j} t_{j,1} = 0, & \ldots, & \sum_{j=1}^{n} m_{1,j} t_{j,l_1 - 1} = 0 \\ \sum_{j=1}^{n} m_{2,j} t_{j,1} = 0, & \ldots, & \sum_{j=1}^{n} m_{2,j} t_{j,l_2 - 1} = 0 \\ \vdots & & \vdots \\ \sum_{j=1}^{n} m_{k,j} t_{j,1} = 0, & \ldots, & \sum_{j=1}^{n} m_{k,j} t_{j,l_k - 1} = 0 \end{cases}$$

and $\sum_{j=1}^{n} m_{i,j} t_{j,l_i} \neq 0$ for all $i \in \{1, \ldots, k\}$. □

Since M is chosen to satisfy Condition 4, by the following proposition, we can certainly choose T so as to satisfy Condition 5. This proposition is an elementary result of linear algebra.

Proposition 3.2. *Let B_1 be a $t \times s$ matrix and B_2 a $t \times s$ matrix, where $t \leq s$. If* $\operatorname{rank} B_1 = t$ *then there exists an $s \times s$ matrix B_3 such that $B_1 B_3 = B_2$.*

Thus we have all of the secret key and the public key. Then the following theorem holds. The proof of the theorem describes the general decryption procedure in our illustrative public key cryptosystem.

Theorem 3.3. *Suppose that Condition 1, Condition 3, and Condition 5 hold. If none of $f_1(v_2, \ldots, v_n)$, $\ldots, f_{n-1}(v_n)$ is zero, then Bob can uniquely and efficiently recover $\boldsymbol{x}$ from $\boldsymbol{y} = E(\boldsymbol{x})$ using the secret key (A, B, Q, F, M).*

Proof. Let $\boldsymbol{z} = (z_1, z_2, \ldots, z_k)^T$ be $MB^{-1}\boldsymbol{y}$. Then we have

$$\boldsymbol{z} = MF\boldsymbol{v}, \tag{22}$$

$$\boldsymbol{v} = Q\boldsymbol{u}. \tag{23}$$

Given $\boldsymbol{y}$, the value of $\boldsymbol{z}$ can be easily calculated by $\boldsymbol{z} = MB^{-1}\boldsymbol{y}$. On the other hand, once the value of $\boldsymbol{u}$ is obtained, the value of $\boldsymbol{x}$ is also easily calculated by $\boldsymbol{x} = A^{-1}\boldsymbol{u}$. Thus, in order to get the plain text vector $\boldsymbol{x}$ from the cipher text vector $\boldsymbol{y}$, it is sufficient for Bob to have the method which solves the equations (22) and (23) on $\boldsymbol{u}$, given the value of $\boldsymbol{z}$. Let $d_{i,j}$ be the (i,j)-entry of MT. It follows from Condition 3 that, for all $i \in \{1, \ldots, k\}$ and all $j \in \{1, \ldots, n\}$, the (i,j)-entry of MF is equal to $f_j(v_{j+1}, \ldots, v_n) d_{i,j}$. Therefore Bob knows the values of $d_{i,j}$'s. We show that Bob can calculate the value of $\boldsymbol{u}$ by the following procedure. Initially Bob can get the value of u_k using the equations

$z_k = d_{k,n}v_n$ and $v_n = q_{n,k}u_k$. Note here that $d_{k,n} \neq 0$ and $q_{n,k} \neq 0$ by Condition 5 and Condition 1, respectively.

For any $\tau \in \{1, \ldots, k-1\}$, assume that Bob has so far had the values of $u_{\tau+1}, \ldots, u_k$. Then, by the form of Q, Bob knows all of the values of $v_{l_\tau+1}, \ldots, v_n$. Here, by (22) and Condition 5, the following equation holds:

$$\begin{aligned} z_\tau = & \sum_{j=l_\tau}^{n} f_j(v_{j+1}, \ldots, v_n) d_{\tau,j} v_j \\ = & f_{l_\tau}(v_{l_\tau+1}, \ldots, v_n) d_{\tau,l_\tau} v_{l_\tau} + \sum_{j=l_\tau+1}^{n} f_j(v_{j+1}, \ldots, v_n) d_{\tau,j} v_j. \end{aligned}$$

Note that, at the most right-hand side of the above equation, only $v_{l_\tau}, \ldots, v_n$ are contained as variable and especially v_{l_τ} is contained only in the first term. On the other hand, it follows from Condition 5 and the assumption on f_i's that $f_{l_\tau}(v_{l_\tau+1}, \ldots, v_n) d_{\tau,l_\tau} \neq 0$. Thus, Bob can calculate the value of v_{l_τ} from the values of $v_{l_\tau+1}, \ldots, v_n$ and z_τ. Moreover, by (23), $v_{l_\tau} = \sum_{j=\tau}^{k} q_{l_\tau,j} u_j$. Hence, Bob can calculate the value of u_τ from the values of $u_{\tau+1}, \ldots, u_k$ and v_{l_τ}.

Thus, according to the above procedure, Bob can finally get all values of $u_1, \ldots, u_k$. Hence the proof is completed. □

Note that, for sufficiently large q, none of $f_1(v_2, \ldots, v_n), \ldots, f_{n-1}(v_n)$ is zero for almost all of $(v_2, \ldots, v_n)^T \in {\mathbf{F}_q}^{n-1}$. Thus, in such a choice of q, Bob can uniquely decipher any cipher text $\boldsymbol{y}$ in all likelihood, provided that Condition 1, Condition 3, and Condition 5 hold.

3.2 Illustration of the Design of Q, F and M

In order to clarify our PH matrix method described in the previous subsection, we consider an example with small values of parameters in this subsection. We derive the nonlinear matrix F and the PH matrix M considered in Subsection 2.2. Thus, we consider the case where $n = 5$ and $k = 3$, and we work on the finite field $\mathbf{F}_7$.

First, the redundantization matrix Q is chosen by (9). For this Q, $(l_1, l_2, l_3) = (1, 3, 5)$. We then choose R and $G(\boldsymbol{v})$ by (19) and (20), respectively. Thus $N(\boldsymbol{v})$ is given by (18). Note that R has rank 2, and therefore Condition 2 holds. In order to determine the PH matrix M, we note that Condition 3 on M is equivalent to the following equations.

$$\begin{cases} 2m_{1,2} + 6m_{1,4} = 0, & 3m_{1,3} + 5m_{1,4} = 0 \\ 2m_{2,2} + 6m_{2,4} = 0, & 3m_{2,3} + 5m_{2,4} = 0 \\ 2m_{3,2} + 6m_{3,4} = 0, & 3m_{3,3} + 5m_{3,4} = 0. \end{cases}$$

By solving the above equations and choosing M so as to satisfy Condition 4, we have the PH matrix M given by (11). Next we determine the matrix T. By (11) we see that Condition 5 on T is equivalent to the following equations.

$$\begin{cases} t_{1,1} + 6t_{2,1} + t_{3,1} + 5t_{4,1} + 0 \neq 0 \\ 5t_{1,1} + 0 + 0 + 0 + 2t_{5,1} = 0 \\ 5t_{1,2} + 0 + 0 + 0 + 2t_{5,2} = 0 \\ 5t_{1,3} + 0 + 0 + 0 + 2t_{5,3} \neq 0 \\ 0 + 4t_{2,1} + 3t_{3,1} + t_{4,1} + 5t_{5,1} = 0 \\ 0 + 4t_{2,2} + 3t_{3,2} + t_{4,2} + 5t_{5,2} = 0 \\ 0 + 4t_{2,3} + 3t_{3,3} + t_{4,3} + 5t_{5,3} = 0 \\ 0 + 4t_{2,4} + 3t_{3,4} + t_{4,4} + 5t_{5,4} = 0 \\ 0 + 4t_{2,5} + 3t_{3,5} + t_{4,5} + 5t_{5,5} \neq 0. \end{cases}$$

Solving the above equations and generating the remaining entries at random, we get

$$T = \begin{pmatrix} 1 & 3 & 0 & 3 & 1 \\ 0 & 0 & 1 & 5 & 1 \\ 0 & 2 & 4 & 1 & 2 \\ 2 & 0 & 2 & 2 & 3 \\ 1 & 3 & 2 & 2 & 5 \end{pmatrix}. \tag{24}$$

Substituting (24) to (16), we have

$$S(\boldsymbol{v}) = \begin{pmatrix} f_1 & 3f_2 & 0 & 3f_4 & 1 \\ 0 & 0 & f_3 & 5f_4 & 1 \\ 0 & 2f_2 & 4f_3 & f_4 & 2 \\ 2f_1 & 0 & 2f_3 & 2f_4 & 3 \\ f_1 & 3f_2 & 2f_3 & 2f_4 & 5 \end{pmatrix}. \tag{25}$$

Thus, substituting (18) and (25) to (15), we finally get the nonlinear matrix F given by (10).

