Presentation of a new class of public key cryptosystems K(XIII)SE(1)PKC along with K_p(XIII)SE(1)PKC that realizes the coding rate of exactly 1.0, constructed by modifying K(XII)SE(1)PKC.

Masao KASAHARA *†

Abstract

In this paper, we present a new class of public key cryptosystems by modifying K(XII)SE(1)PKC [1], referred to as K(XIII)SE(1)PKC, and a particular class of K(XIII)SE(1)PKC, K_p(XIII)SE(1)PKC. We show that K(XIII)SE(1)PKC would improve both the coding rate and the security, compared with K(XII)SE(1)PKC. We also show that K_p(XIII)SE(1)PKC realizes the coding rate of exactly 1.0. In a sharp contrast with the conventional code based PKC (CB·PKC) that uses Goppa code, in K(XII)SE(1)PKC, K(XIII)SE(1)PKC and K_p(XIII)SE(1)PKC, we do not care for the security of the primitive polynominal that generates the Reed-Solomon code.

keyword

Public Key Cryptosystem, Error-Correcting Code, Reed-Solomon code, Code based PKC, McEliece PKC.

1 Introduction

Various studies have been made of the public key cryptosystem(PKC). The security of the PKC's proposed so far, in most cases, depends on the difficulty of discrete logarithm problem or factoring problem. For this reason, it is desired to investigate another classes of PKC's that do not rely on the difficulty of these two problems. The multivariate PKC is one of the very promising candidates of a member of such classes. However, most of the multivariate PKC's are constructed by the simultaneous equations of degree larger than or equal to 2 [2] $\sim$ [7]. Recently the author proposed a several classes of linear multivariate PKC's that are constructed by many sets of linear equations [8] $\sim$ [11] based on error-correcting codes. It should be noted that McEliece PKC [12], a class of code based PKC(CB·PKC), can be regarded as a class of the linear multivariate PKC. Excellent analyses and survey are given, for example, in Refs. [13] and [14].

In this paper, we present a new class of public key cryptosystems, by modifying K(XII)SE(1)PKC [1], referred to as K(XIII)SE(1)PKC, K_p(XIII)SE(1)PKC. We show that K(XIII)SE(1)PKC would improve both the coding rate and the security, compared with K(XII)SE(1)PKC. We also show that K_p(XIII)SE(1)PKC realizes the coding rate of exactly 1.0. In a sharp contrast with the conventional code based PKC (CB·PKC) that uses Goppa code, in K(XII)SE(1)PKC, K(XIII)SE(1)PKC and K_p(XIII)SE(1)PKC, we do not care for the security of the primitive polynominal that generates the Reed-Solomon code.

*Research Institute for Science and Engineerging, Waseda University.
†Research and Development Initiative, Chuo University. kasahara@ogu.ac.jp

Throughout this paper, when the variable v_i takes on a value $\tilde{v}_i$, we shall denote the corresponding vector $\boldsymbol{v} = (v_1, v_2, \cdots, v_n)$ as

$$\tilde{\boldsymbol{v}} = (\tilde{v}_1, \tilde{v}_2, \cdots, \tilde{v}_n). \tag{1}$$

The vector $\boldsymbol{v} = (v_1, v_2, \cdots, v_n)$ will be represented by polynomial as

$$v(x) = v_1 + v_2 x + \cdots + v_n x^{n-1}. \tag{2}$$

The $\tilde{u}$, $\tilde{u}(x)$ et al. will be defined in a similar manner.

Let us define several symbols.

$\boldsymbol{a}$: I·message added on the check symbols of code word as an error vector, $(a_1, a_2, \cdots, a_t)$ over $\mathbb{F}_{2^m}$, where we assume that $a_i \neq 0$; $i = 1, 2, \cdots, t$, for K(XIII)SE(1)PKC.

$\boldsymbol{m}$: II·message, $(m_1, m_2, \cdots, m_\eta)$ over $\mathbb{F}_{2^m}$.

$\boldsymbol{\mu}_i$: carrier for II·message $\boldsymbol{m}$; $i = 1, 2, \cdots, k$.

$\boldsymbol{v}_i$: code word over $\mathbb{F}_{2^m}$ for $\boldsymbol{\mu}_i$; $i = 1, 2, \cdots, k$.

