Chin. Ann. Math. Ser. B
44(6), 2023, 945–960
DOI: 10.1007/s11401-023-0053-6

Chinese Annals of Mathematics, Series B

Lattice-Based Cryptography: A Survey*

Xiaoyun WANG[1] Guangwu XU[2] Yang YU[1]

Abstract Most of current public key cryptosystems would be vulnerable to the attacks of the future quantum computers. Post-quantum cryptography offers mathematical methods to secure information and communications against such attacks, and therefore has been receiving a significant amount of attention in recent years. Lattice-based cryptography, built on the mathematical hard problems in (high-dimensional) lattice theory, is a promising post-quantum cryptography family due to its excellent efficiency, moderate size and strong security. This survey aims to give a general overview on lattice-based cryptography. To this end, the authors begin with the introduction of the underlying mathematical lattice problems. Then they introduce the fundamental cryptanalytic algorithms and the design theory of lattice-based cryptography.

Keywords Lattices, Post-Quantum cryptography, Algorithms
2000 MR Subject Classification 11H06, 94A60, 68Q25

1 Introduction

The security of a public key cryptosystem relies on the intractability of its underlying mathematical hard problems. Nowadays most of deployed public key cryptosystems are based on integer factorization and discrete logarithms over finite fields and elliptic curves. In his seminal work (see [46]), Shor proposed a polynomial-time quantum algorithm solving these problems. This implies that the current public key cryptosystem would be no longer secure once large-scale quantum computers are built. To protect information systems against quantum attacks, now there are two main approaches: Quantum key distribution (QKD for short) and post-quantum cryptography (PQC for short). QKD is based on quantum physics; it achieves the information-theoretic security but cannot offer the authentication function. In addition, QKD relies on specialized devices and its wide applications seem to be impractical at the moment. By contrast, PQC is based on the hardness of mathematical problems and able to provide the computational security for both confidentiality and authentication. More importantly, PQC is well compatible with the current computing architecture and has been practically efficient. Therefore, PQC is believed to be a more economic and versatile solution than QKD.

Manuscript received August 21, 2023. Revised August 29, 2023.
[1]Institute for Advanced Study, Tsinghua University, Beijing 100084, China.
E-mail: xiaoyunwang@mail.tsinghua.edu.cn yu-yang@mail.tsinghua.edu.cn
[2]School of Cyber Science and Technology, Shandong University, Qingdao 266237, Shandong, China.
E-mail: gxu4sdq@sdu.edu.cn
*This work was supported by the National Key Research and Development Program of China (No. 2018YFA0704701), the National Natural Science Foundation of China (Nos. 12271306, 62102216, 12226006), the Major Program of Guangdong Basic and Applied Research (No. 2019B030302008), the Major Scientific and Technological Innovation Project of Shandong Province (No. 2019JZZY010133) and Shandong Key Research and Development Program (No. 2020ZLYS09).

With the advent of quantum computers, post-quantum cryptography has been receiving a great amount of attention from academia, industry and government in recent years. In 2016, the US National Institute of Standards and Technology (NIST for short) initiated the PQC standardization project calling for post-quantum public key encryption, key establishment and digital signature algorithms. After three rounds of competition, the NIST announced the first four PQC algorithms to be standardized in 2022. The Chinese Association for Cryptologic Research also organized a national cryptographic algorithm design competition in 2019 and its public key cryptography track focused on post-quantum cryptography.

There are now several main PQC families as per the different mathematical problems, including lattice-based, code-based, multivariate-based, isogeny-based and symmetric cipher/hash-based schemes. This paper focuses on lattice-based cryptography that is one of the most promising PQC candidate, due to the following advantages:

Strong Security Gurantees. The security foundation of lattice-based cryptography is the mathematical hard problems in lattice theory, particularly SVP (shortest vector problem) and CVP (closest vector problem). It is worth noting that the underlying lattices of lattice-based cryptosystems are generally of a high dimension, say several hundred, while a great amount of researches in lattice theory mainly focuses on low-dimensional lattices, e.g. the E8 lattice (see [50]) and the Leech lattice (see [9, 59]). From the computational complexity aspects, both SVP and CVP are shown to be NP-hard (see [2, 49]) and lattice-based cryptosystems can be provably secure under the worst-case hardness assumptions of these problems. This provides a solid theoretical grounding.

Good Overall Performance. Lattice-based schemes have excellent speed comparable to, even better than, the widely deployed public key algorithms. While lattice-based cryptography are still of larger sizes than RSA and ECC cryptosystems, its bandwidth is moderate among current post-quantum cryptosystems. The overall performance of lattice-based cryptography is sufficient for most real-world applications.

Powerful Versatility. In contrast with other post-quantum families, lattice-based cryptography can provide both practical public encryption, key encapsulation and digital signatures. In addition, based on lattices, one can also construct powerful advanced cryptographic primitives, e.g. fully homomorphic encryption (see [20]), attribute based encryption (see [23]), code obfuscation (see [19]) and much more. Therefore lattice-based cryptography can be made to support various usecases.

Roadmap This survey is expected to provide a preliminary overview of lattice-based cryptography with an emphasize on the underlying mathematics. In the rest of the paper, we start with an introduction to the hard problems of lattice-based cryptography in Section 2. Then we recap the fundamental cryptanalytic tools for lattice-based cryptography in Section 3. Section 4 shows some classical design paradigms of lattice-based cryptosystems. We conclude in Section 5.

2 Lattices and Hard Problems

2.1 Notations

For a positive integer q, let $\mathbb{Z}_q = \{-\lfloor \frac{q}{2} \rfloor, -\lfloor \frac{q}{2} \rfloor + 1, \cdots, q - \lfloor \frac{q}{2} \rfloor - 1\}$. We write $x \leftarrow D$ to

represent the sample x drawn from the distribution D. Given a finite set S, let $U(S)$ be the uniform distribution over S.

2.2 Lattices

Lattices were first introduced by Carl Friedrich Gauss for studying the sphere packing problem. In general, a lattice Λ is a discrete subgroup of the Euclidean space $\mathbb{R}^m$ and is generated by some matrix $\mathbf{B} = (\mathbf{b}_1, \mathbf{b}_2, \cdots, \mathbf{b}_n) \in \mathbb{R}^{m\times n}$, i.e.,

$$\Lambda = \Lambda(\mathbf{B}) = \{\mathbf{Bz} \mid \mathbf{z} \in \mathbb{Z}^n\}.$$

When $\mathbf{B}$ has full column rank, $\mathbf{B}$ is called a basis of $\Lambda(\mathbf{B})$ and n is called the dimension. Any lattice of dimension ≥ 2 has infinitely many bases. Figure 1 shows an example of 2-dimensional lattices. Given the lattice $\Lambda(\mathbf{B})$, a matrix $\mathbf{B}' \in \mathbb{R}^{m\times n}$ is a basis if and only if $\mathbf{B}' = \mathbf{BU}$ for some unimodular $\mathbf{U} \in \mathbb{Z}^{n\times n}$.

