# Lattice-Based Threshold-Changeability for Standard Shamir Secret-Sharing Schemes

Ron Steinfeld, Huaxiong Wang, and Josef Pieprzyk

Dept. of Computing, Macquarie University, North Ryde, Australia
{rons,hwang,josef}@ics.mq.edu.au
http://www.ics.mq.edu.au/acac/

**Abstract.** We consider the problem of increasing the threshold parameter of a secret-sharing scheme after the setup (share distribution) phase, without further communication between the dealer and the shareholders. Previous solutions to this problem require one to start off with a non-standard scheme designed specifically for this purpose, or to have communication between shareholders. In contrast, we show how to increase the threshold parameter of the *standard* Shamir secret-sharing scheme without communication between the shareholders. Our technique can thus be applied to existing Shamir schemes even if they were set up without consideration to future threshold increases.

Our method is a new positive cryptographic application for lattice reduction algorithms, inspired by recent work on lattice-based list decoding of Reed-Solomon codes with noise bounded in the Lee norm. We use fundamental results from the theory of lattices (Geometry of Numbers) to prove quantitative statements about the information-theoretic security of our construction. These lattice-based security proof techniques may be of independent interest.

**Keywords**: Shamir secret-sharing, changeable threshold, lattice reduction, geometry of numbers

## 1 Introduction

*Background.* A $(t, n)$-threshold secret-sharing scheme is a fundamental cryptographic scheme, which allows a *dealer* owning a secret to distribute this secret among a group of $n$ *shareholders* in such a way that any $t$ shareholders can reconstruct the secret, but no subset of less than $t$ shareholders can gain information on the secret. Classical constructions for $(t, n)$ secret-sharing schemes include the polynomial-based Shamir scheme [18] and the integer-based Chinese Remainder Theorem (CRT) scheme [2].

A common application for $(t, n)$ secret-sharing schemes is for achieving *robustness* of distributed security systems. A distributed system is called robust if system security is maintained even against an attacker who manages to break into/eavesdrop up to a certain number of components of the distributed system. For example, access control to a system can be enforced using a secret shared among $n$ system servers using a $(t, n)$-threshold secret-sharing scheme, while maintaining security if less than $t$ servers are compromised. In such applications, the threshold parameter $t$ must be determined by a security policy, based on an assessment which is a compromise between the value of the protected system and attacker capabilities on the one hand (which require as high a threshold as possible) and user convenience and cost on the other hand (which require as low a threshold as possible). In many settings, the system value and attacker capabilities are likely to change over time, thus requiring the security policy and hence threshold parameter $t$ to *vary over time*. In particular, an increase in system value or attacker capabilities after the initial setup with a relatively low threshold parameter $t$, will require an increase in the threshold parameter to a higher value $t' > t$. The longer the lifetime of the system, the more likely that such a change will be needed. Note that we assume that shareholders will cooperate honestly in making the transition to the larger threshold $t' > t$, since the attacker in our setting is an *outsider*.

*Previous Solutions.* A trivial solution to the problem of increasing the threshold parameter of a $(t, n)$-threshold secret-sharing scheme to $t' > t$ is for the shareholders to discard their old shares and for the dealer to distribute new shares of a $(t', n)$ secret-sharing scheme to all shareholders. However, this solution is not very attractive, since it requires the dealer to be involved after the setup stage and moreover requires communication between the dealer and each shareholder (such communication may be difficult to establish after the initial setup stage).

A much better solution would allow the threshold to be changed at any time without any communication between the dealer and shareholders after the setup stage. We say that such schemes allow *dealer-free* threshold changeability. A trivial dealer-free threshold changeable scheme can be constructed as follows: the dealer initially sets up $n - t + 1$ threshold schemes for each possible future threshold $t' \in \{t, t + 1, \ldots, n\}$, and gives to each shareholder $n - t + 1$ shares of the secret. Namely, for each $t' \in \{t, \ldots, n\}$, the shareholder receives a share of the secret for a $(t', n)$-threshold scheme. Such a trivial scheme may not be applicable because of the following drawbacks:

(1) *Non-Standard Initial Scheme:* The dealer must plan ahead for future threshold increases by initially setting up a non-standard $(t, n)$-threshold scheme designed specifically for threshold-changeability, whose shares consist of $n - t + 1$ shares corresponding to the $n - t + 1$ underlying $(t', n)$-threshold schemes. Hence the trivial scheme cannot be applied to increase the threshold of existing *standard* Shamir $(t, n)$-schemes which were not originally designed for threshold changeability and in which each shareholder has only a single share of *one* Shamir $(t, n)$-scheme.
(2) *Large Storage/Communication Requirements for Shareholders:* Each shareholder must receive and store $n - t + 1$ shares, where each share is as long as the secret (assuming that perfect security is desired). Hence the trivial scheme cannot be applied when storage or communication costs for $n - t + 1$ shares are prohibitive.

Other 'dealer-free' solutions to the threshold increase problem have been proposed in the literature (see related work below), but they all suffer from at least one of the two drawbacks above, or they require communication *between the shareholders*.

*Our Contributions.* In this paper, we present a new method for increasing the threshold of the *standard* Shamir $(t, n)$-threshold secret-sharing scheme[18], which does not have any of the drawbacks discussed above. In particular, and in contrast to previous solutions, our method does not require communication between the dealer and shareholders after the initial setup stage nor between shareholders, and can be applied to existing Shamir schemes even if they were set up without consideration to future threshold increase. Storage and communication costs are the same as for the standard Shamir scheme.

The basic idea of our method is the following: to increase the threshold from $t$ to $t' > t$, the shareholders add an appropriate amount of random noise to their shares (or delete a certain fraction of the bits of their share) to compute *subshares* which contain *partial* information about (e.g. half the most-significant bits of) the original shares. Since the subshares contain only partial information about the original shares, a set of $t$ subshares may no longer be sufficient to reconstruct the secret uniquely, but if one observes a sufficiently larger number $t' > t$ of subshares then one can expect the secret to be uniquely determined by these $t'$ subshares (e.g. if the subshares contain only half the information in the original shares then one can expect that $t' = 2t$ subshares will uniquely determine the secret)[1]. By replacing the share *combiner* algorithm of the original $(t, n)$-threshold secret-sharing with an appropriate 'error-correction' algorithm which can uniquely recover the secret from any $t'$ subshares, we obtain the desired threshold increase from $t$ to $t'$, leaving the secret unchanged.

Our efficient 'error-correction' combiner algorithm for the Shamir secret-sharing scheme is constructed using lattice basis reduction techniques. Thus, our method is a new positive cryptographic application for lattice reduction algorithms. Furthermore, we make use of fundamental tools from the theory of lattices (Geometry of Numbers) to prove quantitative statements about the information-theoretic security and correctness of our construction. These lattice-based security proof techniques may be of independent interest.

Although our threshold-increase method does not yield a perfect $(t', n)$ secret-sharing scheme, we obtain a useful result about the information-theoretic security of our method, which we believe suffices for many applications. Roughly speaking, we prove that for any desired $\epsilon > 0$, our method can be used to change the threshold to $t' > t$ (meaning that any $t'$ subshares can be used to recover the secret) such that any $t_s < t' - (t'/t)$ observed subshares leak to the attacker at most a fraction $\epsilon$ of the entropy of the secret, where $\epsilon$ can be made as small as we wish by an appropriate choice of security parameter.

Interestingly, our lattice-based methods can be adapted also to change the threshold of the standard integer-based Chinese Remainder Theorem (CRT)

[1] We remark that this intuitive reasoning is not rigorous, and indeed there exist examples for which it is incorrect. However, our results show that it is approximately true for the Shamir scheme.

secret-sharing scheme[2]. We provide full details of this result in a companion paper [22].

*Related Work.* Several approaches to changing the parameters of a threshold scheme in the absence of the dealer have been proposed in the literature. The technique of *secret redistribution*[5, 16] involves communication among the shareholders to 'redistribute' the secret with the a threshold parameter. Although this technique can be applied to standard secret-sharing schemes, its disadvantage is the need for secure channels for communication between shareholders. Methods for changing threshold which do not require secure channels have been studied in [4, 14, 15, 13], but they all require the initial secret-sharing scheme to be a non-standard one, specially designed for threshold increase (as a simple example of such a non-standard scheme, the dealer could provide each shareholder with two shares of the secret: one share for a $(t, n)$ scheme and one share for a $(t', n)$ scheme).