$\boldsymbol{u}_i$: secret key over $\mathbb{F}_{2^m}$; $i = 1, 2, \cdots, g$.

$G(x)$: generator polynomial of degree g, over $\mathbb{F}_{2^m}$.

E : exponent to which $G(x)$ belongs, exponent of $G(x)$ for short.

$\{v_i(x)\}$: Reed-Solomon code of length E, generated by $G(x)$.

$\textcircled{j}$: location of erasure error on $\boldsymbol{\mu}_i$; $j = 1, 2, \cdots, \eta$.

$\boldsymbol{\mu}_{i\textcircled{j}}$: coefficient of $x^{\textcircled{j}}$; $i = 1, 2, \cdots, k; j = 1, 2, \cdots, \eta$.

$[i]$: location of message symbol a_i on ciphertext.

(i) : location of $\boldsymbol{u}_i$ randomly selected according to a random choice of $[i]$; $i = 1, 2, \cdots, k$.

$|C|$: length of ciphertext.

$P_s[\hat{A}_i]$: probability that event A_i is successfully estimated by an exhaustive attack.

$\#S$: order of set S.

2 K(XIII)SE(1)PKC

2.1 Theoretical background of present paper

In 1970's, the various works were made of the jointly optimization problems for realizing a high speed and a reliable digital transmission system. One of the most popularly known and highly focused result is the optimum decoding scheme, for partial response type channels, with Vitebi decoding [15]. The author was also much involved in the study of the jointly optimization problems for source and channel coding, syndrome coding (see Fig.1), based on algebraic coding theory. However unfortunately syndrome coding itself was considered not worthy of note, although another coding scheme such as vector quantization [16] was the center of attention among the researchers working on source coding theory.

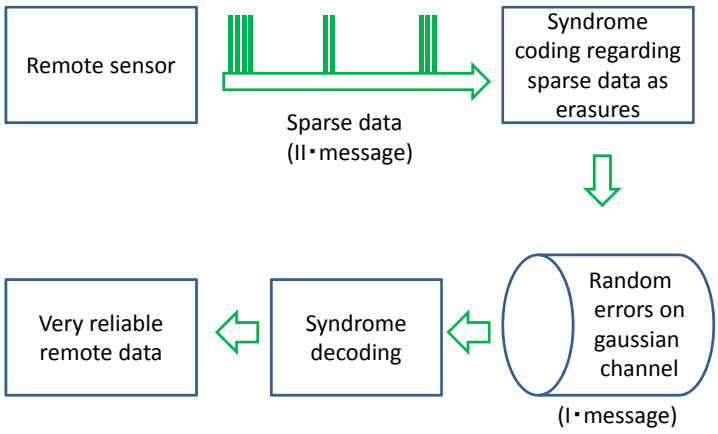

Fig. 1: Syndrome coding

In Feb.1986, the author presented a survey paper on cryptgraphy [17]. In Ref. [17], the author suggested the using of McEliece PKC on noisy channel and presented a very simple scheme of joint coding for encryption and error control coding, based on McEliece PKC.

The author recently proposed a new class of code based PKC, K(XII)SE(1)PKC [1], on the basis of syndrome coding. K(XIII)SE(1)PKC and $\mathrm{K_p}$(XIII)SE(1)PKC presented in this paper are modified versions of K(XII)SE(1)PKC.

2.2 Construction

Let the vector $\boldsymbol{\mu}_i$ over $\mathbb{F}_{2^m}$ be defined by

$$\begin{gathered}\boldsymbol{\mu}_i = (\mu_{i1}, \mu_{i2}, \cdots, \mu_{iK}) \ ; \ i = 1, 2, \cdots, k \ ; \\ \boldsymbol{\mu}_i \neq \boldsymbol{\mu}_j \text{ for } (i \neq j),\end{gathered} \tag{3}$$

where

$$K = E - g. \tag{4}$$

Let $\mu_i(x)$ be

$$\mu_i(x) = \mu_{i\textcircled{1}} x^{\textcircled{1}} + \mu_{i\textcircled{2}} x^{\textcircled{2}} + \cdots + \mu_{i\textcircled{\eta}} x^{\textcircled{\eta}} \ ; \ i = 1, 2, \cdots, k, \tag{5}$$

where the exponent $\textcircled{i}$ takes on a random value such that

$$0 \leq \textcircled{i} \leq K - 1 \ ; \ \textcircled{i} \neq \textcircled{j} \quad \text{for} \quad i \neq j. \tag{6}$$

We assume that $\mu_{i\textcircled{j}}$ takes on a random value over $\mathbb{F}_{2^m}$.