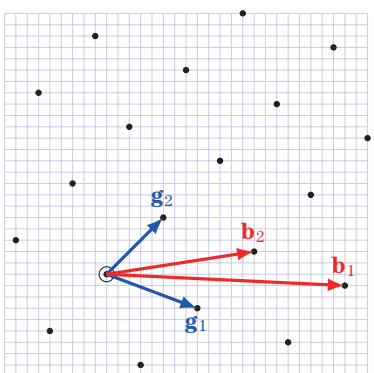

Figure 1 An example of a 2-dimensional lattice Λ. The matrices $(\mathbf{b}_1, \mathbf{b}_2)$ and $(\mathbf{g}_1, \mathbf{g}_2)$ are two bases of Λ.

For $\mathbf{B} = (\mathbf{b}_1, \cdots, \mathbf{b}_n)$, let π_i denote the orthogonal projection to the span of $(\mathbf{b}_1, \cdots, \mathbf{b}_{i-1})^{\perp}$. Let $\mathbf{b}_i^* = \pi_i(\mathbf{b}_i)$, $\mathbf{B}^* = (\mathbf{b}_1^*, \cdots, \mathbf{b}_n^*)$ be the Gram-Schmidt orthogonalization of $\mathbf{B}$. Let $\mathbf{B}_{[i,j]} = (\pi_i(\mathbf{b}_i), \cdots, \pi_i(\mathbf{b}_j))$.

The volume of $\Lambda(\mathbf{B})$ is defined as $\mathrm{vol}(\Lambda(\mathbf{B})) = \sqrt{\det(\mathbf{B}^t\mathbf{B})} = \prod_i \|\mathbf{b}_i^*\|$. Such a volume is an invariant of the lattice, as $\det(\mathbf{B}^t\mathbf{B}) = \det(\mathbf{B}'^t\mathbf{B}')$ for any given $\mathbf{B}' = \mathbf{BU}$ with a unimodular $\mathbf{U}$.

Another important invariant of lattices is successive minima. Given an n-dimensional lattice Λ, the i-th ($i \leq n$) successive minimum $\lambda_i(\Lambda)$ is the smallest $r > 0$ such that there are at least i linearly independent vectors of Λ of norm $\leq r$. Minkowski's theorem gives some upper bounds of the successive minima.

Theorem 2.1 (Minkowski's theorem) *For any n-dimensional lattice Λ,*

$$\lambda_1(\Lambda) \leq \Big(\prod_{i=1}^{n} \lambda_i(\Lambda)\Big)^{\frac{1}{n}} \leq \sqrt{n} \cdot \mathrm{vol}(\Lambda)^{\frac{1}{n}}.$$

The distance from the target $\mathbf{t}$ to the lattice Λ is

$$\mathrm{dist}(\mathbf{t}, \Lambda) = \min_{\mathbf{v} \in \Lambda} \|\mathbf{v} - \mathbf{t}\|.$$

In general, we consider the case $\mathbf{t} \in \mathrm{span}(\Lambda)$.

2.3 SVP and CVP

While lattices have a simple geometric description, they come with many mathematical hard problems. SVP (Shortest Vector Problem) and CVP (Closest Vector Problem) are two most fundamental lattice hard problems.

Definition 2.1 (SVP) *Given a lattice basis* $\mathbf{B}$*,* SVP *asks to find* $\mathbf{v} \in \Lambda(\mathbf{B})$ *such that*

$$\|\mathbf{v}\| = \lambda_1(\Lambda(\mathbf{B})).$$

Definition 2.2 (CVP) *Given a lattice basis* $\mathbf{B}$ *and a target* $\mathbf{t} \in \mathrm{span}(\mathbf{B})$*,* CVP *asks to find* $\mathbf{v} \in \Lambda(\mathbf{B})$ *such that*

$$\|\mathbf{v} - \mathbf{t}\| = \mathrm{dist}(\mathbf{t}, \Lambda(\mathbf{B})).$$

The exact SVP and CVP have been proved to be NP-hard (see [2, 49]). In lattice-based cryptography, the exact SVP and CVP are rarely used as the security foundation directly. Most of lattice-based schemes actually correspond to some variants of SVP and CVP. The first class of SVP and CVP is their approximate versions defined as follows.

Definition 2.3 (SVP_γ) *Given a lattice basis* $\mathbf{B}$*,* SVP_γ *asks to find* $\mathbf{v} \in \Lambda(\mathbf{B})$ *such that*

$$\|\mathbf{v}\| \leq \gamma \cdot \lambda_1(\Lambda(\mathbf{B})).$$

Definition 2.4 (SVP_γ) *Given a lattice basis* $\mathbf{B}$ *and a target* $\mathbf{t} \in \mathrm{span}(\mathbf{B})$*,* CVP_γ *asks to find* $\mathbf{v} \in \Lambda(\mathbf{B})$ *such that*

$$\|\mathbf{v} - \mathbf{t}\| \leq \gamma \cdot \mathrm{dist}(\mathbf{t}, \Lambda(\mathbf{B})).$$

The second class is some relaxed variants in which some extra restrictions are added. Two representative examples are the USVP (Unique Shortest Vector Problem) for SVP and the BDD (Bounded Distance Decoding) problem for CVP.

Definition 2.5 (USVP_γ) *Given a lattice basis* $\mathbf{B}$ *such that* $\frac{\lambda_2(\Lambda(\mathbf{B}))}{\lambda_1(\Lambda(\mathbf{B}))} \geq \gamma$*,* USVP_γ *asks to find* $\mathbf{v} \in \Lambda(\mathbf{B})$ *such that*

$$\|\mathbf{v}\| = \lambda_1(\Lambda(\mathbf{B})).$$

Definition 2.6 (BDD_γ) *Given a lattice basis* $\mathbf{B}$ *and a target* $\mathbf{t} \in \mathrm{span}(\mathbf{B})$ *such that* $\mathrm{dist}(\mathbf{t}, \Lambda(\mathbf{B})) \leq \lambda_1 \frac{(\Lambda(\mathbf{B}))}{(2 \cdot \gamma)}$*,* BDD_γ *asks to find* $\mathbf{v} \in \Lambda(\mathbf{B})$ *such that*

$$\|\mathbf{v} - \mathbf{t}\| = \mathrm{dist}(\mathbf{t}, \Lambda(\mathbf{B})).$$

While above SVP and CVP variants cannot be proved to be NP-hard for any $\gamma = \mathrm{poly}(n)$, it is generally believed that no subexponential algorithm solves these variants for such a polynomial γ. This is the security grounding of lattice-based cryptography.

2.4 Hard problems for lattice-based cryptography

The SVP and CVP problems are not convenient for straightforward constructions of lattice-based schemes. Some design-friendly hard problems, including SIS, LWE and NTRU, were later introduced for lattice-based cryptography. These problems and their variants have been the foundation of modern lattice-based cryptography. Next we give some preliminary descriptions to these problems.

