Technical Note

New Proof Techniques Using the Properties of Circulant Matrices for XOR-based (k, n) Threshold Secret Sharing Schemes

Koji Shima[1,a) Hiroshi Doi[1,b)

Received: June 19, 2020, Accepted: December 1, 2020

Abstract: Several secret sharing schemes with low computational costs have been proposed. XOR-based secret sharing schemes have been reported to be a part of such low-cost schemes. However, no discussion has been provided on the connection between them and the properties of circulant matrices. In this paper, we propose several theorems of circulant matrices to discuss the rank of a matrix and then show that we can discuss XOR-based secret sharing schemes using the properties of circulant matrices. We also present an evaluation of our software implementation.

Keywords: secret sharing scheme, XOR-based scheme, circulant matrix, software implementation

1. Introduction

In modern information society, a strong need exists to securely store large amounts of secret information to prevent information theft or leakage and avoid information loss. Secret sharing schemes are known to simultaneously satisfy the need to distribute and manage secret information, so that such information theft and loss can be prevented.

A secret sharing scheme involves a dealer who has a secret, a set of n participants, and a collection of subsets of k participants. The dealer distributes shares to the n participants through secure channels. Blakley [1] and Shamir [2] independently introduced the basic idea of a (k, n) threshold secret sharing scheme in 1979. In Shamir's (k, n) threshold scheme, n shares are generated from the secret and each of these shares is privately distributed to a participant. Next, the secret can be recovered using any subset k of the n shares, but it cannot be recovered with fewer than k shares. Furthermore, every subset comprising less than k participants cannot obtain any information regarding the secret. Therefore, the original secret is secure even if some of the shares are leaked or exposed. Conversely, the secret can be recovered even if some of the shares are missing.

1.1 XOR-based Secret Sharing Schemes

Shamir's (k, n) threshold scheme requires extensive calculations for generating the n shares and recovering the secret from k shares, because in doing so, a polynomial of degree $k - 1$ must be processed. To tackle the problem, several XOR-based secret sharing schemes that use only XOR operations to distribute and recover the secret with low computational costs have been proposed. The schemes of Fujii et al. [3] and Kurihara et al. [4], [5] are reported as such (k, n) threshold schemes. Kurihara et al. [6] then proposed an XOR-based (k, L, n) *ramp* scheme.

1.2 Fast Schemes

Operations over GF(2^L), where GF is the Galois or finite field, yield fast schemes, but one multiplication operation requires some XOR operations based on the analysis conducted by Kurihara et al. [7]. Shima et al. reported in Ref. [8] that this computational cost can be one operation if we use a lookup table that has been precomputed for the multiplication operation over GF(2^L); otherwise, there remains little choice but to practically choose $L = 8$ in terms of the amount of available memory. To extend the size of L, Ikarashi et al.'s technique [9] is suited for fast multiplication operations over GF(2^{64}). Because their technique requires an extended CPU instruction set, it cannot be applied to all hardware, such as embedded devices, which are used widely in the Internet of Things. Schemes constructed using simple XOR operations can use the maximum bit length of XOR, such as 64 bits. Only simple XOR operations can yield faster schemes.

1.3 Our Contributions

In this paper, we provide a new proof technique that actively uses circulant matrices by referring to Refs. [3], [4], [5] and the $(k, 1, n)$ *ramp* scheme [6]. In other words, we provide a proof technique that differs from those provided by these earlier authors, unlike Fujii et al. [3], who did not provide a proof that shows their scheme is *ideal*. Therefore, our techniques can be applied to Refs. [3], [4], [5], [6].

We are faced with the following two questions:

- Can each block matrix in generator matrix $\mathbf{G}$ in Refs. [4], [5]

[1] Institute of Information Security, Yokohama, Kanagawa 221–0835, Japan
[a) dgs164101@iisec.ac.jp
[b) doi@iisec.ac.jp

The preliminary version [22] of this paper was published at Computer Security Symposium 2019 (CSS 2019), October 2019.

be represented by a matrix systematically reduced from a circulant matrix?
- Can XOR-based secret sharing schemes be tied to Shamir's scheme, using a Vandermonde matrix for recovery?

To answer these questions, we need to study the rank of a matrix based on the properties of circulant matrices. Here, generator matrix $\mathbf{G}$ consists of block matrices. Considering a block matrix obtained by removing one row and one column from a circulant matrix, we can naturally construct a secret sharing scheme. Eventually, our construction is the same as Kurihara et al.'s scheme [6] in the case of $L = 1$; however, we emphasize that we provide another perspective to analyze generator matrix $\mathbf{G}$. More specifically, let matrix $\mathbf{G}_k$ correspond to any subset k of n participants from $\mathbf{G}$ and be hereinafter called a recovery matrix; we can view recovery matrix $\mathbf{G}_k$ as a Vandermonde matrix, in which each element is a circulant matrix, and discuss naturally and neatly the rank of matrix $\mathbf{G}_k$. Here, a circulant matrix is not guaranteed to have a multiplicative inverse because its operations are not performed over a field. For example, circulant matrix

$$\begin{bmatrix} 1\,1\,0\,0\,0 \\ 0\,1\,1\,0\,0 \\ 0\,0\,1\,1\,0 \\ 0\,0\,0\,1\,1 \\ 1\,0\,0\,0\,1 \end{bmatrix}$$

is not regular. Therefore, we provide several theorems of circulant matrices and discuss the rank of matrix $\mathbf{G}_k$.

Our proof techniques are nontrivial because Refs. [3], [4], [5], [6] do not provide the connection between their schemes and a Vandermonde matrix and we need to consider circulant matrices over GF(2). Our overall contributions are summarized as follows:

- We propose several theorems of circulant matrices to discuss naturally the rank of matrix $\mathbf{G}_k$. We then achieve a new security analysis for XOR-based secret sharing schemes.
- We show that XOR-based secret sharing schemes are represented using the properties of circulant matrices. As a result, we establish the connection between XOR-based secret sharing schemes and Shamir's scheme.
- We present a clear evaluation for efficiency between the schemes in Refs. [4], [5] and the $(k, 1, n)$ *ramp* scheme [6]. More specifically, we measure performance that shows the latter is more efficient in distribution if n, or p, is not large, where $k \geq 3$.

Here, several approaches [13], [14], [15], [16] to discussing the connection to Shamir's scheme have been reported, but no discussion has been provided on the connection between XOR-based secret sharing schemes and the properties of circulant matrices. Recent researches [17], [18] also proposed XOR-based related approaches. However, they do not provide a solution as a (k, n) threshold scheme. It is then unclear if our techniques can be applied to Refs. [13], [14], [15], [16], [17], [18].

By using our new techniques, we can shorten many proofs and characterize the categories of matrices that can be used in XOR-based (k, n) threshold schemes. As a result, we can show our proofs in full to prove the *ideal* secret sharing scheme, while in Appendix B, Lemma 2 of Ref. [4], Kurihara et al. omitted a detailed proof on the grounds that it was too long to be described in full. In other words, it is easier to discuss the rank of a matrix and the *ideal* secret sharing scheme by actively using the properties of circulant matrices. Furthermore, our new approach can contribute to providing extensibility for XOR-based secret sharing schemes, for example, discussing the rank of a matrix efficiently for hierarchical secret sharing schemes.

1.4 Organization of the Paper

The rest of this paper is organized as follows: In Section 2, we introduce several preliminaries and review the basic terminology. In Section 3, we propose our several theorems of circulant matrices. Section 4 describes the algorithm of the $(k, 1, n)$ *ramp* scheme [6] using different symbols. In Section 5, we show our proof techniques using the theorems from Section 3. In Section 6, we present our evaluation of our software implementation and comparison with other existing XOR-based schemes. Finally, Section 7 concludes our work.

2. Preliminaries

We introduce several fundamentals as preliminaries.

2.1 Notation

Throughout this paper, we use the following notations:

- $\oplus$ denotes a bitwise XOR operation.
- $||$ denotes a concatenation of bit sequences.
- p is a prime number.
- n denotes the number of participants, where $p \geq n$.
- k is a threshold value, where $2 \leq k \leq n$.
- $\mathcal{P} = \{P_0, \cdots, P_{n-1}\}$ denotes a set of n participants.
- The index values of (1) random numbers, (2) divided pieces of the secret and the shares, (3) XOR-ed terms of elements 1 and 2, (4) participants, and (5) matrices are elements of GF(p); that is, $X_{c(a\pm b)}$ denotes $X_{c(a\pm b) \bmod p}$.
- $H(X)$ denotes the Shannon entropy of a random variable X.
- $|\mathcal{X}|$ denotes the number of elements in a finite set $\mathcal{X}$.
- $\overset{\$}{\leftarrow} \mathcal{X}$ denotes a function to generate an $|\mathcal{X}|$-bit random number from a finite set $\mathcal{X}$.
- $\gcd(a, b)$ denotes the greatest common divisor of a and b.
- $\deg(f(x))$ denotes the degree of a polynomial $f(x)$.
- $[a_{(i,j)}]$ denotes a matrix where $i, j \in \{0, 1, \cdots, t-1\}$. The numbering of the rows and columns starts from zero.
- $[a_{(i,j)}]_{i=a,j=c}^{b,d}$ denotes a matrix where $i \in \{a, a+1, \cdots, b\}$ and $j \in \{c, c+1, \cdots, d\}$.
- $\mathbf{I}_t$ denotes the $t \times t$ identity matrix.
- $\boldsymbol{O}$ denotes a zero matrix.
- $\overset{\circ}{\leftarrow}$ and $\overset{\circ}{\rightarrow}$ denote elementary row operations.

2.2 Circulant Matrix

When $t \times t$ matrix

$$\mathbf{C} = \begin{bmatrix} c_0 & c_1 & \cdots & c_{t-2} & c_{t-1} \\ c_{t-1} & c_0 & \cdots & \cdots & c_{t-2} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ c_2 & \cdots & c_{t-1} & c_0 & c_1 \\ c_1 & c_2 & \cdots & c_{t-1} & c_0 \end{bmatrix}$$

takes the form shown, **C** is called a circulant matrix. When circulant matrices **A** and **B** are given, **A** + **B** and **A**·**B** are also circulant matrices. The addition and the multiplication are commutative. A circulant matrix can be represented by one vector. In this paper, $\mathbf{C} = (c_0, c_1, \cdots, c_{t-2}, c_{t-1})$, which appears as the zeroth row of **C**. The polynomial $c(x) = \sum_{i=0}^{t-1} c_i x^i$ is called the associated polynomial of **C**. The number of elements c_i of **C** that are different from 0 is called the Hamming weight or weight. Theorem 2.1 has been analyzed in Refs. [19], [20].

