A Study on LLL Algorithm

MSc MATHEMATICS

2024-26

A Study on LLL Algorithm

Exam Code: 62023402

Programme Code: 620

Course Code: MM 545

Certificate

This is to certify that the dissertation entitled **"A STUDY ON LLL ALGORITHM"** is a record of studies carried out by **APARNA G**, M.Sc Mathematics student of the year **2024-2026** at **T.K.M College of Arts and Science**, under my guidance and submitted to the University of Kerala in the partial fulfillment of the Degree of Master of Science in Mathematics.

Kollam

July 2026

Dr. ADERSH V K
Associate Professor
Department of Mathematics
T.K.M College of Arts and Science

Dr. ASWATHY M R
H.O.D and Assistant Professor
Department of Mathematics
T.K.M College of Arts and Science

Contents

Introduction

Mathematics has always provided powerful structures for understanding complex problems, and among these structures, lattices occupy a unique position at the intersection of pure mathematics, algorithms, and modern technology. A lattice can be viewed as a discrete arrangement of points generated by integer combinations of basis vectors in Euclidean space. Although the concept appears simple, lattices possess rich geometric properties and have become fundamental objects in areas such as computational number theory, optimization, cryptography, and algorithm design. The study of lattice bases plays an important role in understanding the structure and efficiency of lattice computations. A given lattice may have infinitely many different bases, and finding a basis with desirable properties, such as shorter and more nearly orthogonal vectors, is a central problem in lattice theory. Concepts such as Gram–Schmidt orthogonalization help analyze lattice bases by decomposing them into orthogonal components, providing a foundation for studying lattice reduction techniques. One of the most influential developments in this area is the Lenstra–Lenstra–Lovász (LLL) lattice reduction algorithm, introduced in 1982. The LLL algorithm provides an efficient method for transforming an arbitrary lattice basis into a reduced basis containing relatively short and well-structured vectors. Unlike exact solutions to difficult lattice problems such as the Shortest Vector Problem (SVP), which are computationally challenging, LLL provides an efficient approximation while maintaining strong mathematical guarantees. The importance of lattice reduction has grown significantly with the development of post-quantum cryptography. Many modern cryptographic schemes

rely on the computational difficulty of lattice problems such as the Learning With Errors (LWE) problem and the Shortest Integer Solution (SIS) problem. These problems are closely related to fundamental lattice problems, making the study of lattice algorithms essential for understanding both the strengths and limitations of lattice-based security systems.

This project explores the mathematical foundations of lattices, including their definitions, properties, geometric interpretations, and lattice reduction methods. It focuses particularly on the ideas behind the LLL algorithm, including size reduction, the Lovász condition, and the role of Gram–Schmidt orthogonalization in producing reduced lattice bases. The algorithmic aspects are also demonstrated using SageMath implementations to illustrate the practical computation of lattice reductions. The objective of this study is to develop a deeper understanding of how lattice structures can be transformed and analyzed efficiently.

Chapter 1

Preliminaries

In this chapter [2], we recall some basic definitions and theorems that will be used throughout the report.

Definition 1.1 (**Vector Space**)**.** A *vector space* V is a subset of $\mathbb{R}^m$ with the property that

$$\alpha_1 v_1 + \alpha_2 v_2 \in V \text{ for all } v_1, v_2 \in V \text{ and all } \alpha_1, \alpha_2 \in \mathbb{R}$$

Equivalently, a vector space is a subset of $\mathbb{R}^m$ that is closed under addition and under scalar multiplication by elements of $\mathbb{R}$.[4]

Definition 1.2 (**Linear Combinations**)**.** Let $v_1, v_2, \ldots, v_n \in V$. A *linear combination* of $v_1, v_2, \ldots, v_n \in V$ is any vector of the form

$$w = \alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_n v_n \quad \text{with } \alpha_1, \ldots, \alpha_n \in \mathbb{R}$$

The collection of all such linear combinations,

$$\{\alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_n v_n : \alpha_1 v_1, \alpha_2 v_2, \ldots, \alpha_n v_n \in \mathbb{R}\}$$

is called the *span* of $\{v_1, v_2, \ldots, v_n\}$

Definition 1.3 (**Independence**)**.** A set of vectors $v_1, v_2, \dots v_n \in V$ is said to be *linearly independent* if,

$$\alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_n v_n = 0 \tag{1.1}$$

then $\alpha_1 = \alpha_2 = \cdots = \alpha_n = 0$. The set is said to be *linearly dependent* if (1.1) is true with atleast one α_i nonzero.

Definition 1.4 (**Bases**)**.** A *basis* for V is a set of linearly independent vectors $v_1, v_2, \dots, v_n$ that span V. In other words, if $\mathbf{w} \in V$ can be written as

$$\mathbf{w} = \alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_n v_n$$

for a unique choice of $\alpha_1, \dots, \alpha_n \in \mathbb{R}$, then $v_1, v_2, \dots, v_n$ is a basis of V.

The number of elements in a basis is called the *dimension* of V.

Remark. Let $v_1, v_2, \dots, v_n$ be a basis for V and let $w_1, w_2, \dots, w_n$ be another set of n vectors in V. Write each of the w_j as the linear combination of the v_i,

$$\begin{aligned}
w_1 &= \alpha_{11} v_1 + \alpha_{12} v_2 + \cdots + \alpha_{1n} v_n \\
w_2 &= \alpha_{21} v_1 + \alpha_{22} v_2 + \cdots + \alpha_{2n} v_n \\
&\vdots \qquad\qquad\qquad \vdots \\
w_n &= \alpha_{n1} v_1 + \alpha_{n2} v_2 + \cdots + \alpha_{nn} v_n
\end{aligned}$$

Then $w_1, \ldots, w_n$ is also a basis for V if and only if the determinant of the matrix

$$\begin{pmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1n} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{n1} & \alpha_{n2} & \cdots & \alpha_{nn} \end{pmatrix}$$

is not equal to 0.

Definition 1.5 (**Dot Product**)**.** Let $\boldsymbol{v}$,$\boldsymbol{w} \in \mathbb{R}^n$ and write $\boldsymbol{v}$ and $\boldsymbol{w}$ as follows;

$$v = (x_1, x_2, \ldots, x_n) \text{ and } w = (y_1, y_2, \ldots, y_n).$$

The *dot product* of $\boldsymbol{v}$ and $\boldsymbol{w}$ is the quantity

$$v \cdot w = x_1y_1 + x_2y_2 + \cdots + x_ny_n$$

In terms of inner-product, we can say that $\langle v, w \rangle = x_1y_1 + x_2y_2 + \cdots + x_ny_n$.

Definition 1.6 (**Orthogonal**)**.** We say that $\boldsymbol{v}$ and $\boldsymbol{w}$ are *orthogonal* to one another if $v \cdot w = 0$.
Or in the terms of inner-product, $\langle v, w \rangle = 0$.

Definition 1.7 (**Euclidean norm**)**.** The *Euclidean norm* of $\boldsymbol{v}$ is the quantity

$$\|v\| = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$$

The relation between dot products, inner-product and norm can be expressed by the formula,

$$v \cdot v = \|v\|^2 \text{ and } \langle v, v \rangle = ||v||^2$$

Definition 1.8 (**Orthogonal Basis**)**.** An *orthogonal basis* for a vector space V

is the basis $v_1, v_2, \ldots, v_n$ with the property that

$$v_i \cdot v_j = 0 \text{ for all i} \neq \text{j}$$

The basis is *orthonormal* if in addition, $\|v_i\| = 1$ for all i.