SIS (Short Integer Solution) problem The SIS problem was introduced and studied in Ajtai's breakthrough work (see [1]). Ajtai showed that the average-case SIS problem is at least as hard as the worst-case approximate SVP problem. SIS is the first hard problem for lattice-based cryptography of the worst-case/average-case reduction, which provides a strong security guaranttee for lattice-based schemes. Since its introduction, SIS has been used as the foundations of many lattice-based primitives, e.g. hash functions and digital signatures.

Definition 2.7 (SIS) *For integers $n, m, q > 0$ and a real number $B > 0$,*

– the $\mathrm{SIS}_{n,m,q,B}$ *problem is as follows: Given uniformly random $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, find some non-zero $\mathbf{x} \in \mathbb{Z}^m$ such that $\|\mathbf{x}\| \leq B$ and $\mathbf{Ax} = \mathbf{0} \bmod q$.*

– The $\mathrm{SIS}^{\infty}_{n,m,q,B}$ *problem is as follows: Given uniformly random $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, find some non-zero $\mathbf{x} \in \mathbb{Z}^m$ such that $\|\mathbf{x}\|_\infty \leq B$ and $\mathbf{Ax} = \mathbf{0} \bmod q$.*

Remark 2.1 In general, $B \in (\sqrt{m} \cdot q^{\frac{n}{m}}, q)$ (resp. $(q^{\frac{n}{m}}, q)$) for $\mathrm{SIS}_{n,m,q,B}$ (resp. $\mathrm{SIS}^{\infty}_{n,m,q,B}$). This ensures the hardness of the SIS problem and the existence of the solution. For fixed (n, m, q), the larger B is, the easier the SIS problem is.

LWE (Learning With Errors) problem Another cornerstone of lattice-based cryptography is the introduction of the LWE problem by Oded Regev [41]. The average-case LWE can also be shown to be at least as hard as some worst-case lattice problems. Different from the SIS problem, LWE is able to be used to construct lattice-based public key encryption and much more cryptosystems. This further enriches the lattice-based cryptography.

Definition 2.8 (LWE) *For integers $n, m, q > 0$ and a distribution χ over $\mathbb{Z}$, let $A_{\mathbf{s},\chi}$ for given $\mathbf{s} \in \mathbb{Z}^n$ be the distribution of $(\mathbf{a}, b = \langle \mathbf{a}, \mathbf{s}\rangle + e \bmod q)$ where $\mathbf{a} \leftarrow U(\mathbb{Z}_q^n)$ and $e \leftarrow \chi$.*

– The decision $\mathrm{LWE}_{n,m,q,\chi}$ *problem is as follows: Given m samples drawn from $A_{\mathbf{s},\chi}$ where $\mathbf{s} \leftarrow U(\mathbb{Z}_q^n)$ and m samples from $U(\mathbb{Z}_q^n \times \mathbb{Z}_q)$, distinguish them.*

– The search $\mathrm{LWE}_{n,m,q,\chi}$ *problem is as follows: Given m samples drawn from $A_{\mathbf{s},\chi}$ where $\mathbf{s} \leftarrow U(\mathbb{Z}_q^n)$, find $\mathbf{s}$.*

Remark 2.2 The terms $\mathbf{s}$ and e are called the LWE secret and error respectively. The size of the LWE error, measured by the standard deviation of χ, is usually greatly smaller than the modulus q. For fixed (n, m, q), the smaller the error size is, the easier the LWE problem is. In many cryptographic applications, the LWE secret needs to be small, typically drawn from χ. This modification does not affect the hardness of the LWE problem as shown in [4].

NTRU problem NTRU is also one of the most important and widely used hard problem

for lattice-based cryptography. Unlike SIS and LWE, the researches on NTRU originated from practical designs (see [24–25]) rather than theoretical reductions. NTRU-based schemes are the first practical lattice-based cryptosystems, and particularly the NTRU-based encryption are still one of the most competitive post-quantum candidates until now. In addition, NTRU is also the first lattice-based cryptography problem using the polynomial ring structure. After more than 10 years since its proposal, some theoretical hardness results for NTRU have been successively proved in [15, 38, 47, 56].

Definition 2.9 (NTRU) *For integer $q > 0$, $\mathcal{R} = \mathbb{Z}[X]/(\phi(X))$ with $\phi(X) \in \mathbb{Z}[X]$ and a distribution χ over $\mathcal{R}$, let D_χ be the distribution of $h = f \cdot g^{-1} \bmod q$ where $f, g \leftarrow \chi$ and $\mathcal{R}_q = \frac{\mathcal{R}}{q\mathcal{R}}$.*

– The decision $\mathrm{NTRU}_{\mathcal{R},q,\chi}$ problem is defined as follows: Given samples from D_χ and from $U(\mathcal{R}_q)$, distinguish them.

– The search $\mathrm{NTRU}_{\mathcal{R},q,\chi}$ problem is defined as follows: Given $h \leftarrow D_\chi$, find short (f, g) such that $h = f \cdot g^{-1} \bmod q$.

Remark 2.3 In general, h serves as the public key of the NTRU-based cryptosystems and (f, g) as the secret key. The underlying equation of NTRU is $0 = hg - f \bmod q$, therefore NTRU may be seen as a special case of LWE (over polynomial rings) with $b = 0$ by identifying (f, g) as the LWE secret and error. Similar to the case of LWE, (f, g) has a size $o(\sqrt{nq})$ and the shorter (f, g) implies easier NTRU instances. However, NTRU has many short solutions, e.g. $(x^i f, x^i g)$, which is different from LWE.

Algebraic SIS and LWE Inspired by NTRU, SIS and LWE can be instantiated over some polynomial rings to achieve smaller sizes and better efficiency. This yields the algebraic SIS and LWE, e.g. Ring-SIS/LWE and Module-SIS/LWE (see [27, 33, 48]). These algebraic variants can be reduced to the worst-case lattice problems over ideal lattices and module lattices. The SVP_γ problem over ideal lattices has been shown to be not as hard as the standard SVP_γ for some super-polynomial γ (see [10, 37]). Nevertheless, the algebraic versions of SVP_γ with polynomial γ are believed to be essentially as hard as the standard SVP_γ.

3 Cryptanalysis of Lattice-Based Cryptography

Cryptanalysis aims to provide a reliable analysis of the security of cryptosystems and is crucial to concrete security estimates, parameter selections and avoiding weak designs. This boils down to solving the underlying hard problems of the cryptosystem under given adversary models. Figure 2 shows a general procedure of the cryptanalysis of a lattice-based cryptosystem. At the core of the cryptanlaysis of lattice-based cryptography are two kinds of SVP algorithms:

– Approximate SVP algorithms: Lattice reduction, e.g. LLL and BKZ.
– Exact SVP algorithms: Enumeration and sieving.

These two kinds of SVP algorithms are usually used together. Next we briefly introduce these cryptanalytic algorithms.