4 General Prescription for Enhancement by the PH Method

In the previous sections, we demonstrate how to apply our PH matrix method to the illustrative sequential solution type public key cryptosystem, through the installation of the method in the nonlinear function F_0. In this section, we describe the general prescription for the enhancement of the security of any given multivariate type public key cryptosystem by our PH method. Let $\mathcal{K}$ be any multivariate type public key cryptosystem whose encryption process is described by (1). We construct new multivariate type public key cryptosystem $\widetilde{\mathcal{K}}$ through an application of our PH method directly to the public key $E(\boldsymbol{x})$ of $\mathcal{K}$ in a sequential manner, unlike in the case of the previous cryptosystem where the PH method is installed in an integrated manner with the nonlinear function. A public key $\widetilde{E}(\boldsymbol{x})$ of $\widetilde{\mathcal{K}}$ is constructed from the original public key $E(\boldsymbol{x})$ of $\mathcal{K}$ by the following transformation:

$$\widetilde{E}(\boldsymbol{x}) = S \cdot E(\boldsymbol{x}) + R \cdot H(\boldsymbol{x}).$$

Here S is an $l \times n$ matrix whose entries are in $\mathbf{F}_q$. In order to make our PH method work properly, we assume that $l > n$. Thus S plays a role as the redundantization matrix in the previous illustrative cryptosystem. On the other hand, R is an $l \times l$ matrix whose entries are in $\mathbf{F}_q$, and $H(\boldsymbol{x}) = (h_1(\boldsymbol{x}), \ldots, h_l(\boldsymbol{x}))^T$ is a vector whose components $h_i(\boldsymbol{x})$'s are in $\mathbf{F}_q[\boldsymbol{x}]$. The term $R \cdot H(\boldsymbol{x})$ plays a role in randomizing $\widetilde{E}(\boldsymbol{x})$. Hence the $h_i(\boldsymbol{x})$'s have to be chosen so that in $\widetilde{E}(\boldsymbol{x})$ the $h_i(\boldsymbol{x})$'s cannot be indistinguishable from the polynomials which come from $E(\boldsymbol{x})$. A plain text of $\widetilde{\mathcal{K}}$ is represented by a vector in ${\mathbf{F}_q}^k$ in the same way as in $\mathcal{K}$. For any plain text vector $\boldsymbol{x} \in {\mathbf{F}_q}^k$ of $\widetilde{\mathcal{K}}$, the corresponding cipher text of $\widetilde{\mathcal{K}}$ is represented by a vector $\boldsymbol{z} \in {\mathbf{F}_q}^l$ and is caluculated by $\boldsymbol{z} = \widetilde{E}(\boldsymbol{x})$.

We choose the R, PH matrix M, and S in sequence so as to satisfy the following three conditions. Using the same argument used in Subsection 3.1, we can show that this choice is efficiently possible.

Condition 6. $l \geq n + \operatorname{rank} R$. □

Condition 7. *M is an $n \times l$ matrix such that $MR = 0$ and* $\operatorname{rank} M = n$. □

Condition 8. *$MS = I$, where I is the $n \times n$ identity matrix.* □

Then, q and $\widetilde{E}(\boldsymbol{x})$ form the public key of $\widetilde{\mathcal{K}}$. On the other hand, the PH matrix M together with the secret key of $\mathcal{K}$ for the public key q and $E(\boldsymbol{x})$ of $\mathcal{K}$ form the secret key of $\widetilde{\mathcal{K}}$. The decryption of $\widetilde{\mathcal{K}}$ proceeds as follows. Since $M\widetilde{E}(\boldsymbol{x}) = E(\boldsymbol{x})$ by the above conditions, on receiving the cipher text $\boldsymbol{z} = \widetilde{E}(\boldsymbol{x})$ for a plain text $\boldsymbol{x}$, Bob can efficiently obtain $\boldsymbol{y} = E(\boldsymbol{x})$ from the multiplication of $\boldsymbol{z}$ by M. Then, according to the decryption procedure of $\mathcal{K}$, Bob can recover the plain text $\boldsymbol{x}$ using the secret key of $\mathcal{K}$.

5 Attack by Computing Gröbner Bases

In the previous sections, we have just shown how to use the new concept, piece in hand, in order to enhance the security of a general multivariate type public key cryptosystem.

Recently, [2] showed in an experimental manner that computing a Gröbner basis of the public key is likely to be an efficient attack to HFE [10], which is one of the major variants of multivariate type public key cryptosystem. The attack is simply to compute a Gröbner basis for the ideal generated by polynomial components in $E(\boldsymbol{x}) - \boldsymbol{c}$, where $\boldsymbol{c}$ is a cipher text vector. Thus, because of the simplicity of this attack, it may be a threat to any type of proposed multivariate type public key cryptosystem.

Especially, from the point of view of Gröbner bases, the secret linear transformation B in the general scheme (1) of the encryption process may be useless. This is because any ideal I generated by polynomials remains unchanged under the replacement of the generators of I by their linear combinations. Thus the PH concept might be also useless to the Gröbner attack in its primitive implementation considered in the previous sections by the following reason. We here consider the illustrative cryptosystem given in Section 2 and 3 for a while. The application of MB^{-1} to the public key $E(\boldsymbol{x})$ exposes a system of polynomial equations to which the sequential solution algorithm can be applied. Thus, if $E(\boldsymbol{x}) - \boldsymbol{c}$ and $MB^{-1}(E(\boldsymbol{x}) - \boldsymbol{c})$ are regard as the same system of polynomials from the point of view of Gröbner bases, then the PH method might be useless to the Gröbner attack. However, this issue need to be studied in more detail, since M is a $k \times n$ matrix with $k < n$ and therefore the ideal generated by $MB^{-1}(E(\boldsymbol{x}) - \boldsymbol{c})$ might be different from the ideal generated by $E(\boldsymbol{x}) - \boldsymbol{c}$.

Even if the Gröbner attack is effective to break the cryptosystem whose security is enhanced by the PH matrix method, we have the following countermeasures against the attack.

5.1 Hiding the PH Matrix Method by Nonlinear Matrix

One of the countermeasures is to apply any secret nonlinear matrix $\widetilde{F}$ to the public key $E(\boldsymbol{x})$ and use the result as a public key anew. By this countermeasure, the enhancement of the nonlinear matrix F by the PH matrix method is likely to be hidden from the Gröbner attack. The countermeasure is schematically represented in Figure 3, where C is an $n \times n$ matrix whose entries are in $\mathbf{F}_q$. In the mul-

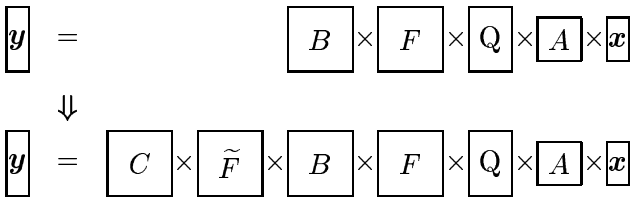

Figure 3: Countermeasure against the Gröbner attack

tivariate type public key cryptosystem proposed in [12], two nonlinear matrices are applied in sequence to generate the public key. Thus the above countermeasure is already taken for this cryptosystem, and we only have to apply the PH matrix method to the first nonlinear matrix for enhancing the security. Note that, in this cryptosystem, each of the two nonlinear matrices is made difficult to invert for the eavesdropper, using a certain birational transformation, called *core transformation*. The use of the core transformation seems to be the reason why no attack to the cryptosystem has been succeeded so far. Thus we can expect the further enhancement of the security of the cryptosystem by the use of the PH matrix method.

5.2 Nonlinearization of the PH Matrix

Another countermeasure against the Gröbner attack is to nonlinearize the PH matrix. The PH matrix was assumed to be a linear matrix so far. In the next stage of the PH method, we can consider a nonlinear PH matrix $M(\boldsymbol{x})$, where some entries in $M(\boldsymbol{x})$ are polynomial functions of $\boldsymbol{x}$. Since an ideal I generated by polynomials may change under the replacement of the generators of I by the multiplication of them by $M(\boldsymbol{x})$, unlike in the case of linear M, the nonlinear PH matrix may provide substantial robustness against the Gröbner attack. Thus, in this countermeasure, we may be allowed to use only a single nonlinear function F_0 as in the cryptosystems considered in the previous sections, unlike in the case of the above countermeasure. In the nonlinear PH matrix scheme, however, some additional r-tuple $a(\boldsymbol{x}) = (a_1(\boldsymbol{x}), \ldots, a_r(\boldsymbol{x}))$ of multivariate polynomials in $\mathbf{F}_q[\boldsymbol{x}]$ has to be published together with $E(\boldsymbol{x})$ by Bob, and $a(\boldsymbol{p})$ has to be transmitted from Alice to Bob in addition to $E(\boldsymbol{p})$ when Alice wants to send Bob a plain text $\boldsymbol{p}$.

In order to illustrate our idea of the nonlinear PH matrix method, we return to the general form of encryption process (1). For simplicity, we here set $n = 4$ and $k = 3$, and A, B to be the identity matrices. Note then that $\boldsymbol{u} = \boldsymbol{x}$. We also assume that $r = 2$ and $q = 7$. As an example, we choose the nonlinear function $F_0(\boldsymbol{x})$ as

$$\begin{pmatrix} \{3a_1(\boldsymbol{x}) + 6a_2(\boldsymbol{x})\}f_1x_1 & +2a_1(\boldsymbol{x})f_2x_2 & +\{3a_1(\boldsymbol{x}) + 2a_2(\boldsymbol{x})\}x_3 & +\{4a_1(\boldsymbol{x}) - 6a_2(\boldsymbol{x})\}g \\ \{4a_1(\boldsymbol{x}) + 3a_2(\boldsymbol{x})\}f_1x_1 & +3a_1(\boldsymbol{x})f_2x_2 & +2a_2(\boldsymbol{x})x_3 & +\{2a_1(\boldsymbol{x}) - 3a_2(\boldsymbol{x})\}g \\ \{a_1(\boldsymbol{x}) + a_2(\boldsymbol{x})\}f_1x_1 & +5a_1(\boldsymbol{x})f_2x_2 & +\{4a_1(\boldsymbol{x}) + 2a_2(\boldsymbol{x})\}x_3 & +\{3a_1(\boldsymbol{x}) - a_2(\boldsymbol{x})\}g \\ \{3a_1(\boldsymbol{x}) + 2a_2(\boldsymbol{x})\}f_1x_1 & +3a_1(\boldsymbol{x})f_2x_2 & +\{a_1(\boldsymbol{x}) + 3a_2(\boldsymbol{x})\}x_3 & +\{6a_1(\boldsymbol{x}) - 2a_2(\boldsymbol{x})\}g \end{pmatrix}, \tag{26}$$

where $f_1(x_2, x_3)$, $f_2(x_3)$, and $g(x_1, x_2, x_3)$ are in $\mathbf{F}_7[x_1, x_2, x_3]$. The $g(x_1, x_2, x_3)$ plays a role in randomizing the public key $E(\boldsymbol{x}) = F_0(\boldsymbol{x})$. We also choose the nonlinear PH matrix $M(\boldsymbol{x})$ as

$$\begin{pmatrix} a_1(\boldsymbol{x})^{q-2} & 0 & 0 & 4a_1(\boldsymbol{x})^{q-2} \\ 5a_1(\boldsymbol{x})^{q-2} + 6a_2(\boldsymbol{x})^{q-2} & 4a_1(\boldsymbol{x})^{q-2} & 6a_1(\boldsymbol{x})^{q-2} + 5a_2(\boldsymbol{x})^{q-2} & 4a_1(\boldsymbol{x})^{q-2} + 4a_2(\boldsymbol{x})^{q-2} \\ a_2(\boldsymbol{x})^{q-2} & 0 & 2a_2(\boldsymbol{x})^{q-2} & 3a_2(\boldsymbol{x})^{q-2} \end{pmatrix}. \tag{27}$$

