An Efficient Lattice Based Multi-Stage Secret Sharing Scheme

Hossein Pilaram and Taraneh Eghlidos

Abstract—In this paper, we construct a lattice based (t, n) threshold multi-stage secret sharing (MSSS) scheme according to Ajtai's construction for one-way functions. In an MSSS scheme, the authorized subsets of participants can recover a subset of secrets at each stage while other secrets remain undisclosed. In this paper, each secret is a vector from a t-dimensional lattice and the basis of each lattice is kept private. A t-subset of n participants can recover the secret(s) using their assigned shares. Using a lattice based one-way function, even after some secrets are revealed, the computational security of the unrecovered secrets is provided against quantum computers. The scheme is multi-use in the sense that to share a new set of secrets, it is sufficient to renew some public information such that a new share distribution is no longer required. Furthermore, the scheme is verifiable meaning that the participants can verify the shares received from the dealer and the recovered secrets from the combiner, using public information.

Index Terms—Multi-stage secret sharing, lattice based cryptography, multi-use secret sharing, verifiability

1 Introduction

Secret sharing schemes (SSS) are used as a tool in many cryptographic protocols including revocable electronic cash [1], electronic voting [2], cloud computing [3] and key management in sensor networks [4]. A secret sharing scheme allows one to share a secret s among a set $\mathcal{P}$ of parties, called participants. The participants are assigned different values called shares and only certain authorized subsets of them can recover the secret using these shares. The collection of the authorized subsets of participants is called the access structure and denoted by Γ. In an SSS, we intend that any subset in Γ can reconstruct the secret, while those not in Γ cannot recover any information about the secret.

A (t, n) threshold secret sharing scheme was introduced by Blakley and Shamir independently in 1979 [5], [6]. In such a scheme, the access structure consists of all subsets of $\mathcal{P}$ including at least t participants. Many other secret sharing schemes have been proposed since then [7], [8]. Some new features have been added to secret sharing schemes such as verifiability of the shares [1], [9], resistance of the scheme in the presence of a number of cheaters [10], [11] and dynamic change of the threshold and/or the number of participants [12], [13]. However, these schemes only work with a single secret and once the secret is changed to a new one, the system has to update the shares and resend the new shares to the participants. This consumes additional resources and might make the system impractical.

A multi-secret sharing scheme is a generalization of a secret sharing scheme, where there is more than one secret to be shared [14]. However, each participant receives one share at the beginning of the secret sharing process, the size of which is the same as the size of the secrets. These schemes only provide computational security [15]. In 1994, He and Dawson [16] proposed a multi-stage (t, n) threshold secret sharing scheme. In 2007, Geng et al. [17] showed that the He-Dawson scheme is actually of one-time-use and vulnerable to collusion attacks. They proposed a multi-use threshold secret sharing scheme using a one-way hash function. The term "multi-use" means that it is not required to redistribute the fresh shares over a secret channel to the participants, when a new set of secrets is to be shared. In 2006, Pang et al. [18] proposed a multi-secret sharing scheme for general access structure in which all of the secrets are revealed at the same time, i.e., when an authorized subset of participants pull their shares together, they recover all of the secrets simultaneously in a single stage. A multi-secret sharing scheme will be called multi-stage if in recovering a number of secrets, the reconstructed secrets do not leak any information about the unrecovered secrets. For this purpose, two security requirements are needed:

1) The shares must be masked during secret reconstruction phase.
2) Recovery of a secret must not jeopardize the secrecy of the other unrecovered secrets.

For a multi-secret sharing scheme to be multi-stage, the participants must provide the combiner with pseudo-secret shares depending on the original shares. All existing MSSS schemes are based on one-way (hash) functions [15], [19], [20], two-variable one-way functions [21], [22] and assumptions such as hardness of solving discrete logarithm problem [23] which can now be tackled by quantum computers.

Advances in quantum computers threaten the security of currently used public-key cryptographic algorithms, which is based on the difficulty of integer factorization and discrete logarithm problems. The introduction of quantum algorithms for factoring and computation of discrete logarithms by Shor in 1994 [24], has changed the

- *H. Pilaram is with the Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran. E-mail: h_pilaram@ee.sharif.edu.*
- *T. Eghlidos is with the Electronics Research Institute, Sharif University of Technology, Tehran, Iran. E-mail: teghlidos@sharif.edu.*

Manuscript received 24 Nov. 2014; revised 18 Apr. 2015; accepted 24 Apr. 2015. Date of publication 20 May 2015; date of current version 18 Jan. 2017. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TDSC.2015.2432800

research trends from classical to post-quantum cryptography. In fact, the first post-quantum cryptographic system is the public key encryption scheme proposed by McEliece in 1978. It is based on hardness of coding problems, which is the beginning of code-based cryptography [25]. Not even quantum attacks have yet been known to represent a serious threat on the McEliece cryptosystem. From the efficiency point of view, no practical application of code-based cryptography is known because of the large size of the public key (100 kilobytes to several megabytes) [26].

Lattice based cryptographic constructions play a great role for post-quantum cryptography, because of its efficient linear computations. Furthermore, they enjoy provable security based on worst-case hardness of lattice problems. In addition, since no quantum algorithm has yet been proposed for solving lattice problems, lattice based cryptography is supposed to be resistant to quantum computers [26].

The first lattice based cryptographic algorithm was introduced by Ajtai in 1996 [27]. He proposed a construction of a family of one-way functions whose security is equivalent to the worst-case hardness of n^c-approximate of lattice problems, where n is the dimension of the lattice space and c is a positive constant. Goldreich et al. [28] proved that the Ajtai's function is collision resistant which is much stronger property than one-wayness. Some lattice based public key encryption schemes like GGH [29] and NTRU [30] have been introduced in literature and enjoy provable security based on hardness of lattice problems in worst case.