Our scheme uses a lattice-based 'error-correction' algorithm which is a slight variant of an algorithm for 'Noisy Polynomial Approximation' with noise bounded in the Lee norm [20]. This algorithm in turn is one of a large of body of recent work on 'list decoding' of Reed-Solomon and Chinese Remainder codes [9, 19, 6, 21]. We remark also that although the *correctness* proof of our scheme is based on the work of [20], our *security* proof is new and the lattice-based techniques used may be of independent interest.

*Organization of This Paper.* Section 2 presents notations, known results on lattices, and a counting lemma that we use. In Section 3, we provide definitions of changeable-threshold secret-sharing schemes and their correctness/security notions. In Section 4 we present the original Shamir $(t, n)$-threshold secret sharing scheme, and our threshold-changing algorithms to increase the threshold to $t' > t$. We then provide concrete proofs of the correctness and security properties of our scheme. Section 5 concludes the paper. Due to page limitations, some proofs have been omitted. They are included in the full version of this paper, available on the authors' web page.

## 2 Preliminaries

### 2.1 Notation

*Vectors and Polynomials.* For a vector $\mathbf{v} \in \mathbb{R}^n$, we write $\mathbf{v} = (\mathbf{v}[0], \ldots, \mathbf{v}[n-1])$, where, for $i = 0, \ldots, n-1$, $\mathbf{v}[i]$ denotes the $i$th coordinate of $\mathbf{v}$. Similarly for a polynomial $a(x) = a[0] + a[1]x + \ldots + a[t-1]x^{t-1}$, we let $a[i]$ denote the coefficient of $x^i$. For a ring $R$, we denote the set of all polynomial of degree at most $t$ with coefficients in the ring $R$ by $R[x; t]$.

*Lee and Infinity Norm.* For a prime $p$ and an integer $z$ we denote *Lee norm of $z$ modulo $p$* as $\|z\|_{L,p} = \min_{k \in \mathbb{Z}} |z - kp|$. Similarly, for a vector $\mathbf{v} \in \mathbb{Z}^n$, we define the Lee norm of $\mathbf{v}$ modulo $p$ by $\|\mathbf{v}\|_{L,p} = \max_{0 \le i \le n-1} \|\mathbf{v}[i]\|_{L,p}$. For a vector $\mathbf{z} = (z_1, \ldots, z_n) \in \mathbb{R}^n$, we denote the infinity norm of $\mathbf{z}$ by $\|\mathbf{z}\|_\infty = \max_{1 \le i \le n} |z_i|$. For integers $a$ and $p$, we denote $a \bmod p$ by $\lfloor a \rceil_p$. For real $z$ we define $Int(z) = \lceil z \rceil - 1$ as the largest integer strictly less than $z$.

*Sets.* For a set $S$, we denote by $\#S$ the size of $S$. For any set $S$ and integer $n$, we denote by $S^n$ the set of all $n$-tuples of elements from $S$ and by $D(S^n)$ the set of all $n$-tuples of *distinct* elements from $S$. For integer $n$, we denote by $[n]$ the set $\{1, 2, \ldots, n\}$.

*Entropy.* We denote by $\log(.)$ the logarithm function with base 2. For a discrete random variable $s$ with probability distribution $P_s(.)$ on a set $S$, we denote by $H(s \in S) = \sum_{x \in S} P_s(x) \log(1/P_s(x))$ the Shannon entropy of $s$. Let $t$ be any other random variable on a set $T$, and let $u$ denote any element of $T$. Let $P_s(.|u)$ denote the conditional probability distribution of $s$ given the event $t = u$. We denote by $H(s \in S|u) = \sum_{x \in S} P_s(x|u) \log(1/P_s(x|u))$ the conditional entropy of $s$ given the event $t = u$.

### 2.2 Lattices

Here we collect several known results that we use about lattices, which can be found in [8, 10, 7]. Let $\{\mathbf{b}_1, \ldots, \mathbf{b}_n\}$ be a set of $n$ linearly independent vectors in $\mathbb{R}^n$. The set

$$\mathcal{L} = \{\mathbf{z}: \mathbf{z} = c_1\mathbf{b}_1 + \ldots + c_n\mathbf{b}_n,\ c_1, \ldots, c_n \in \mathbb{Z}\}$$

is called an *$n$-dimensional (full-rank) lattice* with *basis* $\{\mathbf{b}_1, \ldots, \mathbf{b}_n\}$. Given a basis $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n) \in \mathbb{R}^n$ for a lattice $\mathcal{L}$, we define the associated *basis matrix* $M_{\mathcal{L},\mathbf{B}}$ to be the (full-rank) $n \times n$ matrix whose $i$th row is the $i$th basis vector $\mathbf{b}_i$ for $i = 1, \ldots, n$. The quantity $|\det(M_{\mathcal{L},\mathbf{B}})|$ is independent of $\mathbf{B}$. It is called the *determinant* of the lattice $\mathcal{L}$ and denoted by $\det(\mathcal{L})$.

Given a basis for lattice $\mathcal{L}$, the problem of finding a shortest non-zero vector in $\mathcal{L}$ is known as the *shortest vector problem*, or SVP. An algorithm is called an *SVP approximation algorithm with $\|\cdot\|_\infty$-approximation factor $\gamma_{SVP}$* if it is guaranteed to find a non-zero lattice vector $\mathbf{c}$ such that $\|\mathbf{c}\|_\infty \leq \gamma_{SVP} \min_{\mathbf{v} \in \mathcal{L} \setminus \mathbf{0}} \|\mathbf{v}\|_\infty$. The celebrated *LLL algorithm* of Lenstra, Lenstra and Lovász [12] is a *fully* polynomial time SVP approximation algorithm with $\|\cdot\|_\infty$-approximation factor $\gamma_{LLL} = n^{1/2}2^{n/2}$. Also, as shown in[1, 11], there exists an SVP approximation algorithm with $\|\cdot\|_\infty$-approximation factor $\gamma_{ex} = n^{1/2}$ which polynomial time in the size of elements of $M_{cL}$ but not in dimension of $\mathcal{L}$.

In this paper we actually need to solve a variation of SVP called the *closest vector problem* (CVP): given a basis of a lattice $\mathcal{L}$ in $\mathbb{R}^n$ and a "target" vector $\mathbf{t} \in \mathbb{R}^n$, find a lattice vector $\mathbf{c}$ such that $\|\mathbf{c} - \mathbf{t}\|_\infty$ is minimized. An algorithm is called a *CVP approximation algorithm with $\|\cdot\|_\infty$-approximation factor $\gamma_{CVP}$* if it is guaranteed to find a lattice vector $\mathbf{c}$ such that $\|\mathbf{c} - \mathbf{t}\|_\infty \leq \gamma_{CVP} \min_{\mathbf{v} \in \mathcal{L}} \|\mathbf{v} - \mathbf{t}\|_\infty$. Babai[3] has shown how to convert the LLL algorithm into a *fully* polynomial CVP approximation algorithm with $\|\cdot\|_\infty$-approximation factor $\gamma_{Bab} = n^{1/2}2^{n/2}$.

In our proof of security we use several fundamental theorems from the theory of lattices ('Geometry of Numbers'). The original theorems are quite general, but the restricted versions stated below suffice for our purposes. First, we need the following definition of *successive Minkowski minima* of a lattice.

**Definition 1 (Minkowski Minima).** *Let $\mathcal{L}$ be a lattice in $\mathbb{R}^n$. For $i = 1, \ldots, n$, the ith succesive Minkowski minimum of $\mathcal{L}$, denoted $\lambda_i(\mathcal{L})$, is the smallest real number such that there exists a set $\{\mathbf{b}_1, \ldots, \mathbf{b}_i\}$ of $i$ linearly-independent vectors in $\mathcal{L}$ with $\|\mathbf{b}_j\|_\infty \leq \lambda_i(\mathcal{L})$ for all $j = 1, \ldots, i$.*

Note that $\lambda_1(\mathcal{L})$ is just the shortest infinity-norm over all non-zero vectors in $\mathcal{L}$. Next, we state Minkowski's 'first theorem' in the geometry of numbers.

