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

Ron Steinfeld, Josef Pieprzyk, Huaxiong Wang
Centre for Advanced Computing – Algorithms and Cryptography
Dept. of Computing, Macquarie University, Australia
{rons, josef, hwang}@ics.mq.edu.au

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)-threshold secret-sharing schemes include the polynomial-based Shamir scheme [22] and the integer-based Chinese Remainder Theorem (CRT) scheme [1].

A common application for (t, n)-threshold 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

*This is the full version of a paper presented at Asiacrypt 2004, Dec. 5-9 2004, Jeju Island, South Korea.

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 all 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)-threshold 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)-threshold schemes which were not originally designed for threshold changeability and in which each shareholder has only a single share of *one* Shamir (t, n)-threshold 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 [22], 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. Note that the only communication required for increasing the threshold is a public signal broadcast by the share combiner to instruct the shareholders to modify their shares.

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)-threshold 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 ϵ of the entropy of the secret, where ϵ 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) secret-sharing scheme [1]. The basic common structure of the CRT and Shamir schemes that allows us to apply lattice techniques in both cases is that the shares can be expressed as known integer linear combinations of one or more secret integers modulo other known integers. The differences between the schemes arise from the structure of the different sets the above integers are chosen from, namely a set of prime moduli in the CRT scheme, or a set of polynomials in the Shamir scheme, and accordingly, we use the different (but analogous) properties of those sets to prove bounds on the properties of the different (but analogous) lattices involved in our scheme (e.g. the length of shortest vectors in the lattice). We provide full details of our results for the threshold changeable CRT scheme in a companion paper [26].

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* [7, 19] involves communication among the shareholders to 'redistribute' the secret with the new 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 [5, 17, 18, 16, 3], 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)-threshold scheme and one share for a (t', n)-threshold scheme). On the other hand, some of these non-standard schemes allow the secret to be changed for the new (t', n)-threshold scheme, so their security is maintained even if the t_s shares of the original (t, n)-threshold scheme are known (while in our scheme, the secret for the (t', n)-threshold scheme is the same as the secret for the original (t, n)-threshold scheme and hence we cannot achieve security for $t_s > t$ observed subshares). Nevertheless, for security against outsiders breaking into honest shareholders systems, it is reasonable to assume that shareholders will delete their original shares to protect against future outsider intrusions, so for such applications the weaker security property of our scheme should suffice.

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 [24]. This algorithm in

[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.

turn is one of a large body of recent work on 'list decoding' of Reed-Solomon and Chinese Remainder codes [11, 23, 8, 25]. We remark also that although the *correctness* proof of our scheme is based on the work of [24], our *security* proof is new and the lattice-based techniques used may be of independent interest.

We would also like to comment on the relation between our threshold increase method and the method for making secret-sharing schemes robust against cheating shareholders using error-correction [20]. In both methods, the share combiner (for a scheme with threshold t) receives $t' > t$ 'noisy' shares and applies an error-correction algorithm to overcome the noise and recover the secret. However, the type of noise which needs to be corrected (and hence also the decoding algorithm) is inherently different in the two cases. In the cheater robustness case, the noise vector (whose ith entry is the additive error in the ith share) is bounded in the *Hamming* norm: if the number of cheating shareholders is at most k then we know that up to k of the t' shares will be *arbitrarily corrupted* while the remaining shares will be correct. In our threshold increase case, the noise vector is bounded in the *Lee* norm: we have that *all* t' shares are corrupted but only by a *small* (in absolute value) additive noise. Note that a Hamming-bounded noise is not suitable for our threshold-increase method: we require that all shares be corrupted in an identical manner, to ensure that *any* subset of t shareholders cannot obtain information on the secret, and *any* subset of $t' > t$ shareholders can recover the secret. On the other hand, our Lee-bounded noise error-correction method cannot handle the Hamming-bounded noise where some shares are arbitrarily corrupted.

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.

2 Preliminaries

2.1 Notation

Sets. For a finite 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 $\{1, 2, \ldots, n\}$. We use (A, B) to denote the set of *integers* greater than A and less than B.

Vectors and Polynomials. For an n dimensional vector $\boldsymbol{v}$, we write $\boldsymbol{v} = (v_1, \ldots, v_n)$, where, for $i = 1, \ldots, n$, we denote by v_i the ith coordinate of $\boldsymbol{v}$. Given an n dimensional vector $\boldsymbol{v}$ and a subset $I = \{i_1, \ldots, i_t\}$ of $[n]$ of size t, we denote by $\boldsymbol{v}_I = (v_{i_1}, v_{i_2}, \ldots, v_{i_t})$ the t dimensional vector formed by coordinates of $\boldsymbol{v}$ whose indices are in the subset I (where, by convention, we may assume the ordering $i_1 < i_2 < \ldots < i_t$). 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 polynomials of degree at most t with coefficients in the ring R by $R[x; t]$.

Lee and Infinity Norms. 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 $\boldsymbol{v} \in \mathbb{Z}^n$, we define the Lee norm of $\boldsymbol{v}$ modulo p by $\|\boldsymbol{v}\|_{L,p} = \max_{1 \le i \le n} \|v_i\|_{L,p}$. For a vector $\boldsymbol{z} \in \mathbb{R}^n$, we denote the infinity norm of $\boldsymbol{z}$ by $\|\boldsymbol{z}\|_\infty = \max_{1 \le i \le n} |z_i|$. For integers a and p, we denote a mod p by $\lfloor a \rfloor_p$. For real z we define $Int(z) = \lceil z \rceil - 1$ as the largest integer strictly less than z.

Entropy. We denote by $\log(\cdot)$ the logarithm function with base 2. For a discrete random variable s with probability distribution $P_s(\cdot)$ on a set S, we denote by $H(s) = \sum_{x \in S} P_s(x) \log(1/P_s(x))$ the

Shannon entropy of s. Let $P_s(\cdot|u)$ denote the conditional probability distribution of s given the event u. We denote by $H(s|u) = \sum_{x \in S} P_s(x|u) \log(1/P_s(x|u))$ the conditional entropy of s given the event u.

2.2 Lattices

Here we collect several known results that we use about lattices, which can be found in [10, 13, 9]. Let $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$ be a set of n linearly independent vectors in $\mathbb{R}^n$. The set

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

is called an *n-dimensional (full-rank) lattice* with *basis* $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$. Given a basis $\boldsymbol{B} = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \in \mathbb{R}^n$ for a lattice $\mathcal{L}$, we define the associated *basis matrix* $M_{\mathcal{L},\boldsymbol{B}}$ to be the (full-rank) $n \times n$ matrix whose ith row is the ith basis vector $\boldsymbol{b}_i$ for $i = 1, \ldots, n$. The quantity $|\det(M_{\mathcal{L},\boldsymbol{B}})|$ is independent of the choice of basis $\boldsymbol{B}$. 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 γ_{SVP}* if it is guaranteed to find a non-zero lattice vector $\boldsymbol{c}$ such that $\|\boldsymbol{c}\|_\infty \leq \gamma_{SVP} \min_{\boldsymbol{v} \in \mathcal{L} \setminus \{\boldsymbol{0}\}} \|\boldsymbol{v}\|_\infty$. The celebrated *LLL algorithm* of Lenstra, Lenstra and Lovász [15] is a SVP approximation algorithm with polynomial running time (in the total bit length of the rational input basis) which achieves $\|\cdot\|_\infty$-approximation factor $\gamma_{LLL} = n^{1/2}2^{n/2}$.

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 $\boldsymbol{t} \in \mathbb{R}^n$, find a lattice vector $\boldsymbol{c}$ such that $\|\boldsymbol{c} - \boldsymbol{t}\|_\infty$ is minimized. An algorithm is called a *CVP approximation algorithm with $\|\cdot\|_\infty$-approximation factor γ_{CVP}* if it is guaranteed to find a lattice vector $\boldsymbol{c}$ such that $\|\boldsymbol{c} - \boldsymbol{t}\|_\infty \leq \gamma_{CVP} \min_{\boldsymbol{v} \in \mathcal{L}} \|\boldsymbol{v} - \boldsymbol{t}\|_\infty$. Babai [2] has shown how to convert the LLL algorithm into a polynomial running time CVP approximation algorithm which achieves $\|\cdot\|_\infty$-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. 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 2.1 (Minkowski Minima). *Let $\mathcal{L}$ be a lattice in $\mathbb{R}^n$. For $i = 1, \ldots, n$, the ith successive Minkowski minimum of $\mathcal{L}$, denoted $\lambda_i(\mathcal{L})$, is the smallest real number such that there exists a set $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_i\}$ of i linearly independent vectors in $\mathcal{L}$ with $\|\boldsymbol{b}_j\|_\infty \leq \lambda_i(\mathcal{L})$ for all $j = 1, \ldots, i$.*

Note that $\lambda_1(\mathcal{L})$ is just the shortest infinity-norm over all non-zero vectors in $\mathcal{L}$. Note that usual definitions of Minkowski minima refer to the Euclidean norm, whereas we use the infinity norm. Next, we state Minkowski's 'first theorem'.

Theorem 2.1 (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 Definition 2.1). 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 [10]).

Theorem 2.2 (Blichfeldt-Corput). *Let $\mathcal{L}$ be a lattice in $\mathbb{R}^n$ and let K denote the origin-centered box $\{\boldsymbol{v} \in \mathbb{R}^n : \|\boldsymbol{v}\|_\infty < H\}$ of volume $Vol(K) = (2H)^n$. Then the number of points of the lattice $\mathcal{L}$ contained in the box K is at least $2 \cdot Int\left(\frac{Vol(K)}{2^n \det(\mathcal{L})}\right) + 1$.*

Finally, we will also make use of Minkowski's 'second theorem' [10].

Theorem 2.3 (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 2.1). Then $\lambda_1(\mathcal{L})\cdots\lambda_n(\mathcal{L}) \leq \det(\mathcal{L})$.*

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 (n, t, B), and an arbitrary set A of polynomials of degree at least 1 and at most t over $\mathbb{Z}_p$. The lemma gives us an upper bound on the probability that, for n randomly chosen elements $\alpha_1, \ldots, \alpha_n$ of $\mathbb{Z}_p$, there will exist a polynomial $a(x) \in A$ which has 'small' absolute value modulo p (less than B) at all the points $\alpha_1, \ldots, \alpha_n$. We remark that a similar (and more general) lemma was used in the analysis of a polynomial approximation algorithm [24]. Note that the lemma does not hold in general if we allow A to contain constant polynomials, since these polynomials may have constant coefficient smaller than B.

Lemma 2.1. *Fix a prime p, positive integers (n, t, B), and a non-empty set A of polynomials of degree at least 1 and at most t with coefficients in $\mathbb{Z}_p$. Let $\mathcal{E}(n, B, A, p) \subseteq \mathbb{Z}_p^n$ denote the set of vectors $\boldsymbol{\alpha} \in \mathbb{Z}_p^n$ for which there exists a polynomial $a \in A$ such that $\|a(\alpha_i)\|_{L,p} < B$ for all $i = 1, \ldots, n$. The size of the set $\mathcal{E}(n, B, A, p)$ is upper bounded as follows:*

$$\#\mathcal{E}(n, B, A, p) \leq \#A \cdot (2Bt)^n.$$

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

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

It follows that there exist n integers $r_1, \ldots, r_n$ such that, for each $i = 1, \ldots, n$, we have $a(\alpha_i) - r_i \equiv 0 \pmod{p}$ with $|r_i| < B$ and hence α_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 t over $\mathbb{Z}_p$ and hence has at most t zeros in $\mathbb{Z}_p$. So for each possible value for $\boldsymbol{r} = (r_1, \ldots, r_n) \in (-B, B)^n$ and $a \in A$, there are at most t^n 'bad' values for $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ in $(\mathbb{Z}_p)^n$ such that (1) holds. Using the fact that there are less than $(2B)^n$ possible values for $\boldsymbol{r}$ and less than $\#A$ 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 [21].

Definition 3.1 (Threshold Scheme). *A (t, n)-threshold secret-sharing scheme $\mathsf{TSS} = (\mathsf{GC}, \mathsf{D}, \mathsf{C})$ consists of three efficient algorithms:*

1. 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. 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 $\boldsymbol{s} = (s_1, \ldots, s_n)$, where s_i is in the ith share space $\mathcal{S}_i(k, x)$ for $i = 1, \ldots, n$. We denote by*

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

   the mapping induced by algorithm D (here $\mathcal{R}(k, x)$ denotes the space of random inputs to the probabilistic algorithm D).

3. C *(Share Combiner): Takes as input a security/public parameter pair* (k, x) *and any subset* $\boldsymbol{s}_I = \{s_i : i \in I\}$ *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 [21].

Definition 3.2 (Correctness, Security). *A* (t, n)*-threshold secret-sharing scheme* $\mathsf{TSS} = (\mathsf{GC}, \mathsf{D}, \mathsf{C})$ *is said to be:*

1. δ_c-correct*: If the secret recovery may fail only for a 'bad' set of public parameters with probability* p_f *at most* δ_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* $\mathsf{C}_{k,x}(\boldsymbol{s}_I) \neq s$*, where* $\boldsymbol{s} = \mathsf{D}_{k,x}(s, r)$ *and* $\boldsymbol{s}_I \stackrel{\text{def}}{=} \{s_i : i \in I\}$*.*

   We say that TSS *is asymptotically correct if, for any* $\delta > 0$*, there exists* $k_0 \in \mathcal{N}$ *such that* TSS *is* δ*-correct for all* $k > k_0$*.*

2. $(t_s, \delta_s, \epsilon_s)$-secure *with respect to the probability distribution* $P_{k,x}$ *of the secret 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* ϵ_s*, that is*

$$L_{k,x}(\boldsymbol{\mu}_I) \stackrel{\text{def}}{=} |H(s) - H(s|\boldsymbol{s}_I = \boldsymbol{\mu}_I)| \leq \epsilon_s,$$

   for all share values $\boldsymbol{\mu} \in \mathcal{S}_1(k, x) \times \cdots \times \mathcal{S}_n(k, x)$ *and subsets* $I \subseteq [n]$ *with* $\#I = t_s$*, where secret* s *is sampled from the distribution* $P_{k,x}$ *and* $\boldsymbol{s} = \mathsf{D}_{k,x}(s, r)$ *for a uniformly random* $r \in \mathcal{R}(k, x)$*. We say that* 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* 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 [18].