The design of an SSS using lattice based cryptography is a very recent topic. In 2011, Georgescu [31] proposed an (n, n) secret sharing scheme whose security can be reduced to the hardness of the learning with errors (LWE) problem. This scheme offers the possibility for the participants to check if all the shares distributed by the dealer are valid. In 2012, Bansarkhani and Meziani [32] proposed an (n, n) threshold verifiable secret sharing scheme based on lattices and the usage of linear lattice based hash functions to enable each participant to verify their share as well as the recovered secret. The security of this scheme relies on the hardness of n^c-approximate shortest vector problem (SVP). This scheme uses efficient matrix vector operations to verify the shares instead of exponentiation used in conventional schemes. To the best of our knowledge, Amini et al. [33] and Asaad et al. [34] proposed the first (t, n) threshold secret sharing schemes with asymptotic security, in 2014. To recover the secret, the participants use Babai's nearest plane algorithm [35] to solve the closest vector problem (CVP) in general lattices. Bendlin et al. have proposed two lattice based threshold cryptographic schemes, one in threshold decryption [36] and the other in sharing a lattice trapdoor [37], using Shamir's threshold secret sharing scheme.

In this paper, we propose an MSSS scheme, in which the participants are each given one share to recover the secrets, in such a way that an adversary cannot recover the unreconstructed secrets in polynomial time using the revealed secrets. In the proposed scheme, lattice based one-way functions are applied to the original shares to obtain the corresponding pseudo-secret shares. Then, they are sent to the combiner for recovering the desired secret(s). Hence, the combiner cannot misuse the pseudo-secret shares to obtain the original shares and disclose the unrecovered secrets. The security of the proposed scheme is based on the hardness of lattice problems which are resistant to the quantum algorithms. Furthermore, the scheme enjoys significant features such as being multi-stage, multi-use and verifiable, and hence is favorable in many applications. Moreover, the scheme inherits its efficiency from simple matrix operations used in the secret sharing protocol, especially in the participants' side, and hence is suitable even if the participants have limited processing capabilities.

The paper is organized as follows: Section 2 provides a brief review of lattices, lattice based cryptography and secret sharing schemes. Section 3 is dedicated to the proposed verifiable MSSS scheme including the security requirements and the algorithm. The security and efficiency of the proposed scheme are respectively discussed in Section 4 and Section 5. Section 6 concludes the paper.

2 Preliminaries

In this section, we introduce some basic concepts of lattice, lattice based cryptography, and secret sharing schemes needed later.

2.1 Notations

In this paper, vectors are assumed to be in column form. Lowercase and uppercase letters denote vectors and matrices, respectively. The transpose and pseudo-inverse of a rectangular matrix are denoted by $(\cdot)^T$ and $(\cdot)^\dagger$, respectively. The matrix I_n refers to the $n \times n$ identity matrix and the matrix $\mathbf{0}_{m\times n}$ represents the zero matrix of size $m \times n$. Also, $\mathbb{R}$ will denote the set of reals, $\mathbb{Z}$ the set of integers and $\mathbb{Z}_q$ the finite field modulo q. If $\mathbb{S}$ is a set of numbers, $\mathbb{S}^n$ will denote the set of vectors of size n, and $\mathbb{S}^{m\times n}$ the set of $m \times n$ matrices, whose entries are chosen from $\mathbb{S}$. The sign $\|\cdot\|$, used in this paper, denote an arbitrary norm. The most important class of norms are the ℓ_p norms, defined for any $p \geq 1$ and a vector $x \in \mathbb{R}^n$ as $\|x\|_p = \left(\sum_{i=1}^n |x_i|^p\right)^{1/p}$. The operation $\oplus$ denotes the component-wise XOR operation.

We use the standard big-O and little-o notations to classify the growth of functions. A negligible function, denoted by $\mathrm{negl}(n)$, is such that $f(n) = o(n^{-c})$ for every fixed constant c. We say that a probability is overwhelming if it is equal to $1 - \mathrm{negl}(n)$.

2.2 Lattices

In this paper, we use the concept of lattice as a regular array of points in m-dimensional real vector space.

Definition 1. *[38] Let $b_1, b_2, \ldots, b_n$ be n linearly independent vectors in vector space $\mathbb{R}^m$. $L(b_1, \ldots, b_n)$ is defined to be the set of all integer linear combinations of $b_1, b_2, \ldots, b_n$ as follows:*

$$\Lambda = L(b_1, \ldots, b_n) = \left\{ \sum_{i=1}^{n} x_i b_i : x_i \in \mathbb{Z} \right\} \tag{1}$$

The set of vectors $\{b_1, \ldots, b_n\}$ is called a basis for the lattice Λ, and n is called the rank of the lattice.

Lattice based cryptographic constructions are based on the presumed hardness of lattice problems, the most

famous of which are SVP and CVP [38]. In lattice based cryptography, we usually consider the approximate version of these problems, denoted by an approximation factor γ. For instance, in γ-approximate SVP, we want to find a vector in the lattice whose norm is within the factor γ of the optimum solution, or in γ-approximate CVP, we search for a vector in the lattice whose distance from the target vector is at most γ times that of the closest vector.

Definition 2. *[26] q-ary Lattices are lattices Λ satisfying $q\mathbb{Z}^n \subseteq \Lambda \subseteq \mathbb{Z}^n$ for some (possibly prime) integer q.*

For instance, given an integer matrix $A \in \mathbb{Z}_q^{n\times m}$ and modulus q, the set of vectors $x \in \mathbb{Z}^m$ that satisfy the equation $Ax = 0 \bmod q$, forms a lattice of dimension m, which is closed under congruence modulo q. This lattice is denoted by $\Lambda_q^{\perp}(A)$.

