0.1 Lattice basics

Lattices are useful mathematical tools for connecting different areas of mathematics, computer science and cryptography. They are widely used for cryptoanalysis and building secure cryptosystems. In this section, [1] we will introduce the basics of lattices in the general setting $\mathbb{R}^n$. In addition, we introduce dual lattices and some computational lattice problems that are commonly used to achieve provable security of lattice-based hard problems and cryptosystems. At the end of this section, we will sketch Ajtai (1996)'s polynomial time worst-case-to-average-case reduction to reinforce our understanding of lattices as well as appreciate the great breakthrough in provable security of lattice-based cryptography, even against quantum computing in some cases. Although we introduce lattices in the most general setting, their results also hold for special lattices such as ideal lattices in the ring learning with error problem.

Intuitively, a lattice is similar to a vector space except that it consists of discrete vectors only, that is, elements in lattice vectors have discrete values as opposed to real-valued vectors in a vector space. For example, Figure 1 is a lattice in $\mathbb{R}^2$. More formally, we have the following definition.

Lattice

Definition 0.1.1. *Let $\mathbf{v_1}, \ldots, \mathbf{v_n} \in \mathbb{R}^m$ be a set of linearly independent vectors. The **lattice** L generated by $\mathbf{v_1}, \ldots, \mathbf{v_n}$ is the set of integer linear combinations of $\mathbf{v_1}, \ldots, \mathbf{v_n}$. That is,*

$$L = \{a_1\mathbf{v_1} + \cdots + a_n\mathbf{v_n} \mid a_1, \ldots, a_n \in \mathbb{Z}\}.$$

Dimension, rank

Here, the difference with vector spaces is that the coefficients in the linear combination are integers. The integers m and n are the **dimension** and **rank** of the lattice respectively. If $m = n$, then L is a **full-rank** lattice. In most cases, we work with full-rank lattices.

It follows from the definition that a lattice is closed under addition. Hence, we can say that an n-dimensional lattice is a discrete additive subgroup of $\mathbb{R}^n$. It is isomorphic to the additive group of $\mathbb{Z}^n$. That is,

$$(L, +) \cong (\mathbb{Z}^n, +) \subsetneq (\mathbb{R}^n, +).$$

It is often convenient to work with lattices whose coordinates are integers. These are called **integer lattices** or **integral lattices**. For example, the set of even integers forms an integer lattice, but not the set of odd integers because it is not closed under addition.

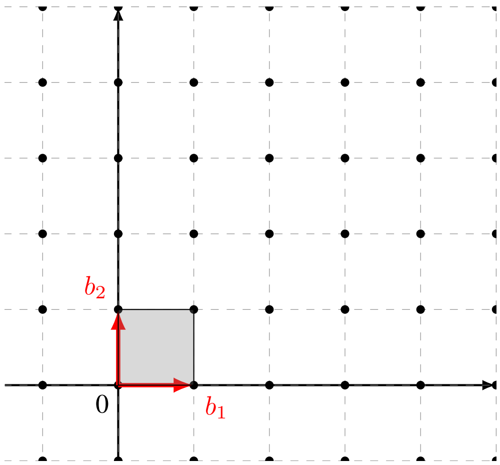


Figure 1: A lattice L with a basis $B = \{b_1, b_2\}$ and its fundamental domain F.

Basis

A **basis** of a lattice L is a set of linearly independent vectors $B = \{b_1, \ldots, b_n\}$ that spans the lattice, that is,

$$L(B) = \{z_1 b_1 + \cdots + z_n b_n \mid z_i \in \mathbb{Z}\}.$$

For example, the vectors $\{b_1, b_2\}$ form a basis of the lattice in Figure 1.

[1] This section is part of the work *A Tutorial Introduction to Lattice-based Cryptography and Homomorphic Encryption* by the authors Yang Li, Kee Siong Ng, Michael Purcell from the School of Computing, Australian National University @2022.

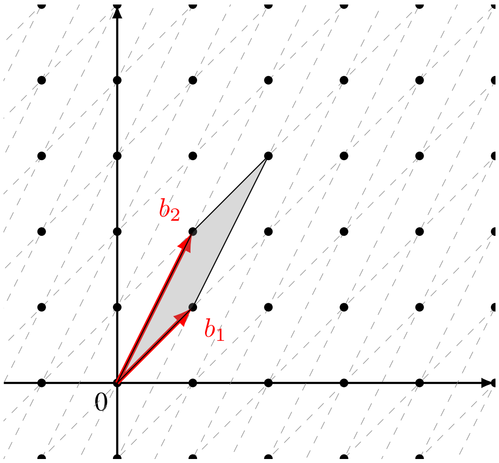


Figure 2: The same lattice L with a different basis $B' = \{b'_1, b'_2\}$ and its fundamental domain F', where $B' = AB$ for a unimodular change of basis matrix $A = \left(\begin{smallmatrix} 1 & 1 \\ 1 & 2 \end{smallmatrix}\right)$.

In what follows, we will frequently appeal to properties of a class of matrices known as *unimodular matrices*. Unimodular matrices can be used to translate between different lattice bases. They are also used, sometimes implicitly, when performing important lattice operations such as lattice basis reduction.

Unimodular matrix

Definition 0.1.2. *A matrix $A \in \mathbb{Z}^{n\times n}$ is **unimodular** if it has a multiplicative inverse in $\mathbb{Z}^{n\times n}$. That is, $A \in \mathbb{Z}^{n\times n}$ is unimodular if and only if $A^{-1} \in \mathbb{Z}^{n\times n}$. Equivalently, a matrix $A \in \mathbb{Z}^{n\times n}$ is unimodular if and only if* $|\det(A)| = 1$.

Similar to a vector space, a lattice does not need to have a unique basis. The following proposition establishes the fact that one basis can be transformed to another via multiplication by the matrix A provided that A is a unimodular matrix.

Proposition 0.1.3. *If B and B' be two basis matrices, then $L(B) = L(B')$ if and only if $B' = AB$ for some unimodular matrix A.*

Proof. Suppose that $B' = AB$ for some unimodular matrix A. Then, by definition both A and A^{-1} have integer entries. Therefore we have $L(B') \subset L(A^{-1}B') = L(B)$ and $L(B) \subset L(AB) = L(B')$.