Theorem 2.1. *The rank of circulant matrix* **C** *is* $t - d$, *where* $d = \deg(\gcd(x^t - 1, c(x)))$ *and* $c(x)$ *is the associated polynomial of* $\mathbf{C} = (c_0, c_1, \cdots, c_{t-2}, c_{t-1})$.

2.3 Circulant Matrix over GF(2)

Here, we show notations and properties of $t \times t$ circulant matrix $\mathbf{C} = (c_0, c_1, \cdots, c_{t-2}, c_{t-1})$ over GF(2).

- g_i denotes circulant matrix **C**, whose i-th element c_i is only 1. The modulus of the index of g_i is t. g_0 means the identity matrix. We also define $1 \overset{\text{def}}{=} g_0$ in the context of a matrix.
- For the multiplication, $g_a \times g_b = g_{a+b}$, $(g_a)^b = g_{ab}$.
- For the distributive property, $(g_a + g_b) \times g_c = g_{a+c} + g_{b+c}$, $g_a \times (g_b + g_c) = g_{a+b} + g_{a+c}$.

Lemma 2.1. *Consider circulant matrix* **C** *over* GF(2) *whose weight is even. The following conditions are satisfied.*

(1) *The sum of the i-th rows for* $i = 0, \cdots, t-2$ *is equivalent to the* $t-1$*-th row.*

(2) *The sum of the i-th columns for* $i = 0, \cdots, t-2$ *is equivalent to the* $t-1$*-th column.*

Proof. Consider each j-th column, where $j = 0, \cdots, t-1$. If the sum of the elements of the i-th rows for $i = 0, \cdots, t-2$ is one, the element of the $t-1$-th row is one. If the sum is zero, the element of the the $t-1$-th row is zero. In the same way, we can consider each j-th row, where $j = 0, \cdots, t-1$. □

Lemma 2.2. *Circulant matrix* **C** *over* GF(2) *whose weight is two has no multiplicative inverse.*

Proof. The rank of **C** is less than or equal to $t-1$ because the sum of the i-th rows for $i = 0, \cdots, t-2$ is equivalent to the $t-1$-th row from Lemma 2.1. Therefore, **C** is singular and has no multiplicative inverse. □

2.4 Perfect Secret Sharing Scheme

In this subsection, we refer to Ref. [10]. A *perfect* secret sharing scheme requires the following conditions:

Correctness Every authorized set $B \in \Gamma$ receives the secret information.

Perfect privacy Every unauthorized set $T \notin \Gamma$ receives no secret information.

The collection Γ is called the access structure. Let S be a random variable in a given probability distribution on the secrets, S_B be a random variable in a given probability distribution on the shares in each authorized set B, and S_T be a random variable in a given probability distribution on the shares of each unauthorized set T. A *perfect* secret sharing scheme would satisfy (Correctness) $H(S|S_B) = 0$ and (Perfect privacy) $H(S|S_T) = H(S)$.

2.5 Ideal Secret Sharing Scheme

In this subsection, we refer to Refs. [4], [5], [11], [12]. The dealer selects secret $s \in \mathcal{S}$ and distributes each share $w_i \in \mathcal{W}_i$ to participant $P_i \in \mathcal{P}$, where $\mathcal{S}$ denotes the set of secrets and $\mathcal{W}_i$ denotes the set of possible shares that P_i might receive. The information rate is then defined as $\rho = \dfrac{H(S)}{\max_{P_i \in \mathcal{P}} H(W_i)}$, where S and W_i denote the random variables induced by $s \in \mathcal{S}$ and $w_i \in \mathcal{W}_i$, respectively. When the probability distributions over $\mathcal{S}$ and shares $\mathcal{W}_i$ are uniform, the information rate is

$$\rho = \frac{\log_2 |\mathcal{S}|}{\max_{P_i \in \mathcal{P}} \log_2 |\mathcal{W}_i|}.$$

A secret sharing scheme is called *ideal* if it is *perfect* and $\rho = 1$. In other words, if the size of every bit of the shares equals the bit size of the secret, the scheme is *ideal*.

2.6 Threshold Access Structure

Let $\mathcal{U}$ be a set of n participants. A (k, n) threshold secret sharing scheme is defined as the access structure

$$\Gamma = \{\mathcal{V} \subseteq \mathcal{U} : |\mathcal{V}| \geq k\}.$$

2.7 Shamir's Scheme and the Vandermonde Matrix

In Shamir's (k, n) threshold scheme, n shares $w_i \in \mathrm{GF}(p)$ for participants $P_{x_i} \in \mathcal{P}$ are generated from secret $s \in \mathrm{GF}(p)$. The dealer randomly selects $k-1$ elements $r_1, \cdots, r_{k-1}$ independently, with a uniform distribution over GF(p). The dealer then constructs a polynomial $f(x) = \sum_{i=1}^{k-1} r_i x^i + s \in \mathrm{GF}(p)[x]$, where $x = x_i$ for which all participant identities x_i are distinct and nonzero. Finally, the dealer sends w_i to P_{x_i} in private. We can view these shares as vector $\mathbf{w} = [w_0, w_1, \cdots, w_{n-1}]^\top = \mathbf{G} \cdot \mathbf{e}$, i.e.,

$$\mathbf{G} = \begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^{k-1} \\ 1 & x_1 & x_1^2 & \cdots & x_1^{k-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n-1} & x_{n-1}^2 & \cdots & x_{n-1}^{k-1} \end{bmatrix}, \ \mathbf{e} = \begin{bmatrix} s \\ r_1 \\ \vdots \\ r_{k-1} \end{bmatrix}.$$

Here, **G** is a Vandermonde matrix, which guarantees that any k rows are invertible. When k participants P_x for $x = t_0, t_1, \cdots, t_{k-1}$ cooperate to recover the secret, they pool their shares w_x together. Considering vector $\mathbf{w}_k = [w_{t_0}, w_{t_1}, \cdots, w_{t_{k-1}}]^\top$ and recovery matrix $\mathbf{G}_k$, consisting of the corresponding k rows from **G**, we can obtain secret s from $\mathbf{e} = \mathbf{G}_k^{-1} \cdot \mathbf{w}_k$.

2.8 Review of XOR-based Secret Sharing Schemes

In Kurihara et al.'s scheme [4], [5], n shares $w_i \in \{0, 1\}^{d(p-1)}$ for participants $P_{x_i} \in \mathcal{P}$ are generated from secret $s \in \{0, 1\}^{d(p-1)}$. The dealer equally divides secret s into $p-1$ blocks $s_1, \cdots, s_{p-1} \in \{0, 1\}^d$. We can view these blocks as vector $\mathbf{s} = (s_1, \cdots, s_{p-1})$. d is, for example, 8 and 64. The dealer also randomly selects $(k-1)p - 1$ pieces $r_0^0, \cdots, r_{p-2}^0, r_0^1, \cdots, r_{p-1}^1, \cdots, r_0^{k-2}, \cdots, r_{p-1}^{k-2}$ independently, with a uniform distribution over the finite set $\{0, 1\}^d$. We can view these pieces as vector **r** in the same manner. The dealer then generates shares $w_i = w_{(i,0)} \| \cdots \| w_{(i,p-2)}$ with a uniquely given generator matrix **G** corresponding to the (k, p) threshold scheme. Here, **G** consists of several $(p-1) \times p$ and $(p-$

$1)\times(p-1)$ block matrices over GF(2). Finally, the dealer sends w_i to P_i in private. We can construct a (k,n) threshold scheme using a (k,p) threshold scheme even if n is a composite number. We can view these shares as vector $\mathbf{w} = [\mathbf{w}_{(0)}, \mathbf{w}_{(1)}, \cdots, \mathbf{w}_{(n-1)}]^{\mathrm{T}} = \mathbf{G} \cdot [\mathbf{r}, \mathbf{s}]^{\mathrm{T}}$, where $\mathbf{w}_{(i)} = (w_{(i,0)}, \cdots, w_{(i,p-2)})$. When k participants P_x for $x = t_0, t_1, \cdots, t_{k-1}$ cooperate to recover the secret, they pool their shares w_x together. We equally divide each share into d-bit pieces and obtain recovery matrix $\mathbf{G}_k$. This $\mathbf{G}_k$ is not a square matrix. Therefore, we first obtain $\mathbf{G}'_k$, where we apply forward elimination of Gaussian elimination to $\mathbf{G}_k$. We then apply backward substitution to a part of $\mathbf{G}'_k$ that is required to obtain secret s. Finally, we obtain secret s.

(k, L, n) *ramp* schemes exhibit a trade-off between security and space efficiency. The secret can be recovered using any subset k of the n shares, but every subset comprising less than or equal to $k - L$ participants cannot obtain any information regarding the secret. If a (k, L, n) *ramp* scheme is linear, every further share reveals $\frac{1}{L}$ bits of information regarding the secret after $k - L$ shares are pooled. Therefore, we can view Kurihara et al.'s $(k, 1, n)$ *ramp* scheme [6] as an XOR-based (k, n) threshold scheme.

In Fujii et al.'s [3] and Kurihara et al.'s [6] schemes, recovery matrix $\mathbf{G}_k$ is a square matrix.

3. Our Proposed Scheme

In this section, we propose several theorems of circulant matrices to discuss the rank of recovery matrix $\mathbf{G}_k$. First, we analyze circulant matrices over GF(2) whose weight is two and the rank of the product of those matrices. Next, we analyze the rank of recovery matrix $\mathbf{G}_k$ that consists of $p\times p$ circulant matrices. Finally, in Section 5, we provide a new security analysis for XOR-based secret sharing schemes.