Definition 1.9 (**Orthogonal Compliment**)**.** Let V be a vector space which is the subspace of $\mathbb{R}^n$ with the basis $B = b_1, \ldots, b_n$. The *Orthogonal compliment* of V is

$$V^{\perp} = \{x \in \mathbb{R}^n : \langle x, v\rangle = 0, \ \forall\ v \in V\}$$

Definition 1.10 (**Projection of a vector onto another vector**)**.** Let $u, v, w \in \mathbb{R}^n \setminus \{0\}$ and $\langle u, w\rangle = 0$, and suppose that v is not a scalar multiple of u.
Then it can be easily seen that $[u, w]$ is an orthogonal basis for $\mathbb{R}^n$. Now,

$$v = \mu u + \omega w, \text{ where } \mu, \omega \in \mathbb{R}$$

Then,

$$\begin{aligned}
\langle u, v\rangle &= \langle u, \mu u + \omega w\rangle \\
&= \langle u, \mu u\rangle + \langle u, \omega w\rangle \\
&= \mu\langle u, u\rangle + \omega\langle u, w\rangle \\
&= \mu\langle u, u\rangle + 0 \\
&= \mu||u||^2
\end{aligned}$$

which implies that,

$$\mu = \langle u, v\rangle / ||u||^2.$$

Also, $\langle u, v - \mu u\rangle = 0$.

Thus, projection of v onto $\langle u \rangle$ and $\langle u \rangle^{\perp}$ is,

$$\text{proj}_{\langle u \rangle}(v) = \mu u \quad \text{and} \quad \text{proj}_{\langle u \rangle^{\perp}}(v) = v - \mu u.$$

Theorem 1.1 (**Gram-Schmidt Algorithm**). *Let $v_1, \ldots v_n$ be a basis for a vector space $V \subset \mathbb{R}^n$. The following algorithm creates an orthogonal basis $v_1^*, \ldots, v_n^*$ for V:*

Set $v_1^* = v_1$.

Loop $i = 2, 3, \ldots, n$.

$\quad$ *Compute* $\mu_{ij} = \dfrac{v_i \cdot v_j^*}{\|v_j^*\|^2}$, $\quad$ *for* $1 \leq j < i$.

$\quad$ *Set* $v_i^* = v_i - \displaystyle\sum_{j=1}^{i-1} \mu_{ij} v_j^*$.

End Loop

The two bases have the property that

$$Span\{v_1, \ldots v_i\} = Span\{v_1^*, \ldots, v_i^*\} \quad for\ all\ i = 1, 2, \ldots, n.$$

Example 1.1. Consider the basis $B = [(0, 0, 2), (5, 2, 4), (-5, 1, -3)]$, for a full rank integer lattice $L \subset \mathbb{Z}^3$. The squared lengths of the basis vectors are:

$$||b_1||^2 = 4, \ ||b_2||^2 = 45, \ ||b_3||^2 = 35$$

The Gram-Schmidt orthogonalization of B is,

$B^* = [(0, 0, 2), (5, 2, 0), (-1.03, 2.59, 0)]$

The squared lengths of the Gram-Schmidt basis vectors are:

$$||b_1^*||^2 = 4, \ ||b_2^*||^2 = 29, \ ||b_3^*|| = 7.769$$

Notice that $||b_i^*||^2 \leq ||b_i||^2$, for $i = 1, 2, 3$.

The following is the SageMath code of the same:

```
def gram_schmidt(vectors):
    """
    Perform Gram-Schmidt orthogonalization.
    Input: list of vectors (over RR)
    Output: list of orthogonal vectors
    """
    orthogonal = []

    for v in vectors:
        w = v
        for u in orthogonal:
            proj = (v.dot_product(u) / u.dot_product(u)) * u
            w = w - proj
        if w.norm() > 1e-10:  # avoid zero vectors
            orthogonal.append(w)

    return orthogonal
def gram_schmidt_orthonormal(vectors):
    """
    Gram-Schmidt producing an orthonormal basis
    """
    orthonormal = []

    for v in vectors:
        w = v
```

```
        for u in orthonormal:
            proj = (v.dot_product(u)) * u  # since u is already unit
            w = w - proj
        if w.norm() > 1e-10:
            orthonormal.append(w / w.norm())

    return orthonormal

V = VectorSpace(RR, 3)

v1 = V([0, 0, 2])
v2 = V([5, 2, 4])
v3 = V([-5, 1, -3])

vectors = [v1, v2, v3]

orth = gram_schmidt(vectors)
orthonorm = gram_schmidt_orthonormal(vectors)

print("Orthogonal basis:")
for v in orth:
    print(v)

print("\nOrthonormal basis:")
for v in orthonorm:
    print(v)
```

Chapter 2

Lattices: Basic definitions and Properties

In this chapter, we will look onto Lattices, some basic definitions and properties regarding it.

Definition 2.1. A subset L of $\mathbb{R}^m$ is an *additive subgroup*, if it is closed under addition and subtraction. It is called a *discrete additive subgroup*, if there is a positive constant $\epsilon > 0$ with the following property that for every $v \in L$,

$$L \cap \{ w \in \mathbb{R}^m : ||v - w|| < \epsilon\} = \{v\}$$

That is, if we take any vector v in L and draw a solid ball of radius ϵ around v, then there are no other points of L inside the ball.

Definition 2.2 (**Lattice**)**.** Let $v_1, \ldots, v_n \in \mathbb{R}^n$ be a set of linearly independent vectors. The *lattice* L generated by $v_1, \ldots, v_n$ is the set of linear combinations of $v_1, \ldots, v_n$ with coefficients in $\mathbb{Z}$,

$$L = \{\alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_n v_n : \alpha_1, \alpha_2, \ldots, \alpha_n \in \mathbb{Z}\}$$

A subset of $\mathbb{R}^n$ is a lattice if and only if it is a discrete additive subgroup.

Definition 2.3 (**Bases of a lattice**). A *basis* for L is any set of independent vectors that generates L. Any two such sets have the same number of elements. The *dimension of* L is the number of vectors in a basis for L.
Let $L = L(B)$ be an n-dimensional lattice in $\mathbb{R}^n$ with ordered basis $B = [b_1, ..., b_n]$. The following operations transform B into another basis for the same lattice:

1. Exchange two basis vectors: $b_i \leftrightarrow b_j$.

2. Replace a basis vector by it's negation: $b_i \leftarrow -b_i$.

3. Add an integer multiple of one basis vector to another: $b_i \leftarrow b_i + cb_j$ where $c \in \mathbb{Z}$.

Each of these corresponds to an elementary column operation on the basis matrix B. The first two operations change the sign of the determinant of B, while the third operation leaves the determinant unchanged. By repeatedly applying these operations, one obtains infinitely many distinct bases for the same lattice. The following theorem characterizes all such bases;

Theorem 2.1. *Let $L = L(B)$ be an n-dimensional lattice. An $n \times n$ matrix B_2 is also a basis for L if and only if $B_2 = B_1U$, where U is an $n \times n$ matrix with integer entries and determinant ± 1. (Such a matrix U is called unimodular).*

Proof. Suppose that B_1 and B_2 are both bases of the lattice L. Since each is a basis of the vector space $\mathbb{R}^n$, there exists an invertible matrix $U \in \mathbb{R}^{n\times n}$ such that $B_2 = B_1U$. Because the columns of B_2 belongs to L, the entries in U must be integers, and hence $U \in \mathbb{Z}^{n\times n}$. Similarly, there exists an invertible matrix $V \in \mathbb{Z}^{n\times n}$ such that $B_1 = B_2V$. Substituting yields $B_1 = B_2V = (B_1U)V = B_1(UV)$. Since B_1 is invertible, it follows that $UV = I_n$. Taking determinants, we obtain $\det(UV) = \det(U)\det(V) = 1$, which implies that $\det(U) = \det(V) = 1$ or $\det(U) = \det(V) = -1$. Hence, U is unimodular and $B_2 = B_1U$.
Conversely, suppose that $B_2 = B_1U$ for some unimodular matrix U. Then $L(B_2) \subseteq$

