EXTENSIONS TO ASMUTH BLOOM SECRET SHARING SCHEME

by

Oğuzhan Ersoy

B.S., Electrical Electronics Engineering, Boğaziçi University, 2012

B.S., Mathematics, Boğaziçi University, 2012

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Electrical Electronics Engineering

Boğaziçi University

2015

ACKNOWLEDGEMENTS

First of all, I would like to thank my advisor, Emin Anarım, for his guidance, caring, patience, and providing me with an excellent atmosphere for doing research.

I am thankful to Thomas B. Pedersen who initiated my research about the secret sharing not only for his continuous advice throughout the thesis but also for his help in improving my writing skills.

I would like to thank Kamer Kaya and Ali Aydın Selçuk for their contributions and joint work.

I am also deeply thankful to all my colleagues at TÜBİTAK BİLGEM for their constant support and for making me want to come to work everyday with great pleasure. I thank our manager, Orhun Kara, for making our unit such an enjoyable environment.

I want to thank the committee members of my thesis, Ali Emre Pusane and Fatih Alagöz, for their help in improving this thesis.

Last but not the least, I am grateful to my family. They were always supporting me and encouraging me with their best wishes.

ABSTRACT

EXTENSIONS TO ASMUTH BLOOM SECRET SHARING SCHEME

The term "Extensions" in the thesis title refers to homomorphism and verifiability additions to the Asmuth-Bloom scheme. Homomorphic abilities enable computations on hidden data without opening it, and verifiability eliminates the necessity of a trusted third party. Both abilities jointly facilitate secure multi-party computation. Multi-party computation has became one of the main research areas of the crypto-community with the goal to create a protocol to jointly compute a function using their inputs without revealing anything but the result. With the technological developments, the demand for personal data storage and computation have increased over the last decades. In order to maintain computational operations over the data, it is usually stored without encryption which brings along some privacy concerns. Rivest *et al.* proposed homomorphic encryption to overcome privacy issues, while keeping the functionality. After more than a quarter century, in 2009, Craig Gentry proposed the first fully homomorphic encryption scheme, and security of his scheme relies on an assumption of the hardness of a mathematical problem, i.e. the approximate GCD problem. Nonetheless, unconditionally or information-theoretically secure computation can be done by secret sharing schemes. In this thesis, we explore homomorphic properties of a well-known secret sharing scheme: Asmuth-Bloom scheme. We propose several modified versions having homomorphic properties with their security analysis. Another important contribution of the thesis is related to Asmuth-Bloom based verifiable secret sharing. First, we analyse the existing schemes and expose their weaknesses. Secondly, we propose the first verifiable secret sharing scheme secure against unbounded adversaries, and we apply this scheme to construct joint random secret sharing scheme.

ÖZET

ASMUTH BLOOM SIR PAYLAŞIM YÖNTEMİNE EKLENTİLER

Tez başlığındaki "Eklentiler" ile Asmuth-Bloom yöntemine eklenen homomorfizm ve doğrulanabilme özellikleri ifade edilmektedir. Homomorfik operasyonlar gizli metinleri açmadan üzerinde işlem yapılabilmesine olanak sağlarken, doğrulanabilme ise güvenilir üçüncü şahıslara olan ihtiyacı ortadan kaldırmaktadır. Bu iki kabiliyet birlikte kullanılarak güvenli çok taraflı hesaplama yapılabilmektedir. Kriptoloji alanındaki ana başlıklardan birisi haline gelen çok taraflı hesaplama, tarafların kendi girdilerini paylaşmadan ortak işlem sonucunu elde etmelerini sağlar. Teknolojik gelişmelerle birlikte, kişisel veri saklama ve bu veri üzerinde işlem yapma talebi son yıllarda daha da artmaktadır. Veri üzerinde işlem yapılabilmesi için genellikle açık bir şekilde saklanması gizlilik konusunda kaygılara yol açmaktadır. Bu konuda ilk çalışma olarak Rivest ve ekibi tarafından homomorfik şifreleme yöntemi ile gizliliği sağlarken işlem yapılabileceği öne sürülmüştür. Yaklaşık çeyrek asır sonra 2009'da, güvenliği matematiksel bir problemin zorluğu kabulüne dayanan, ilk tamamen homomorfik şirfeleme algoritması Craig Gentry tarafından önerilmiştir. Bunun dışında, her koşulda güvenli, enformasyon teorisiyle güvenliği ispatlanabilen sır paylaşımına dayalı yöntemler de bulunmaktadır. Bu tezde, Asmuth-Bloom sır paylaşım yönteminin homomorfik yönü araştırılmıştır. Yöntem üzerinde çeşitli değişiklikler yapılarak bunların homomorfik etkileri ve güvenlik seviyeleri incelenmiştir. Tezin diğer bir önemli katkısı doğrulanabilir sır paylaşımı hakkında olup, Asmuth-Bloom yöntemine dayalı var olan doğrulanabilir yöntemler incelenip, zayıflıkları belirlenmiştir. Ayrıca, sınırsız hesaplama gücüne sahip saldırganlara karşı güvenli Asmuth-Bloom yöntemine dayalı ilk doğrulanabilir sır paylaşım yöntemi önerilmiştir ve bu yapı kullanılarak ortak rasgele sır paylaşım yöntemi inşa edilmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

$\mathcal{A}$	the qualified access structure
$\overline{\mathcal{A}}$	the unqualified access structure
$\mathbf{F}_p$	finite field over p
$gcd(\cdot)$	greatest common divisor
I_i	the share corresponding to the P_i
I_G	$\bigcup_{P_i \in G} I_i$
$lcm[\cdot]$	least common multiple
$M_{(a)}$	$\prod_{i=1}^{a} m_i$
$M^{(b)}$	$\prod_{i=n-b+1}^{n} m_i$
M_G	$\prod_{P_i \in G} m_i$
n	the total number of participants
$\mathcal{P}$	the set of participants
P_i	i^{th} participant
RF	reconstruction function
r	the recovery threshold
S	the secret
$\mathcal{S}$	the set of secret
$\mathcal{S}_{shares}$	the set of shares
SF	sharing function
s	the secrecy threshold
$\mathbb{Z}_p$	the set of all congruence classes modulo p
$\|G\|$	cardinality of set G
$[\cdot]_a$	the arithmetic inside is performed in $\mathbb{Z}_a$
$\{0,1\}^k$	k-bit binary string
κ	security characteristic
ρ	information rate

LIST OF ACRONYMS/ABBREVIATIONS

CRT	Chinese Remainder Theorem
FHE	Fully Homomorphic Encryption
GCD	Greatest Common Divisor
JRSS	Joint Random Secret Sharing
MPC	Multi-Party Computation
PSS	Proactive Secret Sharing
SSS	Secret Sharing Scheme
VSS	Verifiable Secret Sharing

1. INTRODUCTION

Cryptology is the art of hiding information in the presence of third parties, it is composed of two opponent fields: *cryptography* and *cryptanalysis*. Cryptography deals with how to assembling secure systems in the sense of several aspects of information security such as *confidentiality, integrity, authenticity*, whereas cryptanalysis interests breaking, or hacking, them. Some of the well known cryptographic elements are *symmetric* and *public key encryption, secret sharing, signature schemes, (zero knowledge) commitments, oblivious transfer* and so on.

The subject of the thesis is related with secure *multi-party computation* (MPC). In this computer era, MPC has became one of the main research areas of the crypto-community with the goal to create a protocol to jointly compute a function using their inputs without revealing them. Our goal is to explore new solutions to secure MPC using secret sharing schemes based on the Chinese Remainder Theorem (CRT).

The term "Extensions" in the thesis title refers to homomorphism and verifiability additions to the Asmuth-Bloom scheme. Homomorphic abilities enable computations on hidden data without opening it, and verifiability eliminates the necessity of a trusted third party. Both abilities jointly facilitate secure multi-party computation.

1.1. Motivation

With the technological improvements in the last decades, people switched hand-writing documents to electronic copies. Even the electronic storage unit is changed from CDs to memory disk and finally to cloud systems. Cloud computing provides enormous data storage, computational power as well as availability within many devices like smart phones, computers and so on. On the other hand, in addition to these facilities, cloud computing brings along some privacy matters. When it comes to personal data processing, there are lots of issues that needs to be taken into account. Even though encryption ensures privacy concerns, it disables computational abilities. Without any

further details, it can be said that similar problems can be encountered in the data mining concept. With regards to privacy-preserving computation and data mining, the concept of homomorphic encryption is proposed by Rivest *et al.* [1] in 1978.

Homomorphic encryption is the ability to perform computations on the encrypted data without decrypting during the process, and when it is decrypted, the result of operations performed on the data matches. Several partially homomorphic cryptosystems have been around like RSA [2], Paillier [3], and ElGamal [4]. Partially homomorphic systems are not adequate because of the fact that they exhibit either additive or multiplicative homomorphism but not both. The first fully homomorphic encryption scheme is proposed by Craig Gentry in 2009 [5]. Gentry's scheme assures the privacy concerns to some point because the security of his scheme relies on an assumption of the hardness of a mathematical problem, i.e. approximate GCD problem. Nonetheless, unconditionally or information-theoretically secure computation can be done by secret sharing schemes.

Secret sharing refers to methods for distributing a secret amongst a group of participants, each of whom is allocated a share of the secret. The secret can be reconstructed only when a qualified (authorized) group of participants are combined together; individual shares are of no use on their own. Secret sharing schemes (SSS) have information theoretical security, i.e. they cannot be broken even when the adversary has unlimited computing power. On the other hand, most encryption schemes have at best computational security meaning that these systems are secure assuming that an adversary is computationally limited and such an algorithm is vulnerable to future developments in computer power such as quantum computing.

Shamir [6] and Blakley [7] proposed the first secret sharing schemes in 1979. Secret sharing schemes play an important role in cryptosystems, especially for safeguarding keys. Many systems are vulnerable to disclose of the single master key by an accident or an attacker. The result of a disclosure would be catastrophic for crucial cases like launching nuclear missiles. Secret sharing precludes a single point of failure by splitting the master secret into several shares.

Homomorphic secret sharing is a notion introduced by Benaloh [8]. Shamir's SSS is a well-known scheme and it can be used for partially (additive) homomorphic secret sharing. Moreover, there are secret sharing schemes based on the CRT: Asmuth-Bloom [9] and Mignotte [10], GRS [11]. CRT-based homomorphic secret sharing is still an open area for the crypto-community. In this thesis, we are investigating some aspects of CRT-based homomorphic secret sharing as well as CRT-based verifiable secret sharing.

1.2. Contributions of the Thesis

In this thesis, we present three modified versions of the Asmuth-Bloom SSS with homomorphic properties. All three versions have some advantages and also disadvantages with respect to the security level and computational cost. We give detailed explanations of the schemes and their security proofs as well as their properties like information rate.

Secondly, we pointed out certain security concerns for three verifiable secret sharing schemes based on the CRT in the literature. To the best of our knowledge, there exist five such schemes [12–16] where two of them [13, 14] were already proven to be insecure. In this thesis, we first show that two of the remaining schemes [15, 16] are insecure and the remaining one [12] is only secure against a computationally bounded adversary. We propose a modification for this scheme and prove that the modified scheme is a secure verifiable secret sharing scheme against an unbounded adversary. Lastly, as an application, we show how to use the new scheme for joint random secret sharing.

2. PRELIMINARIES

2.1. Mathematical Background

2.1.1. Chinese Remainder Theorem

CRT simply states that the remainders of pairwise relatively prime integers determine a unique solution over the congruences.

Theorem 2.1 (Chinese Remainder Theorem)**.** *Let $m_1, \ldots, m_k$ be pairwise co-primes, and $b_1, \ldots, b_k \in \mathbb{Z}$. The system of equations*

$$\begin{aligned} x &\equiv b_1 \mod m_1 \\ &\vdots \\ x &\equiv b_k \mod m_k \end{aligned}$$

has a unique solution in $\mathbb{Z}_{M_{(k)}}$.

The solution can be found by the following formula:

$$x = \sum_{i=1}^{k} \alpha_i \cdot \beta_i \cdot b_i \mod M_{(k)} \tag{2.1}$$

where $M_{(k)} = \prod_{i=1}^{k} m_i$ and $\alpha_i = \frac{M_{(k)}}{m_i}$, $\beta_i = \frac{m_i}{M_{(k)}} \mod m_i$. Since α_i and β_i can be precomputed, the solution can be found by k weighted additions. Therefore, for the system of k equations, the complexity of calculating the solution is $\mathcal{O}(k)$.

2.1.2. Lagrange's Interpolation

Lagrange's Interpolation formula constructs the polynomial of least degree that coincides with a given of set points. In other words, let $(x_1, y_1), \ldots, (x_k, y_k)$ be given points. The Lagrange formula finds the polynomial, f, of least degree such that $f(x_i) = y_i$ for all i's.

For a given set of points $(x_1, y_1), \ldots, (x_k, y_k)$, the polynomial f can be found by the following formula;

$$f(x) = \sum_{i=1}^{k} y_i \cdot l_i \quad \text{where} \quad l_i = \prod_{\substack{1 \le j \le k \\ i \ne j}} \frac{x - x_j}{x_i - x_j}.$$

Complexity of calculating interpolation is $\mathcal{O}(k \log^2 k)$.

2.2. Homomorphic Encryption

Homomorphic encryption facilitates many cryptographic protocols requiring private and flexible encryption. As an example, we can consider a well-known research problem. In medical research projects, analyzing patients data is crucial to find the recruitment. However, sharing the patient records even with scientific institutes may compromise the privacy of them. This obstacle could prevent many research projects by legal issues. If it would be possible to analyze encrypted data without decrypting it, then the privacy concerns would not be a problem. Here, homomorphic encryption comes into the picture. It is the ability to perform computations on the encrypted data without decrypting during the process, and when it is decrypted, the result of operations performed on the data matches.

Since Rivest *et al.* [1] introduced the notion of homomorphic encryption, it has been investigated [17] and several partially homomorphic systems have been proposed [2–4,18]. These cryptosystems have many application areas like election schemes, watermarking schemes, commitment schemes, lottery protocols etc. On the other hand,

with the demand for more complex structures (e.g. cloud applications), partially homomorphic systems may not be able to satisfy the requirements.

After almost thirty years that the homomorphic encryption idea has been introduced, Gentry proposed the first *fully homomorphic encryption* (FHE) scheme [5]. His novel work is a breakthrough for the crypto-community. Nonetheless, it requires serious improvements to be applicable. Before any detailed explanation of Gentry's scheme, here, we will give the formal definition for homomorphic encryption.

Let $\mathbf{E} : \{0,1\}^n \times \{0,1\}^k \rightarrow \{0,1\}^n$ be an encryption function. $\mathbf{E}$ is said to be homomorphic over $\otimes$ operation if and only if:

$$\exists \odot \text{ such that } \quad \mathbf{E}(pt) \odot \mathbf{E}(pt') = \mathbf{E}(pt \otimes pt') \quad \forall pt, pt' \in \{0,1\}^n.$$

In order to have a fully homomorphic cryptosystem, it should be homomorphic over any operation. For integer domain, it is known that addition and multiplication operations are adequate for any mathematical processing, thus a cryptosystem is fully homomorphic if and only if it is additively and multiplicatively homomorphic. Before the FHE scheme, there has been several proposals for partially homomorphic encryption schemes. *Partially homomorphic* means that it is only capable of doing one operation homomorphically. Here some well known ones;

- *additive*: Paillier [3], Goldwasser-Micali [18](addition modulo 2, $\oplus$).
- *multiplicative*: RSA [2], ElGamal [4].

2.2.1. Gentry's FHE Scheme

Gentry [5] proposed the first fully homomorphic encryption scheme in 2009. The security of the FHE scheme is based on the approximate GCD problem where the best known solutions can be reduced to lattice problems. The scheme includes a noise term in order to blind the plaintext, and the noise grows with each homomorphic addition or multiplication, which causes false decryption after some point. In order to overcome the

growth of the size of the noise term, Gentry came up with a novel idea: bootstrapping.

Bootstrapping. *Bootstrappable encryption* evolves somewhat homomorphic schemes into fully homomorphic schemes. *Somewhat homomorphic* means the ability of doing limited number of homomorphic operations, e.g. unlimited additions but one multiplication [19]. The bootstrapping method is not specific to the FHE scheme, it can be used over any somewhat homomorphic schemes, even we will use it for secret sharing in Section 4.2.