3.1 Rank of the Product of Circulant Matrices over GF(2) whose Weight is Two

We examine the associated polynomial of a circulant matrix whose weight is two to analyze the rank of the circulant matrix using Theorem 2.1.

Lemma 3.1. *Let* $a, b \geq 1$. $\gcd(x^a+1, x^b+1) = x^{\gcd(a,b)}+1$*, where* $x^a + 1, x^b + 1 \in \mathrm{GF}(2)[x]$.

Proof. It is easy to see that the equation is satisfied if $a = b$. Consider $a > b$. First, assuming polynomials $A(x), B(x)$ over a finite field, we can obtain $\gcd(A(x), B(x))$ with the following steps [21].

(1) If $B(x) = 0$, output $A(x)$ and exit.

(2) Compute $Q(x)$ and $R(x)$ such that $A(x) = B(x) \cdot Q(x) + R(x)$, where $\deg(R(x)) < \deg(B(x))$.

(3) $A(x) \leftarrow B(x), B(x) \leftarrow R(x)$ and go back to Step (1).

Next, we confine $A(x) = x^a + 1, B(x) = x^b + 1 \in \mathrm{GF}(2)[x]$. Let q, r be integers such that $a = b \cdot q + r$, where $r < b$. We can compute $Q(x) = \sum_{i=1}^{q} x^{a-b\cdot i}$, $R(x) = x^r + 1$ such that $A(x) = B(x) \cdot Q(x) + R(x)$. If $r = 0$, $R(x) = 0$. Here, we look at the degrees a and b of $A(x)$ and $B(x)$. We see that we take the following steps.

(1) If $b = 0$, output $x^a + 1$ and exit.

(2) Compute q and r such that $a = b \cdot q + r$, where $r < b$.

(3) $a \leftarrow b, b \leftarrow r$ and go back to Step (1).

This algorithm is identical to the Euclidean algorithm for computing the greatest common divisor (GCD) of integers a and b. Therefore, $\gcd(x^a + 1, x^b + 1) = x^{\gcd(a,b)} + 1$. □

Lemma 3.2. *Let* $0 \leq b < a < p$. $\gcd(x^a + x^b, x^p + 1) = x + 1$*, where* $x^a + x^b, x^p + 1 \in \mathrm{GF}(2)[x]$.

Proof. We obtain $x^a + x^b = x^b(x^{a-b} + 1)$. It is also easy to see that $\gcd(x^b, x^p + 1) = 1$. Next, $\gcd(a - b, p) = 1$ because p is a prime number. With Lemma 3.1, $\gcd(x^{a-b} + 1, x^p + 1) = x + 1$. The proof is thus complete. □

In Lemma 3.3 and Theorem 3.1, we consider the product of $p \times p$ circulant matrices over GF(2) whose weight is two.

Lemma 3.3. *Assume* $p \geq 3$. *Let* $0 \leq b < a < p, 0 \leq d < c < p$. $\gcd((x^a+x^b)\cdot(x^c+x^d), x^p+1) = x+1$*, where* $x^a+x^b, x^c+x^d, x^p+1 \in \mathrm{GF}(2)[x]$.

Proof. From Lemma 3.2, we obtain $\gcd(x^a + x^b, x^p + 1) = x + 1$ and $\gcd(x^c + x^d, x^p + 1) = x + 1$. Let $f(x) = x^{p-1} + \cdots + x + 1$, then $x^p + 1 = (x + 1)f(x)$. Because p is a prime number, $f(1) = 1$ and $f(x)$ does not have a divisor $x + 1$. The proof is thus complete. □

Theorem 3.1. *Assume* $p \geq 3$. *Let* $a \geq 0$ *and* $\mathbf{C}$ *be the circulant matrix of the product of* $p \times p$ *circulant matrices* $\mathbf{C}_0, \cdots, \mathbf{C}_a$ *over* GF(2) *whose weight is two, i.e.,*

$$\mathbf{C} = \prod_{l=0}^{a} \mathbf{C}_l = \prod_{l=0}^{a} (g_{i(l)} + g_{j(l)}),$$

where $i(l) \neq j(l)$. *Then the rank of* $\mathbf{C}$ *is* $p - 1$.

Proof. The associated polynomial of $\mathbf{C}_l$ is $c_l(x) = x^{i(l)} + x^{j(l)}$. The associated polynomial of $\mathbf{C}$ is also $c(x) = \prod_{l=0}^{a} c_l(x)$. Because $0 \leq i(l), j(l) < p$ and $i(l) \neq j(l)$, $\gcd(c(x), x^p + 1) = x + 1$ using Lemma 3.3. The rank of $\mathbf{C}$ is thus $p - 1$ from Theorem 2.1. □

3.2 Elementary Row Operations

We consider $n \cdot p \times k \cdot p$ matrix $\mathbf{G}$ that consists of $p \times p$ circulant matrices as a generator matrix of a (k, n) threshold secret sharing scheme, i.e.,

$$\mathbf{G} = \begin{bmatrix} 1 & g_0 & g_0^2 & \cdots & g_0^{k-1} \\ 1 & g_1 & g_1^2 & \cdots & g_1^{k-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & g_{n-1} & g_{n-1}^2 & \cdots & g_{n-1}^{k-1} \end{bmatrix}.$$

Later, we view matrix $\mathbf{G}$ as an $n \times k$ matrix to specify the i-th row and j-th column of $\mathbf{G}$. For example, we view g_1^2 as an element located at the first row and the second column of $\mathbf{G}$. Then, shares of participants $P_x \in \mathcal{P}$ are generated from the x-th row of $\mathbf{G}$. Assume that $x_i = g_{t_i}$. When k participants P_x for $x = t_0, t_1, \cdots, t_{k-1}$ cooperate to recover the secret, we can consider recovery matrix

$$\mathbf{G}_k = \begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^{k-1} \\ 1 & x_1 & x_1^2 & \cdots & x_1^{k-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{k-1} & x_{k-1}^2 & \cdots & x_{k-1}^{k-1} \end{bmatrix}.$$

Apparently, $\mathbf{G}_k$ is associated with a Vandermonde matrix, but there is a case in which a circulant matrix is not invertible. More specifically, to analyze the rank of $\mathbf{G}_k$ using elementary row operations, we first add the zeroth row to each i-th row for $i = 1, \cdots, k - 1$. As a result, we obtain

$$\mathbf{G}_k \overset{\circ}{\to} \begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^{k-1} \\ 0 & x_1 + x_0 & x_1^2 + x_0^2 & \cdots & x_1^{k-1} + x_0^{k-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & x_{k-1} + x_0 & x_{k-1}^2 + x_0^2 & \cdots & x_{k-1}^{k-1} + x_0^{k-1} \end{bmatrix}.$$

However, each element in the i-th row for $i = 1, \cdots, k-1$ cannot be multiplied by $(x_i + x_0)^{-1}$ from Lemma 2.2.

To resolve the preceding issue, we provide Theorem 3.2 and can transform the i-th elements of the first column to zeros with elementary row operations, where $i = 2, \cdots, k-1$.

Theorem 3.2. *Given $p \times p$ circulant matrices $g_a + g_b$ and $g_c + g_d$ over* GF(2) *whose weight is two, there exists $p \times p$ circulant matrix $\mathbf{T}_{a,b}^{c,d}$ over* GF(2) *such that $g_c + g_d = \mathbf{T}_{a,b}^{c,d}(g_a + g_b)$.*

Proof. Let $t = (b-a)^{-1}(d-c) \bmod p$. $a \not\equiv b \pmod p$ and $c \not\equiv d \pmod p$ because $g_a + g_b$ and $g_c + g_d$ are circulant matrices whose weight is two. Because $0 < (b-a)^{-1} < p$ and $0 < d-c < p$, we see that $0 < t < p$. If $t \neq 1$, $\mathbf{T}_{a,b}^{c,d} = g_c \sum_{i=0}^{t-1} g_{i(b-a)} g_{-a}$ because

$$g_{-a}(g_a + g_b) = g_0 + g_{b-a},$$
$$\sum_{i=0}^{t-1} g_{i(b-a)}(g_0 + g_{b-a}) = g_0 + g_{t(b-a)} = g_0 + g_{d-c},$$
$$g_c(g_0 + g_{d-c}) = g_c + g_d.$$

If $t = 1$, meaning that $b - a \equiv d - c \pmod p$, that equation $\mathbf{T}_{a,b}^{c,d}$ is also satisfied as $\mathbf{T}_{a,b}^{c,d} = g_{c-a}$ since $\mathbf{T}_{a,b}^{c,d}(g_a + g_b) = g_c + g_{b+c-a} = g_c + g_d$. The proof is thus complete. □

Later, we view $\mathbf{T}_{a,b}^{c,d} x_i$ as $\mathbf{T}_{t_a,t_b}^{t_c,t_d} x_i$ for $x_i = g_{t_i}$. Here, we provide a brief example to better understand the sequence of elementary matrix transformations. Let $\mathbf{X}_{(a,b)} = x_a + x_b$, as shown later in Definition A.1.1.