Now suppose that $L(B) = L(B')$. Then there exist integer square matrices $A, A' \in \mathbb{Z}^{n\times n}$ such that $B' = AB$ and $B = A'B'$. Therefore we have $B = A'AB$ or equivalently $(I - A'A)B = 0$. Because B is non-singular, we have $A' = A^{-1}$ and A is unimodular. □

For example, the vectors $\{b'_1, b'_2\}$ in Figure 2 form a different basis for the lattice in Figure 1, with the relation $B' = AB$ where the change of basis matrix $A = \left(\begin{smallmatrix} 1 & 1 \\ 1 & 2 \end{smallmatrix}\right)$ is unimodular.

An important concept of a lattice is the fundamental domain. It is closely related to the sparsity of a lattice as can be seen from the following definition.

Fundamental domain

Definition 0.1.4. *Let L be an n-dimensional lattice with a basis $\{v_1, \ldots, v_n\}$. The **fundamental domain** or (**fundamental parallelepiped**) of L is a region defined as*

$$F(v_1, \ldots, v_n) = \{t_1 v_1 + \cdots + t_n v_n \mid t_i \in [0, 1)\}.$$

The lattice L and the given basis in Figure 1 has the fundamental domain coloured in grey. It is the convex region that is surrounded by the given basis vectors and the nearby lattice points.

Determinant

Definition 0.1.5. *Let L be an n-dimensional lattice with a fundamental domain F. Then the n-dimensional volume of F is called the **determinant** of L, denoted by* $\det(L)$.

Given a basis $\{v_1, \ldots, v_n\}$ of an n-dimensional lattice L, we can write each basis vector $v_i = (v_{i1}, \ldots, v_{in})$ as a vector of its coordinates. Then we have a **basis matrix**

$$B = \begin{pmatrix} v_{11} & \cdots & v_{1n} \\ \vdots & \ddots & \vdots \\ v_{n1} & \cdots & v_{nn} \end{pmatrix}. \tag{1}$$

In cryptography, we are interested in full-rank lattices, whose determinant can be easily calculated using a basis matrix as stated in the next proposition.

Proposition 0.1.6. *If L is an n-dimensional full-rank lattice with a basis $\{v_1, \ldots, v_n\}$ and an associated fundamental domain $F = F(v_1, \ldots, v_n)$, then the volume of F (or determinant of L) is equal to the absolute value of the determinant of the basis matrix B, that is,*

$$\det(L) = Vol(F) = |\det B|.$$

Although the fundamental domain may have a different shape under another choice of a basis, it can be proved that area (or volume) stays unchanged. This gives rise to the determinant of a lattice which is an invariant quantity under the choice of a fundamental domain.

Invariant determinant

Corollary 0.1.7. *The determinant of a lattice is an invariant quantity under the choice of a basis for L.*

Proof. Let L be a lattice and let B and B' be the basis matrices for two different bases for L. There exists a unimodular matrix A such that $B' = AB$. Consequently, we have

$$|\det(B')| = |\det(AB)| = |\det(A)| \cdot |\det(B)| = |\det(B)|.$$

So, we have $|\det(L)| = |\det(B')| = |\det(B)|$. □

Example 0.1.8. *Let L be a 3-dimensional lattice with a basis*

$$\{v_1 = (2, 1, 3), v_2 = (1, 2, 0), v_3(2, -3, -5)\}.$$

Then a basis matrix is

$$B = \begin{pmatrix} 2 & 1 & 3 \\ 1 & 2 & 0 \\ 2 & -3 & -5 \end{pmatrix}. \tag{2}$$

The determinant of the lattice is $\det(L) = |\det(B)| = 36$.

Geometrically, this also makes sense. By definition, each fundamental domain contains exactly one lattice vector (in Figure 1 and 2 the origin). Consider fundamental domains that are centered on lattice points rather than having lattice points at one corner. That is, consider

$$\tilde{F}(v_1, v_2, \ldots, v_n) = \{t_1 v_1 + t_2 v_2 + \ldots + t_n v_n \mid t_i \in [-1/2, 1/2)\}.$$

Take a large ball centered at the origin and notice that, because each fundamental domain contains exactly one lattice point, the volume of the ball is approximately equal to the number of lattice points in the ball multiplied by the volume of the fundamental domain. More precisely, we have

$$\lim_{r \to \infty} \frac{\text{Vol}\,(B_r(\mathbf{0}))}{|B_r(\mathbf{0}) \cap L|} = \text{Vol}\left(\tilde{F}(v_1, v_2, \ldots, v_n)\right) = \det(L).$$

By definition, choosing a different basis doesn't change the lattice. So, the volume of the fundamental domain, and therefore the determinant of the lattice, is a property of the lattice and does not depend on the basis used to represent that lattice.

Two remarks. First, a lattice L can be partitioned into disjoint fundamental domains, the union of which covers the entire L. Second, since the choice of a fundamental domain is arbitrary and it covers real vectors that are not in L, each real vector can be uniquely identified by a lattice vector and a real vector in a fundamental domain. These are captured in the following proposition. For the proof, see Proposition 6.18 in Hoffstein et al. (2008).

Proposition 0.1.9. *Let L be an n-dimensional lattice in $\mathbb{R}^n$ with a fundamental domain F. Then every vector $w \in \mathbb{R}^n$ can be written as*

$$w = v + t \tag{3}$$

for a unique lattice vector $v \in L$ and a unique real vector $t \in F$.