A public key encryption scheme is *bootstrappable* if it can handle f_+ and f_*;

$$f_+(sk, ct, ct') = Dec_{sk}(ct) + Dec_{sk}(ct'), \quad f_*(sk, ct, ct') = Dec_{sk}(ct) \cdot Dec_{sk}(ct').$$

where $Dec_sk(ct)$ corresponds to the decryption of ct using the key sk. An public key encryption scheme can *handle* a set of functions F, if for any $f \in F$, a fresh encryption $E_{pk}(f(x_0, \ldots, x_l))$ can be computed from pk and $E_{pk}(x_0), \ldots, E_{pk}(x_l)$. A *fresh encryption* is a ciphertext which follows the description of the encryption algorithm. From the definitions, Pailler scheme can handle f_+, whereas RSA can handle f_*.

2.3. Secret Sharing

In secret sharing schemes, the secret is chosen by a dealer who split it into shares and distributes over the participants (depending on the case the secret may be chosen randomly, or not).

In *distribution phase*, the dealer splits the secret S into n pieces by using the sharing function **SF** and delivers share I_i to P_i via a secure channel for $i \in \{1, \ldots, n\}$ (discrete channel for each participant).

$$\mathbf{SF}(S) \quad = \quad (I_1, I_2, \cdots, I_n).$$

In *reconstruction phase*, an qualified group A can reconstruct the secret with the help of reconstruction function **RF**:

$$\mathbf{RF}(I_G) \quad = \quad \begin{cases} S & \text{if } G \in \mathcal{A}, \\ \perp & \text{if } G \notin \mathcal{A}. \end{cases}$$

A *perfect secret sharing scheme* should satisfy the following two conditions:

(i) *Correctness*: Any qualified group of participants can reconstruct the secret.
(ii) *Perfect Privacy*: No unqualified group of participants can get any information about the secret.

There are *probabilistic* and *information theoretical (entropy)* approaches for these conditions. In the *probabilistic approach*, the conditions can be seen as:

(i) $\forall G \in \mathcal{A} : Pr\left(\mathbf{RF}(I_G) = S\right) = 1$
(ii) $\forall B \notin \mathcal{A} : Pr\left(\mathbf{RF}(I_B) = a\right) = \frac{1}{|\mathcal{S}|} \quad (\forall a \in \mathcal{S})$,

whereas in the *information theoretical (entropy) approach*:

(i) $\forall G \in \mathcal{A} : H(S|I_G) = 0$
(ii) $\forall B \notin \mathcal{A} : H(S|I_B) = H(S)$

and both definitions are proven to be equivalent [20].

A naive solution to the secret sharing is secret splitting which is based on *one-time pad*. Even though it ensures the absolute security, it is infeasible for many applications since it requires all of the share pieces to recover the secret. This sounds good for the confidentiality but for real life applications, one corrupted share causes to deceive the secret forever. For that reason, there have been several construction methods proposed with different access structures.

2.3.1. Constructions

Access structure is constituted of the predetermined subgroups of the participants qualified to recover the secret. There are several types of SSS realizing different types of access structures. In this section, we briefly examine some of the well known ones; threshold and general access structures. There are several other constructions like Multi-Level, Benaloh and Rudich's Undirected s-t Connectivity etc., further details can be found in [6, 20–23].

2.3.1.1. Threshold Scheme. *Threshold structure* is the most commonly used one, and is also called r out of n access structure. First publications are done by Shamir [6] and Blakley [7], independently. Threshold structure is simply based on the size of the subset; accepts all subsets whose size are above a threshold r:

$$\mathcal{A} = \{G \subseteq \mathcal{P} : |G| \geq r\}.$$

In the case of perfectness, the scheme should not leak any information to a subset of size below the threshold. In [9], the authors relax the definition by adding a new variable s for secrecy threshold. In their definition, a (r, s, n) *ramp secret sharing scheme* should satisfy the following two conditions:

(i) r or any more out of the n participants can reconstruct the secret.
(ii) s or any fewer participants could not get any information about the secret.

Ramp schemes can be seen as generalized versions of threshold schemes. An (r, s, n) ramp secret sharing scheme is equivalent to an (r, n) threshold scheme if $r = s + 1$.

Weighted Threshold. Weighted threshold is introduced by Shamir in [6]. Unlike the threshold scheme where every participant has the same capabilities, weighted threshold enables every participant to have individual weights. This structure is useful in the case of hierarchical systems. Consider the well-known example for this case: In

a company, in order to take an action one of the following is required; the president of the company, a vice-president and an executor or three executors. This problem can be seen as (3,n) threshold scheme where the president has three shares, a vice-president has two shares and an executor has a share. In the same manner, every weighted threshold scheme can be implemented as a threshold scheme accordingly.

Compartmented Structure. Compartmented schemes have an additional requirement from the classical threshold schemes: threshold for each compartment. It can be used for the cases, where the participants consist of several groups and each group has its own threshold. The first compartmented scheme is proposed by Simmons [22].

2.3.1.2. General Access Structure. A *monotone access structure*, $\mathcal{A}$, satisfies the following condition:

$$\forall A, B \subseteq \mathcal{P} : (A \in \mathcal{A})(A \subseteq B) \implies B \in \mathcal{A}.$$

It is reasonable to assume an access structure as a monotone set unless working on negative shares, meaning veto capabilities of the participants [24, 25]. In [26], Ito *et al.* proposed a method realizing any monotone access structure. Benaloh and Leichter [27] pointed out the importance of general access structures by showing that there exist access structures which cannot be realized with any threshold scheme.

Even though both constructions given by Ito *et al.* [26] and Benaloh and Leichter [27] can be used for every monotone structure, they are far from the optimal constructions with respect to their exponential share size.

2.3.2. Properties

In order to compare (or classify) SSSs regarding of security and efficiency aspects, *perfectness* and *idealness* properties can be used.

Perfectness. We already used perfectness as in the definition of a secret sharing scheme regarding with some of the basic studies [6, 27]. However, not all of the secret sharing schemes are perfect, e.g. CRT based schemes. For that reason, we define a criteria in order to compare the *security characteristic* of the schemes:

$$\kappa = \min_{B \in \overline{\mathcal{A}}} \frac{|\{s_B : Pr(S = s_B | I_B) \neq 0\}|}{|\mathcal{S}|}.$$

Security characteristic (κ) of a perfect scheme is 1, but a scheme with security characteristic of 1 may not be perfect such as Asmuth-Bloom scheme because this criteria concerns with the number of possible solutions, not the probability of them. Furthermore, to the best of our knowledge, there is no SSS with a difference greater than one between the number of appearances of any two possible secret value. Therefore, this criteria has very close relation with perfectness. Using this formula, we define the *security level* of a scheme as $\kappa \times |S|$. Meaning of the security level of a scheme is the number of possible solutions for an unqualified group of participants.

For Asmuth-Bloom SSS, *asymptotic perfectness* is defined by Quisquater *et al.* [28] saying that special parameter set makes the probabilities almost the same, and the difference goes to zero as the elements in the set increases. Another work was done by Kaya and Selçuk [29] leading to a modification over the original scheme. Detailed explanations given in Section 3.4.

Idealness. Brickell [30] introduced the notion of *ideal SSS* which is simply the proportion of the secret size over the share size. In order to give a formal definition, *information rate* can be defined as

$$\rho = \frac{\log |\mathcal{S}|}{\log |\mathcal{S}_{shares}|}. \tag{2.2}$$

For some SSSs, share sizes may differ for each participant, in that case $|\mathcal{S}_{shares}|$ is assumed to be the maximum one. Note that, in some papers, the information rate is

also defined as the inverse of (2.2). In order to avoid any confusion we will stick with (2.2) which seems to be more commonly used. For both versions, a SSS is ideal if and only if $\rho = 1$, i.e. has information rate 1. For example, Shamir's SSS is ideal since its secret size is equal to the share sizes.

Karin *et al.* [31] showed that a perfect SSS cannot have information rate greater than 1. Also, it is proven that every access structure cannot be realized with an ideal SSS [32]. Based on the information theoretical approach, several works have been done to find the lower bound of the information rate of generalized access structures [31–34]. Nonetheless, there is still a huge gap between the upper and the lower bounds.

2.3.3. Extensions

Secret sharing schemes can differ regarding to their design criteria. Some of them serve for a specific structure and require additional properties like homomorphism, verifiability or proactivity. There are other proposed SSS types like *reactive* [35], *dynamic* [36] secret sharing etc. Here, we briefly explain some of the well-known ones.

Verifiability. The dealer in a SSS has a crucial impact on the system; in the malicious case, the dealer may forge the shares of the participants and misdirect them. The need of a trusted dealer raises practical privacy and authenticity concerns for the system. In addition to a malicious dealer, the participants can also cheat during the reconstruction phase. In order to overcome a corrupted dealer and participants, the concept of *verifiable secret sharing* (VSS) is introduced by Chor *et al.* based on Shamir's SSS [37]. A VSS scheme enables participants to check the validity of the shares during the distribution and reconstruction phases.

Shamir's SSS has long had verifiable variants [37–39]. Using the preliminary works, several VSSs based on CRT have been proposed [12–16] which will be examined in the following sections. Since a VSS implies robustness against a corrupted dealer, a typical application is joint random secret sharing (JRSS) where playing the role of the

dealer, all users jointly generate and share a random secret, e.g., [40, 41].

Homomorphism. Benaloh [8] came up with the idea of secret sharing homomorphism similar to the encryption homomorphism. As we mentioned earlier, homomorphic encryption systems may not be feasible since their security proofs are based on an assumption, whereas SSSs have information theoretical security proofs. At this point, it is reasonable to use secret sharing systems which require several authorities to carry operations. A secret sharing scheme is called homomorphic over $\otimes$ operation if and only if:

$$\exists \odot \text{ such that } \quad \mathbf{RF}(S) \odot \mathbf{RF}(S') = \mathbf{RF}(S \otimes S'). \tag{2.3}$$

Proactivity. In the classical SSSs, the shares are kept the same as long as the secret is the same. Therefore, available time for an attack against shares is the entire life-time of the secret. This structure is not appropriate for some cases where the secret is not accessible and/or it is supposed to remain the same for a long time. Herzberg *et al.* [42] proposed the first *proactive secret sharing* (PSS) scheme to overcome the life-time attacks. PSS has the ability of *renewing* the shares without changing the secret. An adversary trying to mount the shares is restricted by the life-time of a share which is adjustable with respect to the requirements.

Error Correcting Codes vs. SSS. Error correcting codes are similar to the secret sharing schemes in a way that both of them are aiming to reconstruct a hidden information from pieces of it. Secret sharing schemes, in addition to that, provides a upper bound for non-recovery of the hidden (secret) information. This property is necessary for secret sharing since security of the system relay on that.

The first work about the relationship between SSS and error correcting codes is done by McEliece and Sarwate [43]. They discussed the similarity of the Shamir's

SSS with Reed-Solomon Codes. Another important work done by Goldreich *et al.* [11] where they proposed a CRT-based SSS using Chinese Remainder codes.

3. SOME OF THE WELL-KNOWN SECRET SHARING SCHEMES

3.1. Additive SSS

Additive SSS, also called secret splitting, is the simplest way to share a secret. It is based on addition over $\mathbb{Z}_{2^k}$, which is an extension of the *one-time pad* encryption scheme, and the secret is also chosen in the same modulo congruences. In Figure 3.1, the structure of the scheme can be seen.

Distribution Phase

The dealer choose $S \in \mathbb{Z}_{2^k}$ and random $r_i \in \mathbb{Z}_{2^k}$ values for $i = 1, \ldots, n-1$.

He computes and distributes shares $I_i = r_i$, for $i = 1, \ldots, n-1$ and

$I_n = \left[\sum_{i=1}^{n-1} r_i + S\right]_{2^k}$.

Reconstruction Phase

Let G be a group of participants gathered to reconstruct the secret S;

$$\mathbf{RF}(I_G) : \begin{cases} \text{if } |G| = n \implies S = \left[\sum_{i=1}^{n} I_i\right]_{2^k}. \\ \text{if } |G| < n \implies \perp. \end{cases}$$

Figure 3.1. Additive SSS.

3.1.1. Comments on Additive SSS

Because of its naive and simple structure, additive SSS is used for many applications where the threshold is the same with the number of participants ($r = n$). For any other threshold requirements, however, it is not applicable.

It is a perfect SSS since $n-1$ or less shares do not give any information about

the secret, and it is also ideal because all of the shares and the secret have the same size. The recovery complexity of the scheme is just n addition, $\mathcal{O}(n)$.

Homomorphism. The scheme is additively homomorphic because addition of several secrets can be recovered by adding all of the shares. However, this is not the case for multiplication. Bogdanov et al. [44] proposed a protocol to make the scheme multiplicatively homomorphic by sharing additional information between participants. It is based on the work done by Du and Atallah [45] which can be used to multiply two values given by two parties without anyone else knowing them. Including Du and Atallah's method for 3 participants case, and the complexity of multiplication of two secrets requires 3 rounds and 27 messages. Using this multiplication protocol, they proposed a virtual machine, Sharemind, for privacy preserving data processing [44].

3.2. Shamir's SSS

Shamir's SSS is based on polynomial interpolation over a finite field $\mathbf{F}_p$ where p is a prime number. The algorithm of Shamir's SSS can be seen in Figure 3.2. Here x_i's could be public, and it is not necessary to store them by the shareholders. Any r of the participants can reconstruct the secret via Lagrange Interpolation formula. Because the Lagrange Interpolation theorem states that there exists only one polynomial $f \in \mathbf{F}_p$ so that f with degree of $r-1$ or less, $f(x_i) = y_i$ for $i \in \{1, \ldots, r\}$ and distinct x_i's. Shamir's SSS is not representable over 2-dimensional space, but Figure 3.3 can be used for illustration purposes.

3.2.1. Comments on Shamir's SSS

Shamir's SSS is a perfect secret sharing scheme since any $r-1$ or fewer shares can only construct a polynomial of degree $r-2$ and for this polynomial all values over $\mathbf{F}_p$ are equally probable candidates of the secret. Moreover, since the share sizes are the same with the secret size, it is an ideal SSS.

Distribution Phase

The dealer randomly chooses a polynomial f such that $f \in \mathbf{F}_p$ with degree $r-1$ and $f(0) = S$.

The dealer computes and distributes $I_i = (x_i, y_i)$ pairs where $y_i = f(x_i)$ $(\forall\, i)$.

Reconstruction Phase

Let G be a group of participants gathered to reconstruct the secret,

The participants compute the polynomial f by using the Lagrange formula, then extract secret;

$$\mathbf{RF}(I_G) : \begin{cases} \text{if } |G| \geq r & \Longrightarrow\ f : f(x) = \sum_{i=1}^{k} y_i \cdot \prod_{\substack{1 \leq j \leq k \\ i \neq j}} \frac{x - x_j}{x_i - x_j} \\ & \Longrightarrow\ S = f(0). \\ \text{if } |G| < r & \Longrightarrow\ \perp. \end{cases}$$

Figure 3.2. Shamir's SSS.

