A Report on Learning With Errors over Rings

Behzad Abdolmaleki

Supervisor: Prastudy Fauzi
University of Tartu

Abstract

Cryptography based on lattices is the use of conjectured hard problems on point lattices in $\mathbb{R}^n$ as the basis for security in cryptosystems. In this report, firstly we introduce lattices and some hard problems based on lattice then present some of the major developments in lattice-based cryptography. The main focus in this report is on the fundamental learning with errors (LWE) problems, Ring-LWE and its cryptographic applications.

1 Introduction

Lattice-based cryptography is an amazing and fast-growing area of research, because in part of conjectured security against quantum attacks, and some applications same as public key encryption, the NTRU cryptosystem, and also they have been used quite successfully in constructing secure cryptographic protocols that achieve functionalities such as fully homomorphic encryption. A large part of cryptographic constructions based on lattices are built under the average-case learning with errors (LWE) problem [Reg09], [MR09]or its more efficient kind of learning with errors over rings (Ring-LWE) [LPR10]. In fact, they are families of problems, which are represented by choosing a ring, an integer modulus, and an error distribution.

The underlying lattice problem for Ring-LWE, is the approximate Shortest Vector Problem on ideal lattices, which are mathematically structured lattices corresponding to ideals in the ring. To date, there is no any quantum or classical attacks for approximate Shortest Vector Problem has sufficiently better worst-case accomplishment on ideal lattices than on general lattices of the same dimension for well-chosen parameters [LPR13], [CP16].

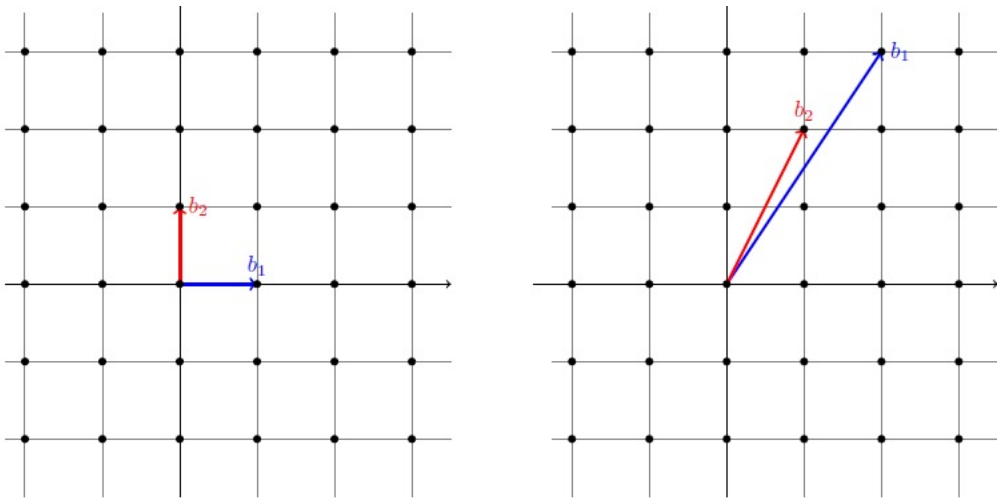

Figure 1: Two bases of the same lattice

2 Lattices

Definition 1 (Lattices). Given n linearly independent vectors $b_1, b_2, ..., b_n \in \mathbb{R}^n$, the lattice L is defined as :

$$L = \{a_1 b_1 + ... + a_n b_n \mid a_i \in \mathbb{Z}\}, \tag{1}$$

where the set of $(b_1, b_2, ..., b_n)$ is called a basis of the lattice. We will use a notational shorthand when dealing with bases and denoting them by a matrix B whose columns are the basis vectors $b_1, b_2, ..., b_n$.

We can also rewrite it a compact form:

$$L(B) = \{Bx \mid x \in \mathbb{Z}^n\}. \tag{2}$$

Note that in lattice-based cryptosystems it is common to use full rank lattices, and in this case the matrix B is also of full rank.