From Eq.(6), we see that the Hamming weight of $\boldsymbol{\mu}_i$ is

$$w(\boldsymbol{\mu}_i) \leq \eta \ ; \ i = 1, 2, \cdots, k. \tag{7}$$

Let carrier $\mu_i(x)$ be transformed into

$$\begin{aligned}\mu_i(x) x^g &\equiv r_i(x) \mod G(x) \ ; i = 1, 2, \cdots, k, \\ &= r_{i1} + r_{i2} x + \cdots + r_{ig} x^{g-1}.\end{aligned} \tag{8}$$

We then have the code word $v_i(x)$ as

$$v_i(x) = \mu_i(x) x^g + r_i(x) \equiv 0 \mod G(x). \tag{9}$$

Let the code words of $\{\boldsymbol{v}_i\}$ be

$$\begin{aligned}
\boldsymbol{v}_1 &= (\mu_{11}, \mu_{12}, \cdots, \mu_{1K}, r_{11}, r_{12}, \cdots, r_{1g}),\\
\boldsymbol{v}_2 &= (\mu_{21}, \mu_{22}, \cdots, \mu_{2K}, r_{21}, r_{22}, \cdots, r_{2g}),\\
&\vdots\\
\boldsymbol{v}_k &= (\mu_{k1}, \mu_{k2}, \cdots, \mu_{kK}, r_{k1}, r_{k2}, \cdots, r_{kg}).
\end{aligned} \tag{10}$$

Let A_r be

$$A_r = \begin{bmatrix} r_{11}, & r_{12}, & \cdots, & r_{1g} \\ r_{21}, & r_{22}, & \cdots, & r_{2g} \\ \vdots & \vdots & & \vdots \\ r_{k1}, & r_{k2}, & \cdots, & r_{kg} \end{bmatrix}, \tag{11}$$

where we let

$$\boldsymbol{r}_i = (r_{i1}, r_{i2}, \cdots, r_{ig}). \tag{12}$$

The matrix A_r is transformed into

$$A_r \cdot P_I = \begin{bmatrix} u_{11}, & u_{12}, & \cdots, & u_{1g} \\ u_{21}, & u_{22}, & \cdots, & u_{2g} \\ \vdots & \vdots & & \vdots \\ u_{k1}, & u_{k2}, & \cdots, & u_{kg} \end{bmatrix}, \tag{13}$$

where P_I is a random column permutation matrix over $\mathbb{F}_{2^m}$.

Let $\boldsymbol{u}_i$ be

$$\boldsymbol{u}_i = (u_{i1}, u_{i2}, \cdots, u_{ig}) \ \ ; \ \ i = 1, 2, \cdots, k. \tag{14}$$

We assume that the elements of the set $\{\boldsymbol{u}_i\}$ are ordered as $\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_k$. The set $\{\boldsymbol{u}_i\}$ will be publicized. We shall refer to subscript i as location i.

For I·message $\boldsymbol{a} = (a_1, a_2, \cdots, a_t)$, Bob constructs the message polynomial :

$$\begin{aligned}
a_t(x) = \quad & a_1 x^{[1]} + a_2 x^{[2]} + \cdots + a_t x^{[t]};\\
& 0 \le [i] \le g - 1,
\end{aligned} \tag{15}$$

where the exponents $[1], [2], \cdots, [t]$ satisfies

$$1 \le [1] < [2] < \cdots < [t-1] < [t] \le g - 1. \tag{16}$$

Throughout this paper (except $\mathrm{K_p}$(XIII)SE(1)PKC), we assume that Bob randomly selects a set of exponents $\{[i]\}$, all over again, for every given I·message.