Definition 3.3 (Threshold-Changeability). *A* (t, n)*-threshold secret-sharing scheme* $\mathsf{TSS} = (\mathsf{GC}, \mathsf{D}, \mathsf{C})$ *is called* threshold-changeable *to* t' *with* δ_c*-correctness and* $(t_s, \delta_s, \epsilon_s)$*-security with respect to probability distribution* $P_{k,x}$ *of the secret on* $\mathcal{S}(k, x)$*, if there exist* n *efficient subshare generation algorithms* $\mathsf{E}_i : \mathcal{S}_i(x, k) \to \mathcal{T}_i(x, k)$ *for* $i = 1, \ldots, n$*, and an efficient subshare combiner algorithm* C' *such that the modified* (t', n)*-threshold scheme* $\mathsf{TSS}' = (\mathsf{GC}, \mathsf{D}', \mathsf{C}')$*, with modified shares*

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

where $(s_1, \ldots, s_n) = \mathsf{D}_{k,x}(s, r)$*, is* δ_c*-correct and* $(t_s, \delta_s, \epsilon_s)$*-secure with respect to* $P_{k,x}$*.* TSS *is called* asymptotically threshold-changeable to (t_s, t') with respect to $P_{k,x}$ *if there exist algorithms* $\mathsf{E}_i : \mathcal{S}_i(k, x) \to \mathcal{T}_i(k, x)$ *(*$i = 1, \ldots, n$*) and* C' *such that the* (t', n)*-threshold scheme* 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 $\mathsf{E}_i(s_i)$ output by the subshare generation algorithm E_i (the original share s_i is then discarded).

Remark on Outsider vs. Insider Attacks. Our security model addresses a passive outsider attacker that can only observe up to t_s subshares. Accordingly, we assume that shareholders cooperate honestly in updating their shares to a higher threshold to protect against such outsider attacks. We also assume that the outsider attacker only sees subshares of the new scheme, rather than also shares of the original scheme. This does not model attackers who store some shares of the original scheme,

and then try to use those in conjunction with some subshares of the new scheme to recover the secret. We heuristically expect the subshare security threshold t_s against such attackers to drop by about t'/t for each additional observed original shares, but leave the rigorous analysis of this scenario as an open problem for future work.

Remark on δ_c-correctness of a (t,n)-threshold scheme. The δ_c-correctness requirement, although probabilistic, is quite strong since it is only probabilistic in the choice of public parameter x but not in the choice of the secret and the randomness used to generate the shares and subshares: for the "good" values of x, the share combiner is guaranteed to recover the secret, whatever the value of the secret and the values of the randomness used by the dealer and shareholders during subshare generation.

Remarks on $(t_s,\delta_s,\epsilon_s)$-security. The $(t_s,\delta_s,\epsilon_s)$ requirement guarantees that with at least $1-\delta_s$ probability, GC will output a 'good' scheme parameter x for which *any* t_s observed shares $\boldsymbol{s}_I$ leak at most $L_{k,x}(\boldsymbol{s}_I) \leq \epsilon_s$ bits of entropy of the secret s. Note that: (1) The requirement that $L_{k,x}(\boldsymbol{s}_I) \leq \epsilon_s$ *for all* $\boldsymbol{s}_I$, is a worst-case requirement and hence stronger than only requiring that the *average value* of $L_{k,x}(\boldsymbol{s}_I)$ over the $\boldsymbol{s}_I$ (which is known as the *average mutual information* between the secret and the share vector) is at most ϵ_s, and (2) When the secret is uniformly distributed, the asymptotic t_s-security requirement says that the *fraction* of secret entropy which is leaked to the attacker by t_s observed shares can be made as small as we wish with a suitably large security parameter k.

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 $\boldsymbol{\alpha} = (\alpha_1,\ldots,\alpha_n) \in D((\mathbb{Z}_p^*)^n)$. Return $x = (p,\boldsymbol{\alpha})$.
2. $\mathsf{D}_{k,x}(s,\boldsymbol{a})$ (Dealer Setup): To share secret $s \in \mathbb{Z}_p$ using $t-1$ uniformly random elements $\boldsymbol{a} = (a_1,\ldots,a_{t-1}) \in \mathbb{Z}_p^{t-1}$, build the polynomial $a(x) = s + a_1x + a_2x^2 + \ldots + a_{t-1}x^{t-1} \in \mathbb{Z}_p[x;t-1]$. The ith share is $\sigma_i = \lfloor a(\alpha_i) \rfloor_p$ for $i = 1,\ldots,n$.
3. $\mathsf{C}_{k,x}(\boldsymbol{s}_I)$ (Share Combiner): To combine shares $\boldsymbol{\sigma}_I = (\sigma_i : i \in I)$ 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 $b(\alpha_i) \equiv \sigma_i \pmod p$ for all $i \in I$. The recovered secret is $s = \lfloor b(0) \rfloor_p$.

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 runs an efficient CVP approximation algorithm $\mathsf{A}_{\mathsf{CVP}}$ with $\|\cdot\|_\infty$-approximation factor γ_{CVP} on a lattice of dimension $t'+t$. 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 ShaTSS': Changing Threshold to $t' > t$

1. $\mathsf{E}_i(\sigma_i)$ (ith Subshare Generation): To transform share $\sigma_i \in \mathbb{Z}_p$ of original (t,n)-threshold scheme into subshare $s_i \in \mathbb{Z}_p$ of desired (t',n)-threshold scheme ($t' > t$) the ith shareholder does the following (for all $i = 1, \ldots, n$):

   (a) Determine noise bound H which guarantees δ_c-correctness ($\delta_c = O(1/poly(k))$ is suitable):

      i. Set $H = \lfloor p^\alpha/2 \rfloor$ with

      ii. $\alpha = 1 - \frac{1+\delta_F}{t'/t} > 0$ (noise bitlength fraction) and

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

   (b) Compute $\mathsf{E}_i(\sigma_i) = s_i = \lfloor \alpha_i \cdot \sigma_i + r_i \rfloor_p$ for a uniformly random integer r_i with $|r_i| < H$.

2. $\mathsf{C}'_{k,x}(\boldsymbol{s}_I)$ (Subshare Combiner): To combine subshares $\boldsymbol{s}_I = (s_i : i \in I)$ for some $I = \{i_1, \ldots, i_{t'}\}$ with $\#I = t'$ (and guaranteed δ_c-correctness), do the following:

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

$$M_{Sha}(\boldsymbol{\alpha}_I, H, p) = \begin{pmatrix} p & 0 & \ldots & 0 & 0 & 0 & \ldots & 0 \\ 0 & p & \ldots & 0 & 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & p & 0 & 0 & \ldots & 0 \\ \alpha_{i_1} & \alpha_{i_2} & \ldots & \alpha_{i_{t'}} & H/p & 0 & \ldots & 0 \\ \alpha_{i_1}^2 & \alpha_{i_2}^2 & \ldots & \alpha_{i_{t'}}^2 & 0 & H/p & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \alpha_{i_1}^t & \alpha_{i_2}^t & \ldots & \alpha_{i_{t'}}^t & 0 & 0 & \ldots & H/p \end{pmatrix}.$$

   (b) Define $\boldsymbol{t}' = (s_{i_1}, \ldots, s_{i_{t'}}, 0, 0, \ldots, 0) \in \mathbb{Z}^{t'+t}$.

   (c) Run the CVP approximation algorithm $\mathsf{A_{CVP}}$ on lattice $\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, H, p)$ given by $M_{Sha}(\boldsymbol{\alpha}_I, H, p)$ with target vector $\boldsymbol{t}'$. Let $\boldsymbol{c} = (c_1, \ldots, c_{t'}, c_{t'+1}, \ldots, c_{t'+t}) \in \mathbb{Q}^{t'+t}$ denote the vector returned by $\mathsf{A_{CVP}}$, approximating the closest vector in $\mathcal{L}_{Sha}$ to $\boldsymbol{t}'$.

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

Intuition. To get some intuition for the correctness of the subshare combiner algorithm, observe that lattice $\mathcal{L}_{Sha}$ is constructed to contain a lattice vector $\boldsymbol{a}'$ related to the secret polynomial $a(x)$, while the target vector $\boldsymbol{t}'$ is 'close' to the lattice vector $\boldsymbol{a}'$, so we may hope that the CVP approximation algorithm $\mathsf{A_{CVP}}$ will return $\boldsymbol{a}'$, from which the coefficients of the secret polynomial, and hence the secret s, can be easily recovered. Namely, consider the lattice vector $\boldsymbol{a}' \in \mathcal{L}_{Sha}$ obtained by multiplying (for $i = 1, \ldots, t$) the $(t'+i)$th row of basis matrix M_{Sha} by the (integer) coefficient of x^i in the secret polynomial $x \cdot a(x) = s \cdot x + a_1 x^2 + a_2 x^3 + \cdots a_{t-1} x^t$, adding up these t scaled row vectors, and then subtracting the appropriate integer multiples of the first t' rows of M_{Sha} in order to reduce modulo p the first t' coordinates of the resulting vector $\boldsymbol{a}'$. Since the $(t'+i)$th row of M_{Sha} contains in its first t' coordinates the values of the monomial x^i at the points $\alpha_{i_1}, \ldots, \alpha_{i_{t'}}$, it follows that the first t' coordinates of the lattice vector $\boldsymbol{a}'$ contains integers congruent to $\alpha_{i_j} \cdot a(\alpha_{i_j}) \equiv \alpha_{i_j} \cdot \sigma_{i_j}$ modulo p. Hence, with the appropriate choice of the first t' row multipliers for mod p reduction, the first t' coordinates of $\boldsymbol{a}'$ differ from the corresponding t' subshares $s_{i_j} = \lfloor \alpha_{i_j} \cdot \sigma_{i_j} + r_{i_j} \rfloor_p$ (the first t' coordinates of the target vector $\boldsymbol{t}'$) by the 'small' noise integers r_{i_j} (with $|r_{i_j}| < H$) added by the subshare generation algorithm. Consequently, $\boldsymbol{a}'$ is 'close' to $\boldsymbol{t}'$, and we may hope that it will be recovered by

$\mathsf{A}_{\mathsf{CVP}}$. In our proof of correctness in Section 4.4, we show that, with high probability over the choice of the α_i's and for sufficiently large security parameters, $\boldsymbol{a}'$ is indeed the closest vector to $\mathcal{L}_{Sha}$ by a sufficient margin to guarantee that CVP approximation algorithm $\mathsf{A}_{\mathsf{CVP}}$ returns $\boldsymbol{a}'$. Note that the last t coordinates of $\boldsymbol{a}'$ contain the (scaled) coefficients of the secret polynomial $a(x)$, including the secret s.

Remark 1 (Unique Secret Recovery). The reason for multiplying the shares σ_i by α_i before adding the noise r_i, is that otherwise, the secret may not be uniquely recoverable, even given all n noisy subshares. Indeed, $\lfloor a(\alpha_i) + r_i \rfloor_p = \lfloor a(\alpha_i) + 1 + (r_i - 1) \rfloor_p$, and typically $|r_i - 1| < H$, so the subshare vector $\boldsymbol{s} = (s_1, \ldots, s_n)$ for secret s with noise integers r_i would be equal to, and hence indistinguishable from, the subshare vector for secret $s' = s + 1$ with noise integers $r'_i = r_i - 1$. In contrast, in our scheme (with multiplication by α_i) subshare vectors of any two distinct secrets $s \neq s'$ (containing at least t' subshares) are unequal except with small probability (less than δ_c) over the random choice of $\alpha_1, \ldots, \alpha_n$'s. To understand the reason for this, note that the ith subshares of s and s' are of the form $s_i = \lfloor \alpha_i \cdot a(\alpha_i) + r_i \rfloor_p$ and $s'_i = \lfloor \alpha_i \cdot a'(\alpha_i) + r'_i \rfloor_p$ respectively, where polynomials a and a' differ in their constant coefficients s and s' respectively. So an equality of subshares $s_i = s'_i$ for $i = 1, \ldots, t'$ implies that $\alpha_i \cdot (a(\alpha_i) - a'(\alpha_i)) \equiv r'_i - r_i \pmod{p}$, i.e. the polynomial $h(x) = x \cdot (a(x) - a'(x))$ has a 'small' value $r'_i - r_i$ (less than $2H$ in absolute value) at the points α_i modulo p. The crucial point is that $h(x)$ has degree at least 1 (thanks to the multiplication of α_i) and at most t, so, as shown in Lemma 2.1, $h(x)$ can have 'small' values (less than $2H$ in absolute value) only at a relatively small 'bad' set of up to $2H \cdot t$ points in $\mathbb{Z}_p$, and this 'bad' set is unlikely to be hit by all the randomly chosen α_i's when t' is sufficiently large. More precisely, since there are less than p^t possibilities for $h(x)$, then, applying Lemma 2.1, the chance that a particular set of t' α_i's will all be 'bad' (and hence the corresponding t' subshare vectors of s and s' will be equal) is at most $p^t (2Ht)^{t'} / p^{t'}$. The error probability bound we obtain for our subshare recovery algorithm is close to this bound, (see Eq. (4) in the proof of Theorem 4.1). Note also that the above non-uniqueness probability upper bound is non-trivial (less than 1) when $t' \cdot \left(1 - \frac{\log(2Ht)}{\log(p)}\right) > t$. This is close to the secret recovery condition $t' \cdot \left(1 - \frac{\log(2H)}{\log(p)}\right) > t$ obtained from following the heuristic intuitive reasoning discussed in Section 1, where we expect that each noisy subshare contains a fraction of about $\left(1 - \frac{\log(2H)}{\log(p)}\right)$ of the $\log(p)$ bits of information about the secret polynomial $a(x)$ in the original share, due to the addition of about $\log(2H)$ bits of noise to the share. Namely, assuming the information on $a(x)$ provided by distinct subshares is additive, we heuristically expect to recover $a(x)$ uniquely once the total information $t' \cdot \left(1 - \frac{\log(2H)}{\log(p)}\right) \cdot \log(p)$ contained in the t' subshares exceeds the total information $t \cdot \log(p)$ in $a(x)$.

Remark 2. Our method of adding a 'small' random noise integer r_i with $|r_i| < H$ to the share multiple $\alpha_i \cdot \sigma_i$ modulo p, is essentially equivalent (in the sense of information on the secret) to passing the residues $\lfloor \alpha_i \cdot \sigma_i \rfloor_p$ through a deterministic function which chops off the $\log(2H) \approx (1 - \frac{1}{t'/t}) \cdot k$ least-significant bits of the k-bit residues $\lfloor \alpha_i \cdot \sigma_i \bmod p \rfloor_p$, 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 to simplify the analysis.

Remark 3. Some special variants of the Shamir scheme use special values for the points α_i, such as $\alpha_i = i$ for $i = 1, \ldots, n$, to which the above method does not apply, because of its reliance on the random choice of the α_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 α_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.

Remark 4. The scheme allows more than one successive increase in the threshold by adding 'additional' noise as required. For example, suppose $s_i = \lfloor \alpha_i \cdot \sigma_i + r_i \rfloor_p$ is the ith subshare after

increasing the threshold from t to $t' > t$ by adding random 'noise' integer $|r_i| < H$, and supppose that $H' \approx (2R+1)H$ for some integer R is the noise bound required by our scheme for increasing the threshold from t to $t'' > t'$. Then the ith shareholder can simulate the threshold increase from t' to t'' by choosing a uniformly random integer u with $|u| \leq R$ and modifying the ith subshare s_i to a new subshare $s'_i = \lfloor s_i + (2H) \cdot u \rfloor_p = \lfloor \alpha_i \cdot \sigma_i + r'_i \rfloor_p$, where integer $r'_i = (2H) \cdot u + r_i$. Note that $r'_i = r_i + (2H) \cdot u$ is almost uniformly random in interval $(-H', H')$ when r_i is uniform in $(-H, H)$ and u is uniform in $[-R, R]$, as required for changing to threshold t''.

