Foundations of Lattices in Cryptography: An Introduction

Lena Heimberger

Last Edit: April 2024

Disclaimer. The following lecture notes accompany the lecture *Mathematical Foundations of Cryptography* and were taught in the Winter Term 2022/23 at Graz University of Technology. While I don't teach the course anymore, I regularly update the notes to accompany other classes. The class is aimed at master students from the degree programs in Computer Science, Software Engineering, and Information and Computer Engineering students, which is why some detail was sacrificed for easier understanding or illustrative exercises. We finished after covering Section 5, but I extended the script with some personal notes over time.

1 Motivation

Lattices are everywhere! Solid-state physics describes lattices as crystalline 3-d structures. Sodium chloride(see Figure 1), which we use every day to season our food, is also a good example of a real-life lattice.

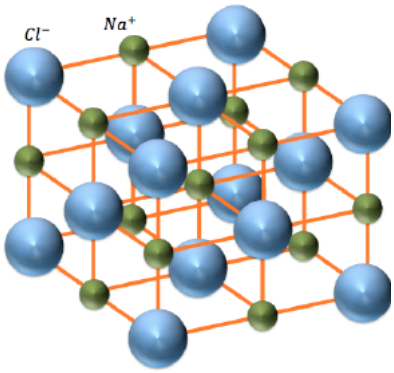


Figure 1: A three-dimensional lattice, better known as table salt. *From: Theoretical study of damage accommodation in salt subject to viscous fatigue, Cheng Zhu, Chloé Arson and Amade Pouya, Proceedings of the Conference on Mechanical Behavior of Salt, SALTMECH VIII, May 2015.*

Another application is number theory, which studies the relationships between different numbers, which is useful in the context of the RSA algorithm. Lattice methods are used for cryptanalysis of this structure. You may remember the LLL algorithm from the cryptanalysis lecture.

In addition, can use lattices to study group theory in greater detail: lattices are a great introduction to algebra and group theory. Moreover, you interact with them daily: Their non-commutative variant is used for coding theory, the commutative variant for lattice cryptography, both of

which are applications for cryptography as there are several conjectured hardness assumptions, which are believed to be resistant against cryptanalysis by quantum computers. Lattice cryptosystems are used for their efficiency, simplicity, and parallelizable properties. They have strong security guarantees and excellent hardness assumptions. We can build efficient hash functions [BBL+15], signature schemes [HHGP+10; LDK+20; PFH+20], encryption and key encapsulation schemes [HHGP+10; SAB+20; CDH+20; DKR+20] and advanced constructions, like fully homomorphic encryption [Bra12; FV12], from lattice schemes.

To illustrate how useful lattices are in cryptography, consider the third round of the NIST PQ not-a-competition: $\frac{3}{4}$ of the key encapsulation submissions and $\frac{2}{3}$ of the signature candidates were lattice-based. The single finalist KEM is Crystals-Kyber and based on lattices. For digital signatures, two out of three finalists are lattice-based, the third candidate, SPHINCS+, was largely added due to a wish for diversification to not solely rely on lattices.

To conclude the introduction, lattices are so useful because they are an abstract, algebraic structure. In the accompanying lecture, we want to study them further and learn about the fundamentals and theorems.

1.1 Credit and further reading

Several figures were taken from the library TikZ for cryptography (https://www.iacr.org/authors/tikz/).

I used the following references to compile these notes:

- *An Introduction to Mathematical Cryptography* by **Jeffrey Hoffstein, Jill Pipher, J.H. Silverman** is a great resource for the entirety of this course.
- *The LLL Algorithm* is an excellent book by **Phong Q. Nguyen, Brigitte Vallée (Eds.)**, and a really good resource to understand the algorithm fully.
- *A decade of lattice cryptography* by **Chris Peikert**[Pei16], probably the main place to look if you want to learn more about lattice *cryptography.*
- The lecture notes of **Daniele Micciancio**'s course *CSE206A: Lattices Algorithms and Applications (Spring 2014)* (https://cseweb.ucsd.edu/classes/sp14/cse206A-a/index.html) are good.
- **Daniel Dadush** has an excellent course on lattices from a mathematical perspective. The lecture Please find the lecture here https://homepages.cwi.nl/~dadush/teaching/lattices-2018/. I strongly encourage you to read the lecture notes.
- The *lattice club* has a great, general repository on lattice cryptography, courses, and surveys, as well as PhD Theses and more resources, like implementations and tools. See here: https://thelatticeclub.com/

In addition, a part of these lecture notes is based on slides by Lukas Helminger and Maria Eichlseder.

2 Applications

To further motivate the study of lattices we'll start with a brief discussion of cryptographic applications. They are further described in several different lectures, denoted in *cursive*.

2.1 Fully Homomorphic Encryption

Fully homomorphic encryption is often branded as *computing on encrypted data*. Classical methods based on modular multiplication [RSA78; ElG85] or addition [CF85; Pai99] enable partially homomorphic encryption. In an ideal fully homomorphic encryption setting, we can perform arbitrary multiplications on the plaintext and the ciphertext that are also applied to the plaintext.

Privacy Enhancing Technologies, VO/KU, 705.054/705.055 gives an excellent practical introduction to partial and homomorphic encryption.