Set of keys are :

Public key	:	$\{\boldsymbol{u}_i\}$
Secret key	:	$\{\boldsymbol{\mu}_i\}, \{\boldsymbol{r}_i\}, A_r, P_I$

After constructiong $a_t(x) = a_1 x^{[1]} + a_2 x^{[2]} + \cdots + a_t x^{[t]}$, Bob selects the location of the public key $(1), (2), \cdots, (\eta)$ based on the transformation :

$$\varphi(\{[i]\}) = \{(i)\}, \tag{17}$$

where the order of $\{(i)\}$ is chosen as

$$\#\{(i)\} = \eta < k. \tag{18}$$

After performing this transformation, Bob selects public keys $\boldsymbol{u}_{(1)}, \boldsymbol{u}_{(2)}, \cdots, \boldsymbol{u}_{(\eta)}$ from $\{\boldsymbol{u}_i\}$.
Let the word $\boldsymbol{w}$ be

$$\boldsymbol{w} = m_1\boldsymbol{u}_{(1)} + m_2\boldsymbol{u}_{(2)} + \cdots + m_\eta\boldsymbol{u}_{(\eta)}. \tag{19}$$

The ciphertext $\boldsymbol{C}$ is

$$\boldsymbol{C} = \boldsymbol{w} + \boldsymbol{a}_t. \tag{20}$$

It should be noted that, in accordance with a random choice of locations (i)'s, the carrier $\boldsymbol{\mu}_{(i)}$'s are selected.

Theorem 1: Erasure error $E_\eta(x)$ due to Ⅱ·message $\boldsymbol{m}$ is

$$\begin{aligned} E_\eta(x) = \sum_{i=1}^{\eta} m_i\mu_{(i)}\textcircled{1}x^{\textcircled{1}} + \sum_{i=1}^{\eta} m_i\mu_{(i)}\textcircled{2}x^{\textcircled{2}} \\ + \cdots + \sum_{i=1}^{\eta} m_i\mu_{(i)}\textcircled{\eta}x^{\textcircled{\eta}} \end{aligned} \tag{21}$$

Proof : Straightforward □

We also see from Eq.(21) that $\boldsymbol{a}_t$ results in random errors. The minimum distance, D, of the Reed-Solomon code generated by $G(x)$ of degree g is

$$D = g + 1. \tag{22}$$

The following relation :

$$2t + \eta + 1 = D, \tag{23}$$

is required to hold so that the messages $\boldsymbol{m}$ and $\boldsymbol{a}$ may be correctly decoded.

Let B_μ over $\mathbb{F}_{2^m}$ be

$$B_\mu = \begin{bmatrix} \mu_{(1)}\textcircled{1}, & \mu_{(1)}\textcircled{2}, & \cdots, & \mu_{(1)}\textcircled{\eta} \\ \mu_{(2)}\textcircled{1}, & \mu_{(2)}\textcircled{2}, & \cdots, & \mu_{(2)}\textcircled{\eta} \\ & \vdots & & \\ \mu_{(\eta)}\textcircled{1}, & \mu_{(\eta)}\textcircled{2}, & \cdots, & \mu_{(\eta)}\textcircled{\eta} \end{bmatrix}. \tag{24}$$

All the row vectors of B_μ are selected based on a set of randomly chosen locations {(i)}.

Let $P_{B_\mu}[\mathrm{NS}]$ be the probability that B_μ over $\mathbb{F}_{2^m}$ proves non-singular under the condition that all the elements are randomly chosen. The probability $P_{B_\mu}[\mathrm{NS}]$ can be bounded by

$$P_{B_\mu}[\mathrm{NS}] > (1 - 2^{-m})^\eta \cong 1 - \eta 2^{-m}\,;\, m \gtrsim 88. \tag{25}$$

We see that, for $m = 88$, $\eta = 64$, $t = 32$, $g = 128$, $k = 128$, non-singular matrix B_μ can be generated with sufficiently high probability of more than $1 - 2.07 \times 10^{-25}$.

Thus even if Bob sends $N = 10^{12}$ ciphertexts of size 1.41KB, the probability that one of the randomly chosen B_μ's proves singular takes on, from Chebyshev's inequality *, an extremely small value of less than 2.07×10^{-13}.