2.3 Ajtai's Reduction

In [27], Ajtai introduced the one-way function $f_A(x) = Ax \bmod q$, where $A \in \mathbb{Z}_q^{n\times m}$ and $x \in \{0,1\}^m$. Inverting this function results in the following problem:

- *Parameters*: n, m, $q \in \mathbb{N}$ such that $m > n \log q$, and $q = O(n^c)$ for some constant c.
- *Input*: A uniformly random matrix $A \in \mathbb{Z}_q^{n\times m}$ and a vector $y = Ax$ for some random vector $x \in \{0,1\}^m$.
- *Output*: A vector $x \in \{0,1\}^m$ such that $Ax = y \bmod q$.

Ajtai proved that solving this problem with non-negligible probability implies the ability to solve any instance of n^c-approximate SVP, which cannot currently be solved in polynomial time by quantum computers.

This problem is a special case of the inhomogeneous small integer solution (ISIS) problem, in which the condition $x \in \{0,1\}^m$ is replaced by $\|x\| \leq \beta$ for a real parameter β. Solving ISIS is equivalent to decoding an arbitrary integer target point $t \in \mathbb{Z}^m$ to within distance β on the q-ary lattice $\Lambda_q^{\perp}(A) = \{x \in \mathbb{Z}^m | Ax = 0 \bmod q\}$, where the syndrome of the target point is $u = At \bmod q$ [39].

2.4 Secret Sharing

A secret sharing scheme is a method of sharing a secret among a set of parties, called participants, denoted by $\mathcal{P}$. A trusted third party, named dealer, assigns a private value, called share, to each participant. Only the authorized subsets of participants can recover the secret by running a prespecified algorithm. The set of all authorized subsets is called an access structure. In general, an access structure is a subset of the power set of $\mathcal{P}$. A specific instance of general access structure is the threshold structure, which for a given t, consists of all subsets of at least t elements of the power set of $\mathcal{P}$. A (t,n) threshold secret sharing scheme is called perfect, if less than t participants can obtain no information about the secret. A secret sharing scheme is called ideal if the entropy of each share is equal to the entropy of the secret.

A secret sharing scheme usually consists of two phases:

- Share distribution: In this phase, the dealer computes the shares using a prespecified algorithm and sends them securely to the participants.
- Secret reconstruction: In this phase, the authorized subset of participants send their shares to a combiner to recover the secret by running the algorithm.

2.5 Shamir's (t,n) Threshold Secret Sharing

For constructing a threshold secret sharing scheme, Shamir uses the fact that for all sets of t distinct points in $\mathbb{R}^2$, there is a unique $(t-1)$-degree polynomial that passes through these points. In this scheme, the secret is chosen from $\mathbb{Z}_q$, where $q > n$ is a prime number. The dealer chooses a polynomial $Q(x) = a_0 + a_1 x + \cdots + a_{t-1}x^{t-1}$, where $a_0 = s$ is the secret and for $i = 1, \ldots, t-1$, the a_i's are chosen uniformly at random from $\mathbb{Z}_q$. The participant $P_j, j = 1, \ldots, n$ is assigned a label $x_j \in \mathbb{Z}_q$. The dealer computes the shares $s_j = Q(x_j)$'s $, j = 1, \ldots, n$, and sends them to the participants through a secure channel.

Any t out of n participants can recover the secret s solving the following linear equations:

$$\begin{bmatrix} 1 & x_{j_1} & \ldots & x_{j_1}^{t-2} & x_{j_1}^{t-1} \\ 1 & x_{j_2} & \ldots & x_{j_2}^{t-2} & x_{j_2}^{t-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & x_{j_t} & \ldots & x_{j_t}^{t-2} & x_{j_t}^{t-1} \end{bmatrix} \times \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{t-1} \end{bmatrix} = \begin{bmatrix} s_{j_1} \\ s_{j_2} \\ \vdots \\ s_{j_t} \end{bmatrix}, s = a_0. \quad (2)$$

The coefficient matrix in (2) is a square Vandermonde matrix, which is invertible, since the $x_j's$ are distinct.

Shamir's scheme has the following advantages [5]:

- It is perfect and ideal.
- The shares can easily be added or omitted, when some participants enter or leave, without changing the other shares.
- Without changing the secret, and only by choosing a new polynomial, the shares can be changed.
- By assigning multiple shares to each participant, a weighted threshold secret sharing scheme can be obtained.

However, Shamir's scheme has the following disadvantages, which makes it impossible to use it directly as a threshold secret sharing protocol:

- The dealer might cheat in the share distribution phase and send wrong shares to the participants. In such case, in the reconstruction phase, the secret cannot be recovered uniquely using different t-subsets of shares. Furthermore, the participants cannot verify the correctness of the used shares before recovering the secret.
- Cheating participants can send wrong shares to the combiner.

2.6 Extensions of Secret Sharing

Verifiable secret sharing. In a verifiable secret sharing scheme, the dealer must prove the correctness of the distributed shares to each party and the parties can verify the validity of the recovered secrets by the combiner [20].

Multi-secret sharing. In a multi-secret sharing scheme, more than one secret is to be shared among participants and it is desirable to give the participants only one share for recovering all the secrets [14].

Multi-stage secret sharing. Multi-stage secret sharing schemes are a special case of multi-secret sharing schemes in which the secrets can be recovered at different stages and the reconstructed secrets do not leak any information about the unrecovered secrets [19], [40].