Example 3.1. *We convert $4p \times 4p$ matrix $\mathbf{G}_4$ to an upper triangular matrix. We take steps $\mathbf{G}_4 \overset{\circ}{\to} \mathbf{G}_4^{(1)} \overset{\circ}{\to} \mathbf{G}_4^{(2)} \overset{\circ}{\to} \mathbf{G}_4^{(3)}$ and then use Theorem 3.2 to transform $\mathbf{G}_4^{(1)}$ to $\mathbf{G}_4^{(2)}$ and $\mathbf{G}_4^{(2)}$ to $\mathbf{G}_4^{(3)}$, i.e.,*

$$\mathbf{G}_4^{(1)} = \begin{bmatrix} 1 & x_0 & x_0^2 & x_0^3 \\ 0 & x_1 + x_0 & x_1^2 + x_0^2 & x_1^3 + x_0^3 \\ 0 & x_2 + x_0 & x_2^2 + x_0^2 & x_2^3 + x_0^3 \\ 0 & x_3 + x_0 & x_3^2 + x_0^2 & x_3^3 + x_0^3 \end{bmatrix},$$

$$\mathbf{G}_4^{(2)} = \begin{bmatrix} 1 & x_0 & x_0^2 & x_0^3 \\ 0 & \mathbf{X}_{(1,0)} & \mathbf{X}_{(1,0)}\mathbf{X}_{(1,0)} & \mathbf{X}_{(1,0)}a_1 \\ 0 & 0 & \prod_{i=0}^{1} \mathbf{X}_{(2,i)} & \prod_{i=0}^{1} \mathbf{X}_{(2,i)}a_2 \\ 0 & 0 & \prod_{i=0}^{1} \mathbf{X}_{(3,i)} & \prod_{i=0}^{1} \mathbf{X}_{(3,i)}a_3 \end{bmatrix},$$

$$\mathbf{G}_4^{(3)} = \begin{bmatrix} 1 & x_0 & x_0^2 & x_0^3 \\ 0 & \mathbf{X}_{(1,0)} & \mathbf{X}_{(1,0)}\mathbf{X}_{(1,0)} & \mathbf{X}_{(1,0)}a_1 \\ 0 & 0 & \prod_{i=0}^{1} \mathbf{X}_{(2,i)} & \prod_{i=0}^{1} \mathbf{X}_{(2,i)}a_2 \\ 0 & 0 & 0 & \prod_{i=0}^{2} \mathbf{X}_{(3,i)} \end{bmatrix},$$

where $a_1 = x_1^2 + x_1 x_0 + x_0^2$, $a_2 = x_2 + x_1 + x_0$, and $a_3 = x_3 + x_1 + x_0$.

The rank of $\mathbf{G}_4$ equals the rank of upper triangular matrix $\mathbf{G}_4^{(3)}$, which is the sum of the ranks of the diagonal matrices. Each rank of the diagonal matrices except for the $p \times p$ identity matrix is $p-1$ from Theorem 3.1. Therefore, the rank of $\mathbf{G}_4$ is $4(p-1)+1$.

Next, we consider transforming $\mathbf{G}_k$ to an upper triangular matrix as a generalization. We define the matrix

$$\mathbf{M}^{(t)} \overset{\text{def}}{=} \left[\mathbf{M}_{(i,j)}^{(t)}\right]_{i=t,j=t}^{k-1,k-1} \quad (t = 1, \cdots, k-1)$$

and then represent the transformations as

$$\mathbf{G}_k \overset{\circ}{\to} \mathbf{G}_k^{(1)} = \left[\begin{array}{c:cccc} 1 & x_0 & \cdots & x_0^{k-1} \\ \hdashline 0 & \mathbf{M}_{(1,1)}^{(1)} & \cdots & \mathbf{M}_{(1,k-1)}^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \mathbf{M}_{(k-1,1)}^{(1)} & \cdots & \mathbf{M}_{(k-1,k-1)}^{(1)} \end{array}\right],$$

$$\mathbf{G}_k^{(1)} \overset{\circ}{\to} \cdots \overset{\circ}{\to} \mathbf{G}_k^{(t+1)} =$$

$$\left[\begin{array}{cccc:cc} 1 & x_0 & \cdots & & \cdots & x_0^{k-1} \\ & \mathbf{M}_{(1,1)}^{(1)} & \cdots & & \cdots & \mathbf{M}_{(1,k-1)}^{(1)} \\ & & \ddots & & & \vdots \\ & & & \mathbf{M}_{(t,t)}^{(t)} & \cdots & \mathbf{M}_{(t,k-1)}^{(t)} \\ \hdashline & \boldsymbol{O} & & & \mathbf{M}^{(t+1)} & \end{array}\right].$$

Step 3.1. *Here, we show the steps of the elementary row operations for $\mathbf{G}_k$.*

(*1*) *We add the zeroth row to each i-th row for $i = 1, \cdots, k-1$. We take this step only once.*

(*2*) *With Theorem 3.2, each element in the m-th row is multiplied by $\mathbf{T}_{m,0}^{i,0} \cdots \mathbf{T}_{m,m-1}^{i,m-1}$. The i-th row for $i = m+1, \cdots, k-1$ is replaced by the sum of that row and the m-th row. We repeat these steps for $m = 1, \cdots, k-2$, i.e.,*

$$\mathbf{M}_{(i,j)}^{(m+1)} = \mathbf{M}_{(i,j)}^{(m)} + \prod_{t=0}^{m-1} \mathbf{T}_{m,t}^{i,t} \mathbf{M}_{(m,j)}^{(m)}. \tag{1}$$

Theorem 3.3. *Given $\mathbf{G}_k \overset{\circ}{\to} \mathbf{G}_k^{(k-1)}$ using Step 3.1, $\mathbf{M}_{(m,m)}^{(m)} = \prod_{t=0}^{m-1} \mathbf{X}_{(m,t)}$, where $m = 1, \cdots, k-1$.*

We show the proof of Theorem 3.3 in Appendix A.1.

Theorem 3.4. *The rank of $\mathbf{G}_k$ is $k(p-1)+1$. The rank of $\mathbf{M}^{(1)}$ is then $(k-1)(p-1)$.*

Proof. If $p \geq 3$, each rank of $\mathbf{M}_{(m,m)}^{(m)}$, represented by Theorem 3.3, is $p-1$ from Theorem 3.1. If $p = 2$, we may only consider the rank of $\mathbf{M}_{(1,1)}^{(1)}$ because $k = 2$ and the rank is $p-1$ from Lemma 3.2. Therefore, the rank of $\mathbf{M}^{(1)}$ is $(k-1)(p-1)$ for both cases and the rank of $\mathbf{G}_k$ is $p + (k-1)(p-1) = k(p-1)+1$. □

4. Construction of the Secret Sharing Scheme

Our construction is the same as Kurihara et al.'s scheme [6] in the case of $L = 1$, as mentioned in Section 1.3. This section does not propose a new secret sharing scheme but describes the algorithm using different symbols.

The dealer equally divides secret $s \in \{0,1\}^{d(p-1)}$ into $p-1$ blocks $s_0, \cdots, s_{p-2} \in \{0,1\}^d$. d is, for example, 8 and 64. The dealer also randomly selects $(k-1)(p-1)$ pieces $r_0^0, \cdots, r_{p-2}^0, \cdots, r_0^{k-2}, \cdots, r_{p-2}^{k-2}$ independently, with a uniform distribution over the finite set $\{0,1\}^d$. The dealer then sends share $w_i \in \{0,1\}^{d(p-1)}$ of a (k, p) threshold scheme to participant $P_i \in \mathcal{P}$ in private. Therefore, we can construct a (k, n) threshold scheme using a (k, p) threshold scheme even if n is a composite number. Here, we define the vectors as mentioned below.

- $\mathbf{w}_{(i)} = \left[w_{(i,0)}, \cdots, w_{(i,p-2)}\right]^\mathsf{T}$
- $\mathbf{s} = (s_0, \cdots, s_{p-2})^\mathsf{T}$
- $\mathbf{r} = (r_0^0, \cdots, r_{p-2}^0, \cdots, r_0^{k-2}, \cdots, r_{p-2}^{k-2})^\mathsf{T}$
- $\mathbf{e} = \begin{bmatrix} \mathbf{r} \\ \mathbf{s} \end{bmatrix}$

Table 1 Distribution algorithm.

```
Input: $s \in \{0,1\}^{d(p-1)}$
Output: $(w_0, \cdots w_{n-1})$
1: $s_{p-1} \leftarrow \{0\}^d$, $s_0 \| \cdots \| s_{p-2} \leftarrow s$
2: for $i \leftarrow 0$ to $k-2$:
     for $j \leftarrow 0$ to $p-2$:
         $r^i_j \overset{\$}{\leftarrow} \{0,1\}^d$
     $r^i_{p-1} \leftarrow \{0\}^d$ (discard $r^0_{p-1}$)
3: for $i \leftarrow 0$ to $n-1$:
     for $j \leftarrow 0$ to $p-2$:
         $w_{(i,j)} \leftarrow \left(\bigoplus_{h=0}^{k-2} r^h_{h \cdot i+j}\right) \oplus s_{(k-1)\cdot i+j}$
     $w_i \leftarrow w_{(i,0)} \| \cdots \| w_{(i,p-2)}$
4: return $(w_0, \cdots w_{n-1})$
```

Table 2 Recovery algorithm.

```
Input: $(w_{t_0}, \cdots, w_{t_{k-1}})$
Output: $s$
1: for $i \leftarrow 0$ to $k-1$:
     $w_{(t_i,0)} \| \cdots \| w_{(t_i,p-2)} \leftarrow w_{t_i}$
2: $\mathbf{w}_k \leftarrow (w_{(t_0,0)}, \cdots, w_{(t_0,p-2)}, \cdots, w_{(t_{k-1},0)}, \cdots, w_{(t_{k-1},p-2)})^{\mathsf{T}}$
3: $\mathbf{M} \leftarrow F_{MAT}(t_0, \cdots, t_{k-1})$
4: $(s_0, \cdots, s_{p-2})^{\mathsf{T}} \leftarrow \mathbf{M} \cdot \mathbf{w}_k$
5: $s \leftarrow s_0 \| \cdots \| s_{p-2}$
6: return $s$

   $F_{MAT}(t_0, \cdots, t_{k-1})$
   F1: for $i \leftarrow 0$ to $k-1$:
         for $j \leftarrow 0$ to $p-2$:
             $\mathbf{v}_{(t_i,j)} \leftarrow w_{(t_i,j)} = \mathbf{v}_{(t_i,j)} \cdot \mathbf{e}$
   F2: $\bar{\mathbf{G}}_k \leftarrow (\mathbf{v}_{(t_0,0)}, \cdots, \mathbf{v}_{(t_{k-1},p-2)})^{\mathsf{T}}$
   F3: $\left[\begin{array}{cc:c} \mathbf{G'}_2 & \mathbf{G'}_1 & \mathbf{J}_1 \\ \boldsymbol{O} & \mathbf{G'}_0 & \mathbf{J}_0 \end{array}\right] = [\mathbf{G'}\ \mathbf{J}] \overset{\circ}{\leftarrow} \left[\bar{\mathbf{G}}_k\ \mathbf{I}_{k(p-1)}\right]$
   F4: $\left[\begin{array}{cc:c} \mathbf{G'}_2 & \mathbf{G'}_1 & \mathbf{J}_1 \\ \boldsymbol{O} & \mathbf{I}_{p-1} & \mathbf{M} \end{array}\right] \overset{\circ}{\leftarrow} [\mathbf{G'}\ \mathbf{J}]$
   F5: return $\mathbf{M}$
```