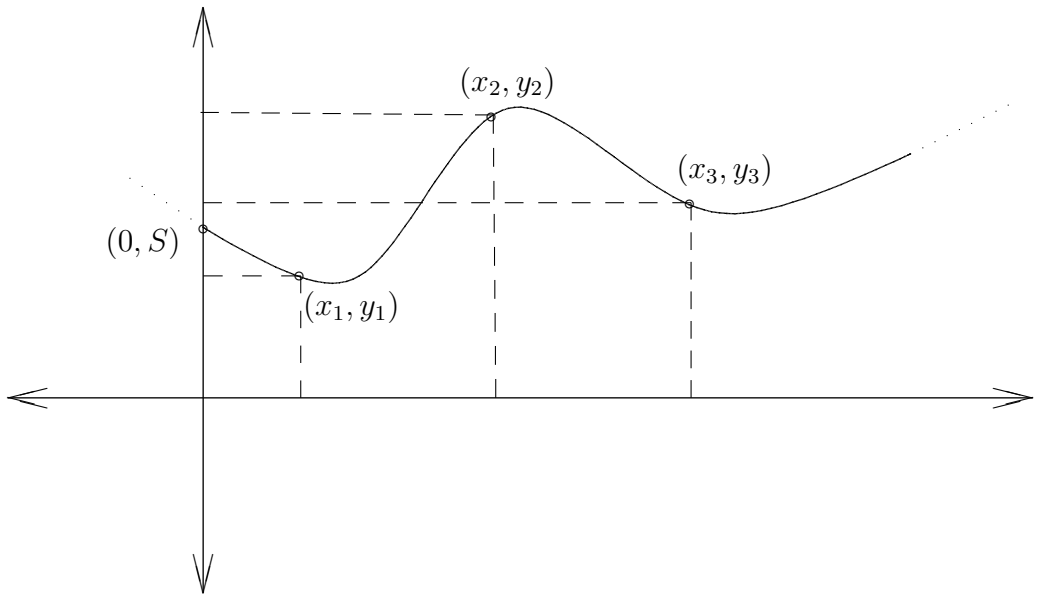

Figure 3.3. Illustration of Shamir's SSS.

Homomorphism. Shamir SSS is additively homomorphic because of the fact that addition of the two functions over the same position gives the result of the summation function on that position (i.e. $\forall f, g \in \mathbf{F}_p \quad f(x) = y,\ g(x) = y' \Rightarrow (f+g)(x) = y + y'$).

On the other hand, it is not multiplicatively homomorphic since multiplication of two polynomials doubles the degree which requires doubling the threshold r. For that reason, a degree reduction is necessary in the Shamir's SSS. Degree reduction can be carried out with secret sharing of the shares with the other participants, which requires $\mathcal{O}(n^2)$ communication over the participants. Note that Gentry's bootstrapping idea can be seen as an extension of re-sharing for encryption schemes. These re-sharing protocols and their security proofs can be found in [46–48]. Homomorphic properties of Shamir's scheme are used in real-life applications, e.g. Danish sugar beet auction [49].

3.3. Mignotte's SSS

Mignotte SSS is based on CRT and the algorithm can be seen in Figure 3.4. By using the CRT formula given in Equation 2.1, any r of the participants can reconstruct the secret S since CRT guarantees unique solution less than $M_{(r)}$.

3.3.1. Comments on Mignotte's SSS

Secret S is chosen in between $M^{(r-1)}$ and $M_{(r)}$ where $M^{(r-1)} = \prod_{i=n-r+1}^{n} m_i$ and $M_{(r)} = \prod_{i=1}^{r} m_i$. This could cause complications for the cases S is not a random integer (e.g. S should be in a specific interval) and the dealer have to choose $(m_1, m_2, \cdots, m_n)$ so that $M^{(r-1)}$ and $M_{(r)}$ cover the range of possible secret values. Furthermore, integrating Mignotte's SSS into a cryptosystem could be trouble because of the infeasible secret space. Even for the security related concerns, secret range comes into the picture.

Theorem 3.1. *The security characteristic of the Mignotte's SSS is equal to* $\frac{1}{M^{(r-1)}}$.

Proof. Let B be an unqualified group of participants ($B \in \overline{\mathcal{A}}$), $M_B = \prod_{j \in B} m_j$. By

Distribution Phase

The dealer chooses a set of integers $(m_1, m_2, \cdots, m_n)$ such that

(i) $m_1 < m_2 < \cdots < m_n$,

(ii) $gcd(m_i, m_j) = 1$ $(\forall\, i \neq j)$,

(iii) $M_{(r)} > S > M^{(r-1)}$.

He computes and distributes $I_i = S \bmod m_i$ $(\forall\, i)$.

Reconstruction Phase

Let G be a group of participants gathered to reconstruct the secret,

The participants compute the secret S by using CRT;

$$\mathbf{RF}(\bigcup_{P_i \in G} I_i) : \begin{cases} \text{if } |G| \geq r \implies S = \left[\sum_{P_i \in G} \frac{M_G}{m_i} \cdot \left[\frac{m_i}{M_G} \right]_{m_i} \cdot I_i \right]_{M_G}. \\ \text{if } |G| < r \implies \perp^{a}. \end{cases}$$

[a]Note that Mignotte' SSS is not perfect, and the security level of the system is explained in Theorem 3.1 and Corollary 3.2

Figure 3.4. Mignotte's SSS.

using CRT, B can recover secret in $(\bmod M_B)$:

$$\mathbf{RF}(I_B): \quad \forall P_i \in B,\ y \bmod m_i \text{ is known} \implies \exists!\, S' \text{ such that } S' \equiv S \bmod M_B.$$

In the case of $B = \{P_{n-r+2}, P_{n-r+3}, \ldots, P_n\}$, M_B will be equal to $M^{(r-1)}$. Therefore, the participants in B can narrow the number of possible secret values from $M_{(r)} - M^{(r-1)}$ to $\frac{M_{(r)} - M^{(r-1)}}{M^{(r-1)}}$ by using S': the possible secret values for the participants in B are $\{S' + M^{(r-1)}, S' + 2 \cdot M^{(r-1)}, \ldots, S' + \frac{M_{(r)} - M^{(r-1)}}{M^{(r-1)}} \cdot M^{(r-1)}\}$. We can use the *security characteristic* formula given in Equation 2.2:

$$\begin{aligned} \kappa_{Mignotte} &= \min_{B \in \overline{\mathcal{A}}} \frac{|\{s_B : Pr(S = s_B | \bigcup_{P_i \in B} I_i) \neq 0\}|}{|\mathcal{S}|} \\ &= \frac{\frac{M_{(r)} - M^{(r-1)}}{M^{(r-1)}}}{M_{(r)} - M^{(r-1)}} = \frac{1}{M^{(r-1)}}. \end{aligned}$$

□

Corollary 3.2. *Mignotte's SSS is not perfect: all possible secret values does not have the same probability for an unqualified group of participants, not even for one participant.*

Proof Sketch. As we mentioned, perfect schemes have security level of 1. Even a single participant P_i can eliminate the possible values of secret into $\frac{M_{(r)}-M^{(r-1)}}{m_i}$ by just using his own share I_i. □

Verifiability. The only VSS based on Mignotte's scheme is proposed by Iftene in [14], which is based on the hardness of the *discrete logarithm problem*. It can detect malicious behavior of any participant, whereas dealer can cheat by choosing the secret out of the its range [12].

Homomorphism. Iftene [50] observed the partially homomorphic property of the Mignotte SSS with a condition: If two secrets S_1, S_2 satisfy $S_1 + S_2 < M_{(r)}$, then it provides homomorphic addition over shares since addition is homomorphic operation in modular arithmetic. However, it should be taken into account that the condition $S_1+S_2 < M_{(r)}$ restricts individual secrets in between $M^{(r-1)}$ and $M_{(r)}/2$. Therefore, the security level decreases to almost half, $\frac{M_{(r)}/2-M^{(r-1)}}{M^{(r-1)}}$ and this is for only one addition. In the case of k homomorphic addition, the condition should be $S_1 + \cdots + S_k < M_{(r)}$ and it will reduce to $\frac{\frac{M_{(r)}}{k}-M^{(r-1)}}{M^{(r-1)}} \approx \frac{M_{(r)}-M^{(r-1)}}{M^{(r-1)}} \cdot \frac{1}{k}$, in other words, $1/k^{th}$ of the original scheme.

Using the additively homomorphic property of his version, Iftene proposed an E-Voting scheme [50]. The security of the scheme is very dependent on $\frac{M_{(r)}-M^{(r-1)}}{M^{(r-1)}}$ factor and also the number of voters.

3.4. Asmuth-Bloom SSS

The Asmuth-Bloom scheme is a CRT-based SSS as shown in Figure 3.5. Since CRT with r moduli guarantees a unique solution for $y < M_{(r)}$, the secret S can be extracted by computing first y and then $y \bmod p$. We will refer A as blinding factor and y as blinded secret.

Distribution Phase

The dealer chooses an *Asmuth-Bloom* sequence $(p, m_1, m_2, \cdots, m_n)$ with $S < p$.

He chooses an arbitrary A such that $y = S + A \cdot p < M_{(r)}$.

He computes and distributes shares $I_i = y \bmod m_i$ $(\forall i)$.

Reconstruction Phase

Let G be a group of participants gathered to reconstruct the secret,

The participants compute the secret S by using CRT;

$$\mathbf{RF}(I_G) : \begin{cases} \text{if } |G| \geq r \quad \Longrightarrow \; y = \left[\sum_{P_i \in G} \frac{M_G}{m_i} \cdot \left[\frac{m_i}{M_G} \right]_{m_i} \cdot I_i \right]_{M_G} \\ \qquad\qquad\quad \Longrightarrow \; S = y \bmod p. \\ \text{if } |G| < r \quad \Longrightarrow \; \perp. \end{cases}$$

Figure 3.5. Asmuth-Bloom SSS.

An integer set $(p, m_1, m_2, \cdots, m_n)$ satisfying all of the following four conditions is called *Asmuth-Bloom* sequence:

$$\begin{aligned} &(1) \quad m_1 < m_2 < \cdots < m_n, \\ &(2) \quad gcd(m_i, m_j) = 1 \quad (\forall\, i \neq j), \\ &(3) \quad gcd(p, m_i) = 1 \quad (\forall\, i), \\ &(4) \quad M_{(r)} > p \cdot M^{(r-1)}. \end{aligned} \tag{3.1}$$

3.4.1. Comments on Asmuth-Bloom SSS

Theorem 3.3 ([9]). *In Asmuth-Bloom SSS, a passive adversary cannot eliminate any candidate from $\mathbb{Z}_p$ for the secret.*

Proof. Let B be an unqualified group of participants ($B \in \overline{\mathcal{A}}$). By using CRT, the participants in the set B can recover solution in $\bmod M_B$:

$$\mathbf{RF}(I_B): \quad \forall P_i \in B,\ y \bmod m_i \text{ is known } \implies \exists!\, y' \text{ such that } y' \equiv y \bmod M_B.$$

Using the fourth condition of the Asmuth-Bloom sequence and $r-1 \geq |B|$, it follows that

$$M_{(r)} > p \cdot \prod_{i=1}^{r-1} m_{n-i+1} \geq p \cdot M_B.$$

Therefore, the set of elements in $\mathcal{S}_B = \{y', y' + M_B, y' + 2 \cdot M_B, \ldots, y' + (p-1) \cdot M_B\}$ are less than $M_{(r)}$, meaning that they are in the set of possible blinded secret y. Since $(M_B, p) = 1$, the elements in the set $\mathcal{S}_B$ cover the all possible solutions in $\bmod p$. In other words, the unqualified group of participants cannot reduce the set of possible solutions for the secret S. □

Theorem 3.4 ([28]). *Asmuth-Bloom SSS is not perfect: Secret candidates do not have the same probability for an unqualified group B having less than t shares; every secret candidate will be obtained either $\left\lfloor \frac{M_{(r)}}{pM_B} \right\rfloor$ or $\left\lfloor \frac{M_{(r)}}{pM_B} \right\rfloor + 1$ times when $y \bmod p$ is computed for each possible y candidate.*

Proof Sketch. Let B be an unqualified group of participants ($B \in \overline{\mathcal{A}}$). In Theorem 3.3, it is shown that the participants in the set B can recover solution in $\bmod M_B$, y'. Since dealer choose a random blinding factor A such that $y \in (0, M_{(r)})$, the participants in the set B will have $\lfloor \frac{M_{(r)} + M_B - y'}{M_B} \rfloor$ possible solutions for blinded secret y. From that, in the possible solution set, every possible secret value will be seen either $\left\lfloor \frac{M_{(r)}}{p \cdot M_B} \right\rfloor$ or

$\left\lfloor \frac{M_{(r)}}{p \cdot M_B} \right\rfloor + 1$ times. Therefore, the possible secret values will not have possibility for an unqualified set of participants. □

Let $\Pr_{(B,S)}(S')$ be the probability of $S' \in \mathbb{Z}_p$ is equal to the shared secret S from an unqualified group B's point of view. For a perfect SSS, $\Pr_{(B,S)}(S') = \Pr_{(B,S)}(S)$ for all possible S, S', and B combination. We should point out that, from Theorem 3.4, the number of appearances of the possible secret values can differ by one and the secret candidates are (negatively or positively) biased to be the secret. Hence, the secret candidates will not be equally likely to be the secret which can be a problem especially when $\frac{M_{(r)}}{pM_B}$ is small and the bias is large. To alleviate this, Quisquater *et al.* proposed that $p, m_1, \ldots, m_n$ should be chosen as consecutive primes to make the scheme *asymptotically perfect* [28]. That is: for every B and positive ϵ value, the dealer can choose a prime p such that $\Pr_{(B,S)}(S') - \Pr_{(B,S)}(S) < \epsilon$. For similar reasons, Kaya and Selçuk [29] proposed to change the fourth condition of the Asmuth-Bloom sequence with

$$M_{(r)} > p^2 \cdot M^{(r-1)}. \tag{3.2}$$

In this case, the scheme becomes *statistical*, i.e., the statistical distance between the distribution $\Pr_{(B,S)}(.)$ and uniform distribution is smaller than a given ϵ with a carefully chosen p.

Theorem 3.5 ([29])**.** *The modified Asmuth-Bloom scheme with* (3.2) *is a statistical secret sharing scheme against a passive adversary.*

Proof. Let B be the set of the users corrupted by the adversary, $|B| < r$. We will prove that from the adversary's point of view, the probability distribution $\Pr_{(S,B)}(.)$ over all $S' \in \mathbb{Z}_p$, which we denote by $\mathcal{X}$, is statistically indistinguishable from a uniform distribution. With the shares of B, the adversary can compute $y' = y \bmod M_B$. Let $\omega = \frac{M_{(r)}}{M^{(r-1)}}$. Due to fourth condition, $M_{(r)}/M_B > \omega > p^2$ and hence, from the adversary's point of view, there are at least ω candidates for y consistent with y'.

Since $\gcd(p, M_B) = 1$, for each secret candidate $S' \in \mathbb{Z}_p$, there are $\left\lfloor \frac{\omega}{p} \right\rfloor$ or $\left\lfloor \frac{\omega}{p} \right\rfloor + 1$ candidate y values consistent with $S' \in \mathbb{Z}_p$. Hence, $\Pr_{(B,S)}(S')$ is $\frac{1}{p} \pm \mathcal{O}(\omega^{-1})$ for each S'. Then the statistical distance between $\mathcal{X}$ and the uniform distribution is bounded by $\mathcal{O}(p \times \omega^{-1})$. Since $\omega > p^2$, we know that $p \times \omega^{-1} < 2^{|p|}$, where $|p|$ is the bit length of p. Hence, $\mathcal{X}$ is asymptotically indistinguishable from the uniform distribution, and the Asmuth-Bloom scheme is a statistical SSS. □

Here, we sightly modified the statement of the theorem, but the meaning and the proof are almost the same.

3.4.1.1. Verifiability. The first work about VSS based on Asmuth-Bloom is done by Qoing *et al.* in [13]. They proposed a method similar to Pedersen's VSS [38] using polynomial evaluation. Kaya and Selçuk [12] proposed another VSS based on Asmuth-Bloom SSS, which is similar to the Iftene's VSS [14] based on Mignotte's SSS using the hardness of the *discrete logarithm problem*. Kaya and Selçuk also proposed JRSS and PSS schemes using their VSS.

Kaya and Selçuk [12] proved that the previous proposals [13, 14] are insecure because the dealer can distribute inconsistent shares that lead to different reconstructed secrets for different qualified subsets. To solve this problem, they used a *range proof* to assure the validity of the range of the secret. Recently, two VSS schemes based on Asmuth-Bloom have been proposed [15, 16], which aim to provide range proof in more efficient way.

In Section 5.1, we show that two of the remaining schemes [15, 16] are insecure and the remaining one [12] is only secure against a computationally bounded adversary. In Section 5.2, we propose a modification for VSS in [12] and prove that the modified scheme is a secure verifiable secret sharing scheme against an unbounded adversary. Lastly, as an application, we show how to use the new scheme for joint random secret sharing.