We see that $\boldsymbol{E}_\mu$ can be decoded by erasure and error decoding [19], as all the erasure locations $\textcircled{1}, \textcircled{2}, \cdots \textcircled{\eta}$ are known to Alice. Let $S_{\boldsymbol{u}}$ and $S_{\widetilde{\boldsymbol{u}}}$ be

$$S_{\boldsymbol{u}} = \{\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_k\} \tag{26}$$

and

$$S_{\widetilde{\boldsymbol{u}}} = \{\widetilde{\boldsymbol{u}}_{(1)}, \widetilde{\boldsymbol{u}}_{(2)}, \cdots, \widetilde{\boldsymbol{u}}_{(\eta)}\}, \tag{27}$$

---

* $\varepsilon = N^{-1} - P_{B_\mu}[\mathrm{NS}] = 10^{-12} - 2.07 \times 10^{-25}$.
$P[\varepsilon] \leq \frac{P_{B_\mu}[\mathrm{NS}](1-P_{B_\mu}[\mathrm{NS}])}{N\varepsilon^2} \cong \frac{2.07\times10^{-25}}{10^{-12}} = 2.07 \times 10^{-13}$

Let a subset of $S_{\boldsymbol{u}}$ be $\overline{S}_i$ such that

$$S_{\widetilde{u}} \subset \overline{S_i} \tag{28}$$

and

$$\#\overline{S}_i = g - t - i. \tag{29}$$

2.3 Security considerations

Remark 1: The using of the Reed-Solomon codes is very attractive because they meet the very nice property of maximum distance seprability. However, so far, the using of Reed-Solomon code has been supposed to be a little dangerous as the generator polynomial can be estimated without much difficulty compared with the Goppa code. However the author strongly feels that even if the generator polynomial is disclosed, K(XII)SE(1)PKC can be made sufficiently secure as we discussed in Ref [1]. Accordingly we do not regard G(x) as a secret key. However G(x) is not recommended to be publicized.

Attack 1: Attack on estimating carrier $\boldsymbol{\mu}_i$

Let the probability that carrier $\boldsymbol{\mu}_i$ is estimated correctly be denoted $P_s[\hat{\boldsymbol{\mu}}_i]$. Then

$$P_s[\hat{\boldsymbol{\mu}}_i] = \left(\begin{array}{c} K \\ \eta \end{array} \right)^{-1} (2^m)^{-\eta}. \tag{30}$$

In order to be secure against Attack 1, we recommend $P_s[\hat{\boldsymbol{\mu}}_i]$ be

$$P_s[\hat{\boldsymbol{\mu}}_i] \leq 2^{-80} = 7.21 \times 10^{-25}. \tag{31}$$

In this paper, we let m be longer than 88 and η, larger than 8. As a result $P_s[\hat{\boldsymbol{\mu}}_i]$ can be made much smaller than 2^{-80}. We conclude that K(XIII)SE(1)PKC can be secure against Attack 1.

Attack 2: Attack on II·message

Let us define the following steps of Attack 2:

Step 1 : Exhaustive attack for disclosing $g - t - i$ error free symbols among the symbols of a given ciphertext $\boldsymbol{C}$.

The probability that Step 1 proves successful is

$$\begin{aligned} P_s[\text{Step 1}] &= \left(\begin{array}{c} g-t \\ g-t-i \end{array} \right) \left(\begin{array}{c} g \\ g-t-i \end{array} \right)^{-1} \\ &= \frac{(g-t)!(t+i)!}{i!g!}. \end{aligned} \tag{32}$$

Step 2 : Exhaustive attack for obtaining $g - t - i$ elements from $\{\boldsymbol{u}_i\}$ that includes $\widetilde{\boldsymbol{u}_{(1)}}, \widetilde{\boldsymbol{u}_{(2)}}, \cdots, \widetilde{\boldsymbol{u}_{(\eta)}}$.

The probability of successfully obtaining $g - t - i$ symbols that includes $\widetilde{\boldsymbol{u}}_{(1)}, \widetilde{\boldsymbol{u}}_{(2)}, \cdots, \widetilde{\boldsymbol{u}}_{(\eta)}$ is

$$\begin{aligned} P_s[\text{Step 2}] &= \left(\begin{array}{c} k-\eta \\ g-t-\eta-i \end{array} \right) \left(\begin{array}{c} k \\ g-t-i \end{array} \right)^{-1} \\ &= \frac{(k-\eta)!(g-t-i)!}{k!(t-i)!}, \end{aligned} \tag{33}$$

where we let $g - t - \eta = t$.