Then, obviously, entries in $M(\boldsymbol{x})$ are polynomial functions of $\boldsymbol{x}$ as desired. Now, in this scheme, Bob publishes $(a_1(\boldsymbol{x}), a_2(\boldsymbol{x}))$ in addition to $E(\boldsymbol{x})$, and then Alice transmits $(a_1(\boldsymbol{p}), a_2(\boldsymbol{p}))$ and $E(\boldsymbol{p})$ to Bob, where $\boldsymbol{p}$ is a plain text which Alice wants to send to Bob. Note that, on receiving them, Bob can efficiently compute the value of $M(\boldsymbol{p})$ using the values of $a_1(\boldsymbol{p})$ and $a_2(\boldsymbol{p})$. It follows from the Fermat's little theorem that

$$M(\boldsymbol{p})F_0(\boldsymbol{p}) = \begin{pmatrix} a_1(\boldsymbol{p})^{q-1}f_1(p_2, p_3)p_1 \\ a_1(\boldsymbol{p})^{q-1}f_2(p_3)p_2 + \{a_1(\boldsymbol{p})^{q-1} + 6a_2(\boldsymbol{p})^{q-1}\}p_3 \\ a_2(\boldsymbol{p})^{q-1}p_3 \end{pmatrix} = \begin{pmatrix} f_1(p_2, p_3)p_1 \\ f_2(p_3)p_2 \\ p_3 \end{pmatrix}$$

in the case where each of $a_i(\boldsymbol{p})$'s is not zero. We see here that the randomizing polynomial g is removed at the most right-hand side of the above equation. Thus, in such a case, Bob obtains the system of multivariate equations on $\boldsymbol{x}$: $M(\boldsymbol{p})E(\boldsymbol{p}) = \begin{pmatrix} f_1(x_2, x_3)x_1 & f_2(x_3)x_2 & x_3 \end{pmatrix}^T$ to which Bob can apply the sequential solution method to recover the original plain text $\boldsymbol{x} = \boldsymbol{p}$.

In the following we describe the general prescription of the nonlinear PH matrix method illustrated above. In this prescription, q is assumed to be a prime number. Let $y_1, \ldots, y_k$ and $z_1, \ldots, z_k$ be variables. We first choose a $k \times n$ matrix $M'(\boldsymbol{y})$ with entries in $\mathbf{F}_q[\boldsymbol{y}]$, $n \times k$ matrices $S'(\boldsymbol{z})$ and $N'(\boldsymbol{z})$ with entries in $\mathbf{F}_q[\boldsymbol{z}]$ so as to satisfy that (i) $M'(\boldsymbol{y})S'(\boldsymbol{z}) = \mathrm{diag}(y_{i_1}z_{i_1}, \ldots, y_{i_k}z_{i_k})$, where $i_1, \ldots . i_k \in \{1, \ldots, k\}$, and (ii) $M'(\boldsymbol{y})N'(\boldsymbol{z}) = 0$. In the above example, these matrices are chosen by

$$M'(\boldsymbol{y}) = \begin{pmatrix} y_1 & 0 & 0 & 0 \\ 0 & y_1 & 0 & 0 \\ 0 & 0 & y_2 & 0 \end{pmatrix}, \quad S'(\boldsymbol{z}) = \begin{pmatrix} z_1 & 0 & 0 \\ 0 & z_1 & 0 \\ 0 & 0 & z_2 \\ z_2 & 0 & -z_1 \end{pmatrix}, \quad \text{and} \quad N'(\boldsymbol{z}) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 3z_1 & 0 & z_2 \end{pmatrix}.$$

We also choose an invertible $k \times k$ matrix C and an invertible $n \times n$ matrix D, where the entries of both C and D are in $\mathbf{F}_q$. We then set the nonlinear PH matrix $M(\boldsymbol{x})$ and the nonlinear function $F_0(\boldsymbol{u})$ by the following, respectively:

$$M(\boldsymbol{x}) = C^{-1}M'(\boldsymbol{y})D^{-1}, \tag{28}$$

$$F_0(\boldsymbol{u}) = DS'(\boldsymbol{z})C\,\mathrm{diag}(f_1(u_2,\ldots,u_k),\ldots,f_{k-1}(u_k),1)\boldsymbol{u} + DN'(\boldsymbol{z})g(\boldsymbol{x}), \tag{29}$$

where $\boldsymbol{y}$ and $\boldsymbol{z}$ are set to be $y_i = a_i(\boldsymbol{x})^{p-2}$ and $z_i = a_i(\boldsymbol{x})$, $f_1(u_2,\ldots,u_k),\ldots,f_{k-1}(u_k)$ are in $\mathbf{F}_q[\boldsymbol{u}]$, and $g(\boldsymbol{x})$ is a column vector which has k components in $\mathbf{F}_q[\boldsymbol{x}]$. Note that $\boldsymbol{u}$ is defined to be $A\boldsymbol{x}$ according to the notation in the general form of encryption process (1). In the above example, the matrices C and D are chosen by

$$C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{pmatrix} \quad \text{and} \quad D = \begin{pmatrix} 3 & 2 & 1 & 6 \\ 4 & 3 & 1 & 3 \\ 1 & 5 & 1 & 1 \\ 3 & 3 & 5 & 2 \end{pmatrix}.$$

Since A and B are chosen as the identity matrices in the above example, we obtain (26) and (27).

Let $e_i(\boldsymbol{x})$ be the i-th component of $E(\boldsymbol{x}) = (B \circ F_0 \circ A)(\boldsymbol{x})$. The public key is the pair $(q, G(\boldsymbol{x}))$, where $G(\boldsymbol{x})$ is a system of polynomials obtained by shuffling the sequence $e_1(\boldsymbol{x}),\ldots,e_1(\boldsymbol{x}),a_1(\boldsymbol{x}),\ldots,a_r(\boldsymbol{x})$ of polynomials with respect to their order. The secret key is $(A, B, f_1,\ldots,f_{k-1}, M(\boldsymbol{x}))$ together with the information on the shuffling of $e_i(\boldsymbol{x})$'s and $a_i(\boldsymbol{x})$'s. Using the public key, Alice encrypts a plain text vector $\boldsymbol{p}$ to generate the corresponding cipher text vector $\boldsymbol{c} = G(\boldsymbol{p})$, and then sends $\boldsymbol{c}$ to Bob. On receiving $\boldsymbol{c}$, Bob first isolates individual $e_i(\boldsymbol{p})$'s and $a_i(\boldsymbol{p})$'s based on the information on the shuffling. Bob then calculates $M(\boldsymbol{p})$ using the values of $a_i(\boldsymbol{p})$'s, and multiplies $M(\boldsymbol{p})B^{-1}$ to $\boldsymbol{c}$. As a result, in the case where each of $a_i(\boldsymbol{p})$'s is not zero, Bob knows the values of $f_1(s_2,\ldots,s_k)s_1,\ldots,f_{k-1}(s_k)s_{k-1}$ and s_k, as in the above example. Here the column vector $\boldsymbol{s} = (s_1,\ldots,s_k)^T$ is defined to be $A\boldsymbol{p}$. This is because, by (28), (29), and the Fermat's little theorem,

$$\begin{aligned} M(\boldsymbol{p})B^{-1}\boldsymbol{c} &= M(\boldsymbol{p})F_0(\boldsymbol{s}) \\ &= C^{-1}\,\mathrm{diag}(a_{i_1}(\boldsymbol{p})^{q-1},\ldots,a_{i_k}(\boldsymbol{p})^{q-1})\,C\,\mathrm{diag}(f_1(s_2,\ldots,s_k),\ldots,f_{k-1}(s_k),1)\boldsymbol{s} \\ &= (f_1(s_2,\ldots,s_k)s_1,\ldots,f_{k-1}(s_k)s_{k-1},s_k)^T \end{aligned}$$

in such a case. Thus, Bob obtains the value of $\boldsymbol{s}$ using the sequential solution method and therefore the plain text vector $\boldsymbol{p}$ by multiplying A^{-1} to $\boldsymbol{s}$.

6 Concluding Remarks

In this paper, we have introduced a new concept, piece in hand (PH), for public key cryptosystems in general, and have proposed the framework of the PH concept where the security of a wide class of multivariate type public key cryptosystems can be enhanced by the concept. In contrast to most traditional public key cryptosystems such as the RSA and ElGamal schemes, in our scheme based on the PH concept a certain part of the secret key, PH matrix, is not contained in the public key but is taken from outside of the public key to plug in during the decryption. The applications of the PH concept to multivariate type public key cryptosystems have been illustrated in both integrated (Section 2, 3) and sequential (Section 4) manners. From the practical point of view, it is important to evaluate the key length and the efficiency of encryption and decryption in the enhanced cryptosystem. However, since the aim of the present paper is mainly to illustrate the use of the PH concept, this issue is discussed in another paper. Because of the same reason, we have not considered the stronger security such as IND-CCA type security but considered just the encryption primitive $E(\boldsymbol{x})$ for a multivariate type public key cryptosystem whose security is enhanced by the PH concept. We leave the consideration of the stronger security to a future study.

