SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

March 2022, Vol. 65 132301:1–132301:14
https://doi.org/10.1007/s11432-020-2958-9

A detailed analysis of primal attack and its variants

Xue ZHANG[1], Zhongxiang ZHENG[1*] & Xiaoyun WANG[1,2*]

[1]*Institute for Advanced Study, Tsinghua University, Beijing* 100084*, China;*
[2]*Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, Shandong University, Jinan* 250100*, China*

Received 17 March 2020/Revised 22 April 2020/Accepted 17 June 2020/Published online 26 May 2021

Abstract Primal attack is a typically considered strategy to estimate the hardness of cryptosystem based on learning with errors problem (LWE), it reduces the LWE problem to the unique-SVP by embedding technique and then employs lattice reduction such as BKZ to find the shortest vector. The main reason for the popularity of primal attack is its conservative estimation, in general, the complexity of primal attack is estimated by the hardness of core-SVP as $\mathcal{T} = 2^{0.292b}$. In this work, we first revisit primal attack and give supplemental proof of the scaling factor in Bai-Galbraith embedding, whose value was given according to the experimental results. Then we refine primal attack in two special cases and analyze the variants in detail. One is that, for sparse secret LWE (or sparse secret-error LWE), primal attack with dropping makes a trade-off between guessing zero components and solving dimension-reduced problems to improve the complexity. The other is that, when $\mathcal{T}_{\mathrm{BKZ}}(b) = \mathrm{poly}(d) \cdot \mathcal{T}_{\mathrm{Sieve}}(b)$ holds in practice, primal attack with preprocessing reduces the time complexity by a factor of 2^6–2^{10} through dividing primal attack into three steps and considering them independently.

Keywords cryptanalysis, lattice-based cryptography, learning with errors problem, primal attack, unique-SVP

Citation Zhang X, Zheng Z X, Wang X Y. A detailed analysis of primal attack and its variants. Sci China Inf Sci, 2022, 65(3): 132301, https://doi.org/10.1007/s11432-020-2958-9

1 Introduction

In recent years, the learning with errors (LWE) problem [1] has become an important part of cryptography to design cryptographic schemes. The main reason is that a reduction from worst-case lattice problems to average-case LWE guarantees the hardness of LWE problem. Moreover, it also has advantages of the great simplicity, the efficient implementation and the wide applications, including public-key encryption/key exchange [2,3], full-homomorphic encryption [4], digital signatures [5], and multilinear maps [6].

Informally, the search-LWE problem characterized by parameters $n, q \in \mathbb{Z}$ and standard deviation σ is that: given a matrix $\boldsymbol{A} \in \mathbb{Z}_q^{m\times n}$ and a vector $\boldsymbol{b} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{e} \mod q$ for a short error $\boldsymbol{e} \leftarrow \chi_\sigma$, find the secret $\boldsymbol{s}$. The decision-LWE problem is to distinguish LWE instances from uniform ones. In fact, the LWE problem is a generalization of the learning parity with noise (LPN) problem and its dual problem is the short integer solution problem (SIS).

To assess the security of the concrete instances of LWE, many strategies were proposed such as a primal attack, dual attack [7], hybrid attack [8], BKW algorithm [9], combinational method [10], and Arora-Ge algebraic attack [11]. However, the typically considered strategy for a search-LWE problem is primal attack, because its estimation is pretty conservative and it only requires polynomial LWE samples.

The primal attack was first proposed by Alkim et al. in NewHope [2] and we recap it as follows. Given a concrete LWE instance $(\boldsymbol{A}, \boldsymbol{b})$, the attacker constructs an embedding lattice as

$$\mathcal{L} = \{\boldsymbol{x} \in \mathbb{Z}^{m+n+1} : (\boldsymbol{I}_m | \boldsymbol{A}| - \boldsymbol{b})\boldsymbol{x} = \boldsymbol{0} \mod q\},$$

and the secret is contained in the unique shortest vector $\boldsymbol{v} := (\boldsymbol{e}^{\mathrm{T}}|\boldsymbol{s}^{\mathrm{T}}|1)^{\mathrm{T}}$ of the embedding lattice. To solve such unique-shortest vector problem (unique-SVP), we perform a lattice reduction such as block

* Corresponding author (email: zhengzx13@tsinghua.org.cn, xiaoyunwang@tsinghua.edu.cn)

Korkine-Zolotarev (BKZ) with appropriate block size b. Under the geometric series assumption, BKZ finds a basis whose Gram-Schmidt norms have the following property: $\|\boldsymbol{b}_i^*\| = \delta_b^{d-2(i-1)} \cdot \mathrm{vol}(\mathcal{L})^{1/d}$ for $1 \leqslant i \leqslant d$. If the projection $\pi_{d-b+1}(\boldsymbol{v})$ is shorter than $\boldsymbol{b}_{d-b+1}^*$, then we get $\pi_{d-b+1}(\boldsymbol{v})$ and even the shortest vector $\boldsymbol{v}$. That is to say, the block size should meet a condition

$$\frac{\|\boldsymbol{v}\|}{\sqrt{d}}\sqrt{b} \leqslant \delta_b^{2b-d} \cdot \mathrm{vol}(\mathcal{L})^{1/d}.$$

The complexity of solving the unique-SVP is estimated by the hardness of core-SVP as $\mathcal{T} = 2^{0.292b}$.

1.1 Related work

The strategy that reduces the LWE problem to the unique-SVP has been studied in [12] since 2013. The differences were that they got the unique-SVP by applying Kannan's embedding technique and paid more attention to the λ_2/λ_1-gap of the embedding lattice. They derived a model for the success of the strategy from experiments, i.e., the shortest vector was found with fixed probability whenever $\lambda_2/\lambda_1 \geqslant \tau \cdot \delta_0^m$ for a constant $\tau \in (0, 1]$.

Bai and Galbraith [13] gave a new embedding technique which is more effective when error and secret are chosen from different distributions. Specifically, the shortest vector was re-balanced as $(\boldsymbol{e}, w\boldsymbol{s}, w)^{\mathrm{T}}$ with scaling factor $w := \sigma_e/\sigma_s$ and the new embedding lattice was

$$\mathcal{L}_w = \left\{ \boldsymbol{x} \in \mathbb{Z}^{m+n+1} : \left(\boldsymbol{I}_m \left| \frac{1}{w}\boldsymbol{A} \right| - \frac{1}{w}\boldsymbol{b} \right) \boldsymbol{x} = \boldsymbol{0} \mod q \right\}.$$

After the primal attack was proposed, Albrecht et al. [14] compared primal attack with the estimation in 2013 [12] and experimentally verified the correctness of primal attack. They also pointed out that the shortest vector $\boldsymbol{v}$ can be recovered with high probability by using size reduction on its projection $\pi(\boldsymbol{v})$.

Moreover, a variant of the primal attack, guess-and-verify decoding attack, was proposed in [15]. This attack finds the projected vector $\pi(\boldsymbol{v})$ by solving batch bounded distance decoding (BDD) problems rather than lattice reduction, and then lifts the projection to the unique shortest vector.

1.2 Our contribution

Inspired by the recent progress and works in primal attack, we refine primal attack for some special cases and analyze the variants of primal attack in detail.

After some fixed notations and necessary backgrounds in Section 2, we review the primal attack and embedding techniques in Section 3. In Section 4, we focus on three different embedding lattices including Kannan's embedding lattice, dual embedding lattice and Bai-Galbraith embedding lattice, then give a supplemental proof that: (1) The optimal scaling factor for Bai-Galbraith embedding is $w = \sigma_e/\sigma_s$ which used to be an experimental result. (2) As a corollary, the complexity of solving unique-SVP in three embedding lattices are

$$\mathcal{T}_{\text{Bai-Galbraith}} \leqslant \mathcal{T}_{\text{dual}} \leqslant \mathcal{T}_{\text{Kannan}},$$

where the two equality holds if and only if $\sigma_e = \sigma_s$.

Our second contribution is to analyze a variant of primal attack called primal attack with dropping (PAD) in Section 5. For LWE instances with sparsity such as binary or ternary LWE, we guess k zero components of secret before employing primal attack on a dimension-reduced problem. The optimal k is obtained such that $\frac{n-k}{n-k-h} \approx 2^{0.292\Delta b}$ where Δb is a constant determined by LWE parameters and h is the Hamming weight of secret. For most LWE-based cryptosystem, we have $\Delta b \in (0, 1]$ and it implies that the LWE instance is "secure" when $h > (1 - 2^{-0.292}) \cdot n \approx \frac{11}{60}n$ holds.

Finally, we show another variant named primal attack with preprocessing (PAP) in Section 6, which in fact divides primal attack into three steps: preprocessing the lattice basis, finding the projection of the shortest vector and lifting to the full lattice. We prove that, under the core-SVP model, the complexity of PAP is at most as low as that of primal attack. However, when $\mathcal{T}_{\text{BKZ}}(b) = \mathrm{poly}(d) \cdot \mathcal{T}_{\text{Sieve}}(b)$ holds in practice, PAP reduces the time complexity of primal attack by a factor of 2^6–2^{10}.

2 Preliminaries

We use small bold letters to represent column vectors (e.g., $\boldsymbol{a}, \boldsymbol{b}$) and use $\boldsymbol{a}^{\mathrm{T}}$ to denote the transposition of $\boldsymbol{a}$. Let a_i denote the i-th coordinate of $\boldsymbol{a}$. We use capital bold letters to represent matrices (e.g., $\boldsymbol{A}, \boldsymbol{B}$) and use $\boldsymbol{B}_{[a,b]}$ to denote the matrix consisted of vectors $\boldsymbol{b}_i$ for $i \in [a, b]$. Let $\|\cdot\|_p$ denote the p-norm of vectors and $\|\cdot\|$ denote Euclidean norm.

