Project 1
Lattice-based Cryptography

Xiuqi Zhang Zikun Zhou Heyin Shen Anna Lee

JI SJTU

July 22, 2021

- Lattice-based cryptography is a general set of cryptography that involves lattices.
- Lattice-based cryptosystems covers encryption, signatures and hash functions.
- Lattice-based cryptosystems are (still) post-quantum computing se- cure, and have proved security basing on worst-case scenario.

Introduction of Lattice

Less formally (while not indicating less accurate), lattice can be viewed as a set of points

$$L = \{a_1 v_1 + a_2 v_2 + ... + a_n v_n | a_i \in \mathbb{Z}\} \tag{1}$$

$(v_1, v_2, ..., v_n) \in \mathbb{R}^n$ and they are linear independent

Example

All points forming the lattice can be generated by linear combination with integer coefficients. We denote the set $B = \{v1, v2, ..., vn\}$ as basis of the latice. Then the lattice can also be denoted as $L(B)$. Apparently basis is not unique.

Column View

- Changing order of $\forall v_i, v_j \in B$ does not chagne the lattice generated.
- $\forall v_i \in B, L(B') = L(B)$ where $B' = (B/v_i) \cup \{-v_i\}$.
- Linear Combination: for some $v_i, v_j \in B$, let $v_i = v_i + kv_j$ where $k \in \mathbb{Z}$.

Theorem

Theorem

$$L(B_1) = L(B_2) \iff B_1 = B_2 U \tag{2}$$

where U is a unimodular U.

Notice that no matter which basis is chosen, the fundamental parallelpiped has the same volume. This can be proved by imagining a very large space where the shape of each small region can be ignored. Since the number of small regions is the same the volumn of them should be the same.
We define the determinant of a lattice $L(B)$ as $\det(L) = |\det(B)|$, which is the volume of the fundomental parallelepiped.

Successive Minima

One very important element of a lattice is the shortest vector in the lattice. We denote the length (Euclidean norm) of the shortest vectors in $\mathcal{L}$ as $\lambda_1(\mathcal{L})$, the second shortests as $\lambda_2(\mathcal{L}),\ldots$, etc.

Gram-Schmidt Orthogonalization

Gram-Schmidt Orthogonalization is a process which takes a set of linearly inde- pendent vectors and output a set of orthogonal vectors with same cardinality. It projects each vector on the orthogonal complement of the previous vectors. A formal design is as Eq. 3

For vector series $B = b_1, b_2, \ldots, b_n$, GSO vector set $\widetilde{B} = \tilde{b}_1, \tilde{b}_2, \ldots, \tilde{b}_n$ is as

$$\tilde{b}_i = b_i - \sum_{j=1}^{i-1} \mu_{i,j} \tilde{b}_j, \text{ where } \mu_{i,j} = \frac{\left\langle b_i, \tilde{b}_j \right\rangle}{\left\langle \tilde{b}_j, \tilde{b}_j \right\rangle} \tag{3}$$

Minkowski's Theorem

Theorem

Minkowski's Theorem: *For any lattice Λ and convex zero-symmetric set S, volume of which is larger than $2^n \det(\Lambda)$, there must exists some lattice point in S. (which is the upper bound of smallest lattice).*

Theorem

Inference of Minkowski's Theorem:

$$\forall \Lambda, \lambda_1(\Lambda) \leq \sqrt{n} \cdot \det(\Lambda)^{\frac{1}{n}} \tag{4}$$

Shortest[1] Vector Problem is the most important and basic computation problem about lattice. From Theorem 2 and Theroem ?? we know about the upper and lower bound of shortest vector, however it does not provide a way to find such vector. This problems remains to be a hard problem, and works in the field of lattice compuatation problems as basic as SAT problem in NP-complete.

[1] If not specified, all discussion about length in this report is Euclidean norm.

Hardness

In Euclidean distance, which as most scenario this report is discussed on, we only know that by applying randomized reductions the problem is NP-hard [?]. If considering uniform norm, the problem has already been proved to be NP-hard [?].

GapSVP

GapSVP$_\gamma$ is variant of SVP$_\gamma$, in which we try to know that whether $\lambda(\mathcal{L}(B))$ is not bigger than one, or larger than γ, where γ is some function $f(n)$, where n is the dimension of the space. Notice that it is a promise problem, which means input should make sure the result will in one of the conditions.

The closest vector problem is about in following scenario:
A basis B and a lattice L, and some vector $v \in \vec{(B)}$.
Try to find:
A vector $v' \in L$, which is closest to v.