Equivalently, the union of the translated fundamental domains cover the span of the lattice basis vectors, i.e.,

$$\textit{span}(L) = \{F + v \mid v \in L\}.$$

Another useful interpretation of Equation 3 is that for any vector $w \in \mathbb{R}^n$, there is a unique real vector $t \in F$ in the fundamental domain such that $w - t \in L(B)$ is a lattice vector. In other words, given an arbitrary vector $w \in \mathbb{R}^n$ in the span, we can efficiently reduce it to a vector $t \in F$ in the fundamental domain by taking w modulo the basis (or modulo the fundamental domain as used by some authors). More precisely, for a basis $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ of $L \in \mathbb{R}^n$, it is obvious that the basis is also a basis of the span $\mathbb{R}^n$, so we have $\mathbf{w} = \alpha_1\mathbf{v}_1 + \cdots + \alpha_n\mathbf{v}_n$ for coefficients $\alpha_1, \dots, \alpha_n \in \mathbb{R}$. The coefficients can also be written as $\alpha_i = a_i + t_i$ for $a_i \in \mathbb{Z}$ and $t_i \in (0, 1)$. This implies the real vector can be re-written as $\mathbf{w} = (a_1\mathbf{v}_1 + \cdots + a_n\mathbf{v}_n) + (t_1\mathbf{v}_1 + \cdots + t_n\mathbf{v}_n) = \mathbf{v} + \mathbf{t}$, where in the first pair of parentheses is a lattice vector $\mathbf{v}$ and in the second pair is a real vector $\mathbf{t}$ within the fundamental domain. From this, we can compute $\mathbf{t} = \mathbf{w} - \mathbf{v}$. This also gives an alternative formula for computing the modulo basis operation by

Modulo basis

$$\mathbf{w} \bmod \mathbf{B} = \mathbf{w} - \mathbf{B} \cdot \lfloor \mathbf{B}^{-1} \cdot \mathbf{w} \rfloor. \tag{4}$$

For example, given a 2-dimensional lattice $L \in R^2$ with a basis $\mathbf{B} = \left(\begin{smallmatrix} 3 & 0 \\ 0 & 2 \end{smallmatrix}\right)$ and a real vector $\mathbf{w} = (2, 3)$. By reducing $\mathbf{w}$ modulo the fundamental domain we get $\mathbf{w} \bmod \mathbf{B} = (2, 1)$.

Similar to a real vector, the length a lattice vector can also be measured by a norm function $|| \cdot ||$. However, unlike in a vector space where there is no shortest non-zero vector, it is possible to define shortest non-zero vector in a lattice because of the discreteness, although this shortest vector may not be unique.

Shortest vector

Definition 0.1.10. *Given a lattice L, **the length of a shortest non-zero vector** in L which is also a **minimum distance** between two lattice vectors is defined as*

$$\begin{aligned} \lambda_1(L) &= \min\{||\mathbf{v}|| \mid \mathbf{v} \in L \setminus \{\mathbf{0}\}\} \\ &= \min\{||\mathbf{x} - \mathbf{y}|| \mid \mathbf{x}, \mathbf{y} \in L, \mathbf{x} \neq \mathbf{y}\}. \end{aligned}$$

The shortest vector problem (formally defined in Section 0.3) is to find the shortest non-zero vector in a given lattice. For a lattice L, notice that $\lambda_1(L)$ is the solution to the shortest vector problem for that lattice.

The shortest vector problem can be generalized to the problem of finding the i^{th} successive minima. The ith successive minima is the minimum length r such that the lattice contains i linearly independent vectors of length at most r. This can also be defined in relation to the dimension of the space spanned by the intersection between L and a zero-centered closed ball $\bar{B}(0, r)$ with radius r.

Successive minima

Definition 0.1.11. *Given a lattice L, the i^{th} **successive minima** of L is defined as*

$$\lambda_i(L) = \min\{r \mid \dim(\textit{span}(L \cap \bar{B}(0, r))) \geq i\},$$

where $\bar{B}(0, r) = \{x \in \mathbb{R}^n \mid ||x|| \leq r\}$ is the closed ball of radius r around 0.

For example, if the lattice $L = \mathbb{Z}^n$, then the 1st to the n^{th} successive minima $\lambda_1 = \cdots = \lambda_n = 1$ are equal to 1. The length of a shortest vector is a special case of the successive minima when $i = 1$. We will see the successive minima again when introducing shortest independent vector problem as a generalization of the shortest independent problem in 0.3.

Notice that a set of vectors that achieves the successive minima of a lattice is not necessarily a basis for that lattice. Consider the following example which is derived from the work of Korkine and Zolotareff (1873) and was presented its current form in Nguyen and Vallée (2010). Let

$$\mathbf{B} = \begin{pmatrix} 2 & 0 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 & 1 \\ 0 & 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Notice that $2\mathbf{e}_5 \in L(\mathbf{B})$ and that $\|v\| \geq 2$ for all $\mathbf{v} \in L(\mathbf{B}) \setminus \{\mathbf{0}\}$. So, $\lambda_i(L(\mathbf{B})) = 2$ for $1 \leq i \leq 5$. If we let

$$\tilde{\mathbf{B}} = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{pmatrix}.$$

then we have $L(\tilde{\mathbf{B}}) \subset L(\mathbf{B})$ and $\det(\tilde{\mathbf{B}}) = 32$. On the other hand, we see that $\det(\mathbf{B}) = 16$. Therefore, $\tilde{\mathbf{B}}$ cannot be a basis for $L(\mathbf{B})$. In fact, it can be shown that no basis of $L(\mathbf{B})$ realizes all of the successive minima of $L(\mathbf{B})$.

0.2 Dual lattice

In this subsection, we introduce dual lattices. This is a useful concept that will be used at several different places, such as defining smoothing parameter for discrete Gaussian distribution and in the hardness proof of the ring learning with error problem. It is important to develop a geometric intuition of the relationship between a lattice and its dual.

The dual (sometimes also called reciprocal) of a lattice is the set of vectors in the span of the lattice (e.g., the span is $\mathbb{R}^n$ if the lattice is $\mathbb{Z}^n$) whose inner product with the lattice vectors are integers.

Dual lattice

Definition 0.2.1. *Given a full-rank lattice L, its **dual lattice** is defined as*

$$L^* = \{\mathbf{y} \in span(L) \mid \forall \mathbf{x} \in L, \mathbf{x} \cdot \mathbf{y} \in \mathbb{Z}\}.$$

For example, the dual lattice of $\mathbb{Z}^n$ is $\mathbb{Z}^n$ and the dual lattice of $2\mathbb{Z}^n$ is $\frac{1}{2}\mathbb{Z}^n$ as shown in Figure 4. An important observation is that the more vectors a lattice has, the less vectors its dual has and vice versa, because there are more (or less) constraints. Most importantly, it can be verified that the dual of a lattice is also a lattice.