Remark 5. As we show in the following sections, the choice $\delta_c = O(1/poly(k))$ achieves both asymptotic correctness and security.

4.3 Summary of Analysis Results

Our analysis results can be summarised by two main theorems.

The first theorem shows that the choice of the parameter δ_F used in our threshold changing algorithm is sufficient to guarantee the δ_c-correctness of our scheme for all sufficiently large security parameters.

Theorem 4.1 (Correctness). *The scheme* ShaTSS′ *with parameter choice* $\delta_c = O(1/poly(k))$ *is asymptotically correct. Concretely, for any choice of parameter* δ_c *(*$0 < \delta_c < 1$*), the* (t', n)*-threshold scheme* ShaTSS′ *is* δ_c*-correct for all security parameters* k *satisfying the inequality* $k \geq k'_0$*, where*

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

The concrete security of our scheme is given by the second theorem. It shows that, for fixed (t', n) and with parameter choice $\delta_c = O(1/poly(k))$, the (t', n)-threshold scheme 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).

Theorem 4.2 (Security). *The scheme* ShaTSS′ *with parameter choice* $\delta_c = O(1/poly(k))$ *is asymptotically* $Int(t' - t'/t)$*-secure with respect to the uniform probability distribution of the secret on* $\mathbb{Z}_p$*. Concretely, for any parameter choice* $\delta_c > 0$*, the* (t', n)*-threshold scheme* ShaTSS′ *is* $(t_s, \delta_s, \epsilon_s)$*-secure with:*

$$t_s \leq \left\lfloor \frac{t' - t'/t}{1 + \frac{t'/t}{k} \left(\log(\delta_c^{-1/t'} nt) + \Gamma_{CVP} + 1 \right)} \right\rfloor,$$

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

for all security parameters $k \geq k_0$*, where, letting* $m = t_s + t$ *and* k'_0 *as defined in Theorem 4.1,*

$$k_0 = \max \left(k'_0 + \frac{(t'/t + 1)^2}{t'/t - 1} (\beta + \log t + 3), (\beta + 3) \left(m^2 + m - 1 \right) + m \left(t_s \log t + \log m \right) + t_s \log t + 1 \right).$$

We would like to make a couple of remarks on the security of our scheme.

First, the limitation $t_s \leq t' - t'/t$ for (close to perfect) security is inherent to schemes which increase the threshold by adding small noise to shares, and is not due to our Shamir based implementation. As remarked in Section 4.2, our approach of increasing the threshold from t to t' by adding small noise integers of bit length $\log(2H) \approx (1 - \frac{1}{m}) \cdot k$ to the k bits shares is essentially equivalent to truncating the shares to just their k/m most significant bits for $m = t'/t$. The information theoretic security limitations of such schemes have been studied in [18], where it is pointed out that for any initial

perfect (t,n)-threshold scheme with a k bits secret and k bits shares, the (t',n)-threshold scheme obtained by truncating shares to k/m bits is a $(t-1,t',n)$ *ramp* scheme [6, 14]. It is known [6, 14] that in any such ramp scheme, the entropy of the secret given t_s observed shares (which is obviously zero for $t_s \geq t'$ and equal to k for $t_s \leq t-1$) is upper bounded by $\frac{t'-t_s}{m} \cdot k$ in the ramp region $t-1 \leq t_s \leq t'$. It follows that for our scheme with $m = t'/t$, the entropy of the secret is substantially less than k bits (so that perfect security cannot be achieved) when $t_s > t' - t'/t$ subshares are observed, matching asymptotically for $k \to \infty$ (up to a factor $1 + o(1)$) the upper bound on t_s for which our security result in Theorem 4.2 applies. On the other hand, in the case $t_s \approx t' - t'/t$, it is an interesting open problem whether our bound in Theorem 4.2 on the absolute secret entropy leaked by the observed subshares is essentially tight or can be improved.

Second, we note that although we state in Theorem 4.2 a lower bound on the conditional *Shannon* entropy of the secret $H(s|\boldsymbol{s}_I = \boldsymbol{\mu}_I)$ for any observed share value $\boldsymbol{\mu}_I$, our proof shows the stronger result that the stated bound is also a lower bound on the conditional *min-entropy* $H_\infty(s|\boldsymbol{s}_I = \boldsymbol{\mu}_I) = \log(1/\max_{s \in \mathcal{S}(k,x)} P_{k,x}(s|\boldsymbol{s}_I = \boldsymbol{\mu}_I))$, and hence also a lower bound on the conditional *Rényi* entropy of s given $\boldsymbol{s}_I = \boldsymbol{\mu}_I$. This means we can apply the privacy amplification results of [4] to derive a secret s' (by hashing s with a public randomly chosen function from a universal hash family) such that a provably negligible absolute amount of entropy of s' is leaked by the observed shares $\boldsymbol{s}_I$.

Before we present the proofs of these theorems, let us present some example parameter settings.

Example 1 (Concrete). Suppose we have $n = 20$ shareholders sharing a secret of length $k+1 = 1000$ bits (using a prime modulus $p \approx 2^{1000}$) with an original threshold $t = 3$, and we wish to increase the threshold to $t' = 8$ with $\delta_c = 2^{-20}$ correctness (subshare combiner error probability less than 1 in a million). We have $t'/t \approx 2.67$, $\log(\delta_c^{-1/t'} nt) \approx 8.4$. The lattice $\mathcal{L}_{Sha}$ used by the subshare combiner has dimension $t' + t = 11$. Assume we use the Babai CVP approximation algorithm $\mathsf{A_{CVP}}$ which has infinity-norm approximation factor $\gamma_{CVP} \leq (t'+t)^{1/2} 2^{(t'+t)/2} \approx 2^{14.5}$ so $\Gamma_{CVP} \approx 7.2$. The subshare combiner parameters are $\delta_F \approx 0.022$, noise bitlength fraction $\alpha = 1 - \frac{1+\delta_F}{t'/t} \approx 0.62$, and noise bound $H = \lfloor p^\alpha/2 \rfloor \approx p^{1262}$. By Theorem 4.1, the subshare combiner will achieve $\delta_c = 2^{-20}$ correctness using $t' = 8$ subshares as long as we use a security parameter k greater or equal to $k'_0 = \left(\frac{t'/t}{t'/t-1}\left(\log\left(\delta_c^{-1/t'} nt\right) + \Gamma_{CVP} + 1\right)\right) \approx 29$, which is satisfied by our choice $k + 1 = 1000$. The security result Theorem 4.2 applies against attackers observing up to t_s shares, where $t_s = 5$. Notice that this 'security threshold' t_s is lower than t by about $t'/t \approx 3$ (this is essentially due to the fact that each subshare contains only approximately a fraction $t/t' = 0.375$ of the information in the original share, hence by correctness, for $t_s > t' - t'/t$ observed subshares a constant fraction of the secret entropy leaks to the attacker). For $t_s = 5$ observed shares, except for a probability of at most $\delta_s = \delta_c = 2^{-20}$ over the choice of the α_i's, the entropy leaked to the attacker is at most $\epsilon_s \approx 104.8$ bits (about 10% of the secret entropy). This bound holds for security parameters k exceeding $k_0 \approx 820$ bits, which is satisfied by our choice $k + 1 = 1000$.

Example 2 (Asymptotic). Asymptotically, suppose that we let original threshold t grow and set $t' = c_1 t$ and $n = c_2 t$ for some constants $c_1 < c_2$, using security parameter $k = c_3 t^2 \log t$ for some constant c_3 so $p \approx 2^{c_3 t^2 \log t}$, and we require correctness $\delta_c = t^{-c_4} = o(1)$ for some constant c_4. The lattice $\mathcal{L}_{Sha}$ dimension is $t' + t = O(t)$, Babai approximation factor $\gamma_{CVP} = 2^{O(t)}$, and hence the subshare combiner parameters increase as follows: $\Gamma_{CVP} = O(t)$, $\delta_F = O(1/t \log t) = o(1)$, noise bitlength fraction $\alpha = 1 - \frac{1+o(1)}{t'/t}$ approaches $1 - t/t' = 1 - 1/c_1$, and noise bound $H \approx p^{1 - \frac{1+o(1)}{t'/t}}$ approaches $p^{1-t/t'} = p^{1-1/c_1}$. The subshare combiner result Theorem 4.1 holds for secrets of lengths at least $k'_0 = O(t)$, while the security result Theorem 4.2 holds for secrets of lengths at least $k_0 = O(t^2 \log t)$, so both requirements can be satisfied by our choice $k = c_3 t^2 \log t$ with a suitable choice for c_3, giving an entropy loss bound $\epsilon_s = O(t \log t)$ for up to $t_s = \lfloor (t' - c_1)/(1 + O(1/t \log t)) \rfloor$ which approaches $\lfloor t' - c_1 \rfloor$ for large t, while the fraction of secret entropy lost $\epsilon_s/k = O(1/t)$ approaches zero for large t.

4.4 Proof of Correctness

In this section we present a proof of our correctness result (Theorem 4.1).

Let us fix a subshare subset $I \subseteq [n]$ with $\#I = t'$. As explained in Section 4.2, we know by construction of lattice $\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I)$ in the subshare combiner algorithm, that the dealer's secret polynomial $a(x) = s + a_1 x + \ldots + a_{t-1}x^{t-1} \in \mathbb{Z}_p[x; t-1]$ gives rise to a lattice vector having its first t' coordinates congruent modulo p to the values of the secret polynomial $x \cdot a(x)$ at the points $\alpha_{i_1}, \ldots, \alpha_{i_{t'}}$, and the last t coordinates equal to the scaled coefficients of $a(x)$. Namely, we have

$$\boldsymbol{a'} = \left(\alpha_{i_1} a(\alpha_{i_1}) - k_1 p, \ldots, \alpha_{i_{t'}} a(\alpha_{i_{t'}}) - k_{t'} p, \frac{s}{p}H, \frac{a_1}{p}H, \ldots, \frac{a_{t-1}}{p}H\right),$$

which is "close" to the target vector

$$\boldsymbol{t'} = \left(\alpha_{i_1} a(\alpha_{i_1}) - k_1 p + r_{i_1}, \ldots, \alpha_{i_{t'}} a(\alpha_{i_{t'}}) - k_{t'} p + r_{i_{t'}}, 0, 0, \ldots, 0\right),$$

where, for $j = 1, \ldots, t'$, $k_j = \left\lfloor \frac{\alpha_{i_j} a(\alpha_{i_j}) + r_{i_j}}{p} \right\rfloor \in \mathbb{Z}$ is the multiple of p which should be subtracted from the integer $\alpha_{i_j} a(\alpha_{i_j}) + r_{i_j}$ to reduce it modulo p and obtain the value $s_{i_j} = \lfloor \alpha_{i_j} a(\alpha_{i_j}) + r_{i_j} \rfloor_p$ of the i_jth subshare. In particular we have, using $|r_{i_j}| < H$ for all $j = 1, \ldots, t'$, that $\|\boldsymbol{a'} - \boldsymbol{t'}\|_\infty < H$. Consequently, since $\mathsf{A_{CVP}}$ is a CVP approximation algorithm with $\|.\|_\infty$ approximation factor γ_{CVP}, its output lattice vector $\boldsymbol{c}$ will also be "close" to the target vector, namely we have $\|\boldsymbol{c} - \boldsymbol{t'}\|_\infty < \gamma_{CVP} \cdot H$. Applying the triangle inequality, we conclude that the lattice vector $\boldsymbol{z} = \boldsymbol{c} - \boldsymbol{a'}$ satisfies

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

Now, either $\frac{p}{H} c_{t'+1} \equiv \frac{p}{H} a'_{t'+1} \equiv s \pmod p$ in which case the combiner succeeds to recover secret s, or otherwise we have the following 'bad' case:

$$\frac{p}{H} z_{t'+1} = \frac{p}{H} c_{t'+1} - \frac{p}{H} a'_{t'+1} \not\equiv 0 \pmod p. \tag{3}$$

Hence, for fixed I, the combiner succeeds except for a fraction δ_I of 'bad' choices of $\boldsymbol{\alpha}_I \in D((\mathbb{Z}_p^*)^{t'})$, for which $\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I)$ contains a 'short' and 'bad' vector $\boldsymbol{z}$ satisfying (2) and (3). To upper bound δ_I, consider the polynomial $f(x) = \frac{p}{H} z_{t'+1} x + \cdots + \frac{p}{H} z_{t'+t} x^t$. Note that, since $\boldsymbol{z} \in \mathcal{L}_{Sha}$, we have $f(\alpha_{i_j}) \equiv z_j \pmod p$ and hence $\|f(\alpha_{i_j})\|_{L,p} < (\gamma_{CVP} + 1)H \leq 2^{\Gamma_{CVP}} H$ for all $j \in [t']$ using (2). Also, $\lfloor f(x) \rfloor_p$ has zero constant coefficient and degree at least 1 and at most t over $\mathbb{Z}_p$ using (3). Applying Lemma 2.1 (with parameters $n = t'$,$B = 2^{\Gamma_{CVP}} H$, $\#A \leq p^t$) we conclude that such a 'bad' polynomial f exists for at most a fraction $\delta_I \leq p^t (2Bt)^{t'} / \#D\left((\mathbb{Z}_p^*)^{t'}\right)$ of $\boldsymbol{\alpha}_I \in D\left((\mathbb{Z}_p^*)^{t'}\right)$, for each fixed I. Hence, the probability δ that a uniformly chosen $\boldsymbol{\alpha} \in D\left((\mathbb{Z}_p^*)^n\right)$ is 'bad' for *some* $I \subseteq [n]$ with $\#I = t'$ is upper bounded as

$$\delta \leq \frac{\binom{n}{t'} p^t (2Bt)^{t'}}{\#D\left((\mathbb{Z}_p^*)^{t'}\right)}. \tag{4}$$

A straightforward calculation detailed below shows that the right-hand side of (4) is upper bounded by δ_c for all k satisfying the inequality $k \geq k'_0$, where $k'_0 = \frac{t'/t}{t'/t-1}\left(\log\left(\delta_c^{-1/t'} nt\right) + \Gamma_{CVP} + 2\right)$, as claimed. To show the asymptotic correctness (for fixed n, t, t' and increasing security parameter k), note that with parameter choice $\delta_c = O(1/poly(k))$, we have $\delta_c^{-1/t'} = O(poly(k))$ so, for any fixed $\delta > 0$, we achieve δ-correctness whenever the conditions $\delta_c < \delta$ and $k \geq O(\log(knt) + \Gamma_{CVP} + 2)$ both hold. Recalling that Γ_{CVP} depends only on the lattice dimension $t' + t$ but not on k we have, since $\log(k) = o(k)$ that both of the latter conditions sufficient for achieving δ-correctness are satisfied for all sufficiently large k, as claimed.