3.4.1.2. Idealness. Asmuth-Bloom is not *ideal*, but it can be constructed to be almost ideal. Since as the $(p, m_1, m_2, \ldots, m_n)$ values increase, it will be possible to choose them closer to each other and eventually the difference between them would be negligible without violating the fourth condition of the Asmuth-Bloom sequence, i.e. information rate of Asmuth-Bloom SSS goes to 1 as its parameters increases. In [51], the authors introduce the notion of *k-compactness*, and they proved that Asmuth-Bloom SSS is asymptotically ideal if the Asmuth-Bloom sequence is chosen as a *1-compact* sequence. We refer to the paper [51] for further details.

3.4.1.3. Homomorphism. Asmuth-Bloom SSS is neither *additively homomorphic* nor *multiplicatively homomorphic* because the range of the secret S exceeds the limit of CRT for a unique solution.

Iftene [50] showed that Asmuth-Bloom SSS would be *homomorphic* in the case of $p \mid m_i$, $\forall i \in \{1, \ldots, n\}$. However, as the author mentioned, in this case any participant could get the secret by itself and this conflicts with idea of SSS:

$$\begin{aligned} \forall i \in \{1, \ldots, n\} : p \mid m_i,\ I_i \equiv y \bmod m_i \quad &\iff I_i \equiv y \bmod p \\ &\iff I_i \equiv S + A \cdot p \bmod p \\ &\iff S = I_i \bmod p. \end{aligned} \tag{3.3}$$

Equation 3.3 shows that every participant can extract S with his/her own share. The author left an open question for a method of constructing Asmuth-Bloom sequence such that

$$\forall\, G \in \mathcal{A} :\ p \mid M_G \quad \text{and} \quad \forall\, i :\ p \nmid m_i$$

and also keeping the same security margin. Note that this statement does not prevent $p \mid M_B$ where B is an unqualified group of participants. Therefore, similar to Equation 3.3, B can recover the secret which concludes Sorin's question. In order to overcome

this vulnerability, we modified his statement with

$$\forall G \in \mathcal{A},\, p \mid M_G \quad \text{and} \quad \forall B \in \overline{\mathcal{A}},\, p \nmid M_B. \tag{3.4}$$

However, for the statement given in (3.4), we show that the scheme is insecure and also give the upper bound for the security level of the scheme (see Theorem 4.5).

Theorem 3.6. *Asmuth-Bloom SSS has fully homomorphism over $\mathbb{Z}_p$ if and only if $p \mid M_G$ for all $G \in \mathcal{A}$.*

Proof Sketch. Let G be a qualified group of participants ($G \in \mathcal{A}$). By the definition of Benaloh [8], it is sufficient to satisfy the condition in Equation (2.3): Find a transition from $\mathbf{RF}(\bigcup_{P_i \in G} I_i \odot I_i')$ to $y \otimes y'$ for any operation $\otimes$. In the case of $p \mid M_G$, transition can be constructed using the same operations (i.e. $\odot \equiv \otimes$):

$$\forall\, y, y' \in \mathbb{Z},\ (y \otimes y' \bmod M_G) \bmod p \equiv y \otimes y' \bmod p \iff p | M_G. \tag{3.5}$$

Meaning of the statement in (3.5) is that the overflow caused by $(\mathrm{mod} M_G)$ will not change the result for $(\mathrm{mod} p)$ if and only if $p \mid M_G$. Since $\otimes$ can be any operation over $\mathbb{Z}_p$, Asmuth-Bloom SSS is *fully homomorphic* in the case of (3.4). □

Another work on homomorphic SSS based on Asmuth-Bloom is done by Kaya and Selçuk [12], where they used additive homomorphism in order to construct a PSS.

In Section 4.2, we improved Kaya and Selçuk's algorithm with the homomorphic multiplication ability. Morever, we give a theoretical results for the security level of the modified version. In Sections 4.1 and 4.3, we proposed another two modifications for homomorphism, and their security proofs are also given.

3.5. Comparison of the Secret Sharing Schemes

In Table 3.1, we summarize some of the properties of the well-known schemes. Here, we use the following abbreviations: **Ac.** for access structure, $\mathcal{S}$ for secret domain, **P** for perfectness, **I** for idealness, **H** for homomorphism, **Comp.** for recovery complexity, r for threshold value, T for threshold, Y for yes, N for no, A for Asymptotic, Add. for additive and flex. for flexible. Int. refers to $(M^{(r-1)}, M_{(r)})$ interval.

Table 3.1. Comparison of the well-known schemes.

	Based on	**Ac.**	$\mathcal{S}$	**P**	**I**	**H**	**Comp.**	r
Additive	XOR	T	$\mathbb{Z}_{2^n}$	Y	Y	Add.	$\mathcal{O}(n)$	$n-1$
Shamir	Lagrange	T	$\mathbb{Z}_p$	Y	Y	Add.	$\mathcal{O}(r \log^2 r)$	flex.
Asmuth-Bloom	CRT	T	$\mathbb{Z}_p$	A	A	-	$\mathcal{O}(r)$	flex.
Mignotte	CRT	T	Int.	N	N	-	$\mathcal{O}(r)$	flex.

4. HOMOMORPHIC EXTENSIONS TO ASMUTH-BLOOM SSS

Asmuth-Bloom SSS does not have a homomorphic property because of the overflow. *Overflow problem* is the uncertainty of whether the homomorphic operation over the blinded secrets will exceed the valid upper bound or not. The recovery requires the knowledge of overflow amount. For a single secret the bound of blinded secret $(0, M_{(r)})$ is feasible to recovery because for any qualified group $G \in \mathcal{A}$, $M_G \geq M_{(r)}$ is guaranteed. However, in the case of homomorphic operations ($\otimes$) like addition, the participants in group G cannot know whether or not $y \otimes y' < M_{(r)}$. By CRT, they can only determine unique solution in $(\mathrm{mod} M_G)$. If there exists $y, y' \in (0, M_{(r)})$ such that $M_G < y \otimes y'$, the group G cannot recovery secret because of the uncertainty of overflow.

Nonetheless, we have found out that some modifications can make Asmuth-Bloom SSS homomorphic. In this chapter, we will present three possible solutions to get homomorphic secret sharing based on Asmuth-Bloom. All of these solutions can be adapted to Mignotte's SSS since they have similar structure. We choose Asmuth-Bloom since its parameters and level of security seem to be more reliable.

Let analyze the homomorphic property in mathematical form. By using the Benaloh's formula, Asmuth-Bloom SSS would be homomorphic over $\otimes$ ('+' or '$\times$') if the following equivalent equations hold:

$$\begin{aligned} \exists \odot \text{ such that } \mathbf{RF}(\mathbf{S}) \odot \mathbf{RF}(\mathbf{S}') &= \mathbf{RF}(\mathbf{S} \otimes \mathbf{S}') \\ ((y \odot y') \bmod M_{(r)}) \bmod p &\equiv S \otimes S' \bmod p \\ ((A \cdot p + S) \odot (A' \cdot p + S') \bmod M_{(r)}) \bmod p &\equiv S \otimes S' \bmod p \end{aligned} \tag{4.1}$$

where x and x' correspond to secrets, y and y' do blinded secrets. Here, it is reasonable to assume that '$\odot$' and '$\otimes$' are the same operations. As seen in Theorem 3.6, Equation 4.1 holds for all secret pairs if and only if $p | M_{(r)}$. In the case of $gcd(p, M_{(n)}) = 1$, two

methodology can be used to deal with the overflow:

- Use knowledge of its overflow amount
- Prevent the overflow by restricting the blinding factor.

Theorem 4.1. *By including the knowledge of* $\left\lfloor \frac{y+y'}{M_{(r)}} \right\rfloor$ *to the Asmuth-Bloom scheme, somewhat additively homomorphic SSS can be obtained.*

Proof Sketch. If the participants in group G have a closer upper and lower bounds for $y + y'$, then they may not need the Equation 4.1. More specifically, they can already calculate the value of $(y+y' \bmod M_G)$ by just adding their shares, if they could also calculate $\left\lfloor \frac{y+y'}{M_G} \right\rfloor$, then they can uniquely determine $y + y' = \left\lfloor \frac{y+y'}{M_G} \right\rfloor \cdot M_G + (y + y' \bmod M_G)$. From that, obviously, $x + x'$ can be extracted, $x + x' = y + y' \bmod p$. □

To sum up, there are three possible ways to make Asmuth-Bloom homomorphic:

- In order to satisfy $p \mid M_G$ for all $G \in \mathcal{A}$: Change structure of the Asmuth-Bloom sequence.
- In order to know $\left\lfloor \frac{y+y'}{M_{(r)}} \right\rfloor$: Share the blinding factor A.
- In order to get $y \otimes y' < M_{(r)}$: Restrict y.

4.1. Sharing the Blinding Factor A

In the first modified version of Asmuth-Bloom SSS, we also share blinding factor A, in addition to the secret. The knowledge of A provides that of closer bounds for blinded secret. As briefly, mentioned in Theorem 4.1, more information about y gives more advantage to recovery. In this case, we use it to recover the exact value of y and this enables homomorphic additions since we do not have to worry about the overflow. Let $(p, m_1, m_2, \cdots, m_n)$ denotes the Asmuth-Bloom sequence used for sharing S, and $(p', m'_1, m'_2, \cdots, m'_n)$ for sharing A. Since A is between 0 and $\frac{M_{(r)}}{p} - 1$, p' can be approximated to $M^{(r-1)}$. How to share A and the security concerns are explained in the following section.

4.1.1. Modified Asmuth-Bloom SSS v1

Algorithm of the scheme can be seen in Figure 4.1. With respect to the notation we used, for this part we will add the following ones:

$$I'_G = \bigcup_{P_i \in G} I'_i, \quad M'_{(r)} = \prod_{i=1}^{r} m'_i, \quad M'^{(r-1)} = \prod_{i=1}^{r-1} m'_{n-i+1}, \quad M'_G = \prod_{P_i \in G} m'_i.$$

Distribution Phase I

The dealer chooses an Asmuth-Bloom sequence $(p, m_1, m_2, \cdots, m_n)$.

He chooses an arbitrary A such that $y = S + A \cdot p < M_{(r)}$.

He computes and distributes the first part of the shares $I_i = y \bmod m_i$.

Distribution Phase II

The dealer chooses an another Asmuth-Bloom sequence $(p', m'_1, m'_2, \cdots, m'_n)$ where the fourth condition replaced by $(M^{(r-1)} + 1) \cdot p' \cdot M'^{(r-1)} < M'_{(r)}$

He chooses an arbitrary A' such that $y' = A + A' \cdot p' < \lfloor \frac{M'_{(r)}}{M^{(r-1)}+1} \rfloor$.

He computes and distributes the second part of the shares $I'_i = y' \bmod m'_i$.

Reconstruction Phase

Let G be a group of participants gathered to reconstruct the secret S and the blinding factor A by using CRT;

$$\mathbf{RF}(I_G, I'_G) : \begin{cases} \text{if } |G| \geq r & \implies y = \left[\sum_{P_i \in G} \frac{M_G}{m_i} \cdot \left[\frac{m_i}{M_G} \right]_{m_i} \cdot I_i \right]_{M_G} \\ & \implies S = y \bmod p. \\ & \implies y' = \left[\sum_{P_i \in G} \frac{M'_G}{m'_i} \cdot \left[\frac{m'_i}{M'_G} \right]_{m'_i} \cdot I'_i \right]_{M'_G} \\ & \implies A = y' \bmod p'. \\ \text{if } |G| < r & \implies \perp. \end{cases}$$

Figure 4.1. Asmuth-Bloom SSS Modified Version 1.

Proposition 4.2. *Security level of the modified scheme is the same with the original one in the sense that no unqualified group of participants can eliminate a possible value of secret S if and only if second sharing part does not leak any information about A.*

Proof. $\Longleftarrow$:
It is important to note that since any two share sets of (S_1, A_1) and (S_2, A_2) are assumed to be independent of each other, thus homomorphic operation does not leak any information more than the scheme does for one share set. For that reason, we will look into one share set and its security issues. Since the only difference of the modified version is the sharing of the blinding factor, it would be enough to investigate its effect on the security of system.

In the case of an unqualified group of participants, $B \in \overline{\mathcal{A}}$, the knowledge of one of the shares, I_B or I'_B, cannot also reduce the possible solution set by using the Theorem 3.3. Therefore, only case that need to be taken into account is the usage of both shares of a group participants.

The information that can be gathered by group B about S and A is $(y \bmod M_B)$ and $(y' \bmod M'_B)$. Since for any $i, j \in \{1, \ldots, n\}$, $(m_i, m'_j) = 1$ is guaranteed by dealer, M_B and M'_B are also co-primes. If B is an unqualified group of participants, then $M_B < M_{(r)}/p$ assures that there are at least p possible values for A that cannot be eliminated with the knowledge of $(y \bmod M_B)$. Also, these solutions contain all possible values for S. Another important point is that $M'_B < M'_{(r)}/p'$ guarantees that the second scheme cannot eliminate any possible value of A. Therefore, the possible set of solutions for A cannot reduced any more than the reduction done in the former sharing, sharing of y. For that reason, none of the possible solutions for S could be eliminated by an unqualified group of participants.

$\Longrightarrow$:
In order to keep the security level of the scheme with the same as before, additional sharing should not eliminate a possible value of S. In other words, as we mentioned

before, in the original version an unqualified group of participants, $B \in \overline{\mathcal{A}}$, can learn $(y \bmod M_B)$ and $p \cdot M_B < M_{(r)}$ inhibits them to eliminate any possible solution for the secret.

Note that with the knowledge of $(y \bmod M_B)$, the number of possible solutions for S is the same with the number of solutions for A unless possible S values start to repeat themselves. It can be concluded that if the second sharing scheme causes an elimination of a possible A value from the possible (A, S) set gathered by the group B, this may reduce the possible solution set for the secret. □

Since Asmuth-Bloom SSS is not perfect all possible solutions may not seen the same amount for both S and A. Using both sharing schemes together may increase the difference between them. For example, let say two A encountered by one more time than the rest in the second sharing scheme and they corresponding the same S value in the possible solutions of the first sharing scheme, then that value has two more element in the possible solution set. On the other hand, using the fact that both sharing schemes are independent and the difference comes from an individual scheme is not more than 1 (see Theorem 3.4), the difference between solutions of S can be assumed to be negligible. In addition to that, using the methods defined in [28, 51] assures that all possible solutions asymptotically has the same probability.

4.1.1.1. Additive Homomorphism. This modified version of Asmuth-Bloom can have additive homomorphic property to some point. When it comes to multiplication, sharing A is not enough to recover the secret. This is because multiplication of two blinded secrets, y_1 and y_2, will give the following equation:

$$y_1 \times y_2 = (A_1 \cdot p + S_1) \times (A_1 \cdot p + S_1) = A_1 \cdot A_2 \cdot p^2 + (A_1 \cdot S_2 + A_2 \cdot S_1) \cdot p + S_1 \cdot S_2.$$

In order to extract $S_1 \cdot S_2$ the rest of the equation should be known. However, the second part, $(A_1 \cdot S_2 + A_2 \cdot S_1)$, is not known by the participants since S_1 and S_2 are not known individually. On the other hand, the number of homomorphic addition can

be done by this scheme is given in the following proposition. Homomorphic addition is operated by adding individual shares as in the Shamir's SSS.

Proposition 4.3. *Sharing blinding factor A enables $M^{(r-1)}$ homomorphic addition.*

Proof. Assume that the scheme can do *k-homomorphic addition*. Since secrets are chosen over $\mathbb{Z}_p$, then the aim is to be able to determine $\left(\left(\sum_{i=1}^{k} S_i\right) \bmod p\right)$. Let $(y_i, S_i, A_i)_{i=1}^{k}$ correspond to the k secrets, their blinding factors and blinded secrets:

$$\forall i \in \{1, \ldots, k\} \quad y_i = S_i + A_i \cdot p \quad \text{where} \quad S_i < p,\, y_i < M_{(r)}.$$

The upper and lower bounds for the addition of the k blinded secret can be seen in the following equation:

$$(\sum_{i=1}^{k} A_i) * p \leq \sum_{i=1}^{k} y_i = \sum_{i=1}^{k} S_i + (\sum_{i=1}^{k} A_i) * p \leq (p-1) \cdot k + (\sum_{i=1}^{k} A_i) * p. \tag{4.2}$$

By an appropriate choice for the secret range of the second sharing (share of A), a qualified group of participants, G, can determine the exact value of $\sum_{i=1}^{k} A_i$ and so the upper and lower bounds in (4.2). In order to determine $\sum_{i=1}^{k} A_i$, we are using the same idea used in [12] where they restrict the blinded secrets and change the fourth condition of Asmuth-Bloom accordingly. Similarly, by simply adding first component of the shares individually, they can determine $\left(\left(\sum_{i=1}^{k} y_i\right) \bmod M_G\right)$.