4.1 Share Generation

We consider $(p-1) \times (p-1)$ matrices $\bar{g}^j_i$ and $\bar{1}$ whose $p-1$-th row and $p-1$-th column are removed from each $p \times p$ circulant matrix of $\mathbf{G}$. We define matrix $\bar{\mathbf{G}}$ that consists of these matrices and then generate shares $\mathbf{w} = \bar{\mathbf{G}} \cdot \mathbf{e}$, i.e.,

$$\mathbf{w} = \begin{bmatrix} \mathbf{w}_{(0)} \\ \mathbf{w}_{(1)} \\ \vdots \\ \mathbf{w}_{(n-1)} \end{bmatrix} = \begin{bmatrix} \bar{1} & \bar{g}_0 & \bar{g}_0^2 & \cdots & \bar{g}_0^{k-1} \\ \bar{1} & \bar{g}_1 & \bar{g}_1^2 & \cdots & \bar{g}_1^{k-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \bar{1} & \bar{g}_{n-1} & \bar{g}_{n-1}^2 & \cdots & \bar{g}_{n-1}^{k-1} \end{bmatrix} \mathbf{e}.$$

Table 1 shows the distribution algorithm. Step 1 divides the secret equally into $p-1$ pieces of the d-bit sequence. Step 2 generates $(k-1)(p-1)$ pieces of the d-bit random number. Step 3 generates shares w_i. We can view this step as $w_{(i,j)} = \mathbf{v}_{(i,j)} \cdot \mathbf{e}$ with a uniquely given vector $\mathbf{v}_{(i,j)}$ corresponding to the (k,p) threshold scheme. For example, $\mathbf{v}_{(1,0)} = (1000\ 0100\ 0010)$ if $k = 3, p = 5$.

4.2 Recovery

Participants P_i for $i = t_0, \cdots, t_{k-1}$ cooperate to recover the secret. Assume that $\bar{x}_i = \bar{g}_{t_i}$. With

$$\bar{\mathbf{G}}_k = \begin{bmatrix} \bar{1} & \bar{x}_0 & \cdots & \bar{x}_0^{k-1} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{1} & \bar{x}_{k-1} & \cdots & \bar{x}_{k-1}^{k-1} \end{bmatrix}, \mathbf{w}_k = \begin{bmatrix} \mathbf{w}_{(t_0)} \\ \vdots \\ \mathbf{w}_{(t_{k-1})} \end{bmatrix},$$

we can recover secret s from $\mathbf{e} = \bar{\mathbf{G}}_k^{-1} \cdot \mathbf{w}_k$.

Table 2 shows the recovery algorithm. Step 1 equally divides each share into d-bit pieces. Step 2 generates the $k(p-1)$-dimensional vector $\mathbf{w}$. Step 3 obtains the matrix $\mathbf{M}$ using the function $F_{MAT}()$. Step 4 recovers $s_0, \cdots, s_{p-2}$ by calculating $\mathbf{M} \cdot \mathbf{w}_k$. Step 5 obtains secret s by concatenating $s_0, \cdots, s_{p-2}$. Step F1 obtains the vectors $\mathbf{v}_{(t_i,j)}$ such that $w_{(t_i,j)} = \mathbf{v}_{(t_i,j)} \cdot \mathbf{e}$. Step F2 obtains the matrix $\bar{\mathbf{G}}_k$. In Step F3, the matrix $\left[\bar{\mathbf{G}}_k\ \mathbf{I}_{k(p-1)}\right]$ is transformed to the row echelon form $[\mathbf{G'}\ \mathbf{J}]$. Matrices $\mathbf{G'}$ and $\mathbf{J}$ correspond to the matrices transformed from $\bar{\mathbf{G}}_k$ and $\mathbf{I}_{k(p-1)}$, respectively. We then view the matrix $[\mathbf{G'}\ \mathbf{J}]$ as a matrix divided into block matrices $\boldsymbol{O}, \mathbf{G'}_0, \mathbf{G'}_1, \mathbf{G'}_2, \mathbf{J}_0$, and $\mathbf{J}_1$. Step F4 transforms the matrix $[\mathbf{G'}_0\ \mathbf{J}_0]$ to the matrix $\left[\mathbf{I}_{p-1}\ \mathbf{M}\right]$. Step 5 outputs the $(p-1) \times k(p-1)$ matrix $\mathbf{M}$.

4.3 Brief Example of Share Generation and Recovery

Consider a $(3,5)$ threshold scheme. We generate each share $w_i = w_{(i,0)} \| \cdots \| w_{(i,3)}$ for participant P_i with $\mathbf{w} = \bar{\mathbf{G}} \cdot \mathbf{e}$, i.e.,

$$\mathbf{w} = \begin{bmatrix} \mathbf{w}_{(0)} \\ \hdashline \mathbf{w}_{(1)} \\ \hdashline \mathbf{w}_{(2)} \\ \hdashline \mathbf{w}_{(3)} \\ \hdashline \mathbf{w}_{(4)} \end{bmatrix} = \begin{bmatrix} w_{(0,0)} \\ w_{(0,1)} \\ w_{(0,2)} \\ w_{(0,3)} \\ \hdashline w_{(1,0)} \\ w_{(1,1)} \\ \vdots \\ w_{(3,2)} \\ w_{(3,3)} \\ \hdashline w_{(4,0)} \\ w_{(4,1)} \\ w_{(4,2)} \\ w_{(4,3)} \end{bmatrix} = \begin{bmatrix} 1000\ 1000\ 1000 \\ 0100\ 0100\ 0100 \\ 0010\ 0010\ 0010 \\ 0001\ 0001\ 0001 \\ \hdashline 1000\ 0100\ 0010 \\ 0100\ 0010\ 0001 \\ \vdots \\ 0010\ 1000\ 0001 \\ 0001\ 0100\ 0000 \\ \hdashline 1000\ 0000\ 0001 \\ 0100\ 1000\ 0000 \\ 0010\ 0100\ 1000 \\ 0001\ 0010\ 0100 \end{bmatrix} \begin{bmatrix} r^0_0 \\ r^0_1 \\ r^0_2 \\ r^0_3 \\ \hline r^1_0 \\ r^1_1 \\ r^1_2 \\ r^1_3 \\ \hline s_0 \\ s_1 \\ s_2 \\ s_3 \end{bmatrix}.$$

When P_0, P_3, and P_4 agree to recover the secret, the secret

$$\mathbf{s} = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{bmatrix} = \begin{bmatrix} 1110\ 0111\ 1001 \\ 1001\ 1011\ 0010 \\ 0101\ 1101\ 1000 \\ 0011\ 1110\ 1101 \end{bmatrix} \begin{bmatrix} \mathbf{w}_{(0)} \\ \mathbf{w}_{(3)} \\ \mathbf{w}_{(4)} \end{bmatrix}$$

can be recovered because

$$\left[\bar{\mathbf{G}}_3\ \mathbf{I}_{12}\right] = \begin{bmatrix} 1000\ 1000\ 1000 & 1000\ 0000\ 0000 \\ 0100\ 0100\ 0100 & 0100\ 0000\ 0000 \\ 0010\ 0010\ 0010 & 0010\ 0000\ 0000 \\ 0001\ 0001\ 0001 & 0001\ 0000\ 0000 \\ \hdashline 1000\ 0001\ 0100 & 0000\ 1000\ 0000 \\ 0100\ 0000\ 0010 & 0000\ 0100\ 0000 \\ 0010\ 1000\ 0001 & 0000\ 0010\ 0000 \\ 0001\ 0100\ 0000 & 0000\ 0001\ 0000 \\ \hdashline 1000\ 0000\ 0001 & 0000\ 0000\ 1000 \\ 0100\ 1000\ 0000 & 0000\ 0000\ 0100 \\ 0010\ 0100\ 1000 & 0000\ 0000\ 0010 \\ 0001\ 0010\ 0100 & 0000\ 0000\ 0001 \end{bmatrix},$$

$$\left[\bar{\mathbf{G}}_3\ \mathbf{I}_{12}\right] \overset{\circ}{\rightarrow} \left[\begin{array}{l} 1000\ 1000\ 1000\ \ 1000\ 0000\ 0000 \\ 0100\ 0100\ 0100\ \ 0100\ 0000\ 0000 \\ 0010\ 0010\ 0010\ \ 0010\ 0000\ 0000 \\ 0001\ 0001\ 0001\ \ 0001\ 0000\ 0000 \\ \hdashline 0000\ 1001\ 1100\ \ 1000\ 1000\ 0000 \\ 0000\ 0100\ 0110\ \ 0100\ 0100\ 0000 \\ 0000\ 0011\ 1111\ \ 1010\ 1010\ 0000 \\ 0000\ 0001\ 0111\ \ 0101\ 0101\ 0000 \\ \hdashline 0000\ 0000\ \underline{1000}\ \ \underline{1110\ 0111\ 1001} \\ 0000\ 0000\ \underline{0100}\ \ \underline{1001\ 1011\ 0010} \\ 0000\ 0000\ \underline{0010}\ \ \underline{0101\ 1101\ 1000} \\ 0000\ 0000\ \underline{0001}\ \ \underline{0011\ 1110\ 1101} \end{array}\right].$$

5. Proof of the Ideal Secret Sharing Scheme

In this section, we provide a new security analysis for XOR-based secret sharing schemes using our theorems, discussed in Section 3.