**Theorem 1 (Minkowski's First Theorem).** *Let $\mathcal{L}$ be a lattice in $\mathbb{R}^n$ and let $\lambda_1(\mathcal{L})$ denote the first Minkowski minimum of $\mathcal{L}$ (see Def. 1). Then $\lambda_1(\mathcal{L}) \leq \det(\mathcal{L})^{\frac{1}{n}}$.*

We will use the following point-counting variant of Minkowski's 'first theorem', which is due to Blichfeldt and van der Corput(see [8]).

**Theorem 2 (Blichfeldt-Corput).** *Let $\mathcal{L}$ be a lattice in $\mathbb{R}^n$ and let $K$ denote the origin-centered box $\{\mathbf{v} \in \mathbb{R}^n : \|\mathbf{v}\|_\infty < H\}$ of volume $Vol(K) = (2H)^n$. Then the number of points of the lattice $\mathcal{L}$ contained in the box $K$ is at least $2 \cdot Int\left(\frac{Vol(K)}{2^n \det(\mathcal{L})}\right) + 1$, where for any $z \in \mathbb{R}$, $Int(z)$ denotes the largest integer which is strictly less than $z$.*

Finally, we will also make use of Minkowski's 'second theorem' [8].

**Theorem 3 (Minkowski's Second Theorem).** *Let $\mathcal{L}$ be a full-rank lattice in $\mathbb{R}^n$ and let $\lambda_1(\mathcal{L}),\ldots,\lambda_n(\mathcal{L})$ denote the $n$ Minkowski minima of $\mathcal{L}$ (see Definition 1). Then $\lambda_1(\mathcal{L}) \cdots \lambda_n(\mathcal{L}) \leq 2^n \det(\mathcal{L})$.*

### 2.3 An Algebraic Counting Lemma

The following is a fundamental lemma that we use, interestingly, for *both* the correctness and security proofs of our construction. Fix a prime $p$ defining the finite field $\mathbb{Z}_p$, positive integer parameters $(\widehat{n}, \widehat{t}, \widehat{H})$, and an arbitrary set $\widehat{A}$ of polynomials of degree at least 1 and at most $\widehat{t}$ over $\mathbb{Z}_p$. The lemma gives us an upper bound on the probability that, for $\widehat{n}$ randomly chosen elements $\alpha_1, \ldots, \alpha_{\widehat{n}}$ of $\mathbb{Z}_p$, there will exist a polynomial $a(x) \in \widehat{A}$ which has 'small' absolute value modulo $p$ (less than $\widehat{H}$) at all the points $\alpha_1, \ldots, \alpha_{\widehat{n}}$. We remark that a similar (and more general) lemma was used in the analysis of a polynomial approximation algorithm [20]. Note that the lemma does not hold in general if we allow $\widehat{A}$ to contain constant polynomials, since these polynomials may have constant coefficient smaller than $\widehat{H}$.

**Lemma 1.** *Fix a prime $p$, positive integers $(\widehat{n}, \widehat{t}, \widehat{H})$, and a non-empty set $\widehat{A}$ of polynomials of degree at least 1 and at most $\widehat{t}$ with coefficients in $\mathbb{Z}_p$. Let $\mathcal{E}(\widehat{n}, \widehat{t}, \widehat{H}, \widehat{A}) \subseteq \mathbb{Z}_p^{\widehat{n}}$ denote the set of vectors $\alpha = (\alpha_1, \ldots, \alpha_{\widehat{n}}) \in \mathbb{Z}_p^{\widehat{n}}$ for which there exists a polynomial $a \in \widehat{A}$ such that $\|a(\alpha_i)\|_{L,p} < \widehat{H}$ for all $i = 1, \ldots, \widehat{n}$. The size of the set $\mathcal{E}(\widehat{n}, \widehat{t}, \widehat{H}, \widehat{A})$ is upper bounded as follows:*

$$\#\mathcal{E}(\widehat{n}, \widehat{t}, \widehat{H}, \widehat{A}) \leq \#\widehat{A} \cdot (2\widehat{H}\widehat{t})^{\widehat{n}}.$$

*Proof.* Suppose that $\alpha = (\alpha_1, \ldots, \alpha_{\widehat{n}}) \in \mathbb{Z}_p^{\widehat{n}}$ is such that there exists a polynomial $a \in \widehat{A}$ such that

$$\|a(\alpha_i)\|_{L,p} < \widehat{H} \text{ for } i = 1, \ldots, \widehat{n}. \quad (1)$$

It follows that there exist $\widehat{n}$ integers $r_1, \ldots, r_{\widehat{n}}$ such that, for each $i = 1, \ldots, \widehat{n}$, we have $a(\alpha_i) - r_i \equiv 0 \pmod p$ with $|r_i| < \widehat{H}$ and hence $\alpha_i$ is a zero of the polynomial $g_i(x) = a(x) - r_i$ over $\mathbb{Z}_p$. But for each $i$, $g_i$ is a polynomial of degree at least 1 and at most $\widehat{t}$ over $\mathbb{Z}_p$ and hence has at most $\widehat{t}$ zeros in $\mathbb{Z}_p$. So for each possible value for $(r_1, \ldots, r_n) \in (-\widehat{H}, \widehat{H})^{\widehat{n}}$ and $a \in \widehat{A}$, there are at most $\widehat{t}^{\widehat{n}}$ 'bad' values for $\alpha = (\alpha_1, \ldots, \alpha_{\widehat{n}})$ in $(\mathbb{Z}_p)^{\widehat{n}}$ such that (1) holds. Using the fact that there are less than $(2\widehat{H})^{\widehat{n}}$ possible values for $(r_1, \ldots, r_{\widehat{n}})$ and less than $\#\widehat{A}$ possible values for $a$, the claimed bound follows. □

## 3 Definition of Changeable-Threshold Secret-Sharing Schemes

We will use the following definition of a threshold secret-sharing scheme, which is a slight modification of the definition in [17].

**Definition 2 (Threshold Scheme).** *A $(t, n)$-threshold secret-sharing scheme* $\mathsf{TSS} = (\mathsf{GC}, \mathsf{D}, \mathsf{C})$ *consists of three efficient algorithms:*

1. $\mathsf{GC}$ *(Public Parameter Generation): Takes as input a security parameter $k \in \mathcal{N}$ and returns a string $x \in \mathcal{X}$ of public parameters.*
2. $\mathsf{D}$ *(Dealer Setup): Takes as input a security/public parameter pair $(k, x)$ and a secret $s$ from the secret space $\mathcal{S}(k, x) \subseteq \{0, 1\}^{k+1}$ and returns a list of $n$ shares $\mathbf{s} = (s_1, \ldots, s_n)$, where $s_i$ is in the $i$th share space $\mathcal{S}_i(k, x)$ for $i = 1, \ldots, n$. We denote by*

$$\mathsf{D}_{k,x}(.,.) : \mathcal{S}(k, x) \times \mathcal{R}(k, x) \to \mathcal{S}_1(k, x) \times \cdots \times \mathcal{S}_n(k, x)$$

   *the mapping induced by algorithm $\mathsf{D}$ (here $\mathcal{R}(k, x)$ denotes the space of random inputs to the probabilistic algorithm $\mathsf{D}$).*
3. $\mathsf{C}$ *(Share Combiner): Takes as input a security/public parameter pair $(k, x)$ and any subset $\mathbf{s}_I = (s_i : i \in I)$ of $t$ out of the $n$ shares, and returns a recovered secret $s \in \mathcal{S}(k, x)$. (here $I$ denotes a subset of $[n]$ of size $\#I = t$).*

The correctness and security properties of a $(t, n)$-threshold secret-sharing scheme can be quantified by the following definitions, which are modifications of those in [17].