Acknowledgments

The authors are grateful to Professor Adi Shamir, Mr. Gwénolé Ars, and Mr. Makoto Sugita for discussions and helpful comments.

References

[1] N. Courtois, A. Klimov, J. Patarin, and A. Shamir. Efficient algorithms for solving overdefined systems of multivariate polynomial equations. *Proc. EUROCRYPT 2000*, Lecture Notes in Computer Science, Vol.1807, pp.392–407, Springer, 2000.

[2] J. C. Faugère and A. Joux. Algebraic cryptanalysis of hidden field equation (HFE) cryptosystems using Gröbner bases. *Proc. CRYPTO 2003*, Lecture Notes in Computer Science, Vol.2729, pp.44–60, Springer, 2003.

[3] H. Imai and T. Matsumoto. Algebraic methods for constructing asymmetric crypto-systems. *Proc. AAECC-3*, Lecture Notes in Computer Science, Vol.229, pp.108–119, Springer, 1985.

[4] M. Kasahara and R. Sakai. A construction of public key cryptosystem for realizing ciphertext of size 100 bit and digital signature scheme. *IEICE Transactions Fundamentals*, E87-A, No.1 (2004), 102–109.

[5] M. Kasahara and R. Sakai. A construction of public-key cryptosystem based on singular simultaneous equations. *Proc. SCIS2004*, 1B5-1, pp.155–160, 2004.

[6] A. Kipnis and A. Shamir. Cryptanalysis of the HFE public key cryptosystem by relinearization. *Proc. CRYPTO '99*, Lecture Notes in Computer Science, Vol.1666, pp.19–30, Springer, 1999.

[7] T. Matsumoto, H. Imai, H. Harashima, and H. Miyakawa. A class of asymmetric cryptosystems using obscure representations of enciphering functions. *1983 National Convention Record on Information Systems, IECE Japan*, S8-5, 1983. In Japanese.

[8] T. Matsumoto and H. Imai. Public quadratic polynomial-tuples for efficient signature-verification and message-encryption. *Proc. EUROCRYPT '88*, Lecture Notes in Computer Science, Vol.330, pp.419–453, Springer, 1988.

[9] T. T. Moh. A public key system with signature and master key functions. *Communications in Algebra*, 27, 2207–2222, 1999.

[10] J. Patarin. Hidden fields equations (HFE) and isomorphisms of polynomials (IP): two new families of asymmetric algorithms. *Proc. EUROCRYPT '96*, Lecture Notes in Computer Science, Vol.1070, pp.33–48, Springer, 1996.

[11] S. Tsujii, K. Kurosawa, T. Itoh, A. Fujioka, and T. Matsumoto. A public-key cryptosystem based on the difficulty of solving a system of non-linear equations. *IEICE Transactions* (D), J69-D, No.12 (1986), 1963–1970. In Japanese.

[12] S. Tsujii, A. Fujioka, and Y. Hirayama. Generalization of the public-key cryptosystem based on the difficulty of solving a system of non-linear equations. *IEICE Transactions* (A), J72-A, No.2 (1989), 390–397. In Japanese.

[13] S. Tsujii. A new structure of primitive public key cryptosystem based on soldiers in hand matrix. Technical Report TRISE 02-03, Chuo University, July 2003.

[14] S. Tsujii, R. Fujita, and K. Tadaki. Proposal of MOCHIGOMA (piece in hand) concept for multivariate type public key cryptosystem. Technical Report of IEICE, ISEC2004-74 (2004-09), September 2004.

Appendix

Generalization of the Public-Key Cryptosystem Based on the Difficulty of Solving a System of Non-linear Equations

Shigeo Tsujii Atsushi Fujioka Yuusuke Hirayama

This is an English version of the Japanese paper which appeared in *IEICE Transactions* (A), J72-A, No.2 (1989), 390–397. The English translation is by Shigeo Tsujii, Kohtaro Tadaki, and Ryou Fujita.

Generalization of the Public-Key Cryptosystem Based on the Difficulty of Solving a System of Non-linear Equations*

Shigeo Tsujii† Atsushi Fujioka† Yuusuke Hirayama†

† Faculty of Engineering, Tokyo Institute of Technology, Tokyo, 152 Japan

Abstract. In the previous work we proposed a public-key cryptosystem based on the sequential solution method for a system of non-linear equations. In this paper we present a new public-key cryptosystem where the sequential solution method is generalized and multivariate rational expressions are used. This cryptosystem has high reliability against not only attacks made on the previous proposal but also others thinkable currently. Moreover, it makes the lengths of public and secret keys relatively short and features fast encryption and decryption.

1 Introduction

Public-key cryptosystems are useful for secret communications and also are fundamental technologies indispensable to authentication. In the last ten years, more than ten such cryptosystems have been proposed. Many of them are based on the difficulty of prime factorization or discrete logarithm problems and require exponentiation for encryption and decryption. The well-known RSA scheme is being incorporated into LSI circuits but it is said that the processing speed is now limited to hundreds of kilobytes per second (Kb/s) in the case where a CMOS is used.

In the future, increased demand for cryptographic communications via faster transmission lines is expected. Therefore, it is very important to find a public-key cryptosystem having high-speed encryption and decryption.

Moreover, it seems necessary to pursue a public-key cryptosystem that is not based on the difficulty of prime factorization or discrete logarithm problems because there is no guarantee that these problems can never be solved.

We continue to study new public-key cryptosystems with this in mind. Hasegawa and Kaneko [2] presented a way to break our previous proposal, a public-key cryptosystem [1] based on the sequential solution method for a system of non-linear equations.

In this paper, we propose a new public-key cryptosystem [4] where the sequential solution method is generalized and the core transformations are used. This cryptosystem has high reliability against not only attacks made on the previous proposal but also others thinkable currently. Moreover, it makes the lengths of public and secret keys relatively short and features fast encryption and decryption.

The organization of the paper is as follows. In Section 2 we define the core transformation and sequential solution method. In Section 3 we present our cryptosystem and consider its reliability in Section 4. In Section 5 we describe the resulting properties and finally, in Section 6, we summarize this work.

2 Core Transformation and Sequential Solution Method

This section describes the core transformation and sequential solution method which are used as trapdoor in decryption.

*This is an English version of the Japanese paper which appeared in *IEICE Transactions* (A), J72-A, No.2 (1989), 390–397. The English translation is by Shigeo Tsujii, Kohtaro Tadaki, and Ryou Fujita.

2.1 Core Transformation [3]

In this subsection, as an example, we present a transformation based on multivariate rational expressions.

Let $\boldsymbol{v}$ and $\boldsymbol{w}$ be as follows:

$$\boldsymbol{v} = (v_1, v_2, \cdots, v_k)^T, \tag{1}$$
$$\boldsymbol{w} = (w_1, w_2, \cdots, w_k)^T \tag{2}$$

where T denotes transposition.

The relations between w_k and w_{k-1} as well as between v_k and v_{k-1} are defined by the following two equations:

$$v_k = \frac{\alpha_1 w_k + \alpha_2 w_{k-1} + \alpha_3}{\alpha_4 w_k + \alpha_5 w_{k-1} + \alpha_6}, \tag{3}$$
$$v_{k-1} = \frac{\beta_1 w_k + \beta_2 w_{k-1} + \beta_3}{\beta_4 w_k + \beta_5 w_{k-1} + \beta_6}. \tag{4}$$

Using the following equations:

$$\Delta_{10}(v_{k-1}, v_k) = \begin{vmatrix} \alpha_4 v_k - \alpha_1 & \alpha_5 v_k - \alpha_2 \\ \beta_4 v_{k-1} - \beta_1 & \beta_5 v_{k-1} - \beta_2 \end{vmatrix}, \tag{5}$$
$$\Delta_{11}(v_{k-1}, v_k) = \begin{vmatrix} \alpha_3 - \alpha_6 v_k & \alpha_5 v_k - \alpha_2 \\ \beta_3 - \beta_6 v_{k-1} & \beta_5 v_{k-1} - \beta_2 \end{vmatrix}, \tag{6}$$
$$\Delta_{12}(v_{k-1}, v_k) = \begin{vmatrix} \alpha_4 v_k - \alpha_1 & \alpha_3 - \alpha_6 v_k \\ \beta_4 v_{k-1} - \beta_1 & \beta_3 - \beta_6 v_{k-1} \end{vmatrix}, \tag{7}$$

we can reverse the equations (3) and (4) as

$$w_k = \frac{\Delta_{11}(v_{k-1}, v_k)}{\Delta_{10}(v_{k-1}, v_k)}, \tag{8}$$
$$w_{k-1} = \frac{\Delta_{12}(v_{k-1}, v_k)}{\Delta_{10}(v_{k-1}, v_k)}. \tag{9}$$

From the equations above, the following expressions, for example, are derived:

$$\begin{pmatrix} w_{k-1} \\ w_k \end{pmatrix} = M(v_{k-1}, v_k) \cdot \begin{pmatrix} v_{k-1} \\ v_k \end{pmatrix} \tag{10}$$
$$M(v_{k-1}, v_k) = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix} \tag{11}$$
$$m_{11} = \frac{(\alpha_1\beta_6 - \alpha_3\beta_4)v_{k-1} + \alpha_6\beta_4 v_{k-1}v_k + \alpha_3\beta_1}{v_{k-1} \cdot \Delta_{10}(v_{k-1}, v_k)} \tag{12}$$
$$m_{12} = \frac{(\alpha_4\beta_3 - \alpha_6\beta_1)v_k - \alpha_4\beta_6 v_{k-1}v_k - \alpha_1\beta_3}{v_k \cdot \Delta_{10}(v_{k-1}, v_k)} \tag{13}$$
$$m_{21} = \frac{(\alpha_3\beta_5 - \alpha_2\beta_6)v_{k-1} - \alpha_6\beta_5 v_{k-1}v_k + \alpha_2\beta_3}{v_{k-1} \cdot \Delta_{10}(v_{k-1}, v_k)} \tag{14}$$
$$m_{22} = \frac{(\alpha_6\beta_2 - \alpha_5\beta_3)v_k + \alpha_5\beta_6 v_{k-1}v_k - \alpha_3\beta_2}{v_k \cdot \Delta_{10}(v_{k-1}, v_k)}. \tag{15}$$

Note that these expressions are not unique. We call $M(v_{k-1}, v_k)$ a core transformation.