We now show the remaining claim that the right-hand side of (4) is upper bounded by δ_c for all $k \geq k_0'$, with $k_0' = \frac{t'/t}{t'/t-1}\left(\log\left(\delta_c^{-1/t'}nt\right) + \Gamma_{CVP} + 2\right)$.

First, observe that $k \geq k_0'$ implies $p^\alpha/2 \geq 1$. Indeed, using $p \geq 2^k$, the condition $p^\alpha/2 \geq 1$ is implied by the condition $k \cdot \alpha \geq 1$. Plugging in the parameter choices $\alpha = 1 - \frac{1+\delta_F}{t'/t}$ and $\delta_F = \frac{t'/t}{k}(\log\left(\delta_c^{-1/t'}nt\right) + \Gamma_{CVP} + 1)$ we find that $k \cdot \alpha \geq 1$ is equivalent to the condition $k \geq k_0'$ as claimed.

The condition that the right-hand side of (4) is upper bounded by δ_c can be written as

$$M \cdot \frac{\binom{n}{t'}p^t(2Bt)^{t'}}{p^{t'}} \leq \delta_c, \tag{5}$$

where $M = p^{t'}/\#D\left((\mathbb{Z}_p^*)^{t'}\right)$. Rearranging this condition, and plugging in $B = 2^{\Gamma_{CVP}}H$ gives the equivalent condition

$$H \leq \left(\frac{1}{2t2^{\Gamma_{CVP}}}\right)\left(\frac{\delta_c p^{t'-t}}{M\binom{n}{t'}}\right)^{1/t'}. \tag{6}$$

Using $H = \lfloor p^\alpha/2 \rfloor \leq p^\alpha/2$ and $\alpha = 1 - \frac{1+\delta_F}{t'/t}$, we see that (6) is implied by the condition

$$\delta_F \geq \frac{t'/t}{\log p}\left(\log t + \Gamma_{CVP} + \log\left(\left(\delta_c^{-1}\binom{n}{t'}M\right)^{1/t'}\right)\right). \tag{7}$$

Now observe that $k_0' \geq \log(n) + 1 \geq \log(2t')$ since $n \geq t'$, so $k \geq k_0'$ implies $p \geq 2^k \geq 2t'$ and hence $p - t' \geq p/2$. Using this bound we get, for $k \geq k_0'$ that

$$\log\left(M^{1/t'}\right) = \log\left(\left(\frac{p^{t'}}{(p-1)\cdots(p-t')}\right)^{1/t'}\right) \leq \log\left(\left(\frac{p^{t'}}{(p/2)^{t'}}\right)^{1/t'}\right) = 1. \tag{8}$$

Plugging (8) and the bounds $\log p \geq k$ and $\log\left(\binom{n}{t'}^{1/t'}\right) \leq \log\left((n^{t'})^{1/t'}\right) = \log n$ into (7) we get assuming $k \geq k_0'$ the sufficient condition

$$\delta_F \geq \frac{t'/t}{k}\left(\log\left(\delta_c^{-1/t'}nt\right) + \Gamma_{CVP} + 1\right),$$

which is satisfied by the parameter choice $\delta_F = \frac{t'/t}{k}\left(\log\left(\delta_c^{-1/t'}nt\right) + \Gamma_{CVP} + 1\right)$. This shows that the right-hand side of (4) is upper bounded by δ_c for all $k \geq k_0'$, as claimed, which completes the proof of the theorem. □

4.5 Proof of Security

This section contains a proof of our security result (Theorem 4.2).

Fix an observed subshare subset I of $[n]$ of size $\#I = t_s$ and observed subshare values $\boldsymbol{\mu} \in \mathbb{Z}_p^n$. Using the fact that the polynomial $a \in \mathbb{Z}_p[x; t-1]$ and the noise vector $\boldsymbol{r}_I \in (-H, H)^{t_s}$ are chosen uniformly at random, the conditional probability $P_{k,x}(s|\boldsymbol{s}_I = \boldsymbol{\mu}_I)$ of the secret taking the value $s \in \mathbb{Z}_p$ given that the observed subshare vector $\boldsymbol{s}_I$ takes the value $\boldsymbol{\mu}_I$ is given by:

$$P_{k,x}(s|\boldsymbol{s}_I = \boldsymbol{\mu}_I) = \frac{\#\{(a, \boldsymbol{r}_I) \in T : \alpha_{i_j}a(\alpha_{i_j}) + r_{i_j} \equiv \mu_{i_j} \pmod p \forall j \in [t_s] \text{ and } a(0) \equiv s \pmod p\}}{\#\{(a, \boldsymbol{r}_I) \in T : \alpha_{i_j}a(\alpha_{i_j}) + r_{i_j} \equiv \mu_{i_j} \pmod p \forall j \in [t_s]\}},$$

where $T = \mathbb{Z}_p[x; t-1] \times (-H, H)^{t_s}$. Since $p > 2H$, we know that for each $a \in \mathbb{Z}_p[x; t-1]$ there is at most one $\boldsymbol{r}_I \in (-H, H)^{t_s}$ such that $\alpha_{i_j} a(\alpha_{i_j}) + r_{i_j} \equiv \mu_{i_j} \pmod{p}$ for all $j \in [t_s]$. Therefore the above expression simplifies to

$$P_{k,x}(s|\boldsymbol{s}_I = \boldsymbol{\mu}_I) = \frac{\#S_{s,p}(\boldsymbol{\alpha}_I, t, p, H, \boldsymbol{\mu}_I)}{\#S_{0,1}(\boldsymbol{\alpha}_I, t, p, H, \boldsymbol{\mu}_I)}, \tag{9}$$

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

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

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 $\boldsymbol{\alpha}_I \in D((\mathbb{Z}_p^*)^{t_s})$ assuming $k \geq k_0$ (with t_s, δ_s and k_0 defined in the theorem statement). We then apply these bounds to (9) to get a bound $P_{k,x}(s|\boldsymbol{s}_I = \boldsymbol{\mu}_I) \leq 2^{\epsilon_s}/p$ for all s (with ϵ_s defined in the theorem statement) so that for fixed I, entropy loss is bounded as $L_{k,x}(\boldsymbol{\mu}_I) \leq \epsilon_s$, except for fraction δ_I of $\boldsymbol{\alpha}_I \in D((\mathbb{Z}_p^*)^{t_s})$. It then follows that $L_{k,x}(\boldsymbol{\mu}_I) \leq \epsilon_s$ 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 $\boldsymbol{\alpha} \in D((\mathbb{Z}_p^*)^n)$ assuming that $k \geq k_0$, which proves the theorem.

Reduction to Lattice Point Counting. We now derive the desired probabilistic upper and lower bounds on $\#S_{\widehat{s},\widehat{p}}$. As a first step, we reduce the problem to a lattice 'point-counting' problem. 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 algorithm) contained in a $(t_s + t)$-dimensional box of side length $2H$, centered on a certain non-lattice vector $\widehat{\boldsymbol{s}}_I$ (for improved readability, we have placed the proofs of the following and subsequent lemmas in separate appendices at the end of the paper).

Lemma 4.1. *Fix positive integers $(t, t_s, p, H, \widehat{p})$ such that $p \geq 2H$ and $\widehat{p}$ is a divisor of p. Let $\widehat{s} \in \mathbb{Z}_{\widehat{p}}$, $\boldsymbol{\alpha}_I = (\alpha_{i_1}, \ldots, \alpha_{i_{t_s}}) \in \mathbb{Z}_p^n$ and $\boldsymbol{\mu}_I = (\mu_{i_1}, \ldots, \mu_{i_{t_s}}) \in \mathbb{Z}_p^{t_s}$. Define $\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, t, p, H, \widehat{p})$ as the full-rank lattice in $\mathbb{Q}^{t_s+t}$ with basis consisting of the rows of the matrix*

$$M_{Sha}(\boldsymbol{\alpha}_I, t, p, H, \widehat{p}) = \begin{pmatrix} p & 0 & \ldots & 0 & 0 & 0 & \ldots & 0 \\ 0 & p & \ldots & 0 & 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & p & 0 & 0 & \ldots & 0 \\ \widehat{p}\alpha_{i_1} & \widehat{p}\alpha_{i_2} & \ldots & \widehat{p}\alpha_{i_{t_s}} & 2H/(p/\widehat{p}) & 0 & \ldots & 0 \\ \alpha_{i_1}^2 & \alpha_{i_2}^2 & \ldots & \alpha_{i_{t_s}}^2 & 0 & 2H/p & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \alpha_{i_1}^t & \alpha_{i_2}^t & \ldots & \alpha_{i_{t_s}}^t & 0 & 0 & \ldots & 2H/p \end{pmatrix},$$

and define the vector $\widehat{\boldsymbol{\mu}}_I \in \mathbb{Q}^{t_s+t}$ by

$$\widehat{\boldsymbol{\mu}}_I \stackrel{\text{def}}{=} \left(\mu_{i_1} - \widehat{s}\alpha_{i_1}, \ldots, \mu_{i_{t_s}} - \widehat{s}\alpha_{i_{t_s}}, H\left(1 - \frac{1+2\widehat{s}}{p}\right), H\left(1 - \frac{1}{p}\right), \ldots, H\left(1 - \frac{1}{p}\right)\right).$$

Then the sizes of the following two sets are equal:

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

and

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

Guiding Heuristics. Before we present our rigorous upper and lower bounds on the number $\#V_{\widehat{s},\widehat{p}}$

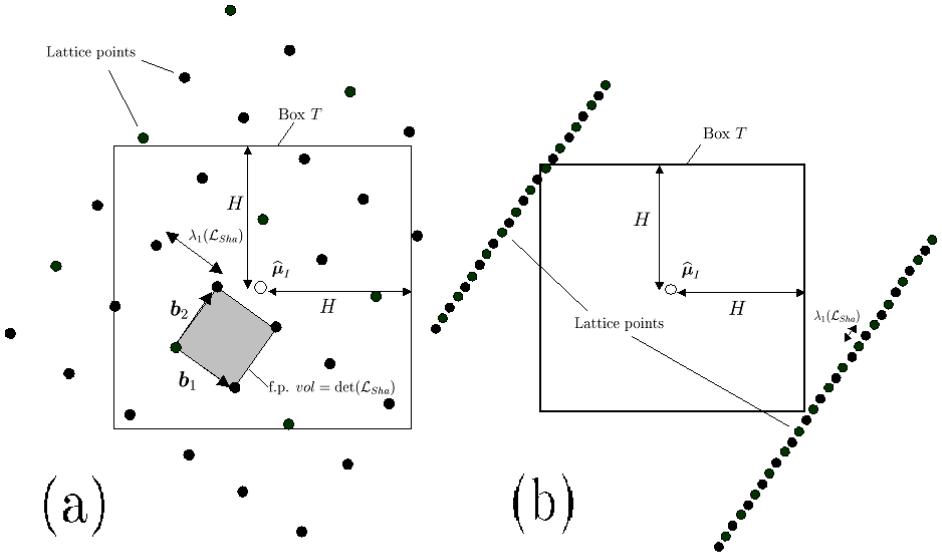

Figure 1: Geometric illustration of lattice point counting.

of lattice points in the box $T_{\widehat{\boldsymbol{\mu}}_I}(H) \stackrel{\text{def}}{=} \{\boldsymbol{v} \in \mathbb{Q}^{t_s+t} : \|\boldsymbol{v} - \widehat{\boldsymbol{\mu}}_I\|_\infty < H\}$, we explain some guiding geometric heuristics which our results make rigorous, in order to give more insight into the problem. As illustrated in Figure 1(a) for the two-dimensional case, it is well known that the determinant $\det(\mathcal{L})$ of a lattice $\mathcal{L}$ in $\mathbb{R}^{t_s+t}$ is the volume of any 'fundamental parallelepiped' (f.p.) of the lattice which tiles the space $\mathbb{R}^{t_s+t}$ (where for a lattice $\mathcal{L}$ with basis vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{t_s+t}$ we define the associated f.p. as the set $\sum_{i=1}^{t_s+t} c_i \boldsymbol{b}_i$ over all real-valued coefficients $c_1, \ldots, c_{t_s+t} \in [0, 1)$). Hence each f.p. is associated with a unique lattice point in $\mathcal{L}$ which serves as the origin of the f.p. So, given a box T in $\mathbb{R}^{t_s+t}$ having a volume $vol(T)$ which is sufficiently large compared to the f.p. volume $\det(\mathcal{L})$, we expect that the volume ratio $vol(T)/\det(\mathcal{L})$ would give a good estimate for the number of points of $\mathcal{L}$ contained in T. For our problem, the box $T_{\widehat{\boldsymbol{\mu}}_I}(H)$ has volume $vol(T_{\widehat{\boldsymbol{\mu}}_I}(H)) = (2H)^{t_s+t}$, compared to $\det(\mathcal{L}_{Sha}(\widehat{p}) = \widehat{p}(2H)^t p^{t_s-t}$, so according to the above heuristic, we expect that $\#V_{0,1} \approx \frac{(2H)^{t_s+t}}{(2H)^t p^{t_s-t}}$ (for the case $\widehat{p} = 1$) and $\#V_{s,p} \approx \frac{(2H)^{t_s+t}}{(2H)^t p^{t_s-(t-1)}}$ for $\widehat{p} = p$, which implies the desired security result $P_{k,x}(s|\boldsymbol{s}_I = \boldsymbol{\mu}_I) = \#V_{s,p}/\#V_{0,1} \approx 1/p$. The validity condition $vol(T_{\widehat{\boldsymbol{\mu}}_I}(H)) > \det(\mathcal{L}_{Sha}(p))$ is $(2H)^{t_s+t} > p(2H)^t p^{t_s-t}$, which implies, using $2H \approx p^\alpha \approx p^{1-\frac{1}{t'/t}}$, the expected bound $t_s < t' - t'/t$ on the number of observed shares.

Our proof below make the above heuristic estimates rigorous and gives quantitative bounds on the estimation errors involved. In particular, one central issue addressed by our proof (in Lemma 4.3) is upper bounding the probability of 'bad' instances of the lattice $\mathcal{L}_{Sha}$ having very 'unbalanced' f.p. due to the existence of short lattice vectors of norm $\lambda_1(\mathcal{L}_{Sha})$ much smaller than the Minkowski bound $\det(\mathcal{L}_{Sha})^{1/(t_s+t)}$. As illustrated in Fig 1(b), for such 'bad' lattices the number $\#V_{0,1}$ of lattice points in the box $T_{\widehat{\boldsymbol{\mu}}_I}(H)$ could be much smaller than the heuristic estimate $vol(T_{\widehat{\boldsymbol{\mu}}_I}(H))/\det(\mathcal{L}_{Sha})$, but fortunately, we show that the probability of such 'bad' instances of the lattice $\mathcal{L}_{Sha}$ is small when the α_i's are chosen at random.

We now proceed to present our rigorous probabilistic lower and upper bounds on $\#V_{\widehat{s},\widehat{p}}$.