$L(B_1)$. Recall the inverse formula $U^{-1} = adj(U)/\det(U)$. Since $adj(U)$ has integer entries and $\det(U) = \pm 1$, it follows that U^{-1} is also unimodular. Thus $B_1 = B_2U^{-1}$, which implies that $L(B_1) \subseteq L(B_2)$. Therefore, $L(B_1) = L(B_2)$. □

Theorem 2.2. *Every subspace $V \subseteq \mathbb{R}^n$ admits an orthogonal basis.*

Proof. Let $B = [b_1, ..., b_k]$ be any basis of V, and define $V_i = Span(b_1, ..., b_i)$ for $1 \leq i \leq k$. We construct vectors $b_1^*, ..., b_k^*$ inductively as follows. Set $b_1^* = b_1$ and for $2 \leq i \leq k$, let $V_{i-1}^* = Span(b_1^*, ..., b_{i-1}^*)$ and $b_i^* = \text{proj}_{(V_{i-1}^*)^\perp}(b_i)$; That is, b_i^* is the projection of b_i onto the orthogonal compliment of the subspace spanned by the previously constructed vectors $b_1^*, ..., b_{i-1}^*$. Also, define $V_k^* = Span(b_1^*, ..., b_k^*)$. To ease the notation, we will write proj_i instead of $\text{proj}_{(V_i^*)^\perp}$.

We claim that for each $1 \leq i \leq k$, $V_i^* = V_i$ and $[b_1^*, ..., b_i^*]$ is an orthogonal basis of V_i^*. This claim is generally proved by induction on i.

Clearly, $V_1^* = V_1$ since $b_1^* = b_1$ and $[b_1^*]$ is an orthogonal basis of V_1^*. Let $i \geq 2$, and suppose that $V_{i-1}^* = V_{i-1}$ and that $[b_1^*, ..., b_{i-1}^*]$ is an orthogonal basis of V_{i-1}^*. Now,

$$b_i^* = \text{proj}_{i-1}(b_i) = b_i - \text{proj}_{V_{i-1}^*}(b_i) = b_i - \text{proj}_{V_{i-1}}(b_i) \in V_i$$

and so $V_i^* \subseteq V_i$. Moreover, $b_i = b_i^* + \text{proj}_{V_{i-1}^*}(b_i) \in V_i^*$, and so $V_i \subseteq V_i^*$. Hence $V_i^* = V_i$ and so $[b_1^*, ..., b_i^*]$ is a basis of V_i^*. Finally, since $b_i^* \in (V_{i-1}^*)^\perp$, we have $\langle b_i^*, b_j^* \rangle = 0$, for all $1 \leq j \leq i-1$. Thus, $[b_1^*, ..., b_i^*]$ is an orthogonal basis of V_i^*. Finally, $B^* = [b_1^*, ..., b_k^*]$ is an orthogonal basis of $V_k^* = V_k = V$, which completes the proof of the theorem. □

Remark. Let L and L' be lattices in $\mathbb{R}^n$. Then L' is a *sublattice* of L if $L' \subseteq L$

Example 2.1. Let $n = 2$ and $B_1 = [(1, 0), (0, 1)]$. Then $L_1 = L(B_1) = \{B_1x : x \in \mathbb{Z}^2\}$

Remark. Suppose that $v_1, \ldots, v_n$ is a basis for a lattice L and suppose that $w_1, \ldots, w_n \in L$ is another collection of vectors in L. Just as we did for vector

spaces, we can write each w_j as a linear combination of the basis vectors,

$$\begin{aligned} w_1 &= \alpha_{11}v_1 + \alpha_{12}v_2 + \cdots + \alpha_{1n}v_n \\ w_2 &= \alpha_{21}v_1 + \alpha_{22}v_2 + \cdots + \alpha_{2n}v_n \\ &\vdots \qquad\qquad\qquad \vdots \\ w_n &= \alpha_{n1}v_1 + \alpha_{n2}v_2 + \cdots + \alpha_{nn}v_n \end{aligned}$$

but since now we are dealing with lattices, we know that all of the coefficients α_{ij}, are integers.

Suppose that we try to express the v_i in terms of the w_j. This involves inverting the matrix

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}$$

Note that we need the v_i to be linear combinations of the w_j using *integer* coefficients, so we need the entries of A^{-1} to have integer entries. Hence

$$1 = \det(I) = \det(AA^{-1}) = \det(A)\det(A^{-1})$$

where $\det(A)$ and $\det(A^{-1})$ are integers, so we must have $\det(A) = \pm 1$. Conversely, if $\det(A) = \pm 1$, then the theory of the adjoint matrix tells us that A^{-1} does indeed have integer entries.

Definition 2.4 (**Integer Lattice**)**.** An *integral (or integer) lattice* is a lattice all of whose vectors have integer coordinates. An integral lattice is an additive subgroup of $\mathbb{Z}^n$ for some $n \geq 1$.

Example 2.2. Consider the three dimensional lattice $L \subset \mathbb{R}^3$ generated by three vectors

$$v_1 = (2, 1, 3), \quad v_2 = (1, 2, 0), \quad v_3 = (2, -3, -5)$$

Form a matrix using v_1, v_2, v_3 as the rows of the matrix.

$$A = \begin{pmatrix} 2 & 1 & 3 \\ 1 & 2 & 0 \\ 2 & -3 & -5 \end{pmatrix}$$

We create three new vectors in L by the formulas

$$w_1 = v_1 + v_3, \quad w_2 = v_1 - v_2 + 2v_3, \quad w_3 = v_1 + 2v_2$$

This is equivalent to multiplying the matrix A on the left by the matrix

$$U = \begin{pmatrix} 1 & 0 & 1 \\ 1 & -1 & 2 \\ 1 & 2 & 0 \end{pmatrix}$$

and we find that w_1, w_2, w_3 are the rows of the matrix

$$B = UA = \begin{pmatrix} 4 & -2 & -1 \\ -2 & 1 & 1 \\ -3 & 2 & 1 \end{pmatrix}$$

The matrix U has determinant -1, so the vectors w_1, w_2, w_3 are also a basis for L. The inverse of U is

$$U^{-1} = \begin{pmatrix} 4 & -2 & -1 \\ -2 & 1 & 1 \\ -3 & 2 & 1 \end{pmatrix}$$

and the rows of U^{-1} tells us how to express the v_i as the linear combinations of

the w_j,

$$v_1 = 4w_1 - 2w_2 - w_3, \;\; v_2 = -2w_1 + w_2 + w_3, \;\; v_3 = -3w_1 + 2w_2 + w_3.$$

Remark. Let $L \subset \mathbb{R}^n$ be a lattice of dimension n. The basis of L can be written as the rows of an n-by-m matrix say A. A new basis for L maybe obtained by multiplying the matrix A on the left by an n-by-m matrix U such that U has integer entries and has determinant ± 1. The set of such matrices is called the *general linear group* (over $\mathbb{Z}$) and is denoted by $GL_n(\mathbb{Z})$. It's the group of matrices with integer entries whose inverses also have integer entries.

Remark. The only difference between a usual vector space and that of lattice is that it is generated by all linear combinations of its basis vectors using integer coefficients, unlike the usual vector spaces that uses arbitrary real coefficients.

Definition 2.5 (**Fundamental Parallelepiped**)**.** Let L be a lattice of dimension n and let $v_1, \ldots, v_n$ be a basis for L. The *fundamental domain* or also known as the *fundamental parallelepiped* for L corresponding to this basis is the set

$$\mathcal{F}(v_1, \ldots, v_n) = \{t_1 v_1 + t_2 v_2 + \cdots + t_n v_n : 0 \leq t_i < 1\}$$

Remark. $\mathcal{F}(B)$ can be used to partition $\mathbb{R}^n$ into non-overlapping regions (parallelepipeds). The corners of these parallelepipeds are the elements of the lattice $L(B)$, where B is the basis of the lattice.