2.2 Combination of the Core Transformation and Sequential Solution Method

In this subsection we describe the sequential solution method [1] based on the core transformation presented in the previous subsection.

Based on the equations (11) to (15) we define the matrix $F(\boldsymbol{v})$ by

$$F(\boldsymbol{v}) = \begin{pmatrix} f_{11} & 0 & \cdots & \cdots & \cdots & 0 & & 0 \\ 0 & f_{22} & 0 & \cdots & \cdots & 0 & & 0 \\ \vdots & 0 & \ddots & \ddots & & \vdots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & & \vdots \\ \vdots & \vdots & & \ddots & \ddots & 0 & & 0 \\ 0 & 0 & \cdots & \cdots & 0 & & & \\ & & & & & & M(v_{k-1}, v_k) & \\ 0 & 0 & \cdots & \cdots & 0 & & & \end{pmatrix}. \tag{16}$$

Here, $f_{ii}v_i$ $(i = 1, \cdots, k-2)$ is a linear function with respect to v_i and a non-linear function with respect to $v_{i+1}, \cdots, v_k$, and is constructed so that the equation

$$\boldsymbol{w} = F(\boldsymbol{v}) \cdot \boldsymbol{v} \tag{17}$$

on $\boldsymbol{v}$ can be solved variable by variable in the order of $v_{k-2}, v_{k-3}, \cdots, v_2$, and v_1.

Concretely, it is defined as follows:

$$f_{ii} = \frac{a_i(\boldsymbol{v}) \cdot v_i + b_i(\boldsymbol{v})}{v_i \cdot \Delta_{10}(v_{k-1}, v_k)} \quad (i = 1, \cdots, k-2) \tag{18}$$

where $a_i(\boldsymbol{v})$ is a linear function with respect to $v_{i+1}, \cdots, v_k$ and $b_i(\boldsymbol{v})$ is a quadratic function with respect to $v_{i+1}, \cdots, v_k$.

Thus, in the system (17) of non-linear equations, v_k and v_{k-1} are first derived from w_k and w_{k-1}, and then the remainder $v_{k-2}, v_{k-3}, \cdots, v_2, v_1$ are obtained sequentially. Hence we have all components of $\boldsymbol{v}$. We call this method a sequential solution method.

3 The Construction of Our Cryptosystem

[Notation]

$$\boldsymbol{x} = (x_1, x_2, \cdots, x_k)^T : \text{Plaintext vector} \tag{19}$$
$$\boldsymbol{y} = (y_1, y_2, \cdots, y_k)^T : \text{Ciphertext vector} \tag{20}$$
$$\boldsymbol{v} = (v_1, v_2, \cdots, v_k)^T : \text{Intermediate variable vector} \tag{21}$$
$$\boldsymbol{w} = (w_1, w_2, \cdots, w_k)^T : \text{Intermediate variable vector} \tag{22}$$
$$\boldsymbol{u} = (u_1, u_2, \cdots, u_k)^T : \text{Intermediate variable vector} \tag{23}$$
$$\boldsymbol{z} = (z_1, z_2, \cdots, z_k)^T : \text{Intermediate variable vector} \tag{24}$$

(Here T denotes transposition.)

A, B, and C: $k \times k$ invertible matrices.

$F(\boldsymbol{v})$ and $G(\boldsymbol{u})$: Matrices that enable the sequential solution method with the core transformation.

First of all, the core transformation $M(v_{k-1}, v_k)$ is defined by the equations (11) to (15) and in the same manner, $N(u_{k-1}, u_k)$ is defined by the following equations:

$$u_k = \frac{\gamma_1 z_k + \gamma_2 z_{k-1} + \gamma_3}{\gamma_4 z_k + \gamma_5 z_{k-1} + \gamma_6}, \tag{25}$$

$$u_{k-1} = \frac{\delta_1 z_k + \delta_2 z_{k-1} + \delta_3}{\delta_4 z_k + \delta_5 z_{k-1} + \delta_6}, \tag{26}$$

$$\Delta_{20}(u_{k-1}, u_k) = \begin{vmatrix} \gamma_4 u_k - \gamma_1 & \gamma_5 u_k - \gamma_2 \\ \delta_4 u_{k-1} - \delta_1 & \delta_5 u_{k-1} - \delta_2 \end{vmatrix}, \tag{27}$$

$$\Delta_{21}(u_{k-1}, u_k) = \begin{vmatrix} \gamma_3 - \gamma_6 u_k & \gamma_5 u_k - \gamma_2 \\ \delta_3 - \delta_6 u_{k-1} & \delta_5 u_{k-1} - \delta_2 \end{vmatrix}, \tag{28}$$

$$\Delta_{22}(u_{k-1}, u_k) = \begin{vmatrix} \gamma_4 u_k - \gamma_1 & \gamma_3 - \gamma_6 u_{k-1} \\ \delta_4 u_{k-1} - \delta_1 & \delta_3 - \delta_6 u_{k-1} \end{vmatrix}, \tag{29}$$

$$z_k = \frac{\Delta_{21}(u_{k-1}, u_k)}{\Delta_{20}(u_{k-1}, u_k)}, \tag{30}$$

$$z_{k-1} = \frac{\Delta_{22}(u_{k-1}, u_k)}{\Delta_{20}(u_{k-1}, u_k)}, \tag{31}$$

$$\begin{pmatrix} z_{k-1} \\ z_k \end{pmatrix} = N(u_{k-1}, u_k) \cdot \begin{pmatrix} u_{k-1} \\ u_k \end{pmatrix}. \tag{32}$$

Using the core transformations $M(v_{k-1}, v_k)$ and $N(u_{k-1}, u_k)$, we define $F(\boldsymbol{v})$ by the equations (16) and (18) and define $G(\boldsymbol{u})$ by the following:

$$G(\boldsymbol{u}) = \begin{pmatrix} g_{11} & 0 & \dots & \dots & \dots & 0 & & 0 \\ 0 & g_{22} & 0 & \dots & \dots & 0 & & 0 \\ \vdots & 0 & \ddots & \ddots & & \vdots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & & \vdots \\ \vdots & \vdots & & \ddots & \ddots & 0 & & 0 \\ 0 & 0 & \dots & \dots & 0 & & & \\ & & & & & & N(u_{k-1}, u_k) & \\ 0 & 0 & \dots & \dots & 0 & & & \end{pmatrix}, \tag{33}$$

$$g_{ii} = \frac{c_i(\boldsymbol{u}) \cdot u_i + d_i(\boldsymbol{u})}{u_i \cdot \Delta_{20}(u_{k-1}, u_k)} \quad (i = 1, \cdots, k-2) \tag{34}$$

where $c_i(\boldsymbol{u})$ is a linear function with respect to $u_{i+1}, \cdots, u_k$ and $d_i(\boldsymbol{u})$ is a quadratic function with respect to $u_{i+1}, \cdots, u_k$.

We here assume that the components of each vector are in an extension field of degree t of GF(2), and the components of each matrix and the coefficients of each rational expression are in an extension field of degree s of GF(2) where $s|t$, i.e., a subfield of the former field.

We assume that the following relations hold:

$$\begin{aligned} \boldsymbol{v} &= A \cdot \boldsymbol{x}, && (35)\\ \boldsymbol{w} &= F(\boldsymbol{v}) \cdot \boldsymbol{v}, && (36)\\ \boldsymbol{u} &= B \cdot \boldsymbol{w}, && (37)\\ \boldsymbol{z} &= G(\boldsymbol{u}) \cdot \boldsymbol{u}, && (38)\\ \boldsymbol{y} &= C \cdot \boldsymbol{z}, && (39)\\ R(\boldsymbol{x}) &= C \cdot G(\boldsymbol{u}) \cdot B \cdot F(\boldsymbol{v}) \cdot A \cdot \boldsymbol{x}. && (40) \end{aligned}$$

By the transformation in the first stage, the plaintext $\boldsymbol{x}$ is transformed to $\boldsymbol{u}$. Accordingly, $u_1, \cdots, u_k$ are the quadratic rational expressions of $x_1, \cdots, x_k$ and are represented as

$$\begin{aligned} u_1 &= \frac{p_1(\boldsymbol{x})}{p_d(\boldsymbol{x})},\\ u_2 &= \frac{p_2(\boldsymbol{x})}{p_d(\boldsymbol{x})}, && (41)\\ &\vdots\\ u_k &= \frac{p_k(\boldsymbol{x})}{p_d(\boldsymbol{x})}, \end{aligned}$$

where $p_i(\boldsymbol{x})$ $(i = 1, \cdots, k)$ and $p_d(\boldsymbol{x})$ are quadratic functions with respect to $x_1, \cdots, x_k$.