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 $T_{\widehat{\boldsymbol{\mu}}_I}(H) \stackrel{\text{def}}{=} \{\boldsymbol{v} \in \mathbb{Q}^{t_s+t} : \|\boldsymbol{v} - \widehat{\boldsymbol{\mu}}_I\|_\infty < H\}$ centered on vector $\widehat{\boldsymbol{\mu}}_I$ seems a difficult 'non-homogenous' problem because $\widehat{\boldsymbol{\mu}}_I$ is in general not a lattice vector. But by 'rounding' $\widehat{\boldsymbol{\mu}}_I$ to a nearby lattice vector $\widehat{\boldsymbol{\mu}}'_I$ (with rounding error $\epsilon = \|\widehat{\boldsymbol{\mu}}'_I - \widehat{\boldsymbol{\mu}}_I\|_\infty$), 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 $T_{\mathbf{0}} \stackrel{\text{def}}{=} \{\boldsymbol{v} \in \mathbb{Q}^{t_s+t} : \|\boldsymbol{v}\|_\infty < H - \epsilon\}$, 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 4.2. *For any full-rank lattice $\mathcal{L}$ in $\mathbb{R}^n$, vector $\boldsymbol{\mu} \in \mathbb{R}^n$, and $H > 0$, we have*

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

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 2.2 in Section 2). To solve the problem (2) above of upper bounding $\lambda_{t_s+t}(\mathcal{L}_{Sha})$, we apply Minkowski's "second theorem" (Theorem 2.3 in Section 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

$$\lambda_{t_s+t}(\mathcal{L}_{Sha}) \leq \frac{\det(\mathcal{L}_{Sha})}{\lambda_1(\mathcal{L}_{Sha})^{t_s+t-1}}. \tag{10}$$

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 2.1 (similar to the argument used in proving Theorem 4.1), to obtain the following result. Note that Lemma 4.3 would not make sense if the conditions (12) and (14) depended on the choice of $\boldsymbol{\alpha}_I$; however this is not the case since $\det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, \widehat{p})) = \widehat{p}p^{t_s-t}(2H)^t$ and $\det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I)) = p^{t_s-t+1}(2H)^{t-1}$ are fixed independent of $\boldsymbol{\alpha}_I$.

Lemma 4.3. *Fix positive integers $(t, t_s, p, H, \widehat{p})$ and a positive real number β, such that p is a prime satisfying*

$$p \geq \max(2H, 2t_s) \tag{11}$$

and $\widehat{p} \in \{1, p\}$. For each $\boldsymbol{\alpha}_I \in D\left((\mathbb{Z}_p^*)^{t_s}\right)$, let $\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, \widehat{p})$ denote the lattice in $\mathbb{Q}^{t_s+t}$ with basis matrix $M_{Sha}(\boldsymbol{\alpha}_I, \widehat{p})$ defined in Lemma 4.1, and let $\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I)$ denote the lattice in $\mathbb{Q}^{t_s+t-1}$ with basis matrix $M'_{Sha}(\boldsymbol{\alpha}_I)$ obtained from $M_{Sha}(\boldsymbol{\alpha}_I, \widehat{p})$ by removing the (t_s+1)th row and column.
In the case $\widehat{p} = 1$, if

$$1 \leq 2^{-\left(\beta+3+\frac{t_s \log t}{t_s+t}\right)} \det\left(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1)\right)^{\frac{1}{t_s+t}} \leq H \tag{12}$$

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

$$\lambda_1(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1)) \geq 2^{-\left(\beta+3+\frac{t_s \log t}{t_s+t}\right)} \det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1))^{\frac{1}{t_s+t}}. \tag{13}$$

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

$$1 \leq 2^{-\left(\beta+3+\frac{t_s \log t}{t_s+t-1}\right)} \det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))^{\frac{1}{t_s+t-1}} \leq H \tag{14}$$

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

$$\lambda_1(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I)) \geq \lambda_1(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, p)) \geq 2^{-\left(\beta+3+\frac{t_s \log t}{t_s+t-1}\right)} \det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))^{\frac{1}{t_s+t-1}}. \tag{15}$$

Combining the above results (for $(\widehat{s}, \widehat{p}) = (0, 1)$) we obtain the desired lower bound on $\#V_{0,1}$, subject to several conditions. The details follow.

Let us fix $\beta > 0$, whose actual value will be chosen later. We will say that $\boldsymbol{\alpha}_I \in D\left((\mathbb{Z}_p^*)^{t_s}\right)$ is *bad* if one of the bounds (13) or (15) does not hold for this $\boldsymbol{\alpha}_I$. According to Lemma 4.3, if conditions (11), (12) and (14) are satisfied by $(t, t_s, p, H, \widehat{p}, \beta)$, then the fraction δ_I of bad $\boldsymbol{\alpha}_I$ in $D((\mathbb{Z}_p^*)^{t_s})$ is upper bounded as

$$\delta_I \leq \delta_I(1) + \delta_I(p) \leq 2^{-\beta(t_s+t)} + 2^{-\beta(t_s+t-1)}. \tag{16}$$

Throughout the following derivation we assume that $\boldsymbol{\alpha}_I$ is not bad. Plugging the lower bound (13) in the inequality (10) resulting from Minkowski's Second Theorem gives

$$\lambda_{t_s+t}(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1)) \leq 2^{\left(\beta+3+\frac{t_s \log t}{t_s+t}\right)(t_s+t)} \cdot \det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1))^{\frac{1}{t_s+t}}. \tag{17}$$

Hence, applying Lemma 4.2 we have $\#V_{0,1} \geq \#\{\boldsymbol{v} \in \mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1) : \|\boldsymbol{v}\|_\infty < H - \epsilon\}$, where $\epsilon \leq (\frac{t_s+t}{2}) 2^{\left(\beta+3+\frac{t_s \log t}{t_s+t}\right)(t_s+t)} \cdot \det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1))^{\frac{1}{t_s+t}}$. So, if the condition

$$\left(\frac{t_s+t}{2}\right) 2^{\left(\beta+3+\frac{t_s \log t}{t_s+t}\right)(t_s+t)} \cdot \det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1))^{\frac{1}{t_s+t}} \leq \frac{H}{2} \tag{18}$$

holds, then

$$\#V_{0,1} \geq \#\{\boldsymbol{v} \in \mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1) : \|\boldsymbol{v}\|_\infty < H/2\} \geq 2 \cdot Int\left(\frac{(H/2)^{t_s+t}}{2^{t_s+t} \det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1))}\right) + 1, \tag{19}$$

where we have used the Blichfeldt-Corput Theorem 2.2 to obtain the last inequality. Using $2Int(z) + 1 \geq 2(z-1) + 1 \geq z$ for all $z \geq 1$, we find that (19) implies, assuming in addition

$$\frac{(H/2)^{t_s+t}}{2^{t_s+t-1} \det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1))} \geq 2, \tag{20}$$

that

$$\#V_{0,1} \geq \frac{H^{t_s+t}}{2^{2(t_s+t)} \det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1))}. \tag{21}$$

Observe that our assumed condition (20) is equivalent to $2 \det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1))^{\frac{1}{t_s+t}} \leq H/2$ and hence, recalling that $\beta > 0$, our condition (18) implies both conditions (20) and the right-hand side of (12). We conclude that the lower bound (21) holds assuming that $\boldsymbol{\alpha}_I$ is not bad, (11) holds, the left-hand side of (12) holds and (18) holds.

Finding an Upper Bound on $\#V_{s,p}$. We first reduce the point counting problem in $\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, p)$ to a point counting problem in the lower-dimensional lattice $\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I)$ defined in Lemma 4.3. This is possible because all the vectors of $\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, p)$ in the desired box have their (t_s+1)th coordinate equal to 0.

Lemma 4.4. *Let $\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, p) \subseteq \mathbb{Q}^{t_s+t}$ and $\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I) \subseteq \mathbb{Q}^{t_s+t-1}$ be the lattices defined in Lemma 4.3, let $\widehat{\boldsymbol{\mu}}_I$ be the vector in $\mathbb{Q}^{t_s+t}$ defined in Lemma 4.1, and let $\widehat{\boldsymbol{\mu}}'_I$ be the vector in $\mathbb{Q}^{t_s+t-1}$ obtained from $\widehat{\boldsymbol{\mu}}_I$ by removing the (t_s+1)th coordinate. Then $\#V_{s,p} \leq \#V'_{s,p}$, where $V_{s,p} \stackrel{\text{def}}{=} \{\boldsymbol{v} \in \mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, p) : \|\boldsymbol{v} - \widehat{\boldsymbol{\mu}}_I\|_\infty < H\}$ and $V'_{s,p} \stackrel{\text{def}}{=} \{\boldsymbol{v} \in \mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I) : \|\boldsymbol{v} - \widehat{\boldsymbol{\mu}}'_I\|_\infty < H\}$.*

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 $T_{\widehat{\boldsymbol{\mu}}'_I}(H) \stackrel{\text{def}}{=} \{\boldsymbol{v} \in \mathbb{Q}^{t_s+t-1} : \|\boldsymbol{v} - \widehat{\boldsymbol{\mu}}'_I\|_\infty < H\}$, to the volume of $\widehat{T}_{\widehat{\boldsymbol{\mu}}'_I}(H) \stackrel{\text{def}}{=} \{\boldsymbol{v} \in \mathbb{Q}^{t_s+t-1} : \|\boldsymbol{v} - \widehat{\boldsymbol{\mu}}'_I\|_\infty < H + \lambda_1(\mathcal{L}'_{Sha})/2\}$ 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 4.5. *For any lattice $\mathcal{L}$ in $\mathbb{R}^n$, vector $\boldsymbol{\mu} \in \mathbb{R}^n$, and $H > 0$, we have*

$$\#\{\boldsymbol{v} \in \mathcal{L} : \|\boldsymbol{v} - \boldsymbol{\mu}\|_\infty < H\} \le \left(\frac{2H}{\lambda_1(\mathcal{L})} + 1\right)^n.$$

So, combining Lemma 4.4 and Lemma 4.5 we have $\#V_{s,p} \le \#V'_{s,p} \le \left(\frac{2H}{\lambda_1(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))} + 1\right)^{t_s+t-1}$. Assuming that $\boldsymbol{\alpha}_I$ is not bad (using same definition of badness as above) and that conditions (11) and (14) hold, we can apply the lower bound (15) on $\lambda_1(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))$ to obtain the upper bound

$$\#V_{s,p} \le \left(\frac{2H}{2^{-\left(\beta+3+\frac{t_s \log t}{t_s+t-1}\right)} \det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))^{\frac{1}{t_s+t-1}}} + 1\right)^{t_s+t-1}. \tag{22}$$

Using $z + 1 \le 2z$ for $z \ge 1$, we find that (22) implies, assuming in addition

$$\frac{2H}{2^{-\left(\beta+3+\frac{t_s \log t}{t_s+t-1}\right)} \det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))^{\frac{1}{t_s+t-1}}} \ge 1 \tag{23}$$

that

$$\#V_{s,p} \le \left(\frac{2^2 H}{2^{-\left(\beta+3+\frac{t_s \log t}{t_s+t-1}\right)} \det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))^{\frac{1}{t_s+t-1}}}\right)^{t_s+t-1} \le 2^{(\beta+5)(t_s+t)+t_s \log t} \cdot \frac{H^{t_s+t-1}}{\det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))}. \tag{24}$$

Observe that our assumed condition (23) is implied by the right-hand side of (14). We conclude that the upper bound (24) holds assuming that $\boldsymbol{\alpha}_I$ is not bad and that conditions (11) and (14) hold.

Putting it Together. We now put together the above results. Let us assume that the parameters (t, t_s, p, H, β) satisfy all the sufficient conditions specified above for the bounds (21) and (24) to hold for non-bad choices of $\boldsymbol{\alpha}_I$; namely the assumed conditions are (11), left-hand side of (12), (18), and (14) (below we will show that all these conditions are satisfied if $k \ge k_0$ and k_0, t_s and β are chosen as in the theorem statement). Then, for each fixed $I \subseteq [n]$ with $\#I = t_s$, for all except a fraction δ_I of bad $\boldsymbol{\alpha}_I \in D((\mathbb{Z}_p^*)^{t_s})$ (with δ_I upper bounded in (16)), plugging the bounds (21) and (24) in (9) (using $\#V_{s,p} = \#S_{s,p}$ and $\#V_{0,1} = \#S_{0,1}$ by Lemma 4.1) we find

$$P_{k,x}(s|\boldsymbol{s}_I = \boldsymbol{\mu}_I) \le \frac{2^{(\beta+5)(t_s+t)+t_s \log t} \cdot H^{t_s+t-1} / \det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))}{2^{-2(t_s+t)} \cdot H^{t_s+t} / \det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1))} \le 2^{(\beta+7)(t_s+t)+t_s \log t+1} p^{-1}$$

for all $s \in \mathbb{Z}_p$ and $\boldsymbol{\mu}_I \in (\mathbb{Z}_p)^{t_s}$ (where we have used $\det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, \widehat{p})) = \widehat{p} p^{t_s-t} (2H)^t$ and $\det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I)) = p^{t_s-t+1}(2H)^{t-1}$) and hence conditional entropy is bounded as $H(s|\boldsymbol{s}_I = \boldsymbol{\mu}_I) \ge \log(p/2^{(\beta+7)(t_s+t)+t_s \log t+1})$ and entropy loss is bounded as

$$L_{k,x}(\boldsymbol{\mu}_I) = |\log p - H(s|\boldsymbol{s}_I = \boldsymbol{\mu}_I)| \le (\beta + 7)(t_s + t) + t_s \log t + 1 = \epsilon_s \tag{25}$$

for all $\boldsymbol{\mu}_I \in (\mathbb{Z}_p)^{t_s}$, as claimed in the theorem statement. From (16), the bound (25) holds for each fixed I for all except a fraction $\delta_I \le 2^{-\beta(t_s+t)} + 2^{-\beta(t_s+t-1)}$ of bad $\boldsymbol{\alpha}_I \in D((\mathbb{Z}_p^*)^{t_s})$. Hence, for each fixed I, (25) holds except with probability at most $2^{-\beta(t_s+t)} + 2^{-\beta(t_s+t-1)}$ over the uniformly random choice of $\boldsymbol{\alpha} \in D((\mathbb{Z}_p^*)^n)$. Finally, by the union bound it follows that (25) holds *for all* $I \subseteq [n]$ with $\#I = t_s$ except with probability at most

$$\delta_s = \binom{n}{t_s}\left(2^{-\beta(t_s+t)} + 2^{-\beta(t_s+t-1)}\right) \le \binom{n}{t_s} 2^{1-\beta(t_s+t-1)} = \delta_c$$

over the uniform choice of $\boldsymbol{\alpha} \in D((\mathbb{Z}_p^*)^n)$, as claimed in the theorem statement, assuming we set the parameter β to the value

$$\beta = \frac{\log\left(2\delta_c^{-1}\binom{n}{t_s}\right)}{t_s + t - 1} \tag{26}$$

as defined in the theorem statement.

It remains to show that the conditions assumed above, namely (11), left inequality of (12), (18), and (14), are all satisfied if $k \geq k_0$ and k_0, t_s and β are chosen as in the theorem statement.