2.1 Lattice

The lattice $\mathcal{L}$ is a discrete additive subgroup in d-dimensional Euclidean space and it is composed of all integer combinations of n linearly independent vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_n \in \mathbb{R}^d$, i.e.,

$$\mathcal{L}(\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_n) = \left\{ \boldsymbol{y} \in \mathbb{R}^d : \ \boldsymbol{y} = \sum_{i=1}^{n} z_i \cdot \boldsymbol{b}_i, \ z_i \in \mathbb{Z} \right\}.$$

The matrix $\boldsymbol{B} := [\boldsymbol{b}_1 | \cdots | \boldsymbol{b}_n]$ is called a basis and n is the rank of lattice. Although the basis for a lattice is not unique that $\boldsymbol{B}' = \boldsymbol{B}\boldsymbol{U}$ is also a basis if and only if $\boldsymbol{U}$ is unimodular matrix, the volume of bases $\det(\boldsymbol{B}) = \sqrt{\det(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{B})}$ is an invariant.

The q-ary lattice that contains $q \cdot \mathbb{Z}^d$ as a sublattice is a kind of lattice commonly used in lattice-based cryptography. A q-ary lattice generated by matrix $\boldsymbol{B} \in \mathbb{Z}_q^{d \times n}$ is defined as

$$\mathcal{L}_q(\boldsymbol{B}) = \{\boldsymbol{y} \in \mathbb{Z}^d : \boldsymbol{y} = \boldsymbol{B}\boldsymbol{z} \mod q, \ \boldsymbol{z} \in \mathbb{Z}^n\}.$$

The basis of lattice is $[\boldsymbol{B}|q\boldsymbol{I}_d]$ and the rank is d. Moreover, if all columns of $\boldsymbol{B}$ are linearly independent over $\mathbb{Z}_q$, then $\mathrm{vol}(\mathcal{L}_q(\boldsymbol{B})) = q^{d-n}$.

For random lattice, according to Gaussian heuristic, the length of the shortest vector (denoted as λ_1) is asymptotically equals to the radius of the ball with volume $\mathrm{vol}(\mathcal{L})$ (denoted as $\mathrm{GH}(\mathcal{L})$). In the q-ary lattice, we also assume that Gaussian heuristic holds then $\lambda_1 \approx \min\{q, q^{\frac{d-n}{d}} \cdot \sqrt{\frac{d}{2\pi e}}\}$.

Assumption 1 (Gaussian heuristic)**.** Given a lattice $\mathcal{L}$ and a measurable set $\mathcal{S}$, the number of points in $\mathcal{L} \cap \mathcal{S}$ is approximately $\mathrm{vol}(\mathcal{S})/\mathrm{vol}(\mathcal{L})$.

2.2 Learning with errors problem

The LWE problem [1] is one of the famous computational problems in lattices, which can be viewed as a generalization of the LPN problem. Due to the LWE problem has worst-case to average-case reduction, it is widely used in designing cryptography, including identity-based encryption, full-homomorphic encryption, digital signatures, and multilinear maps.

Definition 1 (LWE [1])**.** Let $n, m, q > 0$ be integers, Φ be a probability distribution of errors and secret $\boldsymbol{s}$ be chosen from the uniform distribution on $\mathbb{Z}_q^n$. An LWE sample $(\boldsymbol{a}, b = \langle \boldsymbol{a}, \boldsymbol{s} \rangle + e)$ follows the LWE distribution $L_{\boldsymbol{s},\Phi}$ on $\mathbb{Z}_q^n \times \mathbb{Z}_q$, which is obtained by choosing vector $\boldsymbol{a}$ from uniform distribution on $\mathbb{Z}_q^n$ and choosing error e from Φ distribution. Given m LWE samples, the decision-LWE problem is to distinguish LWE distribution $L_{\boldsymbol{s},\Phi}$ from uniform distribution on $\mathbb{Z}_q^n \times \mathbb{Z}_q$ and the search-LWE problem is to find secret $\boldsymbol{s}$.

In LWE-based cryptography, the secret $\boldsymbol{s}$ is usually chosen from the same distribution as an error rather than a uniform distribution. In fact, it is easy to construct such LWE instances at the cost of losing n LWE samples [16]. Moreover, there is also some cryptography in which the standard deviation of distribution for secret $\boldsymbol{s}$ is smaller than that for error $\boldsymbol{e}$. These secret $\boldsymbol{s}$ is always chosen from $\{0, 1\}^n$ or $\{-1, 0, 1\}^n$ and the corresponding LWE problem is called binary-LWE or ternary-LWE problem [13].

2.3 BKZ algorithm

The basic method of solving the LWE problem is to solve the SVP. At present, there are four families of algorithms to solve SVP, including lattice reduction, enumeration, random sieve and Voronoi cell algorithm. However, both random sieve and Voronoi cell algorithm are difficult to be used independently because of the exponential-memory requirement, while enumeration has great time complexity for high-dimension lattices. Therefore, lattice reduction with different SVP oracles (e.g., random sieve, enumeration) is a commonly used algorithm to estimate the security of cryptography.

There are two classical and well-known lattice reduction algorithms: the Lenstra-Lenstra-Lovasz (LLL) algorithm and the BKZ algorithm. LLL algorithm is a special case of the BKZ algorithm whose block size is 2. As for BKZ algorithm, Gama and Nguyen [17] proposed that Hermite factor $\delta^d = \|\boldsymbol{b}_1\|/\mathrm{vol}(\mathcal{L})^{1/d}$ can be used to describe the shape of output basis, i.e., the Gram-Schmidt norms of output basis are satisfied $\|\boldsymbol{b}_i^*\|/\|\boldsymbol{b}_{i+1}^*\| \approx \delta^2$ where δ is called root-Hermite factor. Then, Chen [18] gave a relationship between the block size b and root-Hermite factor δ in (1) which was also effective for finite m.

$$\lim_{m\to+\infty} \delta = \boldsymbol{v}_b^{-\frac{1}{b\cdot(b-1)}} \approx \left(\frac{b}{2\pi e}\cdot(\pi b)^{1/b}\right)^{\frac{1}{2(b-1)}}. \tag{1}$$

To assess the complexity of BKZ, let r denote the round of BKZ and $\mathcal{T}_{\mathrm{svp}}(b)$ denote the cost of SVP oracle solving b-dimensional lattice, then the complexity of BKZ with block size b is $r\cdot d\cdot\mathcal{T}_{\mathrm{svp}}(b)$. Although the best upper bound of r is exponential, according to the work of Hanrot et al. [19], taking $r \approx (d^2\log d)/b^2$ is large enough to make δ_b close to the estimated value.

3 Primal attack

The primal attack is a kind of classical and useful attack model for the search-LWE problem and it only requires polynomial LWE samples. The core idea is that transforming the search-LWE instance into a unique-SVP and solving the unique shortest vector by lattice reduction with appropriate root-Hermite factor δ. We recap this model as follows.

Given an LWE instance $(\boldsymbol{A}, \boldsymbol{b} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{e})$ with matrix $\boldsymbol{A} \in \mathbb{Z}^{m\times n}$, there are three types of embedding techniques to reduce search-LWE problem: Kannan's embedding, dual embedding and Bai-Galbraith embedding.

(1) The Kannan's embedding [20] reduces the BDD problem to SVP. Given an LWE instance, the BDD problem is $(\mathcal{L}'_K, \boldsymbol{b})$-BDD where the lattice $\mathcal{L}'_K$ is defined as

$$\mathcal{L}'_K = \{\boldsymbol{y}' \in \mathbb{Z}^m : \boldsymbol{y}' = \boldsymbol{A}\boldsymbol{x}' \mod q,\ \forall \boldsymbol{x}' \in \mathbb{Z}^n\}.$$

Correspondingly, when reducing $(\mathcal{L}'_K, \boldsymbol{b})$-BDD to SVP, the embedding lattice $\mathcal{L}_K$ is defined as

$$\mathcal{L}_K = \left\{\boldsymbol{y} \in \mathbb{Z}^{m+1} : \boldsymbol{y} = \bar{\boldsymbol{A}}\boldsymbol{x} \mod q,\ \forall \boldsymbol{x} \in \mathbb{Z}^{n+1},\ \bar{\boldsymbol{A}} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{b} \\ \boldsymbol{0} & \mu \end{bmatrix} \in \mathbb{Z}^{(m+1)\times(n+1)}\right\},$$

and in practice it is preferable to use $\mu = 1$. The volume of embedding lattice is $\mathrm{vol}(\mathcal{L}_K) = q^{m-n}$ if all columns of $\bar{\boldsymbol{A}}$ are linearly independent over $\mathbb{Z}_q$, and $\boldsymbol{v}^{\mathrm{T}} = (\boldsymbol{e}^{\mathrm{T}}|1)$ is a short vector in lattice.