Theorem 5.1. *Assume that any set of k participants in* Γ *agrees to recover the secret. Then, correctness holds.*

Proof. $\bar{\mathbf{G}}_k$, shown in Section 4.2, is a $k(p-1) \times k(p-1)$ square matrix. We view $\bar{\mathbf{G}}_k$ as a $k \times k$ matrix, as described in Section 3.2. First, we add the zeroth row of $\bar{\mathbf{G}}_k$ to each i-th row of $\bar{\mathbf{G}}_k$ for $i = 1, \cdots, k-1$ and obtain

$$\bar{\mathbf{G}}_k \overset{\circ}{\rightarrow} \bar{\mathbf{G}}_k^{(1)} = \left[\begin{array}{c:ccc} \bar{1} & \bar{x}_0 & \cdots & \bar{x}_0^{k-1} \\ \hdashline 0 & \bar{\mathbf{M}}_{(1,1)}^{(1)} & \cdots & \bar{\mathbf{M}}_{(1,k-1)}^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \bar{\mathbf{M}}_{(k-1,1)}^{(1)} & \cdots & \bar{\mathbf{M}}_{(k-1,k-1)}^{(1)} \end{array}\right],$$

$$\bar{\mathbf{M}}^{(1)} \overset{\text{def}}{=} \left[\bar{\mathbf{M}}_{(i,j)}^{(1)}\right]_{i=1,j=1}^{k-1,k-1} = \begin{bmatrix} \bar{x}_1 + \bar{x}_0 & \cdots & \bar{x}_1^{k-1} + \bar{x}_0^{k-1} \\ \vdots & \ddots & \vdots \\ \bar{x}_{k-1} + \bar{x}_0 & \cdots & \bar{x}_{k-1}^{k-1} + \bar{x}_0^{k-1} \end{bmatrix}.$$

Next, we look at matrix $\bar{\mathbf{M}}^{(1)}$. We add a new $p-1$-th row to matrix $\bar{\mathbf{M}}_{(i,j)}^{(1)}$ as the sum of all $p-1$ rows of $\bar{\mathbf{M}}_{(i,j)}^{(1)}$. We also add a new $p-1$-th column in the same manner. We can see that $\bar{\mathbf{M}}^{(1)}$ is transformed to $\mathbf{M}^{(1)}$. These operations only add a linearly dependent row and a linearly dependent column. Therefore, the rank of $\bar{\mathbf{M}}^{(1)}$ equals that of $\mathbf{M}^{(1)}$ and the rank of $\bar{\mathbf{G}}_k$ is $k(p-1)$. That means $\bar{\mathbf{G}}_k$ is regular; therefore, we can always recover the secret in the access structure. The proof is thus complete. □

Theorem 5.2. *Let* $1 \leq L \leq k-1$. *Assume that any set of L participants* $T = \{P_{t_0}, \cdots, P_{t_{L-1}}\} \notin \Gamma$ *agrees to recover the secret. Then, perfect privacy holds.*

Proof. Consider $\mathbf{w}_L = \bar{\mathbf{G}}_L \cdot \mathbf{e}$, referring to Section 4.2. Suppose that s and the elements of $\mathbf{r}$ are mutually independent and that the elements of $\mathbf{r}$ are selected from the finite set $\{0,1\}^d$ with uniform probability $1/2^d$. Here, we consider $\bar{\mathbf{G}}_L = [\bar{\mathbf{U}}\ \bar{\mathbf{V}}]$ such that

$$\mathbf{w}_L = \bar{\mathbf{G}}_L \begin{bmatrix} \mathbf{r} \\ \mathbf{s} \end{bmatrix} = \bar{\mathbf{U}} \cdot \mathbf{r} \oplus \bar{\mathbf{V}} \cdot \mathbf{s}.$$

Because all rows of $\bar{\mathbf{U}}$ are linearly independent, including $\bar{\mathbf{U}} \neq \boldsymbol{O}$ for $p = 2$, all elements of the $L(p-1)$-dimensional vector $\bar{\mathbf{U}} \cdot \mathbf{r}$ are d-bit random numbers that are mutually independent and distributed uniformly over $\{0,1\}^d$. Therefore, the vector $\bar{\mathbf{U}} \cdot \mathbf{r}$ is uniformly distributed over $\{0,1\}^{dL(p-1)}$. Next, we suppose that $\mathbf{w}'$ denotes a fixed value of $\mathbf{w}_L$. $\mathbf{w}_L$ such that $\mathbf{w}_L = \mathbf{w}'$ can be obtained with uniform probability $(1/2)^{dL(p-1)}$ from any chosen $\bar{\mathbf{V}} \cdot \mathbf{s}$. Because $\mathbf{s}$ is independent of $\mathbf{w}_L$, we have $H(S|S_T) = H(S)$. □

6. Software Implementation

We evaluated schemes using one general purpose machine, as described in **Table 3**. Table 3 also shows the GCC options related to performance. We used a file size of 888,710 bytes as an example and provided some parameters of k and n.

Table 4 presents our experimental results, taken as the average of 30 experiments in each (k, n) parameter. We used $d = 64$ and used Xorshift for random number generation. The processing time to generate random numbers was included in the results of distribution. Here, the distribution and recovery algorithms of Ref. [4] and those of Ref. [5] are the same. We were able to clearly evaluate that the improved version was more efficient in distribution than Refs. [4], [5] if n, or p, is not large, where $k \geq 3$. The improved version means the scheme to which our approach is directly applied and its algorithm is identical to that of the $(k, 1, n)$ *ramp* scheme [6].

6.1 Comparison with Other Schemes

Consider Kurihara et al.'s schemes [4], [5] and [6] in the case of $L = 1$. They showed that each of their distribution algorithms requires an average of $O(kn)|s|$ bitwise XOR operations to generate n shares. Here, $|s|$ denotes the bit length of secret s, i.e., $|s| = \log_2 |S| = d(p-1)$. Next, in general, the size of the secret would exceed $|s|$. We refer to such an initial computation processed once for that recovery as a precomputation. They also showed that each of their recovery algorithms can estimate $O(k^3p^3)$ for the precomputation and $O(kp)|s|$. Therefore, no differences in efficiency between these algorithms exist, as shown in **Table 5**. However, the distribution algorithm of Ref. [6] requires fewer XOR operations than that of Refs. [4], [5]. **Table 6** shows the average number of XOR operations to generate one share. Fewer XOR operations can also yield a lower number of

Table 3 Test environment.

CPU	Intel ® Celeron ® Processor G1820
	2.70 GHz × 2, 2 MB cache
RAM	3.6 GB
OS	CentOS 7 Linux 3.10.0-229.20.1.el7.x86_64
Programing language	C
Compiler system	gcc 4.8.3 (-O3 -flto -DNDEBUG)

Table 4 Experimental results.

(k, n)	Distribution (Mbps)		Recovery (Mbps)	
	Improved ver.	Refs. [4], [5]	Improved ver.	Refs. [4], [5]
$(3, 5)$	1,553.63	1,262.45	7,640.79	7,678.44
$(3, 11)$	113.29	104.12	1,747.30	1,733.62
$(3, 43)$	11.14	11.00	209.30	201.18
$(41, 43)$	0.86	0.84	7.61	7.57
$(4, 5)$	1,116.33	664.57	4,531.31	4,409.23
$(5, 7)$	311.96	274.85	1,885.87	1,672.97

Table 5 Computational costs.

	Precomputation	Distribution	Recovery
Improved ver.	$O(k^3p^3)$	$O(kn)\|s\|$	$O(kp)\|s\|$
Refs. [4], [5]	$O(k^3p^3)$	$O(kn)\|s\|$	$O(kp)\|s\|$

Table 6 The average number of XOR operations to generate one share.

Improved ver.	$\left(k-1-\frac{k-1}{p}\right)\lvert s\rvert$
Refs. [4], [5]	$\left(k-1-\frac{1}{p}\right)\lvert s\rvert$

random numbers. In Refs. [4], [5], they reported that the average number is $(k-1-\frac{1}{p})|s|$. In our analysis for Ref. [6], the average number is $(k-1-\frac{k-1}{p})\frac{|s|}{L}$. Considering $L = 1$, the distribution algorithm of Ref. [6] requires $\frac{k-2}{p}|s|$ bitwise XOR operations fewer than that of Refs. [4], [5]. Theoretically, that shows if $k \geq 3$, the distribution algorithm of Ref. [6] requires fewer XOR operations than that of Refs. [4], [5]. If $k = 2$, the number of XOR operations of Ref. [6] and that of Refs. [4], [5] are the same. Then, if p is large, this computational advantage is not dominant because the total number of XOR operations is also larger. Therefore, we see that Ref. [6] can be theoretically predicted to be faster than Refs. [4], [5] if n, or p, is not large, where $k \geq 3$.

Consider Fujii et al.'s scheme [3]. $\mathbf{G}_k$ in Ref. [3] is considered close to a Vandermonde matrix that consists of circulant matrices in which each column is a cyclic shift of the adjacent column. This indicates that no differences between Refs. [3] and [6] with $L = 1$ exist in efficiency and the number of XOR operations for distribution.

7. Concluding Remarks

We propose several theorems of circulant matrices to discuss naturally and neatly the rank of recovery matrix $\mathbf{G}_k$ and then achieve a new security analysis for the XOR-based secret sharing schemes [3], [4], [5], [6]. We also show that we can view recovery matrix $\mathbf{G}_k$ as a Vandermonde matrix whose element is a circulant matrix. The XOR-based secret sharing schemes can be represented using the properties of circulant matrices and they can be tied to Shamir's scheme using a Vandermonde matrix.

Lastly, in our analysis, recovery matrix $\mathbf{G}_k$ in Refs. [4], [5] is not a Vandermonde matrix, but we can view the farthest block to the right as $\bar{g}_i^{p-1}$ instead of $\bar{g}_i^{k-1}$. Then, $\mathbf{G}_k$ in Ref. [3] is considered close to a Vandermonde matrix that consists of circulant matrices in which each column is a cyclic shift of the adjacent column.

Acknowledgments The authors would like to thank the anonymous reviewers for their insightful comments. This work was supported by JSPS KAKENHI Grant Number JP18K11306.

References

[1] Blakley, G.R.: Safeguarding cryptographic keys, *AFIPS*, Vol.48, pp.313–317 (1979).

[2] Shamir, A.: How to share a secret, *Comm. ACM*, Vol.22, No.11, pp.612–613 (1979).

[3] Fujii, Y., Tochikubo, K., Hosaka, N., Tada, M. and Kato, T.: (k, n) Threshold Schemes Using XOR Operations, *IEICE Technical Report*, ISEC 2007-5 (2007). (in Japanese)

[4] Kurihara, J., Kiyomoto, S., Fukushima, K. and Tanaka, T.: On a Fast (k, n)-Threshold Secret Sharing Scheme, *IEICE Trans. Fundamentals*, Vol.E91-A, No.9, pp.2365–2378 (2008).