First, we note that (11) is satisfied. This is because by definition we have $2H = 2\lfloor p^\alpha/2 \rfloor \leq p$ since $\alpha < 1$, and also $p \geq 2^k \geq 2^{k_0} \geq 2t_s$, so $p \geq \max(2H, 2t_s)$ as claimed.

Next, we show that the left inequalities of both (12) and (14) are satisfied. Recall that $\det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1)) = (2H)^t p^{t_s - t}$ and $\det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I)) = (2H)^{t-1} p^{t_s - t + 1}$. Thus the left inequality of (12) is equivalent to $(2H)^t p^{t_s - t} \geq 2^{(\beta+3)(t_s+t)+t_s \log t}$ while the left inequality of (14) is equivalent to $(2H)^{t-1} p^{t_s-t+1} \geq 2^{(\beta+3)(t_s+t-1)+t_s \log t}$. So using (11) and $\beta > 0$ we know that the left inequality of (12) implies the left inequality of (14). So it suffices to show that

$$(2H)^t p^{t_s - t} \geq 2^{(\beta+3)(t_s+t)+t_s \log t}. \tag{27}$$

To do so, note that we may assume that $t_s \geq t$ (since for $t_s < t$, the scheme ShaTSS' clearly has perfect security thanks to the perfect security of the original (t, n)-threshold Shamir scheme ShaTSS). Thus $p^{t_s - t} \geq 1$ and (27) is implied by

$$(2H)^t \geq 2^{(\beta+3)(t_s+t)+t_s \log t}. \tag{28}$$

Using $t_s \leq t'$, (28) is satisfied if

$$2H \geq 2^{(\beta+3)(t'/t+1)+t'/t \log t}. \tag{29}$$

Now, by definition we have $2H \geq 2\lfloor \frac{p^\alpha}{2} \rfloor \geq p^\alpha - 2$ so (29) is satisfied if

$$p^\alpha \geq 2^{(\beta+3)(t'/t+1)+t'/t \log t} + 2. \tag{30}$$

Since $\beta > 0$ and $t'/t > 1$ we have $(\beta + 3)(t'/t + 1) + t'/t \log t \geq 6$, and using $z + 2 \leq 2z$ for $z \geq 6$ and $p \geq 2^k$, we know that (30) is satisfied if

$$2^{\alpha k} \geq 2^{(\beta+3)(t'/t+1)+t'/t \log t + 1}. \tag{31}$$

Using the definitions $\alpha = 1 - \frac{1+\delta_F}{t'/t}$ and $\delta_F = \frac{t'/t}{k}\left(\log\left(\delta_c^{-1/t'} nt\right) + \Gamma_{CVP} + 1\right)$, we get by straightforward manipulation that (31) is equivalent to the condition

$$k \geq \frac{t'/t}{t'/t - 1} \cdot \left(\log\left(\delta_c^{-1/t'} nt\right) + \Gamma_{CVP} + (\beta + 3)(t'/t + 1) + t'/t \log t + 2\right). \tag{32}$$

Recalling that (see statement of Theorem 4.1) $k'_0 = \frac{t'/t}{t'/t-1}\left(\log\left(\delta_c^{-1/t'} nt\right) + \Gamma_{CVP} + 2\right)$, we see that the right-hand side of (32) is equal to $k'_0 + \frac{t'/t}{t'/t-1}((\beta + 3)(t'/t + 1) + t'/t \log t)$ which is less than $k_0 \geq k'_0 + \frac{(t'/t+1)^2}{t'/t-1}(\beta + \log t + 3)$, so $k \geq k_0$ implies (32), and hence the left inequalities of both (12) and (14) are satisfied, as claimed.

We now show that the right inequality of (14) implies (18). Recalling that $\det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1)) = \frac{2H}{p} \det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha_I}))$ we see that the right inequality of (14) implies (18) as long as

$$2^{-\left(\left(\left(\beta+3+\frac{t_s \log t}{t_s+t}\right)(t_s+t)+\log(t_s+t)\right)(t_s+t)+1\right)} p \geq 2^{\left(\beta+3+\frac{t_s \log t}{t_s+t-1}\right)(t_s+t-1)}. \tag{33}$$

Using $p \geq 2^k$ we find that (33) is implied by the condition

$$k \geq (\beta+3)(m^2+m-1) + m(t_s \log t + \log m) + t_s \log t + 1, \tag{34}$$

where $m = t_s + t$, and condition (34) holds by the theorem hypothesis $k \geq k_0$. Hence the right inequality of (14) implies (18), as claimed.

Finally, we show that that the right inequality of (14) is satisfied. Recalling that $\det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I) = (2H)^{t-1}p^{t_s-(t-1)}$, we see that the right inequality of (14) is equivalent to

$$p^{t_s-(t-1)} \leq 2^{\left(\beta+3+\frac{t_s \log t}{t_s+t-1}\right)(t_s+t-1)-(t-1)} \cdot H^{t_s}. \tag{35}$$

Now, we know that $p^\alpha/2 \geq 2$ because using $p \geq 2^k$ and the definitions $\alpha = 1 - \frac{1+\delta_F}{t/t}$ and $\delta_F = \frac{t'/t}{k}\left(\log\left(\delta_c^{-1/t'}nt\right) + \Gamma_{CVP} + 1\right)$ we see that the condition $p^\alpha/2 \geq 2$ is satisfied if $k \geq \frac{t'/t}{t'/t-1}\left(\log\left(\delta_c^{-1/t'}nt\right) + \Gamma_{CVP} + 3\right) = k'_0 + \frac{t'/t}{t'/t-1}$, and the latter condition is satisfied since $k \geq k_0$ and k_0 exceeds by definition the value $k'_0 + \frac{t'/t}{t'/t-1}$.

Using $p^\alpha/2 \geq 2$ we have that $H \geq \lfloor p^\alpha/2 \rfloor \geq p^\alpha/4$. Using this we have that (35) is satisfied if

$$p^{t_s-(t-1)} \leq 2^{\left(\beta+3+\frac{t_s \log t}{t_s+t-1}\right)(t_s+t-1)-(t-1)} \cdot \left(\frac{p^\alpha}{2^2}\right)^{t_s}. \tag{36}$$

Using the definitions $\alpha = 1 - \frac{1+\delta_F}{t'/t}$ and $\delta_F = \frac{t'/t}{k}\left(\log\left(\delta_c^{-1/t'}nt\right) + \Gamma_{CVP} + 1\right)$, we get that (36) is equivalent to the condition

$$t_s \leq \frac{(t'-t'/t) + \frac{t'/t}{\log p}\left(\left(\beta+3+\frac{t_s \log t}{t_s+t-1}\right)(t_s+t-1) - (2t_s+t-1)\right)}{1 + \frac{t'/t}{k}\left(\log\left(\delta_c^{-1/t'}nt\right) + \Gamma_{CVP} + 1\right)}. \tag{37}$$

But since $3(t_s+t-1) - (2t_s+t-1) = t_s + 2(t-1) > 0$, we have that the term $\frac{t'/t}{\log p}\left(\left(\beta+3+\frac{t_s \log t}{t_s+t-1}\right)(t_s+t-1) - (2t_s+t-1)\right)$ is positive and hence (37) is satisfied by the choice

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

of the theorem statement.

Therefore the right inequality of (14) is satisfied. This completes the proof of the concrete claims of the theorem. To show the asymptotic security claim (for fixed n, t, t' and increasing security parameter k) with parameter choice $\delta_c = O(1/poly(k))$, observe that with this choice $\delta_s = \delta_c = o(1)$ and $\beta = \frac{\log\left(2\delta_c^{-1}\binom{n}{t_s}\right)}{t_s+t-1} = O(\log k)$, so fractional entropy loss $\epsilon_s/k = O(\log k/k) = o(1)$, and (recalling that Γ_{CVP} is independent of k), we have $t_s = \lfloor (t'-t'/t)/(1+O(\log k/k)) \rfloor$ so $t_s = Int(t'-t'/t)$ for all sufficiently large k, as claimed. This completes the proof of the theorem. □

An immediate consequence of the above results is the following.

Corollary 4.1. *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.

Our scheme raises several open problems.

Firstly, our security result is proven to hold only for sufficiently long security parameters $k \geq k_0 = \Omega(t^2 \log t)$ whereas the correctness of the scheme holds for much smaller security parameters $k \geq k'_0 = \Omega(t)$. Therefore, in order to improve the practicality of the security result, an interesting problem is to find an improved security proof which decreases the bound k_0 to $\Omega(t)$, while at the same time decreasing the bound ϵ_s on the leaked secret entropy to be $o(t)$. A related open problem mentioned above is to reduce the bound ϵ_s on the leaked secret entropy.

Secondly, in our security analysis we have assumed a passive attacker which is assumed to be an outsider. In some cases, stronger security may be needed. For example, the threshold may be increased only after several shareholders have already been compromised by the attacker. Against such 'insider attackers' who know some original shares, the information-theoretic threshold would be reduced below the desired value t'. In our scheme, we expect the effective information-theoretic threshold to be reduced to about $t' - (t'/t) \cdot (s+1)$ against insider attackers knowing s original shares, but we leave for future work a rigorous analysis of this scenario. If private shareholder communication is allowed, a known method of dealing with this problem is to use a *share renewal* protocol [12]. It would be interesting to design an efficient scheme which maintains its new threshold in this insider attacker scenario with no shareholder communication (perhaps under a suitable computational complexity assumption). Another security issue not handled by our scheme is 'active attacks' by insiders who send corrupted subshares to the combiner to prevent recovery of the secret ('denial of service' attack). One way to deal with this is to have more than t' subshares sent to the combiner and use an error correction algorithm that can correct the additional 'Hamming' noise due to corrupted subshares. Efficiently implementing such a combiner for our scheme is another potential area for future research.

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 DP0663452 and DP0451484 and DP0344444.

References

[1] C. Asmuth and J. Bloom. A Modular Approach to Key Safeguarding. *IEEE Trans. on Information Theory*, 29:208–210, 1983.

[2] L. Babai. On Lovász' Lattice Reduction and the Nearest Lattice Point Problem. *Combinatorica*, 6, 1986.

[3] S.G. Barwick, W.A. Jackson, and K.M. Martin. Updating the Parameters of a Threshold Scheme by Minimal Broadcast. *IEEE Trans. on Information Theory*, 51:620–633, 2005.

[4] C.H. Bennett, G. Brassard, C. Crépau, and U.M. Maurer. Generalized Privacy Amplification. *IEEE Trans. on Information Theory*, 41:1915–1923, 1995.

[5] 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.

[6] R.M. Capocelli, A. De Santis, L. Gargano, and U. Vaccaro. A Note on Secret Sharing Schemes. In *Sequences II: Methods in Communications, Security, and Computer Science*, pages 335–344, Berlin, 1993. Springer-Verlag.

[7] 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.

[8] O. Goldreich, D. Ron, and M. Sudan. Chinese Remaindering with Errors. *IEEE Transactions on Information Theory*, 46:1330–1338, 2000.

[9] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer-Verlag, 1993.

[10] P. Gruber and C. Lekkerkerker. *Geometry of Numbers*. Elsevier Science Publishers, 1987.

[11] V. Guruswami and M. Sudan. Improved Decoding of Reed-Solomon Codes and Algebraic-Geometric Codes. *IEEE Trans. Inf. Th.*, 45:1757–1767, Sep. 1999.

[12] A. Herzberg, S. Jarecki, H. Krawczyk, and M. Yung. Proactive Secret Sharing Or: How to Cope With Perpetual Leakage. In *CRYPTO '95*, volume 963 of *LNCS*, pages 339–352, Berlin, 1995. Springer-Verlag.

[13] E. Hlawka, J. Schoißengeier, and R. Taschner. *Geometric and Analytic Number Theory*. Springer-Verlag, 1991.

[14] W.A. Jackson and K.M. Martin. A Combinatorial Interpretation of Ramp Schemes. *Australasian Journal of Combinatorics*, 14:51–60, 1996.

[15] A. K. Lenstra, H. W. Lenstra, and L. Lovász. Factoring Polynomials with Rational Coefficients. *Mathematische Annalen*, 261:515–534, 1982.

[16] 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.

[17] K. Martin. Untrustworthy Participants in Secret Sharing Schemes. In *Cryptography and Coding III*, pages 255–264. Oxford University Press, 1993.

[18] 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.

[19] K. Martin, R. Safavi-Naini, and H. Wang. Bounds and Techniques for Efficient Redistribution of Secret Shares to New Access Structures. *The Computer Journal*, 42:638–649, 1999.

[20] R.J. McEliece and D.V. Sarwate. On Sharing Secrets and Reed-Solomon Codes. *Comm. of the ACM*, 24:583–584, 1981.

[21] 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.

[22] A. Shamir. How To Share a Secret. *Comm. of the ACM*, 22:612–613, 1979.

[23] M.A. Shokrollahi and H. Wasserman. List Decoding of Algebraic-Geometric Codes. *IEEE Transactions on Information Theory*, 45:432–437, March 1999.

[24] I.E. Shparlinski. Sparse Polynomial Approximation in Finite Fields. In *Proc. 33rd STOC*, pages 209–215, New York, 2001. ACM Press.

[25] I.E. Shparlinski and R. Steinfeld. Noisy Chinese Remaindering in the Lee Norm. *Journal of Complexity*, 20:423–437, 2004.

[26] R. Steinfeld, J. Pieprzyk, and H. Wang. Dealer-Free Threshold Changeability for Standard CRT Secret-Sharing Schemes. *Finite Fields and their Applications*, 12:653–680, 2006.

A Proof of Lemma 4.1

We define a mapping $f : V_{\widehat{s},\widehat{p}} \to S_{\widehat{s},\widehat{p}}$ and show that f is one to one and onto.