Hardness

urther from the conclusion that we can solve SVP efficiently if we can solve CVP, Goldreich et al. proved that CVP is at least harder than SVP at any aspect [?]. Dinur et al. proved that, with factor $n^{c/\log\log n}$ for some constant $c > 0$, CVP is NP-hard to approximate [?].

In summary, the hardness about the problem (currently) according to factor is as Figure

Anti-quantum Attack

This is the main advantage the lattice-based cryptography has over the traditional public-key cryptographies as the security of the latter has been challenged in the context of a quantum computer. In fact, the security guarantee of most traditional public-key cryptographies is established based on the hardness of the factorization of large integers, the discrete logarithm and other related problems. However, with the proposition of the quantum algorithm, both factoring and discrete algorithm-based problems are solvable in polynomial time complexity. Therefore, in the research of modern public-key cryptosystem, lattice-based cryptography outstands for its ability to resist quantum attacks.

The main mains problems involved in the lattice-based cryptosystem are based on the calculation of vectors without the engagement of large prime integers, and the algorithm also enjoys relative high concurrency, leading to high efficiency in practice.

The security of the lattice-based cryptography can be guaranteed because it is built based on the "worst case to average case reduction". In other words, the hardness of finding the solutions of a certain problems in average cases is no less than finding the solutions in the worst cases. In practice, efficient lattice-based algorithms, such as those that are based on LWE, their worst-case hardness results may be unknown. Conversely, cryptographic that are based on factoring, which we know is hard in the worst case, can still be decrypted easily when it is easy to solve the factorization on average input. While as lattice-based cryptosystem is hard to solve in the worst case, it possesses very high security.

Introduction

The short integer solution problem (SIS) is an average-case problem based on the worst-case lattice problem. Its difficulty is guaranteed by the short vector problem

Definition

There is a relationship between the SIS and the lattice problems: Let S be the set of all solution z, such that $\mathrm{Az} = 0 \pmod{q}$. As this set is additive, S is a lattice. Therefore, the SIS problem can be regarded as the problem to find a short vector in S. Now we have a new representation of lattices constructed from matrix A:

$$L^{\perp}(\mathrm{A}) = \{\mathrm{z} \in \mathbb{Z}^m : \mathrm{Az} = 0 \pmod{q}\} \tag{5}$$

Using worst-case to average-case reduction, solving the SIS problem in $L^{\perp}(\mathrm{A})$ is approximately solving SVP in all lattices, which will be discussed later.

Hash Function

An one-way & collision-resistant hash function can be easily implied from the SIS:
Set $m > n \lg q$. Given random A in $\mathbb{Z}_q^{n \times m}$, define the hash function $f_A : \{0,1\}^m \to \mathbb{Z}_q^n$ as:

$$f_A(z) = Az \tag{6}$$

A collision $f_A(z) = f_A(y)$ yields a solution $z - y$ of SIS for A, as $z - y$ is in $\{0,1\}^m$ and satisfy $A(z - y) = 0$

Theorem

If $s > 5\lambda_n(B)$, *and* $X \sim \rho_s(x) = (1/s)^n e^{-\pi||x||^2/s^2}$, *then*