Definition 2.6 (**Volume of Lattice**)**.** Let L be a lattice of dimension n and let $\mathcal{F}$ be a fundamental domain for L. Then the n-dimensional volume of $\mathcal{F}$ is called the *determinant of* L. It is denoted by $\det(L)$.

Remark. The volume of a lattice is determined by the basis of the lattice. That is, volume of the lattice is given by $|\det B|$

Proposition 1. The volume is an *invariant* of a lattice L. That is, it is independent of the basis B chosen for L.

Proof. Suppose that B and B' are two bases of L, such that $B \neq B'$. Each bases generate a fundamental parallelepiped. We know that $vol(L) = |\det(B)|$. Since B and B' generate the same lattice L,

$$B = B'U \quad ; \text{where U is uni-modular matrix}$$

Now,

$$\begin{aligned} vol(L) = |\det(B)| &= |\det(B'U)| \\ &= |\det(B')\det(U)| \\ &= |\det(B')| \end{aligned}$$

Therefore, volume is invariant of L. □

Proposition 2. Suppose that $L_1 \subseteq L_2$. Then $vol(L_1) \geq vol(L_2)$, where L_1 and L_2 are lattices.

Proof. Let L_1 and $L_2 \subset \mathbb{R}^n$ with $L_1 \subseteq L_2$. If B_1 and B_2 are the basis matrices of L_1 and L_2 respectively, we can express B_1 as the integer combination of B_2.
That is,

$$B_1 = B_2A \quad ; \text{where A is an integer matrix A} \in \mathbb{Z}^{n\times n}$$

which implies,

$$\begin{aligned} \det(B_1) &= \det(B_2)\det(A) \\ vol(L_1) &= |\det(B_1)| \\ &= |\det(B_2)||\det(A)| \\ &\geq |\det(B_2| = vol(L_2) \quad (\text{since } |\det(A)| \geq 1) \end{aligned}$$

Hence proved. □

Definition 2.7 (**Shortest Vector Problem**)**.** The shortest vector problem comprises of finding a shortest nonzero in a lattice L. That is, we will find a nonzero vector $v \in L$ that minimizes the Euclidean norm $\|v\|$.

Definition 2.8 (**Approximate-SVP problem**)**.** Given a lattice L, in the approximate-SVP problem we are bound to find a nonzero lattice vector of the length at most $\gamma.\lambda_1(L)$. Finding a solution for this is believed to be hard for small γ.

Definition 2.9 (**Closest Vector problem**)**.** Given a vector $w \in \mathbb{R}^n$ that is not in L, find a vector $v \in L$ that is closest to w. That is, find a vector $v \in L$ that maximizes the Euclidean norm $\|w - v\|$.

Remark. There maybe more than one shortest nonzero in a lattice. That is why SVP asks for "a" shortest vector and not "the" shortest vector. Same is the case with CVP.

Example 2.3. In $\mathbb{Z}^2$, the solutions for SVP are (0,$\pm$1) and ($\pm$1, 0)

Definition 2.10 (**Successive minima**)**.** A fundamental problem in a lattice-based cryptanalysis is finding a "good" basis for a lattice.
Let $L \in \mathbb{Z}^n$ be a lattice. For each $i \in [1, n]$, the i-th *successive minimum* $\lambda_i(L)$ is the smallest real number r such that L has the i linearly independent vectors the longest of which has the length r.

Remark. $\lambda_1(L)$ is the length of a shortest nonzero vector in L.

Definition 2.11 (**Gram-Schmidt and lattices**)**.** Consider the lattice L with the basis $B = \{b_1, b_2, \ldots, b_n\}$. Then after Gram-Schmidt orthogonalization the basis obtained is $B' = \{b_1^*, b_2^*, \ldots, b_n*\}$, with $b_1 = b_1^*$and $b_i^* = b_i - \sum_{j=1}^{i-1} \mu_{ij} b_j^*$ for $2 \leq i \leq n$ where $\mu_{ij} = \langle b_i, b_j \rangle / ||b_j^*||^2$.[3]

Remark. The corresponding Gram-Schmidt(GS) basis is $B^* = [b_1^*, ..., b_n^*]$, where $b_1^* = b_1$ and $b_i^* = \text{proj}_{i-1}(b_i)$ for $2 \leq$ i $\leq$ n, where $\text{proj}_{i-1}(b_i)$, the projection of b_i

onto $\langle b_1^*, ..., b_{i-1}^* \rangle^\perp$, is defined as $\text{proj}_{i-1}(b_i) = b_i - \sum_{j=i}^{i-1} \mu_{ij} b_j^*$. The Gram-Schmidt coefficients are $\mu_{ij} = \langle b_i, b_j^* \rangle / ||b_j^*||^2$. B^* is an orthogonal basis for $\mathbb{R}^n$, but in general not a basis for L.

Theorem 2.3. *Let $B = \{b_1, b_2, \ldots, b_n\}$ be a basis for the integer lattice L. Then the first successive minima, $\lambda_1(L) \geq min_i ||b_i^*||$.*

Proof. Let $v \in L \setminus \{0\}$ and $v = \sum_{i=1}^{n} x_i b_i$ where $x_i \in \mathbb{Z}$.

Let k be the largest integer such that $x_k \neq 0$, so $v = \sum_{i=1}^{k} x_i b_i$.

Now,

$$\begin{aligned}
\langle v, b_k^* \rangle &= \langle \sum_{i=1}^{k} x_i b_i, b_k^* \rangle \\
&= \langle x_1 b_1 + x_2 b_2 + \cdots + x_k b_k, b_k^* \rangle \\
&= \langle x_1 b_1, b_k^* \rangle + \langle x_2 b_2, b_k^* \rangle + \cdots + \langle x_k b_k, b_k^* \rangle \\
&= \langle x_k b_k, b_k^* \rangle \\
&= x_k \langle b_k, b_k^* \rangle
\end{aligned}$$

But, $b_k = b_k^* + \sum_{j=1}^{k-1} \mu_{kj} b_j^*$

Therefore,

$$\begin{aligned}
\langle v, b_k^* \rangle &= x_k \langle b_k^* + \sum_{j=1}^{k-1} \mu_{kj} b_j^*, b_k^* \rangle \\
&= x_k \left(\langle b_k^*, b_k^* \rangle + \langle \sum_{j=1}^{k-1} \mu_{kj} b_j^*, b_k^* \rangle \right) \\
&= x_k \langle b_k *, b_k^* \rangle \\
&= x_k ||b_k^*||^2
\end{aligned}$$

We know that, $\text{proj}_u(v) = \mu u$, where $\mu = \frac{\langle u, v \rangle}{||u||^2}$

Hence,

$$\begin{aligned}\text{proj}_{b_k^*}(v) &= \frac{\langle v, b_k^* \rangle}{||b_k^*||^2}.b_k^* \\ &= \frac{x_k||b_k^*||^2}{||b_k^*||^2}.b_k^* \\ &= x_k b_k^*\end{aligned}$$

Since the projection cannot be longer than the vector itself, $||\text{proj}_{b_k^*}(v)|| \leq ||v||$. So,

$$||v|| \geq ||x_k b_k^*|| = |x_k|.||b_k^*|| \geq ||b_k^*|| \geq min_i||b_i^*|| \quad (x_k \in \mathbb{Z},\ x_k \neq 0,\ \text{so},\ |x_k| > 1)$$

□

Theorem 2.4. *Let $B = \{b_1, b_2, \ldots, b_n\}$ be a basis for the integer lattice $L \subseteq \mathbb{Z}^n$. Then, $vol(L) = \prod_{i=1}^{n} ||b_i^*||$*

Proof. Let B and B' be the $n \times n$ matrices whose i-th columns are b_i, b_i^* respectively and Q be the $n \times n$ matrix whose i-th column is $\frac{b_i^*}{||b_i^*||}$.