Definition 2 (Span). Given n linearly independent vectors $b_1, b_2, ..., b_n \in \mathbb{R}^n$, their span is defined as,

$$span(b_1, b_2, ..., b_n) = \{\sum_{i=1}^{n} b_i x_i \mid x_i \in \mathbb{R}\}. \tag{3}$$

Note the difference between Definition 1 of a lattice generated by a set of vectors which consists of all of its integer linear combinations and the Definition 2 of the span of a set of vectors which consists of all of its linear combinations with real coefficients. The crucial power of lattices comes from the fact that it is a discrete set (which the span is not) so we can conclude that $L(b_1, b_2, ..., b_n) \subset span(b_1, b_2, ..., b_n)$.

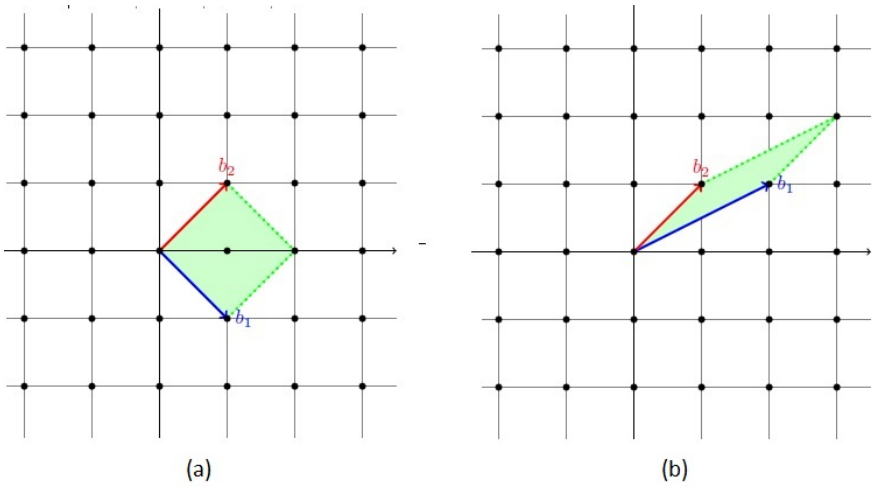

Figure 2: Different fundamental parallelepiped

2.1 Same lattice and many bases

Any lattice has infinitely many bases which the bases can have arbitrarily large coefficients (see Fig.1). A common question to ask is: how can we efficiently tell if two given bases B and B_1 generate the same lattice? There are two answers to this question: an *algebraic* answer and a *geometric* answer. In the algebraic domain we use *unimodular matrix* and also we can use of properties of *fundamental parallelepiped*.

Unimodular matrix: For any $x \in \mathbb{R}$, we will let $|x|$ represent the absolute value of x. A matrix $U \in \mathbb{Z}^{n\times n}$ is unimodular if $|det(U)| = 1$. Now we can state the characterization of equivalent bases in the following theorem,

Theorem 1.2 Given two full-rank bases $B, B_1 \in \mathbb{R}^{n\times n}$, the following two conditions are equivalent:

- $L(B) = L(B_1)$.
- There exists a unimodular matrix U such that $B_1 = BU$.

Fundamental Parallelepiped: Given n linearly independent vectors $b_1, b_2, ..., b_n \in \mathbb{R}^n$ their fundamental parallelepiped can be defined as follows,

$$\mathcal{P}(b_1, b_2, ..., b_n) = \{\sum_{i=1}^{n} b_i x_i \mid x_i \in \mathbb{R}, 0 \leq x < 1\}. \tag{4}$$

Thus, a fundamental parallelepiped is the region enclosed by the vectors $b_1, b_2, ..., b_n$.

Note that in Fig.2 (b), the vectors b_1 and b_2 form a basis of the lattice, and the parallelepiped associated to the basis does not contain any lattice point other than 0. On the other hand, in Fig.2 (a), the vectors b_1 and b_2 do not form a basis of the lattice, and the parallelepiped associated to the basis contains a non-zero lattice point.