2.2 Attribute-based Encryption

In attribute-based encryption, the secret key and resulting ciphertext are dependent on certain *attributes* defined by the protocol functionality. The decryption of the ciphertext is only possible if the key used for the decryption has the same attributes. A nice popular science article on ABE for digital wallets is here (https://ntt-research.com/ntt-research-cis-cryptography-attribute-based-encryption/).

This is further discussed in *Selected Topics in Information Security- Modern Public Key Cryptography, VU, 705.008.*

2.3 Post-Quantum Cryptography

As mentioned in the beginning, it seems like cryptosystems using noisy systems of equations are not only secure against adversaries with quantum computers, but also efficient enough to be deployed as a replacement for today's cryptographic protocols based on discrete logarithm and factoring problems.

This is discussed in *Cryptography, VO/KU, 705.066/705.067* and *Selected Topics in Information Security- Modern Public Key Cryptography, VU, 705.008.*

2.4 Cryptanalysis

Lattices are extensively used in cryptanalysis to (attempt to) attack cryptosystems by using approximation algorithms. For example, Bleichenbacher's attack recovering m given $c \equiv m^e$ uses lattices to find a well-formed plaintext message from the ciphertext [Ble98].

Consider taking *Cryptanalysis, VO/KU, 705.068/705.069* to learn more about this.

3 Intuition

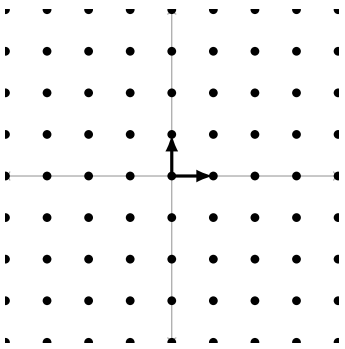


Figure 2: Points in a discrete, two-dimensional lattice. https://www.iacr.org/authors/tikz/

Consider the (fairly boring) basis vectors $(1, 0)$ and $(0, 1)$ in Figure 2. Combining these vectors is sufficient to traverse all integer coordinates and therefore generates the lattice $\mathbb{Z}^2$, the two-dimensional integer lattice. Notation-wise, a power will always describe the dimension, so e.g. $\mathbb{R}^{11}$ would be the eleven-dimensional lattice over the reals. A subscripted number will denote the denominator, e.g. the lattice $\mathbb{Z}_4$ is the integer lattice mod 4. This notation can be combined: the lattice $\mathbb{Z}_3^{15}$ is the 15-dimensional lattice over the integers mod 3.

Based on the lattice example in Figure 2, we will now define a general lattice.

Definition 3.1 (Lattice)**.** An n-dimensional lattice $\mathcal{L}$ is any subset of $\mathbb{R}^n$ that is both:

- An **additive subgroup** (Remember, this means that adding two elements from the subgroup together will always produce another element of the subgroup).
- **Discrete.**

The lattice $\mathcal{L}$ is generated a *basis*, which are linearly independent vectors $v_1, \ldots, v_n$. The elements of the lattice are a set of linear combinations of $v_1, \ldots, v_n$ with coefficients in $\mathbb{Z}$ is denoted as

$$\mathcal{L} = \{a_1 v_1 + \cdots + a_n v_n \ : \ a_1, \ldots, a_n \in \mathbb{Z}^k\}.$$

In addition to all these combinations, note that $0 \in \mathcal{L}$ due to the combination of $0 * v_1 + \cdots + 0 * v_n$. It also serves as a nice starting point! You may notice that the basis is not necessarily unique. This will be important when we construct cryptographic primitives and protocols.

Let's look at another, slightly more interesting lattice in $\mathbb{R}_2$. It is two-dimensional, so we have two basis vectors $\boldsymbol{v}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $\boldsymbol{v}_2 = \begin{pmatrix} 1/4 \\ \sqrt{2} \end{pmatrix}$. The vectors span the lattice in Figure 3.

The number of basis vectors is the *rank* of the lattice. The number of linearly independent vectors n is the dimension. We restrict ourselves to full-rank lattices, where $k = n$.

While lattice dimensions are usually quite high(their exact parameterization depends on the use case), we will largely work with very low dimensions in this class as we will compute on the blackboard. If you want to play around with how many dimensions you need, I'd refer you to the Lattice Estimator: https://lattice-estimator.readthedocs.io/.

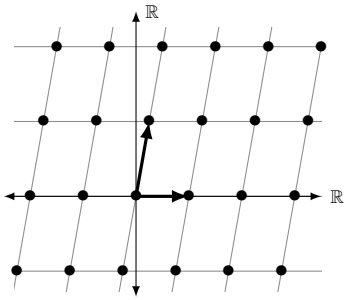


Figure 3: The lattice defined by $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$, over the reals.

3.1 Vector Spaces

Before we dive into vector spaces, we will do a quick detour into vector spaces. A *vector space* V is a subset of $\mathbb{R}^m$ that is closed under addition and under scalar multiplication by elements of $\mathbb{R}$. A *linear combination* of the vectors $v_1, \ldots, v_k$ is any vector of the form

$$w = \alpha_1 v_1 + \cdots + \alpha_k v_k, \text{ with } \alpha_1, \ldots, \alpha_k \in \mathbb{R}.$$

Definition 3.2 (Span)**.** The linear space formed by all vectors $\{v_1, \ldots, v_k\}$ is called the *span*. In particular, a lattice can be defined as the span of its basis vectors.