**Definition 3 (Correctness, Security).** *A $(t, n)$ threshold secret-sharing scheme* $\mathsf{TSS} = (\mathsf{GC}, \mathsf{D}, \mathsf{C})$ *is said to be:*

1. $\delta_c$-correct*: If the secret recovery fails for a 'bad' set of public parameters of probability $p_f$ at most $\delta_c$. Precisely, $p_f$ is the probability (over $x = \mathsf{GC}(k) \in \mathcal{X}$) that there exist $(s, r) \in \mathcal{S}(k, x) \times \mathcal{R}(k, x)$ and $I \subseteq [n]$ with $\#I = t$ such that $\mathsf{C}_{k,x}(\mathbf{s}_I) \neq s$, where $\mathbf{s} = \mathsf{D}_{k,x}(s, r)$ and $\mathbf{s}_I \stackrel{\text{def}}{=} \{s_i : i \in I\}$.*
*We say that $\mathsf{TSS}$ is asymptotically correct if, for any $\delta > 0$, there exists $k_0 \in \mathcal{N}$ such that $\mathsf{TSS}$ is $\delta$-correct for all $k > k_0$.*
2. $(t_s, \delta_s, \epsilon_s)$-secure *with respect to the secret probability distribution $P_{k,x}$ on $\mathcal{S}(k, x)$: If, with probability at least $1-\delta_s$ over the choice of public parameters $x = \mathsf{GC}(k)$, the worst-case secret entropy loss for any $t_s$ observed shares is at most $\epsilon_s$, that is*

$$|L_{k,x}(\mathbf{s}_I)| \stackrel{\text{def}}{=} |H(s \in \mathcal{S}(k, x)) - H(s \in \mathcal{S}(k, x)|\mathbf{s}_I)| \leq \epsilon_s,$$

*for all $\mathbf{s} \in \mathcal{S}_1(k, x) \times \cdots \times \mathcal{S}_n(k, x)$ and $I \subseteq [n]$ with $\#I \leq t_s$. We say that $\mathsf{TSS}$ is asymptotically $t_s$-secure with respect to $P_{k,x}$ if, for any $\delta > 0$ and $\epsilon > 0$ there exists $k_0 \in \mathcal{N}$ such that $\mathsf{TSS}$ is $(t_s, \delta, \epsilon \cdot k)$-secure with respect to $P_{k,x}$ for all $k > k_0$.*

The following definition of the *Threshold Changeability* without dealer assistance for a secret sharing scheme is a modification of the definition in [15].

**Definition 4 (Threshold-Changeability).** *A $(t, n)$-threshold secret-sharing scheme $\mathsf{TSS} = (\mathsf{GC}, \mathsf{D}, \mathsf{C})$ is called* threshold-changeable *to $t'$ with $\delta_c$-correctness and $(t_s, \delta_s, \epsilon_s)$-security with respect to secret distribution $P_{k,x}$, if there exist $n$ efficient subshare generation algorithms $\mathsf{H}_i : \mathcal{S}_i(x, k) \to \mathcal{T}_i(x, k)$ for $i = 1, \ldots, n$, and an efficient subshare combiner algorithm $\mathsf{C}'$ such that the modified $(t', n)$-threshold scheme $\mathsf{TSS}' = (\mathsf{GC}, \mathsf{D}', \mathsf{C}')$, with modified shares*

$$\mathsf{D}'_{k,x}(s, r) \stackrel{\text{def}}{=} (\mathsf{H}_1(s_1), \ldots, \mathsf{H}_n(s_n)) \in \mathcal{T}_1(k, x) \times \cdots \mathcal{T}_n(k, x),$$

*where $(s_1, \ldots, s_n) = \mathsf{D}_{k,x}(s; r)$, is $\delta_c$-correct and $(t_s, \delta_s, \epsilon_s)$-secure with respect to $P_{k,x}$. $\mathsf{TSS}$ is called* asymptotically threshold-changeable to $(t_s, t')$ with respect to *$P_{k,x}$ if there exist algorithms $\mathsf{H}_i : \mathcal{S}_i(k, x) \to \mathcal{T}_i(k, x)$ $(i = 1, \ldots, n)$ and $\mathsf{C}'$ such that the $(t', n)$-threshold scheme $\mathsf{TSS}'$ defined above is asymptotically correct and asymptotically $t_s$-secure with respect to $P_{k,x}$.*

The idea captured by the above definition is that the change of threshold from $t$ to $t'$ is implemented by getting each shareholder to replace his original share $s_i$ by the subshare $\mathsf{H}_i(s_i)$ output by the subshare generation algorithm $\mathsf{H}_i$ (the original share $s_i$ is then discarded).

## 4 Threshold-Changeability for Shamir Secret-Sharing

### 4.1 The Standard Shamir Scheme

The standard Shamir $(t, n)$-threshold secret sharing scheme is defined as follows.

**Scheme** $\mathsf{ShaTSS} = (\mathsf{GC}, \mathsf{D}, \mathsf{C})$**: Shamir** $(t, n)$**-Threshold Secret-Sharing**

1. $\mathsf{GC}(k)$ (Public Parameter Generation):
   (a) Pick a (not necessarily random) prime $p \in [2^k, 2^{k+1}]$ with $p > n$.
   (b) Pick uniformly at random $n$ distinct non-zero elements $\alpha = (\alpha_1, \ldots, \alpha_n) \in D((\mathbb{Z}_p^*)^n)$. Return $x = (p, \alpha)$.
2. $\mathsf{D}_{k,x}(s, \mathbf{a})$ (Dealer Setup): To share secret $s \in \mathbb{Z}_p$ using $t-1$ uniformly random elements $\mathbf{a} = (a_1, \ldots, a_{t-1}) \in \mathbb{Z}_p^{t-1}$, build the polynomial $a_{s,\mathbf{a}}(x) = s + a_1 x + a_2 x^2 + \ldots + a_{t-1} x^{t-1} \in \mathbb{Z}_p[x; t-1]$. The $i$th share is $s_i = a(\alpha_i) \bmod p$ for $i = 1, \ldots, n$.
3. $\mathsf{C}_{k,x}(\mathbf{s}_I)$ (Share Combiner): To combine shares $\mathbf{s}_I = (s_i : i \in I)$ for some $I \subseteq [n]$ with $\#I = t$, compute by Lagrange interpolation the unique polynomial $b \in \mathbb{Z}_p[x; t-1]$ such that $b(\alpha_i) \equiv s_i \pmod p$ for all $i \in I$. The recovered secret is $s = b(0) \bmod p$.

### 4.2 Threshold-Changing Algorithms

Our threshold-changing subshare generation and combiner algorithms to change the $(t, n)$-threshold scheme $\mathsf{ShaTSS} = (\mathsf{GC}, \mathsf{D}, \mathsf{C})$ into a $(t', n)$-threshold scheme $\mathsf{ShaTSS}' = (\mathsf{GC}, \mathsf{D}', \mathsf{C}')$ are defined as follows. Note that the subshare combiner algorithm uses an efficient CVP approximation algorithm $\mathsf{A}_{\mathsf{CVP}}$ with $\|\cdot\|_\infty$-approximation factor $\gamma_{CVP}$. We define $\Gamma_{CVP} = \log(\lceil \gamma_{CVP} + 1 \rceil)$ (if we use the Babai poly-time CVP algorithm, we have $\Gamma_{CVP} \leq 1 + 0.5(t' + t + \log(t' + t))$).