Proposition 0.2.2. *If L is a lattice then L^* is a lattice.*

Proof. It suffices to show that L^* is closed under subtraction. That is, to show that if $x, y \in L^*$ then $x - y \in L^*$. This follows from the linearity of the inner product. More explicitly, for every $\mathbf{z} \in L$ we have $(\mathbf{x} - \mathbf{y}) \cdot \mathbf{z} = \mathbf{x} \cdot \mathbf{z} - \mathbf{y} \cdot \mathbf{z}$. Because $\mathbf{x} \cdot \mathbf{z} \in \mathbb{Z}$ and $\mathbf{y} \cdot \mathbf{z} \in \mathbb{Z}$, we have $(\mathbf{x} - \mathbf{y}) \cdot \mathbf{z} \in \mathbb{Z}$. The result then follows from the definition of L^*. □

Given a lattice L, it is natural to ask if we can find a basis for L^*. This leads us to define the dual basis of a lattice.

Dual basis

Definition 0.2.3. *For a lattice L and a basis $B = (b_1, \ldots, b_n) \in \mathbb{R}^{m \times n}$, the **dual basis** $D = (d_1, \ldots, d_n) \in \mathbb{R}^{m \times n}$ is defined as the unique basis that satisfies*

- *$span(B) = span(D)$ and*
- *$B^T D = I$.*

The first condition says both bases span the same vector space. The second condition implies that $b_i \cdot d_j = \delta_{ij} = 1$ if $i = j$ and 0 otherwise. Abusing notation, we use B to denote both the basis of a lattice and the basis matrix. If L is a full-rank lattice (i.e., $m = n$), then the basis matrix B is invertible, so the dual basis matrix can be expressed as $D = (B^T)^{-1} = (B^{-1})^T$.

Proposition 0.2.4. *If L is a lattice with basis B, then the dual basis is a basis for L^*.*

Proof. This follows immediately from the definition of the dual lattice and the linearity of the inner product. □

Having established that the dual of a lattice is itself a lattice, we can ask what we get if repeat the process and compute the dual of a dual lattice.

Proposition 0.2.5. *For any lattice L, we have $(L^*)^* = L$.*

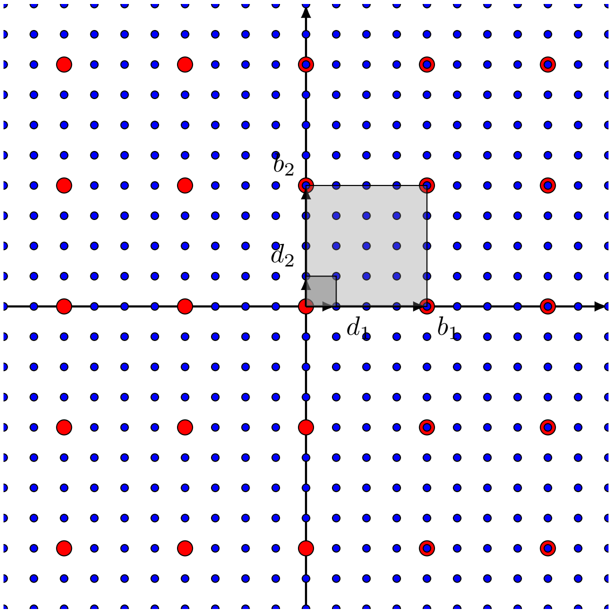


Figure 3: A lattice $L = 2\mathbb{Z}^2$ (black points) and its dual $L^* = \frac{1}{2}\mathbb{Z}^2$ (blue points). The basis of L is $B = \{b_1 = (2,0), b_2 = (0,2)\}$ and the dual basis of L^* is $D = \{d_1 = (\frac{1}{2}, 0), d_2 = (0, \frac{1}{2})\}$.

Proof. If B is a basis for a full-rank lattice L, then a dual basis is $D = (B^T)^{-1}$. Then the dual basis of D is $(D^T)^{-1}$ that is equal to B. The same argument works for rank-deficient lattices, but with slight variation because their bases are non-square matrices. □

Proposition 0.2.6. *For any lattice L, we have* $\det(L^*) = \frac{1}{\det(L)}$.

Proof. Again, we give a proof for full-rank lattices. If L is full-rank, then

$$\det(L^*) = |\det(D)| = |\det((B^T)^{-1})| = \frac{1}{|\det(B^T)|} = \frac{1}{|\det(B)|} = \frac{1}{\det(L)}.$$