Group G can determine the exact value of $\sum_{i=1}^{k} y_i$ if $(p-1) \cdot k < M_G$ because of the fact that there is at most one possible value satisfying the modular condition $\left(\left(\sum_{i=1}^{k} y_i\right) \bmod M_G\right)$ within the bounds given by (4.2).

For the case of k is equal to $M^{(r-1)}$, $(p-1) \cdot k < M_G$ is already satisfied in fourth condition of the distribution phase. After that, $\left(\left(\sum_{i=1}^{k} S_i\right) \bmod p\right)$ can be easily extracted; $\left(\left(\sum_{i=1}^{k} S_i\right) \bmod p\right) = \left(\left(\sum_{i=1}^{k} y_i\right) \bmod p\right)$. □

Information Rate. Propositions 4.2 and 4.3 require an appropriate secret range for the second sharing so that $\sum_{i=1}^{M^{(r-1)}} A_i$ can be uniquely determined by an qualified group. Therefore, secret size of the second mechanism p' should be greater than or equal to $\sum_{i=1}^{M^{(r-1)}} A_i \leq \sum_{i=1}^{M^{(r-1)}} M^{(r-1)} = (M^{(r-1)})^2$. In this case, the information rate of the whole scheme can be calculated as:

$$\begin{aligned} \rho = \frac{\log|\mathcal{S}|}{\log|\mathcal{S}_{shares}|} &= \frac{\log|\mathcal{S}|}{\log|\mathcal{S}^1_{shares}| + \log|\mathcal{S}^2_{shares}|} \\ &\approx \frac{\log p}{\log m_i + \log(M^{(r-1)})^2} \approx \frac{1}{1+2\cdot(r-1)} = \frac{1}{2\cdot r-1} \end{aligned}$$

where $\mathcal{S}$ denotes the set of secret, $\mathcal{S}^1_{shares}$ and $\mathcal{S}^2_{shares}$ denote sets of shares of y and A accordingly. Here, we use the same idea given in Section 3.4 saying that the information rate of Asmuth-Bloom scheme goes to 1 as the integer set increases. Similarly, as the $(p', m'_1, m'_2, \ldots, m'_n)$ set increases the information rate of second scheme goes to 1 and it corresponds to $\frac{1}{2\cdot(r-1)}$ of the general scheme because $p' \approx p^{2\cdot(r-1)}$.

Note that inside this scheme we used another method for additively homomorphism where the fourth condition of Asmuth-Bloom sequence changed to compute summation of A's. The latter method is much more efficient in a sense of share sizes. Therefore, instead of sharing A and using the latter as a tool inside of the scheme, just changing the fourth condition would be much easier and elegant solution. It would be an efficient solution, if there would be an elegant methodology to share A in a way that after recovery a qualified group G gets only $\frac{\sum_{i=1}^{k} A_i}{M_G}$ instead of the whole summation $\sum_{i=1}^{k} A_i$ which is redundant. The next section consist of a method which includes modifications of the fourth condition and the range of the blinded secret in order to get not only homomorphic additions but also homomorphic multiplications.

4.2. Restriction on the Blinded Secret y

In the second modified version of Asmuth-Bloom SSS, we did not change the basic structure. What we did is to modify the fourth condition of the Asmuth-Bloom sequence. Iftene [50] used a similar idea onto Mignotte's SSS without mentioning

security level of the scheme. His proposed method makes Mignotte's scheme somewhat additively homomorphic. The original condition of Asmuth-Bloom sequence gives a sharp bound satisfying the minimum share size and properties of a threshold scheme. If the condition is relaxed then it loses from either the security level or share size, but also it may gain homomorphic properties.

There are two possible ways to do relax the fourth condition; changing threshold scheme into ramp scheme or increasing the share sizes. We combined both techniques to get homomorphism. Our scheme enables K_a homomorphic addition and K_m homomorphic multiplication. Detailed explanations are given in the following section.

4.2.1. Modified Asmuth-Bloom SSS v2

In this version, we changed the threshold scheme into ramp SSS by adjusting secrecy and recovery bounds accordingly. As we mentioned in Section 2.3, ramp schemes can be seen as unstrained versions of the threshold schemes. Unlike (r, n) threshold schemes, (r, s, n) ramp schemes assure (perfect) secrecy for s or any fewer participants, instead of $r - 1$. With the help of this flexibility, ramp schemes can have additional properties like homomorphism in this case.

This version, which can be seen in Figure 4.2, includes modifications on the fourth condition and upper bound for the blinded secret. There is a trade-off between security and homomorphic abilities; as s value, in other words the security threshold for secrecy, increases the number of homomorphic operations which can be handled by the scheme decreases. s and K_m are inversely proportional and their relation can be seen as $(s + 1) \cdot (K_m + 1) \approx r - 1$.

Proposition 4.4. *The scheme in Figure 4.2 satisfies the security and recovery requirements of an (r, s, n) ramp scheme.*

Proof. Let M denotes the range of blinded secret after some homomorphic operations, i.e. $y \in (0, M)$. $M_{(r)} > M$ and $M^{(s)} \cdot p < M$ are guaranteed by the condition given in

Distribution Phase

The dealer chooses an Asmuth-Bloom sequence $(p, m_1, m_2, \cdots, m_n)$ where the fourth condition replaced by

$$M_{(r)} > (K_a + 1) \cdot \left(p \cdot M^{(s)}\right)^{K_m+1}. \tag{4.3}$$

He chooses an arbitrary A such that $y = S + A \cdot p < p \cdot M^{(s)}$.

He computes and distributes shares $I_i = y \bmod m_i$.

Reconstruction Phase

Let G be a group of participants gathered to reconstruct the secret,

The participants compute the secret S by using CRT;

$$\mathbf{RF}(I_G) : \begin{cases} \text{if } |G| \geq r & \implies y = \left[\sum_{P_i \in G} \frac{M_G}{m_i} \cdot \left[\frac{m_i}{M_G}\right]_{m_i} \cdot I_i\right]_{M_G} \\ & \implies S \equiv y \bmod p. \\ \text{if } |G| < s+1 & \implies \perp. \end{cases}$$

Figure 4.2. Asmuth-Bloom SSS Modified Version 2.

Equation (4.3) and homomorphic operation limits. The rest is similar to Theorem 3.3;

- For an unqualified group of participants, the solution set for y can be narrow down to unique solution in at most mod$M^{(s)}$. Since $M^{(s)} < M/p$, they would have at least p possible solutions for y. Because $gcd(m_i, p) = 1$ for any i, all possible solutions for S will be in the possible solution set of the group.
- For any qualified group of participants would have unique solution in at least $M_{(r)}$ for y. Since $y < M < M_{(r)}$, their solution should be the true value of y and S can be easily obtained. □

4.2.1.1. Somewhat Homomorphism. Since our scheme has limited number of homomorphic operations, it is a somewhat homomorphic scheme where the limits are K_a for

addition and K_m for multiplication.

Boot Strapping. Even though Gentry [5] proposed this novel technique for homomorphic encryption, it can be used for homomorphic secret sharing. Instead of re-encryption, we propose to re-share of the shares in order to get fresh secret. Since each homomorphic operation increases the size of the blinded secret, at some point its size will reach to the upper bound, $M_{(r)}$. After that point, no homomorphic operation can be done because of the overflow problem. On the other hand, re-sharing that secret would reduce its size, and this enables additional homomorphic operations. For that reason, re-sharing provides unlimited operations, and makes this scheme fully homomorphic with the expense of communicational cost.

Additive Homomorphism. The bound of addition operations K_a only affects the share size of the scheme whereas K_m determines the secrecy threshold, $s \approx \frac{r-1}{K_m+1}$. In other words, cost of an addition is much less than cost of a multiplication. If we consider just additively homomorphic scheme while keeping threshold property, the only parameter that will be changed is the share size. In addition to that, the increase in the share sizes is much less than the one when using the previous version given in Figure 4.1.

Information Rate. Share sizes of this scheme depends on the limit of homomorphic additions. In the original scheme, it is assumed that as the parameters in Asmuth-Bloom sequence increases the information rate goes to 1. Here, instead of p, consider $p\cdot(K_a+1)$. Then, the share sizes could be assumed to have a size of $p\cdot(K_a+1)$. In that case, information rate of the scheme would be approximated as $\frac{1}{1+\log_p K_a}$.

4.3. Overwhelming the Overflow Problem

In this section, first we will show that it is not possible to find Asmuth-Bloom sequences which lead to homomorphic properties, without affecting the security of the scheme if Equation (3.4) is satisfied. Nonetheless, in Figure 4.3, we define a construction satisfying Equation (3.4). We determine its security level by using the following theorems. After that, we will give the security proof of our proposed version which uses secret splitting as an additional tool.

Theorem 4.5. *It is not possible to satisfy the same security level of Asmuth-Bloom SSS with* $(p, M_{(n)}) \neq 1$. *In the case of Equation 3.4, the security level of the modified version of Asmuth-Bloom SSS will be* $\min_{i \in \{1,\dots,n\}} gcd(m_i, p)$.

Proof. Firstly, if $(p, M_{(n)}) \neq 1$, then the following equation shows that the scheme leaks information about the secret:

$$\begin{aligned} &(p, M_{(n)}) \neq 1 \implies \exists m_i \quad \text{such that} \quad gcd(m_i, p) = d \neq 1, p = d \cdot x, m_i = d \cdot y \\ &y \equiv y_i \bmod m_i \implies A \cdot d \cdot x + S \equiv y_i \bmod d \cdot l \implies y_i \equiv S \bmod d. \end{aligned} \tag{4.4}$$

(4.4) states that the participant P_i knows the $S \bmod d$ just using his own share. For that reason, the security characteristic of the scheme should be less than or equal to $\frac{1}{d}$. Since $d > 1$, this scheme cannot be perfect.

Let $G = \{P_{n-r+1}, P_{n-r+2}, P_{n-r+3}, \dots, P_n\}$ and $B_i = G - \{P_i\}$ where $G \in \mathcal{A}$ and $B_i \in \overline{\mathcal{A}}$ for $i = n - r + 1, \dots, n$. Theorem 3.3 and 3.4 states that there are about p possible solutions of the secret for each group B_i where it is assumed Asmuth-Bloom sequence is constructed with the primes of the same size. Since B_i also know $S \bmod M_{B_i}$, possible solution set can be reduced to $\frac{p}{gcd(p, M_{B_i})}$. Using Equation 3.4, the security level of the scheme should be less than or equal to $\min_{i \in \{n-r+1,\dots,n\}} \frac{p}{gcd(p, M_{B_i})} = \min_{i \in \{1,\dots,n\}} gcd(m_i, p)$. □

Distribution Phase

The dealer chooses a set of integers $(p, m_1, m_2, \cdots, m_n)$ such that

(i) $m_1 < m_2 < \cdots < m_n$ and $S < p = \prod_{i=1}^{n} m_i$,

(ii) $gcd(m_i, m_j) = 1$ $(\forall\, i \neq j)$.

He computes and distributes shares $I_i = S \bmod m_i$.

Reconstruction Phase

Let G be a group of participants gathered to reconstruct the secret S;

$$\mathbf{RF}(I_G) : \begin{cases} \text{if } |G| = n \implies S = \left[\sum_{i=1}^{n} \frac{M_{(n)}}{m_i} \cdot \left[\frac{m_i}{M_{(n)}} \right]_{m_i} \cdot I_i \right]_{M_{(n)}}. \\ \text{if } |G| < n \implies \perp^{a}. \end{cases}$$

[a] Using the Theorem 4.5, security level of the scheme is $\min_{i \in \{1,\ldots,n\}} m_i$.

Figure 4.3. Asmuth-Bloom SSS with Fully Homomorphic Properties.

4.3.1. Modified Asmuth-Bloom SSS v3

Here, we use secret splitting idea with Asmuth-Bloom structure. Bozkurt [52] used a similar idea to construct *multipartite* SSS based on CRT. Our algorithm can be seen in Figure 4.4.

Proposition 4.6. *Our proposed scheme is perfect ramp SSS in the sense that no group with less than or equal to s participants can learn any information about the secret.*

Proof Sketch. An unqualified group of participants could have at most $n - 1$ shares. For each random value, r_i, they will have all possible solutions in some p_j with the same probability. It can be seen that for all p_j values the same outcome occurs. For that reason, at the end, they will have all possible solution for $\sum_{i=1}^{s+1} r_i$ with the same probability. Therefore, even for the group of $n - 1$ participants, S is indistinguishable in $\mathcal{S}$. □

Distribution Phase

The dealer chooses a set of integers $(p, m_1, m_2, \cdots, m_n)$ such that

(i) $m_1 < m_2 < \cdots < m_n$ and $S < p = \prod_{i=1}^{n} m_i$

(ii) $gcd(m_i, m_j) = 1$ $(\forall\, i \neq j)$

He randomly chooses $s+1$ integers $(r_1, \ldots, r_{s+1})$ in $\mathbb{Z}_p$.

He computes and distributes shares as $(I_i^j \equiv r_j \bmod m_{i+j-1})$[a]

where I_i^j corresponds to j^{th} share of participant i.

He announces $S_{public} = \left[S + \sum_{i=1}^{s+1} r_i\right]_{M_{(n)}}$ value.

Reconstruction Phase

Let G be a group of participants gathered to reconstruct the secret,

The participants compute the secret S by using CRT;

$$\mathbf{RF}\begin{pmatrix} \bigcup_{P_i \in G} I_i^1 \\ \vdots \\ \bigcup_{P_i \in G} I_i^{s+1} \end{pmatrix} : \begin{cases} \text{if } |G| = n \implies & \begin{pmatrix} r_1 = \left[\sum_{i=1}^{n} \frac{M_{(n)}}{m_i} \cdot \left[\frac{m_i}{M_{(n)}}\right]_{m_i} \cdot I_i\right]_{M_{(n)}} \\ \vdots \\ r_{s+1} = \left[\sum_{i=1}^{n} \frac{M_{(n)}}{m_i} \cdot \left[\frac{m_i}{M_{(n)}}\right]_{m_i} \cdot I_i\right]_{M_{(n)}} \end{pmatrix} \\ & \implies S = \left[S_{public} - \sum_{i=1}^{s+1} r_i\right]_{M_{(n)}}. \\ \text{if } |G| < s+1 \implies & \perp. \end{cases}$$

[a] all indices are given in mod n

Figure 4.4. Asmuth-Bloom SSS Modified Version 3.

4.3.1.1. Homomorphism. Our proposed scheme is *additively homomorphic* SSS. Using Theorem 3.6, it can be said that algorithm defined in Figure 4.3 is fully homomorphic. Since our algorithm can be seen as addition of several secrets using fully homomorphic Asmuth-Bloom sequence, addition of two secrets can be constructed with addition of the shares. On the other hand, it is not multiplicatively homomorphic because addition of randoms (r_i's) . Nonetheless, using the same idea in secret splitting scheme [44], multiplication can be provided with communication cost.

Information Rate. Since it is a ramp scheme, share sizes can be smaller than the secret size. Indeed, in our scheme, secret size is n/s times greater than share sizes where n is the number of participants and s is the secrecy threshold. This property can be handful for constraint environments where the secret size could be much larger than available data memory.

$s = n - 1$. In this case, the scheme will be very similar to the basic additive SSS and also satisfies the same properties like threshold property, perfectness, idealness and additive homomorphism.

4.4. Comparison of the Proposed Schemes

In Table 4.1, we summarize some of the properties of the proposed modifications on Asmuth-Bloom scheme. Here, we use the following abbreviations: **Ac.** for access structure, ρ for information rate, **I** for security characteristic, **H.A** for homomorphic addition, **H.M** for homomorphic multiplication, T for threshold and R for ramp, unlim. for unlimited.