Remember that vector spaces allow real combinations, contrary to lattices: $0.17 \cdot \boldsymbol{v_1} - 1.72 \cdot \boldsymbol{v_2}$ is a valid combination. Lattices only allow integers. Regardless, we still briefly discuss the general subject of vector spaces. It will help you to feel familiar with lattices and with more advanced constructions.

Before, we mentioned that the vectors need to be *linearely independent*.

Definition 3.3 (Linear Independence)**.** A set of vectors $v_1, \ldots, v_k \in V$ is linearly independent if there exists only one trivial solution such that

$$\alpha_1 v_1 + \cdots + \alpha_k v_k = 0 \Rightarrow \alpha_1 = \cdots = \alpha_k = 0.$$

We will now show why this is important.

3.1.1 Special Cases

We will first consider some special cases for a vector basis. They were taken from the linear algebra lecture by Jiwen He at the University of Houston https://www.math.uh.edu/~jiwenhe/math2331/lectures/sec1_7.pdf. We will briefly discuss this in the lecture perhaps, but I find it quite interesting.

One vector Consider the set containing one vector $\{\boldsymbol{v}_1\}$. There is a single solution x_1 s.t. $x_1 \boldsymbol{v}_1 = 0$, which is $x_1 = 0$.

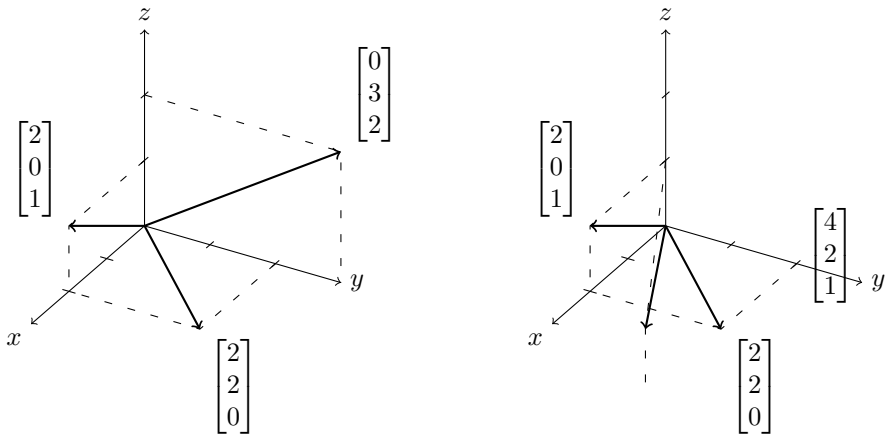


Figure 4: Linearly independent and linearly dependent vectors. Drawing inspired by https://tex.stackexchange.com/questions/414403/drawing-vectors-on-3-d-coordinate-system.

Two vectors

Example 3.1. Are the following vectors independent? $\boldsymbol{v}_1 = \begin{pmatrix}2\\4\end{pmatrix}$ $\boldsymbol{v}_2 = \begin{pmatrix}1\\2\end{pmatrix}$

No, they are multiples of each other.

Example 3.2. What when we change $\boldsymbol{v}_2$? $\boldsymbol{v}_1 = \begin{pmatrix}2\\4\end{pmatrix}$ $\boldsymbol{v}_2 = \begin{pmatrix}2\\3\end{pmatrix}$ Try to find $c, d \neq 0$ s.t. $c\boldsymbol{v}_1 + d\boldsymbol{v}_2 = 0$

The zero vector Consider the set containing the zero vector $\{\boldsymbol{v}_0\}$. It has infinitely many solutions k for $k \cdot \boldsymbol{v}_0$ and therefore the vector set cannot be linearly independent.

Too many vectors

Theorem 3.1. A set containing more vectors than entries in each vector is linearly dependent. I.e. any set $\{\boldsymbol{v}_1, \boldsymbol{v}_2 \cdots, \boldsymbol{v}_p\}$ is linearly independent if $p > n$.

Proof. Construct the matrix $A = [\boldsymbol{v}_1 \boldsymbol{v}_2 \cdots \boldsymbol{v}_p]$, an $n \times p$ matrix. The equation $Ax = 0$ has more variables than equations, and hence the columns of A are linearly independent. This is trivial to see: In a vector space spanning dimension n with more than n vectors, some have to be linearly dependent. □

All of those approaches bring us closer to our actual solution.

While it is clear in the figure the three vectors span $\mathbb{R}^3$, we may not want to draw the vectors all the time- we also cannot do this with more than three dimensions, as we do not have a good abstraction.

To determine linear independence in a more general setting we check for free variables. If there are free variables, there are infinitely many non-trivial solutions. Each linear dependence relation corresponds to a nontrivial solution $Ax = 0$. In Figure 4, I drew the graphical difference between dependent and independent vectors.

Example 3.3. Take the three vectors $x_1 = \begin{pmatrix}1\\3\\5\end{pmatrix}$, $x_2 = \begin{pmatrix}2\\5\\9\end{pmatrix}$, $x_3 = \begin{pmatrix}-3\\9\\3\end{pmatrix}$.