(2) The dual embedding proposed by Bai and Galbraith [13] constructs a lattice related to both secret $\boldsymbol{s}$ and error $\boldsymbol{e}$. The corresponding embedding lattice is

$$\mathcal{L}_D = \{\boldsymbol{x} \in \mathbb{Z}^{m+n+1} : (\boldsymbol{I}_m|\boldsymbol{A}| - \boldsymbol{b})\boldsymbol{x} = \boldsymbol{0} \mod q\},$$

which has a basis

$$\boldsymbol{B} = \begin{bmatrix} q\boldsymbol{I}_m & -\mathbf{A} & \boldsymbol{b} \\ \boldsymbol{0} & \boldsymbol{I}_n & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & 1 \end{bmatrix}.$$

The volume of lattice is $\mathrm{vol}(\mathcal{L}_D) = q^m$ and the $\boldsymbol{v}^{\mathrm{T}} = (\boldsymbol{e}^{\mathrm{T}}|\boldsymbol{s}^{\mathrm{T}}|1)$ is a short vector in lattice.

(3) The Bai-Galbraith embedding improves dual embedding for such LWE instance that secret and error are chosen from different distributions, its core idea is to balance the size of the error and the secret. Specifically, the short vector in lattice $\mathcal{L}_D$ can be re-balanced as $(\boldsymbol{e}^{\mathrm{T}}|w\boldsymbol{s}^{\mathrm{T}}|w)$ with scaling factor $w = \sigma_e/\sigma_s$ and the new embedding lattice is

$$\mathcal{L}_w = \left\{\boldsymbol{x} \in \mathbb{Z}^{m+n+1} : \left(\boldsymbol{I}_m \left|\frac{1}{w}\boldsymbol{A}\right| - \frac{1}{w}\boldsymbol{b}\right)\boldsymbol{x} = \boldsymbol{0} \mod q\right\}.$$

The volume of new lattice increases to $\mathrm{vol}(\mathcal{L}_w) = q^m w^{n+1}$.

In general, the length of the short vector $\boldsymbol{v}$ in embedding lattice is much shorter than $\mathrm{GH}(\mathcal{L})$, so the search-LWE problem is reduced to the unique-SVP. Then the unique shortest vector can be founded by applying lattice reduction such as BKZ with appropriate block size b.

At present, the condition proposed by Alkim et al. in NewHope [2] is the most popular approach to predict block size. To be more concrete, assuming that the behavior of BKZ satisfies the geometric series assumption (GSA), BKZ finds a basis whose Gram-Schmidt norms are given by $\|\boldsymbol{b}_i^*\| = \delta_b^{d-2(i-1)} \cdot \mathrm{vol}(\mathcal{L})^{1/d}$ for $1 \leqslant i \leqslant d$. Let the unique shortest vector be $\boldsymbol{v} = \sum_{i=1}^{d} u_i \boldsymbol{b}_i$ and the projection of $\boldsymbol{v}$ onto the vector space spanned by the last b Gram-Schmidt vectors be $\pi_{d-b+1}(\boldsymbol{v})$. If the projection $\pi_{d-b+1}(\boldsymbol{v})$ is shorter than $\boldsymbol{b}_{d-b+1}^*$, i.e.,

$$\frac{\|\boldsymbol{v}\|}{\sqrt{d}}\sqrt{b} \leqslant \delta_b^{2b-d} \cdot \mathrm{vol}(\mathcal{L})^{1/d}, \tag{2}$$

then we can find $\pi_{d-b+1}(\boldsymbol{v})$ and even the shortest vector $\boldsymbol{v}$. Therefore, such block size b is suitable for BKZ and the total complexity is estimated as $2^{0.292b}$ for classical case and $2^{0.265b}$ for quantum one.

Moreover, Albrecht et al. [14] compared primal attack with the estimation in 2013 [12] and experimentally verified the correctness of primal attack. They also pointed out that the short vector $\boldsymbol{v}$ can be recovered with high probability by size reduction. The detailed requirements for the size reduction are described in Lemma 1.

Lemma 1 (Claim 1 in [14])**.** Let the block size of BKZ algorithm be b, the current basis of d-dimensional embedding lattice be $\boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d]$ and the unique shortest vector be $\boldsymbol{v} = \sum_{i=1}^{d} u_i \boldsymbol{b}_i$. If Eq. (2) holds for b, the GSA holds for basis until index $d-b+1$ and $\boldsymbol{b}_{d-b+1}^* = \pi_{d-b+1}(\boldsymbol{v})$, then the size reduction called on $\boldsymbol{b}_{d-b+1}$ recovers $\boldsymbol{v}$ with high probability p that

$$p = \sum_{i=1}^{d-b} p_i,\ p_i = \Pr\big[\|\pi_i(\boldsymbol{v})\| < \min\{\|\pi_i(\boldsymbol{v}) + \boldsymbol{b}_i^*\|, \|\pi_i(\boldsymbol{v}) - \boldsymbol{b}_i^*\|\}\big].$$

Note that the probability p only relies on b and it is close to 1 for $b \geqslant 50$.

4 Supplemental proof of Bai-Galbraith embedding

In this section, we focus on the binary (ternary)-LWE problem whose hardness is an open question. For all we know, the Bai-Galbraith embedding [13] proposed in 2014 is the most popular and useful method to solve it. However, in the work of Bai and Galbraith, they only gave the experimental comparison between this embedding and Kannan's embedding, the assignment of scaling factor w is heuristic and intuitive. In this section, we try to compare three embedding techniques and give supplemental proof of the optimal w.

As mentioned in Section 3, the essence of primal attack is to solve a unique-SVP, then λ_2/λ_1-gap is an important index to estimate the hardness of unique-SVP. Specifically, let λ_1 and λ_2 denote the length of the shortest vector and the second shortest vector respectively and λ_2/λ_1-gap $= \lambda_2/\lambda_1$. Ref. [12] shows that the bigger the λ_2/λ_1-gap is, the simpler the unique-SVP is. Therefore, the optimal w is obtained when λ_2/λ_1-gap approaches to the maximum value.

Next, we apply the inequality (2) to strictly prove the optimal parameters for Bai-Galbraith embedding, including the re-balanced dimension (i.e., the dimension of secret which need to be re-balanced with error), scaling factor and samples.

Theorem 1. For Bai-Galbraith embedding, the optimal re-balanced dimension is $n+1$ and the optimal scaling factor is σ_e/σ_s. Moreover, the optimal samples for primal attack is $m \approx \lceil\sqrt{\frac{(n+1)(\log q - \log w)}{\log \delta_b}} - (n+1)\rfloor$.

Proof. Due to Eq. (2) of primal attack, we have $\delta_b^{d-2b}\sqrt{\frac{b}{2\pi e}} \leqslant \frac{\sqrt{\frac{d}{2\pi e}}\mathrm{vol}(\mathcal{L})^{1/d}}{\|\boldsymbol{v}\|}$. Notice that:

(1) When the block size b is large enough (i.e., $b \geqslant 50$), the left side of inequality is a decreasing function of b.

(2) In this embedding lattice, we have $\lambda_1 = \|\boldsymbol{v}\|$ and $\lambda_2 \approx \mathrm{GH}(\mathcal{L}) \approx \sqrt{\frac{d}{2\pi e}}\mathrm{vol}(\mathcal{L})^{1/d}$, then the right side of inequality equals to λ_2/λ_1-gap.

It implies that, if the block size b is large enough, then d-dimension lattice with larger λ_2/λ_1-gap requires the smaller block size and the lower complexity for solving unique-SVP. Therefore, let $k \in [0, n+1]$ be

the number of re-balanced dimension and $t \in \mathbb{R}^+$ be the scaling factor, we want to choose the optimal values to get the largest λ_2/λ_1-gap, i.e.,

$$\max_{t,k}\{\lambda_2/\lambda_1\text{-gap}\}, \quad \lambda_2/\lambda_1\text{-gap} \approx \frac{\mathrm{GH}(\mathcal{L})}{\|\boldsymbol{v}\|} \approx \frac{(t^k q^m)^{1/d}\sqrt{\frac{d}{2\pi e}}}{\sqrt{\sigma_e^2 m + \sigma_s^2(n+1-k) + \sigma_s^2 t^2 k}}, \quad d = m+n+1.$$

Let $F(t,k)$ be the above-mentioned function of factor t and dimension k and $\Delta := \sigma_e^2 m + \sigma_s^2(n+1-k) + \sigma_s^2 t^2 k$, then

$$\begin{aligned} F_t'(t,k) &= \frac{q^{\frac{m}{d}}\sqrt{\frac{d}{2\pi e}} \cdot \frac{k}{d} t^{\frac{k}{d}-1} \cdot \Delta^{\frac{1}{2}} - q^{\frac{m}{d}}\sqrt{\frac{d}{2\pi e}} t^{\frac{k}{d}} \cdot \frac{1}{2}\Delta^{-\frac{1}{2}} \cdot 2t\sigma_s^2 k}{(\Delta^{\frac{1}{2}})^2} \\ &= \frac{q^{\frac{m}{d}} t^{\frac{k}{d}} \sqrt{\frac{d}{2\pi e}} k}{\Delta^{\frac{1}{2}}} \cdot \left(\frac{1}{d} t^{-1} - \Delta^{-1} \cdot t\sigma_s^2\right). \end{aligned}$$

When $F_t' = 0$ we have

$$\begin{aligned} &\sigma_s^2 t^2 d = \sigma_e^2 m + \sigma_s^2(n+1-k) + \sigma_s^2 t^2 k, \\ &t = \sqrt{\frac{\sigma_e^2 m + \sigma_s^2(n+1-k)}{\sigma_s^2(d-k)}}, \end{aligned}$$

the $F_t' > 0$ for $t < \sqrt{\frac{\sigma_e^2 m + \sigma_s^2(n+1-k)}{\sigma_s^2(d-k)}}$ and $F_t' < 0$ for $t > \sqrt{\frac{\sigma_e^2 m + \sigma_s^2(n+1-k)}{\sigma_s^2(d-k)}}$. Therefore, the function $F(t,k)$ takes the maximum at $t = \sqrt{\frac{\sigma_e^2 m + \sigma_s^2(n+1-k)}{\sigma_s^2(d-k)}}$.