Let U be the following $n \times n$ upper triangular matrix,

$$U = \begin{bmatrix} 1 & \mu_{21} & \mu_{31} & \cdots & \mu_{n1} \\ 0 & 1 & \mu_{32} & \cdots & \mu_{n2} \\ 0 & 0 & 1 & \cdots & \mu_{n3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

Let D be the $n \times n$ diagonal matrix with the diagonal entries $||b_1^*||, ||b_2^*||, \ldots, ||b_n^*||$

By the definition of Gram-Schmidt basis, $B = B^*U$

Here,

$$q_i = \frac{b_i^*}{||b_i^*||}$$

That is, q_i's are orthonormal as b_i^* is orthogonal.

From this, it can be observed that,

$$QD = ||b_i^*||.q_i = ||b_i^*||.\frac{b_i^*}{||b_i^*||} = b_i^*$$
$$= B^*$$

Therefore, $B^* = QD$

Since, B^* is an orthogonal matrix and Q is an orthogonal matrix, $QQ^T = I$ which implies, $B = QDU$.

So,

$$vol(L) = |\det(B)| = |\det(Q).\det(D).\det(U)|$$
$$= |\det(D)|$$
$$= \prod_{i=1}^{n} ||b_i^*||$$

□

2.1 Gaussian Lattice Reduction:Main Ideas

The Gaussian Lattice reduction algorithm solves SVP in the lattices of dimension 2. It helps us to find an optimal basis in a lattice of dimension 2. The underlying idea is to alternatively subtract multiples of one basis vector from the other until further improvements is not possible. So, suppose that $L \subset \mathbb{R}^2$ is a 2-dimensional lattice with basis vectors v_1 and v_2. Swapping v_1 and v_2 if necessary, we may assume that $||v_1|| \leq ||v_2||$. We now try to make v_2 smaller by subtracting a multiple of v_1. If we were allowed to subtract an arbitrary multiple of v_1, then we could replace v_2 with the vector

$$v_2^* = v_2 - \frac{v_1.v_2}{||v_1||^2}v_1$$

which is orthogonal to v_1. The vector v_2^* is the projection of v_2 onto the orthogonal compliment of v_1. Since the vector v_2^* is unlikely to be in L, in reality we are allowed to subtract only integer multiples of v_1 from v_2. So we replace v_2 with the vector

$$v_2 - mv_1 \quad \text{with } m = \left\lfloor \frac{v_1.v_2}{\|v_1\|^2} \right\rceil$$

If v_2 is still longer than v_1, then we stop. Otherwise, we swap v_1 and v_2 and repeat the process. Gauss proved that this process terminates and that the resulting basis for L is extremely good. The next proposition makes this precise.

Proposition 3 (Gaussian Lattice Reduction)**.** Let $L \subset \mathbb{R}^2$ be a 2-dimensional lattice with basis vectors v_1 and v_2. The following algorithm terminates and yields a good basis for L.

Loop

If $||v_2|| < ||v_1||$, swap v_1 and v_2.

Compute $m = \lfloor v_1.v_2/||v_1||^2 \rceil$.

If $m = 0$, return the basis vectors v_1 and v_2.

Replace v_2 with $v_2 - mv_1$.

Continue Loop

More precisely, when the algorithm terminates, the vector v_1 is a shortest nonzero vector in L, so the algorithm solves SVP.

Proof. We will prove that v_1 is the shortest nonzero lattice vector. So we suppose that the algorithm has terminated and returned the vectors v_1 and v_2. This means that $||v_2|| \geq ||v_1||$ and since the algorithm terminates when $m = 0$ and $m = \lfloor v_1.v_2/||v_1||^2 \rceil$,

$$\frac{|v_1.v_2|}{||v_1||^2} \leq \frac{1}{2}$$

[Geometrically, it says that we cannot make v_2 shorter by subtracting an integral

multiple of v_1 from v_2.]

Now, let $v \in L$ is any nonzero vector. Therefore,

$$v = \alpha_1 v_1 + \alpha_2 v_2$$

which implies,

$$\begin{aligned}
||v||^2 &= ||\alpha_1 v_1 + \alpha_2 v_2||^2 \\
&= ||\alpha_1 v_1||^2 + 2\alpha_1\alpha_2||v_1.v_2|| + ||\alpha_2 v_2||^2 \\
&\geq \alpha_1^2||v_1||^2 - 2|\alpha_1\alpha_2||v_1.v_2| + \alpha_2^2||v_2||^2 \\
&\geq \alpha_1^2||v_1||^2 - |\alpha_1\alpha_2|||v_1||^2 + \alpha_2^2||v_2||^2 \\
&\geq \alpha_1^2||v_1||^2 - |\alpha_1\alpha_2|||v_1||^2 + \alpha_2^2||v_1||^2 \\
&= (\alpha_1^2 - |\alpha_1||\alpha_2| + \alpha_2^2)||v_1||^2
\end{aligned}$$

For any real numbers, say t_1 and t_2, the quantity,

$$t_1^2 - t_1 t_2 + t_2^2 = (t_1 - \frac{1}{2}t_2)^2 + \frac{3}{4}t_2^2 = \frac{3}{4}t_1^2 + (\frac{1}{2}t_1 - t_2)^2$$

This is not zero unless $t_1 = t_2 = 0$. So since α_1 and α_2 are integers and not both zero tells us that $||v||^2 \geq ||v_1||^2$, which implies that v_1 is the smallest nonzero vector in L.

Hence the proof. □

Chapter 3

LLL Algorithm: Main ideas

3.1 Public-key cryptographic schemes

The public-key cryptographic schemes that are widely deployed in practice are **RSA** and **Elliptic curve cryptosystems(ECC)**. In 1994, Peter Shor discovered a very fast quantum algorithm that can break all **RSA** and **elliptic curve cryptosystems**.[1]
In August 2024, the U.S. government's National Institute of Standards and Technology(NIST), published standards for quantum-safe lattice-based schemes:

- **Kyber**: A key encapsulation mechanism(KEM).
- **Dilithium**: A digital signature scheme.

Kyber and **Dilithium** are already deployed in large-scale applications. The security of **Kyber** and **Dilithium** is based on the hardness of solving two problems:

- Learning With Errors (LWE) problem.
- Shortest Integer Solutions (SIS) problem.

Both **LWE** and **SIS** can be reformulated as instances of the *Shortest Vector Problem* and it's approximate version. The fastest methods known for solving **SVP** are refinements of the **Lenstra-Lenstra-Lovasz (LLL) algorithm**.

3.2 LLL Algorithm: Main ideas

The *LLL Algorithm* (also known as L^3) is a polynomial-time algorithm that finds a relatively short basis for lattice. It was discovered by Arjen Lenstra, Hendrik Lenstra, and Laszlo Lovasz in 1982. The first major application of the LLL was a polynomial-time algorithm for factoring polynomials with rational coefficients. Since then LLL has found numerous applications in computational number theory and cryptography. It's most prominent application is solving the Shortest Vector Problem (SVP), which is central for assessing the security of lattice based cryptosystems.

Let L be an n-dimensional integer lattice. As with Gauss's reduction in two dimensions, it is not difficult to find a lattice basis $B = [b_1, b_2, ..., b_n]$ so that the Gram-Schmidt (GS) coefficients μ_{ij} are small, specifically $|\mu_{ij}| \leq \frac{1}{2}$ for $1 \leq j < i \leq n$. This condition, is known as *size reduction*, ensures that the basis vectors are reasonably close to being orthogonal. However, it does not guarantee that the basis vectors themselves are short.