□

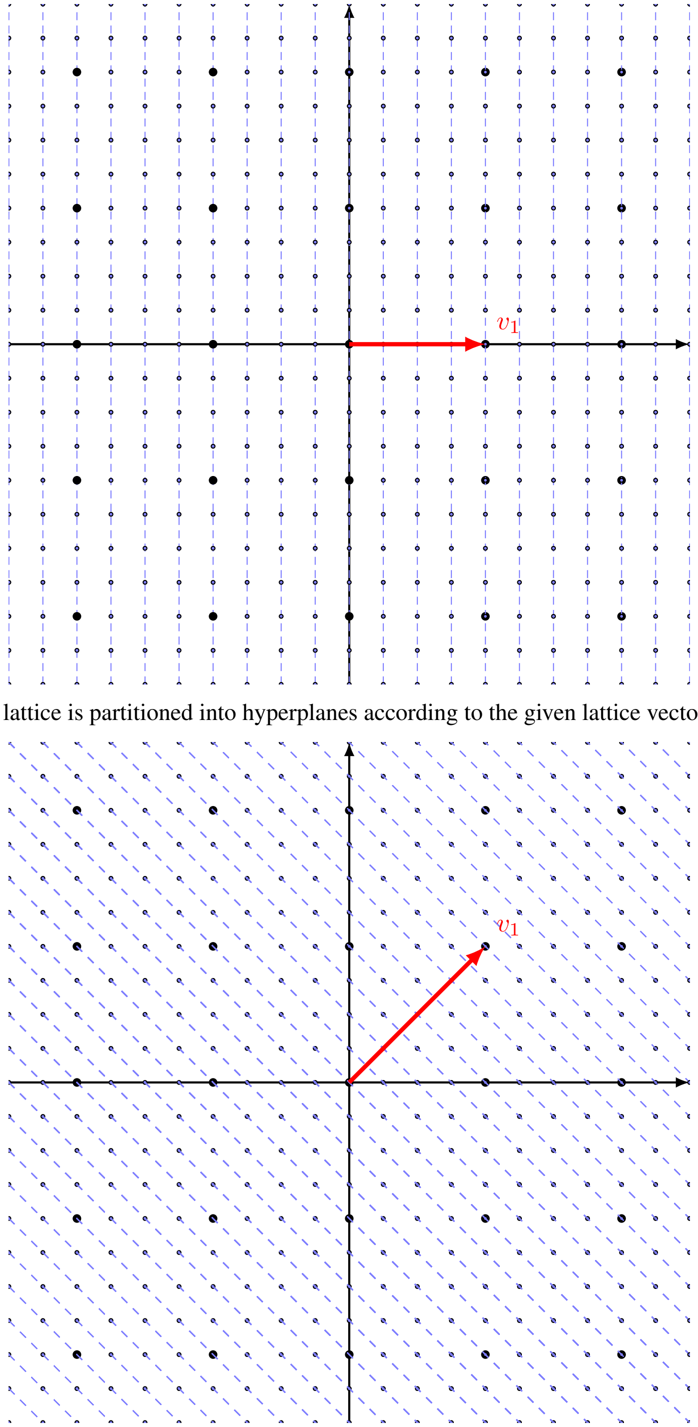
Although a lattice and its dual are both lattices, they are fundamentally different objects. The dual of a lattice can be thought as functions that are applied to the lattice such that the inner products of the lattice vectors and each dual vector are integers.

Hyperplanes

Here is a geometric interpretation of a lattice and its dual. For each lattice vector $\mathbf{v}$, its inner products with the dual vectors produce integers of different values. So $\mathbf{v}$ partitions the dual lattice into parallel non-overlapping hyperplanes that are perpendicular to $\mathbf{v}$ according to its inner product values with the dual vectors. Elements in the same hyperplane have the same inner product with the lattice vector $\mathbf{v}$, so they form an equivalence class. Alternatively, we can say $\mathbf{v}$ partitions the dual lattice into a set of equivalence classes. Figure. 4 gives two examples of how a lattice vector $\mathbf{v} \in L = 2\mathbb{Z}^2$ partitions the dual lattice $L^* = \frac{1}{2}\mathbb{Z}^2$. In addition, the distance between two neighbouring hyperplanes is the inverse of the vector length (i.e., $1/||\mathbf{v}||$).

Example 0.2.7. *When $L = 2\mathbb{Z}$ and $L^* = \frac{1}{2}\mathbb{Z}$, the vector $\mathbf{v} = \frac{1}{2}$ partitions L to $|2\mathbb{Z}|$ hyperplanes, each contains exactly one integer from L and the neighbouring hyperplanes are distance 2 apart.*

When $L = 2\mathbb{Z}^2$ and $L^* = \frac{1}{2}\mathbb{Z}^2$, the vector $\mathbf{v} = (2,0)$ partitions the dual lattice into hyperplanes as shown in Figure 4a, where the hyperplanes are the vertical lines that are perpendicular to the lattice vector $\mathbf{v}$. The distance between the neighbouring hyperplanes is $\frac{1}{||\mathbf{v}||} = \frac{1}{2}$. So the dual is denser than L. If $\mathbf{v} = (2,2)$, the dual is partitioned into hyperplanes as shown in Figure 4b. The distance between the neighbouring hyperplanes is $\frac{1}{||\mathbf{v}||} = \frac{1}{2\sqrt{2}}$.



(a) The dual lattice is partitioned into hyperplanes according to the given lattice vector $v = (2, 0)$.

(b) The dual lattice is partitioned into hyperplanes according to the given lattice vector $v = (2, 2)$.

Figure 4: For a given lattice vector $v \in L = 2\mathbb{Z}^2$, the dual lattice $L^* = \frac{1}{2}\mathbb{Z}^2$ can be partitioned into parallel non-overlapping hyperplanes (vertical lines) that are perpendicular to v. Elements in the same hyperplane have the same dot product with v, so they form an equivalence class.

0.3 Some lattice problems

Having briefly introduced lattices and some related concepts, we are ready to define some computational lattice problems in this subsection. The most well known two are the shortest vector problem and closest vector problem. These two are search problems because the aims are to find a shortest or closest lattice vector. Few cryptosystems, however, are based on these two problems directly. Instead, most cryptosystems are based on their decision versions or relaxed approximation variants. Below, we state the two well known lattice problems and some variants.

The Shortest Vector Problem (SVP)
Given a lattice basis B, find a shortest non-zero vector in the lattice $L(B)$, i.e., find a non-zero vector $\mathbf{v} \in L(B)$ such that $||\mathbf{v}|| = \lambda_1(L(B))$.

SVP is hard to solve in high-dimensional lattices. An important variant of SVP is finding a set of short linearly independent lattice vectors as stated below.

The Shortest Independent Vectors Problem (SIVP)
Given a lattice basis B of an n-dimensional lattice $L(B)$, find n linearly independent vectors $\mathbf{v_1}, \ldots, \mathbf{v_n} \in L(B)$ such that $\max_{i\in[1,n]} ||\mathbf{v_i}|| = \lambda_n(L(B))$.

The Closest Vector Problem (CVP)
Given a lattice basis B and a target vector $\mathbf{t}$ that is not in the lattice $L(B)$, find a vector in $L(B)$ that is closest to $\mathbf{t}$, i.e., find a vector $\mathbf{v} \in L(B)$ such that for all $w \in L(B)$ it satisfies $||\mathbf{v} - \mathbf{t}|| \leq ||\mathbf{w} - \mathbf{t}||$.