Defining f. Let $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{t_s+t}$ denote the basis for $\mathcal{L}_{Sha}$ where $\boldsymbol{b}_i$ is the ith row of matrix M_{Sha}. For each $\boldsymbol{v} \in V_{\widehat{s},\widehat{p}}$, let $(k_1^{\boldsymbol{v}}, \ldots, k_{t_s}^{\boldsymbol{v}}, k^{\boldsymbol{v}}, a_1^{\boldsymbol{v}}, \ldots, a_{t-1}^{\boldsymbol{v}}) \in \mathbb{Z}^{t_s+t}$ denote the expansion coefficient vector for $\boldsymbol{v}$ in basis $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{t_s+t}$, i.e. $\boldsymbol{v} = k_1^{\boldsymbol{v}}\boldsymbol{b}_1 + \ldots + k_{t_s}^{\boldsymbol{v}}\boldsymbol{b}_{t_s} + k^{\boldsymbol{v}}\boldsymbol{b}_{t_s+1} + a_1^{\boldsymbol{v}}\boldsymbol{b}_{t_s+2} + \ldots + a_{t-1}^{\boldsymbol{v}}\boldsymbol{b}_{t_s+t}$. To each such $\boldsymbol{v} \in V_{\widehat{s},\widehat{p}}$, we associate the polynomial $a_{\boldsymbol{v}} = f(\boldsymbol{v})$ defined by

$$a_{\boldsymbol{v}} = f(\boldsymbol{v}) = \lfloor \widehat{s} + k^{\boldsymbol{v}}\widehat{p} \rceil_p + \lfloor a_1^{\boldsymbol{v}} \rceil_p x + \ldots + \lfloor a_{t-1}^{\boldsymbol{v}} \rceil_p x^{t-1} \in \mathbb{Z}_p[x; t-1].$$

To show that f indeed maps $V_{\widehat{s},\widehat{p}}$ into $S_{\widehat{s},\widehat{p}}$, note that $\boldsymbol{v} \in V_{\widehat{s},\widehat{p}}$ implies that $\|\alpha_{i_j} a_{\boldsymbol{v}}(\alpha_{i_j}) - \mu_{i_j}\|_{L,p} < H$ for all $j \in [t_s]$, and since $a_{\boldsymbol{v}}(0) \equiv \widehat{s} + k^{\boldsymbol{v}}\widehat{p} \pmod{p}$ we have from the fact that $p \equiv 0 \pmod{\widehat{p}}$ that $a_{\boldsymbol{v}}(0) \equiv \widehat{s} \pmod{\widehat{p}}$. Thus $a_{\boldsymbol{v}} = f(\boldsymbol{v}) \in S_{\widehat{s},\widehat{p}}$ for all $\boldsymbol{v} \in V_{\widehat{s},\widehat{p}}$, as required.

Showing that f is one to one. Suppose that $f(\boldsymbol{v}) = f(\boldsymbol{w})$ for some pair of vectors $\boldsymbol{v} = k_1^{\boldsymbol{v}}\boldsymbol{b}_1 + \ldots k_{t_s}^{\boldsymbol{v}}\boldsymbol{b}_{t_s} + k^{\boldsymbol{v}}\boldsymbol{b}_{t_s+1} + a_1^{\boldsymbol{v}}\boldsymbol{b}_{t_s+2} + \ldots + a_{t-1}^{\boldsymbol{v}}\boldsymbol{b}_{t_s+t}$ and $\boldsymbol{w} = k_1^{\boldsymbol{w}}\boldsymbol{b}_1 + \ldots k_{t_s}^{\boldsymbol{w}}\boldsymbol{b}_{t_s} + k^{\boldsymbol{w}}\boldsymbol{b}_{t_s+1} + a_1^{\boldsymbol{w}}\boldsymbol{b}_{t_s+2} + \ldots + a_{t-1}^{\boldsymbol{w}}\boldsymbol{b}_{t_s+t}$ in $V_{\widehat{s},\widehat{p}}$, and define the associated polynomials

$$a_{\boldsymbol{v}}(x) = k^{\boldsymbol{v}}\widehat{p} + a_1^{\boldsymbol{v}}x + \ldots + a_{t-1}^{\boldsymbol{v}}x^{t-1}$$

and

$$a_{\boldsymbol{w}}(x) = k^{\boldsymbol{w}}\widehat{p} + a_1^{\boldsymbol{w}}x + \ldots + a_{t-1}^{\boldsymbol{w}}x^{t-1}.$$

Then $f(\boldsymbol{v}) = f(\boldsymbol{w})$ implies that

$$a_{\boldsymbol{v}}(x) \equiv a_{\boldsymbol{w}}(x) \pmod{p}. \tag{38}$$

But, for each $j \in [t_s]$, we know that the jth coordinates satisfy $v_j \equiv \alpha_{i_j} a_{\boldsymbol{v}}(\alpha_{i_j}) \pmod{p}$ and $w_j \equiv \alpha_{i_j} a_{\boldsymbol{w}}(\alpha_{i_j}) \pmod{p}$. It follows from (38) that v_j and w_j differ by an integer multiple of p for all $j \in [t_s]$ while the fact that $\boldsymbol{v}$ and $\boldsymbol{w}$ are in $V_{\widehat{s},\widehat{p}}$ implies $\|\boldsymbol{v} - \boldsymbol{w}\|_\infty \le \|\boldsymbol{v} - \widehat{\boldsymbol{\mu}}_I\|_\infty + \|\boldsymbol{w} - \widehat{\boldsymbol{\mu}}_I\|_\infty < 2H \le p$ so we conclude $v_j = w_j$ over the integers. Similarly, for $j \ge t_s + 1$, (38) implies that v_j and w_j differ by an integer multiple of $2H$, so $\|\boldsymbol{v} - \boldsymbol{w}\|_\infty < 2H$ implies $v_j = w_j$. This completes the proof that $f(\boldsymbol{v}) = f(\boldsymbol{w})$ implies $\boldsymbol{v} = \boldsymbol{w}$, so f is one to one.

Showing that f is onto. Let $a(x) = \widehat{s} + \widehat{k}\widehat{p} + a_1 x + \ldots + a_{t-1}x^{t-1}$ be any polynomial in $S_{\widehat{s},\widehat{p}}$. We construct a vector $\boldsymbol{v} \in V_{\widehat{s},\widehat{p}}$ such that $f(\boldsymbol{v}) = a$ as follows. We set

$$v_j = \alpha_{i_j}(a(\alpha_{i_j}) - \widehat{s}) + k_j p = \widehat{k}\widehat{p}\alpha_{i_j} + a_1\alpha_{i_j}^2 + \cdots + a_{t-1}\alpha_{i_j}^t + k_j p \quad \text{for all } j \in [t_s],$$

where k_j is an integer chosen such that $\left|v_j - \widehat{\mu}_{I_j}\right| < H$. It is clear that such an integer k_j exists because using $\widehat{\mu}_{I_j} = \mu_{i_j} - \widehat{s}\alpha_{i_j}$ we have $\left|v_j - \widehat{\mu}_{I_j}\right| = \left|\alpha_{i_j}(a(\alpha_{i_j}) - \widehat{s}) + k_j p - (\mu_{i_j} - \widehat{s}\alpha_{i_j})\right| = \left|\alpha_{i_j} a(\alpha_{i_j}) - \mu_{i_j} + k_j p\right|$ and the latter is less than H for suitable integer k_j because $a \in S_{\widehat{s},\widehat{p}}$ satisfies (by definition of $S_{\widehat{s},\widehat{p}}$), the condition $\|\alpha_{i_j} a(\alpha_{i_j}) - \mu_{i_j}\|_{L,p} < H$ for all $j \in [t_s]$. To complete the definition of vector $\boldsymbol{v}$, we set

$$v_{t_s+1} = \frac{2H}{p/\widehat{p}}\widehat{k} \quad \text{and} \quad v_j = \frac{2H}{p} a_{j-(t_s+1)} \text{ for } j > t_s + 1.$$

Observe that $\boldsymbol{v}$ is in lattice $\mathcal{L}_{Sha}$ and in fact has the basis expansion $\boldsymbol{v} = k_1\boldsymbol{b}_1 + \cdots k_{t_s}\boldsymbol{b}_{t_s} + \widehat{k}\boldsymbol{b}_{t_s+1} + a_1\boldsymbol{b}_{t_s+2} + \cdots + a_{t-1}\boldsymbol{b}_{t_s+t}$, so by definition of the mapping f we have

$$f(\boldsymbol{v}) = \lfloor \widehat{s} + \widehat{k}\widehat{p} \rceil_p + \lfloor a_1 \rceil_p x + \ldots + \lfloor a_{t-1} \rceil_p x^{t-1} = a(x),$$

because the coefficients $\widehat{s} + \widehat{k}\widehat{p}$ and $a_1, \ldots, a_{t-1}$ of $a(x)$ are in $\mathbb{Z}_p$. It remains to show that $\boldsymbol{v}$ satisfies $\|\boldsymbol{v} - \widehat{\boldsymbol{\mu}}_I\|_\infty < H$. We already know by construction that $\left|v_j - \widehat{\mu}_{I_j}\right| < H$ for all $j \in [t_s]$. For the case $j = t_s + 1$, we have

$$\left|v_{t_s+1} - \widehat{\mu}_{I_{t_s+1}}\right| = \left|\frac{2H}{p/\widehat{p}}\widehat{k} - H\left(1 - \frac{1 + 2\widehat{s}}{p}\right)\right| = \left|\frac{2H}{p}\left(\widehat{s} + \widehat{k}\widehat{p}\right) - H\left(1 - \frac{1}{p}\right)\right|,$$

which is less than $H(1 - 1/p) < H$ for all $\widehat{s} + \widehat{k}\widehat{p} \in \mathbb{Z}_p$. Similarly, for the case $j > t_s + 1$, we have

$$\left|v_j - \widehat{\mu}_{I_j}\right| = \left|\frac{2H}{p} a_{j-(t_s+1)} - H\left(1 - \frac{1}{p}\right)\right|,$$

which is less than $H(1 - 1/p) < H$ for all $a_{j-(t_s+1)} \in \mathbb{Z}_p$, as required. This completes the proof that

$\boldsymbol{v} \in V_{\widehat{s},\widehat{p}}$, so f is onto. $\square$

B Proof of Lemma 4.2

We lower bound the number of lattice points in the box $K_1 = \{\boldsymbol{v} \in \mathbb{R}^n : \|\boldsymbol{v} - \boldsymbol{\mu}\|_\infty < H\}$ of side length $2H$ which is centered on the non-lattice vector $\boldsymbol{\mu}$, by the number of lattice points in the box $K_2 = \{\boldsymbol{v} \in \mathbb{R}^n : \|\boldsymbol{v} - \boldsymbol{\mu}'\|_\infty < H - \epsilon\}$ of side length $2(H-\epsilon)$, which is centered on a lattice vector $\boldsymbol{\mu}'$. We obtain the lattice vector $\boldsymbol{\mu}'$ by 'rounding' the non-lattice vector $\boldsymbol{\mu}$ to a 'nearby' lattice vector. Suppose that the 'rounding error' satisfies $\|\boldsymbol{\mu} - \boldsymbol{\mu}'\|_\infty \leq \epsilon$. Then it is easy to see by the triangle inequality that the box K_2 defined above is fully contained within the box K_1, and thus the number of lattice points inside K_2 is indeed a lower bound on the number of lattice points in K_1. In turn, since any lattice is invariant under additions of any lattice vector, it follows that the number of lattice points in the box K_2 is equal to the number of points in the origin-centered box $\{\boldsymbol{v} \in \mathbb{R}^n : \|\boldsymbol{v}\|_\infty < H - \epsilon\}$, which is the desired result.

It remains to prove the claimed bound $\epsilon = \frac{n}{2}\lambda_n(\mathcal{L})$ on the rounding error $\|\boldsymbol{\mu} - \boldsymbol{\mu}'\|_\infty$. By definition of the nth Minkowski minimum λ_n of the lattice, we know that there exists a set $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ of n linearly-independent lattice vectors such that $\|\boldsymbol{b}_j\|_\infty \leq \lambda_n$ for all $j = 1, \ldots, n$. Note that although the vectors $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ do not necessarily form a basis for the lattice, they do necessarily form a basis for the vector space $\mathbb{R}^n$ over $\mathbb{R}$. Hence any vector $\boldsymbol{\mu} \in \mathbb{R}^n$ can be expanded as $\boldsymbol{\mu} = c_1\boldsymbol{b}_1 + \cdots + c_n\boldsymbol{b}_n$ for some real coefficients $c_1, \ldots, c_n$. Now let $\boldsymbol{\mu}'$ denote the lattice vector which is obtained by rounding the coefficients $c_1, \ldots, c_n$ to the nearest integers, i.e. we let

$$\boldsymbol{\mu}' = \widehat{c}_1\boldsymbol{b}_1 + \cdots + \widehat{c}_n\boldsymbol{b}_n,$$

where for $i = 1, \ldots, n$, $\widehat{c}_i$ denotes integer closest to c_i. Then the rounding error is

$$\|\boldsymbol{\mu} - \boldsymbol{\mu}'\|_\infty = \|\sum_j (c_j - \widehat{c}_j)\boldsymbol{b}_j\|_\infty \leq \frac{1}{2}\sum_j \|\boldsymbol{b}_j\|_\infty \leq \left(\frac{n}{2}\right)\lambda_n,$$

as claimed. This completes the proof. $\square$

C Proof of Lemma 4.3

Fix $\widehat{p} \in \{1, p\}$ and an arbitrary positive integer $\Delta \leq 2H$. Let us find an upper bound on the fraction $\delta_I(\widehat{p})$ of $\boldsymbol{\alpha}_I \in D\left((\mathbb{Z}_p^*)^{t_s}\right)$ for which $\lambda_1(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, \widehat{p})) < \Delta$. To do so, observe that any vector $\boldsymbol{v}_{a,\boldsymbol{k}} \in \mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, \widehat{p})$ has the form

$$\boldsymbol{v} = \left(a(\alpha_{i_1}) + k_1 p, \ldots, a(\alpha_{i_{t_s}}) + k_{t_s} p, 2H\frac{a_1}{p}, \ldots, 2H\frac{a_{t_s}}{p}\right),$$

for some polynomial $a(x) = a_1 x + \cdots + a_t x^t \in \mathbb{Z}[x; t]$ with $a_1 \equiv 0 \pmod{\widehat{p}}$ and $\boldsymbol{k} = (k_1, \ldots, k_{t_s}) \in \mathbb{Z}^{t_s}$. We consider several cases.

The first case is $a(x) = 0$, i.e. $a_i = 0$ for all $i \in [t]$. In this case, $v_j = k_j p$ for $j \in [t_s]$ and $v_j = 0$ for $j > t_s$. So if $\boldsymbol{v}$ is non-zero, there must exist $j \in [t_s]$ such that k_j is a non-zero integer, which means $\|\boldsymbol{v}\|_\infty \geq p \geq 2H \geq \Delta$ in this case.

The second case is $a(x) \neq 0$ but $a(x) \equiv 0 \pmod{p}$. In this case, there must exist $i \in [t]$ such that a_i is a non-zero integer but $a_i \equiv 0 \pmod{p}$. So $\|\boldsymbol{v}\|_\infty \geq |v_{t_s+i}| = |2H(a_i/p)| \geq 2H \geq \Delta$ in this case too.