Table 4.1. Comparison of the proposed versions of Asmuth-Bloom SSS.

		Ac.	ρ	κ	**H.A**	**H.M**
v1	(Figure 4.1)	T	$\frac{1}{2 \cdot r - 1}$	1	$M^{(r-1)}$	-
v2	(Figure 4.2)	R	$\frac{1}{1+\log_p K_a}$	1	K_a	K_m
v3	(Figure 4.4)	R	$\frac{n}{s}$	1	unlim.	-
v3.2	(Figure 4.3)	T	n	$\frac{1}{m_n}$	unlim.	unlim.

5. VERIFIABLE SSS BASED ON CRT

This chapter consists of a submitted joint work [53]. Here, we point out that two of the CRT-based VSS schemes [15, 16] are insecure and [12] is only secure against a computationally bounded adversary. We propose a modification for this scheme and prove that the modified scheme is a secure verifiable secret sharing scheme against an unbounded adversary. Lastly, as an application, we show how to use the new scheme for joint random secret sharing.

The original versions of the Asmuth-Bloom and Mignotte SSSs [9, 10] are not verifiable. The first CRT-based VSS scheme has been proposed by Qiong *et al.* in [13] which uses a similar approach to Pedersen's polynomial-evaluation-based VSS [38]. Later, Iftene proposed the only VSS based on Mignotte's scheme [14], and showed that the security of the scheme is based on the hardness of the *discrete logarithm problem*.

Kaya and Selçuk [12] proposed another VSS based on the Asmuth-Bloom scheme with robustness analyses of the Quiong *et al.*'s and Iftene's schemes [13, 14]. They showed that the existing schemes are not robust against a malicious dealer because the dealer can distribute inconsistent shares that lead to different reconstructed secrets for different qualified subsets. To solve this problem, they used a *range proof* to prove that the blinded secret y is in the desired (CRT) range. Their scheme assures the validity of the shares not only for malicious participants (reconstruction phase), but also for a malicious dealer (distribution phase).

Recently, two VSS schemes based on Asmuth-Bloom have been proposed [15,16]. In 2014, Harn *et al.* proposed a very efficient scheme aiming at detecting malicious behavior of the dealer with the assumption of the participants act honestly (which already makes the scheme insecure against an active adversary) [15]. The scheme uses additional verification-secrets generated within a given range. Based on these ranges, the participants can have a range guarantee on y. This assures that the dealer cannot distribute inconsistent shares. With the same motivation, Liu *et al.* [16] proposed a

VSS where every participant adds an adjusting value (from a guaranteed range since the participants are again assumed to be honest) to his share, then all the participants recover an adjusted value for y which is supposed to give no additional information but the range of the y.

5.1. Analysis of the Existing Schemes

5.1.1. Kaya and Selçuk's VSS Scheme

Instead of Boudot's range proof, Kaya and Selçuk [12] use the range proof technique in [54] as a black box. The algorithm can be seen in Figure 5.1.

Their scheme prevents malicious behavior of both dealer and participants in a way that misleading shares can be detected by the participants. Since the commitment is computationally hiding, the secret is leaked to an unbounded adversary. Furthermore, even a computationally bounded adversary can extract the secret from the commitment in the case of small sizes of p.

Lemma 5.1. *The order of $g \in \mathbb{Z}_Q$ is $M_{(n)}$.*

Proof Sketch. Let $ord(g) = d$ in $\mathbb{Z}_Q$. Since $g^d \equiv 1 \bmod q_i$, then $m_i \mid d$ (for all i's) which concludes to $M_{(n)} \mid d$.

Similarly, since $g^{M_{(n)}} \equiv 1 \bmod q_i$ (for all i's), then $g^{M_{(n)}} \equiv 1 \bmod Q$ by CRT which implies that $d \mid M_{(n)}$. Therefore, $d = M_{(n)}$. □

Lemma 5.2. *There is exactly one y value satisfying the commitment in* mod $M_{(n)}$.

Proof. Assume that y' and y'' satisfies the commitment such that $E(y') \equiv E(y'') \bmod Q$. By using Lemma 5.1:

$$E(y') \equiv E(y'') \bmod Q \implies 1 = g^{y'-y''} \bmod Q \implies ord(g) \mid y' - y''$$
$$\implies y' \equiv y'' \bmod M_{(n)} \implies y' = y''$$

Distribution Phase

The dealer chooses an Asmuth-Bloom sequence $(p, m_1, m_2, \cdots, m_n)$ with the following additional requirements

(i) $q_i = 2 \cdot m_i + 1$ is a prime number $(\forall i)$,

(ii) $M_{(r)} > p^2 \cdot M^{(r-1)}$.

He chooses an arbitrary A such that $y = S + A \cdot p < M_{(r)}$.

He commits $E(y) = g^y \bmod QN$ and $Range\_Proof(E(y), M)$ where

(i) $Q = \prod_{i=1}^{n} q_i$ and $g \equiv g_i \bmod m_i$ where $g_i \in \mathbb{Z}_{q_i}^*$ be an element of order m_i,

(ii) factorization of N is not known by the dealer and participants,

(iii) $Range\_Proof(E(y), M)$ symbolizes the commitment scheme in [54].

He computes shares as $I_i = y \bmod m_i$ and privately distributes $(\forall i)$.

He announces $E(y), Range\_Proof(E(y), M)$.

Verification Phase

All participants check

(i) the validity of their shares: $g_i^{I_i} \stackrel{?}{\equiv} E(y) \bmod p_i$,

(ii) the validity of range proof: $Range\_Proof(E(y), M)$.

Reconstruction Phase

Let G be a group of participants gathered to reconstruct the secret,

share of $P_i \in G$ is verified by other participants: $g_i^{I_i} \stackrel{?}{\equiv} E(y) \bmod p_i$.

After all the shares are verified, G may reconstruct S using CRT.

Figure 5.1. Kaya and Selçuk's VSS.

since $y', y'' \in (0, M_{(r)})$, which implies that only one element satisfies the commitment. □

Theorem 5.3. *Kaya and Selçuk's VSS scheme is insecure against a bounded passive adversary since the secret value can be found by $\mathcal{O}(p^2)$ operations independent of the security parameters.*

Proof. From Theorems 3.3 and 3.4, it follows that an unqualified group B can compute $y \bmod M_B$, thus there are at most $\frac{M_{(r)}}{M_B} + 1$ possible solutions (denoted by PS_B) for group B ($|PS_B| \leq \frac{M_{(r)}}{M_B} + 1$). By using Lemma 5.2, trying all values $y_B \in PS_B$ in the commitment would give the exact one: $E(y) \stackrel{?}{=} E(y_B) = g^{y_B} \bmod Q$.

For the VSS using the original Asmuth-Bloom sequence, the time complexity of the attack is $\frac{M_{(r)}}{M_B} + 1 \approx p$, whereas for the case of modified Asmuth-Bloom given in [29], the time complexity will be $\mathcal{O}(p^2)$. □

An attack on this scheme is feasible for small (i.e. 32-bit) secret ranges and insecure against a bounded passive adversary.

5.1.2. Harn *et al.*'s VSS Scheme

Harn *et al.*'s VSS scheme [15] aims to provide the range-proof of the blinded secret, i.e., it just assures that the dealer choose y between 0 and $M_{(r)}$; all participants are assumed to be honest. The algorithm of Harn *et al.*'s VSS can be seen in Figure 5.2. Detailed explanations can be found in [15].

Theorem 5.4. *The VSS [15] in Figure 5.2 is not secure against a passive adversary.*

Proof. We assume that all the participants and the dealer act honestly. Therefore, the verification part is assumed to be completed without any flaws.

For simplicity, we use $B = \{P_{n-r+2}, P_{n-r+3}, \ldots, P_n\}$ as unqualified group of

Distribution Phase

The dealer chooses an Asmuth-Bloom sequence $(p, m_1, m_2, \cdots, m_n)$.

He chooses an arbitrary A such that $y = S + A \cdot p < M_{(r)}$.

He chooses k verification secrets $y_1, \ldots, y_k$ such that

(i) $M^{(r-1)} < y_i$, $y_i < y$ and $y + y_i < M_{(r)}$ $(i = 1, \ldots, k)$.

He computes shares as $I_{i,0} = y \bmod m_i$, $I_{i,j} = y_j \bmod m_i \quad (j = 1, \ldots, k)$

and privately distributes $(\forall i)$.

Verification Phase

All participants check the validity of the range of y;

(i) Open randomly chosen half of the verification secrets, say $y_1^{(1)}, \ldots, y_{k/2}^{(1)}$.

- Check that $M^{(r-1)} \stackrel{?}{<} y_i^{(1)} \stackrel{?}{<} M_{(r)}$ $(i = 1, \ldots, k/2)$.

(ii) Divide the rest into two sets such that and calculate $y - y_{i_1}^{(2)}$'s, $y + y_{i_2}^{(2)}$'s.

- Check that $0 \stackrel{?}{<} y - y_{i_1}^{(2)}$ and $y + y_{i_2}^{(2)} \stackrel{?}{<} M_{(r)}$ $(i_1, i_2 = 1, \ldots, k/4)$.

Reconstruction Phase

The same with Asmuth-Bloom SSS.

Figure 5.2. Harn *et al.*'s VSS.

participants such that the group moduli M_B is equal to $M^{(r-1)}$, i.e., B knows $y' = y \bmod M^{(r-1)}$. However, for any unqualified group of participants, the scheme does not provide secrecy as Asmuth-Bloom does in Theorem 3.3.

In the first part of the *verification phase*, every participant learns the exact values of $y_i^{(1)}$'s for $i = 1, \ldots, k/2$. From that $y_{max}^{(1)}$ can be calculated where $y_{max}^{(j)} = \max_{i=1}^{k/2} y_i^{(j)}$ and $y_{max} = \max_{i=1}^{k} y_i$. Since $\{y_i^{(1)}\}$ is chosen randomly from $\{y_i\}$ set, the probability distribution of $y_i^{(1)}$'s is the same with that of $y_i^{(2)}$'s. Using this observation, it can be assumed that $y_{max}^{(2)} \approx y_{max}^{(1)}$.

In the second part of the *verification phase*, every participant verifies that y is between $\max\{y_{i_1}^{(2)}\}$ and $\min\{M_{(r)} - y_{i_2}^{(2)}\}$. This information and the approximation $y_{max}^{(2)} \approx y_{max}^{(1)}$ state that $y \in \left(y_{max}^{(1)} - \mathcal{O}(1), M_{(r)} - y_{max}^{(1)} + \mathcal{O}(1)\right)$.

With respect to the range of y_i's, there are two cases:

(i) If $M_{(r)} - M^{(r-1)} < 2 \cdot y_{max}$, then there exists a unique solution for the group B:

$$M_{(r)} - M^{(r-1)} < 2 \cdot y_{max} \implies (M_{(r)} - y_{max}^{(1)}) - (y_{max}^{(1)}) < M^{(r-1)} + \mathcal{O}(1)$$
$$\implies \exists!\, y \text{ such that } y \equiv y' \bmod M^{(r-1)} \text{ and } y \in \left(y_{max}^{(1)}, M_{(r)} - y_{max}^{(1)}\right).$$

Note that, such large y_i is not necessary for a valid attack. In order to reduce the possible solution set of the secret, $(M_{(r)} - M^{(r-1)})/(p-1) < y_{max}$ would be enough.

(ii) On the other hand, if $M_{(r)} - M^{(r-1)} > 2 \cdot y_{max}$, then the adversarial group can narrow the range of y using $y \pm y_i^{(2)}$'s values calculated in the second part. Since $K_{i_1} = y - y_{i_1}^{(2)}$'s and $K_{i_2} = y + y_{i_2}^{(2)}$'s are calculated in the second part, the adversary knows that y is between $(K_{i_1}, K_{i_1} + y_{max})$ and also $(K_{i_2} - y_{max}, K_{i_2})$. Therefore, even using just one $y \pm y_i^{(2)}$, the possible solutions of y is restricted to an interval about y_{max}, which is less than $M_{(r)}/2$. It follows that the possible solution set of the secret for B is less than $\frac{M_{(r)}/2}{M^{(r-1)}} \approx p/2$. Note that, in case of

all $y \pm y_i^{(2)}$ values included, the solution set can be narrowed down to a smaller interval:

$$y \in \left(\max_{i_1,i_2}\{K_{i_1}, K_{i_2} - y_{max}\}, \min_{i_1,i_2}\{K_{i_1} - y_{max}, K_{i_2}\} \right).$$

Moreover, if the upper bound of y_{max} is much less than $M_{(r)}$, for example if $(M_{(r)} - M^{(r-1)})/(p-1) > y_{max}$, the secret can be easily extracted by B. □

5.1.3. Liu *et al.*'s VSS Scheme

In the scheme of [16], every participant, assumed to being honest, adds an adjusting value to his share, then all the participants recover an adjusted value for y which is supposed to give no additional information but the range of the y. It does not check the validity of shares given by the participants in the recovery part. The algorithm of Liu *et al.*'s VSS can be seen in Figure 5.3. Detailed explanations can be found in [16].

Distribution Phase

The dealer chooses an Asmuth-Bloom sequence $(p, m_1, m_2, \cdots, m_n)$.

He chooses an arbitrary A such that $y = S + A \cdot p \in (M^{(r-1)} + 2T, M_{(r)} - 2T)$ where $T = \sum_{i=1}^{n} m_i$.

He computes and distributes shares $I_i = y \bmod m_i$ $(\forall i)$.

Verification Phase

All participants check the validity of the range of y;

(i) Each participant P_i selects an adjusting value, $\lambda_i \in (-(m_i - 1), m_i - 1)$ and broadcasts the value $y_i^{(adj)} = M_{(n)}/m_i \cdot \left[m_i/M_{(n)}\right]_{m_i} \cdot I_i + \lambda_i$.

(ii) They calculate adjusted value $y^{(adj)} = \left[\sum_{i=1}^{n} y_i^{(adj)}\right]_{M_{(n)}}$.

- Check that $y^{(adj)} \stackrel{?}{\in} (M^{(r-1)} + T, M_{(r)} - T)$.

Reconstruction Phase

The same with Asmuth-Bloom SSS.

Figure 5.3. Liu *et al.*'s VSS.

Theorem 5.5. *The VSS proposed by Liu et al. [16] is insecure against a passive adversary.*

Proof. First of all, in the verification phase, every participant will learn $y^{(adj)}$. An adversarial group B can compute $y' = y \bmod M_B$ using their own shares. If $T \ll M_B$ (which in practice is satisfied for all of the unqualified groups with $t-1$ participants) then using y' and $y^{(adj)}$ values, the exact value of y can be easily found, since it is already known that $y^{(adj)} - T < y < y^{(adj)} + T$, and only one value in that interval satisfies the modulo condition y'. □

Note that since m_i's are large primes and assumed to be close to each other, $|B| \geq 2$ implies that $T \ll M_B$. In any case, for $B = \{n-1, n\}$ this condition is already satisfied:

$$M_B = m_n \cdot m_{n-1} \gg m_n \cdot n > \sum_{i=1}^{n} m_i = T.$$

5.2. CRT-based VSS Secure Against an Unbounded Adversary

As shown before, Kaya and Selçuk's VSS [12] is vulnerable because of the computationally hiding commitment they used. In the proposed scheme, we use Fujisaki-Okamoto commitment $E(y, x) = g^y \cdot h^x \mod Q$ and Boudot's range proof. This is more challenging than [12], since the random value x needs to be collectively constructed by the participants in a way that the participants can then verify their shares by using $E(y, x)$. Morever, we used the modified version of Asmuth-Bloom given in [29] to make it compatible with Kaya and Selçuk's VSS. However, using the original Asmuth-Bloom , the same security level can be obtained (see Theorem 5.7). The proposed VSS scheme is described in Figure 5.4.

Distribution Phase

The dealer chooses an Asmuth-Bloom sequence $(p, m_1, m_2, \cdots, m_n)$ with the following additional requirements

(i) $q_i = 2 \cdot m_i + 1$ is a prime number $(\forall i)$,

(ii) $M_{(r)} > p^2 \cdot M^{(r-1)}$.

The dealer chooses an arbitrary A such that $y = S + A \cdot p < M_{(r)}$.