Let $P_s[\text{Step 1}] * P_s[\text{Step 2}] = P_s[\text{Attack 2}]$ be

$$P_s[\text{Attack 2}] = AB, \tag{34}$$

where

$$A = \frac{(g-t)!(k-\eta)!}{g!k!}, \tag{35}$$

$$
\begin{aligned}
B &= \frac{(t+i)!(g-t-i)!}{i!(t-i)!} \\
&= \frac{1}{t!}\left(\begin{array}{c} t \\ i \end{array} \right)(t+i)!(g-t-i)! \\
&< \frac{1}{t!}\left(\begin{array}{c} t \\ i \end{array} \right) t!(g-t)!.
\end{aligned} \tag{36}
$$

As the following relation holds

$$
\left(\begin{array}{c} t \\ i \end{array} \right) < \left(\begin{array}{c} t \\ t/2 \end{array} \right). \tag{37}
$$

We now have the upper bound of B as

$$
B < \left(\begin{array}{c} t \\ t/2 \end{array} \right)(g-t)!. \tag{38}
$$

The upper bound of probability $P_s[\text{Attack 2}]$ is

$$
P_s[\text{Attack 2}] < \frac{(k-\eta)!}{g!k!}\left(\begin{array}{c} t \\ t/2 \end{array} \right)\{(g-t)!\}^2. \tag{39}
$$

Example 1 : $m = 96$, $k = 312$, $g = 212$, $t = 64$, $\eta = 84$.

We see that the relation $2t + \eta = g$ holds.

The probabilities $P_{B_\mu}[\text{NS}]$, and $P_s[\text{Attack 1}]$ are

$$
P_{B_\mu}[\text{NS}] > 1 - \eta \cdot 2^{-m} = 1 - 1.06 \times 10^{-27}, \tag{40}
$$

$$
\begin{aligned}
P_s[\text{Attack 2}] &< \frac{(k-\eta)!}{g!k!}\left(\begin{array}{c} t \\ t/2 \end{array} \right)\{(g-t)!\}^2 \\
&= 1.78 \times 10^{-72},
\end{aligned} \tag{41}
$$

yielding an extremely small value.

2.4 Encryption and decryption processes.

[Encryption process]

Step 1 : Given I·message $\widetilde{\boldsymbol{a}} = (\widetilde{a}_1, \widetilde{a}_2, \cdots, \widetilde{a}_t)$, Bob randomly chooses the locations : $[1], [2], \cdots, [t]$.

Step 2 : Bob transforms I·message vector $\boldsymbol{a}$ into $\widetilde{a}_t(x) = \widetilde{a}_1 x^{[1]} + \widetilde{a}_2 x^{[2]} + \cdots + \widetilde{a}_t x^{[t]}$.

Step 3 : Bob transforms $\{[i]\}$ into $\{(i)\}$, yielding $\widetilde{\boldsymbol{u}}_{(1)}, \widetilde{\boldsymbol{u}}_{(2)}, \cdots, \widetilde{\boldsymbol{u}}_{(\eta)}$.

Step 4 : Given II·message $\widetilde{\boldsymbol{m}} = (\widetilde{m}_1, \widetilde{m}_2, \cdots, \widetilde{m}_\eta)$, Bob calculates the word : $\widetilde{\boldsymbol{w}} = m_1\widetilde{\boldsymbol{u}}_{(1)} + m_2\widetilde{\boldsymbol{u}}_{(2)} + \cdots + m_\eta\widetilde{\boldsymbol{u}}_{(\eta)}$.

Step 5 : Bob calculates the ciphertext $\widetilde{\boldsymbol{C}} = \widetilde{\boldsymbol{w}} + \widetilde{\boldsymbol{a}}_t$.

Step 6 : Bob sends the ciphertext $\widetilde{\boldsymbol{C}}$ to Alice.

[Decryption process]

Step 1 : Receieving the ciphertext $\widetilde{\boldsymbol{C}}(=\widetilde{\boldsymbol{w}}+\widetilde{\boldsymbol{a}}_t)$ from Bob, Alice calculates the following :
$(\widetilde{\boldsymbol{w}}+\widetilde{\boldsymbol{a}}_t)P_I^{-1}=\widetilde{\boldsymbol{r}}+\widetilde{\boldsymbol{\alpha}}_t$, where $\widetilde{\boldsymbol{\alpha}}_t=\widetilde{\boldsymbol{a}}_tP_I^{-1}$.
Let $\widetilde{\boldsymbol{r}}+\widetilde{\boldsymbol{\alpha}}_t$ be denoted $\widetilde{\boldsymbol{r}}+\widetilde{\boldsymbol{\alpha}}^t=\widetilde{\boldsymbol{C}}^{-T}=(\widetilde{c}_1^{-T},\widetilde{c}_2^{-T},\cdots,\widetilde{c}_g^{-T})$.