A special case of CVP is the bounded distance decoding problem, which is used in the learning with error problem's hardness proof (Regev, 2009). The name reflects that the problem is to "decode" a given $\mathbb{R}^n$ vector. The extra condition makes it a special case of CVP is that the given non-lattice vector is within a bounded distance to the lattice.

The α-Bounded Distance Decoding Problem (BDD$_\alpha$)
Given a lattice basis B of an n-dimensional lattice L and a target vector $\mathbf{t} \in \mathbb{R}^n$ satisfies $dist(\mathbf{t}, B) \leq \alpha\lambda_1(L)$, find a lattice vector $\mathbf{v} \in L$ that is closest to $\mathbf{t}$, i.e., for all $\mathbf{w} \in L$ it satisfies $||\mathbf{v} - \mathbf{t}|| \leq ||\mathbf{w} - \mathbf{t}||$.

An alternative way of defining BDD is to find the lattice vector $\mathbf{x} \in L$ given the instance $\mathbf{y} = \mathbf{x} + \mathbf{e} \in \mathbb{R}^n$, where $\mathbf{e}$ is often interpreted as a noise with norm $||e|| \leq \alpha\lambda_1(L)$.

As discussed in **??**, knowing c-gap problems are hard implies the corresponding c-approximate problems are also hard. But c-approximations are often used to prove some problems are hard to solve (e.g., SIS) because it is relatively easier to build reductions from them. Below we state the gap/approximate variants of the standard lattice problems. Let $\gamma(n) : \mathbb{N} \to \mathbb{N}$ be a gap function in the input size such that $\gamma(n) \geq 1$, for example $\gamma(n)$ is a polynomial of n.

The γ-GAP Shortest Vector Problem (GAPSVP$_\gamma$)
INSTANCE: For a function $\gamma(n) \geq 1$, given a real number $d > 0$ and a lattice basis B, the instance (B, d) is

- either a YES instance if $\lambda_1(L(B)) \leq d$
- or a NO instance if $\lambda_1(L(B)) \geq \gamma(n)d$.

QUESTION: Is (B, d) a YES or NO instance?

The (ζ, γ)-GAP Shortest Vector Problem (GAPSVP$_{\zeta,\gamma}$)
INSTANCE: For functions $\zeta(n) \geq \gamma(n) \geq 1$, given a real number $d > 0$ and a lattice basis B of an n-dimensional lattice $L(B)$ such that

- $\lambda_1(L(B)) \leq \zeta(n)$,
- $\min_{i\in[1,n]} ||\tilde{b}_i|| \geq 1$,
- $1 \leq d \leq \zeta(n)/\gamma(n)$,

the instance (B, d) is

- either a YES instance if $\lambda_1(L(B)) \leq d$
- or a NO instance if $\lambda_1(L(B)) \geq \gamma(n)d$.

QUESTION: Is (B, d) a YES or NO instance?

The γ-Shortest Independent Vectors Problem (SIVP$_\gamma$)
Given a lattice basis B of an n-dimensional lattice $L(B)$, find n linearly independent vectors $\mathbf{v_1}, \ldots, \mathbf{v_n} \in L(B)$ such that $\max_{i\in[1,n]} ||\mathbf{v_i}|| \leq \gamma(n)\lambda_n(L(B))$.

0.4 Ajtai's worst-case to average-case reduction

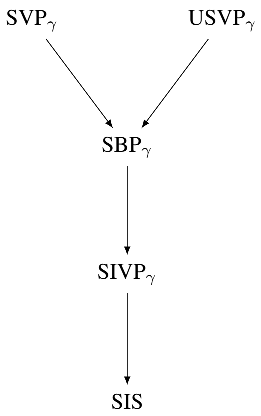


Figure 5: Reductions to the SIS problem from hard lattice problems (SVP$_\gamma$, USVP$_\gamma$ and SBP$_\gamma$). The intermediate lattice problem in the reductions is the γ-approximation of the shortest independent vector problem (SIVP$_\gamma$).

To finish off this section, we present a high level overview of Ajtai (1996)'s worst-case to average-case reduction. As briefly explained in **??**, such a reduction allows one to build cryptosystems based on an average-case hardness problem, so that users can rest assured that their random encryption instances are guaranteed to be secure with high confidence.

Ajtai's proof is based on three well-studied lattice problems, SVP$_\gamma$, USVP$_\gamma$ and SBP$_\gamma$. The second problem is a variant of SVP that finds the unique shortest non-zero vector in the lattice $L(B)$, i.e., find the non-zero vector $\mathbf{v} \in L(B)$ such that $||\mathbf{v}|| = \lambda_1(L(B))$ and if $\mathbf{w} \in L(B)$ such that $||\mathbf{w}|| \leq n^c||\mathbf{v}||$ then $\mathbf{w}$ is parallel to $\mathbf{v}$. The third problem is to find a shortest basis $\{\mathbf{b}_1, \ldots, \mathbf{b}_n\}$ of a given lattice, where the basis length is defined as $\max_{i=1}^n ||\mathbf{b}_i||$. All three problems are used in their gap (or approximation) versions.

SIS The average-case hard problem constructed by Ajtai (1996) is known as the short integer solution (SIS) problem. Let $\mathbf{a_i} \in \mathbb{Z}_q^n$ be a length n vector with entries taken uniformly from $\mathbb{Z}_q$. Let $A = [\mathbf{a_1} \mid \cdots \mid \mathbf{a_m}]$ be an $n \times m$ matrix whose columns are m linearly independent $\mathbf{a_i}$s. The SIS problem is to find a non-zero vector $\mathbf{x} \in \mathbb{Z}^m$ such that

- $||\mathbf{x}|| \leq \beta$ and

- $A\mathbf{x} = \mathbf{0} \in \mathbb{Z}_q^n$, i.e., $\mathbf{x}_1\mathbf{a_1} + \cdots + \mathbf{x}_m\mathbf{a_i} = \mathbf{0} \bmod q$.