Each participant P_i chooses a random $x_i \in \mathbb{Z}_{m_i}$ and sends it to the dealer.

The dealer commits $E(y, x) = g^y \cdot h^x \mod Q$ and $Range\_Proof(E(y, x), M_{(r)})$ where

- $x \equiv x_i \mod m_i$ $(\forall i)$,
- $Q = \prod_{i=1}^{n} q_i$ and $g \equiv g_i \mod q_i$ where $g_i \in \mathbb{Z}_{q_i}^*$ be an element of order m_i,
- $h \equiv h_i \mod q_i$ where h_i is an element of the group generated by g_i,
- $Range\_Proof(E(y, x), M_{(r)})$ symbolizes the commitment scheme in [55].

The dealer computes shares as $I_i = y \mod m_i$ and privately distributes $(\forall i)$.

Then, he announces $E(y, x), Range\_Proof(E(y, x), M_{(r)})$.

Verification Phase

All participants check the validity of the range proof of y,

$Range\_Proof(E(y, x), M_{(r)})$, and the validity of their shares:

$$g_i^{I_i} \cdot h_i^{x_i} \stackrel{?}{\equiv} E(y, x) \mod q_i. \tag{5.1}$$

Reconstruction Phase

Let G be a group of participants gathered to reconstruct the secret,

The share of $P_i \in G$ is verified by other participants: $g^{I_i} \cdot h^{x_i} \stackrel{?}{\equiv} E(y, x) \mod q_i$.

After all the shares are verified, G may reconstruct S using CRT.

Figure 5.4. Our Proposed VSS Scheme.

5.2.1. Analysis of the Proposed Scheme

Our scheme is based on the following assumptions: the factorization of N is unknown, the discrete logarithm problem (DLP) in $\mathbb{Z}_{q_i}^*$ is a computationally hard problem, and $\log_{g_i} h_i$ is not know by the dealer nor the participants.

There are unique g and h in $\mathbb{Z}_Q$ satisfying $g \equiv g_i \mod m_i$, $h \equiv h_i \mod m_i$ for all i's, and they can be computed by the CRT formula:

$$g = \left[\sum_{i=1}^{n} \frac{Q}{q_i} \cdot \left[\frac{q_i}{Q}\right]_{q_i} \cdot g_i\right]_Q, \; h = \left[\sum_{i=1}^{n} \frac{Q}{q_i} \cdot \left[\frac{q_i}{Q}\right]_{q_i} \cdot h_i\right]_Q.$$

Correctness. If the dealer and the participants are honest, then the verification phase passes.

$$\begin{aligned} E(y,x) \mod q_i &= g^y \cdot h^x \mod Q \mod q_i \\ &= g^y \cdot h^x \mod q_i \\ &= g_i^y \cdot h_i^x \mod q_i \\ &= g_i^{I_i} \cdot h_i^{x_i} \mod q_i. \end{aligned}$$

Lemma 5.6. *The discrete logarithm of h in base g is co-prime to $M_{(n)}$.*

Proof. Let $a = log_g h$ be the discrete logarithm of h in base g and $log_{g_i} h_i = a_i$, in other words $g_i^{a_i} \equiv h_i \mod q_i$, for each $i = 1, \ldots, n$. Then, it can be seen that $g^a \equiv h \mod Q$ where $a \equiv a_i \mod m_i$ for all i's. Since m_i's are primes and a_i's are not equal to zero, a and $M_{(n)}$ are co-primes. □

Theorem 3.3 states the security of Asmuth-Bloom secret sharing by showing the existence of a set of elements, $\mathcal{S}_B$, such that no element of $\mathcal{S}_B$ can be ruled out as a possible value of y. In Theorem 3.5, it is shown that the modified version of Asmuth-

Bloom in [29] is a statistical secret sharing scheme. We now show that the elements of $\mathcal{S}_B$ are also consistent with the additional information obtained by the adversary in the VSS scheme which concludes the following theorem:

Theorem 5.7. *For an unbounded passive adversary, no possible secret value can be ruled out, and the VSS is a statistical secret sharing scheme.*

Proof. Let B be an unqualified group of participants ($|B| \leq r-1$). B knows $\{I_i \equiv y \pmod{m_i} : i \in B\}, \{x_i \equiv x \pmod{m_i} : i \in B\}$, and the commitment $c = E(y,x) = g^y h^x \bmod Q$. Let $y' \in [0, M_B]$ be the unique solution to the congruences $I_i \equiv y' \pmod{m_i}$. Since the adversary is unbounded, he can compute the discrete logarithms $\log_g(c) = \log_g(E(y,x)) = \log_g(g^y h^x) = y + ax$, and $\log_g(h) = a$. It follows from (3.2) that

$$M_{(r)} > p^2 M^{(r-1)} \geq p^2 M_B.$$

Therefore, all elements of the set $\mathcal{S}_B = \{y', y' + M_B, \ldots, y' + p^2 M_B\}$ are possible solutions to the set of congruences $\{I_i \equiv y \pmod{m_i} : i \in B, y \in [0, M_{(r)}]\}$.

Likewise, we define x' as the unique solution in $\mathbb{Z}_{M_B}$ to the set of congruences $\{x_i \equiv x \pmod{m_i} : i \in B\}$, and the set $\mathcal{X}_B = \{x', x' + M_B, \ldots, x' + \frac{M_{(n)} - M_B}{M_B} M_B\}$ of possible solutions to the same set of congruences modulo $M_{(n)}$.

Let $\tilde{y}$ be an arbitrary element of $\mathcal{S}_B$. The solution to the congruence $\log_g(c) \equiv \tilde{y} + a\tilde{x} \pmod{\text{ord}(g)}$, with respect to $\tilde{x}$, is in $\mathcal{X}_B$: $\tilde{x} \equiv a^{-1}(\log_g(c) - \tilde{y}) \equiv a^{-1}((y - \tilde{y}) + ax) \pmod{\text{ord}(g)}$ (where the existence of $a^{-1} \bmod \text{ord}(g)$ follows from Lemmas 5.1 and 5.6). Since $y \equiv \tilde{y} \pmod{M_B}$, and $M_B \mid \text{ord}(g)$, $\tilde{x} \equiv x \pmod{M_B}$, so $\tilde{x} \in \mathcal{X}_B$. We conclude that the pair $(\tilde{y}, \tilde{x})$ is consistent with all information available to the adversary, so $\tilde{y}$ cannot be ruled out as a possibility for the true value of y.

Since $(M_B, p) = 1$ the set $\{\tilde{y} \bmod p : \tilde{y} \in \mathcal{S}_B\} = \mathbb{Z}_p$, so no possible secret value, $s \in \mathbb{Z}_p$ can be ruled out. From Theorems 3.3 and 3.5, it follows that the VSS is a

statistical secret sharing scheme. □

Consistency of the shares comes with the range proof; by completeness of the range proof, the participants can be sure that every qualified group of participants will get the same secret. Participants can check that their shares are actually derived from the blinded secret y by confirming Equation 5.1.

Theorem 5.8. *A computationally bounded corrupted dealer cannot distribute inconsistent shares without being detected.*

Proof Sketch. Since the random x is determined by participants, the dealer cannot give an inconsistent share without knowing a_i which contradicts with our assumption:

$$\begin{aligned} g^{I_i} \cdot h^{x_i} \equiv g^{I'_i} \cdot h^{x'_i} \mod q_i \quad &\iff \quad g_i^{I_i} \cdot h_i^{x_i} \equiv g_i^{I'_i} \cdot h_i^{x'_i} \mod q_i \\ &\iff \quad I_i + a_i \cdot x_i \equiv I'_i + a_i \cdot x'_i \mod m_i \\ &\iff \quad a_i = (I'_i - I_i) \cdot (x_i - x'_i)^{-1} \mod m_i. \end{aligned}$$

The range proof of y is based on the commitment scheme given by Boudot [55]. For that reason, it is enough to satisfy the requirements of that scheme. Since the proposed VSS scheme uses the bases (g, h) where $g \in \mathbb{Z}_Q^*$ and h is an element of the group generated by g with an unknown order, the range proof commitment is statistically secure in the case that factorization of N is unknown. □

Theorem 5.9. *A computationally bounded corrupted participant cannot cheat without being detected.*

Proof Sketch. Similar to Theorem 5.8, participant i cannot cheat unless he knows a_i which contradicts with the assumption:

$$\begin{aligned} g^{I_i} \cdot h^{x_i} \equiv g^{I'_i} \cdot h^{x'_i} \mod q_i \quad &\iff \quad g_i^{I_i} \cdot h_i^{x_i} \equiv g_i^{I'_i} \cdot h_i^{x'_i} \mod q_i \\ &\iff \quad I_i + a_i \cdot x_i \equiv I'_i + a_i \cdot x'_i \mod m_i \\ &\iff \quad a_i = (I'_i - I_i) \cdot (x_i - x'_i)^{-1} \mod m_i. \end{aligned}$$

□

The efficiency of the proposed VSS scheme is analyzed in Section 5.4.

5.3. Joint Random Secret Sharing

Joint random secret sharing (JRSS) protocols enable a group of users to jointly generate and share a random secret where a dealer is not available. In this section, we proposed a CRT-based JRSS by adapting the JRSS scheme given by Kaya and Selçuk [12]. We modify the commitment with respect to our VSS and also use a modified version of the original scheme;

$$M_{(r)} > np^2 M^{(r-1)}, \tag{5.2}$$

$$M = \left\lfloor \frac{M_{(r)}}{n} \right\rfloor \tag{5.3}$$

where M denotes the domain of y, i.e. $y \in \mathbb{Z}_M$. The CRT based JRSS scheme is given in Figure 5.5. Since, we are using a commitment requiring a random value, our scheme has an additional, initialization, phase for constructing local and global randoms between participants.

Theorem 5.10. *In the modified Asmuth-Bloom scheme with* (5.2) *and* (5.3)*, no possible secret value can be ruled out for an adversary, and the JRSS is a statistical secret sharing scheme.*

Proof. Let B be the set of $r-1$ users corrupted by the adversary. Let $\mathcal{X}$ be the probability distribution $\Pr(S = \delta)$ over the secret candidates $\delta \in \mathbb{Z}_p$ from the adversary's point of view. The adversary can compute $y' = y \bmod M_B$ and $x' = x \bmod M_B$. Due to (5.2) and (5.3), $M/M_B > p^2$. The rest of the proof is similar to that of Theorems 3.5 and 5.7. □

Initialization Phase

P_i chooses random $x_j^{(i)} \in \mathbb{Z}_{m_i}$ for $j = 1, \ldots, n$ and shares $x_j^{(i)}$ with P_j $(\forall i)$.

P_i calculates his own random as $x^{(i)} \equiv x_i^{(j)} \mod m_j$ by CRT $(\forall i)$.

Distribution Phase

P_i chooses a secret $S_i \in \mathbb{Z}_p$ and shares it using the VSS scheme as follows $(\forall i)$:

- He first chooses an arbitrary A_i such that $y^{(i)} = S_i + A_i \cdot p < \lfloor M_{(r)}/n \rfloor = M$
- Then the share for the P_j is computed as $y_j^{(i)} = y^{(i)} \mod m_j$ and privately send to P_j
- Then announces $E(y^{(i)}, x^{(i)})$ and $Range\_Proof(E(y^{(i)}, x^{(i)}), M)$.

Verification Phase

Verification of the shares of each user can be done by the same procedure as the VSS in Figure 5.4.

Let $\mathcal{B}$ be the set of users whose shares are verified correctly.

P_i computes his overall share and x_i as $(\forall i)$

$$I_i = \left(\sum_{j \in \mathcal{B}} y_i^{(j)} \right) \bmod m_i, \quad x_i = \left(\sum_{j \in \mathcal{B}} x_j^{(i)} \right) \bmod m_i.$$

Reconstruction Phase

Let G be a group of participants gathered to reconstruct the secret,

Share of $P_i \in G$ is verified by other participants:

$$g_i^{I_i} \cdot h_i^{x_i} \stackrel{?}{\equiv} \left(\prod_{j \in \mathcal{B}} E(y^{(j)}, x^{(j)}) \right) \pmod{q_i}. \tag{5.4}$$

After all the shares are verified, G may reconstruct S using CRT.

Figure 5.5. Our Proposed JRSS Scheme.

Correctness. Observe that when all users behave honestly, the JRSS scheme works correctly. Let $y = \sum_{i\in\mathcal{B}} y^{(i)}$. It is easy to see that $y < M_{(r)}$, since $y^{(i)} < M$ for all $i \in \mathcal{B}$, where $|\mathcal{B}| \leq n$ and $M = \lfloor M_{(r)}/n \rfloor$. One can see that $I_j = y \bmod m_j$ for all $j \in \mathcal{B}$ by checking

$$\begin{aligned} y \bmod m_j &= \left(\sum_{i\in\mathcal{B}} y^{(i)}\right) \bmod m_j \\ &= \left(\sum_{i\in\mathcal{B}} I_j^{(i)}\right) \bmod m_j \\ &= I_j \bmod m_j = I_j. \end{aligned}$$

Hence, each I_i satisfies $I_i = y \bmod m_i$ and $y < M_{(r)}$; y can be constructed with t shares.

For correctness of the verification procedure in (5.4), one can observe that

$$\begin{aligned} \left(\prod_{j\in\mathcal{B}} E(y^{(j)}, x^{(j)})\right) \bmod q_i &= g^{\sum_{j\in\mathcal{B}} y^{(j)}} \cdot h^{\sum_{j\in\mathcal{B}} x^{(j)}} \bmod q_i \\ &= g_i^{\sum_{j\in\mathcal{B}} y^{(j)}} \cdot h_i^{\sum_{j\in\mathcal{B}} x^{(j)}} \bmod q_i \\ &= {g_i}^{I_i} {h_i}^{x_i} \bmod q_i \end{aligned}$$

where $x_i = \left(\sum_{j\in\mathcal{B}} x_j^{(i)}\right) \bmod m_i$. Hence, when every user behaves honestly, the proposed JRSS scheme works correctly. The privacy of the secret shared by the JRSS follows from Theorem 5.10 and the privacy of the modified Asmuth-Bloom scheme.

The consistency and the commitment correctness of the JRSS follows from that of the underlying VSS scheme: If any participant tries to deal inconsistent shares in the sharing phase or tries to provide false shares in the reconstruction phase, this will be detected by the VSS as shown in Theorems 5.8 and 5.9. The practicality of the scheme is analyzed in Section 5.4.

5.4. Practicality and Efficiency of the Schemes

In this section, we analyze the practicality and efficiency of our proposed VSS and JRSS schemes.

If both q and $2q+1$ are prime numbers, q is called a Sophie Germain prime. It is believed that the number of Sophie Germain primes is infinite and due to the conjecture of Hardy and Littlewood [56], for sufficiently large N, the number of Sophie Germain primes less than N is

$$2C\int_2^N \frac{dx}{\log x \log 2x} \approx \frac{2CN}{(\ln N)^2} \tag{5.5}$$

where $C \approx 0.66$ is the twin prime constant. The accuracy of the conjecture and the ratio is in Table 5.1 [57]. The second column refers to the actual number of Sophie Germain primes less than N. The third and fourth columns are the integral and ratio approximations on the left and right side of (5.5), respectively.

Table 5.1. Number of Sophie Germain primes less than N.

N	Actual	Integral	Ratio
1,000,000	7,746	7,811	6,917
10,000,000	56,032	56,128	50,822
100,000,000	423,140	423,295	389,107
1,000,000,000	3,308,859	3,307,888	3,074,425
10,000,000,000	26,569,515	26,568,824	24,902,848
100,000,000,000	218,116,524	218,116,102	205,808,662

For the proposed VSS, a sequence $m_1 < m_2 < \cdots < m_n$ consisting of n Sophie Germain primes is needed. Also, for security issues, this sequence must also satisfy inequality (3.2). Let us assume that p, the number of secret candidates, is a k-bit prime. From (3.2), first, each m_i must be at least a $2k$-bit Sophie Germain prime. We know that such primes exist since the number of Sophie Germain primes is infinite. Second, we need to know that we can find a Sophie Germain sequence for every r, n,

and k such that the product of the r smallest numbers in the sequence is larger than the product of the $r-1$ largest ones and p^2. Note that the Hardy-Littlewood conjecture says that the density of the Sophie Germain primes less than N is proportional to $1/(\ln N)^2$, where the prime number theorem says that the density of primes less than N is proportional to $1/(\ln N)$. Hence, considering $N \gg \ln N$, finding an Asmuth-Bloom sequence with Sophie Germain primes satisfying (3.2) should not be much harder than finding such a sequence with ordinary primes.