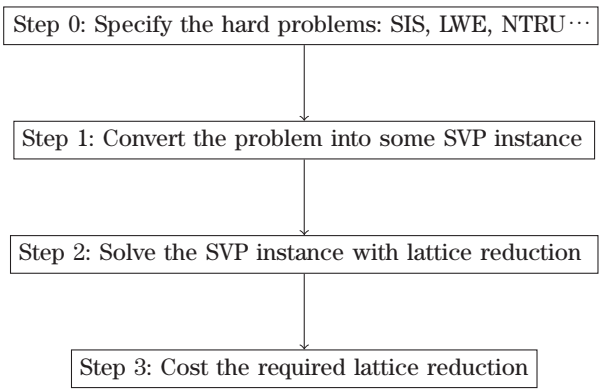

Figure 2 A general flowchart of cryptanalysis of some lattice-based scheme.

3.1 Approximate SVP algorithms – lattice reduction

For almost all practical lattice schemes, a standard method to solve the underlying SIS, LWE and NTRU problems is converting these problems into certain approximate SVP instances. The most efficient and widely used approximate SVP algorithms are lattice reduction. The goal of lattice reduction is to transform a basis into a high-quality one, i.e., a basis consisting of short and nearly orthogonal vectors. For a basis of sufficiently high quality, its vector would be the solution of the approximate SVP instances. Additionally, a high-quality basis allows to solve SVP and CVP with a relatively low cost. For this, lattice reduction also serves as the preprocessing of exact SVP algorithms.

LLL The LLL algorithm, invented by Lenstra, Lenstra, and Lovász in 1982 (see [28]), is a pioneering lattice reduction algorithm. It outputs an LLL-reduced basis defined as follows.

Definition 3.1 *A basis* $\mathbf{B}$ *is* δ-LLL-*reduced with* $\delta \in \left(\frac{1}{2}, 1\right)$, *if*:

(1) $|\mu_{i,j}| \leq \frac{1}{2}$ *for* $1 \leq j < i \leq n$, *where* $\mu_{i,j} = \langle \mathbf{b}_i, \mathbf{b}_j^* \rangle / \langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle$;

(2) $\delta \|\mathbf{b}_i^*\| \leq \|\mathbf{b}_{i+1}^* + \mu_{i+1,i}\mathbf{b}_i^*\|$ *for* $1 \leq i < n$.

LLL can be viewed as a high-dimensional generalization of the Euclidean algorithm. It proceeds by successively reducing the projected lattice $\mathbf{B}_{[i,i+1]}$ until the basis is LLL-reduced. Algorithm 1 gives a description of the LLL algorithm. It is shown that the LLL algorithm terminates in polynomial time and the output basis satisfies

$$\|\mathbf{b}_1\| \leq \mathrm{vol}(\Lambda)^{\frac{1}{n}} \cdot \left(\frac{4}{4\delta - 1}\right)^{\frac{n-1}{4}}.$$

Algorithm 1: The LLL Algorithm

Input: a basis $\mathbf{B} = (\mathbf{b}_1, \cdots, \mathbf{b}_n)$ and $\delta \in (\frac{1}{2}, 1)$
Output: a δ-LLL-reduced basis of $\Lambda(\mathbf{B})$

```
compute (b_1^*, ..., b_n^*)
for i = 2 to n do
    for j = i − 1 to 1 do
        b_i ← b_i − c_{i,j} b_j where c_{i,j} = ⌊⟨b_i, b_j^*⟩/⟨b_j^*, b_j^*⟩⌉
        if ∃i such that δ‖b_i^*‖ > ‖b_{i+1}^* + μ_{i+1,i} b_i^*‖ then
            swap (b_i, b_{i+1})
            go to Step 1
        else
            return (b_1, ..., b_n)
        end if
    end for
end for
```

BKZ The BKZ algorithm (see [43]) and its practical variants (see [8]) are the most efficient lattice reduction algorithms and widely used in the concrete security estimates of lattice-based cryptography. BKZ is a generalization of LLL. Instead of reducing the 2-dimensional block $\mathbf{B}_{[i,i+1]}$, BKZ performs lattice reduction and SVP algorithm on the β-dimensional block $\mathbf{B}_{[i,i+\beta]}$ where β is called the blocksize. BKZ is much more expensive than LLL. Its cost is estimated as $\mathrm{poly}(n) \cdot \mathrm{Cost}_{\mathrm{SVP}}(\beta)$ where $\mathrm{Cost}_{\mathrm{SVP}}(\beta)$ denotes the cost of the β-dimensional SVP algorithm. For the best known SVP algorithm, $\mathrm{Cost}_{\mathrm{SVP}}(\beta) = 2^{O(\beta)}$ and this dominates the cost of BKZ for some large β. However, compared to LLL, BKZ is able to output a shorter basis with

$$\|\mathbf{b}_1\| \approx \mathrm{vol}(\Lambda)^{\frac{1}{n}} \cdot \delta_\beta^n \quad \text{and} \quad \delta_\beta \approx \Big(\frac{(\pi\beta)^{\frac{1}{\beta}}\beta}{2\pi\mathrm{e}}\Big)^{\frac{1}{2(\beta-1)}}$$

when $n \gg \beta > 50$.

3.2 Exact SVP algorithms – enumeration and sieving

BKZ requires to call some exact SVP algorithm on each local block. The cost of the used SVP algorithm determines the cost of the whole BKZ. Enumeration and sieving are two main exact SVP algorithms.

Enumeration Typically, enumeration algorithms run in super-exponential time and with polynomial space. The high-level idea of enumeration is to compute short vectors by searching all possible integer coefficient combinations of a given lattice basis, based on a simple fact that $\|\pi_i(\mathbf{v})\| \leq \|\mathbf{v}\|$. In the early works, Pohst gave a first enumeration algorithm of complexity $2^{O(n^2)}$ (see [39]) and Kannan proposed to apply lattice reduction as the preprocess to reduce the enumeration cost down to $2^{O(n \log n)}$ (see [26]). Later, by using the pruning strategies (see [18, 43–44]) and improved representations of short vectors (see [11, 52, 58]), the practical performance of enumeration has been greatly improved but the overall complexity is still of complexity shape $2^{O(n \log n)}$.

Sieving In contrast with enumeration, sieving algorithms run in $2^{O(n)}$ time and with $2^{O(n)}$ space. The geometric intuition behind sieving is that given sufficiently many vectors on a

sphere, by reducing them pair-wise, one can always find a shorter vector. The early sieving algorithm only had an asymptotic complexity analysis (see [3]). Nguyen and Vidick presented the first heuristic sieving algorithm of time $2^{0.415n+o(n)}$ and space $2^{0.2075n+o(n)}$ (see [36]). Wang et al. proposed to use partially pairwise checking to achieve better time-space tradeoff: Time $2^{0.3836n+o(n)}$ and space $2^{0.2557n+o(n)}$ (see [51]). Inspired by this work, the complexity of sieving gets further improved and the best known result (see [5])

$$\text{Time: } 2^{0.292n+o(n)}, \quad \text{Space: } 2^{0.292n+o(n)}$$

have become the primary cost model for the current security estimates of lattice-based cryptography.