The remaining third case is $a(x) \not\equiv 0 \pmod{p}$, and from the first two cases above, we conclude that

the fraction $\delta_I(\widehat{p})$ defined above is equal to the fraction of $\boldsymbol{\alpha}_I \in D\left((\mathbb{Z}_p^*)^{t_s}\right)$ for which there exists $a(x) \in \mathbb{Z}[x;t]$ and $\boldsymbol{k} \in \mathbb{Z}^{t_s}$ satisfying

$$a(x) \not\equiv 0 \pmod{p} \text{ and } a_1 \equiv 0 \pmod{\widehat{p}} \text{ and } a_0 = 0 \tag{39}$$

and

$$\|\boldsymbol{v}\|_\infty < \Delta. \tag{40}$$

Let us now consider the case $\widehat{p} = 1$. The existence of $a(x)$ and $\boldsymbol{k}$ satisfying (39) and (40) implies that $\widehat{a}(x) \stackrel{\text{def}}{=} \lfloor a(x) \rfloor_p$ is a polynomial in $\mathbb{Z}_p[x;t]$ of degree at least 1 (using (39)) and satisfying $\|\widehat{a}(\alpha_{i_j})\|_{L,p} \leq \left|a(\alpha_{i_j}) + k_j p\right| < \Delta$ for all $j \in [t_s]$ (using (40)). Also from (40) we know that the coefficients of $\widehat{a}$ satisfy $\widehat{a}_0 = 0$ and $\|\widehat{a}_i\|_{L,p} \leq |a_i| < \left(\frac{\Delta}{2H}\right) p$ for all $i \in [t]$, so $\widehat{a}$ belongs to a subset $\widehat{A}$ of $\mathbb{Z}_p[x;t]$ containing at most $\left(2\frac{\Delta}{2H}p + 1\right)^t$ polynomials. Applying Lemma 2.1 (with parameters $n = t_s$, $B = \Delta$, $\#A \leq \left(2\frac{\Delta}{2H}p + 1\right)^t$) we conclude that such $\widehat{a}$ (and hence also a and $\boldsymbol{k}$ satisfying (39) and (40)) can exist for at most a fraction $\delta_I(1)$ of $\boldsymbol{\alpha}_I \in D\left((\mathbb{Z}_p^*)^{t_s}\right)$, where

$$\delta_I(1) \leq \frac{\left(2\frac{\Delta}{2H}p + 1\right)^t \cdot (2\Delta t)^{t_s}}{\#D\left((\mathbb{Z}_p^*)^{t_s}\right)}. \tag{41}$$

Now, using $\Delta \geq 1$ and $p \geq 2H$ we have $2\frac{\Delta}{2H}p \geq 2\frac{p}{2H} \geq 2$ so $2^2\frac{\Delta}{2H}p \geq 2\frac{\Delta}{2H}p + 1$. Plugging this inequality in (41) we get

$$\delta_I(1) \leq \left(\frac{\#(\mathbb{Z}_p)^{t_s}}{\#D\left((\mathbb{Z}_p^*)^{t_s}\right)}\right) \cdot \frac{\left(2^2\frac{\Delta}{2H}p\right)^t \cdot (2\Delta t)^{t_s}}{\#(\mathbb{Z}_p)^{t_s}}. \tag{42}$$

Observe that $\frac{\#(\mathbb{Z}_p)^{t_s}}{\#D\left((\mathbb{Z}_p^*)^{t_s}\right)} = \frac{p^{t_s}}{(p-1)\cdots(p-t_s)} \leq 2^{t_s}$ using $p - t_s \geq p/2$, which follows from the lemma hypothesis that $p \geq 2t_s$. Using this and the fact that $\det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1)) = (2H)^t p^{t_s - t}$ we obtain

$$\delta_I(1) \leq 2^{2(t_s+t)+t_s\log t}\frac{\Delta^{t_s+t}}{\det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1))}. \tag{43}$$

Hence, a sufficient condition for achieving $\delta_I(1) \leq 2^{-\beta(t_s+t)}$ is to pick Δ as an integer in the interval $[1, 2H]$ such that the right-hand side of (43) is at most $2^{-\beta(t_s+t)}$. Rearranging the latter sufficient condition for Δ we get

$$\Delta \leq 2^{-\left(\beta+2+\frac{t_s\log t}{t_s+t}\right)} \det\left(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1)\right)^{\frac{1}{t_s+t}} \text{ and } 1 \leq \Delta \leq 2H.$$

Hence it suffices to pick

$$\Delta = \lfloor 2^{-\left(\beta+2+\frac{t_s\log t}{t_s+t}\right)} \det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1))^{\frac{1}{t_s+t}} \rfloor \geq 2^{-\left(\beta+3+\frac{t_s\log t}{t_s+t}\right)} \det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1))^{\frac{1}{t_s+t}},$$

with the latter inequality satisfied (and also $\Delta \in [1, 2H]$) due to the lemma hyptothesis that

$$1 \leq 2^{-\left(\beta+3+\frac{t_s\log t}{t_s+t}\right)} \det(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, 1))^{\frac{1}{t_s+t}} \leq H.$$

This establishes the claims of the lemma in the case $\widehat{p} = 1$.

We now consider the remaining case $\widehat{p} = p$. Notice that in this case we have from (39) that $a_1 \equiv 0 \pmod{p}$. If $t = 1$, it follows that the third case $a(x) \not\equiv 0 \pmod{p}$ is not satisfied for any $\boldsymbol{v}_{a,\boldsymbol{k}} \in \mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, p)$, so $\delta_I(p) = 0$ for $t = 1$, and from now on we may assume that $t \geq 2$. Then, the existence

of $a(x)$ and $\boldsymbol{k}$ satisfying (39) and (40) implies that $\widehat{a}(x) \stackrel{\text{def}}{=} \lfloor a(x) \rfloor_p$ is a polynomial in $\mathbb{Z}_p[x;t]$ of degree at least 2 (using (39)) and satisfying $\|\widehat{a}(\alpha_{i_j})\|_{L,p} \leq \left|a(\alpha_{i_j}) + k_j p\right| < \Delta$ for all $j \in [t_s]$ (using (40)). Also from (40) we know that the coefficients of $\widehat{a}$ satisfy $\widehat{a}_0 = \widehat{a}_1 = 0$ and $\|\widehat{a}_i\|_{L,p} \leq |a_i| < (\frac{\Delta}{2H})p$ for $2 \leq i \leq t$, so $\widehat{a}$ belongs to a subset $\widehat{A}$ of $\mathbb{Z}_p[x;t]$ containing at most $\left(2\frac{\Delta}{2H}p+1\right)^{t-1}$ polynomials. Applying Lemma 2.1 (with parameters $n = t_s$, $B = \Delta$, $\#A \leq \left(2\frac{\Delta}{2H}p+1\right)^{t-1}$) we conclude that such $\widehat{a}$ (and hence also a and $\boldsymbol{k}$ satisfying (39) and (40)) can exist for at most a fraction $\delta_I(p)$ of $\boldsymbol{\alpha}_I \in D\left((\mathbb{Z}_p^*)^{t_s}\right)$, where

$$\delta_I(p) \leq \frac{\left(2\frac{\Delta}{2H}p+1\right)^{t-1} \cdot (2\Delta t)^{t_s}}{\#D\left((\mathbb{Z}_p^*)^{t_s}\right)}. \tag{44}$$

Now, using the bounds $\Delta \geq 1$, $p \geq 2H$ and $\frac{\#(\mathbb{Z}_p)^{t_s}}{\#D\left((\mathbb{Z}_p^*)^{t_s}\right)} = \frac{p^{t_s}}{(p-1)\cdots(p-t_s)} \leq 2^{t_s}$ as in the $\widehat{p}=1$ case above, we obtain

$$\delta_I(p) \leq 2^{t_s} \cdot \frac{(2^2\frac{\Delta}{2H}p)^{t-1} \cdot (2\Delta t)^{t_s}}{p^{t_s}}. \tag{45}$$

Using the fact that $\det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I)) = (2H)^{t-1}p^{t_s-(t-1)}$ we obtain

$$\delta_I(p) \leq 2^{2(t_s+t-1)+t_s\log t}\frac{\Delta^{t_s+t-1}}{\det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))}. \tag{46}$$

Hence, a sufficient condition for achieving $\delta_I(p) \leq 2^{-\beta(t_s+t-1)}$ is to pick Δ as an integer in the interval $[1, 2H]$ such that the right-hand side of (46) is at most $2^{-\beta(t_s+t-1)}$. Rearranging the latter sufficient condition for Δ we get

$$\Delta \leq 2^{-\left(\beta+2+\frac{t_s\log t}{t_s+t-1}\right)} \det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))^{\frac{1}{t_s+t-1}} \text{ and } 1 \leq \Delta \leq 2H.$$

Hence it suffices to pick

$$\Delta = \lfloor 2^{-\left(\beta+2+\frac{t_s\log t}{t_s+t-1}\right)} \det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))^{\frac{1}{t_s+t-1}} \rfloor \geq 2^{-\left(\beta+3+\frac{t_s\log t}{t_s+t-1}\right)} \det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))^{\frac{1}{t_s+t-1}},$$

with the latter inequality satisfied (and also $\Delta \in [1, 2H]$) due to the lemma hyptothesis that

$$1 \leq 2^{-\left(\beta+3+\frac{t_s\log t}{t_s+t-1}\right)} \det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))^{\frac{1}{t_s+t-1}} \leq H.$$

This shows that $\lambda_1(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, p)) \geq 2^{-\left(\beta+3+\frac{t_s\log t}{t_s+t-1}\right)} \det(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I))^{\frac{1}{t_s+t-1}}$ for at least a fraction $1 - 2^{-\beta(t_s+t-1)}$ of $\boldsymbol{\alpha}_I \in D((\mathbb{Z}_p^*)^{t_s})$.

To complete the proof of the lemma in the case $\widehat{p} = p$, we observe that the (t_s+1)th column (which is removed to form $\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I)$) of the basis matrix for $\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, p)$ has zero entries for all entries except row t_s+1. It follows that to each non-zero vector $\boldsymbol{v}' \in \mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I)$ there corresponds a non-zero vector $\boldsymbol{v} \in \mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, p)$ with the same entries as $\boldsymbol{v}'$ and a zero entry added in the new coordinate t_s+1, so $\|\boldsymbol{v}\|_\infty = \|\boldsymbol{v}'\|_\infty$. Hence $\lambda_1(\mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I)) \geq \lambda_1(\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, p))$, as claimed. This completes the proof of the lemma. □

D Proof of Lemma 4.4

The lemma follows from the fact (to be established below) that all vectors $\boldsymbol{v} \in V_{s,p}$ have their (t_s+1)th coordinate equal to zero. Indeed, using this fact we can define a one to one mapping $f : V_{s,p} \to V'_{s,p}$ as follows: for each $\boldsymbol{v} \in V_{s,p}$ we let $\boldsymbol{v}' = f(\boldsymbol{v})$ be the vector in Q^{t_s+t-1} obtained from $\boldsymbol{v}$ by removing

the (t_s+1)th coordinate. For any $\boldsymbol{v} \in V_{s,p}$, we have $\|\boldsymbol{v} - \widehat{\boldsymbol{\mu}}_I\|_\infty < H$ and therefore $\|\boldsymbol{v'} - \widehat{\boldsymbol{\mu}}'_I\|_\infty < H$ because the corresponding coordinates of $\boldsymbol{v}$,$\boldsymbol{v'}$ and $\widehat{\boldsymbol{\mu}}_I$,$\widehat{\boldsymbol{\mu}}'_I$ are equal. Also $\boldsymbol{v'} \in \mathcal{L}'_{Sha}(\boldsymbol{\alpha}_I)$ because the (t_s+1)th coordinate of $\boldsymbol{v}$ is 0, so $\boldsymbol{v}$ can be written as an integer linear combination in the rows of $M_{Sha}(\boldsymbol{\alpha}_I, p)$ with the coefficient of row (t_s+1) being 0, and hence $\boldsymbol{v'}$ can be written as the same integer linear combination of corresponding rows of $M'_{Sha}(\boldsymbol{\alpha}_I)$. Thus f indeed maps $V_{s,p}$ into $V'_{s,p}$. The mapping f is clearly one to one because $f(\boldsymbol{v}_1) = f(\boldsymbol{v}_2)$ for some $\boldsymbol{v}_1$,$\boldsymbol{v}_2$ in $V_{s,p}$ means by definition of f that $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are equal on all coordinates except possibly the (t_s+1)th, but in fact they are also both equal to 0 in their (t_s+1)th coordinate since $\boldsymbol{v}_1$,$\boldsymbol{v}_2$ are both in $V_{s,p}$ (by the abovementioned result). This proves that $\#V'_{s,p} \geq \#V_{s,p}$ (in fact equality holds but we do not use this).

It remains to show that $\boldsymbol{v} \in V_{s,p}$ implies that $v_{t_s+1} = 0$. To see this, suppose towards a contradiction that $\boldsymbol{v} \in V_{s,p}$ and $v_{t_s+1} \neq 0$. But $\boldsymbol{v} \in \mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, p)$, and from the definition of $\mathcal{L}_{Sha}(\boldsymbol{\alpha}_I, p)$ it is clear that $v_{t_s+1} = k \cdot 2H$ for some non-zero integer k. This means that $|v_{t_s+1}| \geq 2H$, which together with $\left|\widehat{\mu}_{I_{t_s+1}}\right| = \left|H(1 - \frac{1+2\widehat{s}}{p})\right| < H$ for all $\widehat{p} \in \mathbb{Z}_p$ implies that $\left|v_{t_s+1} - \widehat{\mu}_{I_{t_s+1}}\right| > H$, which contradicts our assumption that $\boldsymbol{v} \in V_{s,p}$. So $\boldsymbol{v} \in V_{s,p}$ implies that $v_{t_s+1} = 0$, as required. This completes the proof of the lemma. □

E Proof of Lemma 4.5

Let N denote the number of points of the lattice $\mathcal{L}$ in the box $K_1 = \{\boldsymbol{v} \in \mathbb{R}^n : \|\boldsymbol{v} - \boldsymbol{\mu}\|_\infty < H\}$. Suppose that on each lattice point $\boldsymbol{v}$ in the box , we center an open box $S_{\boldsymbol{v}} = \{\boldsymbol{z} \in \mathbb{R}^n : \|\boldsymbol{z} - \boldsymbol{v}\|_\infty < \lambda_1(\mathcal{L})/2\}$ of side length $\lambda_1(\mathcal{L})/2$. Note that as $\boldsymbol{v}$ runs through all lattice vectors in the box K_1, the boxes $S_{\boldsymbol{v}}$ are disjoint (because by the triangle inequality, the existence of a vector $\boldsymbol{z}$ in two of the boxes $S_{\boldsymbol{v}_1}$ and $S_{\boldsymbol{v}_2}$ implies that $\|\boldsymbol{v}_1 - \boldsymbol{v}_2\|_\infty < \lambda_1(\mathcal{L})$, which is a contradiction since $\boldsymbol{v}_1 - \boldsymbol{v}_2$ is itself a lattice vector), and occupy a total volume $N \cdot \lambda_1(\mathcal{L})^n$.

On the other hand, applying the triangle inequality again, we have that all the above N disjoint boxes $S_{\boldsymbol{v}}$ are contained within the box $K_2 = \{\boldsymbol{z} \in \mathbb{R}^n : \|\boldsymbol{v} - \boldsymbol{\mu}\|_\infty < H + \lambda_1(\mathcal{L})/2\}$, which has volume $Vol(K_2) = (2H + \lambda_1(\mathcal{L}))^n$.

It follows that

$$Vol(K_2) = (2H + \lambda_1(\mathcal{L}))^n \geq N \cdot \lambda_1(\mathcal{L})^n,$$

and therefore,

$$N \leq \left(\frac{2H}{\lambda_1(\mathcal{L})} + 1\right)^n,$$

as required. This completes the proof. □