Multi-use secret sharing. In a multi-use secret sharing scheme, when all of the secrets are recovered, for sharing a new set of secrets, the participants and the dealer can change the shares and the public information in such a way that there is no need to send the new shares through a secure channel between the dealer and the participants [19], [40].

3 The Scheme

3.1 Security Requirements

Here, we describe security requirements of our algorithm:

- *Threshold*. Every secret can only be recovered by any t or more participants who received the shares, and any subset of participants with less than t members cannot obtain any information about the secrets.
- *Shares*. In an MSSS scheme, shares are reused for recovering multiple secrets. Therefore, an adversary may try to obtain some information about the shares during secret recovery in earlier stages. We give the adversary an access to all revealed information sent by the participants to the combiner during the previous secret reconstruction phase(s). The scheme is secure, if the adversary cannot obtain the shares in polynomial time.
- *Secrets*. In an MSSS scheme, a reconstructed secret must not give any information about the unrecovered secrets. We give the adversary an access to all recovered secrets and corresponding pseudo-secret shares. The scheme is secure, if the adversary cannot obtain the unrecovered secrets in polynomial time.

3.2 The Algorithm

In this section, we propose a lattice based (t, n) threshold MSSS scheme, where t participants are required to recover each of the secrets. This scheme enables participants to recover any number of secrets independently and it is computationally difficult to use this information to obtain other secrets. In this scheme, we have m secrets $s_i \in \mathbb{Z}_q^t$, $i = 1, \ldots, m$, where q is a prime number and t is the threshold. It should be noted that all matrix operations are performed on $\mathbb{F}_q$.

The dealer randomly selects a vector $v \in \mathbb{Z}_q^t$, whose last entry is equal to 1 and publishes it. Then, for each secret s_i, he finds a private lattice basis B_i such that

$$s_i = B_i v, i = 1, \ldots, m, \tag{3}$$

where $B_i \in \mathbb{Z}_q^{t\times t}$ is a basis for a t-dimensional lattice. Therefore, the dealer must solve Equation (3) for the unknown B_i that leads to a system of t linear equations and t^2 unknowns. Since the number of unknowns is greater than the number of equations, Equation (3) does not have a unique solution. Therefore, the dealer must choose a solution. Here, we propose a method for accomplishing this task. First, the dealer chooses a random matrix $B_i' \in \mathbb{Z}_q^{t\times(t-1)}$ with linearly independent columns and sets $B_i = [\, B_i' \; b_i \,]$, where $b_i \in \mathbb{Z}_q^t$ is a column vector, and then solves (3) for unknown b_i as follows:

$$s_i = B_i v \Rightarrow s_i = [\, B_i' \; b_i \,] \begin{bmatrix} v' \\ 1 \end{bmatrix} \Rightarrow b_i = s_i - B_i' v' \tag{4}$$

where v' is the first $(t-1)$ entries of the vector v.

After computing the private lattice bases $B_i, i = 1, \ldots, m$, the dealer chooses n public vectors $\lambda_j \in \mathbb{Z}_q^t, j = 1, \ldots, n$, such that every t of these vectors are linearly independent. A class of these vectors are the columns of the transpose of an $n \times t$ Vandermonde matrix of rank t, or the right product of any $t \times t$ uniformly random invertible matrix and the Vandermonde matrix. Then the dealer must find public matrices $A_i \in \mathbb{Z}_q^{t\times r}, i = 1, \ldots, m$ and private vectors $c_j \in \{0,1\}^r, j = 1, \ldots, n$, such that the equality $A_i c_j = B_i \lambda_j$ holds, for $i = 1, \ldots, m$ and $j = 1, \ldots, n$, where $r \geq \max(t \log t, n)$. Hence, the dealer first randomly chooses n vectors c_j from $\{0,1\}^r$ so that the first n entries of these vectors form n linearly independent vectors on $\mathbb{Z}_q^n$. Then he solves the following system of linear equations to find the matrices A_i for each secret:

$$\begin{cases} A_i c_1 = B_i \lambda_1 \\ A_i c_2 = B_i \lambda_2 \\ \quad\vdots \\ A_i c_n = B_i \lambda_n \end{cases} \Rightarrow A_i[\, c_1 \ldots c_n \,] = B_i[\, \lambda_1 \ldots \lambda_n \,] \Rightarrow A_i C = B_i \Lambda$$

$$\Rightarrow [\, A_i' A_i'' \,] \begin{bmatrix} C' \\ C'' \end{bmatrix} = B_i \Lambda \Rightarrow A_i' = (B_i \Lambda - A_i'' C'') C'^{-1} \tag{5}$$

where $C = [\, c_1 \ldots c_n \,]$ and $\Lambda = [\, \lambda_1 \ldots \lambda_n \,]$. The invertible matrix C' contains the first n rows of C, the matrix C'' contains the last $(r-n)$ rows of C, and A_i' contains the first n columns of A_i. The dealer chooses A_i'' randomly from $\mathbb{Z}_q^{n\times(r-n)}$ and then computes A_i' from (5).

Now, the dealer prepares all the prerequisites for the verification of the shares by the participants. He chooses a random matrix $F \in \mathbb{Z}_q^{t\times r}$ and publishes it along with hash vectors of the shares, i.e. $h_j = Fc_j, j = 1, \ldots, n$. Furthermore, the dealer publishes $H(s_i), i = 1, \ldots, m$ for verification of the recovered secrets by the participants, where $H(\cdot)$ is a public hash function.

3.2.1 Distribution Phase

- The dealer distributes the share vector c_j to the participant P_j through a secure channel and publishes the matrices $A_i, i = 1, \ldots, m$ and the vectors $\lambda_j, j = 1, \ldots, n$.
- Upon receipt of the own share, the participant P_j verifies whether the hash value of his share is the same as that on the bulletin board, i.e., $Fc_j \stackrel{?}{=} h_j$.