Next, $\boldsymbol{u}$ generated in the first stage is transformed to the ciphertext $\boldsymbol{y}$ in the same manner as the transformation in the first stage. Accordingly, $y_1, \cdots, y_k$ are the quadratic rational expressions of $u_1, \cdots, u_k$ and are represented as

$$\begin{aligned} y_1 &= \frac{q_1(\boldsymbol{u})}{q_d(\boldsymbol{u})},\\ y_2 &= \frac{q_2(\boldsymbol{u})}{q_d(\boldsymbol{u})}, && (42)\\ &\vdots\\ y_k &= \frac{q_k(\boldsymbol{u})}{q_d(\boldsymbol{u})}, \end{aligned}$$

where $q_i(\boldsymbol{u})$ $(i = 1, \cdots, k)$ and $q_d(\boldsymbol{u})$ are quadratic functions with respect to $u_1, \cdots, u_k$. By taking the form of $\boldsymbol{u}$ in the equations (41) into consideration, substituting $u_1, \cdots, u_k$ of the equations (41) into the equations (42), and then multiplying the numerators and denominators by $p_d^2(\boldsymbol{x})$, we see that the ciphertext $\boldsymbol{y}$ is quartic rational expressions of $x_1, \cdots, x_k$ as follows:

$$\begin{aligned} y_1 &= \frac{r_1(\boldsymbol{x})}{r_d(\boldsymbol{x})},\\ y_2 &= \frac{r_2(\boldsymbol{x})}{r_d(\boldsymbol{x})}, && (43)\\ &\vdots\\ y_k &= \frac{r_k(\boldsymbol{x})}{r_d(\boldsymbol{x})}, \end{aligned}$$

where $r_i(\boldsymbol{x})$ $(i = 1, \cdots, k)$ and $r_d(\boldsymbol{x})$ are quartic functions with respect to $x_1, \cdots, x_k$.

The above shows the construction of the ciphertext $\boldsymbol{y}$ based on the two-stage construction.

⟨**Secret Key**⟩

$$A, B, C, F(\boldsymbol{v}), \text{ and } G(\boldsymbol{u})$$

⟨**Public Key**⟩

$$\boldsymbol{y} = R(\boldsymbol{x}) \tag{44}$$

($\boldsymbol{y}$ is a non-linear function of $\boldsymbol{x}$.)

Encryption

Substitute a plaintext $\boldsymbol{x}$ into the public key.

Decryption

⟨1⟩ $\boldsymbol{z}$ is derived from $\boldsymbol{z} = C^{-1} \cdot \boldsymbol{y}$.

⟨2⟩ $\boldsymbol{u}$ is derived from $\boldsymbol{z} = G(\boldsymbol{u}) \cdot \boldsymbol{u}$.

⟨3⟩ $\boldsymbol{w}$ is derived from $\boldsymbol{w} = B^{-1} \cdot \boldsymbol{u}$.

⟨4⟩ $\boldsymbol{v}$ is derived from $\boldsymbol{w} = F(\boldsymbol{v}) \cdot \boldsymbol{v}$.

⟨5⟩ $\boldsymbol{x}$ is derived from $\boldsymbol{x} = A^{-1} \cdot \boldsymbol{v}$.

4 Consideration of Reliability

Attacks made to date on public-key cryptosystems based on the sequential solution method for a system of non-linear equations are listed below.

1) Okamoto and Nakamura's attack [5]

2) Kurosawa's attack [1]

3) Hasegawa and Kaneko's attack [2]

Attack 1) is used for cryptosystems in which a public key is represented as a polynomial or rational equation, like our cryptosystem, and is made by solving a system of multivariate linear equations to find the inverse function of the cryptosystem. We call this the inversion method.

The other attacks can be categorized under one method. They assume that the coefficients of a secret key are unknown variables and create a public key. This public key is then compared with the coefficients of the public key actually published to obtain simultaneous equations with respect to the coefficients of the secret key. We call this the coefficient equivalence method. The reason the cryptosystem proposed in the previous work could be broken is that the resulting simultaneous equations are linear ones from which the secret key can be found easily.

In this section we describe the reliability of our cryptosystem against both the attacks.

Our cryptosystem assumes the following parameter values:

$$k = 5, \tag{45}$$

$$t = 32, \tag{46}$$

$$s = 8. \tag{47}$$

4.1 Attack by the Coefficient Equivalence Method

In this subsection we describe the reliability of our cryptosystem against the coefficient equivalence method. The reliability against this attack depends on the core transformation.

4.1.1 Determination of $p_{k-1}(\boldsymbol{x}), p_k(\boldsymbol{x})$, and $p_d(\boldsymbol{x})$

In the equation (41), $p_i(\boldsymbol{x})$ and $p_d(\boldsymbol{x})$ are quadratic polynomials of $x_1, \cdots, x_k$. Thus, regarding the coefficients of these polynomials as unknown variables, we first represent the polynomials as

$$\begin{pmatrix} p_1(\boldsymbol{x}) \\ p_2(\boldsymbol{x}) \\ \vdots \\ p_k(\boldsymbol{x}) \\ p_d(\boldsymbol{x}) \end{pmatrix} = \begin{pmatrix} p_{11} & \cdots & \cdots & \cdots & p_{1n} \\ p_{21} & \cdots & \cdots & \cdots & p_{1n} \\ & & \vdots & & \\ p_{k1} & \cdots & \cdots & \cdots & p_{kn} \\ p_{d1} & \cdots & \cdots & \cdots & p_{dn} \end{pmatrix} \cdot \widetilde{\boldsymbol{x}} \tag{48}$$

$$\widetilde{\boldsymbol{x}} = \begin{pmatrix} x_1^2 & x_1x_2 & \cdots & x_{k-1}x_k & x_k^2 & x_1 & \cdots & x_k & 1 \end{pmatrix}^T \tag{49}$$

where n is the number of terms contained in a quadratic polynomial having k variables, and we define

$$\mathrm{Term}_{k,2} \;=\; {}_{k+2}\mathrm{C}_k\,. \tag{50}$$

On the other hand, the denominator $q_d(\boldsymbol{u})$ of $\boldsymbol{y}$ in the equations (42) is equal to $\Delta_{20}(u_{k-1}, u_k)$ defined by the equation (27). Thus we have

$$\begin{aligned} q_d(\boldsymbol{u}) &= \Delta_{20}(u_{k-1}, u_k) \\ &= (\gamma_4\delta_5 - \gamma_5\delta_4)u_{k-1}u_k - (\gamma_4\delta_2 - \gamma_5\delta_1)u_k \\ &\quad -(\gamma_1\delta_5 - \gamma_2\delta_4)u_{k-1} + (\gamma_1\delta_2 - \gamma_2\delta_1). \end{aligned} \tag{51}$$

Substituting the equations (41) into u_{k-1} and u_k in this equation gives

$$\begin{aligned} q_d(\boldsymbol{u}) &= q'_d(\boldsymbol{x}) \\ &= (\gamma_4\delta_5 - \gamma_5\delta_4)\frac{p_{k-1}(\boldsymbol{x})p_k(\boldsymbol{x})}{p_d^2(\boldsymbol{x})} \\ &\quad -(\gamma_4\delta_2 - \gamma_5\delta_1)\frac{p_k(\boldsymbol{x})}{p_d(\boldsymbol{x})} \\ &\quad -(\gamma_1\delta_5 - \gamma_2\delta_4)\frac{p_{k-1}(\boldsymbol{x})}{p_d(\boldsymbol{x})} \\ &\quad +(\gamma_1\delta_2 - \gamma_2\delta_1). \end{aligned} \tag{52}$$

Moreover, since $r_d(\boldsymbol{x})$ in the equations (43) is given by multiplying $q'_d(\boldsymbol{x})$ in the equation (52) by $p_d^2(\boldsymbol{x})$, it can be represented as

$$\begin{aligned} r_d(\boldsymbol{x}) &= q'_d(\boldsymbol{x})p_d^2(\boldsymbol{x}) \\ &= (\gamma_4\delta_5 - \gamma_5\delta_4)p_{k-1}(\boldsymbol{x})p_k(\boldsymbol{x}) \\ &\quad -(\gamma_4\delta_2 - \gamma_5\delta_1)p_k(\boldsymbol{x})p_d(\boldsymbol{x}) \\ &\quad -(\gamma_1\delta_5 - \gamma_2\delta_4)p_{k-1}(\boldsymbol{x})p_d(\boldsymbol{x}) \\ &\quad +(\gamma_1\delta_2 - \gamma_2\delta_1)p_d^2(\boldsymbol{x}). \end{aligned} \tag{53}$$

Thus, substituting $p_{k-1}(\boldsymbol{x}), p_k(\boldsymbol{x})$, and $p_d(\boldsymbol{x})$ in the equation (48) into the equation above, we see that $r_d(\boldsymbol{x})$ can be represented as a quartic polynomial of $x_1, \cdots, x_k$ whose coefficients are quartic polynomials of $\gamma_1, \gamma_2, \gamma_4, \gamma_5$; $\delta_1, \delta_2, \delta_4, \delta_5$; $p_{k-11}, \cdots, p_{k-1n}$; $p_{k1}, \cdots, p_{kn}$; $p_{d1}, \cdots, p_{dn}$.

Assuming that the coefficients above are equal to the coefficients of the polynomial of the denominator of the public key, we have a system of non-linear equations. In this system of non-linear equations, the number of unknown variables is given by

$$4 + 4 + 3 \times n \quad = \quad 8 + \frac{3(k+2)(k+1)}{2}. \tag{54}$$

Since $r_d(\boldsymbol{x})$ is a quartic polynomial having k variables, the number of terms $\mathrm{Term}_{k,4}$ is given by

$$\mathrm{Term}_{k,4} \quad = \quad {}_{k+4}\mathrm{C}_k = \frac{(k+4)(k+3)(k+2)(k+1)}{4 \cdot 3 \cdot 2 \cdot 1}. \tag{55}$$

Subtracting the equation (54) from the equation (55) gives

$$(55) - (54) \quad = \quad \frac{1}{24}(k^4 + 10k^3 - k^2 - 58k - 240) > 0 \tag{56}$$

for $k \geq 4$.