Then we write $F(k) = \max_t F(t,k)$ as a function of dimension k and the optimal t is also a function of k.

$$\begin{aligned} F_k'(k) &= \frac{q^{\frac{m}{d}}\sqrt{\frac{d}{2\pi e}} \cdot (t^{\frac{k}{d}})_k' \cdot \Delta^{\frac{1}{2}} - q^{\frac{m}{d}}\sqrt{\frac{d}{2\pi e}} t^{\frac{k}{d}} \cdot \frac{1}{2}\Delta^{-\frac{1}{2}} \cdot \sigma_s^2(-1 + (t^2 k)_k')}{(\Delta^{\frac{1}{2}})^2} \\ &= \frac{q^{\frac{m}{d}}\sqrt{\frac{d}{2\pi e}}}{\Delta^{\frac{1}{2}}} \cdot \left((t^{\frac{k}{d}})_k' - t^{\frac{k}{d}} \frac{\Delta^{-1}}{2}\sigma_s^2((t^2 k)_k' - 1)\right), \end{aligned}$$

where $(t^{\frac{k}{d}})_k' = t^{\frac{k}{d}} \frac{1}{d}\left(\ln t + \frac{k}{t} \cdot t_k'\right)$, $(t^2 k)_k' = t^2 + 2tk \cdot t_k'$ and $t_k' = \frac{1}{2t} \cdot \frac{m(\sigma_e^2 - \sigma_s^2)}{(d-k)^2 \sigma_s^2}$. Note that

$$\frac{\sigma_s^2}{\Delta} = \frac{1}{d} \cdot \frac{\sigma_s^2 d}{\sigma_e^2 m + \sigma_s^2(n+1-k) + \sigma_s^2 t^2 k} = \frac{1}{d} \cdot \frac{\sigma_s^2 d}{\sigma_e^2 m + \sigma_s^2(n+1-k)} \cdot \frac{d-k}{d} = \frac{1}{t^2 d},$$

we have

$$\begin{aligned} F_k'(k) &= \frac{q^{\frac{m}{d}}\sqrt{\frac{d}{2\pi e}}}{\Delta^{\frac{1}{2}}} \cdot \left(t^{\frac{k}{d}} \frac{1}{d}\left(\ln t + \frac{k}{t} \cdot t_k'\right) - t^{\frac{k}{d}} \frac{\sigma_s^2}{2\Delta}(t^2 + 2tk \cdot t_k' - 1)\right) \\ &= \frac{q^{\frac{m}{d}}\sqrt{\frac{d}{2\pi e}} t^{\frac{k}{d}}}{\Delta^{\frac{1}{2}} d} \cdot \left(\left(\ln t + \frac{k}{t} \cdot t_k'\right) - \frac{1}{2t^2}(t^2 + 2tk \cdot t_k' - 1)\right) \\ &= \frac{q^{\frac{m}{d}}\sqrt{\frac{d}{2\pi e}} t^{\frac{k}{d}}}{\Delta^{\frac{1}{2}} d} \cdot \left(\ln t + \frac{1}{2t^2} - \frac{1}{2}\right). \end{aligned}$$

Let $G(t) = \ln t + \frac{1}{2t^2} - \frac{1}{2}$. If $\sigma_e^2 \geqslant \sigma_s^2$ holds, then $t \geqslant 1$ and $G'(t) = \frac{1}{t}(1 - \frac{1}{t^2}) \geqslant 0$. What is more, $G(t) \geqslant G(1) = 0$ and $F_k'(k) \geqslant 0$ hold. Therefore, the function $F(k)$ takes the maximum at $k = n+1$.

In conclusion, when taking $k = n+1$ and $t = \sqrt{\frac{\sigma_e^2 m + \sigma_s^2(n+1-k)}{\sigma_s^2(d-k)}} = \sigma_e/\sigma_s (= w)$, the $F(t,k) = \frac{(w^{n+1} q^m)^{1/d}}{\sigma_e\sqrt{2\pi e}}$ is maximum. Furthermore, Eq. (2) equals to $\frac{\delta_b^{2b-d} w^{\frac{n+1}{d}} q^{\frac{m}{d}}}{\sigma_e\sqrt{b}} \geqslant 1$ then the optimal number of samples is calculated as $m \approx \left\lceil \sqrt{\frac{(n+1)(\log q - \log w)}{\log \delta_b}} - (n+1) \right\rfloor$.

Remark 1. In the proof of the theorem, we ignore the component 1 in the shortest vector $\boldsymbol{v}$. Certainly, we can modify the proof using $\sigma_s' = \sqrt{\frac{n\sigma_s^2+1}{n+1}}$, but the influence is negligible.

Remark 2. Eq. (2) equals to $\frac{\delta_b^{2b-d}\cdot \mathrm{vol}(\mathcal{L})^{1/d}}{\frac{\|\boldsymbol{v}\|}{\sqrt{d}}\sqrt{b}} \geqslant 1$, and let the left side of inequality be a function of b and d, i.e., $H(b,d) = \frac{\delta_b^{2b-d}\cdot \mathrm{vol}(\mathcal{L})^{1/d}\cdot\sqrt{d}}{\|\boldsymbol{v}\|\sqrt{b}}$. For Kannan's embedding, if b_K and $d_K = m_K + 1$ is the optimal parameters, then $H_K(b_K, d_K) = \frac{\delta_{b_K}^{2b_K-d_K}\cdot q^{(m_K-n)/d_K}\cdot\sqrt{d_K}}{\|\boldsymbol{v}_K\|\sqrt{b_K}} \geqslant 1$. As for dual embedding, if we use the same parameters b_K and $d_K = m_D'+n+1$ as Kannan's embedding, we have $H_D(b_K, d_K) = \frac{\delta_{b_K}^{2b_K-d_K}\cdot q^{m_D'/d_K}\cdot\sqrt{d_K}}{\|\boldsymbol{v}_D\|\sqrt{b_K}}$. Since $m_D' = m_K - n$ and $\|\boldsymbol{v}_D\| \leqslant \|\boldsymbol{v}_K\|$ for $\sigma_s \leqslant \sigma_e$, the $H_D(b_K, d_K) \geqslant H_K(b_K, d_K) \geqslant 1$ holds. That is to say, using the optimal m_D in dual embedding, the corresponding optimal b_D must satisfy that $b_D \leqslant b_K$. Moreover, the dual embedding can also be viewed as an instance of Bai-Galbraith embedding with scaling factor $t = 1$ and dimension $k = n + 1$, then we have $b_{\mathrm{BG}} \leqslant b_D$. Therefore, the complexity of solving unique-SVP in three embedding lattices are

$$\mathcal{T}_{\text{Bai-Galbraith}} \leqslant \mathcal{T}_{\text{dual}} \leqslant \mathcal{T}_{\text{Kannan}},$$

where the two equality holds if and only if $\sigma_e = \sigma_s$.

5 The detailed analysis of primal attack with dropping

For convenience in writing, we default the primal attack using Bai-Galbraith embedding in this section.

5.1 Primal attack with dropping for sparse secret LWE

In Bai and Galbraith's work, they pointed out that the primal attack for binary (ternary)-LWE can be combined with exhaustive search because of the narrow secret space, i.e., we first guess a few entries of the secret and then apply the primal attack against the reduced LWE instance.

In fact, the exhaustive method may perform well when the entries of secret are chosen uniformly from $\{0,1\}^n$ or $\{-1,0,1\}^n$. However, in many cases the secret follows other sparse distributions. Notice that the secret in binary (ternary)-LWE always has sufficient zeros, for example, the 1024-dimensional secret in RLizard-128[1)] only has fixed Hamming weight 128. Therefore, it is possible to exploit the sparsity of secret to improve the primal attack, i.e., we guess the entries with $s_i = 0$ and use the primal attack on the simpler instance.

This idea was briefly mentioned by Albrecht et al. [7, 14, 15] under different situations and "the core idea is that the lower running time of the dimension-reduced problem will trade-off positively against the probability of guessing zero components". However, as far as we know, this method called PAD has not been analyzed in detail. In the following section, we show the concrete algorithm (see Algorithm 1) and give the theoretical analysis about it.

Algorithm 1 Primal attack with dropping (PAD)

Input: A binary (ternary)-LWE instance $(\boldsymbol{A}, \boldsymbol{b} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{e})$ where $\boldsymbol{A} \in \mathbb{Z}_{m\times n}$ and $m \leqslant 2n$, the number k of guessing zeros.
Output: The secret $\boldsymbol{s}$.

1: **repeat**
2: Randomly guess k entries of $\boldsymbol{s}$ and the set J consists of its subscripts;
3: Drop the J columns of $\boldsymbol{A}$ to get reduced instance $(\boldsymbol{A}', \boldsymbol{b})$;
4: Apply primal attack to solve the unique-SVP form the reduced instance, the output vector is $\boldsymbol{v}' \in \mathbb{Z}^{d'}$ for $d' = m' + n - k + 1$;
5: Recover the secret $\boldsymbol{s}$ by adding zeros in the corresponding entries of $\boldsymbol{s}'$ where $\boldsymbol{v}' = (\boldsymbol{e}'^{\mathrm{T}}|\boldsymbol{s}'^{\mathrm{T}}|c)^{\mathrm{T}} \in \mathbb{Z}^{m'} \times \mathbb{Z}^{n-k} \times \mathbb{Z}$ and compute $\boldsymbol{e} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{s}$;
6: **until** $(\boldsymbol{s} \in \{0,1\}^n\ (\boldsymbol{s} \in \{-1,0,1\}^n)$ and $\|\boldsymbol{e}\| \approx \sigma_e\sqrt{m})$;
7: **return** the secret $\boldsymbol{s}$.