**Scheme** $\mathsf{ShaTSS}'$**: Changing Threshold to** $t' > t$

1. $\mathsf{H}_i(s_i)$ ($i$th Subshare Generation): To transform share $s_i \in \mathbb{Z}_p$ of original $(t, n)$-threshold scheme into subshare $t_i \in \mathbb{Z}_p$ of desired $(t', n)$-threshold scheme ($t' > t$) the $i$th shareholder does the following (for all $i = 1, \ldots, n$):
   (a) Determine noise bound $H$ which guarantees $\delta_c$-correctness (typically, we set $\delta_c = k^{-t'}$):
      i. Set $H = \max(\lfloor p^\alpha / 2 \rfloor, 1)$ with
      ii. $\alpha = 1 - \frac{1+\delta_F}{(t'/t)} > 0$ (noise bitlength fraction) and
      iii. $\delta_F = \frac{(t'/t)}{k} \left( \log(\delta_c^{-1/t'} nt) + \Gamma_{CVP} + 1 \right)$.
   (b) Compute $t_i = \alpha_i \cdot s_i + r_i \bmod p$ for a uniformly random integer $r_i$ with $|r_i| < H$.
2. $\mathsf{C}'_{k,x}(\mathbf{t}_I)$ (Subshare Combiner): To combine subshares $\mathbf{t}_I = (t_i : i \in I)$ for some $I = \{i[1], \ldots, i[t']\}$ with $\#I = t'$ (and guaranteed $\delta_c$-correctness), do the following:

(a) Build the following $(t'+t) \times (t'+t)$ matrix $M_{Sha}(\alpha_I, H, p)$, whose rows form a basis for a full-rank lattice $\mathcal{L}_{Sha}(\alpha_I, H, p)$ in $\mathbb{Q}^{t'+t}$:

$$M_{Sha}(\alpha_I, H, p) = \begin{pmatrix} p & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & p & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & p & 0 & 0 & \dots & 0 \\ \alpha_{i[1]} & \alpha_{i[2]} & \dots & \alpha_{i[t']} & H/p & 0 & \dots & 0 \\ \alpha_{i[1]}^2 & \alpha_{i[2]}^2 & \dots & \alpha_{i[t']}^2 & 0 & H/p & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \alpha_{i[1]}^t & \alpha_{i[2]}^t & \dots & \alpha_{i[t']}^t & 0 & 0 & \dots & H/p \end{pmatrix}.$$

Here $H = \lfloor p^{\alpha}/2 \rfloor$, $\alpha = 1 - \frac{1+\delta_F}{(t'/t)}$, $\delta_F = \frac{(t'/t)}{k}\left(\log(\delta_c^{-1/t'} nt) + \Gamma_{CVP} + 1\right)$.

(b) Define $\mathbf{t}' = (t_{i[1]}, \dots, t_{i[t']}, 0, 0, \dots, 0) \in \mathbb{Z}^{t'+t}$.

(c) Run CVP Approx. alg. $\mathsf{A}_{\mathsf{CVP}}$ on lattice $\mathcal{L}_{Sha}(\alpha_I, H, p)$ given by $M_{Sha}(\alpha_I, H, p)$ with target vector $\mathbf{t}'$. Let $\mathbf{c} = (c_1, \dots, c_{t'}, c_{t'+1}, \dots, c_{t'+t}) \in \mathbb{Q}^{t'+t}$ denote the output vector returned by $\mathsf{A}_{\mathsf{CVP}}$, approximating the closest vector in $\mathcal{L}_{Sha}$ to $\mathbf{t}'$.

(d) Compute the recovered secret $\widehat{s} = (p/H) \cdot c_{t'+1} \bmod p$.

*Remark 1.* The reason for multiplying the shares $s_i$ by $\alpha_i$ before adding the noise $r_i$, is that otherwise, the secret may not be uniquely recoverable from the noisy subshares (indeed, $a(\alpha_i)+r_i = a(\alpha_i)+1+(r_i-1)$, and typically $|r_i-1| < H$, so secrets $s$ and $s+1$ would be indistinguishable).

*Remark 2.* It is not difficult to see that our method of adding a 'small' random noise integer $r_i$ with $|r_i| < H$ to the share multiple $\alpha_i \cdot s_i$ modulo $p$, is essentially equivalent (in the sense of information on the secret) to passing the residues $\alpha_i \cdot s_i \bmod p$ through a deterministic function which chops off the $\log(2H) \approx (1 - 1/(t'/t)) \cdot k$ least-significant bits of the $k$-bit residues $\alpha_i \cdot s_i \bmod p$, and this also yields shorter subshares than in our method above. However, since reducing the length of the original shares is not our main goal, we have chosen to present our scheme as above since it slightly simplifies our scheme and its analysis. Similar results can be obtained, however, for the 'deterministic' approach of chopping off least-significant bits.

*Remark 3.* Some special variants of the Shamir scheme use special values for the points $\alpha_i$, such as $\alpha_i = i$ for $i = 1, \dots, n$, to which the above method does not apply, because of its reliance on the random choice of the $\alpha_i$'s. However, it turns out that our method can be modified to work even for these special Shamir variants. The idea is to make up for the loss of randomness in the $\alpha_i$'s by getting the shareholders to multiply their shares by additional random integers (say $B_i \in \mathbb{Z}_p$) prior to adding the random noise $r_i$. The $B_i$'s are then sent along to the combiner with the noisy subshares. We do not analyze this variant of our scheme in this paper.

### 4.3 Correctness

The following theorem shows that the choice of the parameter $\delta_F$ used in our threshold changing algorithm is sufficient to guarantee the $\delta_c$-correctness of our scheme for all sufficiently large security parameters.

**Theorem 4 (Correctness).** *The scheme* $\mathsf{ShaTSS}'$ *(with parameter choice $\delta_c = k^{-t'}$) is asymptotically correct. Concretely, for any choice of parameter $\delta_c$ ($0 < \delta_c < 1$), the $(t', n)$ scheme* $\mathsf{ShaTSS}'$ *is $\delta_c$-correct for all security parameters $k$ satisfying the inequality $k \geq k'_0$, where*

$$k'_0 = \left(\frac{t'/t}{t'/t - 1}\right)\left(\log(\delta_c^{-1/t'} nt) + \Gamma_{CVP} + 2\right).$$

*Proof.* Fix a subshare subset $I \subseteq [n]$ with $\#I = t'$. We know by construction of lattice $\mathcal{L}_{Sha}(\alpha_I)$, that the dealer's secret polynomial $a_{s,\mathbf{a}}(x) = s + a_1 x + \ldots + a_{t-1}x^{t-1} \in \mathbb{Z}_p[x; t-1]$ gives rise to the lattice vector

$$\mathbf{a}' = (\alpha_{i[1]} a_{s,\mathbf{a}}(\alpha_{i[1]}) - k_1 p, \ldots, \alpha_{i[t']} a_{s,\mathbf{a}}(\alpha_{i[t']}) - k_{t'} p, (\frac{s}{p}H), \frac{a_1}{p}H, \ldots, \frac{a_{t-1}}{p}H),$$

which is "close" to the target vector

$$\mathbf{t}' = (\alpha_{i[1]} a_{s,\mathbf{a}}(\alpha_{i[1]}) - k_1 p + r_{i[1]}, \ldots, \alpha_{i[t']} a_{s,\mathbf{a}}(\alpha_{i[t']}) - k_{t'} p + r_{i[t']}, 0, 0, \ldots, 0),$$

where $k_j = \lfloor \frac{\alpha_{i[j]} a(\alpha_{i[j]}) + r_{i[j]}}{p} \rfloor \in \mathbb{Z}$ for all $j = 1, \ldots, t'$. In particular we have, using $|r_{i[j]}| < H$ for all $j = 1, \ldots, t'$, that $\|\mathbf{a}' - \mathbf{t}'\|_\infty < H$. Consequently, since $\mathsf{A}_{\mathsf{CVP}}$ is a CVP approximation algorithm with $\|.\|_\infty$ approximation factor $\gamma_{CVP}$, its output lattice vector $\mathbf{c}$ will also be "close" to the target vector, namely we have $\|\mathbf{c} - \mathbf{t}'\|_\infty < \gamma_{CVP} \cdot H$. Applying the triangle inequality, we conclude that the lattice vector $\mathbf{z} = \mathbf{c} - \mathbf{a}'$ satisfies

$$\|\mathbf{z}\|_\infty = \|\mathbf{c} - \mathbf{a}'\|_\infty < (\gamma_{CVP} + 1)H. \tag{2}$$

Now, either $\frac{p}{H}\mathbf{c}[t'+1] \equiv \frac{p}{H}\mathbf{a}'[t'+1] \equiv s \pmod p$ in which case the combiner succeeds to recover secret $s$, or otherwise we have the 'bad' case that

$$\frac{p}{H}\mathbf{z}[t'+1] = \frac{p}{H}\mathbf{c}[t'+1] - \frac{p}{H}\mathbf{a}'[t'+1] \not\equiv 0 \pmod p. \tag{3}$$