We are trying to show that there is only one trivial solution such that $\alpha_1 \cdot \begin{pmatrix}1\\3\\5\end{pmatrix} \alpha_2 \cdot \begin{pmatrix}2\\5\\9\end{pmatrix} \alpha_3 \cdot \begin{pmatrix}-3\\9\\3\end{pmatrix}$, which is that $\alpha_1 = \alpha_2 = \alpha_3 = 0$.

Writing the vectors as a matrix, we get:

$$\begin{bmatrix}1 & 2 & -3 & 0\\3 & 5 & 9 & 0\\5 & 9 & 3 & 0\end{bmatrix} \sim \begin{bmatrix}1 & 2 & -3 & 0\\0 & -1 & 18 & 0\\0 & -1 & 18 & 0\end{bmatrix} \sim \begin{bmatrix}1 & 2 & -3 & 0\\0 & -1 & 18 & 0\\0 & 0 & 0 & 0\end{bmatrix}$$

Hence x_3 is a free variable and we have an arbitrary number of solutions. The vectors are not linearly independent.

Let's get back to lattices. We now know how to represent lattices of dimension n with n or vectors. We have even more constraints, as we only consider integer multiples of the vectors. We have seen how lattices are discrete versions of subspaces. In cryptography, we consider *q-ary* matrices where each vector entry is taken $\mod q$, some integer.

3.2 Minimum distance

The minimum distance λ of a lattice $\mathcal{L}$ is the length of the shortest nonzero lattice vector.

$$\lambda(\mathcal{L}) = \mathsf{inf}\{||\mathbf{v} : \mathbf{v} \in \mathcal{L}\}\{\mathbf{0}$$

Daniele Micciancio has an entire great lecture on the minimum distance, with an application to prove that any prime number p congruent to 1 mod 4 can be written as the sum of two squares https://cseweb.ucsd.edu/classes/sp07/cse206a/lec3.pdf. As mentioned, I would encourage you to think about it at home!

3.3 Finding a good basis

Having a good basis is an important criterion for the difficulty of a lattice problem.

Let's look at Figure 5. If we want to traverse between the points of a lattice, the upper lattice seems like a much friendlier option. We will first discuss some preliminaries to then study some algorithms to find good bases, so we can work

with something close to the upper image, even if we only have $\mathbf{v}_1'$ and $\mathbf{v}_1'$ as a basis.

$\boldsymbol{v}_1 = \begin{pmatrix}3\\0\end{pmatrix}$, $\boldsymbol{v}_2 = \begin{pmatrix}2\\2\end{pmatrix}$

$\boldsymbol{v}_1' = \begin{pmatrix}8\\2\end{pmatrix}$, $\boldsymbol{v}_2' = \begin{pmatrix}5\\2\end{pmatrix}$

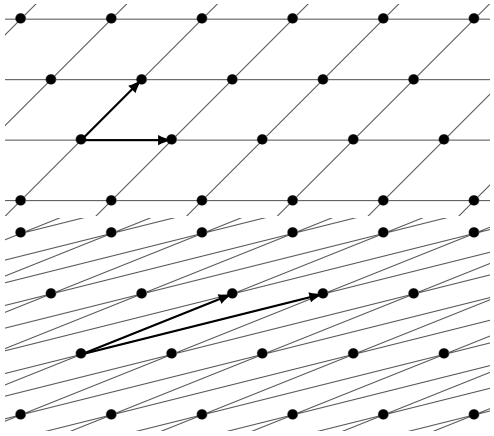


Figure 5: The same lattice, with two bases. One of them looks slightly more comfortable to traverse.

3.4 Gram Matrix

By now, we only have a bunch of linearly independent vectors. We take them to compute a *Gram matrix*.

Let $v_1, \ldots, v_n$ be vectors in $\mathbb{R}^m$. The entries of the *Gram matrix* are given by $G_{ij} = v_i \cdot v_j$. The determinant of G is called the *Gram determinant*. $\det G \neq 0 \Rightarrow v_1, \ldots, v_n$ linearly independent. $\sqrt{\det G}$ is the n-dimensional volume spanned by $v_1, \ldots, v_n$. **Example:** Let $v_1 = (2, 3), v_2 = (1, 4)$.

$$G = \begin{pmatrix}2 & 3\\1 & 4\end{pmatrix} \cdot \begin{pmatrix}2 & 1\\3 & 4\end{pmatrix} = \begin{pmatrix}13 & 14\\14 & 17\end{pmatrix}$$

$\text{vol}(v_1, v_2) = \sqrt{\det G} = \sqrt{25} = 5$

A good application is computing linear independence, as the vector set is only linearly independent when the determinant of the Gram matrix is non-zero. We need it for general lattices, as we obtain a unique vector realization. First, we are going to study an algorithm you should all know from your introductory algorithm classes: the Gram-Schmidt Algorithm. It gives us an orthogonal basis for our vector space, meaning that the vectors are mutually orthogonal.

Lemma 3.1. A basis is orthogonal if for each $\boldsymbol{v}_i \cdot \boldsymbol{v}_j = 0 \forall i \neq j$.

To add to your mathematical vocabulary, note that a *orthonormal* basis is a normalized orthogonal basis.

