Hindawi
Security and Communication Networks
Volume 2021, Article ID 6678345, 9 pages
https://doi.org/10.1155/2021/6678345

Research Article

Grouped Secret Sharing Schemes Based on Lagrange Interpolation Polynomials and Chinese Remainder Theorem

Fuyou Miao,[1] Yue Yu,[1] Keju Meng,[1] Yan Xiong,[1] and Chin-Chen Chang[2,3]

[1]*University of Science and Technology of China, Hefei, Anhui, China*
[2]*Feng Chia University, Taichung, Taiwan*
[3]*Hangzhou Dianzi University, Hangzhou, China*

Correspondence should be addressed to Keju Meng; mkj@mail.ustc.edu.cn

Received 16 November 2020; Revised 7 March 2021; Accepted 23 March 2021; Published 5 April 2021

Academic Editor: Angel M. Del Rey

In a (t,n) threshold secret sharing (SS) scheme, whether or not a shareholder set is an authorized set totally depends on the number of shareholders in the set. When the access structure is not threshold, *(t,n)* threshold SS is not suitable. This paper proposes a new kind of SS named grouped secret sharing (GSS), which is specific multipartite SS. Moreover, in order to implement GSS, we utilize both Lagrange interpolation polynomials and Chinese remainder theorem to design two GSS schemes, respectively. Detailed analysis shows that both GSS schemes are correct and perfect, which means any authorized set can recover the secret while an unauthorized set cannot get any information about the secret.

1. Introduction

The notion of secret sharing (SS) was first introduced by Shamir [1] and Blakley [2], respectively, in 1979. In an SS scheme, a dealer D divides a secret s into some pieces $s_1, s_2, \ldots, s_n$. Each piece s_i is called a share of the secret. Then, the dealer can design an access structure $\Gamma = \{\mathscr{A}_1, \mathscr{A}_2, \ldots, \mathscr{A}_m\}$, where $\mathscr{A}_i$ is a minimal authorized set, i.e., any superset of $\mathscr{A}_i$ can recover the secret. According to the access structure, the dealer sends each shares to the corresponding shareholder in private. After share generation, if a shareholder set $\mathscr{P}$ is a superset of an authorized set, i.e., $\mathscr{A}_i \subset \mathscr{P}$, $\mathscr{P}$ can reconstruct the secret as long as the shareholders in $\mathscr{A}_i$ release shares to the others in $\mathscr{P}$. If there does not exist any $\mathscr{A}_i$ such that $\mathscr{A}_i \subset \mathscr{P}$, the secret cannot be recovered by $\mathscr{P}$.

SS schemes are classified into many types, such as (t,n) threshold SS [3, 4], weighted threshold SS [5, 6], hierarchical threshold SS [7, 8], multilevel threshold SS [9, 10], multipartite SS [11, 12], and so on. The most classical SS is (t,n) threshold SS. There are n shareholders in a (t,n) threshold SS scheme. A dealer divides a secret s into n shares and sends each share to a shareholder securely. Then, any t or more than t shareholders can collaborate to obtain the secret s by pooling their shares together while any up to $t-1$ shareholders cannot. The value of t is called the threshold of the SS scheme.

Obviously, in a (t,n) threshold SS scheme, whether or not a shareholder set is able to recover the secret is totally dependent on the number of shareholders in the set. Hence, (t,n) SS cannot work in many cases. For example, suppose a big company BC consists of five constituent companies which share the final decision rights of BC equally. Each constituent company has several shareholders who can represent the constituent company to confer with representatives of the other constituent companies on the final decision of BC. Although different shareholders in a constituent company own different shares, they have the same rights. In such a scene, if a shareholder set is able to make a decision of BC, at least five shareholders are included in the set. However, it is far from that any five shareholders can do that. Only if a shareholder set includes the representatives from all the five constituent companies, the set can make a decision of BC. Therefore, the given access structure is not just threshold.

Farras et al. [11] first proposed the notion of multipartite SS which can solve the above problem. In a multipartite SS scheme, shareholders are divided into several disjoint partitions and each partition has a part access structure. If a shareholder set satisfies all the part access structures, it can recover the secret. But as long as the shareholder set does not satisfy any one part access structure of a partition, the secret cannot be obtained. Later, Tassa and Dyn [12] and Hsu and Harn [13] utilized bivariate interpolation and Chinese remainder theorem (CRT) to implement multipartite SS schemes. Obviously, if each threshold of part access structure equals one, the above problem can be solved. However, in the multipartite SS schemes, if the threshold is equal to one in a partition, all the shareholders in the partition get the same share. In terms of security, different shareholders are supposed to keep different shares even when they are in the same partition. For this purpose, we propose a new SS named grouped secret sharing (GSS) in this paper.

Informally, suppose that there are totally n shareholders in a SS scheme. They are divided into m disjoint groups. Each shareholder keeps a distinct share and belongs to a group. If a shareholder set shares at least one shareholder with every group, the shareholder set is allowed to recover the secret. Otherwise, the secret cannot be reconstructed. Then, the SS scheme is a GSS scheme.

In order to implement GSS, this paper uses Lagrange interpolation polynomials (LIPs) and Chinese remainder theorem (CRT) to design two GSS schemes, respectively. Thereinto, LIP as a method of linear combination plays an important role in numerical analysis. Shamir first used it to design a (t, n) threshold SS scheme [1] in 1979. Later, LIP became the most common tool to design SS schemes because it is very simple and efficient. There are many schemes [14, 15] based on LIP. CRT is used to solve systems of linear congruence equations. Mignotte first proposed a (t, n) threshold SS scheme [16] based on CRT in 1982, but the scheme is ramp instead of perfect because even an unauthorized shareholder set can obtain part of information about the secret. Asmuth and Bloom modified Mignotte's scheme to give a perfect (t, n) threshold SS scheme [17]. There are also many other kinds of SS schemes [18, 19] based on CRT.

Based on the above, the contributions of the paper are listed as follows:

(1) The paper proposes a kind of SS named grouped secret sharing (GSS).

(2) Both LIP and CRT are utilized to construct two GSS schemes.

(3) Although shareholders in a same group keep different shares, each one of them can represent the group to participate in secret reconstruction.

The outline of the paper is as follows. In the next section, we provide the notion of CRT, Shamir (t, n) SS scheme, Asmuth–Bloom (t, n) SS scheme, and the formal definition of GSS as the preliminaries. A GSS scheme based on LIP and correlative correctness and security analyses are shown in Section 3. Analogously, a GSS scheme based on CRT and correlative analyses are given in Section 4. For a better illustration, we give two numerical examples in Section 5. Some discussions about perfectness and information rate are shown in Section 6. We conclude the work in Section 7.

2. Preliminaries

In this section, we introduce some preliminaries including CRT, Shamir (t, n) SS scheme, Asmuth–Bloom (t, n) SS scheme, and the formal definition of GSS.