An analysis of the existence of a desired sequence and the information rate of the proposed schemes can be given as follows: Let p be a k-bit prime. Provided that $2^k \gg n$, the number of $2k$-bit Sophie Germain primes is approximately equal to

$$\frac{2C2^{2k+1}}{(\ln 2^{2k+1})^2} - \frac{2C2^{2k}}{(\ln 2^{2k})^2} = \frac{C2^{2k+1}}{(\ln 2)^2}\left(\frac{2}{(2k+1)^2} - \frac{1}{(2k)^2}\right)$$

which is much greater than n. Let m_1 be a $2k$-bit Sophie Germain prime and $\ell = \ln m_1$. Let m_i be the $(i-1)$st Sophie Germain prime after m_1. Due to (5.5), we can assume that $m_i \approx m_1 + (i-1)\ell^2$. Note that the ratio m_i/m_j for $i < j$ is bounded above by $(1 + n\ell^2/m_1)$. Hence, the inequality

$$m_1 > \frac{p^2 \prod_{i=1}^{r-1} m_{n-i+1}}{\prod_{i=1}^{r-1} m_{i+1}}$$

is satisfied when

$$m_1 > p^2 \left(1 + \frac{n\ell^2}{m_1}\right)^{r-1}.$$

Since $m_1 \gg n\ell^2$ and $m_1 \gg t$, we can choose $m_1 \approx p^2$, and the information rate of the VSS scheme becomes $|p|/|m_n| \approx |p|/|p^2 + 4n(\ln p)^2| \approx 1/2$. A similar analysis can be done for the JRSS scheme as well: (3.2) is replaced by (5.2); hence,

$$m_1 > n\, p^2 \left(1 + \frac{n\ell^2}{m_1}\right)^{r-1}.$$

So the information rate is again

$$\frac{|p|}{|n\,p^2 + 4n(\ln p)^2|} \approx \frac{1}{2},$$

respectively. Although the proposed scheme is not ideal, they are highly practical since the information rates is only 1/2.

6. CONCLUSION

In this thesis, first of all, we summarized some of the well-known SSS with their security and efficiency concerns. Then, we proposed different modifications for homomorphism as well as verifiability properties.

We presented three modified versions of Asmuth-Bloom SSS with homomorphic properties. All three versions have some advantages and also disadvantages with respect to the security level and computational cost and additional properties like homomorphism. Obviously, there is a trade-off between security level and computational cost. It is up to users to choose the most convenient one for their systems.

Secondly, we pointed out certain security concerns for three verifiable secret sharing schemes based on the Chinese Remainder Theorem in the literature. We first show that two of the schemes are insecure and the remaining one is only secure against a computationally bounded adversary. We propose a modification for this scheme and prove that the modified scheme is a secure verifiable secret sharing scheme against an unbounded adversary. Lastly, as an application, we show how to use the new scheme for joint random secret sharing.

APPENDIX A: RANGE PROOF

As we mention in Section 5.2, CRT-based VSS schemes require a commitment with range proof. *Range proof* techniques prove that a committed number belong to the given interval. If it is *zero-knowledge* commitment, as in [55], it is guaranteed that no information about the committed number is leaked. Whereas [12] uses the range proof of [54], we use the one presented by Boudot in [55] for our proposed VSS scheme.

A.1. Boudot's Range Proof

Boudot [55] proposed an efficient and non-interactive technique to prove that a committed number lies within an interval. He used the Fujisaki-Okamoto integer commitment scheme [58], where the commitment of an integer y is:

$$D = D(y, x) = g_N^y h_N^x \bmod N,$$

where g_N is an element of high order in $\mathbb{Z}_N^*$, h_N is an element of the group generated by g_N, x is a random integer, and N is an RSA composite whose factorization is unknown. As proved in [55, 58], this commitment scheme is statistically hiding and computationally binding assuming that the prime factorization of N is unknown. That is the committer cannot find another valid proof unless he is computationally unbounded, and the receiver of the commitment cannot distinguish the discrete logarithm, i.e., y, from a random value.

The commitment scheme we use, however, is slightly different: Let $Q = \prod_{i=1}^{n} q_i$ be a composite number. The commitment to a value, y, is

$$E = E(y, x) = g^y h^x \bmod Q,$$

where g is an element in $\mathbb{Z}_Q^*$, h is an element of the group generated by g. In [55] the author shows how to reduce a range proof for the commitment E to a range proof

for the commitment D by a zero-knowledge proof of equality of committed values (see section 3.2 and appendix A of [55]).

REFERENCES

1. Rivest, R. L., L. Adleman and M. L. Dertouzos, "On Data Banks and Privacy Homomorphisms", *Foundations of Secure Computation*, Vol. 4, No. 11, pp. 169–180, 1978.

2. Rivest, R. L., A. Shamir and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems", *Communications of the ACM*, Vol. 21, No. 2, pp. 120–126, 1978.

3. Paillier, P., "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes", *Advances in Cryptology - EUROCRYPT'99*, pp. 223–238, 1999.

4. ElGamal, T., "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms", *Advances in Cryptology*, pp. 10–18, 1985.

5. Gentry, C., "Fully Homomorphic Encryption Using Ideal Lattices", *Proceedings of the 41st Annual ACM Symposium on Theory of Computing-STOC'09*, pp. 169–169, 2009.

6. Shamir, A., "How to Share a Secret", *Communications of the ACM*, Vol. 22, No. 11, pp. 612–613, 1979.

7. Blakley, G. R., "Safeguarding Cryptographic Keys", *Proceedings of the National Computer Conference*, Vol. 48, pp. 313–317, 1979.

8. Benaloh, J. C., "Secret Sharing Homomorphisms: Keeping Shares of a Secret Secret", *Advances in Cryptology - CRYPTO'86*, pp. 251–260, 1987.

9. Asmuth, C. and J. Bloom, "A Modular Approach to Key Safeguarding", *IEEE Transactions on Information Theory*, Vol. 30, No. 2, pp. 208–210, 1983.

10. Mignotte, M., "How to Share a Secret", *Cryptography*, pp. 371–375, 1983.

11. Goldreich, O., D. Ron and M. Sudan, "Chinese Remaindering with Errors", *Proceedings of the Thirty-first Annual ACM Symposium on Theory of Computing*, pp. 225–234, 1999.

12. Kaya, K. and A. A. Selçuk, "A Verifiable Secret Sharing Scheme Based on the Chinese Remainder Theorem", *Progress in Cryptology - INDOCRYPT 2008*, Vol. 5365 of *Lecture Notes in Computer Science*, pp. 414–425, 2008.

13. Qiong, L., W. Zhifang, N. Xiamu and S. Shenghe, "A Non-Interactive Modular Verifiable Secret Sharing Scheme", *Proceedings of International Conference on Communications, Circuits and Systems*, Vol. 1, pp. 84–87, IEEE, 2005.

14. Iftene, S., "Secret Sharing Schemes with Applications in Security Protocols", *Scientific Annals Cuza University*, Vol. 16, pp. 63–96, 2006.

15. Harn, L., M. Fuyou and C.-C. Chang, "Verifiable Secret Sharing Based on the Chinese Remainder Theorem", *Security and Communication Networks*, Vol. 7, No. 6, pp. 950–957, 2014.

16. Liu, Y., L. Harn and C.-C. Chang, "A Novel Verifiable Secret Sharing Mechanism Using Theory of Numbers and a Method for Sharing Secrets", *International Journal of Communication Systems*, Vol. 28, No. 7, pp. 1282–1292, 2015.

17. Brickell, E. F. and Y. Yacobi, "On Privacy Homomorphisms", *Advances in Cryptology - EUROCRYPT'87*, pp. 117–125, 1988.

18. Goldwasser, S. and S. Micali, "Probabilistic Encryption & Eow to Play Mental Poker Keeping Secret All Partial Information", *Proceedings of the Fourteenth Annual ACM Symposium on Theory of Computing*, pp. 365–377, 1982.

19. Boneh, D., E.-J. Goh and K. Nissim, "Evaluating 2-DNF Formulas on Cipher-

texts", *Theory of Cryptography*, pp. 325–341, 2005.

20. Beimel, A., "Secret-Sharing Schemes: A Survey", *Coding and Cryptology*, Vol. 6639, pp. 11–46, 2011.

21. Bertilsson, M. and I. Ingemarsson, "A Construction of Practical Secret Sharing Schemes Using Linear Block Codes", *Advances in Cryptology - AUSCRYPT'92*, pp. 67–79, 1993.

22. Simmons, G. J., "How to (Really) Share a Secret", *Proceedings on Advances in Cryptology*, pp. 390–448, 1990.

23. Ghodosi, H., J. Pieprzyk and R. Safavi-Naini, "Secret Sharing in Multilevel and Compartmented Groups", *Information Security and Privacy*, pp. 367–378, 1998.

24. Beutelspacher, A., "How to Say "No"", *Advances in Cryptology - EUROCRYPT'89*, pp. 491–496, 1990.

25. Blundo, C., A. De Santis, L. Gargano and U. Vaccaro, "Secret Sharing Schemes with Veto Capabilities", *Algebraic Coding*, pp. 82–89, 1994.

26. Ito, M., A. Saito and T. Nishizeki, "Secret Sharing Scheme Realizing General Access Structure", *Electronics and Communications in Japan (Part III: Fundamental Electronic Science)*, Vol. 72, No. 9, pp. 56–64, 1989.

27. Benaloh, J. and J. Leichter, "Generalized Secret Sharing and Monotone Functions", *Proceedings on Advances in Cryptology*, pp. 27–35, 1990.

28. Quisquater, M., B. Preneel and J. Vandewalle, "On the Security of the Threshold Scheme Based on the Chinese Remainder Theorem", *Public Key Cryptography*, pp. 199–210, 2002.

29. Kaya, K. and A. A. Selçuk, "Threshold Cryptography Based on Asmuth-Bloom Secret Sharing", *Information Sciences*, Vol. 177, No. 19, pp. 4148–4160, 2007.

30. Brickell, E. F., "Some Ideal Secret Sharing Schemes", *Advances in Cryptology - EUROCRYPT'89*, pp. 468–475, 1990.

31. Karnin, E. D., J. Greene and M. E. Hellman, "On Secret Sharing Systems", *IEEE Transactions on Information Theory*, Vol. 29, No. 1, pp. 35–41, 1983.

32. Capocelli, R. M., A. De Santis, L. Gargano and U. Vaccaro, "On the Size of Shares for Secret Sharing Schemes", *Journal of Cryptology*, Vol. 6, No. 3, pp. 157–167, 1993.

33. Csirmaz, L., "The Size of a Share Must Be Large", *Journal of Cryptology*, Vol. 10, No. 4, pp. 223–231, 1997.

34. van Dijk, M., "A Linear Construction of Perfect Secret Sharing Schemes", *Advances in Cryptology - EUROCRYPT'94*, pp. 23–34, 1995.

35. Dolev, S., L. Lahiani and M. Yung, "Secret Swarm Unit Reactive k- Secret Sharing", *Progress in Cryptology–INDOCRYPT 2007*, pp. 123–137, 2007.

36. Blundo, C., A. Cresti, A. De Santis and U. Vaccaro, "Fully Dynamic Secret Sharing Schemes", *Advances in Cryptology - CRYPTO'93*, pp. 110–125, 1994.

37. Chor, B., S. Goldwasser, S. Micali and B. Awerbuch, "Verifiable Secret Sharing and Achieving Simultaneity in the Presence of Faults", *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pp. 383–395, 1985.

38. Pedersen, T. P., "A Threshold Cryptosystem Without a Trusted Party", *Advances in Cryptology - EUROCRYPT'91*, pp. 522–526, 1991.

39. Feldman, P., "A Practical Scheme for Non-Interactive Verifiable Secret Sharing", *Foundations of Computer Science, 1987., 28th Annual Symposium on*, pp. 427–438, IEEE, 1987.

40. Gennaro, R., S. Jarecki, H. Krawczyk and T. Rabin, "Robust Threshold DSS

Signatures", *Information and Computation*, Vol. 164, No. 1, pp. 54–84, 2001.

41. Ingemarsson, I. and G. J. Simmons, "A Protocol to Set up Shared Secret Schemes Without the Assistance of a Mutually Trusted Party", *Proceedings of EUROCRYPT'90*, Vol. 547 of *LNCS*, pp. 266–282, 1991.

42. Herzberg, A., S. Jarecki, H. Krawczyk and M. Yung, "Proactive Secret Sharing or: How to Cope with Perpetual Leakage", *Advances in Cryptology - CRYPT0'95*, pp. 339–352, 1995.

43. McEliece, R. J. and D. V. Sarwate, "On Sharing Secrets and Reed-Solomon Codes", *Communications of the ACM*, Vol. 24, No. 9, pp. 583–584, 1981.

44. Bogdanov, D., S. Laur and J. Willemson, "Sharemind: A Framework for Fast Privacy-Preserving Computations", *Computer Security-ESORICS 2008*, pp. 192–206, 2008.

45. Du, W. and M. J. Atallah, "Protocols for Secure Remote Database Access with Approximate Matching", *E-Commerce Security and Privacy*, pp. 87–111, 2001.

46. Ben-Or, M., S. Goldwasser and A. Wigderson, "Completeness Theorems for Non-Cryptographic Fault-Tolerant Distributed Computation", *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, pp. 1–10, 1988.

47. Chaum, D., C. Crépeau and I. Damgard, "Multiparty Unconditionally Secure Protocols", *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, pp. 11–19, 1988.

48. Gennaro, R., M. O. Rabin and T. Rabin, "Simplified VSS and Fast-Track Multiparty Computations with Applications to Threshold Cryptography", *Proceedings of the Seventeenth Annual ACM Symposium on Principles of Distributed Computing*, pp. 101–111, 1998.

49. Bogetoft, P., D. L. Christensen, I. Damgård, M. Geisler, T. Jakobsen, M. Krøigaard, J. D. Nielsen, J. B. Nielsen, K. Nielsen, J. Pagter *et al.*, "Secure multiparty Computation Goes Live", *Financial Cryptography and Data Security*, pp. 325–343, 2009.

50. Iftene, S., "General Secret Sharing Based on the Chinese Remainder Theorem with Applications in E-voting", *Electronic Notes in Theoretical Computer Science*, Vol. 186, pp. 67–84, 2007.

51. Dragan, C. C. and F. L. Tiplea, "On the Asymptotic Idealness of the Asmuth-Bloom Threshold Secret Sharing Scheme", *Designs, Codes and Cryptography, submitted for publication*, 2013.

52. Bozkurt, I. N., *Function and Secret Sharing Extensions for Blakley and Asmuth-Bloom Secret Sharing Schemes*, Ph.D. Thesis, Bilkent University, 2009.

53. Ersoy, O., T. B. Pedersen, K. Kaya, A. A. Selçuk and E. Anarım, "A CRT-Based Verifiable Secret Sharing Scheme Secure against Unbounded Adversaries", *Submitted*, 2015.

54. Cao, Z. and L. Liu, "Boudot's Range-Bounded Commitment Scheme Revisited", *Information and Communications Security*, pp. 230–238, 2007.

55. Boudot, F., "Efficient Proofs That a Committed Number Lies in an Interval", *Advances in Cryptology - EUROCRYPT 2000*, pp. 431–444, 2000.

56. Hardy, G. H. and J. E. Littlewood, "Some Problems of 'Partitio numerorum'; III: On the Expression of a Number As a Sum of Primes", *Acta Mathematica*, Vol. 44, No. 1, pp. 1–70, 1923.

57. Caldwell, C. K., "An Amazing Prime Heuristic", *Technical Report, University of Tennessee at Martin*, 2000.

58. Fujisaki, E. and T. Okamoto, "Statistical Zero Knowledge Protocols to Prove Modular Polynomial Relations", *Advances in Cryptology - CRYPTO'97*, pp. 16–30, 1997.