4 Practical Designs of Lattice-Based Cryptography

A large number of lattice-based schemes of practical performance have been proposed in the past decade, making lattice-based cryptography a desirable post-quantum alternative to the current public key cryptosystems. Most of these schemes are constructed using several generic design paradigms along with individual tweaks. In this section, we recall the main design paradigms of lattice-based encryption/KEM (key encapsulation mechanism) and signature schemes.

4.1 Lattice-based encryption and KEM schemes

The design modes for most practical lattice-based encryption schemes can be classified into two types: The LWE encryption and the NTRU encryption. As for lattice-based KEMs, the standard design is to apply the Fujisaki-Okamoto transformation (see [17]) or its variants to convert a weakly secure lattice-based encryption scheme to a lattice-based KEM of stronger security. For this, we only introduce the design paradigms of lattice-based encryption.

LWE encryption The first LWE encryption was proposed by Oded Regev in [41], but it is inefficient in practice. Currently, the widely used LWE encryption framework is due to Lindner and Peikert [29]. We now describe the Lindner-Peikert LWE encryption in a simplistic manner. Let χ be the used LWE error distribution.

– **Key Generation** Generate $(\mathbf{A}, \mathbf{s}, \mathbf{e}) \leftarrow U(\mathbb{Z}_q^{m\times n}) \times \chi^n \times \chi^m$ and compute $\mathbf{b} = \mathbf{As} + \mathbf{e} \bmod q$. Return the public key $(\mathbf{A}, \mathbf{b})$ and the secret key $\mathbf{s}$.

– **Encryption.** Given message $m \in \{0,1\}$, sample $(\mathbf{u}, \mathbf{e}_1, e_2) \leftarrow \chi^m \times \chi^n \times \chi$, compute $(\mathbf{c}_1, c_2) = (\mathbf{A}^t\mathbf{u} + \mathbf{e}_1, \mathbf{b}^t\mathbf{u} + e_2 + \lfloor \frac{q}{2} \rceil m) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$. Return the ciphertext $\mathbf{c} = (\mathbf{c}_1, c_2)$.

– **Decryption** Given ciphertext $\mathbf{c} = (\mathbf{c}_1, c_2)$, compute $M = c_2 - \langle \mathbf{c}_1, \mathbf{s} \rangle \bmod q$. Return the message $m = \lfloor \frac{2}{q} \cdot M \rceil$.

It is easy to check that

$$M = \left\lfloor \frac{q}{2} \right\rceil m + (e_2 + \mathbf{e}^t\mathbf{u} - \mathbf{s}^t\mathbf{e}_1) := \left\lfloor \frac{q}{2} \right\rceil m + E.$$

In the Lindner-Peikert encryption, $(\mathbf{s}, \mathbf{e}, \mathbf{u}, \mathbf{e}_1, e_2)$ are drawn from χ and thus are small. By selecting proper parameters, $|E| < \frac{q}{4}$ with overwhelming probability, which ensures the correct decryption. In fact, this decryption can be seen as an error (i.e., E) correction procedure. Therefore, the message encoding $(\lfloor \frac{q}{2} \rceil m)$ can be implemented with other codes. Under the LWE assumption, the public key $(\mathbf{A}, \mathbf{b})$ and the ciphertext $(\mathbf{c}_1, c_2)$ are indistinguishable from uniform samples, then the security follows. Using various LWE variants and message encoding, the LWE encryption can achieve different tradeoff among efficiency, security and simplicity. The representative algorithms include

– Kyber. Kyber (see [45]) is based on Module-LWE and uses a simple and modular designs. It has been selected by NIST for the PQC standardization.

– SCloud. SCloud (see [57]) is based on the standard LWE assumption, avoiding the potential weakness of algebraic structures. SCloud uses some error correction code technique to obtain better performance than Frodo (see [34]) that is the standard LWE-based encryption in the NIST third round.

NTRU encryption The NTRU encryption was the first practical lattice-based scheme proposed by Hoffstein, Pipher and Silverman [25]. A typical NTRU encryption scheme is specified by the ring $\mathcal{R}$, the modulus q and the masking modulus p. The following is an example for $\mathcal{R} = \mathbb{Z}[x]/(x^n + 1)$ with $n = 2^k$, q being a prime and $p = 3$. Let χ be the NTRU secret key distribution.

– **Key Generation** Generate $(f, g) \leftarrow \chi^2$ and compute $g' = pg + 1$ and $h = \frac{pf}{g'} \bmod q$. Return the public key h and the secret key g'.

– **Encryption** Given message $m \in \mathcal{R}_p$, sample $r \leftarrow \chi$, return the ciphertext $c = hr + m \bmod q$.

– **Decryption** Given ciphertext c, compute $M = cg' \bmod q$. Return the message $m = M \bmod p$.

The correctness of the NTRU encryption follows from the fact that

$$cg' = pfr + g'm = pfr + pgm + m \bmod q$$

and that $M = pfr + pgm + m$ when (f, g, p, r, m) are short polynomials. The public key h is indistinguishable from uniform under the NTRU assumption, and the ciphertext c along with h can be seen as a Ring-LWE sample with secret r and error m. This gives the security of the NTRU encryption scheme. The representative NTRU-based encryption algorithms include

– NTRU. NTRU (see [7]) is one of four finalists in the NIST third round. It is built on the prime cyclotomic rings following the classic NTRU design.

– NTRUPrime. NTRUPrime (see [6]) is an alternate candidates. To reduce the security risk, NTRUPrime uses $\mathbb{Z}[x]/(x^p - x - 1)$ to replace widely used cyclotomic rings.

– BAT. Different from other NTRU schemes, BAT (see [16]) uses a full basis as the secret key and works with a modified decryption allowing the masking modulus $p = 1$. For this, BAT

has the smallest size among known lattice-based schemes and good efficiency comparable to Kyber.

4.2 Lattice-based signature schemes

Practical lattice-based signature schemes have two main families: The hash-and-sign family and the Fiat-Shamir one. Two families come with very distinct design paradigms, and both have pratical instantiations based on various lattice hard problem assumptions, which is different from the case of lattice-based encryption.

Hash-and-sign The hash-and-sign family dates back to the earliest lattice-based signature proposals GGH (see [22]) and NTRUSign (see [24]). The hash-and-sign lattice-based signatures use a lattice trapdoor, i.e., a high-quality basis, as the secret key. With a lattice trapdoor, one can efficiently solve the approximate CVP on the lattice. This actually corresponds to the signing procedure: One first derives a random target in the ambient space via computing the hash value of the message, and then computes the signature that is a lattice point close to the target by using the trapdoor. A hash-and-sign scheme can be described at a very high level as follows.