2.1. Chinese Remainder Theorem (CRT) [20]. Given the following system of linear congruence equations:

$$\begin{cases} x \equiv a_1 \bmod m_1; \\ x \equiv a_2 \bmod m_2; \\ \vdots \\ x \equiv a_n \bmod m_n, \end{cases} \tag{1}$$

if all moduli are pairwise co-prime, i.e., $\gcd(m_i, m_j) = 1$ for $i \neq j$, CRT illustrates that the system must have solutions for any integer of $a_1, a_2, \ldots, a_n$. Define $M = m_1, m_2, \ldots, m_n$, $M_i = M/m_i$, and $t_i \equiv M_i^{-1} \bmod m_i$ for $i = 1, \ldots, n$. Then, the system has an unique solution in $\mathbb{Z}_M$:

$$x = \left(\sum_{i=1}^{n} a_i t_i M_i \right) \bmod M. \tag{2}$$

2.2. Shamir's (t, n) Threshold SS Scheme. In Shamir's (t, n) threshold SS scheme, there are n shareholders $\mathcal{U} = \{U_1, U_2, \ldots, U_n\}$ and a dealer D who is trusted by all shareholders. The scheme consists of two algorithms.

2.2.1. Share Generation. The dealer D randomly selects a polynomial $f(x)$ of degree $t-1$: $f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{t-1} x^{t-1} \bmod p$, where all coefficients are in $\mathbb{Z}_p$. The secret is constant term of $f(x)$, i.e., $s = f(0) = a_0$. D picks n different positive integers to compute n shares $s_i = f(x_i)$ for $i = 1, 2, \ldots, n$, where x_i is public information associated with shareholder U_i. Then, the dealer D securely sends the share s_i to the corresponding shareholder U_i.

2.2.2. Secret Reconstruction. Assume that $m (m \geq t)$ shareholders $U_1, U_2, \ldots, U_m$ work together to recover the secret. Each shareholder U_i releases its share s_i to the others. After a shareholder receives the other $m-1$ shares, it can use LIP to recover the secret:

$$s = f(0) = \sum_{i=1}^{m} f(x_i) \prod_{j=1, j \neq i}^{m} \frac{x_j}{x_j - x_i} \bmod p. \tag{3}$$

2.3. Asmuth–Bloom (t, n) Threshold SS Scheme. There is a dealer D and n shareholders $\mathcal{U} = \{U_1, U_2, \ldots, U_n\}$ in

Asmuth–Bloom (t, n) threshold SS scheme which also consists of two algorithms.

2.3.1. Share Generation. At first, the dealer D picks a prime number p_0 and a sequence of pairwise co-prime positive integers $p_1, p_2, \ldots, p_n$ with $p_1 < p_2 < \cdots < p_n$, $p_0, p_{n-t+2}, p_{n-t+1}, \ldots, p_n < p_1, p_2, \ldots, p_t$, and $\gcd(p_0, p_i) = 1$ for $i = 1, 2, \ldots, n$. Then, D picks a random integer α and a secret s in $\mathbb{Z}_{p_0}$, such that $s + \alpha p_0 < p_1, p_2, \ldots, p_t$. Next, D computes n shares $s_i = (s + \alpha p_0) \bmod p_i$ for $i = 1, 2, \ldots, n$, where p_i is public information associated with shareholder U_i. Finally, the dealer D securely sends the share s_i to the corresponding shareholder U_i.

2.3.2. Secret Reconstruction. Assume that $m (m \geq t)$ shareholders $U_1, U_2, \ldots, U_m$ want to recover the secret. Each of them releases its share to the others. After a shareholder receives the other $m - 1$ shares, he gets a system of linear congruence equations:

$$\begin{cases} s + \alpha p_0 \equiv s_1 \bmod p_1; \\ s + \alpha p_0 \equiv s_2 \bmod p_2; \\ \vdots \\ s + \alpha p_0 \equiv s_m \bmod p_m. \end{cases} \tag{4}$$

Using the standard CRT, the value of $s + \alpha p_0$ can be computed as

$$s + \alpha p_0 = \sum_{i=1}^{m} \frac{P}{p_i} y_i s_i \bmod P, \tag{5}$$

where $P = p_1, p_2, \ldots, p_m$ and $y_i (P/p_i) \bmod p_i = 1$. Then, the secret s can be obtained as $s = s + \alpha p_0 \bmod p_0$.

2.4. Grouped Secret Sharing (GSS). In the following, we give a formal definition of GSS.

Definition 1 (grouped secret sharing). For an SS scheme, let $\mathcal{U}$ be a set of n shareholders and assume that $\mathcal{U}$ is composed of m disjoint groups, i.e., $\mathcal{U} = \cup_{i=1}^{m} \mathcal{U}_i$, where $\mathcal{U}_i$ is a set of n_i *shareholders such that* $n = n_1 + n_2 + \cdots + n_m$ and $\mathcal{U}_i \cap \mathcal{U}_j = \varnothing$ for all $1 \leq i \leq m$, $1 \leq j \leq m$ and $i \neq j$. Every shareholder keeps a unique share. If the access structure Γ of the scheme is shown as

$$\Gamma = \{\mathcal{P} | \mathcal{P} \subset \mathcal{U}, \mathcal{P} \cap \mathcal{U}_i \neq \varnothing, \quad \forall i \in \{1, 2, \ldots, m\}\}, \tag{6}$$

where $\mathcal{P}$ is a shareholder set.

In other words, if $\mathcal{P}$ can reconstruct the secret, it must share at least one shareholder with each of the m disjoint groups. If so, the SS scheme is a GSS scheme. Figure 1 shows the model of GSS.

2.5. Introduction to Lattice. Because some security analyses need to use lattice, we give some definitions about lattice.

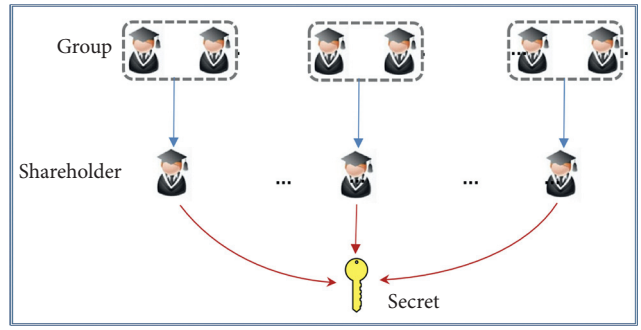

Figure 1: The model of GSS.

Definition 2 (lattice). Given n linearly independent vectors $\mathbf{b}_1, \mathbf{b}_1, \ldots, \mathbf{b}_n \in \mathbb{Z}^m$, the lattice generated by the n vectors is

$$\mathcal{L}(\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n|) = \left\{ \sum x_i b_i x_i | \in \mathbb{Z} \right\}. \tag{7}$$

Vectors $\mathbf{b}_1, \mathbf{b}_1, \ldots, \mathbf{b}_n \in \mathbb{Z}^m$ are the basis of L.

Definition 3 (the closest vector problem). Given a vector $\mathbf{w}$ that is not in a lattice L, find a vector $\mathbf{v} \in L$ that is closest to $\mathbf{w}$.