2.2 Determinant of a lattice

Another quantity related to a lattice is its determinant which denoted by $det(L)$. Indeed the determinant of a lattice is the n-dimensional volume of its fundamental parallelepiped and it can be computed as the absolute value of the determinant of its basis matrix B.

Some facts about the determinant of a lattice are as follows:

1. The parallelepipeds associated with different bases of a lattice have the same volume. For example let B and B_1 be any two lattice bases so by Theorem 1.2, there is a unimodular matrix U such that $B_1 = BU$. Thus, $|det(B_1)| = |det(B)|\,|det(U)| = |det(B)|$ since $|det(U)| = 1$.

2. One interesting thing is the determinant of a lattice is inversely proportional to its *density*. So means that the larger the determinant, the sparser the lattice.

Shortest vector: In the lattice we would to define a vector λ_1 which is called the *shortest vector*, that can be generated by the basis B. We are interested in lower and upper bounds on λ_1, so to this aim we can use the Gram-Schmidt orthogonalization to define the lower bound and also by using the Minkowski's theorem and determinant's property, the upper bound can be defined as follows:

Theorem 2.2. Let B be a rank-n lattice basis, and let $B^{'}$ be its Gram-Schmidt orthogonalization,

$$\lambda_1(L) \geq min \left|b_i^{'}\right|, \quad \text{for } i = 1, ..., n, \tag{5}$$

where the vectors $b_i^{'}$ are the orthogonalization of the vectors b_i.

Mikowski Theorem: For any full-rank lattice $\mathcal{L}$ of rank n,

$$\lambda_1(L) \leq n^{1/2} \cdot det(L(B))^{1/n}. \tag{6}$$

2.3 Computational Problems

One way is to say that you introduce some algebraic problems related to lattices that they can be define as follows,

1 Given a basis B and a vector v, it is easy to decide if v is in $L(B)$.

2 Similarly, given two bases B_1 and B_2, it is easy to decide if $L(B_1) = L(B_2)$.

It makes sense that the problems in point of algebraic view are easy. But there are some hard problems in the lattice which can be defined base on geometry of lattice properties. In the rest of report some of the of them will presented.

Shortest Vector Problem (SVP_γ): Given a basis B, the goal is to find a vector in $L(B)$ which has length $\leq \gamma\lambda_1(L(B))$ where λ_1 is the length of the shortest non-zero vector.

Gap SVP_γ: Given a basis B and a lattice $L(B)$, outputs:

$$\text{Yes} \quad \text{if } \lambda_1 < 1$$

$$\text{No} \quad \text{if } \lambda_1 \geq \gamma.$$

Shortest Independent Vectors Problem ($SIVP_\gamma$): Given set basis B, and a lattice $L(B)$, find n linearly independent vectors in $L(B)$ of length $\leq \gamma\lambda_1(L(B))$.
Closest Vector Problem (CVP_γ): Given a set basis B and an arbitrary point v which is outside of the lattice points, find a lattice point that is at most γ times farther than the closest lattice point.
BDD$_d$: Given a lattice $L(B)$ and an arbitrary point v which is outside of the lattice points and has distance d from B, find the nearest lattice point to the point v.

3 Learning With Errors

Learning With Errors (LWE) is parameterized by positive integers n and q, and an error distribution χ over $\mathbb{Z}$. The error distribution χ is usually taken to be a discrete Gaussian of width q for some $\alpha < 1$, which is often called the relative error rate [Reg09].

3.1 LWE distribution

For a vector $s \in \mathbb{Z}_q^n$ called the secret, the LWE distribution $A_{s,\chi}$ over $\mathbb{Z}_q^n \times \mathbb{Z}_q$ is sampled by choosing $a \in \mathbb{Z}_q^n$ uniformly at random, choosing $e \leftarrow \chi$ and outputting:

$$(a, b = \langle a, s\rangle + e \bmod q), \tag{7}$$

where $\langle ., .\rangle$ is an inner product.