– **Key Generation** Generate a lattice $\Lambda \subseteq \mathbb{Z}^n$ with a trapdoor $\mathbf{T}$ and a public representation $\mathbf{P}$ of Λ. Return the public key $\mathbf{P}$ and the secret key $\mathbf{T}$.

– **Signing** Given message m, compute $\mathbf{c} = \mathsf{hash}(m) \in \mathbb{Z}^n$. Compute $\mathbf{v} \in \Lambda$ close to $\mathbf{c}$ using $\mathbf{T}$. Return the signature $\mathbf{s} = \mathbf{v} - \mathbf{c}$.

– **Verification** Given message m and its signature $\mathbf{s}$, compute $\mathbf{c} = \mathsf{hash}(m)$. Accept if $\mathbf{s}$ is short and $\mathbf{s} + \mathbf{c} \in \Lambda$, otherwise reject.

Without the knowledge of the trapdoor, one cannot solve the approximate CVP instance regarding to $(\Lambda, \mathbf{c})$. This gives the security against forgery attacks. In early hash-and-sign schemes, the signatures leaked some information of the trapdoor, which was exploited to mount statistical attacks (see [13, 35, 53]). The modern designs follow the provably secure framework by Gentry, Peikert and Vaikutanathan [21], in which the signature distribution is some discrete Gaussian independent of the trapdoor. The GPV hash-and-sign signatures can be classified into two families: NTRU trapdoor based and gadget trapdoor based. The representative algorithms are as follows:

– Falcon. Falcon (see [40]) is one of three signature algorithms to be standardized by NIST. It uses the compact NTRU trapdoor (see [12]) along with a ring-efficient lattice Gaussian sampler, which offers competitive efficiency in both size and speed. More recently, a Falcon variant, named Mitaka (see [14]), was proposed to overcome the complicated implementation while maintaining the good performance.

– HuFu. HuFu (see [54]) makes use of the compact gadget techniques (see [55]) and is based on the standard LWE assumptions to avoid potential security risk caused by algebraic structures. Very recently, HuFu gets in the first round for additional NIST PQC signatures.

Fiat-Shamir The Fiat-Shamir signature paradigm was invented by Lyubashevsky [30–31].

Its signing procedure is a non-interactive zero-knowledge proof that the signer knows the secret short vector. At a high-level, Fiat-Shamir signatures is similar to the Schnorr signature (see [42]) based on the discrete logarithm problem. However, the key technical point is that in the context of lattices, Fiat-Shamir signatures need to use some rejection sampling to ensure the signature distribution leaking nothing about the secret. We present a simplistic description of Fiat-Shamir signatures. Let χ be some distribution of small elements.

– **Key Generation** Generate $\mathbf{A} \leftarrow U(\mathbb{Z}_q^{n\times m})$ and $\mathbf{S} \in \mathbb{Z}^{m\times k}$ of small coefficients. Return the public key $(\mathbf{A}, \mathbf{T} = \mathbf{AS})$ and the secret key $\mathbf{S}$.

– **Signing** Given message m, sample $\mathbf{y} \leftarrow \chi^m$, compute $\mathbf{d} = (\mathbf{Ay} \bmod q)$ and $\mathbf{c} = \mathsf{hash}(\mathbf{d}, m)$ where the hash domain is a set of short vectors. Return $(\mathbf{z} = \mathbf{Sc}+\mathbf{y}, \mathbf{c})$ with certain probability, otherwise restart.

– **Verification** Given message m and its signature $(\mathbf{z}, \mathbf{c})$, compute $\mathbf{d} = (\mathbf{Az} - \mathbf{Tc} \bmod q)$. Accept if $\mathbf{c} = \mathsf{hash}(\mathbf{d}, m)$ and $\mathbf{z}$ is short, otherwise reject.

In the Fiat-Shamir signatures, the used hash function is modeled as a random oracle and the distribution of $\mathbf{z}$ is independent of the secret due to the rejection sampling. Therefore one can simulate the signing procedure and derive the solution to the SIS problem from a successful forgery, which gives the security proof. The representative algorithms include

– Dilithium. As the signature run-mate of Kyber, Dilithium (see [32]) is also based on Module-LWE and follows a modular and easy-to-implement designs. Compared to Falcon, Dilithium has larger key and signature sizes but much simpler implementation. For this, Dilithium is selected as the primary signature algorithm for NIST PQC standardization.

5 Conclusion

This survey gives a preliminary introduction to the mathematical hard problems, the main cryptanalytic algorithms and the classical design paradigms of lattice-based cryptography. We hope it to be informative to the readers interested in the mathematics of lattice-based cryptography.

Lattice-based cryptography is the primary family of post-quantum cryptography: Three of four NIST to-be-standardized algorithms are lattice-based schemes. These lattice-based schemes are based on algebraic lattices and have highly mature and sophisticated designs, achieving balanced security and efficiency for most applications. However, it is a pity that the schemes based on standard lattice hard problems have not been selected for standardization. Indeed the schemes based on standard lattice hard problems can offer more convincing security in spite of some efficiency loss, which can be of interest for the cases where stronger provable security is needed.

With the post-quantum standardization and migration underway, lattice-based cryptography will receive continuous and widespread attention in the next decade. More efficient designs and more thorough cryptanalysis of lattice-based cryptography remain challenging scientific problems, requiring more new insights and tools from different mathematical fields. Addition-

ally, lattices can be used to implement various privacy-preserving techniques, in particular fully homomorphic encryption (FHE for short) (see [20]). FHE has wide application perspective given the increasing importance of data security and privacy, but the current FHE algorithms are still impractical for most usecases. Optimizing the performance of FHE would be a promising research direction.

Declarations

Conflicts of interest The authors declare no conflicts of interest.

References

[1] Ajtai, M., Generating hard instances of lattice problems (Extended Abstract), 28th Annual ACM Symposium on Theory of Computing, ACM Press, New York, 1996, 99–108.

[2] Ajtai, M., The shortest vector problem in L2 is NP-hard for randomized reductions (Extended Abstract), 30th Annual ACM Symposium on Theory of Computing, ACM Press, 1998, 10–19.

[3] Ajtai, M., Kumar, R. and Sivakumar, D., A sieve algorithm for the shortest lattice vector problem, 33rd Annual ACM Symposium on Theory of Computing, ACM Press, New York, 2001, 601–610.

[4] Applebaum, B., Cash, D., Peikert, C. and Sahai, A., Fast cryptographic primitives and circular-secure encryption based on hard learning problems, Shai Halevi editor, Advances in Cryptology - CRYPTO 2009, Lecture Notes in Computer Science, **5677**, Springer-Verlag, Heidelberg, 2009, 595–618.

[5] Becker, A., Ducas, L., Gama, N. and Laarhoven, T., New directions in nearest neighbor searching with applications to lattice sieving, Robert Krauthgamer, editor, 27th Annual ACM-SIAM Symposium on Discrete Algorithms, ACM-SIAM, New York, 2016, 10–24.

[6] Bernstein, D., Brumley, B. B., Chen M.-S., et al., NTRU Prime, Technical report, National Institute of Standards and Technology, 2020, https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions.