Theorem 3.2 (Gram-Schmidt Algorithm)**.** Let $v_1, \ldots, v_n$ be a basis for a vector space $V \subset \mathbb{R}^m$. The following algorithm creates an orthogonal basis $v_1^*, \ldots, v_n^*$ for V:

$v_1^* \leftarrow v_1$
for $i = 2..n$ **do**
  for $j = 1..i-1$
    $\mu_{i,j} \leftarrow \dfrac{v_i \cdot v_j^*}{\|v_j^*\|^2}$
  $v_i^* = v_i - \sum_{j=1}^{i-1} \mu_{i,\,j}\, v_j^*$

Given the resulting matrix $V*$, it holds that:

$$\text{Span}\{v_1, \ldots, v_i\} = \text{Span}\{v_1^*, \ldots, v_i^*\} \quad \forall i = 1, \ldots, n.$$

3.5 Volume

Leo Ducas also has a really nice intro to the LLL and BKZ algorithms: https://heat-project.eu/School/Leo%20Ducas/LLL-BKZ.pdf. Look again at Figure 5, specifically at the colored area. It is the *volume* of the lattice.

Definition 3.4 (Volume)**.** Let L be a lattice of dimension n and let F be a fundamental domain of L. Then the n-dimensional volume of F is called the *volume* of L (or sometimes the *determinant* of L).

Example: Let L be generated by the vectors

$$v_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, v_2 = \begin{pmatrix} 1/4 \\ \sqrt{2} \end{pmatrix}.$$

First, compute the Gram matrix:

$$G = \begin{pmatrix} 1 & 0 \\ \frac{1}{4} & \sqrt{2} \end{pmatrix} \cdot \begin{pmatrix} 1 & \frac{1}{4} \\ 0 & \sqrt{2} \end{pmatrix} = 2$$

Lemma 3.2. The volume is equal to the square root of the determinant. $\text{vol}(L) = \sqrt{\det G} = \sqrt{2}$

But what is the smallest volume of a lattice? It is the so-called fundamental domain, depicted as the blue-ish area in Figure 5.

Definition 3.5 (Fundamental Domain)**.** Let L be a lattice of dimension n and let $v_1, \ldots, v_n$ be a basis for L. The *fundamental domain* is the set

$$F = [0, 1)v_1 + \cdots + [0, 1)v_n.$$

Lemma 3.3. Every fundamental domain for a given lattice L has the same volume.

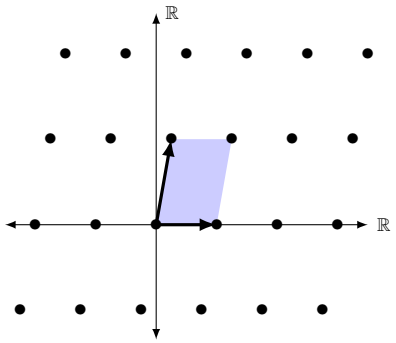


Figure 6: A lattice with the fundamental parallelepiped between the two shortest basis vectors.

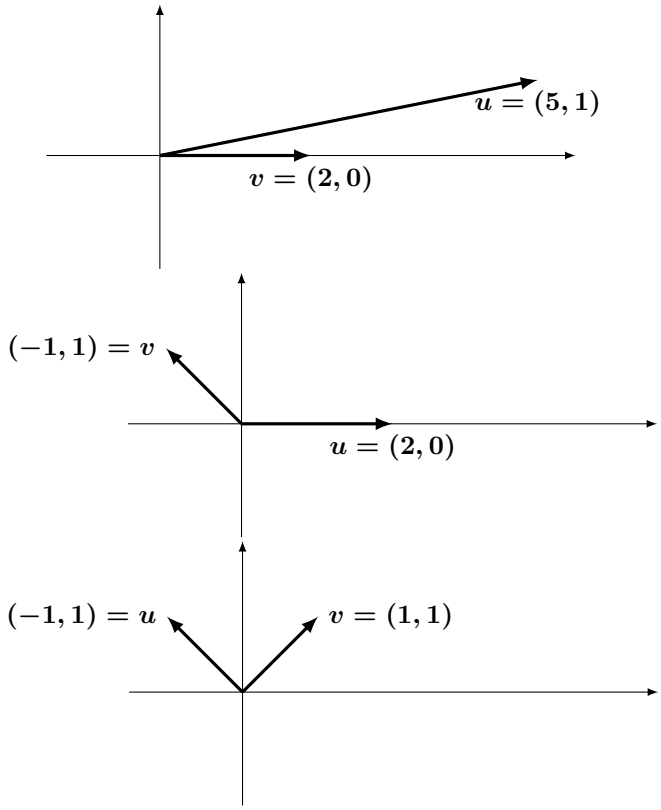
3.6 Lagrange reduction

We will now look at how to find good, meaning nearly orthogonal bases that are very close to an ideal solution.