In $r_d(\boldsymbol{x})$, comparing the coefficients of the public key with ones represented by the unknown variables $\gamma_i, \delta_i, p_{ij}$, and p_{di} presents $\mathrm{Term}_{k,4}$ quartic equations with respect to $\gamma_i, \delta_i, p_{ij}$, and p_{di}. Thus, there is some possibility of determining $\gamma_i, \delta_i, p_{ij}$, and p_{di} by which $r_d(\boldsymbol{x})$ is represented.

However, if we set e.g. $k = 5$ in the concrete, then

$$\text{(Number of unknown variables)} \quad = \quad 71, \tag{57}$$

$$\text{(Number of equations)} \quad = \quad 126. \tag{58}$$

Thus, in the case of $k = 5$, we have to solve 126 quartic polynomial equations of 71 variables in order to determine $p_{k-1}(\boldsymbol{x}), p_k(\boldsymbol{x}), p_d(\boldsymbol{x})$, and $\Delta_{20}(u_{k-1}, u_k)$.

4.1.2 Determining the Core Transformation $N(u_{k-1}, u_k)$ and the Bottom Two Rows of the Matrix C^{-1}

We assume that $\gamma_i, \delta_i, p_{ij}$, and p_{di} composing $r_d(\boldsymbol{x})$ are already found. Then $p_k(\boldsymbol{x}), p_{k-1}(\boldsymbol{x})$, and $p_d(\boldsymbol{x})$ are determined, and therefore u_k and u_{k-1} are represented by expressions with respect to $\boldsymbol{x}$.

First of all, multiplying the equation (39) by the inverse matrix C^{-1} gives

$$C^{-1}\boldsymbol{y} \quad = \quad \boldsymbol{z}. \tag{59}$$

Let the elements of C^{-1} be denoted as follows:

$$C^{-1} \quad = \quad \begin{pmatrix} c'_{11} & \cdots & c'_{1k} \\ \vdots & \ddots & \vdots \\ c'_{k1} & \cdots & c'_{kk} \end{pmatrix}. \tag{60}$$

We focus on the kth and $(k-1)$th components of the equation (59). Then, by the equations (30) and (31), we have

$$c'_{k1}y_1 + \cdots + c'_{kk}y_k \quad = \quad \frac{\Delta_{21}(u_{k-1}, u_k)}{\Delta_{20}(u_{k-1}, u_k)}, \tag{61}$$

$$c'_{k-11}y_1 + \cdots + c'_{k-1k}y_k \quad = \quad \frac{\Delta_{22}(u_{k-1}, u_k)}{\Delta_{20}(u_{k-1}, u_k)}. \tag{62}$$

Here, unknowns in $\Delta_{21}(u_{k-1}, u_k)$ and $\Delta_{22}(u_{k-1}, u_k)$ are $\gamma_3, \gamma_6, \delta_3$, and δ_6.

Since u_k and u_{k-1} is represented by expressions with respect to $\boldsymbol{x}$, by substituting the public key into y_i in the equations (61) and (62) to obtain simultaneous equations with respect to $\boldsymbol{x}$, we have simultaneous equations with respect to $\gamma_3, \gamma_6, \delta_3, \delta_6$; $c'_{k1}, \cdots, c'_{kk}$; $c'_{k-11}, \cdots, c'_{k-1k}$, which are linear with respect to $c'_{k1}, \cdots, c'_{kk}$; $c'_{k-11}, \cdots, c'_{k-1k}$ and quadratic with respect to $\gamma_3, \gamma_6, \delta_3$, and δ_6. Thus, we can determine the values of these coefficients.

Accordingly, there is some possibility of determining the bottom two rows of C^{-1}, which is the inverse matrix of the secret key C, the core transformation $N(u_{k-1}, u_k)$, and the bottom two components of $\boldsymbol{u}$ represented by the equations (41), based on the procedures shown in sections 4.1.1 and 4.1.2.

4.1.3 Determining $g_{i,i}$ and C^{-1}

Hereafter we assume that the above have been determined. Starting from the $(k-2)$th row of $N(\boldsymbol{u})$, every row are determined sequentially as follows.

We first note that, from the equation (59), z_{k-2} can be represented as a linear function of $c'_{k-2,1}, \cdots, c'_{k-2,k}$, which are the elements in the $(k-2)$th row of C^{-1}.

On the other hand, using the equation (34), z_{k-2} is rewritten as

$$\begin{aligned} z_{k-2} &= g_{k-2,k-2} u_{k-2} \\ &= \frac{c_{k-2}(\boldsymbol{u}) \cdot u_{k-2} + d_{k-2}(\boldsymbol{u})}{\Delta_{20}(u_{k-1}, u_k)}. \end{aligned} \tag{63}$$

Since it is assumed that $\Delta_{20}(u_{k-1}, u_k)$ has been found through the procedures shown in Section 4.1.1, the unknowns in the equation (63) consist of three in $c_{k-2}(\boldsymbol{u})$, $\mathrm{Term}_{2,2} = {}_{2+2}\mathrm{C}_2$ in $d_{k-2}(\boldsymbol{u})$, and $\mathrm{Term}_{k,2} = {}_{k+2}\mathrm{C}_2$ in z_{k-2}. In the same manner as the above, by rewriting both z_{k-2} represented by $c'_{k-2,1}, \cdots, c'_{k-2,k}$ and z_{k-2} represented by the equation (63) as an expression with respect to $\boldsymbol{x}$ and then comparing the respective coefficients, we have $\mathrm{Term}_{k,4} = {}_{k+4}\mathrm{C}_4$ simultaneous equations, which have degree two with respect to unknown variables.

$$\begin{aligned} (\text{Number of unknown variables}) &= k + 3 + {}_{2+2}\mathrm{C}_2 + {}_{k+2}\mathrm{C}_2 \\ &= \frac{k^2 + 5k + 20}{2}, \end{aligned} \tag{64}$$

$$\begin{aligned} (\text{Number of equations}) &= {}_{k+4}\mathrm{C}_4 \\ &= \frac{(k+4)(k+3)(k+2)(k+1)}{4 \cdot 3 \cdot 2 \cdot 1}, \end{aligned} \tag{65}$$

$$(65) - (64) = \frac{1}{24}(k^4 + 10k^3 + 23k^2 - 10k - 216) > 0 \tag{66}$$

if $k \geq 3$.

Thus, since the number of equations is greater than the number of unknown variables, there is some possibility of determining the values of the unknown variables.

If we set e.g. $k = 5$ in the concrete, then the values of (64) and (65) are 35 and 126, respectively, and therefore 126 quadratic polynomial equations of 35 variables have to be solved in order to determine g_{k-2k-2} and the $(k-2)$th row of C^{-1}.

If the $(k-2)$th row of the second stage has been determined, then each row of $G(\boldsymbol{u})$ can be determined in the order of the $(k-3)$th row, the $(k-4)$th row, $\cdots$, the 1st row in the same manner. The 1st row, which is determined at the last, has the largest number of unknown variables, and the

following hold for it:

$$\begin{aligned}(\text{Number of unknown variables}) &= k + {}_{(k-1)+1}\mathrm{C}_{k-1} + {}_{(k-1)+2}\mathrm{C}_{k-1} + {}_{k+2}\mathrm{C}_{k} \\ &= k^2 + 4k + 1, \qquad (67)\\ (65) - (67) &= \frac{1}{24}(k^4 + 10k^3 + 11k^2 - 46k) > 0 \qquad (68)\end{aligned}$$

if $k \geq 2$. Thus, the number of equations is greater than the number of unknown variables again, and therefore there is some possibility of determining every row.

However, if we set e.g. $k = 5$ in the concrete, then the values of (67) is 46, and therefore 126 quadratic polynomial equations of 46 variables have to be solved in order to determine g_{11} and the first row of C^{-1}.

4.1.4 Reliability against the Coefficient Equivalence Method

Thus, there is some possibility of finding C, g_{ii}, and $N(u_{k-1}, u_k)$ of the second stage represented by the equations (38) and (39), using the procedures shown in the above. However, in the case of $k = 5$, 126 quartic polynomial equations of 71 variables have to be solved in order to break our cryptosystem, as shown in Section 4.1.1. On the other hand, since the public key of our cryptosystem is represented by quartic rational expressions with respect to the plaintext $\boldsymbol{x}$, five quartic equations of five variables have only to be solved in order to obtain the plaintext $\boldsymbol{x}$ directly from the ciphertext $\boldsymbol{y}$ using the public key. Therefore, by comparison, its computational complexity is thought to be smaller than that of the cryptanalysis considered above. Thus our cryptosystem is thought to have sufficient reliability against conventional attacks based on coefficient comparison.

If we set each parameter by the equations (45) to (47), then solving 126 quartic polynomial equations of 71 variables in round-robin fashion requires

$$(2^8)^{71} \simeq 10^{189} \qquad (69)$$

trials.

4.2 Attack by the Inversion Method

In this subsection we examine the reliability of the one-stage cryptosystem against the inversion method rather than our original two-stage cryptosystem for simplicity. If the reliability of the one-stage cryptosystem can be proven, then the reliability of the two-stage cryptosystem is also guaranteed due to the nature of the inversion method.

The difference between our cryptosystem and the previous proposal is the use of the core transformation. However, it is obvious from the construction that a cryptosystem consisting of only the core transformation is vulnerable to attack by the inversion method. Thus the reliability of our cryptosystem against the inversion method depends on the use of the sequential solution method, and the combination of the sequential solution method and core transformation presents high reliability against both the coefficient equivalence and inversion methods.