[5] Kurihara, J., Kiyomoto, S., Fukushima, K. and Tanaka, T.: A New (k, n)-Threshold Secret Sharing Scheme and Its Extension, *ISC 2008*, LNCS 5222, pp.455–470 (2008).

[6] Kurihara, J., Kiyomoto, S., Fukushima, K. and Tanaka, T.: A Fast (k, L, n)-Threshold Ramp Secret Sharing Scheme, *IEICE Trans. Fundamentals*, Vol.E92-A, No.8, pp.1808–1821 (2009).

[7] Kurihara, J. and Uyematsu, T.: A Novel Realization of Threshold Schemes over Binary Field Extensions, *IEICE Trans. Fundamentals*, Vol.E94-A, No.6, pp.1375–1380 (2011).

[8] Shima, K. and Doi, H.: A Hierarchical Secret Sharing Scheme over Finite Fields of Characteristic 2, *Journal of Information Processing*, Vol.25, pp.875–883 (2017).

[9] Ikarashi, D., Tsuyuzaki, K. and Kawahara, Y.: SHSS: "Super High-speed (or, Sugoku Hayai) Secret Sharing" Library for Object Storage Systems, *SIG Technical Reports*, 2015-SPT-14 (2015). (in Japanese)

[10] Beimel, A.: Secret-Sharing Schemes, A Survey, *IWCC 2011*, LNCS 6639, pp.11–46 (2011).

[11] Blundo, C., De Santis, A., Gargano, L. and Vaccaro, U.: On the information rate of secret sharing schemes, *TCS*, Vol.154, pp.283–306 (1996).

[12] Blundo, C., De Santis, A., Gargano, L. and Vaccaro, U.: On the Information Rate of Secret Sharing Schemes, *Advances in Cryptology - CRYPTO '92*, LNCS 740, pp.149–169 (1993).

[13] McEliece, R.J. and Sarwate, D.V.: On sharing secrets and Reed-Solomon codes, *Comm. ACM*, Vol.24, No.9, pp.583–584 (1981).

[14] Lv, C., Jia, X., Lin, J., Jing, J., Tian, L. and Sun, M.: Efficient secret sharing schemes, *STA 2011*, CCIS 186, pp.114–121 (2011).

[15] Wang, Y. and Desmedt, Y.: Efficient secret sharing schemes achieving optimal information rate, *ITW 2014*, pp.516–520 (2014).

[16] Chen, L., Laing, T.M. and Martin, K.M.: Efficient, XOR-Based, Ideal (t, n) threshold Schemes, *CANS 2016*, LNCS 10052, pp.467–483 (2016).

[17] Deepika, M.P. and Sreekumar, A.: Secret sharing scheme using Gray code and XOR operation, *ICECCT 2017*, pp.1–5 (2017).

[18] Binu V.P. and Sreekumar A.: Simple and Efficient Secret Sharing Schemes for Sharing Data and Image, *IJCSIT*, Vol.6, No.1, pp.404–409 (2015).

[19] Ingleton, A.W.: The Rank of Circulant Matrices, *Journal of the London Mathematical Society*, Vol.1, No.4, pp.445–460 (1956).

[20] Fabšič, T., Hromada, V., Stankovski, P., Zajac, P., Guo, Q. and Johansson, T.: A Reaction Attack on the QC-LDPC McEliece Cryptosystem, *International Workshop on Post-Quantum Cryptography*, LNCS 10346, pp.51–68 (2017).

[21] Cohen, H.: A Course in Computational Algebraic Number Theory, Graduate texts in mathematics 138, Springer (1996).

[22] Shima, K. and Doi, H.: New Proof Techniques of XOR-based Secret Sharing Schemes, *CSS 2019*, 2F3-1, pp.839–846 (2019). (in Japanese)

Appendix

A.1 Proof of Theorem 3.3

Definition A.1.1. *Let $m \geq 0$.*

$$\mathbf{X}^m_{(a,b)} \overset{\text{def}}{=} x_a^m + x_b^m = \mathbf{X}_{(a,b)}\mathbf{H}_{(a,b)}$$

$$\mathbf{H}^m_{(a,b)} \overset{\text{def}}{=} \sum_{i=0}^{m} x_a^i x_b^{m-i}, \quad \mathbf{H}^0_{(a,b)} = 1$$

$$\mathbf{A}^m_{(a,b,c)} \overset{\text{def}}{=} \mathbf{H}^m_{(a,c)} + \mathbf{H}^m_{(b,c)}, \quad \mathbf{A}^0_{(a,b,c)} = 0$$

$$\bar{\mathbf{A}}^m_{(a,b,c)} \overset{\text{def}}{=} \sum_{i=0}^{m} x_c^i \mathbf{H}^{m-i}_{(a,b)}, \quad \bar{\mathbf{A}}^0_{(a,b,c)} = 1$$

$$\bar{\bar{\mathbf{A}}}^m_{(a,b,c,d)} \overset{\text{def}}{=} \bar{\mathbf{A}}^m_{(a,c,d)} + \bar{\mathbf{A}}^m_{(b,c,d)}, \quad \bar{\bar{\mathbf{A}}}^0_{(a,b,c,d)} = 0$$

Definition A.1.2. *Let $t(u) = \sum_{i=-1}^{u} t_i$ and $t_{-1} = 0$.*

$$\bar{\mathbf{S}}^{(m)}_{(c)} \overset{\text{def}}{=} \prod_{u=0}^{m-3}\Big(\sum_{t_u=0}^{c-m-t(u-1)} x_u^{t_u}\Big) \quad (m \geq 3)$$

$$\bar{\bar{\mathbf{S}}}^{(m)}_{(c)} \overset{\text{def}}{=} \prod_{u=0}^{m-3}\Big(\sum_{t_u=0}^{c-(m+1)-t(u-1)} x_u^{t_u}\Big) \quad (m \geq 3)$$

$$\mathbf{S}^{(m)}_{(a,b)} \overset{\text{def}}{=} \begin{cases} \bar{\mathbf{S}}^{(m)}_{(b)} \bar{\mathbf{A}}^{b-m-t(m-3)}_{(a,m-1,m-2)} & (m \geq 3) \\ \bar{\mathbf{A}}^{b-2}_{(a,1,0)} & (m = 2) \end{cases}$$

$$\mathbf{B}^{(m)}_{(a,b,c)} \overset{\text{def}}{=} \mathbf{S}^{(m)}_{(a,c)} + \mathbf{S}^{(m)}_{(b,c)} \quad (m \geq 2)$$

Lemma A.1.1. $\mathbf{S}^{(m)}_{(m,m)} = 1$ *holds for* $m \geq 2$.

Proof. $\mathbf{S}^{(2)}_{(2,2)}$ is easily seen to be true. Consider $m \geq 3$. We have that $\mathbf{S}^{(m)}_{(m,m)} = \prod_{u=0}^{m-3}(\sum_{t_u=0}^{-t(u-1)} x_u^{t_u})\bar{\mathbf{A}}^{-t(m-3)}_{(m,m-1,m-2)}$. This always gives $t_0 = 0$; $t_0, \cdots, t_{m-3}$ are also zeros, and $t(m-3) = 0$. Therefore, $\mathbf{S}^{(m)}_{(m,m)} = 1$ also holds for $m \geq 3$. □

Lemma A.1.2. $\mathbf{A}^m_{(a,b,c)} = \mathbf{X}_{(a,b)}\bar{\mathbf{A}}^{m-1}_{(a,b,c)}$ *holds for* $m \geq 1$.

Proof. Note that $\mathbf{A}^0_{(a,b,c)} = 0$. We have

$$\begin{aligned}\mathbf{A}^m_{(a,b,c)} &= \mathbf{X}^m_{(a,b)} + \mathbf{H}^m_{(a,c)} + x_a^m + \mathbf{H}^m_{(b,c)} + x_b^m \\ &= \mathbf{X}^m_{(a,b)} + x_c(\mathbf{H}^{m-1}_{(a,c)} + \mathbf{H}^{m-1}_{(b,c)}) = \mathbf{X}^m_{(a,b)} + x_c\mathbf{A}^{m-1}_{(a,b,c)}.\end{aligned}$$

Recursively, we then have

$$\begin{aligned}\mathbf{A}^m_{(a,b,c)} &= \mathbf{X}_{(a,b)}\mathbf{H}^{m-1}_{(a,b)} + x_c(\mathbf{X}_{(a,b)}\mathbf{H}^{m-2}_{(a,b)} + x_c(\cdots + x_c\mathbf{X}_{(a,b)})\cdots) \\ &= \mathbf{X}_{(a,b)}\sum_{i=0}^{m-1} x_c^i\mathbf{H}^{m-1-i}_{(a,b)} = \mathbf{X}_{(a,b)}\bar{\mathbf{A}}^{m-1}_{(a,b,c)}. \qquad \square\end{aligned}$$

Lemma A.1.3. $\bar{\bar{\mathbf{A}}}^m_{(a,b,c,d)} = \mathbf{X}_{(a,b)}\sum_{i=0}^{m-1} x_d^i\bar{\mathbf{A}}^{m-1-i}_{(a,b,c)}$ *holds for* $m \geq 1$.

Proof. Note that $\mathbf{A}^0_{(a,b,c)} = 0$. We have

$$\bar{\bar{\mathbf{A}}}^m_{(a,b,c,d)} = \bar{\mathbf{A}}^m_{(a,c,d)} + \bar{\mathbf{A}}^m_{(b,c,d)} = \sum_{i=0}^{m} x_d^i(\mathbf{H}^{m-i}_{(a,c)} + \mathbf{H}^{m-i}_{(b,c)})$$

$$= \sum_{i=0}^{m} x_d^i\mathbf{A}^{m-i}_{(a,b,c)} = \sum_{i=0}^{m-1} x_d^i\mathbf{A}^{m-i}_{(a,b,c)} = \mathbf{X}_{(a,b)}\sum_{i=0}^{m-1} x_d^i\bar{\mathbf{A}}^{m-1-i}_{(a,b,c)}. \qquad \square$$

Theorem A.1.1. $\mathbf{B}^{(m)}_{(a,m,c)} = \mathbf{X}_{(a,m)}\mathbf{S}^{(m+1)}_{(a,c)}$ *holds for* $m \geq 2$.