In the lattice with high dimensions, it is difficult to solve CVP in polynomial time. However, CVP is solvable in a reduced basis with low dimensions. LLL algorithm was proposed by Lenstra, Lenstra, and Lovász, and thus it is called LLL algorithm. LLL algorithm uses Schmidt orthogonalization repeatedly to obtain a reduced basis. The detailed algorithm is shown in paper [21].

3. GSS Based on LIP

In this section, we first show how to implement a GSS based on Shamir's (t, n) threshold SS scheme. The correctness and security analyses are given in the next two subsections, respectively.

3.1. Implementing a GSS Based on Shamir SS Scheme. According to the above definition of GSS, any shareholder U_k^i can represent the group $\mathcal{U}_i$ to participate in secret reconstruction but different shareholders should keep different shares even when they are in a same group. Hence, this paper focuses on how to generate different shares in a group while all shares can be used to recover the same secret.

Our GSS scheme based on LIP consists of three algorithms: (1) main share generation for a group; (2) subshare generation for a shareholder; and (3) secret reconstruction.

3.1.1. Main Share Generation for a Group. Let $\mathcal{U}$ be a set of n shareholders. All the shareholders are divided into m disjoint groups $\mathcal{U} = \{\mathcal{U}_1 \cup \mathcal{U}_2 \cup \cdots \cup \mathcal{U}_m\}$. A shareholder only belongs to one group, i.e., $\mathcal{U}_i \cap \mathcal{U}_j = \varnothing$ if $i \neq j$. Each group has n_i shareholders such that $n = n_1 + n_2 + \cdots + n_m$.

A dealer D chooses two prime numbers p and g with $p > mg^2$. Then, D randomly picks a polynomial $f(x)$ of degree $m - 1$: $f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{m-1} x^{m-1} \bmod p$, where $a_0 \in \mathbb{Z}_g$ but the other coefficients are limited in $\mathbb{Z}_p$,

i.e., $a_i \in \mathbb{Z}_p$ for $i = 1, 2, \ldots, m-1$. The secret s is equal to the constant term of $f(x)$, i.e., $s = f(0) = a_0$.

As for a group $\mathcal{U}_i$, the dealer D selects a positive integer x_i from $\mathbb{Z}_p$ as public information associated with $\mathcal{U}_i$, where $x_i \neq 0$ and $x_i \neq x_j$ if $i \neq j$. Then, D computes a main share for group $\mathcal{U}_i$ as $s_i = f(x_i)$.

3.1.2. Subshare Generation for a Shareholder. After Algorithm 1, each group $\mathcal{U}_i$ is allocated a main share. If there just exists one shareholder in a group, the dealer can use 0 as the random integer to generate a subshare for the only shareholder. Otherwise, for each shareholder $U_{k_i}^i$ (superscript i denotes that the shareholder is in group $\mathcal{U}_i$ and subscript k_i is an integer from the interval $[1, n_i]$ and denotes that the shareholder is the k_i-th shareholder in $\mathcal{U}_i$), the dealer D computes a subshare $s_{k_i}^i$ as

$$s_{k_i}^i = \left(s_i \prod_{j=1, j\neq i}^{m} \frac{x_j}{x_j - x_i} + r_{k_i}^i g \right) \bmod p, \tag{8}$$

where $r_{k_i}^i$ is a random integer picked by D and $r_{k_i}^i \in \mathbb{Z}_g$. D sends $s_{k_i}^i$ as private share to the corresponding shareholder $U_{k_i}^i$ securely.

3.1.3. Secret Reconstruction. Note that any SS scheme has the monotone property. In other words, if a shareholder set $\mathscr{M}$ can recover the secret, any superset of $\mathscr{M}$ can also realize secret reconstruction.

If a shareholder set $\mathscr{P}$ is allowed to recover the secret s, at least one shareholder $U_{k_i}^i$ is included in $\mathscr{P}$ for $i = 1, 2, \ldots, m$, where k_i is a random integer in $\mathbb{Z}_{m_i+1}$. Then, without loss of generality, $\mathscr{P}$ can be divided into two subsets $\mathscr{M}_1$ and $\mathscr{M}_2$, where $\mathscr{M}_1 = \{U_{k_1}^1, U_{k_2}^2, \ldots, U_{k_m}^m\}$ and $\mathscr{M}_2 = \mathscr{P} - \mathscr{M}_1$. In terms of definition of GSS, the subset $\mathscr{M}_1$ is able to recover the secret.

Each shareholder in $\mathscr{M}_1$ releases its subshare to the other shareholders in $\mathscr{P}$. After that, all the shareholders in $\mathscr{P}$ get m shares $s_{k_1}^1, s_{k_2}^2, \ldots, s_{k_m}^m$. Then, the secret can be obtained by computing

$$s = \left(\sum_{i=1}^{m} s_{k_i}^i \bmod p \right) \bmod g. \tag{9}$$

3.2. Correctness Analysis. In order to demonstrate that the proposed GSS scheme based on LIP can work correctly, we give two steps to prove the equation $s = (\sum_{i=1}^{m} s_{k_i}^i \bmod p) \bmod g$ (Table 1).

Table 1: Proof of step 1 in GSS based on LIP.

Formulas	Remarks
$s + \sum_{i=1}^{m} r_{k_i}^i g$	
$\leq s + mg(g-1)$	*Remark* 3.1: $r_{k_i}^i \in \mathbb{Z}_g$
$< g + mg^2 - mg$	*Remark* 3.2: $s \in \mathbb{Z}_g$
$< p$.	*Remark* 3.3: $p > mg^2$

Step 1: $\mathbf{s} + \sum_{\mathbf{i}=1}^{\mathbf{m}} \mathbf{r}_{\mathbf{k_i}}^{\mathbf{i}} \mathbf{g} < \mathbf{p}$.

Step 2: $\mathbf{s} = (\sum_{\mathbf{i}=1}^{\mathbf{m}} \mathbf{s}_{\mathbf{k_i}}^{\mathbf{i}} \bmod \mathbf{p}) \bmod \mathbf{g}$.

$$\begin{aligned}
&\cdot \left(\sum_{i=1}^{m} s_{k_i}^i \bmod p \right) \bmod g \\
&= \left\{ \sum_{i=1}^{m} \left(s_i \prod_{j=1, j\neq i}^{m} \frac{x_j}{x_j - x_i} + r_{k_i}^i g \right) \bmod p \right\} \bmod g \\
&= \left\{ \sum_{i=1}^{m} \left(s_i \prod_{j=1, j\neq i}^{m} \frac{x_j}{x_j - x_i} \right) \bmod p + \sum_{i=1}^{m} r_{k_i}^i g \right\} \bmod p \bmod g \\
&= \left(s + \sum_{i=1}^{m} r_{k_i}^i g \right) \bmod p \bmod g \\
&= \left(s + \sum_{i=1}^{m} r_{k_i}^i g \right) \bmod g \\
&= s.
\end{aligned} \tag{10}$$

On account of Step 1 and Step 2, it is proven that the secret s can be obtained by computing $(\sum_{i=1}^{m} s_{k_i}^i \bmod p) \bmod g$.