From the definition of decryption, v_{k-1} and v_k have the following forms:

$$v_{k-1} = \frac{\text{Linear function with respect to } \boldsymbol{w}}{\text{Linear function with respect to } \boldsymbol{w}}, \qquad (70)$$

$$v_k = \frac{\text{Linear function with respect to } \boldsymbol{w}}{\text{Linear function with respect to } \boldsymbol{w}}. \qquad (71)$$

Let dDv_i and dNv_i be the degrees of the polynomials of the denominator and numerator of v_i, respectively. Then we have

$$dDv_k = dNv_k = 1, \tag{72}$$
$$dDv_{k-1} = dNv_{k-1} = 1. \tag{73}$$

Next, from the definition of encryption, w_{k-2} is given by

$$w_{k-2} = \frac{a_{k-2}(\boldsymbol{v}) \cdot v_{k-2} + b_{k-2}(\boldsymbol{v})}{\Delta_{10}(v_{k-1}, v_k)} \tag{74}$$

where $a_{k-2}(\boldsymbol{v})$ is a linear function with respect to v_{k-1} and v_k, $b_{k-2}(\boldsymbol{v})$ is a quadratic function with respect to v_{k-1} and v_k, and $\Delta_{10}(v_{k-1}, v_k)$ is a quadratic function with respect to v_{k-1} and v_k. We here solve the above equation with respect to v_{k-2} to obtain

$$v_{k-2} = \frac{\Delta_{10}(v_{k-1}, v_k) \cdot w_{k-2} - b_{k-2}(\boldsymbol{v})}{a_{k-2}(\boldsymbol{v})}. \tag{75}$$

Next, we rewrite the three rational expressions $a_{k-2}(\boldsymbol{v}), b_{k-2}(\boldsymbol{v})$, and $\Delta_{10}(v_{k-1}, v_k)$ as rational expressions with respect to $\boldsymbol{w}$ and obtain

$$a_{k-2}(\boldsymbol{v}) = \frac{Na_{k-2}(\boldsymbol{w})}{Da_{k-2}(\boldsymbol{w})}, \tag{76}$$
$$b_{k-2}(\boldsymbol{v}) = \frac{Nb_{k-2}(\boldsymbol{w})}{Db_{k-2}(\boldsymbol{w})}, \tag{77}$$
$$\Delta_{10}(v_{k-1}, v_k) = \frac{N\Delta_{10}(\boldsymbol{w})}{D\Delta_{20}(\boldsymbol{w})}. \tag{78}$$

Let dDa_{k-2}, dDb_{k-2}, and $dD\Delta_{10}$ be the degrees of the polynomials of the denominators of the above three expressions, respectively, and let dNa_{k-2}, dNb_{k-2}, and $dN\Delta_{10}$ be those of the numerators, respectively. Then we have

$$dDa_{k-2} = 2, \tag{79}$$
$$dNa_{k-2} = 2, \tag{80}$$
$$dDb_{k-2} = 8, \tag{81}$$
$$dNb_{k-2} = 8, \tag{82}$$
$$dD\Delta_{10} = 8, \tag{83}$$
$$dN\Delta_{10} = 8. \tag{84}$$

Thus, when v_{k-2} is represented as a rational expression with respect to $\boldsymbol{w}$, the degrees dDv_{k-2} and dNv_{k-2} of the polynomials of its denominator and numerator are given respectively by

$$dDv_{k-2} = 18, \tag{85}$$
$$dNv_{k-2} = 19. \tag{86}$$

In the same manner, we solve the following equation with respect to v_i

$$w_i = \frac{a_i(\boldsymbol{v}) \cdot v_i + b_i(\boldsymbol{v})}{\Delta_{10}(v_{k-1}, v_k)} \tag{87}$$

where $a_i(\boldsymbol{v})$ is a linear function with respect to $v_{i+1}, \cdots, v_k$, $b_i(\boldsymbol{v})$ is a quadratic function with respect to $v_{i+1}, \cdots, v_k$, and $\Delta_{20}(v_{k-1}, v_k)$ is a quadratic function with respect to v_{k-1} and v_k. Then we have

$$v_i = \frac{\Delta_{10}(v_{k-1}, v_k) \cdot w_i - b_i(\boldsymbol{v})}{a_i(\boldsymbol{v})}. \tag{88}$$

Next, we rewrite the two rational expressions $a_i(\boldsymbol{v})$ and $b_i(\boldsymbol{v})$ as rational expressions with respect to $\boldsymbol{w}$ and obtain

$$a_i(\boldsymbol{v}) = \frac{Na_i(\boldsymbol{w})}{Da_i(\boldsymbol{w})}, \tag{89}$$

$$b_i(\boldsymbol{v}) = \frac{Nb_i(\boldsymbol{w})}{Db_i(\boldsymbol{w})}. \tag{90}$$

Let dDa_i and dDb_i be the degrees of the polynomials of the denominators of the above two expressions, respectively, and let dNa_i and dNb_i be those of the numerators, respectively. Then we have

$$dDa_i = \sum_{j=i+1}^{k} dDv_j, \tag{91}$$

$$dNa_i = dDav_i + 1, \tag{92}$$

$$dDb_i = \sum_{j=i+1}^{k} 3dDv_j + \sum_{j=i+1}^{k-1} \sum_{n=j+1}^{k-1} (dDv_j + dDv_n), \tag{93}$$

$$dNb_i = dD\Delta_{10}v_i + 1. \tag{94}$$

Thus, when v_i is represented as a rational expression with respect to $\boldsymbol{w}$, the degrees dDv_i and dNv_i of the polynomials of its denominator and numerator are given respectively by

$$dDv_i = 9 + \sum_{j=i+1}^{k} 4dDv_j + \sum_{j=i+1}^{k-1} \sum_{n=j+1}^{k} (dDv_j + dDv_n), \tag{95}$$

$$dNv_i = 10 + \sum_{j=i+1}^{k} 4dDv_j + \sum_{j=i+1}^{k-1} \sum_{n=j+1}^{k} (dDv_j + dDv_n). \tag{96}$$

Since the transformation between the vectors $\boldsymbol{w}$ and $\boldsymbol{y}$ is linear, when the intermediate vector $\boldsymbol{v}$ is represented as a rational expression with respect to the ciphertext $\boldsymbol{y}$ instead of $\boldsymbol{w}$, the degrees of the polynomials of the denominator and numerator do not change.

Finally, by taking the linear combinations of these v_i $(1 \leq i \leq k)$, we see that the degrees of rational expressions representing the plaintext x_i in terms of the ciphertext $\boldsymbol{y}$ are given by

$$dDx_i = \sum_{j=1}^{k} dDv_j, \tag{97}$$

$$dNx_i = dDx_i + 1. \tag{98}$$

Table 1 shows the results given by calculating the above values and counting the total number, Term, of terms in polynomials of the denominator and numerator.

In the above, we have examined the reliability of the one-stage construction. Although our cryptosystem is really proposed based on the two-stage construction at the present time, the degree of its inverse transformation is higher than the values shown in Table 1. Thus, the reliability of our cryptosystem against the inverse method is guaranteed.

k	dDx	Term
4	149	4.448×10^7
5	1201	4.225×10^{13}
6	10818	4.462×10^{21}

Table 1: The number of variables in our cryptosystem and the number of unknown variables in the inverse expression.

5 Properties

Table 2 shows the lengths of the public and secret keys of our cryptosystem when each parameter is set by the equations (45) to (47).

k	s	Public Key (kbit)	Secret Key (kbit)
5	8	6.05	1.48
6	8	11.76	2.18

Table 2: The lengths of the public and secret keys.

6 Conclusions

In this paper, we have proposed the cryptosystem obtained by incorporating the core transformation into a cryptosystem which is based on the sequential solution method for a system of non-linear equations, and have described its properties. Since the encryption and decryption of our cryptosystem are carried out on GF(2^t), their computational complexities are $O(t^2)$. Compared with the previous proposal, the computational complexities in our cryptosystem are a little complex regarding the number k of variables. It is not obvious that the computational complexity for the plaintext of length m is simply $O(m^2)$. However, if we use a parallel processing, a relatively high-speed transformation can be achieved. For example, if a CMOS is used, the transmission of Mbits/s orders seems to be achieved.

A public-key cryptosystem may make little sense unless it is equivalent to a problem, such as prime factorization, which is thought to be difficult from mathematical point of view. However, it seems meaningful to pursue a safer cryptosystem based on our cryptosystem if demand for high-speed processing is taken into consideration. In order to put our cryptosystem to practical use in the future, we will make its reliability higher. We would greatly appreciate any criticisms and suggestions from cryptographic researchers.

Acknowledgments

The authors are grateful to Professor Toshinobu Kaneko at the Tokyo University of Science, Mr. Sakae Hasegawa, Dr. Kaoru Kurosawa, a lecturer of the Tokyo Institute of Technology, and Dr. Toshiya Itoh, a research associate, for their helpful suggestions.

References

[1] S. Tsujii, K. Kurosawa, T. Itoh, A. Fujioka, and T. Matsumoto. A public-key cryptosystem based on the difficulty of solving a system of non-linear equations. *IEICE Transactions* (D), J69-D, No.12 (1986), 1963-1970.

[2] S. Hasegawa and T. Kaneko. An attacking method for a public-key cryptosystem based on the difficulty of solving a system of non-linear equations. *Proc. 10th Symposium on Information Theory and Its Applications*, JA5-3, November 1987.

[3] Tsujii, Fujioka, and Itoh. Generalization of the public-key cryptosystem based on the difficulty of solving a system of non-linear equations. *Proc. 10th Symposium on Information Theory and Its Applications*, JA5-4, November 1987.

[4] Tsujii, Fujioka, and Hirayama. Generalization of the public-key cryptosystem based on the difficulty of solving a system of non-linear equations. *Proc. 1988's Symposium for Cryptography and Information Security*, E1, February 1988.

[5] Okamoto and Nakamura. Evaluation of public-key cryptosystems proposed recently. *Proc. 1986's Symposium for Cryptography and Information Security*, D1, February 1986.

[6] Tsujii, Akiyama, and Itoh. Hardware configuration for moon letter cryptosystem. *Proc. 1987's Workshop for Cryptography and Information Security, WCIS87-12*, July 1987.

(This paper was received on June 15, 1988 and accepted on September 8, 1988.)