There are two main versions of the LWE problem: *search*, which is to find the secret given LWE samples, and *decision*, which is to distinguish between LWE samples and uniformly random ones. We additionally parameterize these problems by the number m of available samples, which we typically take to be large enough that the secret is uniquely defined with high probability.
Search-LWE$_{n,q,\chi,m}$. Given m independent samples $(a_i, b_i) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ drawn from $A_{s,\chi}$ for a uniformly random $s \in \mathbb{Z}_q^n$ (fixed for all samples), find s.
Decision-LWE$_{n,q,\chi,m}$. Given m independent samples $(a_i, b_i) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ where every sample is distributed according to either: (1) $A_{s,\chi}$ for a uniformly random $s \in \mathbb{Z}_q^n$ (fixed for all samples), or (2) the uniform distribution, distinguish which is the case.
We highlight several useful observations about search- and decision-LWE:

- Without the error terms from χ, both problems are easy to solve, because we can efficiently recover s from LWE samples by Gaussian elimination. (In the uniform case of decision-LWE, with high probability no solutions will exist.)

- It is often convenient to combine the given samples into a matrix $A \in \mathbb{Z}_q^{n \times m}$ (whose columns are the vectors $a_i \in \mathbb{Z}_q^n$) and a vector $b \in \mathbb{Z}_q^m$ (whose entries are the $b_i \in \mathbb{Z}_q$), so that for LWE samples we have:

$$b^t = s^t A + e^t (\text{mod } q),$$

where $e \leftarrow \chi^m$. In the uniform case of decision-LWE, b is uniformly random and independent of A.

3.2 Hardness of LWE

For any $m = poly(n)$, any modulus $q < 2^{poly(n)}$, and any (discretized) Gaussian error distribution χ of parameter $q\alpha > 2\sqrt{n}$ where $0 < \alpha < 1$, solving the decision-LWEn, q, χ, m problem is at least as hard as quantumly solving $GapSVP_\gamma$ and $SIVP_\gamma$ on arbitrary n-dimensional lattices, for some $\gamma = O(n/\alpha)$ [Reg09].

4 Ring-LWE

In [LPR10], [LPR13] introduced Ring-LWE, which it analogue of learning with errors, and also considered a special case of RLWE for rings of the form $R = \mathbb{Z}[X]/(X^n + 1)$ for power-of-two n.

RLWE is parameterized by a ring R of degree n over $\mathbb{Z}$, a positive integer modulus q defining the quotient ring $R_q = R/qR$, and an error distribution χ over R. Typically, one takes R to be a cyclotomic ring, and χ to be some kind of discretized Gaussian in the canonical embedding of R, which we can roughly think of as having an "error rate" $\alpha < 1$ relative to q.

4.1 Ring-LWE distribution

For an $s \in R_q$ called the secret, the ring-LWE distribution $A_{s,\chi}$ over $R_q \times R_q$ is sampled by choosing $a \in R_q$ uniformly at random, choosing $e \leftarrow \chi$, and outputting:

$$(a, b = \langle s, a \rangle + e \text{ mod } q). \tag{8}$$

The decision version of the RLWE problem is to distinguish between RLWE samples and uniformly random ones. As usual, we also parameterize the problem by the number m of available samples, which is sometimes left unspecified.

Decision-R-LWE$_{q,\chi,m}$: Given m independent samples $(a_i, b_i) \in R_q \times R_q$ where every sample is distributed according to either: (1) $A_{s,\chi}$for a uniformly random $s \in R_q$ (fixed for all samples), or (2) the uniform distribution, distinguish which is the case (with non-negligible advantage).

Just as in LWE, without errors the RLWE problem is easy, because in case (1) we can efficiently find s: given a sample (a_i, b_i) where $a_i \in R_q$ is invertible, we have $s = b_i \cdot a^{-1}$,

whereas in case (2) there will almost never be a single s that is consistent with all samples. Similarly, RLWE has a normal form, in which the secret s is chosen from the error distribution (modulo q), rather than uniformly [LPR10].