Let us explain why this strategy is anticipated to perform better than primal attack directly. Let the subscript set J in Algorithm 1 correspond to a permutation matrix $\boldsymbol{P}$ such that $\boldsymbol{P}\boldsymbol{s} = [\boldsymbol{s}_0^{\mathrm{T}}|\boldsymbol{s}_1^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{Z}^k \times \mathbb{Z}^{n-k}$. If $J \subseteq I := \{i \,|\, s_i = 0\}$ holds, then $\boldsymbol{s}_0 = \boldsymbol{0}$ and $\boldsymbol{b} = \boldsymbol{A}'\boldsymbol{s}_1 + \boldsymbol{e}$. Obviously, when applying the

1) Cheon J H, Kim D H, Lee J H, et al. Lizard public key encryption submission to nist proposal. Submission for Post-Quantum Cryptography Standardization of NIST, 2017. https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-1-Submissions.

primal attack on the reduced LWE instance, it requires smaller block size b' and fewer LWE samples m'. Accordingly, the total complexity of this strategy is $\mathcal{T} = 1/\mathrm{pr}_k \cdot \mathcal{T}_{\mathrm{BKZ}}(b')$ where pr_k denotes the success probability of $J \subseteq I$. Moreover, if the secret has fixed Hamming weight h, then the success probability is $\mathrm{pr}_k = \frac{C_{n-h}^k}{C_n^k} = \prod_{i=0}^{k-1}(1 - \frac{h}{n-i})$.

5.1.1 *The comparison with exhaustive search*

We compare this strategy with exhaustive search and show a sufficient condition in Theorem 2 when the strategy performs better.

Theorem 2. Let h be the Hamming weight of n-dimensional secret and k be the number of guessing entries. Given a ternary-LWE instance, if $(2n - 3h + 2)/2 \geqslant k$ holds then the strategy of guessing zero has lower complexity than exhaustive search. Similarly, given a binary-LWE instance, if $(n - 2h + 1) \geqslant k$ holds then the strategy of guessing zero has lower complexity than exhaustive search.

Proof. As for ternary-LWE, assuming that we need to guess k entries of the secret $\boldsymbol{s}$, the probability that the k entries contain j nonzero entries and $k - j$ zero entries is $p_j = \frac{C_h^j \cdot C_{n-h}^{k-j}}{C_n^k}$ and the complexity of determining the k entries is

$$\mathcal{T}_j = 1/p_j \cdot C_k^j 2^j = \frac{C_n^k \cdot C_k^j}{C_h^j \cdot C_{n-h}^{k-j}} \cdot 2^j,$$

where $0 \leqslant j \leqslant k$. Then we have

$$\frac{\mathcal{T}_{j+1}}{\mathcal{T}_j} = \frac{\frac{C_n^k \cdot C_k^j \frac{k-j}{j+1}}{C_h^j \frac{h-j}{j+1} \cdot C_{n-h}^{k-j-1}} \cdot 2^{j+1}}{\frac{C_n^k \cdot C_k^j}{C_h^j \cdot C_{n-h}^{k-j-1} \frac{n-h-(k-j-1)}{k-j}} \cdot 2^j} = \frac{2(n-h-(k-j-1))}{h-j}.$$

Notice that $\mathcal{T}_{j+1}/\mathcal{T}_j \geqslant 1 \Leftrightarrow 2(n-k+1) \geqslant 3(h-j)$. That is to say, under the sufficient condition $(2n - 3h + 2)/2 \geqslant k$, guessing k zero entries has the lowest complexity than other strategies.

Similarly, we can deduce that the sufficient condition for binary secret is $(n - 2h + 1) \geqslant k$.

Remark 3. Here we give two examples to show that it is easy to achieve the sufficient condition. For RLizard-128, the 1024-dimension ternary secret has fixed Hamming weight 128 then the sufficient condition equals to $k \leqslant 0.8125n + 1$. For LAC-192, the 1024-dimension ternary secret has expected Hamming weight 256 then the sufficient condition equals to $k \leqslant 0.625n + 1$.

5.1.2 *The success of PAD*

Although the strategy of guessing zero entries is better than an exhaustive search, its cost depends heavily on the parameters of the LWE instance, especially on the Hamming weight of a secret. We want to know when it is effective to combine with a primal attack.

Notice that, guessing an extra entry (i.e., the number k form β to $\beta + 1$) means that the success probability pr_k decrease by $\frac{n-\beta-h}{n-\beta}$ while the complexity of lattice reduction decrease by $2^{0.292 \cdot \Delta b}$ for $\Delta b = b(\beta) - b(\beta + 1)$. To estimate Δb, we calculate the values $b(k)$ by Eq. (2) and we observe that $b(k)$ is linear with respect to k. Therefore, the difference of block size Δb is a constant determined by the parameters of the LWE instance and the optimal k is obtained when $\frac{n-k}{n-k-h} \approx 2^{0.292\Delta b}$ holds.

For example, we conduct the experiments on RLizard, in which the Hamming weight of secret is pretty small compared with other submissions of the NIST PQC project, to compare primal attack and PAD. The estimated k for RLizard128, RLizard192, and RLizard256 using the fitting data in Figure 1 are 225, 919, and 479 respectively, and they are consistent well with the actual value in Table 1. Moreover, both the classic complexity and quantum complexity of PAD are lower than that of primal attack.

Remark 4. For the most LWE-based cryptography, we have $\Delta b \leqslant 1$ since the required block size in primal attack is less than the dimension n with high probability. Then $h > (1 - 2^{-0.292}) \cdot n \approx \frac{11}{60} n$ is a sufficient condition when the LWE instance is "secure" against PAD.

5.2 PAD for sparse secret-error LWE

As for the sparse secret-error LWE, the above-mentioned strategy is also useful and it works even better than on the binary (ternary)-LWE. Let the error be chosen from the same distribution as secret and the samples $m = 2n$.

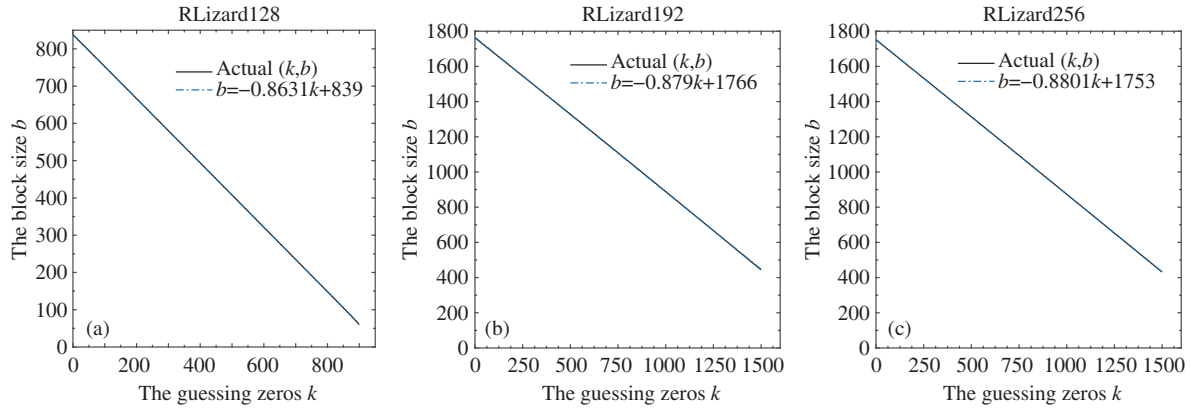

Figure 1 (Color online) The linear relationship of $b(k)$ and k, and the difference of block size Δb for (a) RLizard128 is 0.8631, (b) RLizard192 is 0.879, and (c) RLizard256 is 0.8801, respectively.

Table 1 The complexity of two attacks against RLizard

Schemes	Attack	Block size	Guess zeros	Classic	Quantum
RLizard-128	Primal attack	838	–	245	223
$n=1024,\ q=1024,\ h=128,\ \sigma_e=0.907$	PAD	648	221	**238**	221
RLizard-192	Primal attack	1762	–	515	467
$n=2048,\ q=2048,\ h=184,\ \sigma_e=1.827$	PAD	950	930	**449**	424
RLizard-256	Primal attack	1751	–	512	464
$n=2048,\ q=4096,\ h=256,\ \sigma_e=2.448$	PAD	1329	482	**495**	460

Notice that: (1) the secret and error are chosen independently from the sparse distribution, (2) determining one entry of error is equivalent to determining one entry of secret. For the two reasons, we guess zeros in secret and error at the same time, and then the success probability pr_{3k} decreases more slowly, i.e.,

$$\mathrm{pr}_{3k}=\frac{C_{n-h}^{k}}{C_{n}^{k}}\cdot\frac{C_{2n-2h}^{2k}}{C_{2n}^{2k}}=\prod_{i=0}^{k-1}\left(1-\frac{h}{n-i}\right)\left(1-\frac{2h}{2n-2i}\right)\left(1-\frac{2h}{2n-2i-1}\right).$$

According to the analysis in Subsection 5.3, the optimal k is the same such that $(1-\frac{h}{n-k})(1-\frac{2h}{2n-2k})(1-\frac{2h}{2n-2k-1})\approx(1-\frac{h}{n-k})^3\approx 2^{-0.292\cdot 3\Delta b}$, but the number of entries to be guessed is $3k$.