3.2.2 Combination Phase

Here, different secrets are reconstructed independently from each other. Suppose that a subset $\{j_1, \ldots, j_t\} \subseteq \{1, \ldots, n\}$ of the participants intend to recover the secret $s_i, i \in \{1, \ldots, m\}$. For this purpose, the participant $j_l, l = 1, \ldots, t$ computes the

vector $d^i_{j_l} = A_i c_{j_l}, l = 1, \ldots, t$ as his pseudo-secret share and sends the result to the combiner in a secure manner. The combiner puts these vectors into the columns of a matrix D_i and then computes the private basis B_i as $B_i = D_i W^{-1}$, where $W = \big[\lambda_{j_1} \ldots \lambda_{j_t}\big]$. Then, the combiner recovers the secret s_i by computing $s_i = B_i v$. The participants can verify the recovered secret using the corresponding hash value, $H(s_i)$, published on the bulletin board.

Correctness.

Here, we show that the scheme recovers the private basis B_i and hence the secret s_i correctly.

$$\begin{aligned} D_i W^{-1} &= \big[d^i_{j_1}, \ldots, d^i_{j_t}\big] W^{-1} = \big[A_i c_{j_1}, \ldots, A_i c_{j_t}\big] W^{-1} \\ &= \big[B_i \lambda_{j_1}, \ldots, B_i \lambda_{j_t}\big] W^{-1} \\ &= B_i \big[\lambda_{j_1}, \ldots, \lambda_{j_t}\big] W^{-1} = B_i W W^{-1} = B_i. \end{aligned} \tag{6}$$

4 Security Analysis

The combination phase implies that any subset of participants of size t can recover each secret using their pseudo-secret shares. Furthermore, a subset of participants of size less than t cannot recover the secret s_k even by the knowledge of the already recovered secrets and the corresponding pseudo-secret shares.

In the following lemma, we prove that the matrices $A_i, i = 1, \ldots, m$, used in computing pseudo-secret shares, have a uniform distribution on $\mathbb{Z}_q^{t\times r}$, which is compatible with the definition of Ajtai's function.

Lemma 1. *Assume that q is a prime number. The matrices $A_i, i = 1, \ldots, m$ obtained by the Equation (5), have a uniform distribution on $\mathbb{Z}_q^{t\times r}$.*

Proof. Let X and Y be two independent random variables. We show that if X has a uniform distribution on $\mathbb{Z}_q$ and Y has an arbitrary distribution on $\mathbb{Z}_q$, then both $Z_1 = X + Y \bmod q$ and $Z_2 = X \cdot Y \bmod q$ have a uniform distribution on $\mathbb{Z}_q$. In the latter case, Y must be chosen from $\mathbb{Z}_q - \{0\}$.

$$\begin{aligned} Pr\{Z_1 = z_1\} &= Pr\{X + Y = z_1\} = \sum_y Pr\{X = z_1 - y\}.Pr\{Y = y\} \\ &= 1/q \sum_y Pr\{Y = y\} = 1/q \\ Pr\{Z_2 = z_2\} &= Pr\{X \cdot Y = z_2\} = \sum_y Pr\{X = z_2 \cdot y^{-1}\}.Pr\{Y = y\} \\ &= 1/q \sum_y Pr\{Y = y\} = 1/q. \end{aligned} \tag{7}$$

It can be easily shown that the above argument can be extended to random matrices. In (5), since Λ is uniformly distributed on $\mathbb{Z}_q^{t\times n}$ and $B_i \neq 0$, hence $B_i\Lambda$ and hence $B_i\Lambda - A_i'' C''$ is uniformly distributed on $\mathbb{Z}_q^{t\times n}$. Furthermore, since $C''^{-1} \neq 0$, then A_i' is uniformly distributed on $\mathbb{Z}_q^{t\times n}$. Finally $A_i = [\,A_i'\ A_i''\,]$ has a uniform distribution on $\mathbb{Z}_q^{t\times r}$. □

Lemma 2. *Let $A \in \mathbb{Z}_q^{t\times r}$ be a uniformly random matrix. The matrix A has full row rank with overwhelming probability.*

Proof. From [41], the probability of a uniformly random matrix chosen from $\mathbb{Z}_q^{m\times n} (m \leq n)$ having full row rank is $P(m,n) = \prod_{k=1}^{m} (1 - q^{k-1-n})$. Hence

$$\begin{aligned} P(t,r) &= \prod_{k=1}^{t} (1 - q^{-(r+1-k)}) \ > \ 1 - \sum_{k=1}^{t} q^{-(r+1-k)} \\ &\geq 1 - \sum_{k=1}^{t} q^{-(t\log t + 1 - k)}. \end{aligned} \tag{8}$$

Therefore, the probability that the matrix A does not have full row rank is $1 - P(t,r) \leq \sum_{k=1}^{t} q^{-(t\log t+1-k)} \approx q^{-(t\log t+1-t)} = \mathrm{negl}(t)$, which is a negligible function. Therefore, the matrix A has full row rank with overwhelming probability. □

Theorem 1. *Let A_k and A_i be randomly chosen matrices from $\mathbb{Z}_q^{t\times r}, r \geq \max(t\log t, n)$ and $x \in \{0,1\}^r$ is a random vector. $A_k x$ cannot be computed from $A_i x$ in polynomial time, where $1 \leq k \neq i \leq m$.*