One of the advantage of RLWE is its compactness and efficiency: each sample (a_i, b_i) yields an n-dimensional pseudorandom ring element $b_i \in R_q$, rather than just a single pseudorandom scalar $b_i \in \mathbb{Z}_q$ as in LWE. In addition, ring multiplication can be performed in only quasi-linear $O(n)$ time using FFT-like techniques, so we can generate these n pseudorandom scalars in just $O(1)$ amortized time each. For example, this all yields a public-key encryption scheme with only $O(1)$-factor overheads in encryption/decryption time and ciphertext space, versus sending the plaintext in the clear [LPR13].

4.2 Hardness of RLWE

Like LWE, RLWE has a worst-case hardness guarantee, informally stated here:

Theorem 4.4.1 [LPR10]. For any $m = poly(n)$, cyclotomic ring R of degree n (over $\mathbb{Z}$), and appropriate choices of modulus q and error distribution χ of error rate $\alpha < 1$, solving the $\text{RLWE}_{q,\chi,m}$ problem is at least as hard as quantumly solving the SVP_γ problem on arbitrary ideal lattices in R, for some $\gamma = poly(n)/\alpha$.

4.3 Parameters

The parameters guarantees efficiency and security of the cryptosystems that in the rest of the report we will discuss about it in more detail.

- In the RLWE cryptosystems, the important parameters are the dimension n, modulus $q \geq 2$ and error rate $\alpha \ll 1$.

- Due to requiring [Reg09] and also to prevent of the $exp((\alpha q)^2)$-times attack presented in [LP11], the parameters n, q and α should chose such that $\alpha q > \sqrt{n}$. Indeed αq is the standard deviation (variance) of a Gaussian distribution over $\mathbb{Z}$.

- Another parameter which is as important as the previous ones, is γ which is the parameter related to the hardness in SVP_γ problem, it also should be $\gamma > n^{O(1)}/\alpha$.

- There is another limitation on choosing parameter q which it also should satisfy $q = 1 \bmod n$.

4.4 NTRU cryptosystem

The NTRU cryptosystem is parameterized by a polynomial ring $R = \mathbb{Z}[X]/(f(X))$, e.g., $f(X) = X^n + 1$ for an n that is a power of two, and a sufficiently large odd modulus q that defines the quotient ring $R_q = R/qR$. In brief, the public key is $pk = 2g \cdot s^{-1} \in R_q$ for two short polynomials $g, s \in R$, i.e., ones having relatively small integer coefficients, where the secret key s is also chosen to be invertible modulo both q and two.

Encryption phase: The encryption involves multiplying pk by a short blinding factor $r \in R$ and adding a short error term $e \in R$ that encodes the message bits in its coefficients modulo two, to get a ciphertext $c = pk \cdot r + e \in R_q$.

Decryption phase: This phase can be done by multiplying the ciphertext by the secret key to get $c \cdot s = 2g \cdot r + e \cdot s \in R_q$ and interpreting the result as a short element of R, which works because all of g, r, e, and s are short. From this one recovers $e \cdot s$ modulo two, and thereby e modulo two, to recover the message bits. There are slightly more efficient variants of this basic template, e.g., choosing $s = 1 \pmod 2$, so that $e \cdot s = e \pmod 2$ [SS11].

4.5 LPR Cryptosystem

As one example application [LPR10], here we sketch a simple and efficient semantically secure public-key cryptosystem. The key-generation algorithm chooses a uniformly random element $a \in R_q$ as well as two random "small" elements $s, e \in R$ from the error distribution. It outputs s as the secret key and the pair $(a, b = a \cdot s + e) \in R_q^2$ as the public key.