Proof. This equation holds for $m = 2$. We have

$$\begin{aligned}\mathbf{B}^{(2)}_{(a,2,c)} &= \mathbf{S}^{(2)}_{(a,c)} + \mathbf{S}^{(2)}_{(2,c)} = \bar{\mathbf{A}}^{c-2}_{(a,1,0)} + \bar{\mathbf{A}}^{c-2}_{(2,1,0)} = \bar{\bar{\mathbf{A}}}^{c-2}_{(a,2,1,0)} \\ &= \mathbf{X}_{(a,2)}\sum_{t_0=0}^{c-3} x_0^{t_0}\bar{\mathbf{A}}^{c-3-t_0}_{(a,2,1)} = \mathbf{X}_{(a,2)}\bar{\mathbf{S}}^{(3)}_{(c)}\bar{\mathbf{A}}^{c-3-t_0}_{(a,2,1)} = \mathbf{X}_{(a,2)}\mathbf{S}^{(3)}_{(a,c)}\end{aligned}$$

from Definition A.1.2 and Lemma A.1.3. Next, consider $m \geq 3$. Because the transformation of $\bar{\mathbf{S}}^{(m)}_{(c)}$ to $\bar{\bar{\mathbf{S}}}^{(m)}_{(c)}$ is true, we can have

$$\begin{aligned}\mathbf{B}^{(m)}_{(a,b,c)} &= \mathbf{S}^{(m)}_{(a,c)} + \mathbf{S}^{(m)}_{(b,c)} = \bar{\mathbf{S}}^{(m)}_{(c)}(\bar{\mathbf{A}}^{c-m-t(m-3)}_{(a,m-1,m-2)} + \bar{\mathbf{A}}^{c-m-t(m-3)}_{(b,m-1,m-2)}) \\ &= \bar{\mathbf{S}}^{(m)}_{(c)}\bar{\bar{\mathbf{A}}}^{c-m-t(m-3)}_{(a,b,m-1,m-2)} = \bar{\bar{\mathbf{S}}}^{(m)}_{(c)}\bar{\bar{\mathbf{A}}}^{c-m-t(m-3)}_{(a,b,m-1,m-2)}.\end{aligned}$$

Considering the transformation for $m = 3$ as an example,

$$\mathbf{B}^{(3)}_{(a,b,c)} = \sum_{t_0=0}^{c-3} x_0^{t_0}\bar{\bar{\mathbf{A}}}^{c-3-t_0}_{(a,b,2,1)} = \sum_{t_0=0}^{c-4} x_0^{t_0}\bar{\bar{\mathbf{A}}}^{c-3-t_0}_{(a,b,2,1)}$$

because $\bar{\bar{\mathbf{A}}}^0_{(a,b,2,1)} = 0$ for $t_0 = c-3$. In general, we need to multiply $x_0^{t_0}\cdots x_{m-3}^{t_{m-3}}$ by $\bar{\bar{\mathbf{A}}}^{c-m-t(m-3)}_{(a,b,m-1,m-2)}$ for $\bar{\mathbf{S}}^{(m)}_{(c)}$. Because $\bar{\bar{\mathbf{A}}}^0_{(a,b,c,d)} = 0$ for $t(m-3) = c-m$, we can transform $\bar{\mathbf{S}}^{(m)}_{(c)}$ to $\bar{\bar{\mathbf{S}}}^{(m)}_{(c)}$. Next, let $b = m$ and $m' = c - (m+1) - t(m-3)$. Then, we have

$$\mathbf{B}^{(m)}_{(a,m,c)} = \bar{\bar{\mathbf{S}}}^{(m)}_{(c)}\bar{\bar{\mathbf{A}}}^{c-m-t(m-3)}_{(a,m,m-1,m-2)} = \mathbf{X}_{(a,m)}\bar{\bar{\mathbf{S}}}^{(m)}_{(c)}\sum_{i=0}^{m'} x_{m-2}^i\bar{\mathbf{A}}^{m'-i}_{(a,m,m-1)}$$

from Lemma A.1.3. Algebraically, we have

$$\mathbf{B}^{(m)}_{(a,m,c)} = \mathbf{X}_{(a,m)}\prod_{u=0}^{m-3}\left(\sum_{t_u=0}^{c-(m+1)-t(u-1)} x_u^{t_u}\right)\sum_{i=0}^{m'} x_{m-2}^i\bar{\mathbf{A}}^{m'-i}_{(a,m,m-1)}$$

$$= \mathbf{X}_{(a,m)}\prod_{u=0}^{m-3}\left(\sum_{t_u=0}^{c-(m+1)-t(u-1)} x_u^{t_u}\right)\sum_{t_{m-2}=0}^{c-(m+1)-t((m-2)-1)} x_{m-2}^{t_{m-2}}\bar{\mathbf{A}}^{m'-t_{m-2}}_{(a,m,m-1)}$$

$$= \mathbf{X}_{(a,m)}\prod_{u=0}^{m-2}\left(\sum_{t_u=0}^{c-(m+1)-t(u-1)} x_u^{t_u}\right)\bar{\mathbf{A}}^{c-(m+1)-t(m-2)}_{(a,m,m-1)}$$

$$= \mathbf{X}_{(a,m)}\bar{\mathbf{S}}^{(m+1)}_{(c)}\bar{\mathbf{A}}^{c-(m+1)-t(m-2)}_{(a,m,m-1)} = \mathbf{X}_{(a,m)}\mathbf{S}^{(m+1)}_{(a,c)}. \qquad \square$$

Finally, we prove Theorem 3.3.

Proof. We prove that $\mathbf{M}^{(m)}_{(m,m)} = \prod_{t=0}^{m-1}\mathbf{X}_{(m,t)}$ for $m = 1, \cdots, k-1$. If $m = 1$, the equation $\mathbf{M}^{(1)}_{(1,1)} = \mathbf{X}_{(1,0)}$ holds, because the actual $\mathbf{M}^{(1)}_{(i,j)} = \mathbf{X}^j_{(i,0)}$. Consider $m = 2, \cdots, k-1$. Using $\mathbf{S}^{(m)}_{(m,m)} = 1$ from Lemma A.1.1, we consider whether the statement

$$\mathbf{M}^{(m)}_{(i,j)} = \prod_{t=0}^{m-1}\mathbf{X}_{(i,t)}\mathbf{S}^{(m)}_{(i,j)} \qquad \text{(A.1)}$$

for $i, j \in \{m, \cdots, k-1\}$ holds for $m = 2, \cdots, k-1$ by mathematical induction. Here, Eq. (1) represents the steps of the elementary row operations for $\mathbf{G}_k$ and is true. In the base case, we have

$$\begin{aligned}\mathbf{M}^{(2)}_{(i,j)} &= \mathbf{M}^{(1)}_{(i,j)} + \mathbf{T}^{i,0}_{1,0}\mathbf{M}^{(1)}_{(1,j)} = \mathbf{X}^j_{(i,0)} + \mathbf{T}^{i,0}_{1,0}\mathbf{X}^j_{(1,0)} \\ &= \mathbf{X}_{(i,0)}(\mathbf{H}^{j-1}_{(i,0)} + \mathbf{H}^{j-1}_{(1,0)}) = \mathbf{X}_{(i,0)}\mathbf{A}^{j-1}_{(i,1,0)} \\ &= \mathbf{X}_{(i,0)}\mathbf{X}_{(i,1)}\bar{\mathbf{A}}^{j-2}_{(i,1,0)} = \prod_{t=0}^{1}\mathbf{X}_{(i,t)}\mathbf{S}^{(2)}_{(i,j)}\end{aligned}$$

using Eq. (1), the actual values, and Definitions A.1.1 and A.1.2. Therefore, the statement (A.1) holds for $m = 2$. Next, consider the inductive step. Assuming the induction hypothesis that the statement is true, it must be shown that $\mathbf{M}^{(m+1)}_{(i,j)}$ is true. Algebraically, we have

$$\begin{aligned}\mathbf{M}^{(m+1)}_{(i,j)} &= \mathbf{M}^{(m)}_{(i,j)} + \prod_{t=0}^{m-1}\mathbf{T}^{i,t}_{m,t}\mathbf{M}^{(m)}_{(m,j)} \\ &= \prod_{t=0}^{m-1}\mathbf{X}_{(i,t)}\mathbf{S}^{(m)}_{(i,j)} + \prod_{t=0}^{m-1}\mathbf{T}^{i,t}_{m,t}\prod_{t=0}^{m-1}\mathbf{X}_{(m,t)}\mathbf{S}^{(m)}_{(m,j)} \\ &= \prod_{t=0}^{m-1}\mathbf{X}_{(i,t)}\mathbf{S}^{(m)}_{(i,j)} + \prod_{t=0}^{m-1}\mathbf{X}_{(i,t)}\mathbf{S}^{(m)}_{(m,j)} \\ &= \prod_{t=0}^{m-1}\mathbf{X}_{(i,t)}\mathbf{B}^{(m)}_{(i,m,j)} = \prod_{t=0}^{m}\mathbf{X}_{(i,t)}\mathbf{S}^{(m+1)}_{(i,j)}\end{aligned}$$

using Eq. (1), the induction hypothesis, Theorem 3.2, Definition A.1.2, and Theorem A.1.1. This shows that $\mathbf{M}^{(m+1)}_{(i,j)}$ holds. Because both the base case and the inductive step are performed, the statement (A.1) holds and the proof is complete. □

Koji Shima received his B.S. degree from Tokyo University of Science in 1997. He received his M.S. degree and Ph.D. from Institute of Information Security in 2016 and 2019, respectively. His research interests include information security and network protocols.

Hiroshi Doi received his B.S. degree in Mathematics from Okayama University in 1988, M.I.S. degree in Information Science from JAIST in 1994, and D.S. degree from Okayama University in 2000, respectively. He is currently a professor of Graduate School of Information Security, Institute of Information Security. His research interests include information security and cryptography.