[7] Chen, C., Danba, O., Hoffstein, J., et al., NTRU, Technical report, National Institute of Standards and Technology, 2020, https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions.

[8] Chen, Y. M. and Nguyen, P. Q., BKZ 2.0: Better lattice security estimates, Dong Hoon Lee and Xiaoyun Wang, editors, Advances in Crytology – ASIACRYPT 2011, Lecture Note in Computer Science, **7073**, Springer-Verlag, Heidelberg, 2011, 1–20.

[9] Cohn, H., Kumar, A., Miller, S., et al., Universal optimality of the E_8 and Leech lattices and interpolation formulas, *Annals of Mathematics*, **196**(3), 2022, 983–1082.

[10] Cramer, R., Ducas, L. and Wesolowski, B., Short stickelberger class relations and application to ideal-SVP, Jean-ébastien Coron and Jesper Buus Nielsen, editors, Advances in Cryptology – EUROCRYPT 2017, Part I, Lecture Notes in Computer Science, **10210**, Springer-Verlag, Heidelberg, 2017, 324–348.

[11] Ding, D., Zhu, G. Z. and Wang, X. Y., A genetic algorithm for searching the shortest lattice vector of SVP challenge, Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation, 2015, 823–830.

[12] Ducas, L., Lyubashevsky, V. and Prest, T., Efficient identity-based encryption over NTRU lattices, Palash Sarkar and Tetsu Iwata, editors, Advances in Cryptology – ASIACRYPT 2014, Part II, Lecture Notes in Computer Science, **8874**, Springer-Verlag, Heidelberg, 2014, 22–41.

[13] Ducas, L. and Nguyen, P. Q., Learning a zonotope and more: Cryptanalysis of NTRUSign countermeasures, Xiaoyun Wang and Kazue Sako, editors, Advances in Crytology – ASIACRYPT 2012, Lecture Notes in Computer Science, **7658**, Springer-Verlag, Heidelberg, 2012, 433–450.

[14] Espitau, T., Fouque, P.-A., Gérard, F., et al., Mitaka: A simpler, parallelizable, maskable variant of falcon, Orr Dunkelman and Stefan Dziembowski, editors, Advances in Cryptology – EUROCRYPT 2022, Part III, Lecture Notes in Computer Science, **13277**, Springer-Verlag, Heidelberg, 2022, 222–253.

[15] Felderhoff, J., Pellet-Mary, A. and Stehlé, D., On module unique-SVP and NTRU, Shweta Agrawal and Dongdai Lin, editors, Advances in Cryptology – EUROCRYPT 2022, Part III, Lecture Notes in Computer Science, **13793**, Springer-Verlag, Heidelberg, 2022, 709–740.

[16] Fouque, P.-A., Kirchner, P., Pornin, T. and Yu, Y., BAT: Small and fast KEM over NTRU lattices, *IACR Transactions on Cry ptographic Hardware and Embedded Systems*, **2022**(2), 2022, 240–265.

[17] Fujisaki, E. and Okamoto, T., How to enhance the security of public-key encryption at minimum cost, Hideki Imai and Yuliang Zheng, editors, PKC'99: 2nd International Workshop on Theory and Practice in Public Key Cryptography, Lecture Notes in Computer Science, **1560**, Springer-Verlag, Heidelberg, 1999, 53–68.

[18] Gama, N., Nguyen, P. Q. and Regev, O., Lattice enumeration using extreme pruning, Henri Gilbert, editor, Advances in Cryptology – EUROCRYPT 2010, Lecture Notes in Computer Science, **6110**, Springer-Verlag, Heidelberg, 2010, 257–278.

[19] Garg, S., Gentry, C., Halevi, S., et al., Candidate indistinguishability obfuscation and functional encryption for all circuits, 54th Annual Symposium on Foundations of Computer Science, IEEE Computer Society Press, Los Alamitos, CA, 2013, 40–49.

[20] Gentry, C., Fully homomorphic encryption using ideal lattices, Michael Mitzenmacher, editor, 41st Annual ACM Symposium on Theory of Computing, ACM Press, New York, 2009, 169–178.

[21] Gentry, C., Peikert, C. and Vaikuntanathan, V., Trapdoors for hard lattices and new cryptographic constructions, Richard E. Ladner and Cynthia Dwork, editors, 40th Annual ACM Symposium on Theory of Computing, ACM Press, New York, 2008, 197–206.

[22] Goldreich, O., Goldwasser, S. and Halevi, S., Public-Key cryptosystems from lattice reduction problems, Burton S. Kaliski Jr., editor, Advances in Cryptology – CRYPTO'97, Lecture Notes in Computer Science, **1294**, Springer-Verlag, Heidelberg, 1997, 112–131.

[23] Gorbunov, S., Vaikuntanathan, V. and Wee, H., Attribute-based encryption for circuits, Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, editors, 45th Annual ACM Symposium on Theory of Computing, ACM Press, New York, 2013, 545–554.

[24] Hoffstein, J., Howgrave-Graham, N., Pipher, J., et al., NTRUSIGN: Digital signatures using the NTRU lattice, Marc Joye, editor, Topics in Cryptology – CT-RSA 2003, Lecture Notes in Computer Science, **2612**, Springer-Verlag, Heidelberg, 2003, 122–140.

[25] Hoffstein, J., Pipher, J. and Silverman, J. H., NTRU: A ring-based public key cryptosyst, ANTS 1998, Lecture Notes in Comput. Sci, **1423**, Springer-Verlag, Berlin, 1998, 267–288.

[26] Kannan, R., Improved algorithms for integer programming and related lattice problems, 15th Annual ACM Symposium on Theory of Computing, ACM Press, 1983, 193–206.

[27] Langlois, A. and Stehlé, D., Worst-case to average-case reductions for module lattices, *Des. Codes Cryptogr.*, **75**(3), 2015, 565–599.

[28] Lenstra, A. K., Lenstra, H. W. and Lovász, L., Factoring polynomials with rational coefficients, *Mathematische Annalen*, **261**(4), 1982, 515–534.

[29] Lindner, R. and Peikert, C., Better key sizes (and attacks) for LWE-based encryption, Aggelos Kiayias, editor, Topics in Cryptology – CT-RSA 2011, Lecture Notes in Computer Scinece, **6558**, Springer-Verlag, Heidelberg, 2011, 319–339.

[30] Lyubashevsky, V., Fiat-Shamir with aborts: Applications to lattice and factoring-based signatures, Mitsuru Matsui, editor, Advances in Cryptology – ASIACRYPT 2009, Lecture Notes in Computer Science, **5912**, Springer-Verlag, Heidelberg, 2009, 598–616.

[31] Lyubashevsky, V., Lattice signatures without trapdoors, David Pointcheval and Thomas Johansson, editors, Advances in Cryptology – EUROCRYPT 2012, Lecture Notes in Computer Scinece, **7237**, Springer-Verlag, Heidelberg, 2012, 738–755.