For a basis of two vectors, there is a simple method that is similar to the Euclidian algorithm: We iteratively reduce the larger of the two vectors by adding or subtracting an integer multiple of the smaller vector. This algorithm is sometimes called the Lagrange-Gauss Algorithm. **Input:** A basis (u, v) of a 2-dimensional lattice L.
Ouput: A Lagrange-reduced basis of L.

if $\|u\| < \|v\|$ **then**
    swap u and v
while $\|v\| > \|u\|$ **do**
    $r \leftarrow u - qv$ where $q = \left\lfloor \frac{u \cdot v}{\|v\|^2} \right\rceil$
    $u \leftarrow v$
    $v \leftarrow r$
return (u, v)

Example 3.4. Input: $v = \begin{pmatrix} 2 \\ 0 \end{pmatrix}, u = \begin{pmatrix} 5 \\ 1 \end{pmatrix}$



Example Task: Solve SVP for the lattice generated by

$$v_1 = (66586820, 65354729)^T, v_2 = (6513996, 6393464)^T.$$

3.7 Lenstra-Lenstra-Lováz Algorithm (LLL)

This entire section so far has been concerned with finding a good basis. Now, we will finally learn how we can do this in the general case. If you remember the Gram-Schmidt orthogonalization, you will find this easy.

Input: A basis $(v_1, \dots v_n)$ of a lattice L.
Output: A size-reduced basis of L.
Result: Compute all the Gram-Schmidt coefficients $\mu_{i,j}$
for $i = 2..n$ **do**
    for $j = (i-1)..1$ **do**
        $v_i \leftarrow v_i - \lfloor \mu_{i,j} \rceil v_j$
        for $k = 1..j$ **do**
            $\mu_{i,k} \leftarrow \mu_{i,k} - \lfloor \mu_{i,j} \rceil \mu_{j,k}$

http://thijs.com/docs/lec1.pdf has a nice visualization of this.

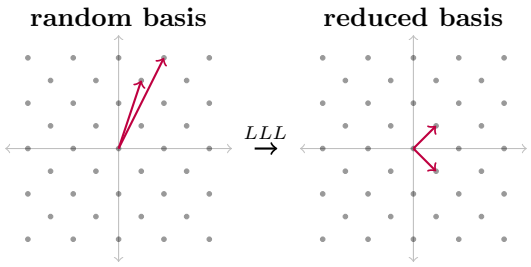


Figure 7: Two basis vectors, before and after applying the LLL algorithm for basis reduction.

3.7.1 Size-Reduction

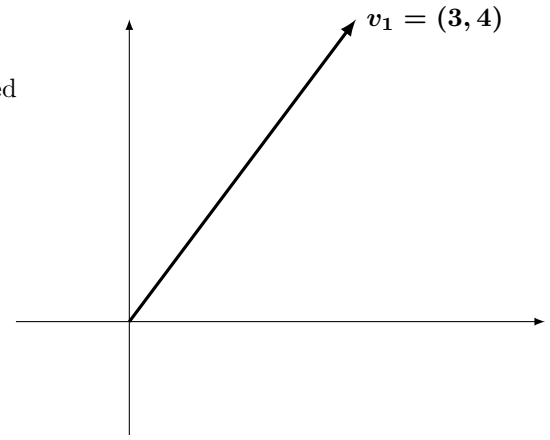
Definition 3.6 (Size-Reduced)**.** A basis $v_1, \ldots, v_n$ of a lattice is *size-reduced* if its Gram-Schmidt orthogonalization satisfies $|\mu_{i,\,j}| \leq \frac{1}{2}$.

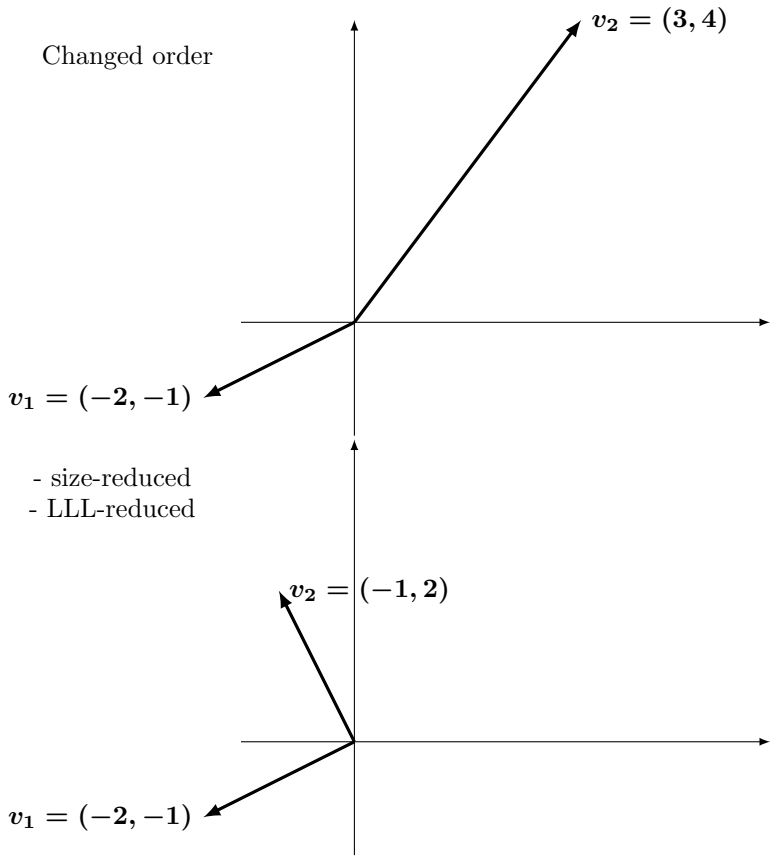
Definition 3.7 (LLL-Reduced)**.** Let $B = \{v_1, \ldots, v_n\}$ be a basis for a lattice L and denote its associated Gram-Schmidt orthogonal basis as $v_1^*, \ldots, v_n^*$. The basis is said to be *LLL-reduced* if it is size-reduced and satisfies for all $1 < i \leq n$.