3.3. Security Analysis. In this section, we give two theorems to prove the security of the proposed GSS based on LIP. Because secret s is uniformly distributed in $\mathbb{Z}_g$, the probability of obtaining s from no share is $1/q$. In general, g is a large number such that it is impossible to guess the secret s directly without any information. Therefore, if the probability of an event occurring is equal to or less than $1/q$, the event can be considered as impossibility.

Theorem 1. *In the GSS based on LIP, a subshare $s_{k_i}^i$ is valid just in the group $\mathcal{U}_i$, while it is invalid in any other group $\mathcal{U}_j$ where $i \neq j$.*

Table 2: Proof of step 1 in GSS based on CRT.

Formulas	Remarks
$y + \sum_{i=1}^{m} r_{k_i}^i (P/p_i) p_0$	
$< s + \alpha p_0 + m p_0^2 P/p_i$	
$< s + \alpha p_0 + m p_0^2 P/p_1$	
$< s + \alpha p_0 + (p_0 - 1) P/p_0$	
$< P/p_0 + (p_0 - 1) P/p_0$	
$= P.$	
	Remark 4.1: $r_{k_i}^i \in \mathbb{Z}_{p_0}$
	Remark 4.2: $p_1 < p_2 < \cdots < p_m$
	Remark 4.3: $m(p_0)^3 < p_1(p_0 - 1)$
	Remark 4.4: $s + \alpha p_0 < (p_1, p_2, \ldots, p_m)/p_0$

Proof. In the light of correctness analysis, we have proved that the subshare $s_{k_i}^i$ is valid in $\mathcal{U}_i$. Hence, we give the proof of the latter part of Theorem 1 in the following. In more detail, we should prove that the probability of $s_{k_i}^i$ being valid in $\mathcal{U}_j$ is no more than $1/q$.

If $s_{k_i}^i$ is valid in $\mathcal{U}_j$, it means $s_{k_i}^i = (s_j \prod_{h=1, h\neq j}^{m} (x_h/x_h - x_j) + r_{k_i}^j g) \bmod p$. In the equation, s_j has a uniform distribution in $\mathbb{Z}_p$ since it is computed as $s_j = f(x_j)$ and all the coefficients of $f(x)$ are unknown. And $r_{k_j}^j$ can be any integer over $\mathbb{Z}_g$. From the equation, we get

$$s_j = \left(\prod_{h=1, h\neq j}^{m} \frac{x_h}{x_h - x_j} \right)^{-1} \left(s_{k_i}^i - r_{k_j}^j g \right) \bmod p. \tag{11}$$

Define $q(r_{k_j}^j) = (s_{k_i}^i - r_{k_j}^j g) \bmod p$ with only one independent variable $r_{k_j}^j$ ($r_{k_j}^j \in \mathbb{Z}_g$). Then, given a fixed value of $s_{k_i}^i$, $q(r_x^j) = q(r_y^j)$ holds if and only if $r_x^j = r_y^j$ for $r_x^j, r_y^j \in \mathbb{Z}_g$. Because if there exist r_x^j and r_y^j such that $q(r_x^j) = q(r_y^j)$, there must be an integer v such that $(r_x^j - r_y^j)g = vp$, i.e., $p | (r_x^j - r_y^j)$. In this way, we can get $r_x^j = r_y^j$, i.e., $v = 0$ because $r_x^j, r_y^j \in \mathbb{Z}_g$ and $p > mg^2$. Furthermore, $q(r_{k_j}^j)$ is a 1-to-1 function, i.e., $q(r_{k_j}^j)$ also has g values corresponding to g values of $r_{k_j}^j$. In the same way, s_j has g values corresponding to g values of $q(r_{k_j}^j)$ because $(\prod_{h=1, h\neq j}^{m} (x_h/x_h - x_j))^{-1}$ is a fixed value. However, s_j has a uniform distribution in $\mathbb{Z}_p$. Hence, the probability that *equation* (11) holds is g/p, which is less than $1/g$ due to $p > mg^2$. This means it is impossible to make *equation* (11) true. In other words, the subshare $s_{k_i}^i$ is invalid in any other group $\mathcal{U}_j$, where $i \neq j$. □

Theorem 2. *In the GSS based on LIP, for a shareholder set $\mathcal{P}$, if there exists a group $\mathcal{U}_i$ such that $\mathcal{P} \cap \mathcal{U}_i = \varnothing$, the set $\mathcal{P}$ cannot recover the secret s.*

Proof. Without loss generality, suppose that $\mathcal{P} = \mathcal{U} - \mathcal{U}_1$. In other words, $\mathcal{P}$ includes shareholders in all the groups except $\mathcal{U}_1$. Besides, suppose that at least two shareholders exist in a group.

Now, let us prove that if $\mathcal{P}$ can recover the secret s, it must know the exact value of the main share s_1.

In group $\mathcal{U}_i$ ($i \neq 1$), we have

$$\begin{cases} s_1^i = \left(s_i c + r_1^i g \right) \bmod p, \\ s_2^i = \left(s_i c + r_2^i g \right) \bmod p, \end{cases} \tag{12}$$

where $c = \prod_{j=1, j\neq i}^{m} (x_j/x_j - x_i)$. In the equation, s_1^i, s_2^i, c, and g are known, while s_i, r_1^i, and r_2^i are unknown. Although there exist three unknown numbers in two equations, they can still be recovered because $s_1^i, s_2^i \in \mathbb{Z}_p, s_i, r_1^i, r_2^i \in \mathbb{Z}_g$ while $p > mg^2$. In other words, s_i, r_1^i, r_2^i are very short compared with s_1^i, s_2^i. Therefore, we can construct a lattice and utilize lattice basis reduction algorithm to recover s_i, r_1^i, r_2^i. From equation (12), we can deduce

$$\left(s_1^i c - s_2^i c \right) \left(g^{-1} \bmod p \right) \equiv r_1^i c - r_2^i c \bmod p. \tag{13}$$

Then, construct a lattice L as follows:

$$L = \begin{pmatrix} Mp & 0 & 0 \\ Mc & 1 & 0 \\ Mc & 0 & 1 \end{pmatrix}. \tag{14}$$

Define a target vector $\mathbf{t} = (M(s_1^i c - s_2^i c)(g^{-1} \bmod p), 0, 0)$, where M is a prime greater than g to guarantee the distance of any lattice vector far away from $\mathbf{t}$ is greater than M.

Note that $r_1^i, r_2^i \in \mathbb{Z}_g$; then, the distance of $\mathbf{t}$ far away from lattice L is

$$\left\| \left(0, r_1^i, r_2^i \right) \right\| = \sqrt{\left(r_1^i \right)^2 + \left(r_2^i \right)^2} \leq \sqrt{2} g. \tag{15}$$

We claim that vector $\mathbf{v} = (M(s_1^i c - s_2^i c)(g^{-1} \bmod p), r_1^i, r_2^i))$ is the closest lattice point far away from $\mathbf{t}$. Finally, we invoke LLL algorithm to recover r_1^i and r_2^i by solving the closest vector problem (CVP$(\mathbf{v}, L)$). Once we recover r_1^i and r_2^i, s_i can be obtained easily. After that, we get $m - 1$ main shares s_i for $i = 2, 3, \ldots, m$ to construct $m - 1$ coordinates on the original polynomial $f(x)$ such as (x_i, s_i).