[32] Lyubashevsky, V., Ducas, L. and Kiltz, E., CRYSTALS-DILITHIUM, Technical report, National Institute of Standards and Technology, 2020, https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions.

[33] Lyubashevsky, V., Peikert, C. and Regev, O., On ideal lattices and learning with errors over rings, Henri Gilbert, editor, Advances in Cryptology – EUROCRYPT 2010, Lecture Notes in Computer Scinece, **6110**, Springer-Verlag, Heidelberg, 2010, 1–23.

[34] Naehrig, M., Alkim, E. and Bos, J., et al., FrodoKEM, Technical report, National Institute of Standards and Technology, 2020, https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions.

[35] Nguyen, P. Q. and Regev, O., Learning a parallelepiped: Cryptanalysis of GGH and NTRU signatures, Serge Vaudenay, editor, Advances in Cryptology – EUROCRYPT 2006, Lecture Notes in Computer Science, **4004**, Springer-Verlag, Heidelberg, 2006, 271–288.

[36] Nguyen, P. Q. and Vidick, T., Sieve algorithms for the shortest vector problem are practical, *Journal of Mathematical Cryptology*, **2**(2), 2008, 181–207.

[37] Pellet-Mary, A., Hanrot, G. and Stehlé, D., Approx-SVP in Ideal Lattices with Pre-processing, Yuval Ishai and Vincent Rijmen, editors, Advances in Cryptology – EUROCRYPT 2019, Part II, Lecture Notes in Computer Science, **11477**, Springer-Verlag, Heidelberg, 2019, 685–716.

[38] Pellet-Mary, A. and Stehlé, D., On the hardness of the NTRU problem, Mehdi Tibouchi and Huaxiong Wang, editors, Advances in Cryptology - ASIACRYPT 2021, Part I, Lecture Notes in Computer Science, Springer-Verlag, Heidelberg, 2021, **13090**, 3–35.

[39] Pohst, M., On the computation of lattice vectors of minimal length, successive minima and reduced bases with applications, *ACM Sigsam Bulletin*, **15**(1), 1981, 37–44.

[40] Prest, T., Fouque, P.-A., Hoffstein, J., et al., FALCON, Technical report, National Institute of Standards and Technology, 2020, https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions.

[41] Regev, O., On lattices, learning with errors, random linear codes, and cryptography, Harold N. Gabow and Ronald Fagin, editors, 37th Annual ACM Symposium on Theory of Computing, ACM Press, New York, 2005, 84–93.

[42] Schnorr, C.-P., Efficient identification and signatures for smart cards, Gilles Brassard, editor, Advances in Cryptology – CRYPTO'89, Lecture Notes in Computer Science, **435**, Springer-Verlag, Heidelberg, 1990, 239–252.

[43] Schnorr, C.-P. and Euchner, M., Lattice basis reduction: Improved practical algorithms and solving subset sum problems, *Math. Program.*, **66**, 1994, 181–199.

[44] Schnorr, C.-P. and Hörner, H. H., Attacking the Chor-Rivest cryptosystem by improved lattice reduction, Louis C. Guillou and Jean-Jacques Quisquarter, editors, Advances in Cryptology – EUROCRYPT'95, Lecture Notes in Computer Science, **921**, Springer-Verlag, Heidelberg, 1995, 1–12.

[45] Schwabe, P., Avanzi, R., Bos, J., et al., CRYSTALS-KYBER, Thchnical report, National Institute of Standards and Technology, 2020, https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions.

[46] Shor, P. W., Algorithms for quantum computation: Discrete logarithms and factoring, 35th Annual Symposium on Foundations of Computer Science, IEEE Computer Society Press, Los Alamitos, CA, 1994, 124–134.

[47] Stehlé, D. and Steinfeld, R., Making NTRU as secure as worst-case problems over ideal lattices, Kenneth G. Paterson, editor, Advances in Cryptology – EUROCRYPT 2011, Lecture Notes in Computer Science, **6632**, Springer-Verlag, Heidelberg, 2011, 27–47.

[48] Stehlé, D., Steinfeld, R., Tanaka, K. and Xagawa, K., Efficient public key encryption based on ideal lattices, Mirsuru Matsui, editor, Advances in Cryptology – ASIACRYPT 2009, Lecture Notes in Computer Science, **5912**, Springer-Verlag, Heidelberg, 2009, 617–635.

[49] van Emde Boas, P., Another NP-complete problem and the complexity of computing short vectors in a lattice, Tecnical Report, Department of Mathmatics, University of Amsterdam, 1981.

[50] Viazovska, M. S., The sphere packing problem in dimension 8, Annals of mathematics (2), **185**(3), 2017, 991–1015.

[51] Wang, X. Y., Liu, M. J., Tian, C. L. and Bi, J. G., Improved Nguyen-Vidick heuristic sieve algorithm for shortest vector problem (Keynote Talk), Bruce S. N. Cheung, Lucas Chi Kwong Hui, Ravi S. Sandhu, and Duncan S. Wong, editors, ASIACCS 11: 6th ACM Symposium on Information, Computer and Communications Security, ACM Press, 2011, 1–9.

[52] Xu, G. W. and Wang, X. Y., Computational aspects of lattices and their cryptographic applications, *Science China* [Ser A], **50**(2020), 2020, 1417–1436 (in Chinese).

[53] Yu, Y. and Ducas, L., Learning strikes again: The case of the DRS signature scheme, Thomas Peyrin and Steven Galbraith, editors, Advances in Cryptology – ASIACRYPT 2018, Part II, Lecture Notes in Computer Science, **11273**, Springer-Verlag, Heidelberg, 2018, 525–543.

[54] Yu, Y., Jia, H. W., Li, L. B., et al., HuFu, Technical report, National Institute of Standards and Technology, 2023, https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures.

[55] Yu, Y., Jia, H. W., and Wang, X. Y., Compact lattice gadget and its applications to hash-and-sign signatures, CRYPTO 2023, 2023, 390–420.

[56] Yu, Y. Xu, G. W., Wang, X. Y., Provably secure NTRU instances over prime cyclotomic rings, Serge Fehr, editor, PKC 2017: 20th International Conference on Theory and Practice of Public Key Cryptography, Part I, Lecture Notes in Computer Scienece, **10174**, Springer-Verlag, Heidelberg, 2017, 409–434.

[57] Zheng, Z. X., Wang, A. Y., Fan, H. N., et al., Scloud: Public key encryption and key encapsulation mechanism based on learning with errors, IACR Cryptol. ePrint Arch., 2020, 95.

[58] Zheng, Z. X., Wang, X. Y., Xu, G. W. and Yu, Y., Orthogonalized lattice enumeration for solving SVP, *Sci. China Inf. Sci.*, **61**(3), 2018, 32115:1–32115:15.

[59] Zong, C. M., What is the leech lattice?, *Notices of the AMS*, **60**(9), 2013, 1168–1169.