To encrypt an n-bit message $m \in \{0,1\}^n$, we view it as an element of R by using its bits as the coefficients of a polynomial. The encryption algorithm then chooses three random "small" elements$r, e_a, e_b \in R$ from the error distribution and outputs the pair (c_a, c_b) as the encryption of m.

The structure of the LPR encryption is shown in Fig.3 with more details [LPR10], [LPR13].

Alice		**Bob**
$s \leftarrow \chi$.		
$e_{pk} \leftarrow \chi$.		
	pk_a	$r, e_a, e_b \leftarrow \chi$.
		$M \leftarrow z_t$
$pk_b \leftarrow pk_a \cdot s + e_{pk}$	$\xrightarrow{pk_b}$	
		$c_a \leftarrow pk_a \cdot r + e_a$
		$c_b \leftarrow pk_b \cdot r + e_b + [q/t] \cdot M$
	$\xleftarrow{(c_a, c_b)}$	
$computes : [\frac{c_b - c_a \cdot s \ (\text{mod q})}{[q/t]}] \ (\text{mod } t)$		

Figure 3: LPR encryption.

The decryption phase of LPR cryptosystem can be written as follows,

$$[\frac{c_b - c_a \cdot s \pmod q}{[q/t]}] \pmod t = [\frac{(pk_b \cdot r + e_b + [q/t] \cdot M) - (pk_a \cdot r + e_a) \cdot s \pmod q}{[q/t]}] \pmod t$$

$$= [\frac{pk_a \cdot s \cdot r + e_{pk} \cdot r + e_b + [q/t] \cdot M - pk_a \cdot s \cdot r - e_a \cdot s \pmod q}{[q/t]}] \pmod t$$

$$= [\frac{e_{pk} \cdot r + e_b - e_a \cdot s \pmod q}{[q/t]} + M \pmod q] \pmod t = M \pmod t.$$

Choosing parameters: Generally in this cryptosystem there are integers n, m, ℓ, q, r, t, and $0 < \alpha < 1$. The secret $S \in \mathbb{Z}^{n \times \ell}$ is known as a private key and $PK_a \in \mathbb{Z}_q^{m \times n}$ is chosen uniformly random and the error $E \in \mathbb{Z}_q^{m \times \ell}$ by choosing each input according to the nose distribution χ_α. The public key is set $(PK_a, PK_b = PK_a \cdot S + E)$.

We present some properties of the cryptosystem in the following, which all sizes are in bits, and all logarithm is based on two.

- Secret key size: $n \cdot \ell \cdot \log q$.
- Public key size: $m \cdot (n + \ell) \cdot \log q$.
- Message size: $\ell \log t$.
- Ciphertext size: $(n + \ell) \cdot \log q$.
- Number of operations for encryption: $O(m(1 + n\ell))$, (per bit).
- Number of operations for decryption: $O(n)$, (per bit).

5 Conclusions

In this report, we present a review on lattices and some hard problems in this area. Then we detailed main intuition and efficiency of the *learning with error* (LWE) and the *Ring-LWE*. Finally we introduced NTRU and LPR cryptosystems and their parameters for implementation.

References

[CP16] Eric Crockett and Chris Peikert. Challenges for ring-lwe. Technical report, Cryptology ePrint Archive, Report 2016/782, 2016. http://eprint. iacr. org/2016/782, 2016.

[LP11] Richard Lindner and Chris Peikert. Better key sizes (and attacks) for lwe-based encryption. In *Cryptographers? Track at the RSA Conference*, pages 319–339. Springer, 2011.

[LPR10] Vadim Lyubashevsky, Chris Peikert, and Oded 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.

[LPR13] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. *Journal of the ACM (JACM)*, 60(6):43, 2013.

[MR09] Daniele Micciancio and Oded Regev. Lattice-based cryptography. In *Post-quantum cryptography*, pages 147–191. Springer, 2009.

[Reg09] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):34, 2009.

[SS11] Damien Stehlé and Ron Steinfeld. Making ntru as secure as worst-case problems over ideal lattices. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 27–47. Springer, 2011.