$$\|v_i^*\|^2 \geq \left(\frac{3}{4} - \mu_{i,i-1}^2\right) \|v_{i-1}^*\|^2. \qquad \textit{(Lovász Condition).}$$

3.7.2 Why Lovász Condition?

- size-reduced
- not LLL-reduced





3.8 The LLL-reduced basis is a good basis

Theorem 3.3. Let L be a lattice of dimension n. Any LLL reduced basis $v_1, \ldots, v_n$ for L has the following property:

$$\prod_{i=1}^{n} \|v_i\| \leq 2^{\frac{n(n-1)}{4}} \operatorname{vol}(L).$$

In particular,

$$\|v_1\| \leq 2^{\frac{n-1}{2}} \lambda_1(L).$$

Thus an *LLL* reduced basis solves approximate SVP within a factor of $2^{(n-1)/2}$.

Theorem 3.4. Given a basis $v_1, \ldots, v_n$ of a Lattice L the LLL algorithm calculates an LLL-reduced basis in time

$$\mathcal{O}\left(n^6 \log^3 B\right), \qquad \text{where } B = \max_i \|v_i\|.$$

First, need another theorem (sorry):

Theorem 3.5 (Hermite's Theorem)**.** Every lattice L of dimension n contains a nonzero $v \in L$ satisfying

$$\|v\| \leq \sqrt{n}\,\mathrm{vol}(L)^{\frac{1}{n}}.$$

3.9 Proof sketch

It is clear that the output is LLL-reduced. So we only have to show a finite number of steps.

- L_l = lattice spanned by $v_1, \ldots v_l$.
- $d_l = \prod_{i=1}^{l} \|v_i^*\|^2$ and $D = \prod_{i=1}^{l} d_l \Rightarrow \det(L_l)^2 = d_l$.
- D changes only when swapping. More precisely, D is reduced by a factor of at least $(3/4)^N$ (argumentation with the fact that Lovász condition is violated).
- Bound D from above with Hermite's Theorem.

3.10 Minkowski Theorems, or: How long is the shortest vector?

First, we need to define how we measure *shortness*. We again use high school mathematics, this time the Euclidian distance:

$$\| (c_1, c_2, \ldots, c_n) \| = \sqrt{c_1^2 + c_2^2 + \cdots + c_n^2}$$

Theorem 3.6 (Minkowski's Theorem)**.** Let $L \subset \mathbb{R}^n$ be a lattice of dimension n. Let $S \subset \mathbb{R}^n$ be convex, closed, and symmetric. Suppose that $\mathrm{vol}(S) \geq 2^n \mathrm{vol}(L)$, then

$$S \cap L \supsetneq \{0\}.$$

4 Further Reading: Flatter-Fast Approximate Lattice Reduction

Very recently, the LLL algorithm got an update! I strongly recommend reading either the paper [RH23] or watching the presentation from CRYPTO 2023(`https://www.youtube.com/watch?v=KDnFbT6Z3xM`. It is quite easy to understand, and shows how the LLL algorithm becomes closer to the polynomial runtime.

5 Computational Problems

Disclaimer: parts of the hardness explanations were inspired by Tanja Lange's very good series on post-quantum cryptography. If you want to learn more, here is the lattice lecture link- but the other videos are also excellent. https://www.youtube.com/watch?v=UU2EaVXkKLY.

We will denote $\lambda_1(L)$ as the length of the shortest nonzero vector in the lattice L.

- **Shortest Vector Problem (SVP):** Find a shortest nonzero vector v in L, i.e. $\|v\| = \lambda_1(L)$. This is very (exponentially) slow if we want to compute an exact solution, but if an approximate solution suffices we can compute this in polynomial time. To understand this problem better, please refer to Daniele Micciancio's slides from 2020 which also have beautiful illustrations: https://simons.berkeley.edu/sites/default/files/docs/14967/sis.pdf
- **Approximate Shortest Vector Problem (SVP$_\gamma$):** TODO
- **Approximate Shortest Independent Vector Problem (SIVP$_\gamma$):** TODO
- **Closest Vector Problem (CVP):** Given a vector w, find closest vector to w in L. This is an NP-hard problem: When we have a short basis and close-to-orthogonal vectors, we can simply round.
- **Learning with Errors (LWE):**
- **Learning with Rounding (LWR):** Ajtai's cryptosystem and NTRU

Example 5.1. Given the lattice generated by v_1, v_2

$$v_1 = \begin{pmatrix} 8 \\ 2 \end{pmatrix}, v_2 = \begin{pmatrix} 5 \\ 2 \end{pmatrix}$$

and given the vector $w = (2, 3)^T$. What is the shortest nonzero vector of L? Which vector is closest to w?

$$\begin{pmatrix} -1 \\ 2 \end{pmatrix} \text{ and } \begin{pmatrix} 2 \\ 2 \end{pmatrix}$$

5.1 SIS to LWE

References

[BBL+15] Abhishek Banerjee, Hai Brenner, Gaëtan Leurent, Chris Peikert, and Alon Rosen. "SPRING: Fast Pseudorandom Functions from Rounded Ring Products". In: *FSE 2014*. Ed. by Carlos Cid and Christian Rechberger. Vol. 8540. LNCS. Springer, Heidelberg, Mar. 2015, pp. 38–57. DOI: 10.1007/978-3-662-46706-0_3 (cit. on p. 2).