For the sake of completeness, we explain why determining one entry of error is equivalent to determining one entry of secret. Given a (m,n,q,σ)-LWE instance, the PAD algorithm guesses k entries of secret and $2k$ entries of error such that $3k<n$. Before the discussion, let us define three generator matrices as follows:

$$\boldsymbol{B}_1=\left[\begin{array}{cc|c}\boldsymbol{A} & \boldsymbol{b} & \\ \boldsymbol{I}_n & \boldsymbol{0} & q\boldsymbol{I}_d \\ \boldsymbol{0} & 1 & \end{array}\right],\quad \boldsymbol{B}_2=\left[\begin{array}{cc|c}\boldsymbol{A}_1 & \boldsymbol{b}_1 & \\ \boldsymbol{A}_2 & \boldsymbol{b}_2 & q\boldsymbol{I}_d \\ \boldsymbol{0} & 1 & \end{array}\right],\quad \boldsymbol{B}_3=\left[\begin{array}{cc|c}\boldsymbol{A}' & \boldsymbol{b}' & \\ \boldsymbol{I}_n & \boldsymbol{0} & q\boldsymbol{I}_d \\ \boldsymbol{0} & 1 & \end{array}\right].$$

It is known that the embedding lattice for LWE instance is generated by $\boldsymbol{B}_1$ and the shortest vector is $\boldsymbol{v}^{\mathrm{T}}=(\boldsymbol{e}^{\mathrm{T}}|\boldsymbol{s}^{\mathrm{T}}|1)$. Then we swap the positions of $2k$ guessed entries of error and $2k$ entries of secret without guessing, the new lattice generated by $\boldsymbol{B}_2$ has the shortest vector $\boldsymbol{v}'^{\mathrm{T}}=(\boldsymbol{e}'^{\mathrm{T}}|\boldsymbol{s}'^{\mathrm{T}}|1)$ where $\boldsymbol{s}'$ consists of $2k$ guessed entries of error, k guessed entries of secret and $n-3k$ entries of secret without guessing. Next, applying elementary column transformations on matrix $\boldsymbol{B}_2$, we could obtain an equivalent generator matrix $\boldsymbol{B}_3$ with probability $p_{\mathrm{inv}}=\Pi_{i=1}^{2k}(q^{n-k}-q^{i-1})/q^{2k\cdot(n-k)}$, i.e., the sub-matrix $\boldsymbol{A}_2$ is invertible with probability p_{inv}.

At this point, the (m,n,q,σ)-LWE instance could be reduced to a $(m,n-3k,q,\sigma)$-LWE instance because $3k$ guessed entries of $\boldsymbol{s}'$ are zeros. Therefore, the total complexity of PAD for sparse secret-error LWE is $\mathcal{T}=1/\mathrm{pr}_{3k}\cdot 1/p_{\mathrm{inv}}\cdot\mathcal{T}_{\mathrm{BKZ}}(b)$ where $\mathcal{T}_{\mathrm{BKZ}}(b)$ is the cost of solving $(m,n-3k,q,\sigma)$-LWE instance.

To emphasize the importance of sparsity, we conduct experiments to compare the primal attack, PAD with guessing secret and PAD with guessing secret and error for different distributions. Given the LWE instances $(n=1024,\ q=251,\ m=2n)$ in which secrets and errors are ternary with variances

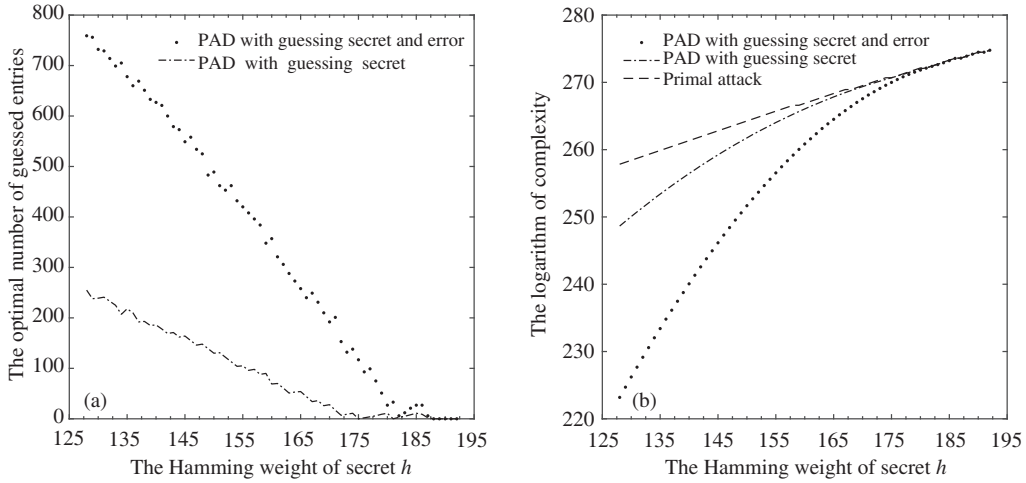

Figure 2 The comparison between primal attacks, PAD with guessing secret and PAD with guessing secret and error. (a) shows the relationship between the Hamming weight of secret and the optimal number of guessed entries, and (b) compares the complexity of these three algorithms, when the LWE instances use the secrets with different Hamming weights.

$\sigma^2 \in [1/8, 3/16]$, the complexity of three algorithms are shown in the right subgraph of Figure 2 and the optimal number of guessed entries for PAD are given in the left subgraph. We observe that: (1) the variance of distribution has more influence on PAD than primal attack; (2) the PAD with guessing secret and PAD with guessing secret and error have the same complexity as a primal attack when $h \geqslant \frac{11}{60}n$ holds.

6 The detailed analysis of primal attack with preprocessing

6.1 Primal attack with preprocessing

Primal attack solves the unique-SVP by applying lattice reduction directly, but its core idea is to find the projected vector of the unique shortest vector in the projected sublattice. Therefore, we divide primal attack into three steps: preprocessing the lattice basis by lattice reduction, finding the projected short vector, and lifting the projected short vector to the full lattice.

There are two related studies to support this division: One is [14], Albrecht et al. pointed out that the shortest vector $\boldsymbol{v}$ can be recovered by using Babai's nearest plane algorithm on the projected vector $\pi(\boldsymbol{v})$ (see Lemma 1). The other is [15], they proposed the guess-and-verify decoding attack which finds the projected vector $\pi(\boldsymbol{v})$ by solving batch BDD problems then lift the projected vector to the unique shortest vector.

Inspired by the above-mentioned ideas and studies, we consider the block size of lattice reduction and the dimension of projected sublattice independently, and give another variant of primal attack, PAP, in Algorithm 2. In fact, we can view it as a degenerate algorithm of guess-and-verify decoding mentioned in [15].

Algorithm 2 Primal attack with preprocessing (PAP)

Input: An LWE instance $(\boldsymbol{A}, \boldsymbol{b} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{e} \mod q)$ for $\boldsymbol{A} \in \mathbb{Z}^{m\times n}$ and $m \leqslant 2n$, the number β and η.
Output: The short vector $\boldsymbol{v}^{\mathrm{T}} = (\boldsymbol{e}^{\mathrm{T}}|\boldsymbol{s}^{\mathrm{T}}|1)$.
1: Applying BKZ algorithm with block size β on the d-dimensional LWE embedding lattice and get the basis $\boldsymbol{B}$;
2: Performing sieve (or enumeration) on the last η-dimensional projected lattice of $\mathcal{L}(\boldsymbol{B})$ to get $\pi(\boldsymbol{v}) = \sum_{i=d-\eta+1}^{d} c_i \boldsymbol{b}_i^*$;
3: Using Babai's nearest plane with $\sum_{i=d-\eta+1}^{d} c_i \boldsymbol{b}_i$ and $\boldsymbol{B}_{[1,d-\eta]}$ to get $\boldsymbol{v}$.

Obviously, Algorithm 2 is a probability algorithm whose time complexity can be expressed as

$$\mathcal{T}_{\mathrm{PAP}} = \mathcal{T}_{\mathrm{BKZ}} + \mathcal{T}_{\mathrm{Sieve}} + \mathcal{T}_{\mathrm{NP}},$$

where $\mathcal{T}_{\mathrm{BKZ}}$, $\mathcal{T}_{\mathrm{Sieve}}$, and $\mathcal{T}_{\mathrm{NP}}$ denote the complexity of BKZ, sieve and nearest plane algorithm respectively, and its success probability is p_{NP} which is the success probability of nearest plane algorithm. Meanwhile,

in order to ensure the success of getting $\pi(\boldsymbol{v})$, the variant condition from primal attack is that the projected vector $\pi(\boldsymbol{v})$ is the shortest vector in projected lattice, i.e.,

$$\frac{\|\boldsymbol{v}\|}{\sqrt{d}}\sqrt{\eta} \leqslant \mathrm{GH}(\mathcal{L}(\boldsymbol{B}^*_{[d-\eta+1,d]})), \tag{3}$$

where $\boldsymbol{B}^*$ denotes the Gram-Schmidt basis of $\boldsymbol{B}$. Eq. (3), according to GSA, can be written as the formula between parameters β and η,

$$\frac{\|\boldsymbol{v}\|}{\sqrt{d}}\sqrt{\eta} \leqslant \sqrt{\frac{\eta}{2\pi e}} \cdot \delta_\beta^{\eta+1-d}\mathrm{vol}(\mathcal{L})^{1/d}. \tag{4}$$

6.2 The detailed analysis of PAP

The total complexity of finding $\boldsymbol{v}$ is $\mathcal{T} = \mathcal{T}_{\mathrm{PAP}}/p_{\mathrm{NP}}$. Firstly, we consider the probability p_{NP}.