$$\Delta(X \mod B, \text{Uniform}(B)) < n2^{-110} \tag{7}$$

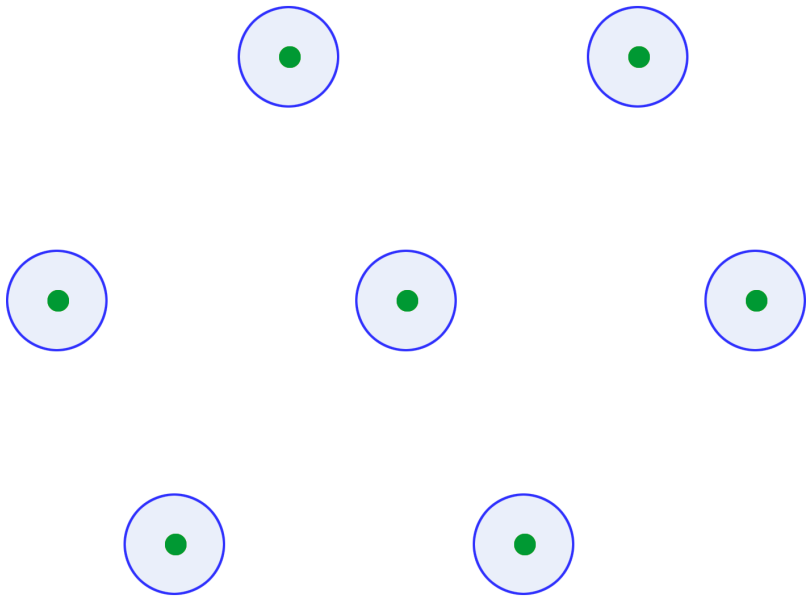

Figure: The distribution when s is small

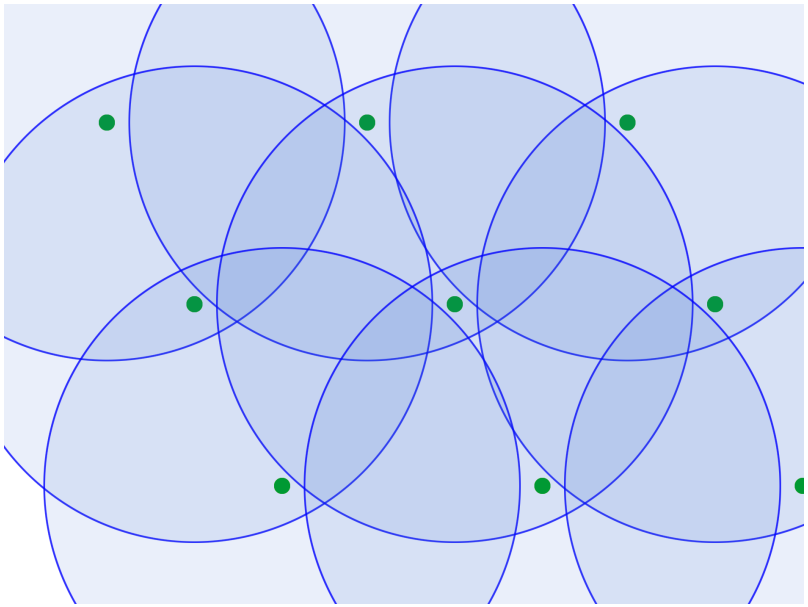

Figure: The distribution when s increases

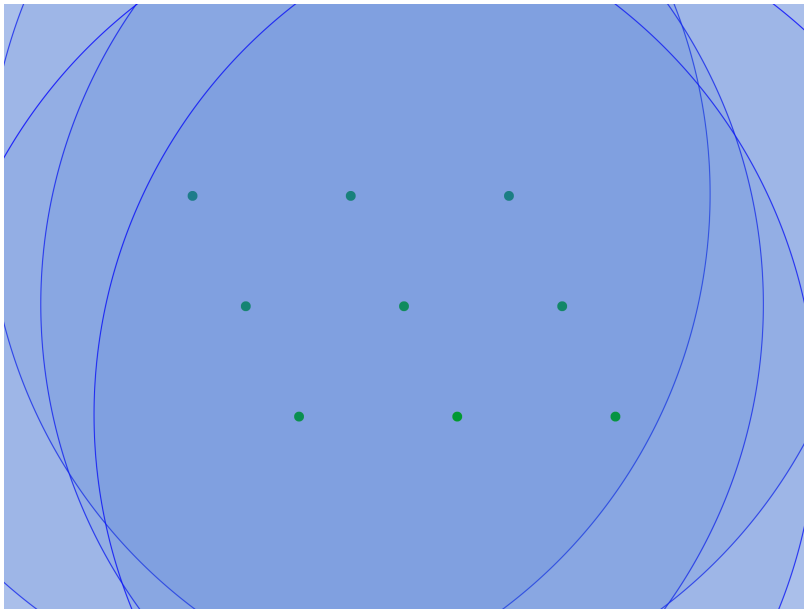

Figure: The distribution when s is large enough

Consider an additive group $\mathbb{T} = \mathbb{R}/\mathbb{Z}$ which is constructed modulo one. For error, produce a fixed probability distribution over $\mathbb{T}$ denoted as φ.
We than define a distribution over $\mathbb{Z}_q^n \times \mathbb{T}$ as

1. Randomly get a vector $a \in \mathbb{Z}_q^n$ following uniform distribution.
2. Randomly get a number $\epsilon \in \mathbb{T}$ following distrubtion φ.
3. Compute addtion and division under $\mathbb{T}$, inner product in $\mathbb{Z}_q^n$ calculate $t =< a, s > /q + e$.
4. Pair (a, t) is a sample.

Denote the entire sample set as $A_{s,\varphi}$.