Consider a randomly chosen lattice basis B. One expects that the GS vectors $b_1^*, ..., b_n^*$ rapidly decrease in length. This is because each vector b_i^* is obtained by projecting b_i onto the orthogonal compliment of the subspace spanned by the earlier basis vectors $b_1, ..., b_{i-1}$. As i increases, this orthogonal compliment has smaller dimension, and consequently the projection of b_i typically tends to be shorter.

A key invariant of lattices comes into play here. Although the individual GS vectors depend on the chosen lattice basis, the product of their lengths doesn't, it is equal to the volume of the lattice. The central idea behind the LLL algorithm is to exploit this fact by reshaping the basis so that the lengths of the GS vectors are distributed more evenly, rather than dropping off too quickly.

The mechanism that makes this possible is a sequence of local operations: *size reduction* and *swaps* of adjacent basis vectors. The swaps are designed to slow down the rate at which the GS lengths decrease, thereby forcing earlier vectors to

become relatively short. In particular, the first vector b_1^* is expected to be relatively short, and thus so will the first lattice basis vector b_1, since $b_1 = b_1^*$. More precisely, we'll prove that the first vector in the lattice basis produced by the LLL algorithm is guaranteed to have length no more than $2^{(n-1)/2}$ times the length of a shortest nonzero lattice vector.[5]

3.3 LLL-reduced bases and Lovasz condition

Definition 3.1. A lattice basis $[b_1, ..., b_n]$ is called *size reduction* if $|\mu_{ij}| \leq \frac{1}{2}$ for all $1 \leq j < i \leq n$. A size reduced basis is said to be *LLL-reduced* if it also satisfies the *Lovasz condition*,

$$||b_k^*||^2 \geq (\frac{3}{4} - \mu_{k,k-1}^2)||b_{k-1}^*||^2 \quad \text{for all} \quad 2 \leq k \leq n.$$

Since $0 \leq \mu_{k,k-1}^2 \leq \frac{1}{4}$, the Lovasz condition implies that $||b_k^*||^2 \geq \frac{1}{2}||b_{k-1}^*||^2$. In other words, the GS basis vectors are not allowed to shrink too quickly as we move along the basis.

An equivalent way to express the Lovasz condition is,

$$||b_k^* + \mu_{k,k-1}b_{k-1}^*||^2 \geq \frac{3}{4}||b_{k-1}^*||^2$$

Theorem 3.1. *Let $[b_1, ..., b_n]$ be an LLL-reduced basis of a lattice L. Then*

$$||b_1|| \leq 2^{(n-1)/2}\lambda_1(L)$$

Proof. For each $2 \leq i \leq n$ we have,

$$\begin{aligned}
||b_i^*||^2 &\geq (\frac{3}{4} - \mu_{i,i-1}^2)||b_{i-1}^*|| \\
&\geq ||(\frac{3}{4} - \frac{1}{4})||b_{i-1}^*||^2 \quad (\text{Since } |\mu_{i,i-1}| \leq \frac{1}{2} \text{ implies that, } \mu_{i,i-1}^2 \leq \frac{1}{4}) \\
&= \frac{1}{2}||b_{i-1}^*||^2
\end{aligned}$$

therefore, $||b_i^*||^2 \geq \frac{1}{2}||b_{i-1}^*||^2$ implies that, $||b_{i-1}^*||^2 \leq 2||b_i^*||^2$

For $i = 2$,

$$||b_1^*||^2 \leq 2||b_2^*||^2 \tag{3.1}$$

For $i = 3$,

$$||b_2^*||^2 \leq 2||b_3^*||^2$$

Therefore, (3.1) implies that,

$$||b_1^*||^2 \leq 2(2||b_3^*||^2) \leq 2^2||b_3^*||^2 \tag{3.2}$$

For $i = 4$,

$$||b_3^*||^2 \leq 2||b_4^*||^2$$

Therefore, (3.2) implies that,

$$||b_1^*||^2 \leq 2^2(2||b_4^*||^2) \leq 2^3||b_4^*||^2$$

Generally,

$$||b_1^*|| \leq 2^{i-1}||b_i||^2$$

Since $b_1 = b_1^*$, which implies that,

$$||b_1||^2 = ||b_1^*||^2 \leq 2^{i-1}||b_i||^2$$

So, taking minimum over all i;

$$||b_1||^2 \leq 2^{n-1} min_i ||b_i^*||^2 \leq 2^{n-1}\lambda_1(L)^2$$

which implies that,

$$||b_1|| \leq 2^{(n-1)/2}\lambda_1(L)$$

□

3.4 Size reduction

The given Algorithm is a simple procedure for size reducing a lattice basis. The size reduction is performed in steps 4-7. In step 7, b_i is size reduced with respect to b_j.

Algorithm 1: Size reduction

1 **Input:** Lattice basis $B = [b_1, \ldots, b_n]$

2 **Output:** Size-reduced basis $\tilde{B} = [\tilde{b}_1, \ldots, \tilde{b}_n]$

3 Compute the Gram–Schmidt orthogonalization $B^* = [b_1^*, \ldots, b_n^*]$.

4 **for** $i = 2$ **to** n **do**

5 **for** $j = i - 1$ **to** 1 **do**

6 Compute $\mu_{ij} = \langle b_i, b_j^* \rangle / \|b_j^*\|^2$.

7 Set $b_i \leftarrow b_i - \lfloor \mu_{ij} \rceil \, b_j$; `// size-reduce` b_i `with respect to` b_j

8 **return** $(\tilde{B} = [b_1, \ldots, b_n])$

Theorem 3.2. *Size reduction works. That is, the basis* $\tilde{B} = [\tilde{b_1}, ..., \tilde{b_n}]$ *produced by the above mentioned algorithm is a size-reduced basis of* $L(B)$*. Moreover, size-reduction leaves the Gram-Schmidt basis unchanged.*

Proof. Claim 1: $\tilde{B}$ is a basis for $L(B)$.

We know,

$$B = [b_1, ..., b_n]$$

$$\tilde{B} = [\tilde{b_1}, ..., \tilde{b_n}]$$

So,

$$L(B) = \{\alpha_1 b_1 + ... + \alpha_n b_n \ ; \alpha_i \in \mathbb{Z}\}$$

$$L(\tilde{B}) = \{\beta_1 \tilde{b_1} + ... + \beta_n \tilde{b_n} \ ; \beta_i \in \mathbb{Z}\}$$

Also, $\tilde{b_i} = b_i - \sum_{j=1}^{i-1} \lfloor \mu_{ij} \rceil b_j \in L(B)$, since $b_i, b_j \in B$. So, $\tilde{b_i} \in L(B)$ implies that, $L(\tilde{B}) \subseteq L(B)$. Now,

$$b_i = \tilde{b_i} + \sum_{j=1}^{i-1} \lfloor \mu_{ij} \rceil b_j$$

which implies $b_i \in L(\tilde{B})$. Therefore, $L(B) \subseteq L(\tilde{B})$. So, lattice does not change under change of basis. Hence, $\tilde{B}$ is a basis of L.

Claim 2: The Gram-Schmidt basis of $\tilde{B}$ is identical to that of B.

We need to prove that $\tilde{b_i^*} = b_i^*$, for all $1 \leq i \leq n$

Let $V_i = Span(b_1, b_2, ..., b_i)$ and $\tilde{V_i} = Span(\tilde{b_1}, ...\tilde{b_i})$

We know,

$$\tilde{b_i} = b_i - \sum_{j=1}^{i-1} \lfloor \mu_{ij} \rceil b_j, \text{ for all } 1 \leq j < i < n$$

which implies that, $\tilde{b_i} = b_i - \text{Span}\{b_1, ..., b_{i-1}\}$. That is, $\tilde{b_i} = \text{Span}\{b_1, ..., b_i\}$. Hence $\tilde{b_i} \in V_i$ and $\tilde{V_i} \subseteq V_i$.