Notice that the norm bound exists to ensure the problem is not easily solvable by for example Gaussian elimination. It must satisfy $\beta < q$ to avoid the trivial solution $\mathbf{x} = (q, 0, \ldots, 0)$. Moreover, β and m must be large enough to allow a solution to exist. See Section 4 of (Peikert, 2016) for more detailed insights.

The structure of the reduction is shown in Figure 5. The essential part of the proof is a polynomial-time reduction from SBP$_\gamma$ to SIS. The other two lattice problems can be reduced to SBP$_\gamma$, we skip the reduction details and refer the reader to the appendix of (Ajtai, 1996).

To further simplify the reduction steps, SBP$_\gamma$ is related to SIVP$_\gamma$ because given a set of linearly independent lattice vectors $\mathbf{r_1}, \ldots, \mathbf{r_n} \in L$, a basis $\{\mathbf{s_1}, \ldots, \mathbf{s_n}\}$ of L can be constructed in polynomial time such that $\max_{i=1}^n ||\mathbf{s_i}|| \leq n \max_{i=1}^n ||\mathbf{r_i}||$. So the task becomes finding linearly independent lattice vectors $\mathbf{r_1}, \ldots, \mathbf{r_n} \in L$ such that $\max_{i=1}^n ||\mathbf{r_i}|| \leq n^{c_3 - 1} bl(L)$, where $bl(L)$ is the length of the shortest basis of L.

SIVP$_\gamma$ to SIS

The reduction starts by assuming there is a probabilistic polynomial time (PPT) algorithm $\mathcal{A}$ that solves SIS with a non-negligible probability.[2]

The next step is to turn a (hard) instance of SIVP$_\gamma$ to a random SIS instance and show that if such an $\mathcal{A}$ exists, it gives rise to a PPT algorithm $\mathcal{B}$ that solves SIVP$_\gamma$ by finding a set of linearly independent vectors $\{\mathbf{a}_1, \ldots, \mathbf{a}_n\}$ such that the maximum length is bounded $\max_i ||a_i|| = M \leq n^{c_3-1} bl(L)$. From this upper bound and SIVP$_\gamma$'s relation to SBP$_\gamma$, we can get a basis whose length is within the desired bound $n^{c_3} bl(L)$, so SBP$_\gamma$ is solved as well as SVP$_\gamma$ and USVP$_\gamma$.

The key to produce short linearly independent vectors $\mathbf{a_1}, \ldots, \mathbf{a_n}$ to satisfy $M < n^{c_3-1} bl(L)$ is to iteratively produce from the longer vectors shorter ones of maximum length $\frac{M}{2}$. Repeating this steps at most $\log_2 M$ steps we get vectors of the desired length.

Each run of the recursive step is as follows:

1. Starting from the lattice vectors $\mathbf{a_1}, \ldots, \mathbf{a_n}$, construct other lattice vectors $\mathbf{f_1}, \ldots, \mathbf{f_n}$ such that they are nearly pairwise orthogonal and have similar length, but constraint the maximum length $\max_{i=1}^n ||\mathbf{f_i}|| \leq n^3 M$. The reason is to form a parallelepiped $W = P(\mathbf{f_1}, \ldots, \mathbf{f_n})$ that is almost a hypercube, as shown in a 2-dimensional lattice in Figure 6. This step was proved in Lemma 3 Ajtai (1996).
2. We then evenly cut W into q^n small non-overlapping parallelepipeds which have the form $w_j = (\sum_{i=1}^n \frac{t_i^j}{q} \mathbf{f_i}) + \frac{1}{q} W$, where $t_i^j \in [0, q)$ is an integer. Now sample m random lattice vectors from L, then reduce them modulo W to ensure they are within the bigger parallelepiped. Denote these reduced vectors by $\xi_\mathbf{1}, \ldots, \xi_\mathbf{m}$. If $\xi_\mathbf{k}$ is in a smaller parallelepiped $w_j = (\sum_{i=1}^n \frac{t_i^j}{q} \mathbf{f_i}) + \frac{1}{q} W$, then take $(t_1^j, \ldots, t_n^j)$ and put it as a column of a matrix A. The claim is that each of the w_j's is selected with almost equal chance, so we have a random $n \times m$ matrix A. The key intuition is that for a short basis of L, if W intersects with a translation of the fundamental domain formed by the short basis, then W will contain a large proportion of the translated fundamental domain. This property remains true for an arbitrary translation and scaling of W using $\mathbf{u} + \frac{1}{q} W$ for a vector $\mathbf{u} \in \mathbb{R}^n$. With this property, if W is cut into small non-overlapping regions evenly, then random samples of lattice vectors within W will induce a near uniform distribution over the w_j's. This helps ensure that the matrix A generated randomly as above is a random instance of SIS. This step was proved in Lemma 8 Ajtai (1996).
3. Now give the matrix A to the PPT algorithm $\mathcal{A}$ to output an SIS solution $(h_1, \ldots, h_m) \in \mathbb{Z}^m$. It remains to prove that the vector $\mathbf{u} = \sum_{i=1}^n h_i \xi_\mathbf{i}$ is only half of size of the starting vectors, i.e., $||\mathbf{u}|| \leq \frac{M}{2}$ and they are non-zero. This step was proved in Lemma 13 Ajtai (1996).

A couple of remarks about the reduction. First, the approximation (or gap) factors in the lattice problems are relatively large, typically larger than n^8 as analysed by Cai and Nerurkar (1997) and

[2] Ajtai related SIS with finding a short vector in a *q-ary lattice* $L_q^\perp(A) = \{\mathbf{x} \mid A\mathbf{x} = \mathbf{0} \bmod q\}$. His reduction starts with assuming $\mathcal{A}$ is a PPT algorithm to find a short lattice vector in a given $L_q^\perp(A)$. For the purpose of sketching the main steps of the proof, it is not necessary to relate SIS with the q-ary lattice problem.