Algorithm 1 Solving SVP using SIS oracle

for m times **do**

Pick a random lattice point v_i

Gaussian sample a point $a_i = v_i + r_i$ round to $\mathbb{Z}_q^n$ around v_i

end for

$A = (a_1, a_2, ..., a_m) \rightarrow$ SIS oracle

SIS oracle $\rightarrow$ z

Output the short lattice vector: Rz

With above definition, we define the LWE search problem as trying to find s, given polynomial amout of samples from $A_{s,\varphi}$ Most times we studied a special case of LWE, where φ is the nomal distrubtion as origin point with variance of $\frac{\alpha^2}{2\pi}$, i.e. $e^{-\pi(|x|/\alpha)^2}/\alpha$.

On the other hand, LWE decision problem is to tell the difference between a LWE distributed input and a uniformly random input.

Peikert proved that the worst case of LWE can be reducted to GapSVP in polynomial time, considering a approximate output [?].

Fault Attacks

Xiuqi Zhang
Zikun Zhou
Heyin Shen
Anna Lee

Although lattice-based Cryptography is the most safe algorithm, there still exists the implementation level fault, which makes the attack possible. Here we mainly talked about three fault attacks

Xiuqi Zhang
Zikun Zhou
Heyin Shen
Anna Lee

Algorithm 2 Initiation

$s_l, e_l \leftarrow$ *polynomials with coefficients from* χ_α *distribution*

$p_l \leftarrow as_l + 2e_l$

return p_l

Algorithm 3 Response

$E \leftarrow \left\{-\left\lfloor \frac{q}{4} \right\rfloor, \ldots, \left\lfloor \frac{q}{4} \right\rceil\right\} of Zq = \left\{-\frac{q-1}{2}, \ldots, \frac{q-1}{2}\right\}$

$s_R, e_R \leftarrow$ *polynomials with coefficients from* χ_α *distribution*

$p_R \leftarrow as_R + 2e_R$

$e'_R \leftarrow samplefrom\chi_\alpha$ *distribution*

$k_R \leftarrow p_I s_R + 2e'_R$

for each coefficient k_{R_i} of k_R **do**

  if $k_{R_i} \in E$ **then**

    $w_i \leftarrow 0$

  else

    $w_i \leftarrow 1$

  end if

end for

$sk_R = \left(k_R + w \cdot \frac{q-1}{2}\right) \bmod q \bmod 2$

return p_R, w

Loop-Abort Faults on Lattice-based Signature

in 2018, Espitau proposed a loop-abort faults applied on lattice-based signature. [?] In this research, they applied two attack but with roughly the same type of faults, so that the attacker could lead to a loop inside the algorithm of the signature and abort early. The first attack is in the Fiat-Shamir family. By inputing a fault in the loop, they could get the commitment value, which is a random polynomial. This could leak enough information for recovering the entire signing key. For the GPV-based hash-and-sign signature scheme, when it is applied into the early loop abort, the original ciphertext will become a linear combination of the parts of the secret lattice. Therefore, in this way, we could recover the key by repeating this process.

Xiuqi Zhang
Zikun Zhou
Heyin Shen
Anna Lee

One classical way of attacking the lattice-based schemes is the physical attack, since there are little research results on the physical security of lattice-based cryptography. Moreover, physical attack is easier comparing to other attack. [?]

Xiuqi Zhang
Zikun Zhou
Heyin Shen
Anna Lee

In 2018,Yupu Hu and Huiwen Jia presented several efficient attacks on GGH. [?] In the attack testing experiments, the most important part is the specific modular operations so as they could reduce the effect of noise of GGH. In this way, with little lattice-reduction tools, MKE(multipartite key exchange) would be attacked. To break WE(witness encryption), they made use of the hardness of exact-3-call(X3C) problem. By combining these two steps, enough useful information for attack could be gotten and the attack towards GGH succeeds.

Xiuqi Zhang
Zikun Zhou
Heyin Shen
Anna Lee

Algorithm 4 The Ciphertext-Only Attack

Input: $v_{\sigma(0)}, v_{\sigma(1)}, ..., v_{\sigma(m)}$: The public key

b: the maximum of the public key

C: the ciphertext

Output: $M = (a_0, a_1, ..., a_m)$: The corresponding message

"Failure": message corresponding to errors

Use public keys to compute the Gram-Schmidt orthogonalization vectors:$v^*_{\sigma(0)}, v^*_{\sigma(1)}, ..., v^*_{\sigma(m)}$

if $min_{0 \le i \le m} \parallel v^*_{\sigma(i)} \le b \rightarrow$"Failure" **then**

halt $3 - 8$

end if

$i := m$

repeat

$$a_i := \left\lceil \frac{<v^*_{\sigma(i)}, C>}{\|v^*_{\sigma(i)}\|^2} \right\rceil$$

$C := C - a_i v_{\sigma(i)}, i := i - 1$

until $i < 0$