Now, it is known that,

$$\tilde{b_i} = b_i - \sum_{j=1}^{i-1} \lfloor \mu_{ij} \rceil b_j, \text{which implies that } b_i = \tilde{b_i} + \sum_{j=1}^{i-1} \lfloor \mu_{ij} \rceil b_j$$

Clearly $b_1 = \tilde{b}_1 \in \tilde{V}_1$. Assume by induction that $b_j \in \tilde{V}_j$ for $j \leq i-1$. Then So,

$$b_i = \tilde{b_i} + \sum_{j=1}^{i-1} \lfloor \mu_{ij} \rceil b_j \in \tilde{V}_i$$

Therefore, $V_i \subseteq \tilde{V}_i$. That is, $V_i = \tilde{V}_i$, for $i = 1, 2, \ldots, n$.

Thus,

$$\begin{aligned}
\tilde{b_i^*} &= \text{proj}_{(\tilde{V}_{i-1})^\perp}(\tilde{b_i}) \\
&= \text{proj}_{(V_{i-1})^\perp}(\tilde{b_i}) \\
&= \text{proj}_{(V_{i-1})^\perp}(b_i + c_i) \text{ for some } c_i \in V_{i-1} \\
&= b_i^*
\end{aligned}$$

This proves the claim.

Claim 3: $\tilde{B} = [\tilde{b_1}, ..., \tilde{b_n}]$ is size reduced.

We need to prove that $|\tilde{\mu}_{ij}| \leq \frac{1}{2}$, for all $1 \leq j < i \leq n$.

We will show this for $i = 2$ and $i = 3$ and general case follows by induction. Now,

$$\tilde{b_i} = b_i - \sum_{j=1}^{i-1} \lfloor \mu_{ij} \rceil b_j, \text{ for all } 1 \leq j < i \leq n$$

So, $\tilde{b_1} = b_1$. Now for $i = 2$ and $j = 1$;

$$\tilde{b_2} = b_2 - \lfloor \mu_{ij} \rceil b_1, \text{ where } \mu_{21} = \frac{\langle b_2, b_1^* \rangle}{||b_1^*||^2}$$

The new Gram-Schmidt coefficient,

$$\begin{aligned}
\tilde{\mu}_{21} &= \frac{\langle \tilde{b_2}, b_1^* \rangle}{||b_1^*||^2} \\
&= \frac{\langle b_2 - \lfloor \mu_{21} \rceil b_1, b_1^* \rangle}{||b_1^*||^2} \\
&= \frac{\langle b_2, b_1^* \rangle - \langle \lfloor \mu_{21} \rceil b_1, b_1^* \rangle}{||b_1^*||^2} \\
&= \frac{\langle b_2, b_1^* \rangle - \lfloor \mu_{21} \rceil \langle b_1, b_1^* \rangle}{||b_1^*||^2} \\
&= \frac{\langle b_2, b_1^* \rangle - \lfloor \mu_{21} \rceil \langle b_1^*, b_1^* \rangle}{||b_1^*||^2} \\
&= \frac{\langle b_2, b_1^* \rangle - \lfloor \mu_{21} \rceil ||b_1^*||^2}{||b_1^*||^2} \\
&= \frac{\langle b_2, b_1^* \rangle}{||b_1^*||^2} - \lfloor \mu_{21} \rceil \\
&= \mu_{21} - \lfloor \mu_{21} \rceil
\end{aligned}$$

Hence,

$$|\tilde{\mu}_{21}| \leq \frac{1}{2}$$

Now, for $i = 3$ and $j = 2, 1$

$$\begin{aligned}
\tilde{b}_3 &= b_3 - \lfloor \mu_{32} \rceil b_2, \text{ where } \mu_{32} = \frac{\langle b_3, b_2^* \rangle}{||b_2^*||^2} \\
\tilde{b}_3 &= b_3 - \lfloor \mu_{31} \rceil b_1, \text{ where } \mu_{31} = \frac{\langle b_3, b_1^* \rangle}{||b_1^*||^2}
\end{aligned}$$

First we will find the value of $\tilde{\mu}_{32}$,

$$\begin{aligned}
\tilde{\mu}_{32} &= \frac{\langle \tilde{b_3}, b_2^* \rangle}{||b_2^*||^2} = \frac{\langle b_3 - \lfloor \mu_{32} \rceil b_2, b_2^* \rangle}{||b_2^*||^2} \\
&= \frac{\langle b_3, b_2^* \rangle - \langle \lfloor \mu_{32} \rceil b_2, b_2^* \rangle}{||b_2^*||^2} = \frac{\langle b_3, b_2^* \rangle - \lfloor \mu_{32} \rceil \langle b_2, b_2^* \rangle}{||b_2^*||^2} \\
&= \frac{\langle b_3, b_2^* \rangle - \lfloor \mu_{32} \rceil \langle b_2^*, b_2^* \rangle}{||b_2^*||^2} = \frac{\langle b_3, b_2^* \rangle - \lfloor \mu_{32} \rceil ||b_2^*||^2}{||b_2^*||^2} \\
&= \frac{\langle b_3, b_2^* \rangle}{||b_2^*||^2} - \lfloor \mu_{32} \rceil = \mu_{32} - \lfloor \mu_{32} \rceil
\end{aligned}$$

So we can conclude that

$$|\tilde{\mu}_{32}| \leq \frac{1}{2}$$

Now, lastly

$$\begin{aligned}
\tilde{\mu}_{31} &= \frac{\langle \tilde{b_3}, b_1^* \rangle}{||b_1^*||^2} \\
&= \frac{\langle b_3 - \lfloor \mu_{31} \rceil b_1, b_1^* \rangle}{||b_1^*||^2} \\
&= \frac{\langle b_3, b_1^* \rangle - \langle \lfloor \mu_{31} \rceil b_1, b_1^* \rangle}{||b_1^*||^2} \\
&= \frac{\langle b_3, b_1^* \rangle - \lfloor \mu_{31} \rceil \langle b_1, b_1^* \rangle}{||b_1^*||^2} \\
&= \frac{\langle b_3, b_1^* \rangle - \lfloor \mu_{31} \rceil \langle b_1^*, b_1^* \rangle}{||b_1^*||^2} \\
&= \frac{\langle b_3, b_1^* \rangle - \lfloor \mu_{31} \rceil ||b_1^*||^2}{||b_1^*||^2} \\
&= \frac{\langle b_3, b_1^* \rangle}{||b_1^*||^2} - \lfloor \mu_{31} \rceil \\
&= \mu_{31} - \lfloor \mu_{31} \rceil
\end{aligned}$$

Hence, in this case too,

$$|\tilde{\mu}_{31}| \leq \frac{1}{2}$$

The general case follows similarly and hence $|\tilde{\mu}_{ij}| \leq \frac{1}{2},$ for all $1 \leq j < i \leq n$

Therefore, the theorem is proved. □

3.5 Algorithm Description

The LLL lattice basis reduction algorithm begins by size reduction of the input basis. It then swaps two adjacent basis vectors b_{j-1} and b_j for which the Lovasz condition is violated. The purpose of the swap is to reduce the rate at which the Gram-Schmidt vectors decrease in length. If multiple violations occur, the algorithm is free to choose any one of them; different choices may lead to different LLL-reduced bases, but all offer the same guarantees. After the swap, the new

basis is size reduced. The process of swapping a pair of adjacent basis vectors and size reducing the new basis is repeated until all adjacent pairs satisfy the Lovasz condition, at which point the basis is LLL-reduced.