Hence, for fixed $I$, the combiner succeeds except for a fraction $\delta_I$ of 'bad' choices of $\alpha_I \in D((\mathbb{Z}_p^*)^{t'})$, for which $\mathcal{L}_{Sha}(\alpha_I)$ contains a 'short' and 'bad' vector $\mathbf{z}$ satisfying (2) and (3). To upper bound $\delta_I$, consider the polynomial $f(x) = \frac{p}{H}\mathbf{z}[t'+1]x + \cdots + \frac{p}{H}\mathbf{z}[t'+t]x^t$. Note that, since $\mathbf{z} \in \mathcal{L}_{Sha}$, we have $f(\alpha_{i[j]}) \equiv \mathbf{z}[j] \pmod p$ and hence $\|f(\alpha_{i[j]})\|_{L,p} < (\gamma_{CVP} + 1)H$ for all $j \in [t']$ using (2). Also, $f(x) \bmod p$ has zero constant coefficient and degree at least 1 and at most $t$ over $\mathbb{Z}_p$ using (3). Applying Lemma 1 (with parameters $\widehat{n} = t'$,$\widehat{t} = t$,$\widehat{H} = 2^{\Gamma_{CVP}}H$, $\#\widehat{A} \leq p^t$) we conclude that such a 'bad' polynomial $f$ exists for at most a fraction $\delta_I \leq p^t(2\widehat{H}t)^{t'}/\#D((\mathbb{Z}_p^*)^{t'})$ of $\alpha_I \in D((\mathbb{Z}_p^*)^{t'})$, for each fixed $I$. Hence, the probability $\delta$ that a uniformly chosen $\alpha \in D((\mathbb{Z}_p^*)^n)$ is 'bad' for *some* $I \subseteq [n]$ with $\#I = t'$ is upper bounded as

$$\delta \leq \frac{\binom{n}{t'} p^t (2\widehat{H}t)^{t'}}{\#D((\mathbb{Z}_p^*)^{t'})}, \tag{4}$$

and a straightforward calculation (see full paper) shows that the right-hand side of (4) is upper bounded by $\delta_c$ for all $k \geq \left(\frac{t'/t}{t'/t-1}\right)\left(\log(\delta_c^{-1/t'} nt) + \varGamma_{CVP} + 2\right)$. This completes the proof. □

### 4.4 Security

The concrete security of our scheme is given by the following result. It shows that, for fixed $(t', n)$ and with parameter choice $\delta_c = k^{-t'}$, the $(t', n)$ scheme $\mathsf{ShaTSS}'$ leaks at most fraction $\epsilon_s/k = O(\log k/k) = o(1)$ of the entropy of the secret to an attacker observing less than $t' - (t'/t)$ subshares (for all except a fraction $\delta_s \leq \delta_c = o(1)$ of public parameters, and assuming the security parameter $k$ is sufficiently large).

**Theorem 5 (Security).** *The scheme* $\mathsf{ShaTSS}'$ *(with parameter choice* $\delta_c = k^{-t'}$*) is asymptotically* $Int(t' - (t'/t))$*-secure with respect to the uniform secret distribution on* $\mathbb{Z}_p$*. Concretely, for any parameter choice* $\delta_c > 0$*, the* $(t', n)$ *scheme* $\mathsf{ShaTSS}'$ *is* $(t_s, \delta_s, \epsilon_s)$*-secure with:*

$$t_s = \frac{t' - (t'/t)}{1 + \frac{(t'/t)}{k}\left(\log(\delta_c^{-1/t'} nt) + \varGamma_{CVP} + 1\right)},$$

$$\delta_s = \delta_c,\ \epsilon_s = (\beta + 7)(t_s + t) + t_s \log t + 1,\ \beta = \frac{\log(2\delta_c^{-1}\binom{n}{t_s})}{t_s + t - 1},$$

*for all security parameters* $k \geq k_0$*, where, letting* $m = t_s + t + 1$ *and* $k_0'$ *as defined in Theorem 4,*

$$k_0 = \max\left(k_0' + \frac{(t'/t+1)^2}{t'/t-1}(\beta + \log t + 3), (\beta+4)m^2 + 5t_s m \log m\right).$$

*Proof.* (Sketch) Fix an observed subshare subset $I \subseteq [n]$ with $\#I = t_s$. Assuming the secret is uniformly distributed on $\mathbb{Z}_p$ it is easy to show (see full paper) that the conditional probability $P_{k,x}(s|\mathbf{s}_I)$ of the secret taking the value $s \in \mathbb{Z}_p$ given that the observed sub-share vector takes the value $\mathbf{s}_I$ is given by:

$$P_{k,x}(s|\mathbf{s}_I) = \frac{\#S_{s,p}(\alpha_I, t, p, H, \mathbf{s}_I)}{\#S_{0,1}(\alpha_I, t, p, H, \mathbf{s}_I)}, \tag{5}$$

where, for any integers $\widehat{s} \geq 0$ and $\widehat{p} \geq 1$, we define the set

$$S_{\widehat{s},\widehat{p}}(\alpha_I, t, p, H, \mathbf{s}_I) \stackrel{\text{def}}{=} \{a \in \mathbb{Z}_p[x; t-1] : \|\alpha_{i[j]} a(\alpha_{i[j]}) - s_{i[j]}\|_{L,p} < H \forall j \in [t_s]$$
$$\text{and } a(0) \equiv \widehat{s} \pmod{\widehat{p}}\}.$$

We will derive a probabilistic lower bound on $\#S_{0,1}$ and upper bound on $\#S_{s,p}$ which both hold for all except a fraction $\delta_I \leq \delta_s/\binom{n}{t_s}$ of 'bad' choices for $\alpha_I \in D((\mathbb{Z}_p^*)^{t_s})$ assuming $k \geq k_0$ (with $t_s$, $\delta_s$ and $k_0$ defined in the theorem statement). We then apply these bounds to (5) to get a bound $P_{k,x}(s|\mathbf{s}_I) \leq 2^{\epsilon_s}/p$ for all $s$ (with $\epsilon_s$ defined in the theorem statement) so that for fixed $I$, entropy loss is bounded as $L_{k,x}(\mathbf{s}_I) \leq \epsilon_s$, except for fraction $\delta_I$ of $\alpha_I \in D((\mathbb{Z}_p^*)^{t_s})$. It then follows that $L_{k,x}(\mathbf{s}_I) \leq \epsilon_s$ for *all* $I \subseteq [n]$ with $\#I = t_s$ except for a fraction $\delta \leq \binom{n}{t_s}\delta_I \leq \delta_s$ of $\alpha \in D((\mathbb{Z}_p^*)^n)$ assuming that $k \geq k_0$, which proves the theorem.

*Reduction to Lattice Point Counting.* It remains to derive the desired probabilistic upper and lower bounds on $\#S_{\widehat{s},\widehat{p}}$. The following lemma shows that $\#S_{\widehat{s},\widehat{p}}$ is equal to the number of points of a certain lattice $\mathcal{L}_{Sha}$ (closely related to the lattice used in our subshare combiner algoritm) contained in a $(t_s+t)$-dimensional box of side length $2H$, centered on a certain non-lattice vector $\widehat{\mathbf{s}}_I$.

**Lemma 2.** *Fix positive integers $(t, t_s, p, H, \widehat{p})$ such that $p \geq 2H$ and $\widehat{p}$ is a divisor of p. Let $\widehat{s} \in \mathbb{Z}_{\widehat{p}}$, $\alpha_I = (\alpha_{i[1]}, \ldots, \alpha_{i[t_s]}) \in \mathbb{Z}_p^n$ and $\mathbf{s}_I = (s_{i[1]}, \ldots, s_{i[t_s]}) \in \mathbb{Z}_p^{t_s}$.Define $\mathcal{L}_{Sha}(\alpha_I, t, p, H, \widehat{p})$ as the full-rank lattice in $\mathbb{Q}^{t_s+t}$ with basis consisting of the rows of the matrix*

$$M_{Sha}(\alpha_I, t, p, H, \widehat{p}) = \begin{pmatrix} p & 0 & \ldots & 0 & 0 & 0 & \ldots & 0 \\ 0 & p & \ldots & 0 & 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & p & 0 & 0 & \ldots & 0 \\ \widehat{p}\alpha_{i[1]} & \widehat{p}\alpha_{i[2]} & \ldots & \widehat{p}\alpha_{i[t_s]} & 2H/(p/\widehat{p}) & 0 & \ldots & 0 \\ \alpha_{i[1]}^2 & \alpha_{i[2]}^2 & \ldots & \alpha_{i[t_s]}^2 & 0 & 2H/p & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \alpha_{i[1]}^t & \alpha_{i[2]}^t & \ldots & \alpha_{i[t_s]}^t & 0 & 0 & \ldots & 2H/p \end{pmatrix},$$