Now, suppose that we can recover the secret s; it means that we obtain another coordinate $(0, s)$. Then, $f(x)$ can be reconstructed from $(0, s)$ and $m - 1$ coordinates (x_i, s_i) because the degree of $f(x)$ is $m - 1$.

If we obtain $f(x)$, the main share s_1 can be computed as $f(x_1)$ easily. However, we do not have any information about s_1, since no shareholder in group $\mathcal{U}_1$ is concluded in $\mathcal{P}$. As a deduction, it means that $\mathcal{P}$ cannot recover the secret s if there exists a group $\mathcal{U}_i$ such that $\mathcal{P} \cap \mathcal{U}_i = \varnothing$. □

4. GSS Based on CRT

In this section, we first implement a GSS based on Asmuth–Bloom (t, n) threshold SS scheme. The related correctness and security analyses are given in the following.

4.1. Implementing a GSS Based on Asmuth–Bloom SS Scheme. Our GSS scheme based on CRT consists of three algorithms: (1) main share generation for a group; (2) subshare generation for a shareholder; and (3) secret reconstruction.

4.1.1. Main Share Generation for a Group. Let $\mathcal{U}$ be a set of n shareholders. All the shareholders are divided into m disjoint groups $\mathcal{U} = \{\mathcal{U}_1 \cup \mathcal{U}_2 \cup \cdots \cup \mathcal{U}_m\}$. A shareholder only belongs to one group, i.e., $\mathcal{U}_i \cap \mathcal{U}_j = \varnothing$ if $i \neq j$. Each group has n_i shareholders such that $n = n_1 + n_2 + \cdots + n_m$.

At first, the dealer D picks a prime number p_0 and a sequence of pairwise co-prime positive integers $p_1, p_2, \ldots, p_m$ with $p_1 < p_2 < \cdots < p_m$, $(m(p_0)^3/p_0 - 1) < p_1$ and $\gcd(p_0, p_i) = 1$ for $i = 1, 2, \ldots, m$, where p_i is a public modulus associated with the group $\mathcal{U}_i$. Then, D picks a random integer α and secret s in $\mathbb{Z}_{p_0}$, such that $y = s + \alpha p_0 < p_1, p_2, \ldots, p_m/p_0$. Finally, for each group $\mathcal{U}_i$, the dealer D computes a main share s_i as $s_i = (s + \alpha p_0) \bmod p_i$.

4.1.2. Subshare Generation for a Shareholder. After *Algorithm* 1, each group $\mathcal{U}_i$ is allocated a main share. Let $P = p_1, p_2, \ldots, p_m$ and $a_i = (P/p_i)^{-1} \bmod p_i$. If there just exists one shareholder in a group, the dealer can use 0 as the random integer to generate a subshare for the only shareholder. Otherwise, for each shareholder $U^i_{k_i}$ in group $\mathcal{U}_i$, the dealer D computes a subshare $s^i_{k_i}$ as

$$s^i_{k_i} = \left(s_i \frac{P}{p_i} a_i + r^i_{k_i} \frac{P}{p_i} p_0 \right) \bmod P, \tag{16}$$

where $r^i_{k_i}$ is a random integer picked by D and $r^i_{k_i} \in \mathbb{Z}_{p_0}$. D securely sends $s^i_{k_i}$ as a private share to the corresponding shareholder $U^i_{k_i}$.

4.1.3. Secret Reconstruction. If a shareholder $\mathcal{P}$ can recover the secret, without loss of generality, it is divided into two subsets $\mathcal{M}_1$ and $\mathcal{M}_2$, where $\mathcal{M}_1 = \{U^1_{k_1}, U^2_{k_2}, \ldots, U^m_{k_m}\}$ and $\mathcal{M}_2 = \mathcal{P} - \mathcal{M}_1$. $\mathcal{M}_1$ can recover the secret due to the definition of GSS.

Each shareholder in $\mathcal{M}_1$ releases its subshare to the other shareholders in $\mathcal{P}$. After that, all the shareholders in $\mathcal{P}$ get m shares $s^1_{k_1}, s^2_{k_2}, \ldots, s^m_{k_m}$. Then, the secret can be obtained by computing

$$s = \left(\sum_{i=1}^{m} s^i_{k_i} \bmod P \right) \bmod p_0. \tag{17}$$

4.2. Correctness Analysis. In order to demonstrate that the proposed GSS scheme based on CRT can work correctly, we give two steps to prove $s = (\sum_{i=1}^{m} s^i_{k_i} \bmod P) \bmod p_0$ (Table 2).

Step 1: $\mathbf{y} + \sum_{\mathbf{i=1}}^{\mathbf{m}} \mathbf{r^i_{k_i}} (\mathbf{P/p_i}) \mathbf{p_0} < \mathbf{P}$.

Step 2: $\mathbf{s} = (\sum_{\mathbf{i=1}}^{\mathbf{m}} \mathbf{s^i_{k_i}} \bmod \mathbf{P}) \bmod \mathbf{p_0}$.

$$\begin{aligned}
&\cdot \left(\sum_{i=1}^{m} s^i_{k_i} \bmod P \right) \bmod p_0 \\
&= \left\{ \sum_{i=1}^{m} s_i \left(\frac{P}{p_i} \right) a_i + r^i_{k_i} \left(\frac{P}{p_i} \right) p_0 \bmod P \right\} \bmod p_0 \\
&= \left\{ \left(\sum_{i=1}^{m} s_i \left(\frac{P}{p_i} \right) a_i \bmod P + \sum_{i=1}^{m} r^i_{k_i} \left(\frac{P}{p_i} \right) p_0 \right) \bmod P \right\} \bmod p_0 \\
&= \left\{ y + \sum_{i=1}^{m} r^i_{k_i} \left(\frac{P}{p_i} \right) p_0 \right\} \bmod P \bmod p_0 \\
&= \left\{ s + \alpha p_0 + \sum_{i=1}^{m} r^i_{k_i} \left(\frac{P}{p_i} \right) p_0 \right\} \bmod p_0 = s.
\end{aligned} \tag{18}$$

On account of Step 1 and Step 2, it is proven that the secret s can be obtained by computing $(\sum_{i=1}^{m} s^i_{k_i} \bmod P) \bmod p_0$.

4.3. Security Analysis. In this section, we give two theorems to prove the security of the proposed GSS based on CRT. Because the secret s is uniformly distributed in $\mathbb{Z}_{p_0}$, the probability of obtaining s from no information is $1/p_0$. Therefore, if the probability of an event occurring is equal to or less than $1/p_0$, the event can be considered as impossibility.