Theorem 3. Let β and η are the optimal parameters of PAP such that $\beta < \eta$ and Eq. (4). Under GSA, the success probability of nearest plane algorithm in Algorithm 2 has a lower bound $\widetilde{p_{\mathrm{NP}}}$,

$$\widetilde{p_{\mathrm{NP}}} \approx \prod_{k=1}^{\tilde{k}} I_{x^2}\left(\frac{1}{2}, \frac{\eta+k-1}{2}\right), \quad \tilde{k} = \max\{k : x < 1\}, \quad x = \frac{\delta_\eta^{2k}\sqrt{\eta}}{2\sqrt{\eta+k}},$$

where I is the regularized incomplete beta function. Moreover, $\widetilde{p_{\mathrm{NP}}}$ is only determined by parameter η.

Proof. In [14], the probability p_{NP} is calculated as

$$p_{\mathrm{NP}} = \prod_{i=1}^{d-\eta} p_i, \quad p_i \approx \begin{cases} 1 - \dfrac{\int_0^{2\frac{hi}{ri}-(\frac{hi}{ri})^2} t^{\frac{d-i}{2}-1}(1-t)^{-\frac{1}{2}}\mathrm{d}t}{\mathrm{Beta}\left(\frac{d-i}{2}, \frac{1}{2}\right)}, & R_i < 2r_i, \\ 1, & R_i \geqslant 2r_i, \end{cases} \tag{5}$$

where $R_i = \|\boldsymbol{b}_i^*\|$, $r_i = \|\pi_i(\boldsymbol{v})\|$, and $h_i = r_i - R_i/2$. We notice that the integral is a incomplete beta function, then Eq. (5) can be simplified as

$$p_{\mathrm{NP}} = \prod_{i=1}^{d-\eta} p_i, \quad p_i \approx \begin{cases} I_{(\frac{Ri}{2ri})^2}\left(\frac{1}{2}, \frac{d-i}{2}\right), & R_i < 2r_i, \\ 1, & R_i \geqslant 2r_i, \end{cases}$$

where I is the regularized incomplete beta function and $I_x(a,b) = 1 - I_{1-x}(b,a)$.

Let $\mathrm{gh} := \mathrm{GH}(\mathcal{L}(\boldsymbol{B}^*_{[d-\eta+1,d]}))$ and parameters $\beta < \eta$. Since $\|\pi_{d-\eta+1}(\boldsymbol{v})\| \approx \mathrm{gh} = \alpha \cdot \|\boldsymbol{b}^*_{d-\eta+1}\|$ holds, we have

$$\frac{R_{d-\eta+1-k}}{2r_{d-\eta+1-k}} = \frac{\delta_\beta^{2k}\sqrt{\eta}}{2\sqrt{\eta+k}} \cdot \frac{R_{d-\eta+1}}{r_{d-\eta+1}} = \frac{\delta_\beta^{2k}\sqrt{\eta}}{2\alpha\sqrt{\eta+k}},$$

and

$$p_{\mathrm{NP}} \approx \prod_{k=1}^{\min\{\tilde{k}, d-\eta\}} I_{x^2}\left(\frac{1}{2}, \frac{\eta+k-1}{2}\right), \quad \tilde{k} = \max\{k : x < 1\}, \quad x = \frac{\delta_\beta^{2k}\sqrt{\eta}}{2\alpha\sqrt{\eta+k}}.$$

Moreover, since $\mathrm{gh} = \sqrt{\frac{\eta}{2\pi e}} \cdot \delta_\beta^{\eta+1-d}\mathrm{vol}(\mathcal{L})^{1/d}$, $\|\boldsymbol{b}^*_{d-\eta+1}\| = \delta_\beta^{\eta-1} \cdot \delta_\beta^{\eta+1-d}\mathrm{vol}(\mathcal{L})^{1/d}$ and $\delta_\beta^{\eta-1} - \sqrt{\frac{\eta}{2\pi e}} > 0$, we have

$$\mathrm{gh} < \|\boldsymbol{b}^*_{d-\eta+1}\| \Rightarrow \alpha < 1 \Rightarrow \frac{\delta_\beta^{2k}\sqrt{\eta}}{2\alpha\sqrt{\eta+k}} > \frac{\delta_\beta^{2k}\sqrt{\eta}}{2\sqrt{\eta+k}} > \frac{\delta_\eta^{2k}\sqrt{\eta}}{2\sqrt{\eta+k}}.$$

It is well known that the regularized incomplete beta function is the cumulative distribution function of the beta distribution, i.e., I_{x^2} is an increasing function of x and $I_{x^2} < 1$, then $p_{\mathrm{NP}} > \widetilde{p_{\mathrm{NP}}}(\eta)$ where

$$\widetilde{p_{\mathrm{NP}}} \approx \prod_{k=1}^{\tilde{k}} I_{x^2}\left(\frac{1}{2}, \frac{\eta+k-1}{2}\right), \quad \tilde{k} = \max\{k : x < 1\}, \quad x = \frac{\delta_\eta^{2k}\sqrt{\eta}}{2\sqrt{\eta+k}}.$$

That is, the probability p_{NP} has a lower bound $\widetilde{p_{\mathrm{NP}}}$ which is only determined by parameter η.

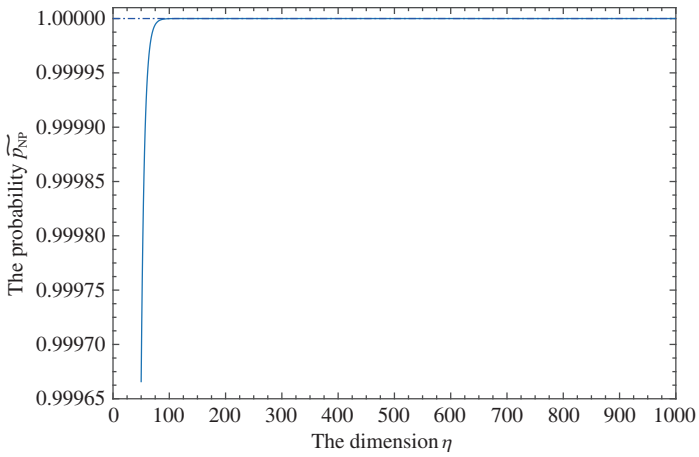

Figure 3 (Color online) The probability $\widetilde{p_{\rm NP}}$ for different η.

In fact, $\widetilde{p_{\rm NP}}$ is the probability mentioned in Lemma 1, which is the very close to 1 for $\eta \geqslant 50$. For the sake of completeness, we show the values of $\widetilde{p_{\rm NP}}$ in Figure 3.

Next, we analyze the complexity $\mathcal{T}_{\rm PAP}$. Because the complexity of the nearest plane algorithm is polynomial, we mainly consider the complexity of BKZ and sieve which depend on parameters β and η respectively. We give a theorem to state the relationship between the optimal block size b of primal attack and the optimal parameters (β, η) of PAP.

Theorem 4. Let b denote the optimal block size of the primal attack and (β, η) denote the block size of BKZ and dimension of sieve in PAP such that Eq. (4). When the condition $\beta < b$ holds, there must be $\eta \geqslant b$.

Proof. Let F, G, H be three inequalities as follows:

$$\begin{aligned} F:&\ \frac{\|\boldsymbol{v}\|}{\sqrt{d}}\sqrt{\eta} \leqslant \sqrt{\frac{\eta}{2\pi e}} \cdot \delta_\beta^{\eta+1-d}\mathrm{vol}(\mathcal{L})^{1/d},\\ G:&\ \frac{\|\boldsymbol{v}\|}{\sqrt{d}}\sqrt{b'} \leqslant \sqrt{\frac{b'}{2\pi e}} \cdot \delta_{b'}^{b'+1-d}\mathrm{vol}(\mathcal{L})^{1/d},\\ H:&\ \frac{\|\boldsymbol{v}\|}{\sqrt{d}}\sqrt{b} \leqslant \delta_b^{b-1} \cdot \delta_b^{b+1-d}\mathrm{vol}(\mathcal{L})^{1/d}. \end{aligned}$$

Let F_l, G_l, H_l denote the left items of inequalities and F_r, G_r, H_r denote the right ones. The (β, η, d_β), $(b', d_{b'})$ and (b, d_b) are the optimal parameters of inequalities respectively, where $d_t \approx \lceil \sqrt{\frac{(n+1)(\log q - \log w)}{\log \delta_t}} \rfloor$ and d_t is an increasing function of t. (Obviously, the optimal d for F only depends on β.)

Compare inequalities F and G, we first assume $\eta = b'$. If $\beta < b'$ holds, then

$$\delta_\beta > \delta_{b'} \Rightarrow \delta_\beta^{\eta+1-d} < \delta_{b'}^{b'+1-d} \Rightarrow F_r(d_\beta) < G_r(d_\beta)$$

for $b' + 1 - d < 0$. Therefore, when $(b', d_{b'})$ are the optimal parameters such that $1 \approx G_r/G_l$, we have

$$1 \approx G_r/G_l(d_{b'}) > G_r/G_l(d_\beta) > F_r/F_l(d_\beta).$$

To make sure the inequality F holds, the parameter η should meet condition $\eta \geqslant b'$.