*and define the vector $\widehat{\mathbf{s}}_I \in \mathbb{Q}_{t_s+t}$ by*

$$\widehat{\mathbf{s}}_I \stackrel{\text{def}}{=} \left( s_{i[1]} - \widehat{s}\alpha_{i[1]}, \ldots, s_{i[t_s]} - \widehat{s}\alpha_{i[t_s]}, H(1 - \frac{1+2\widehat{s}}{p}), H(1-\frac{1}{p}), \ldots, H(1-\frac{1}{p}) \right).$$

*Then the sizes of the following two sets are equal:*

$$S_{\widehat{s},\widehat{p}}(\alpha_I, t, p, H, \mathbf{s}_I) \stackrel{\text{def}}{=} \{a \in \mathbb{Z}_p[x; t-1] : \|\alpha_{i[j]} a(\alpha_{i[j]}) - s_{i[j]}\|_{L,p} < H \forall j \in [t_s] \\ \text{and } a(0) \equiv \widehat{s} \pmod{\widehat{p}}\},$$

*and*

$$V_{\widehat{s},\widehat{p}}(\alpha_I, t, p, H, \widehat{\mathbf{s}}_I) \stackrel{\text{def}}{=} \{\mathbf{v} \in \mathcal{L}_{Sha}(\alpha_I, t, p, H, \widehat{p}) : \|\mathbf{v} - \widehat{\mathbf{s}}_I\|_\infty < H\}.$$

*Finding a Lower Bound on $\#V_{0,1}$.* Lower bounding the number $\#V_{0,1}$ of points of the lattice $\mathcal{L}_{Sha}$ in a symmetric box $T_{\mathbf{s}_I}(H) \stackrel{\text{def}}{=} \{\mathbf{v} \in \mathbb{Q}^{t_s+1} : \|\mathbf{v} -$

$\widehat{\mathbf{s}}_I\|_\infty < H\}$ centered on vector $\widehat{\mathbf{s}}_I$ seems a difficult 'non-homogenous' problem because $\widehat{\mathbf{s}}_I$ is in general not a lattice vector. But by 'rounding' $\widehat{\mathbf{s}}_I$ to a nearby lattice vector $\widehat{\mathbf{s}}'_I$ (with rounding error $\epsilon = \|\widehat{\mathbf{s}}'_I - \widehat{\mathbf{s}}_I\|_\infty$), we reduce the problem to two simpler problems: (1) The 'homogenous' problem of lower bounding the number of lattice points in an *origin-centered* box $T_{\mathbf{0}} \stackrel{\text{def}}{=} \{\mathbf{v} \in \mathbb{Q}^{t_s+t} : \|\mathbf{v}\|_\infty < H - \epsilon\}$, and (2) Upper bounding the largest Minkowski minimum $\lambda_{t_s+t}(\mathcal{L}_{Sha})$ of the lattice. This general reduction is stated precisely as follows.

**Lemma 3.** *For any full-rank lattice $\mathcal{L}$ in $\mathbb{R}^n$, vector $\mathbf{s} \in \mathbb{R}^n$, and $H > 0$, we have*

$$\#\{\mathbf{v} \in \mathcal{L} : \|\mathbf{v} - \mathbf{s}\|_\infty < H\} \geq \#\{\mathbf{v} \in \mathcal{L} : \|\mathbf{v}\|_\infty < H - \epsilon\},$$

*where $\epsilon = \frac{n}{2} \cdot \lambda_n(\mathcal{L})$.*

To solve the 'homogenous' counting problem (1) above we directly apply the Blichfeldt-Corput theorem (Theorem 2 in Sec. 2). To solve the problem (2) above of upper bounding $\lambda_{t_s+t}(\mathcal{L}_{Sha})$, we apply Minkowski's "second theorem" (Theorem 3 in Sec. 2) to reduce this problem further to the problem of *lower bounding* the *first* Minkowski minimum $\lambda_1(\mathcal{L}_{Sha})$. Namely, since $\lambda_i(\mathcal{L}_{Sha}) \geq \lambda_1(\mathcal{L}_{Sha})$ for all $i \in [t_s]$, then Minkowski's second theorem gives $\lambda_{t_s+t}(\mathcal{L}_{Sha}) \leq \frac{2^{t_s+t}\det(\mathcal{L}_{Sha})}{\lambda_1(\mathcal{L}_{Sha})^{t_s+t-1}}$. Finally, to lower bound $\lambda_1(\mathcal{L}_{Sha})$ (i.e. the infinity norm of the shortest non-zero vector in $\mathcal{L}_{Sha}$), we use a probabilistic argument based on the algebraic counting lemma 1 (similar to the argument used in proving Theorem 4), to obtain the following result.

**Lemma 4.** *Fix positive integers $(t, t_s, p, H, \widehat{p})$ and a positive real number $\beta$, such that $p \geq \max(2H, 2t_s)$ is prime and $\widehat{p} \in \{1, p\}$. For each $\alpha_I \in D((\mathbb{Z}_p^*)^{t_s})$, let $\mathcal{L}_{Sha}(\alpha_I, \widehat{p})$ denote the lattice in $\mathbb{Q}^{t_s+t}$ with basis matrix $M_{Sha}(\alpha_I, \widehat{p})$ defined in Lemma 2, and let $\mathcal{L}'_{Sha}(\alpha_I)$ denote the lattice in $\mathbb{Q}^{t_s+t-1}$ with basis matrix $M'_{Sha}(\alpha_I)$ obtained from $M_{Sha}(\alpha_I, \widehat{p})$ by removing the $(t_s+1)$th row and column. In the case $\widehat{p} = 1$, if*

$$1 \leq 2^{-[\beta+3+\frac{t_s \log t}{t_s+t}]} \det\left(\mathcal{L}_{Sha}(\alpha_I, 1)\right)^{\frac{1}{t_s+t}} \leq H$$

*then, for at least a fraction $1 - 2^{-\beta(t_s+t)}$ of $\alpha_I \in D((\mathbb{Z}_p^*)^{t_s})$ we have*

$$\lambda_1(\mathcal{L}_{Sha}(\alpha_I, 1)) \geq 2^{-[\beta+3+\frac{t_s \log t}{t_s+t}]} \det(\mathcal{L}_{Sha}(\alpha_I, 1))^{\frac{1}{t_s+t}}.$$

*In the case $\widehat{p} = p$, if*

$$1 \leq 2^{-[\beta+3+\frac{t_s \log t}{t_s+t-1}]} \det(\mathcal{L}'_{Sha}(\alpha_I))^{\frac{1}{t_s+t-1}} \leq H$$

*then, for at least a fraction $1 - 2^{-\beta(t_s+t-1)}$ of $\alpha_I \in D((\mathbb{Z}_p^*)^{t_s})$ we have*

$$\lambda_1(\mathcal{L}'_{Sha}(\alpha_I)) \geq \lambda_1(\mathcal{L}_{Sha}(\alpha_I, p)) \geq 2^{-[\beta+3+\frac{t_s \log t}{t_s+t-1}]} \det(\mathcal{L}'_{Sha}(\alpha_I))^{\frac{1}{t_s+t-1}}.$$

Combining the above results (for $(\widehat{s}, \widehat{p}) = (0, 1)$) we obtain the desired lower bound on $\#V_{0,1}$.

*Finding an Upper Bound on* $\#V_{s,p}$. We first reduce the point counting problem in $\mathcal{L}_{Sha}(\alpha_I, p)$ to a point counting problem in the lower-dimensional lattice $\mathcal{L}'_{Sha}(\alpha_I)$ defined in Lemma 4. This is possible because all the vectors of $\mathcal{L}_{Sha}(\alpha_I, p)$ in the desired box have their $(t_s + 1)$th coordinate equal to 0.