Proof. The proof is by contradiction. Assume that there exists an algorithm $\mathcal{A}$ which outputs $A_k x$ on input $A_i x$, in polynomial time with a non-negligible probability, where A_i and A_k are chosen at random from $\mathbb{Z}_q^{t\times r}$, $r \geq \max(t\log t, n)$ and x is randomly chosen from $\{0,1\}^r$. Using $\mathcal{A}$, we propose an algorithm $\mathcal{B}$, that inverts Ajtai's function on input Ex and E, where E is randomly chosen from $\mathbb{Z}_q^{t\times r}$ and x is uniformly distributed on $\{0,1\}^r$. First, the algorithm $\mathcal{B}$ chooses a random matrix $A_k \in \mathbb{Z}_q^{t\times r}$. By Lemma 2, this matrix has full row rank with overwhelming probability. Then, it computes a matrix $G \in \mathbb{Z}_q^{r\times(r-t)}$, whose columns form a basis for the null space of matrix A_k, i.e., $A_k G = 0_{t\times(r-t)}$. Then, using Lemma 1, $\mathcal{B}$ computes $A_i = E[\,A_k^{\dagger} \quad G\,]^{-1}$, which is uniformly distributed on $\mathbb{Z}_q^{t\times r}$. The matrix $A_k^{\dagger} \in \mathbb{Z}_q^{r\times t}$ is the pseudo-inverse of the matrix A_k, which is equal to $A_k^T (A_k A_k^T)^{-1}$ because A_k has full row rank by Lemma 2. It should be noted that the matrix $[\,A_k^{\dagger} \quad G\,]$ is invertible because both $A_k^{\dagger}$ and G have full column rank and no column of $A_k^{\dagger}$ is in the linear space spanned by columns of G, for the reason that $A_k A_k^{\dagger} = I_t$, where I_t does not have any zero column and G is the null space of the matrix A_k. Let $y = [\,A_k^{\dagger} \quad G\,]x$. On input $Ex = A_i y$, the algorithm $\mathcal{A}$ outputs $z = A_k y = A_k[\,A_k^{\dagger} \quad G\,]x = [\,I_t \quad A_k G\,]x = x_1 + A_k G x_2 = x_1$, where $x = [\,{}^{x_{1_{t\times 1}}}_{x_{2_{(r-t)\times 1}}}\,]$. Therefore, on input Ex, the algorithm $\mathcal{B}$ outputs the first part of x, i.e. x_1, which contradicts with one-wayness of Ajtai's function. □

Theorem 2. *In the proposed scheme, any subset of participants of size less than t cannot recover the undisclosed secret $s_k, k = 1, \ldots, m$.*

Proof. Here, we consider the worst case, where $t-1$ participants take part in recovering the secret s_k. With regard to the Ajtai's definition of the one-way function $f_A(x) = Ax$, we proved in Theorem 1 that the pseudo-secret share corresponding to the secret s_k cannot be computed in

TABLE 1
Memory Requirements for Different Schemes

Scheme	Size of public values/Size of each secret	Size of shares/Size of each secret
The proposed MSSS scheme	$m \times r$	$r/(t \log q)$
He & Dawson [16]	$m \times n$	1
Harn [42]	$m \times (n-t)$	1
Chang [19]	$m \times n$	1

polynomial time from the revealed pseudo-secret shares corresponding to the already recovered secrets. On the other hand, any $t-1$ participants cannot compute the basis B_k, since they cannot solve the linear system of $(t-1) \times t$ equations and t^2 unknowns: $A_k c_{j_l} = B_k \lambda_{j_l}, l = 1, \ldots, t-1$, which has t degrees of freedom. We can consider the last column of B_k as $t (t > 1)$ free variables from $\mathbb{Z}_q$. Since $s_k = B_k v$ is a linear combination of the columns of B_k, hence, s_k has a uniform distribution over $\mathbb{Z}_q^t$ and no information about s_k can be extracted from the $t-1$ pseudo-secret shares. □

Furthermore, for sharing a new set of secrets, since the shares are not revealed in any secret reconstruction phase, they can be reused with some modifications. The dealer only needs to publish a vector $g \in \{0,1\}^r$ and the participants can use the vector $c_j \oplus g, j = 1, \ldots, n$ as their new shares. Since g is chosen at random, the new shares $c_j \oplus g$ are independent of the old shares c_j. So, the scheme is multi-use and a new share distribution through a secure channel is no longer required. It should be noted that the matrices $\{A_i, B_i\}, i = 1, \ldots, m$ are required to be recomputed by the dealer using the new shares and new secrets and he publishes the new matrices A_i on the bulletin board.

5 Performance Analysis

In this section, we investigate the performance of the proposed scheme. From memory consumption point of view, the matrices $A_i, i = 1, \ldots, m$ and the vectors $\lambda_j, j = 1, \ldots, n$ are published on the bulletin board, the latter of which can be neglected when compared with the former. The shares $c_j, j = 1, \ldots, n$ are sent securely to the participants. The memory requirements for these matrices are given in Table 1.

From complexity point of view, in the share distribution phase, the computation of $A_i, i = 1, \ldots, m$ from Equation (5) has the complexity of order $O(t^2 n) + O(tn(r-n)) + O(n^3) + O(tn^2) \sim O(n^3)$ for each secret and consists of three matrix multiplications and one matrix inversion. Recovery of each secret consists of two steps:

1) Computing the pseudo-secret shares: Since the shares are binary vectors, computing the pseudo-secret shares only requires simple column addition in matrix A_i, which has the complexity of $O(tr)$ for each participant. This makes the scheme suitable for the applications, where low-complex operations are needed at the participants' side.
2) The combiner's side: This step has the complexity of $O(t^3)$, which consists of one matrix multiplication and one matrix inversion.