Theorem 3. *In the GSS based on CRT, a subshare $s^i_{k_i}$ is valid just in the group $\mathcal{U}_i$, while it is invalid in any other group $\mathcal{U}_j$ where $i \neq j$.*

Proof. According to the correctness analysis, we have proved that the subshare $s^i_{k_i}$ is valid in $\mathcal{U}_i$. Hence, we give the proof of the latter part of Theorem 3 in the following. In more detail, we should prove that the probability of $s^i_{k_i}$ being valid in $\mathcal{U}_j$ is no more than $1/p_0$.

If $s^i_{k_i}$ is valid in $\mathcal{U}_j$, it means

$$s^i_{k_i} = \left(s_j \frac{P}{p_j} a_j + r^j_{k_j} \frac{P}{p_j} p_0 \right) \bmod, \tag{19}$$

where $s^i_{k_i}$, P/p_j, and a_j are fixed values; s_j has uniform distributions in $\mathbb{Z}_{p_i}$; and $r^j_{k_j}$ can be any integer over $\mathbb{Z}_{p_0}$. Because $r^j_{k_j}$ has p_0 alternative values, there are at most p_0 alternative values of s_j to make *equation* (19) true. And due to $s_j \in \mathbb{Z}_{p_j}$, the probability that *equation* (19) holds is less

than p_0/p_j. Then, on account of $p_1 < p_2 < \cdots < p_m$ and $(m(p_0)^3/p_0 - 1) < p_1$, we can get $p_0/p_j < p_0/p_1 < 1/p_0$. Therefore, *equation* (19) cannot be satisfied. In other words, the subshare $s_{k_i}^i$ is invalid in any other group $\mathcal{U}_j$, where $i \neq j$. □

Theorem 4. *In the GSS based on CRT, for a shareholder set $\mathcal{P}$, if there exists a group $\mathcal{U}_i$ such that $\mathcal{P} \cap \mathcal{U}_i = \varnothing$, the set $\mathcal{P}$ cannot recover the secret s.*

Proof. Without loss generality, suppose that $\mathcal{P} = \mathcal{U} - \mathcal{U}_1$. In other words, $\mathcal{P}$ includes shareholders in all the groups except $\mathcal{U}_1$. Besides, suppose that at least two shareholders exist in a group.

Let us prove the following proposition firstly. □

Proposition 1. *If $\mathcal{P}$ can recover the secret s, s can also be recovered from $m-1$ main shares s_i for $i = 2, 3, \ldots, m$.*

From the correctness analysis, we know all subshares in the same group $\mathcal{U}_i$ are totally equivalent, i.e., s_k^i is equivalent to s_h^i when they are used to participate in secret reconstruction, where $s_k^i = (s_i(P/p_i)a_i + r_k^i(P/p_i)p_0) \bmod P$, $s_h^i = (s_i(P/p_i)a_i + r_h^i(P/p_i)p_0) \bmod P$ and $r_k^i \neq r_h^i$. Therefore, we can use $m-1$ main shares and random integers which are all selected from $\mathbb{Z}_{p_0}$ to generate another set $\mathcal{P}'$ such that $|\mathcal{P} \cap \mathcal{U}_i| = |\mathcal{P}' \cap \mathcal{U}_i|$ for $i = 2, 3, \ldots, m$. Then, if $\mathcal{P}$ can recover the secret, it means s can also be reconstructed from $\mathcal{P}'$.

Now, the correctness of Proposition 1 has been proved. Its converse-negative proposition can be stated as follows. If s cannot be recovered from $m-1$ main shares s_i for $i = 2, 3, \ldots, m$, $\mathcal{P}$ also cannot recover the secret s. Obviously, the converse-negative proposition is also true.

Then, we will use the deduction to prove Theorem 4. In group $\mathcal{U}_i (i \neq 1)$, we have

$$s_1^i = \left(s_i \frac{P}{p_i} a_i + r_1^i \frac{P}{p_i} p_0\right) \bmod P \Longrightarrow s_1^i \frac{p_i}{P} = s_i a_i + r_1^i p_0 \bmod p_i, \tag{20}$$

because $(P/p_i) | s_1^i$, $\gcd((P/p_i), p_i) = 1$, and $\gcd(a_i, p_i) = 1$. Then, we get

$$\begin{cases} e_1^i = \left(s_i a_i + r_1^i p_0\right) \bmod p_i, \\ e_2^i = \left(s_i a_i + r_2^i p_0\right) \bmod p_i, \end{cases} \tag{21}$$

where $e_1^i = s_1^i (p_j/P) \bmod p_i$ and $e_2^i = s_2^i (p_j/P) \bmod p_i$. In the equation, e_1^i, e_2^i, c, and p_0 are known, while s_i, r_1^i, and r_2^i are unknown. In a similar way, although there exist three unknown numbers in two equations, they can still be recovered because $e_1^i, e_2^i \in \mathbb{Z}_{p_i}$, $s_i, r_1, r_2 \in \mathbb{Z}_{p_0}$ while $p_i > p_1 > (m(p_0)^3/p_0 - 1)$. In other words, s_i, r_1^i, r_2^i are very short compared with e_1^i, e_2^i, Therefore, we still construct a lattice and utilize lattice basis reduction algorithm to recover s_i, r_1^i, r_2^i. From *equation* (21), we can deduce

$$\left(e_1^i a_i - e_2^i a_i\right)\left(p_0^{-1} \bmod p_i\right) \equiv r_1^i a_i - r_2^i a_i \bmod p_i. \tag{22}$$

Then, we still construct the lattice L as follows:

$$L = \begin{pmatrix} Mp & 0 & 0 \\ Ma_i & 1 & 0 \\ Ma_i & 0 & 1 \end{pmatrix}. \tag{23}$$

Define a target vector $\mathbf{t} = (M(e_1^i a_i - e_2^i a_i)(p_0^{-1} \bmod p_i), 0, 0)$, where M is a prime greater than p_0 to guarantee the distance of any lattice vector far away from $\mathbf{t}$ is greater than M.

Note that $r_1^i, r_2^i \in \mathbb{Z}_{p_0}$; then, the distance of $\mathbf{t}$ far away from lattice L is

$$\left\|\left(0, r_1^i, r_2^i\right)\right\| = \sqrt{\left(r_1^i\right)^2 + \left(r_2^i\right)^2} \leq \sqrt{2} p_0. \tag{24}$$

We claim that vector $\mathbf{v} = (M(e_1^i a_i - e_2^i a_i)(p_0^{-1} \bmod p_i), r_1^i, r_2^i))$ is the closest lattice point far away from $\mathbf{t}$. Finally, we still invoke LLL algorithm to recover r_1^i and r_2^i by solving the closest vector problem ($\mathrm{CVP}(\mathbf{v}, L)$). Once we recover r_1^i and r_2^i, s_i can be obtained easily.