Step 2 : Given $\widetilde{\boldsymbol{C}}^{-T}$, Alice decodes an erasure value $\boldsymbol{E}_\eta$ and $\widetilde{\boldsymbol{\alpha}}_t$,
for example, with Euclidean erasure and error decoding algorithm [18], [19].

Step 3 : Alice decodes $\widetilde{\boldsymbol{a}}=(\widetilde{a}_1,\widetilde{a}_2,\cdots,\widetilde{a}_t)$ by performing P_I on $\widetilde{\alpha}_t$.

Step 4 : Alice decodes $\{\widetilde{(i)}\}$ from $\{\widetilde{[i]}\}$.

Step 5 : From $\widetilde{E}_\eta(x)$, Ⅱ·message $\boldsymbol{m}$ is decoded by solving linear simultaneous equations given by Eq.(21).

3 A particular class of K(XIII)SE(1)PKC, $\mathrm{K_p}$(XIII)SE(1)PKC

In $\mathrm{K_p}$(XIII)SE(1)PKC, the locations for Ⅱ· message are not randomly chosen but are predetermined. Without loss of generality, let these predetermined locations be $x^0,x^1,x^2,\cdots,x^{t-1}$. As a result $a_t(x)$ is now

$$a_t(x)=a_1+a_2x+\cdots+a_tx^{t-1}. \tag{42}$$

Bob selects the locations $\{(i)\}$ based on

$$\varphi(\{a_i\})=\{(i)\}. \tag{43}$$

Considering that a_t results in an erasure error, the coding rate ρ is

$$\rho=\frac{t+\eta}{|c|}=\frac{g}{g}=1.0. \tag{44}$$

We see that the coding rate ρ takes on the value of exactly 1.0.

Attack 3: Exhaustive attack on disclosing the set of locations $\{(i)\}$

The probability, $P_s[\{(i)\}]$, that $\{(i)\}$ is successfully disclosed by an attacker is

$$P_s[\{(i)\}]=\left(\begin{array}{c}k\\ \eta\end{array}\right)^{-1}. \tag{45}$$

In order to be secure against Attack 3, we let $P_s[\{(i)\}]$ be

$$P_s[\{(i)\}]<2^{-80}=8.27\times10^{-25}. \tag{46}$$

Example 2 : $m=88$, $k=96$, $g=64$, $t=\eta=32$

The probability $P_s[\{(i)\}]$, coding rate ρ and the size of public key S_{PK} are

$$P_s[\{(i)\}]=\left(\begin{array}{c}96\\ 32\end{array}\right)^{-1}=3.36\times10^{-26}, \tag{47}$$

$$\rho=\frac{\eta+t}{|c|}=1.0, \tag{48}$$

$$S_{\mathrm{PK}}=mgk=67.6(KB). \tag{49}$$

We see that the coding rate ρ takes on exactly 1.0. We also see that the size of public key is smaller than that of Example 1, by a factor of about 12.

The probability $P_s[\{(i)\}]$ takes on a value less than 2^{-80}. We thus conclude $\mathrm{K_p}$(XIII)SE(1)PKC would realizes a secure PKC with a smaller size of public key compared with K(XII)SE(1)PKC and K(XIII)SE(1)PKC.

Besides K_p(XIII)SE(1)PKC realizes the coding rate of exactly 1.0, yeilding a simple digital signature scheme compared with K(XIII)SE(1)PKC.

4 Conclusion

In this paper, we have presented a new class of public key cryptosystem by modifying K(XII)SE(1)PKC, referred to as K(XIII)SE(1)PKC and K_p(XIII)SE(1)PKC. We have clarified the followings :

(i) K(XIII)SE(1)PKC would improve both the coding rate and the security, compared with K(XII)SE(1)PKC.

(ii) In K_p(XIII)SE(1)PKC, the coding rate ρ takes on the value of exactly 1.0, yielding a simple signature scheme compared with K(XIII)SE(1)PKC.