In verification of the received shares by the participants, since the shares are binary vectors, the verification complexity is of $O(tr)$ per participant, resulting in an efficient scheme.

6 Conclusions

In this paper, we have considered a generalization of a secret sharing scheme which we call multi-stage secret sharing scheme. The desired level of security in such schemes is the computational security. In an MSSS scheme, the secrets may be recovered at different stages. The participants use pseudo-secret shares derived from their original shares to recover the secrets. In the proposed scheme, we obtain the pseudo-secret shares from the shares using lattice based one-way functions introduced by Ajtai. The new scheme is multi-stage, multi-use, verifiable, and provides the computational security based on the worst case hardness of lattice problems which makes it secure against quantum computers. Moreover, the scheme is significantly efficient in the participants' side and is suitable even if the participants have limited processing capabilities.

References

[1] M. Stadler, "Publicly verifiable secret sharing," in *Proc. Int. Conf. Theory Appl. Cryptographic Techn.*, 1996, pp. 190–199.
[2] B. Schoenmakers, "A simple publicly verifiable secret sharing scheme and its application to electronic voting," in *Proc. 19th Annu. Int. Cryptol. Conf. Adv. Cryptology*, 1999, pp. 148–164.
[3] V. Attasena, N. Harbi, and J. Darmont, "Sharing-based privacy and availability of cloud data warehouses," in *Proc. 9èmes journées francophones sur les Entrepôts de Données et l'Analyse en ligne*, 2013, pp. 17–32.
[4] W. Chunying, L. Shundong, and Z. Yiying, "Key management scheme based on secret sharing for wireless sensor network," in *Proc. 4th Int. Conf. Emerging Intell. Data Web Technol.*, Sep. 2013, pp. 574–578.
[5] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979.
[6] G. R. Blakley, "Safeguarding Cryptographic Keys," in *Proc. AFIPS Nat. Comput. Conf.*, Jun. 1979, vol. 48, pp. 313–317.
[7] M. Mignotte, "How to share a secret," in *Proc. Conf. Cryptography*, 1983, pp. 371–375.
[8] J. Benaloh and J. Leichter, "Generalized secret sharing and monotone functions," in *Proc. Adv. Cryptol.*, 1990, pp. 27–35.
[9] B. Chor, S. Goldwasser, S. Micali, and B. Awerbuch, "Verifiable secret sharing and achieving simultaneity in the presence of faults," in *Proc. 26th Annu. Symp. Found.Comput. Sci.*, 1985, pp. 383–395.
[10] E. Brickell and D. Stinson, "The detection of cheaters in threshold schemes," in *Proc. 8th Annu. Int. Cryptol. Conf. Adv. Cryptology*, 1990, vol. 403, pp. 564–577.
[11] W. Ogata and K. Kurosawa, "Optimum secret sharing scheme secure against cheating," in *Proc. 8th Annu. Int. Cryptol. Conf. Adv. Cryptol.*, 1996, vol. 1070, pp. 200–211.
[12] K. Martin, R. Safavi-Naini, and H. Wang, "Bounds and techniques for efficient redistribution of secret shares to new access structures," *Comput. J.*, vol. 42, no. 8, pp. 638–649, 1999.