[Ble98] Daniel Bleichenbacher. "Chosen Ciphertext Attacks Against Protocols Based on the RSA Encryption Standard PKCS #1". In: *CRYPTO'98*. Ed. by Hugo Krawczyk. Vol. 1462. LNCS. Springer, Heidelberg, Aug. 1998, pp. 1–12. DOI: 10.1007/BFb0055716 (cit. on p. 3).

[Bra12] Zvika Brakerski. "Fully Homomorphic Encryption without Modulus Switching from Classical GapSVP". In: *CRYPTO 2012*. Ed. by Reihaneh Safavi-Naini and Ran Canetti. Vol. 7417. LNCS. Springer, Heidelberg, Aug. 2012, pp. 868–886. DOI: 10.1007/978-3-642-32009-5_50 (cit. on p. 2).

[CDH+20] Cong Chen, Oussama Danba, Jeffrey Hoffstein, Andreas Hulsing, Joost Rijneveld, John M. Schanck, et al. *NTRU*. Tech. rep. available at https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions. National Institute of Standards and Technology, 2020 (cit. on p. 2).

[CF85] Josh D. Cohen and Michael J. Fischer. "A Robust and Verifiable Cryptographically Secure Election Scheme (Extended Abstract)". In: *26th FOCS*. IEEE Computer Society Press, Oct. 1985, pp. 372–382. DOI: 10.1109/SFCS.1985.2 (cit. on p. 3).

[DKR+20] Jan-Pieter D'Anvers, Angshuman Karmakar, Sujoy Sinha Roy, Frederik Vercauteren, Jose Maria Bermudo Mera, Michiel Van Beirendonck, et al. *SABER*. Tech. rep. available at https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions. National Institute of Standards and Technology, 2020 (cit. on p. 2).

[ElG85] Taher ElGamal. "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms". In: *IEEE Transactions on Information Theory* 31.4 (1985), pp. 469–472. DOI: 10.1109/TIT.1985.1057074 (cit. on p. 3).

[FV12] Junfeng Fan and Frederik Vercauteren. *Somewhat Practical Fully Homomorphic Encryption*. Cryptology ePrint Archive, Report 2012/144. https://eprint.iacr.org/2012/144. 2012 (cit. on p. 2).

[HHGP+10] Jeffrey Hoffstein, Nick Howgrave-Graham, Jill Pipher, and William Whyte. "Practical Lattice-Based Cryptography: NTRUEncrypt and NTRUSign". In: ed. by Phong Q. Nguyen and Brigitte Vallée. ISC. Springer, Heidelberg, 2010, pp. 349–390. ISBN: 978-3-642-02294-4. DOI: 10.1007/978-3-642-02295-1 (cit. on p. 2).

[LDK+20] Vadim Lyubashevsky, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Peter Schwabe, Gregor Seiler, et al. *CRYSTALS-DILITHIUM*. Tech. rep. available at https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions. National Institute of Standards and Technology, 2020 (cit. on p. 2).

[Pai99] Pascal Paillier. "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes". In: *EUROCRYPT'99*. Ed. by Jacques Stern. Vol. 1592. LNCS. Springer, Heidelberg, May 1999, pp. 223–238. DOI: 10.1007/3-540-48910-X_16 (cit. on p. 3).

[Pei16] Chris Peikert. "A Decade of Lattice Cryptography". In: *Found. Trends Theor. Comput. Sci.* 10.4 (2016), pp. 283–424. DOI: 10.1561/0400000074 (cit. on p. 2).

[PFH+20] Thomas Prest, Pierre-Alain Fouque, Jeffrey Hoffstein, Paul Kirchner, Vadim Lyubashevsky, Thomas Pornin, et al. *FALCON*. Tech. rep. available at https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions. National Institute of Standards and Technology, 2020 (cit. on p. 2).

[RH23] Keegan Ryan and Nadia Heninger. "Fast Practical Lattice Reduction Through Iterated Compression". In: *CRYPTO 2023, Part III*. Ed. by Helena Handschuh and Anna Lysyanskaya. Vol. 14083. LNCS. Springer, Heidelberg, Aug. 2023, pp. 3–36. DOI: 10.1007/978-3-031-38548-3_1 (cit. on p. 14).

[RSA78] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". In: *Commun. ACM* 21.2 (1978), pp. 120–126. DOI: 10.1145/359340.359342 (cit. on p. 3).

[SAB+20] Peter Schwabe, Roberto Avanzi, Joppe Bos, Léo Ducas, Eike Kiltz, Tancrède Lepoint, et al. *CRYSTALS-KYBER*. Tech. rep. available at https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions. National Institute of Standards and Technology, 2020 (cit. on p. 2).