Compare inequalities G and H, we first assume $b = b'$. Since

$$\delta_b^{b-1} - \sqrt{\frac{b}{2\pi e}} \approx \sqrt{\frac{b}{2\pi e}}\big((\pi b)^{1/(2b)} - 1\big) > 0$$

due to Eq. (1), we have $G_r < H_r$. That is to say, when $(b', d_{b'})$ are the optimal parameters such that $G_l \approx G_r$, the optimal parameters for H satisfy $b \leqslant b'$ and $d_b \leqslant d_{b'}$.

Notice that b is the optimal block size of primal attack and (β, η) is the optimal parameters of PAP. If $\beta < b$ holds, then we have $\beta < b' \Rightarrow \eta \geqslant b' \Rightarrow \eta \geqslant b$.

In general, the complexity of primal attack is estimated by the core-SVP as $2^{0.292b}$. In such case, according to the Theorems 3 and 4 we have either $\beta \geqslant b$ or $\eta \geqslant b$, then the complexity of PAP is at most as low as that of primal attack.

Table 2 The advantages of PAP under two models[a)]

Scheme-(n, q, σ)	PrimalAttack-(m, b)	PAP-model1-(m_1, β_1, η_1)	$\Delta\mathcal{T}_1$	PAP-model2-(m_2, β_2, η_2)	$\Delta\mathcal{T}_2$
Frodo-$(640, 2^{15}, 2.8)$	$(742, 483)$	$(723, 464, 509)$	2^4	$(717, 458, 516)$	2^6
Frodo-$(976, 2^{16}, 2.3)$	$(1046, 706)$	$(1025, 686, 734)$	2^5	$(1019, 681, 741)$	2^6
Frodo-$(1344, 2^{16}, 1.4)$	$(1285, 930)$	$(1263, 909, 959)$	2^5	$(1257, 904, 966)$	2^6
LAC-$(512, 251, \frac{1}{\sqrt{2}})$	$(406, 509)$	$(388, 481, 526)$	2^7	$(385, 477, 528)$	2^8
LAC-$(1024, 251, \frac{1}{2})$	$(631, 985)$	$(612, 955, 1002)$	2^8	$(609, 951, 1004)$	2^9
LAC-$(1024, 251, \frac{1}{\sqrt{2}})$	$(705, 1105)$	$(685, 1072, 1120)$	2^8	$(682, 1067, 1122)$	2^{10}
NewHope-$(512, 12289, 2)$	$(575, 385)$	$(555, 365, 410)$	2^4	$(550, 360, 417)$	2^6
NewHope-$(1024, 12289, 2)$	$(1053, 886)$	$(1032, 863, 911)$	2^6	$(1028, 858, 916)$	2^7
Saber-$(512, 8192, 1.58)$	$(548, 382)$	$(530, 363, 407)$	2^4	$(524, 357, 414)$	2^6
Saber-$(768, 8192, 1.41)$	$(763, 608)$	$(744, 587, 632)$	2^5	$(738, 581, 639)$	2^7
Saber-$(1024, 8192, 1.22)$	$(953, 824)$	$(932, 801, 849)$	2^5	$(928, 796, 854)$	2^7
Kyber-$(512, 3329, 1)$	$(493, 381)$	$(475, 361, 406)$	2^5	$(470, 356, 411)$	2^6
Kyber-$(768, 3329, 1)$	$(697, 623)$	$(678, 601, 647)$	2^5	$(674, 596, 652)$	2^7
Kyber-$(1024, 3329, 1)$	$(893, 873)$	$(873, 849, 896)$	2^6	$(868, 844, 901)$	2^7

a) The model1 comes from [15] such that $\mathcal{T}_{\mathrm{BKZ}}(b, d) = 8d \cdot \mathcal{T}_{\mathrm{Sieve}}(b)$, model2 comes from [16] such that $\mathcal{T}_{\mathrm{BKZ}}(b, d) = \frac{d^3 \log d}{b^2} \cdot \mathcal{T}_{\mathrm{Sieve}}(b)$. The $(n, q, \sigma_e, \sigma_s)$ are the parameters of different round2 submissions from NIST PQC project, m is the number of samples, b is the block size in primal attack, (β, η) are the block size of BKZ and dimension of sieve in PAP, $\Delta\mathcal{T}$ is the quotient of the cost of primal attack and that of PAP.

6.3 The comparison with primal attack for different models

Obviously, the estimation of primal attack is pretty conservative. In practice, we are unable to get BKZ reduced basis or projection $\pi(\boldsymbol{v})$ by running sieve only once. To the best of our knowledge, one of the reliable models for BKZ is that $\mathcal{T}_{\mathrm{BKZ}}(b) = \frac{d^2 \log d}{b^2} \cdot d \cdot \mathcal{T}_{\mathrm{Sieve}}(b)$ according to the work of Hanrot et al [19], and another useful model based on experiments from [21] is that $\mathcal{T}_{\mathrm{BKZ}}(b) = 8d \cdot \mathcal{T}_{\mathrm{Sieve}}(b)$. In other words, BKZ needs to call sieve as a subroutine for polynomial times, namely $\mathcal{T}_{\mathrm{BKZ}}(b) = \mathrm{poly}(d) \cdot \mathcal{T}_{\mathrm{Sieve}}(b)$. Next, we demonstrate PAP under these two models and show the experimental data in Table 2.

From the experiments we know that: on the one hand, the results of experiments verify the correctness of Theorem 4; on the other hand, PAP is effective to improve the practical complexity. Moreover, we could roughly give the upper bound of improvements that

$$\Delta\mathcal{T} \leqslant \mathrm{poly}(d) \cdot \mathcal{T}_{\mathrm{Sieve}}(b)/\mathcal{T}_{\mathrm{Sieve}}(\eta) \leqslant \mathrm{poly}(d).$$

In brief, PAP has advantages over primal attack in practice, and it can be viewed as a technique to improve the polynomial term in the complexity of the primal attack.

Acknowledgements This work was supported by National Key Research and Development Program of China (Grant Nos. 2017YFA0303903, 2018YFA0704701), Major Program of Guangdong Basic and Applied Research (Grant No. 2019B030302008), and Major Scientific and Technological Innovation Project of Shandong Province (Grant No. 2019JZZY010133).

References

1 Regev O. On lattices, learning with errors, random linear codes, and cryptography. J ACM, 2009, 56: 1–40
2 Alkim E, Ducas L, Pöppelmann T, et al. Post-quantum key exchange — a new hope. In: Proceedings of the 25th USENIX Security Symposium, Austin, 2016. 327–343
3 Bos J W, Costello C, Ducas L, et al. Frodo: take off the ring! practical, quantum-secure key exchange from LWE. In: Proceedings of ACM SIGSAC Conference on Computer and Communications Security, Vienna, 2016. 1006–1018
4 Brakerski Z, Vaikuntanathan V. Efficient fully homomorphic encryption from standard LWE. SIAM J Comput, 2014, 42: 831–871
5 Lyubashevsky V. Lattice signatures without trapdoors. In: Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques, 2012. 738–755
6 Garg S, Gentry C, Halevi S. Candidate multilinear maps from ideal lattices. In: Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques, 2013
7 Albrecht M R. On dual lattice attacks against small-secret LWE and parameter choices in helib and SEAL. In: Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques, 2017. 103–129
8 Wunderer T. A detailed analysis of the hybrid lattice-reduction and meet-in-the-middle attack. J Math Cryptol, 2019, 13: 1–26
9 Guo Q, Johansson T, Mårtensson E, et al. Coded-BKW with sieving. In: Proceedings of International Conference on the Theory and Application of Cryptology and Information Security, 2017. 323–346
10 Bai S, Galbraith S D, Li L Z, et al. Improved combinatorial algorithms for the inhomogeneous short integer solution problem. J Cryptol, 2019, 32: 35–83
11 Albrecht M R, Cid C, Faugére J C, et al. Algebraic algorithms for LWE problems. ACM Commun Comput Algebra, 2015, 49: 62

12 Albrecht M R, Fitzpatrick R, Göpfert F. On the efficacy of solving LWE by reduction to unique-svp. In: Proceedings of International Conference on Information Security and Cryptology, 2013. 293–310
13 Bai S, Galbraith S D. Lattice decoding attacks on binary LWE. In: Proceedings of Australasian Conference on Information Security and Privacy, 2014. 322–337
14 Albrecht M R, Göpfert F, Virdia F, et al. Revisiting the expected cost of solving usvp and applications to LWE. In: Proceedings of International Conference on the Theory and Application of Cryptology and Information Security, 2017. 297–322
15 Albrecht M R, Curtis B R, Wunderer T. Exploring trade-offs in batch bounded distance decoding. In: Proceedings of International Conference on Selected Areas in Cryptography, 2019. 467–491
16 Albrecht M R, Player R, Scott S. On the concrete hardness of learning with errors. J Math Cryptol, 2015, 9: 169–203
17 Gama N, Nguyen P Q. Predicting lattice reduction. In: Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques, 2008. 31–51
18 Chen Y M. Réduction de réseau et sécurité concrete du chiffrement completement homomorphe. Dissertation for Ph.D. Degree. Paris: École Normale Supérieure, 2013
19 Hanrot G, Pujol X, Stehlé D. Analyzing blockwise lattice algorithms using dynamical systems. In: Proceedings of Annual Cryptology Conference, 2011. 447–464
20 Kannan R. Minkowski's convex body theorem and integer programming. Math Oper Res, 1987, 12: 415–440
21 Chen Y M, Nguyen P Q. BKZ 2.0: better lattice security estimates. In: Proceedings of International Conference on the Theory and Application of Cryptology and Information Security, 2011