[13] S. G. Barwick, W.-A. Jackson, and K. Martin, "Updating the parameters of a threshold scheme by minimal broadcast," *IEEE Trans. Inf. Theory*, vol. 51, no. 2, pp. 620–633, Feb. 2005.
[14] C. Blundo, A. De Santis, G. Di Crescenzo, A. G. Gaggia, and U. Vaccaro, "Multi-secret sharing schemes," in *Proc. 14th Annu. Int. Cryptol. Conf. Adv. Cryptol.*, 1994, pp. 150–163.
[15] M. Fatemi, R. Ghasemi, T. Eghlidos, and M. Aref, "Efficient multistage secret sharing scheme using bilinear map," *IET Inf. Security*, vol. 8, no. 4, pp. 224–229, Jul. 2014.
[16] J. He and E. Dawson, "Multistage secret sharing based on one-way function," *Electron. Lett.*, vol. 30, no. 19, pp. 1591–1592, Sep. 1994.
[17] Y. J. Geng, X. H. Fan, and H. Fan, "A new multi-secret sharing scheme with multi-policy," in *Proc. 9th Int. Conf. Adv. Commun. Technol.*, Feb. 2007, vol. 3, pp. 1515–1517.
[18] L. Pang, H. Li, and Y. Wang, "An efficient and secure multi-secret sharing scheme with general access structures," *Wuhan Univ. J. Natural Sci.*, vol. 11, no. 6, pp. 1649–1652, 2006.
[19] T. Y. Chang, M. S. Hwang, and W. P. Yang, "A new multi-stage secret sharing scheme using one-way function," *SIGOPS Oper. Syst. Rev.*, vol. 39, no. 1, pp. 48–55, Jan. 2005.
[20] A. Das and A. Adhikari. (2010). An efficient multi-use multi-secret sharing scheme based on hash function. *Appl. Math. Lett.* [Online]. *23(9)*, pp. 993–996. Available: http://www.sciencedirect.com/science/article/pii/S0893965910001308
[21] H.-Y. Chien, J.-K. Jan, and Y.-M. Tseng, "A practical (t,n) multi-secret sharing scheme," *IEICE Trans. Fundamentals Electron., Commun. Comput. Sci.*, vol. 83, no. 12, pp. 2762–2765, 2000.
[22] C.-C. Yang, T.-Y. Chang, and M.-S. Hwang, "A (t,n) multi-secret sharing scheme," *Appl. Math. Comput.*, vol. 151, no. 2, pp. 483–490, 2004.
[23] J. Shao and Z. Cao. (2005). A new efficient (t,n) verifiable multi-secret sharing (VMSS) based on YCH scheme. *Appl. Math. Comput.* [Online]. *168(1)*, pp. 135–140. Available: http://www.sciencedirect.com/science/article/pii/S0096300304005922
[24] P. W. Shor, "Algorithms for quantum computation: Discrete logarithms and factoring," in *Proc. 35th Annu. Symp. Found. Comput. Sc.*, 1994, pp. 124–134.
[25] R. J. McEliece. (1978). A public-key cryptosystem based on algebraic coding theory. *DSN Progress Report* [Online]. *42(44)*, pp. 114–116. Available: http://www.cs.colorado.edu/\~{}jrblack/class/csci7000/f03/papers/mceliece.pdf
[26] D. Bernstein, J. Buchmann, and E. Dahmen. (2009). *Post-Quantum Cryptography*. New York, NY, USA: Springer [Online]. Available: http://books.google.com/books?id=VB598lO47NAC
[27] M. Ajtai, "Generating hard instances of lattice problems (extended abstract)," in *Proc. 28th Annu. ACM Symp. Theory Comput.*, 1996, pp. 99–108.
[28] O. Goldreich, S. Goldwasser, and S. Halevi, "Collision-free hashing from lattice problems," in *Studies in Complexity and Cryptography: Miscellanea on the Interplay*. New York, NY, USA: Springer, 1996.
[29] O. Goldreich, S. Goldwasser, and S. Halevi, "Public-key cryptosystems from lattice reduction problems," in *Proc. 17th Annu. Int. Cryptol. Conf. Adv. Cryptol.*, 1997, vol. 1294, pp. 112–131.
[30] J. Hoffstein, J. Pipher, and J. Silverman, "Ntru: A ring-based public key cryptosystem," in *Proc. 3rd Int. Symp. Algorithmic Number Theory*, 1998, vol. 1423, pp. 267–288.
[31] A. Georgescu, "Article: A LWE-based secret sharing scheme," *IJCA Special Issue Netw. Security Cryptography*, vol. NSC, no. 3, pp. 27–29, Dec. 2011.
[32] R. El Bansarkhani and M. Meziani, "An efficient lattice-based secret sharing construction," in *Proc. 6th Inf. Security Theory Practice: Security, Privacy Trust Comput. Syst. Ambient Intell. Ecosyst.*, 2012, pp. 160–168.
[33] H. Amini, S. Asaad, T. Eghlidos, and M. Aref, "A lattice-based threshold secret sharing scheme," in *Proc. 11th Int. ISC Conf. Inf. Security Cryptology*, 2014, pp. 173–179.
[34] S. Asaad, H. Amini, T. Eghlidos, and M. Aref, "Sharing secret using lattice construction," in *Proc. Int. Symp. Telecommun.*, 2014, pp. 901–906.
[35] L. Babai, "On lov'asz lattice reduction and the nearest lattice point problem," *Combinatorica*, vol. 6, no. 1, pp. 1–13, 1986.
[36] R. Bendlin and I. Damgård, "Threshold decryption and zero-knowledge proofs for lattice-based cryptosystems," in *Proc. 7th Theory Cryptography Conf.*, 2010, pp. 201–218.
[37] R. Bendlin, S. Krehbiel, and C. Peikert, "How to share a lattice trapdoor: Threshold protocols for signatures and (h) ibe," in *Proc. 11th Int. Conf. Appl. Cryptography Netw. Security*, 2013, pp. 218–236.
[38] D. Micciancio and S. Goldwasser. (2002). *Complexity of Lattice Problems: A Cryptographic Perspective*. ser. Milken Institute Series on Financial Innovation and Economic Growth. New York, NY, USA: Springer. [Online]. Available: http://books.google.com/books?id=N4lHlGwy1AUC
[39] C. Gentry, C. Peikert, and V. Vaikuntanathan, "Trapdoors for hard lattices and new cryptographic constructions," in *Proc. 14th Annu. ACM Symp. Theory Comput.*, 2008, pp. 197–206.
[40] T.-Y. Chang, M.-S. Hwang, and W.-P. Yang, "An improved multi-stage secret sharing scheme based on the factorization problem," *Inf. Technol. Control*, vol. 40, no. 3, pp. 246–251, 2011.
[41] G. Mullen and D. Panario. (2013). *Handbook of Finite Fields*, ser. Discrete Mathematics and Its Applications. Boca Raton, FL, USA: CRC Press [Online]. Available: https://books.google.com/books?id=YADSBQAAQBAJ
[42] L. Harn, "Comment on 'multistage secret sharing based on one-way function'," *Electron. Lett.*, vol. 31, no. 4, p. 262, Feb. 1995.

Hossein Pilaram received the BSc degree in electrical engineering and the MSc degree in communication systems from the Sharif University of Technology, Tehran, Iran, in 2010 and 2012, respectively. He is currently working toward the PhD degree in the Department of Electrical Engineering, Sharif University of Technology. His research interests are cryptography, coding theory, and mobile networks.

Taraneh Eghlidos received the BSc degree in mathematics from the University of Shahid Beheshti, Tehran, Iran, in 1986, and the MSc degree in industrial mathematics from the University of Kaiserslautern, Germany, in 1991. She received the PhD degree in mathematics from the University of Giessen, Germany, in 2000. She joined the Sharif University of Technology (SUT) in 2002 and she is currently an associate professor in the Electronics Research Institute of SUT. Her research interests include interdisciplinary research areas such as symmetric and asymmetric cryptography, application of coding theory in cryptography, and mathematical modeling for representing and solving real world problems. Her current research interests include lattice based cryptography and code based cryptography.