After that, we get $m-1$ main shares s_i for $i = 2, 3, \ldots, m$. Given the $m-1$ main shares, we can just obtain y' by CRT from the following system of equations:

$$\begin{cases} y' \equiv s_2 \bmod p_2, \\ y' \equiv s_3 \bmod p_3, \\ \cdots \\ y' \equiv s_m \bmod p_m, \end{cases} \tag{25}$$

where $y' \in \mathbb{Z}_{P'}$, $P' = p_2, p_3, \ldots, p_m$, and $y' = y \bmod P'$, i.e., $y = y' + \beta P'$ from some integer β. However, from Figure 2, there are at least $(P/p_0)/P' = p_1/p_0 > m(p_0)^2/(p_0 - 1) > p_0$ possible values of β such that $y = y' + \beta P'$. In other words, the probability of recovering s from $m-1$ main shares is less than $1/p_0$. According to the converse-negative nature of Proposition 1, the set $\mathcal{P}$ cannot recover the secret s if there exists a group $\mathcal{U}_i$ such that $\mathcal{P} \cap \mathcal{U}_i = \varnothing$.

5. Numerical Examples

In this section, we give two numerical examples to illustrate the two GSS schemes, respectively. In both examples, suppose that there are $m = 3$ groups $\mathcal{U}_1$ with 2 shareholders, $\mathcal{U}_2$ with 3 shareholders, and $\mathcal{U}_3$ with 2 shareholders.

Example 1. Firstly, the dealer D selects two prime numbers $g = 7$ and $p = 157$ such that $p > mg^2$. Then, D generates a degree-2 polynomial $f(x) = 5 + 128x + 73x^2 \bmod p$, where secret s is equal to 5.

For group $\mathcal{U}_1$, D selects an integer $x_1 = 35$ as the public information associated with $\mathcal{U}_1$ and computes $f(x_1) = 24$ as its main share. For group $\mathcal{U}_2$, D selects an integer $x_2 = 92$ as the public information associated with $\mathcal{U}_2$ and computes $f(x_2) = 83$ as its main share. For group $\mathcal{U}_3$, D selects an integer $x_3 = 136$ as the public information associated with $\mathcal{U}_3$ and computes $f(x_3) = 151$ as its main share.

For the two shareholders U_1^1 and U_2^1 in $\mathcal{U}_1$, the dealer D uses two random integers $r_1^1 = 2$ and $r_2^1 = 5$ to compute subshares $s_1^1 = s_1(x_2/x_2 - x_1)(x_3/x_3 - x_1) + r_1^1 g \bmod p = 12$

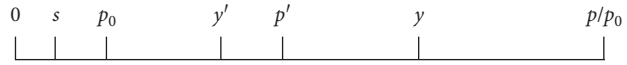

Figure 2: Relationship among parameters.

and $s_2^1 = s_1 (x_2/x_2 - x_1)(x_3/x_3 - x_1) + r_2^1 g \bmod p = 33$. For three shareholders U_1^2, U_2^2, and U_3^2 in $\mathcal{U}_2$, D selects three random integers $r_1^2 = 1$, $r_2^2 = 3$, and $r_3^2 = 6$ to compute subshares $s_1^2 = 24$, $s_2^2 = 38$, and $s_3^2 = 59$. For the two shareholders U_1^3 and U_2^3 in $\mathcal{U}_3$, D picks two random integers $r_1^3 = 2$ and $r_2^3 = 6$ to compute subshares $s_1^3 = 4$ and $s_2^3 = 32$. After that, the dealer D sends each subshare to the corresponding shareholder securely.

Secret reconstruction 1: If U_1^1, U_3^2 and U_2^3 work together to recover the secret, each of them releases its share to the others. Then, the secret is evaluated as $s = (s_1^1 + s_3^2 + s_2^3) \bmod p \bmod g = (12 + 59 + 32) \bmod 157 \bmod 7 = 5$.

Secret reconstruction 2: If U_2^1, U_1^2 and U_1^3 collaborate to recover the secret, each of them releases its share to the others. Then, the secret is evaluated as $s = (s_2^1 + s_1^2 + s_1^3) \bmod p \bmod g = (33 + 24 + 4) \bmod 157 \bmod 7 = 5$.

Example 2. Firstly, the dealer D picks a prime number $p_0 = 7$ and 3 pairwise co-prime moduli $p_1 = 173$, $p_2 = 179$ and $p_3 = 181$ such that $(mp_0^3/p_0 - 1) < p_1$ and $\gcd(p_0, p_1) = \gcd(p_0, p_2) = \gcd(p_0, p_3) = 1$, where p_i is public modulus associated with group $\mathcal{U}_i$ for $i = 1, 2, 3$. Then, D selects a secret $s = 5$ and a random integer $\alpha = 7569$ such that $s + \alpha p_0 < p_1 p_2 p_3 / p_0$.

For group $\mathcal{U}_1$, the dealer D computes a main share $s_1 = s + \alpha p_0 \bmod p_1 = 50$. Main shares for $\mathcal{U}_2$ and $\mathcal{U}_3$ are computed as $s_2 = s + \alpha p_0 \bmod p_2 = 4$ and $s_3 = s + \alpha p_0 \bmod p_3 = 136$.

For the two shareholders U_1^1 and U_2^1 in $\mathcal{U}_1$, the dealer D uses two random integers $r_1^1 = 2$ and $r_2^1 = 4$ to compute subshares $s_1^1 = s_1 (P/p_1) a_1 + r_1^1 (P/p_1) p_0 \bmod P = 4924648$ and $s_2^1 = s_1 (P/p_1) a_1 + r_2^1 (P/p_1) p_0 \bmod P = 5378234$, where $P = p_1 * p_2 * p_3$ and $a_1 (P/p_1) = 1 \bmod p_1$. For the three shareholders U_1^2, U_2^2 and U_3^2 in $\mathcal{U}_2$, D selects three random integers $r_1^2 = 1$, $r_2^2 = 3$ and $r_3^2 = 5$ to compute subshares $s_1^2 = 3945438$, $s_2^2 = 4383820$ and $s_3^2 = 4822202$. For the two shareholders U_1^3 and U_2^3 in $\mathcal{U}_3$, D picks two random integers $r_1^3 = 3$ and $r_2^3 = 6$ to compute subshares $s_1^3 = 3716040$ and $s_2^3 = 4366347$. After that, the dealer D sends each subshare to the corresponding shareholder securely.

Secret reconstruction 1: if U_2^1, U_1^2, and U_2^3 work together to recover the secret, each of them releases its share to the others. Then, the secret is evaluated as $s = (s_2^1 + s_1^2 + s_2^3) \bmod P \bmod p_0 = (5378234 + 3945438 + 4366347) \bmod 5605027 \bmod 7 = 2479965 \bmod 7 = 5$.

Secret reconstruction 2: if U_1^1, U_2^2, and U_1^3 collaborate to recover the secret, each of them releases its share to the others. Then, the secret is evaluated as $s = (s_1^1 + s_2^2 + s_1^3) \bmod P \bmod p_0 = (4924648 + 4383820 + 3716040) \bmod 5605027 \bmod 7 = 1814454 \bmod 7 = 5$.

6. Discussion

In this section, we show both the two GSS schemes are perfect SS schemes. Then, we give some discussions about the information rate for the two GSS schemes.