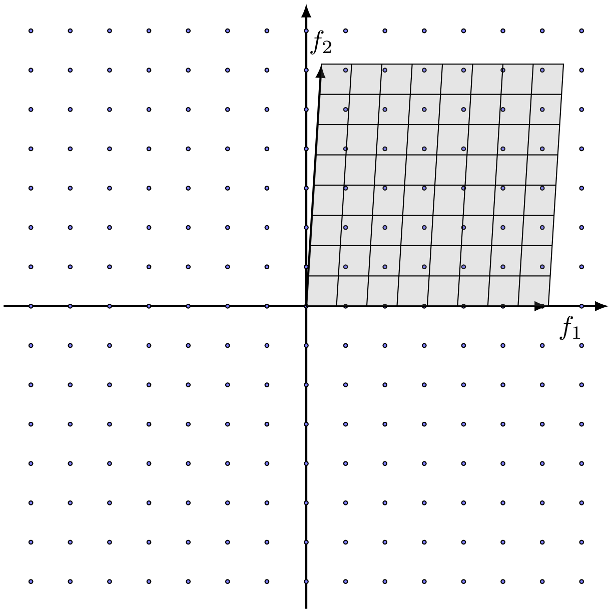


Figure 6: In a lattice $L = \mathbb{Z}^2$, the near cubic parallelepiped W formed by the large independent vectors $\{f_1, f_2\}$. It is divided into q^2 smaller pieces, each of which is hit with equal probability by random lattice vectors reduced within W.

suggested in the Introduction section in (Micciancio and Regev, 2007). This could raise minor concerns about the hardness of the lattice problems, because the larger the gap factors the easier the problems are. In a following section, we will introduce the discrete Gaussian strategy to reduce these factors down to $\tilde{O}(n)$ for the SIS problem. The proof strategy will follow Ajtai's technique of constructing, from a set of linearly independent vectors of a lattice, a random set of lattice vectors that are well spread out and as short as possible. Second, the public key size required by an SIS-based cryptosystem is $\tilde{O}(n^4)$ that is quite inefficient for practical purposes. This will be dramatically improved by developing different average-case problems as we will see in the learning with error and ring learning with error problems.

0.5 An application of SIS: Collision resistant hash functions

SIS has been used as the foundation of one-way functions and hash functions, see e.g. (Lyubashevsky et al., 2010).

A hash function maps inputs of arbitrary length and compresses them into short fixed-length outputs known as *digests*.

Hash function

Definition 0.5.1. *A **(keyed) hash function** with output length l is a pair of probabilistic polynomial-time algorithms (Gen, H) satisfying the following:*

- *The algorithm $Gen(1^n) \to s$ generates a key s from the security parameter 1^n.*
- *For a string $x \in \{0,1\}^*$ of arbitrary length, the algorithm H outputs a string $H^s(x) \in \{0,1\}^{l(n)}$.*

The general interest in hash functions is the case when the outputs are shorter than the inputs for both computational and storage efficiency. In such a case, a hash function's domain is larger than its range, which implies the possibility of having two distinct inputs being mapped to the same output. We often say the two distinct inputs *collide* and the scenario is called a *collision*.

For a hash function $\Pi = (\text{Gen}, H)$, an adversary $\mathcal{A}$ and the security parameter n, we can define the **collision-finding experiment Hash-coll**$_{\mathcal{A},\Pi}(n)$ as:

Hash-coll$_{\mathcal{A},\Pi}(n)$

1. Run the algorithm $\text{Gen}(1^n) \to s$.
2. The adversary $\mathcal{A}$ is given the key s.

3. The adversary produces two strings x, and x'.
4. **Hash-coll**$_{\mathcal{A},\Pi}(n) = 1$ if $x \neq x'$ and $H^s(x) = H^s(x')$ and 0 otherwise.

A cryptographic hash function requires the chance of finding a collision is negligible, which is defined more formally as follows.

Collision resistant

Definition 0.5.2. *A hash function $\Pi = (Gen, H)$ is* ***collision resistant*** *if for any probabilistic polynomial time adversary $\mathcal{A}$, it satisfies*

$$Pr[\textit{Hash-coll}_{\mathcal{A},\Pi}(n) = 1] \leq \textit{negl}(n).$$

From Ajtai's SIS problem and the worst-case-to-average-case reduction, one can easily build a collision resistant hash function where the key is the matrix $A \in \mathbb{Z}_q^{n\times m}$ and the hash function is given by

$$f_A : \{0, \ldots, d-1\}^m \to \mathbb{Z}_q^n$$
$$f_A(\mathbf{x}) = A\mathbf{x} \bmod q.$$

If there is a collision $f_A(\mathbf{x}) = f_A(\mathbf{x}')$ between distinct inputs $\mathbf{x}$ and $\mathbf{x}'$, then $A(\mathbf{x} - \mathbf{x}') = 0$ and $\mathbf{x} - \mathbf{x}' \in L_q^{\perp}(A)$. Furthermore, because each element of $\mathbf{x} - \mathbf{x}'$ is in the set $\{-1, 0, 1\}$, we see that $\mathbf{x} - \mathbf{x}'$ is a short vector. Hence, an efficient algorithm that produces collisions for this hash function could be used to solve SIS in the lattice $L_q^{\perp}(A)$.

References

M. Ajtai. Generating hard instances of lattice problems. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, pages 99–108, 1996.

J.-Y. Cai and A. P. Nerurkar. An improved worst-case to average-case connection for lattice problems. In *Proceedings of the 38th Annual Symposium on Foundations of Computer Science*, pages 468–477. IEEE, 1997.

J. Hoffstein, J. Pipher, and J. H. Silverman. *An introduction to mathematical cryptography*, volume 1. Springer, 2008.

A. Korkine and G. Zolotareff. Sur les formes quadratiques. *Mathematische Annalen*, 6:366–389, 1873.

V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 1–23. Springer, 2010.

D. Micciancio and O. Regev. Worst-case to average-case reductions based on gaussian measures. *SIAM J. Comput.*, 37(1):267–302, 2007.

P. Nguyen and B. Vallée. *The LLL algorithm*. Springer, Berlin, Heidelberg, 2010.

C. Peikert. A decade of lattice cryptography. *Found. Trends Theor. Comput. Sci.*, 10(4):283–424, 2016.

O. Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):1–40, 2009.