(iii) In a sharp contrast with the conventional CB·PKC that uses Goppa code, in K(XII)SE(1)PKC, K(XIII)SE(1)PKC and K_p(XIII)SE(1)PKC, we do not care for the security of the primitive polynominal that generates the Reed-Solomon code.

References

[1] M. Kasahara, "A New Class of Public Key Cryptosystems Constructed Based on Reed-Solomon Codes K(XII)SE(1)PKC.– Along with a presentation of K(XII)SE(1)PKC over $\mathbb{F}_{2^8}$, the field extensively used for various storage and transmission systems –", Cryptology ePrint Archive, 2013/363 (2013-06).

[2] M. Kasahara and R. Sakai "A Construction of Public Key Cryptosystem for Realizing Ciphertext of size 100 bit and Digital Signature Scheme", IEICE Trans. Vol. E87-A, 1, pp.102-109 (2004-01).

[3] M. Kasahara and R. Sakai "A Construction of Public Key Cryptosystem Based on Singular Simultaneous Equations", IEICE Trans. Vol. E88-A, 1, pp.74-79 (2005-01).

[4] N. Koblitz "Algebraic Aspect of Cryptography", Springer Verlag, Berlin Heidelberg.

[5] T. Mastumoto and H. Imai "Public Quadratic Polynomial-Tuples for Efficient Signature - Verification and Message-Encryption", Advances in Cryptology, Eurocrypt'88, Springer-Verlag, pp.419-453 (1988).

[6] J. C. Faugere and A. Joux "Algebraic cryptanalysis of hidden field equation (HFE) cryptosystems using Gröbner bases", In Advances in Cryptoglogy-CRYPTO 2003 pp.44-60 (2003).

[7] C. Wolf: "Multivariate Quadratic Polynomials in Public Key Cryptography", Dr. Thesis, Katholieke Universiteit Leuven, (2005-11).

[8] M. Kasahara "Construction of New class of Linear Multivariate Public Key Cryptosystem - Along With a Note on the Number 9999990 and its Application", Technical Report of IEICE, ISEC 2009-44 (2009-09).

[9] M. Kasahara "A New Class of Public Key Cryptosystems Constructed Based on Perfect Error-Correcting Codes Realizing Coding Rate of exactly 1.0", Cryptology ePrint Archive , Report 2010/139 (2010-03).

[10] M. Kasahara "A Construction of New Class of Linear Multivariate Public Key Cryptosystem Constructed Based on Error Correcting Codes", Technical Report of IEICE , ISEC 2009-135 (2010-03).

[11] M. Kasahara: "Public Key Cryptosystems Constructed Based on Pseudo Cyclic Codes, K(IX)SE(1)PKC, Realizing Coding Rate of Exactly 1.0", Cryptology ePrint Archive, Report 2011/545, (2011-09).

[12] R. J. McEliece: "A Public-key Cryptosystem Based on Algebraic Coding Theory", DSN Progress Report, no.42-44, pp.114-116 (1978).

[13] J. C. Faugere and A. Otomoni, L. Perret, J. P. Tillich: "Algebraic Cryptanalysis of McElierce Variants with Compact Keys", Eurocrypt'10.

[14] E. M. Gabidulin: "Public-key cryptosystems based on linear codes", Report 95-30, TU Delft (1995).

[15] A. J. Viterbi: "Error-Bounds for Convolutional Codes and an Asymptotially Optimum Decoding Algorithm", IEEE Trans. Inform. Theory, IT-13, 2, pp.260-269 (April 1967).

[16] H. Kumazawa, M. Kasahara, and T. Namekawa: "A Construction of Vector Quantizers for Noisy Channels", Trans. of IEICE, Vol. J67-B, No.1 pp.1-8 (1984-01).

[17] M. Kasahara: "On Cryptography", Proc. of Symposium on Information and Communication Network Security, pp.154-179 (Feb. 1986).

[18] Y. Sugiyama, M. Kasahara, S. Hirasawa and T. Namekawa: "A method for solving key equation for decoding Goppa codes", Info. and Control, 27, pp.87-99 (1975).

[19] Y. Sugiyama, M. Kasahara, S. Hirasawa and T. Namekawa: "An Erasures-and-Errors decoding Algorithm for Goppa Codes", IEEE Trans. on Inform. Theory, IT-22, 2, pp.238-241 (1976-03).