**Lemma 5.** *Let $\mathcal{L}_{Sha}(\alpha_I, p) \subseteq \mathbb{Q}^{t_s+t}$ and $\mathcal{L}'_{Sha}(\alpha_I) \subseteq \mathbb{Q}^{t_s+t-1}$ be the lattices defined in Lemma 4, let $\widehat{\mathbf{s}}_I$ be the vector in $\mathbb{Q}^{t_s+t}$ defined in Lemma 2, and let $\widehat{\mathbf{s}}'_I$ be the vector in $\mathbb{Q}^{t_s+t-1}$ obtained from $\widehat{\mathbf{s}}_I$ by removing the $(t_s+1)$th coordinate. Then $\#V_{s,p} \leq \#V'_{s,p}$, where $V_{s,p} \stackrel{\text{def}}{=} \{\mathbf{v} \in \mathcal{L}_{Sha}(\alpha_I, p) : \|\mathbf{v} - \widehat{\mathbf{s}}_I\|_\infty < H\}$ and $V'_{s,p} \stackrel{\text{def}}{=} \{\mathbf{v} \in \mathcal{L}'_{Sha}(\alpha_I) : \|\mathbf{v} - \widehat{\mathbf{s}}'_I\|_\infty < H\}$.*

By comparing the total volume of the $\#V_{s,p}$ disjoint boxes of sidelength $\lambda_1(\mathcal{L}'_{Sha})$ centered on the lattice points in $T_{\widehat{\mathbf{s}}'_I}(H) \stackrel{\text{def}}{=} \{\mathbf{v} \in \mathbb{Q}^{t_s+t-1} : \|\mathbf{v} - \widehat{\mathbf{s}}'_I\|_\infty < H\}$, to the volume of $\widehat{T}_{\widehat{\mathbf{s}}'_I}(H) \stackrel{\text{def}}{=} \{\mathbf{v} \in \mathbb{Q}^{t_s+t-1} : \|\mathbf{v} - \widehat{\mathbf{s}}'_I\|_\infty < H + \lambda_1(\mathcal{L}'_{Sha})/2\}$ which contains those disjoint boxes, we reduce the problem of upper bounding $\#V_{s,p}$ to the problem of lower bounding the $\lambda_1(\mathcal{L}'_{Sha})$. This general reduction can be stated as follows.

**Lemma 6.** *For any lattice $\mathcal{L}$ in $\mathbb{R}^n$, vector $\mathbf{s} \in \mathbb{R}^n$, and $H > 0$, we have*

$$\#\{\mathbf{v} \in \mathcal{L} : \|\mathbf{v} - \mathbf{s}\|_\infty < H\} \leq \left[\frac{2H}{\lambda_1(\mathcal{L})} + 1\right]^n.$$

Now we apply the probabilistic lower bound on $\lambda_1(\mathcal{L}'_{Sha})$ from Lemma 4 in Lemma 6 (with $(\widehat{s}, \widehat{p}) = (s, p)$) to get the desired upper bound on $\#V_{s,p}$.

After some straightforward calculation (see full paper), we find that the probabilistic lower and upper bounds on $\#V_{\widehat{s},\widehat{p}}$ obtained above hold for all except a fraction $\delta_I \leq \delta_s/\binom{n}{t_s}$ of 'bad' choices for $\alpha_I \in D((\mathbb{Z}_p^*)^{t_s})$ assuming $k \geq k_0$ (with $t_s$, $\delta_s$ and $k_0$ defined in the theorem statement), and plugging the bounds in (5) gives the desired bound $P_{k,x}(s|\mathbf{s}_I) \leq 2^{\epsilon_s}/p$ for all $s$ (with $\epsilon_s$ defined in the theorem statement). This completes the proof sketch. □

An immediate consequence of the above results is the following.

**Corollary 1.** *For any $(t, n)$ and $t' > t$, the standard Shamir $(t, n)$-threshold secret-sharing scheme* ShaTSS *is asymptotically threshold-changeable to $(Int(t' - t'/t), t')$ with respect to the uniform secret distribution.*

## 5 Conclusions

We presented a new cryptographic application of lattice reduction techniques to achieve threshold-changeability for the standard Shamir $(t, n)$-threshold scheme. We proved concrete bounds on the correctness and security of our method, making use of fundamental results from lattice theory in our analysis.

**Acknowledgements.** We would like to thank Scott Contini and Igor Shparlinski for helpful discussions and encouragement to work on this problem. This work was supported by ARC Discovery Grants DP0345366 and DP0451484.

## References


1. M. Ajtai, R. Kumar, and D. Sivakumar. A Sieve Algorithm for the Shortest Lattice Vector Problem. In *Proc. 33-rd ACM Symp. on Theory of Comput.*, pages 601–610, New York, 2001. ACM Press.
2. C. Asmuth and J. Bloom. A Modular Approach to Key Safeguarding. *IEEE Trans. on Information Theory*, 29:208–210, 1983.
3. L. Babai. On Lovasz' Lattice Reduction and the Nearest Lattice Point Problem. *Combinatorica*, 6, 1986.
4. C. Blundo, A. Cresti, A. De Santis, and U. Vaccaro. Fully Dynamic Secret Sharing Schemes. In *CRYPTO '93*, volume 773 of *LNCS*, pages 110–125, Berlin, 1993. Springer-Verlag.
5. Y. Desmedt and S. Jajodia. Redistributing Secret Shares to New Access Structures and Its Application. Technical Report ISSE TR-97-01, George Mason University, 1997.
6. O. Goldreich, D. Ron, and M. Sudan. Chinese Remaindering with Errors. *IEEE Transactions on Information Theory*, 46:1330–1338, 2000.
7. M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer-Verlag, 1993.
8. P. Gruber and C. Lekkerkerker. *Geometry of Numbers*. Elsevier Science Publishers, 1987.
9. V. Guruswami and M. Sudan. Improved Decoding of Reed-Solomon Codes and Algebraic-Geometric Codes. *IEEE Trans. Inf. Th.*, 45:1757–1767, Sep. 1999.
10. E. Hlawka, J. Schoi$\beta$engeier, and R. Taschner. *Geometric and Analytic Number Theory*. Springer-Verlag, 1991.
11. R. Kannan. Algorithmic Geometry of Numbers. *Annual Review of Comp. Sci.*, 2:231–267, 1987.
12. A. K. Lenstra, H. W. Lenstra, and L. Lovász. Factoring Polynomials with Rational Coefficients. *Mathematische Annalen*, 261:515–534, 1982.
13. A. Maeda, A. Miyaji, and M. Tada. Efficient and Unconditionally Secure Verifiable Threshold Changeable Scheme. In *ACISP 2001*, volume 2119 of *LNCS*, pages 402–416, Berlin, 2001. Springer-Verlag.
14. K. Martin. Untrustworthy Participants in Secret Sharing Schemes. In *Cryptography and Coding III*, pages 255–264. Oxford University Press, 1993.
15. K. Martin, J. Pieprzyk, R. Safavi-Naini, and H. Wang. Changing Thresholds in the Absence of Secure Channels. *Australian Computer Journal*, 31:34–43, 1999.
16. K. Martin, R. Safavi-Naini, and H. Wang. Bounds and Techniques for Efficient Redistribution of Secret Shares to New Access Structures. *The Computer Journal*, 8, 1999.
17. M. Quisquater, B. Preneel, and J. Vandewalle. On the Security of the Threshold Scheme Based on the Chinese Remainder Theorem. In *PKC 2002*, volume 2274 of *LNCS*, pages 199–210, Berlin, 2002. Springer-Verlag.
18. A. Shamir. How To Share a Secret. *Comm. of the ACM*, 22:612–613, 1979.
19. M.A. Shokrollahi and H. Wasserman. List Decoding of Algebraic-Geometric Codes. *IEEE Transactions on Information Theory*, 45:432–437, March 1999.
20. I.E. Shparlinski. Sparse Polynomial Approximation in Finite Fields. In *Proc. 33rd STOC*, pages 209–215, New York, 2001. ACM Press.
21. I.E. Shparlinski and R. Steinfeld. Noisy Chinese Remaindering in the Lee Norm. *Journal of Complexity*, 20:423–437, 2004.
22. R. Steinfeld, J. Pieprzyk, and H. Wang. Dealer-Free Threshold Changeability for Standard CRT Secret-Sharing Schemes. Preprint, 2004.