6.1. Perfect SS

Definition 4. Perfect SS: in an SS scheme, let s, $\mathcal{S}$, $\mathcal{P}$, and λ be the secret, secret space, a shareholder set, and share set of $\mathcal{P}$. The SS is perfect with respect to probability distribution of s on the secret space $\mathcal{S}$ if

(1) $H(s) \geq 0$.

(2) $H(s|\lambda) = 0$ if $\mathcal{P}$ is an authorized set.

(3) $I(s;\lambda) = H(s) - H(s|\lambda) = 0$ if $\mathcal{P}$ is not an authorized set.

In the GSS scheme based on LIP, secret space is $\mathcal{S} = \mathbb{Z}_g$, and hence $H(s) = \log_2 g > 0$. From the correctness analysis in Section 3, any authorized set can recover the secret by executing the algorithm in secret reconstruction, so the second condition holds. From Theorem 2, the probability of obtaining the secret from the shares kept by an unauthorized shareholder set $\mathcal{P}$ is also $1/g$, i.e., $H(s|\lambda) = H(s) = \log_2 g$. Therefore, the GSS scheme based on LIP is a perfect SS scheme.

In the same way, we can get that the GSS scheme based on CRT is also a perfect SS scheme.

6.2. Information Rate

Definition 5. Information rate: in an SS scheme, let s be the secret and $\mathbb{S} = \{s_1, s_2, \ldots, s_n\}$ be the share set. Then, the information rate of the scheme is defined as

$$\rho = \min_{s_i \in \mathbb{S}} \frac{\log_2 |s|}{\log_2 |s_i|}. \tag{26}$$

According to [22], the information rate of a prefect SS scheme is no more than 1. Besides, the higher ρ is, the more effectively the SS scheme works. In the GSS scheme based on LIP, secret s is in $\mathbb{Z}_g$ while every share s_i is in $\mathbb{Z}_p$. Therefore, the information rate ρ is equal to $\log_2 g / \log_2 p$, which is between 1/3 and 1/2 because $p > mg^2$. Although ρ is less than 1, it is still acceptable. However, in the GSS based on CRT, secret s is in $\mathbb{Z}_{p_0}$ while every share s_i is in $\mathbb{Z}_P$, where P is a product of m module and each modulus is greater than p_0. Therefore, the information rate is very low. We just show the scheme to prove that CRT also can be used to design GSS scheme. In practice, the first GSS scheme based on LIP is more advisable.

7. Conclusions

In this paper, we propose a kind of secret sharing which is named group secret sharing (GSS). By modifying Shamir

and Asmuth–Bloom (t, n) threshold SS schemes, we implement two GSS schemes based on LIP and CRT, respectively. The correctness analysis shows that the two GSS schemes can work correctly and the security analysis proves that the two schemes are secure. For a better illustration, two numerical examples are also given. Both the two GSS schemes are perfect, but the GSS based on LIP is more effective because its information rate is higher than the other one.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was supported by the National Key Research and Development Program of China (2018YFB2100300 and 2018YFB0803400) and the National Natural Science Foundation of China (61520106007).

References

[1] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612-613, 1979.

[2] G. R. Blakley, "Safeguarding cryptographic keys," *National Computer Conference*, vol. 48, pp. 313–317, 1979.

[3] J. Kurihara, S. Kiyomoto, K. Fukushima, and T. Tanaka, "Threshold secret sharing scheme and its extension," in *Proceedings of the International Conference on Information Security*, pp. 455–470, Springer, Busan, Korea, 2008.

[4] C. C. Drăgan and F. L. Tiplea, "On the asymptotic idealness of the asmuth-bloom threshold secret sharing scheme," *Information Sciences*, vol. 463, pp. 75–85, 2018.

[5] A. Beimel, T. Tassa, and E. Weinreb, "Characterizing ideal weighted threshold secret sharing," in *Proceedings of the Theory of Cryptography Conference*, pp. 600–619, Springer, Durham, NC, USA, 2005.

[6] P. Morillo, C. Padró, G. Sáez, and J. L. Villar, "Weighted threshold secret sharing schemes," *Information Processing Letters*, vol. 70, no. 5, pp. 211–216, 1999.

[7] O. Farras and C. Padró, "Ideal hierarchical secret sharing schemes," *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 3273–3286, 2012.

[8] P. S. Roy, S. Dutta, K. Morozov et al., "Hierarchical secret sharing schemes secure against rushing adversary: cheater identification and robustness," in *Proceedings of the International Conference on Information Security Practice and Experience*, pp. 578–594, Springer, Melbourne, Australia, 2018.

[9] L. Harn and M. Fuyou, "Multilevel threshold secret sharing based on the Chinese remainder theorem," *Information Processing Letters*, vol. 114, no. 9, pp. 504–509, 2014.

[10] C. Lin, L. Harn, and D. Ye, "Ideal perfect multilevel threshold secret sharing scheme," in *Proceedings of the 2009. IAS'09. Fifth International Conference*, pp. 118–121, IEEE, 2009.

[11] O. Farras, J. Martí-Farré, and C. Padró, "Ideal multipartite secret sharing schemes," in *Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 448–465, Springer, Barcelona, Spain, 2007.

[12] T. Tassa and N. Dyn, "Multipartite secret sharing by bivariate interpolation," *Journal of Cryptology*, vol. 22, no. 2, pp. 227–258, 2009.

[13] C.-F. Hsu and L. Harn, "Multipartite secret sharing based on crt," *Wireless Personal Communications*, vol. 78, no. 1, pp. 271–282, 2014.

[14] C.-P. Lai and C. Ding, "Several generalizations of shamir's secret sharing scheme," *International Journal of Foundations of Computer Science*, vol. 15, no. 02, pp. 445–458, 2004.

[15] Y. Liu and Q. Zhao, "E-voting scheme using secret sharing and k-anonymity," *World Wide Web*, vol. 22, no. 4, pp. 1657–1667, 2019.

[16] M. Mignotte, "How to share a secret," in *Proceedings of the Workshop on Cryptography*, pp. 371–375, Springer, Burg Feuerstein, Germany, 1982.

[17] C. Asmuth and J. Bloom, "A modular approach to key safeguarding," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 208–210, 1983.

[18] A. N. Tentu, V. C. Venkaiah, and V. K. Prasad, "Crt based multi-secret sharing schemes: revisited," *International Journal of Security and Networks*, vol. 13, no. 1, pp. 1–9, 2018.

[19] Y. Ning, F. Miao, W. Huang, K. Meng, Y. Xiong, and X. Wang, "Constructing ideal secret sharing schemes based on Chinese remainder theorem," in *Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security*, pp. 310–331, Springer, Brisbane, Australia, 2018.

[20] H. Cohen, *A Course In Computational Algebraic Number Theory*, Springer Science & Business Media, 2013.

[21] A. K. Lenstra, H. W. Lenstra, and L. Lovász, "Factoring polynomials with rational coefficients," *Mathematische Annalen*, vol. 261, no. 4, pp. 515–534, 1982.

[22] M. Van Dijk, "On the information rate of perfect secret sharing schemes," *Designs, Codes and Cryptography*, vol. 6, no. 2, pp. 143–169, 1995.