Given below is the described LLL Algorithm :

Algorithm 2: LLL lattice basis reduction algorithm

1 **Input:** Lattice basis $B = [b_1, \ldots, b_n]$

2 **Output:** LLL-reduced basis $[b_1, \ldots, b_n]$

3 Size-reduce $[b_1, \ldots, b_n]$ using Algorithm 1.

4 **if** *there exists an index* $j \in [2, n]$ *such that*

$$\|b_j^*\|^2 < \left(\frac{3}{4} - \mu_{j,j-1}^2\right) \|b_{j-1}^*\|^2,$$

then

5 Swap b_{j-1} and b_j.

6 Go to step 1.

7 **return** $([b_1, \ldots, b_n])$

The following is the SageMath code of the same :

```
# The rows of matrix B are the basis vectors for an integer lattice L
B = Matrix(ZZ, [[],[],[],[]])


print(det(B))              # Determine the volume of L


C,mu = B.gram_schmidt()    # Determine the Gram-Schmidt basis and coefficients
print(C)
print(mu)


D = B.LLL(0.75)            # Determine a (3/4)-LLL-reduced basis for L
```

```
print(D)


# Determine a shortest nonzero vector in L
from sage.modules.free_module_integer import IntegerLattice
L = IntegerLattice(B)
v = L.shortest_vector()
print(v)
```

Example 3.1. The LLL reduction of the basis,

$B = [(7, 8, -6, -6, -4), (2, 2, -1, -1, -1), (-2, 1, 0, -2, -4), (-5, -10, 7, 7, 6), (0, -3, 0, -3, -3)]$

is as follows:

step	k	swap	basis after swap
			$\begin{pmatrix} 2 & 2 & -1 & -1 & -1 \\ 7 & 8 & -6 & -6 & -4 \\ -2 & 1 & 0 & -2 & -4 \\ -5 & -10 & 7 & 7 & 6 \\ 0 & -3 & 0 & -3 & -3 \end{pmatrix}$
2	1	swap b_1 and b_0	$\begin{pmatrix} 2 & 2 & -1 & -1 & -1 \\ -1 & 0 & -2 & -2 & 0 \\ 0 & -1 & 1 & -1 & -2 \\ -1 & 1 & 2 & 0 & -4 \\ 0 & -3 & 0 & -3 & -3 \end{pmatrix}$
11	3	swap b_3 and b_2	$\begin{pmatrix} 2 & 2 & -1 & -1 & -1 \\ -1 & 0 & -2 & -2 & 0 \\ 0 & -1 & 1 & -1 & -2 \\ 1 & -1 & 0 & 1 & 1 \\ -1 & 2 & 1 & 1 & -2 \end{pmatrix}$
23	4	swap b_4 and b_3	

27	3	swap b_3 and b_2	$\begin{pmatrix} 2 & 2 & -1 & -1 & -1 \\ -1 & 0 & -2 & -2 & 0 \\ 1 & -1 & 0 & 1 & 1 \\ 0 & -1 & 1 & -1 & -2 \\ -1 & 2 & 1 & 1 & -2 \end{pmatrix}$
30	2	swap b_2 and b_1	$\begin{pmatrix} 2 & 2 & -1 & -1 & -1 \\ 1 & -1 & 0 & 1 & 1 \\ -1 & 0 & -2 & -2 & 0 \\ 0 & -1 & 1 & -1 & -2 \\ -1 & 2 & 1 & 1 & -2 \end{pmatrix}$
32	1	swap b_1 and b_0	$\begin{pmatrix} 1 & -1 & 0 & 1 & 1 \\ 2 & 2 & -1 & -1 & -1 \\ -1 & 0 & -2 & -2 & 0 \\ 0 & -1 & 1 & -1 & -2 \\ -1 & 2 & 1 & 1 & -2 \end{pmatrix}$
37	2	swap b_2 and b_1	$\begin{pmatrix} 1 & -1 & 0 & 1 & 1 \\ 0 & -1 & -2 & -1 & 1 \\ 2 & 2 & -1 & -1 & -1 \\ 0 & -1 & 1 & -1 & -2 \\ -1 & 2 & 1 & 1 & -2 \end{pmatrix}$
46	3	swap b_3 and b_2	$\begin{pmatrix} 1 & -1 & 0 & 1 & 1 \\ 0 & -1 & -2 & -1 & 1 \\ 0 & -1 & 1 & -1 & -2 \\ 2 & 2 & -1 & -1 & -1 \\ -1 & 2 & 1 & 1 & -2 \end{pmatrix}$

58	4	swap b_4 and b_3	$\begin{pmatrix} 1 & -1 & 0 & 1 & 1 \\ 0 & -1 & -2 & -1 & 1 \\ 0 & -1 & 1 & -1 & -2 \\ 0 & 0 & -1 & 1 & 0 \\ 2 & 2 & -1 & -1 & -1 \end{pmatrix}$
62	3	swap b_3 and b_2	$\begin{pmatrix} 1 & -1 & 0 & 1 & 1 \\ 0 & -1 & -2 & -1 & 1 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & -1 & 1 & -1 & -2 \\ 2 & 2 & -1 & -1 & -1 \end{pmatrix}$
65	2	swap b_2 and b_1	$\begin{pmatrix} 1 & -1 & 0 & 1 & 1 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & -1 & -2 & -1 & 1 \\ 0 & -1 & 1 & -1 & -2 \\ 2 & 2 & -1 & -1 & -1 \end{pmatrix}$
67	1	swap b_1 and b_0	$\begin{pmatrix} 0 & 0 & -1 & 1 & 0 \\ 1 & -1 & 0 & 1 & 1 \\ 0 & -1 & -2 & -1 & 1 \\ 0 & -1 & 1 & -1 & -2 \\ 2 & 2 & -1 & -1 & -1 \end{pmatrix}$
76	3	swap b_3 and b_2	$\begin{pmatrix} 0 & 0 & -1 & 1 & 0 \\ 1 & -1 & 0 & 1 & 1 \\ 0 & -1 & 0 & 0 & -2 \\ 0 & -1 & -2 & -1 & 1 \\ 2 & 2 & -1 & -1 & -1 \end{pmatrix}$

Conclusion

This project presented a systematic study of lattice theory and the Lenstra–Lenstra –Lovász (LLL) lattice reduction algorithm, emphasizing both their mathematical foundations and computational significance. Beginning with the basic concepts of vector spaces, lattices, lattice bases, and Gram–Schmidt orthogonalization, the study established the theoretical framework required to understand lattice reduction. The properties of lattice volume, successive minima, and fundamental computational problems such as the Shortest Vector Problem (SVP) and Closest Vector Problem (CVP) were discussed to highlight their importance in computational mathematics. The principles of Gaussian reduction and the LLL algorithm, including size reduction and the Lovász condition, were examined to demonstrate how arbitrary lattice bases can be transformed into reduced bases containing shorter and nearly orthogonal vectors. SageMath implementations further illustrated the practical application of these concepts. The project also emphasized the growing role of lattice reduction in post-quantum cryptography, where the hardness of lattice problems forms the basis of secure cryptographic schemes. Overall, this study provides a strong foundation for advanced research in lattice algorithms, computational number theory, and lattice-based cryptography.

References

[1] Daniel J. Bernstein, Johannes Buchmann, and Erik Dahmen, eds. *Post-Quantum Cryptography*. Springer. Berlin, Heidelberg: Springer, 2009.

[2] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. *An Introduction to Mathematical Cryptography*. 2nd ed. Undergraduate Texts in Mathematics. New York: Springer, 2014.

[3] Alfred Menezes. *A Gentle Introduction to Lattice-Based Cryptography*. Revised version, March 25, 2026. Mar. 2026. URL: `https://cryptography101.ca/wp-content/uploads/lattice-based-cryptography.pdf`.

[4] M. Thamban Nair and Arindama Singh. *Linear Algebra*. Singapore: Springer, 2018.

[5] Phong Q. Nguyen and Jacques Stern. "The Two Faces of Lattices in Cryptology". In: *Cryptography and Lattices*. Ed. by Joseph H. Silverman. Vol. 2146. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2001, pp. 146–180.