# Topics in Algebraic Computation

## Table of contents

# 1 Introduction

Notes for Lectures in the course "Topics in Algebraic Computation" given by Prof.Palash Sarkar, ASU, Indian Statistical Institute. This course has been offered as a part of M.Tech Computer Science course curriculum.

## 1.1 Overview

Course primarily concentrates on algorithmic approach to compute algebraic objects. Focus of the course will be on Algebraic objects that are built using Integers, Polynomials and Matrices. Algorithms for computing the following will be discussed as part of the course:

i. Addition, Multiplication, Inversion, GCD for Integers.

ii. Strassen's algorithm for Matrix Multiplication.

iii. Solving systems of Linear Equations.

iv. Computing Characteristic polynomial, Minimal Polynomail, Inverse Polynomial.

v. Conversion of Matrices into standard canonical forms like Hermite Canonical Form (HCF).

vi. Berlekamp's Algorithm for Polynomial factorization

Asymptotic computational complexity analysis and correctness proofs for these algorithms will also be covered. Beyond this, some more advanced topics will be discussed :

i. Lower bounds on complexity of Algebraic operations.

ii. Grobner Basis algorithm.

iii. Lattice reduction algorithm via $L^3$.

iv. Function Field Sieve, Number Field Sieve algorithm for factoring integers and finding discrete log.

## 1.2 References

Except for few topics course most content of the course is taken from Research publications. For standard topics, following text books are suggested:

i. Aho, Hopcroft, Ullman : Design and Analysis of Algorithms.

ii. Grobner Basis from Ideals, Varieties and Algorithms by Cox, Little, O'shea

iii. Lattice Reduction from Algorithmic aspects of Algebraic Number Theory by Cohen.

iv. Computational Algebra. by Abijit Das

## 1.3 Evaluation

Course has 3 evaluation components. Components and their weightage in the evaluation are as follows:

i. Mid Semester Examination - 30%

ii. Termpaper Assignment - 20%

iii. End Semester Examination - 50%

# 2 Hermite Canonical Form

We shall consider matrices with elements from a Field. Today's discussion will focus on key results and algorithms for conversion of matrices in to Hermite Normal Form (HNF), computing inverse of a square matrix, testing the consistency of linear equations and LUB decomposition of matrices. These forms of matrix help in computing matrix parameters like rank, basis of row and column, basis of null space.

We shall state the following result without proof. Subsequently we progress towards the construction.

**Theorem 1. (Rank Factorization Theorem)** *If $A$ is an $m\times n$ matrix with rank $r$. Then there exist two full-rank matrices $B$ and $C$ with orders $m\times r$ and $r\times n$ such that $A=B\times C$.*

**Notation 1.** *Linear space spanned by columns of a matrix $X$ is denoted by* $\mathrm{Col.Sp}(X)$ *and Linear space spanned by rows of a matrix $X$ is denoted by* $\mathrm{Row.Sp}(X)$.

**Notation 2. (Augumented Matrix)** *If $A, B$ are two matrices of orders $m\times k$ and $n\times k$, then the matrix $[A\colon B]$ of order $(m+n)\times k$ obtained by extending rows of $A$ with corresponding rows of $B$.*

**Note 1.** System of linear equations : $Ax=b$ is consistent if $b\in\mathrm{Col.Sp}(A)$. Putting this in another way $\mathrm{Rank}(A)=(\mathrm{Rank}([A\colon b]))$.

**Definition 1. (Null Space)** *If $A$ is an $n\times n$ matrix in $\mathbb{F}$ then space of vectors $v\in\mathbb{F}^n$ such that $Av=0$ is called Null Space.*

**Note 2.** For a given matrix $A$, $Ax=0$ has a unique solution $\Leftrightarrow$ $A$ has a full rank. More generally $A_{m\times n}x=b$ has a unique solution $\Leftrightarrow m=n$ and $A$ has a full rank. Otherwise there are more than one solutions.

**Definition 2. (Generalized Inverse)** *Let $A$ be an $m\times n$ matrix. A matrix $G$ is called generalized inverse of $A$ if $Gb=b'$ is a solution of $Ax=b$, whenever $Ax=b$ has a solution.*

**Note 3.** If $A$ is an invertible square matrix, then Generalized Inverse of $A$ is $A^{-1}$.

## 2.1 Sweep-out Method

Following operations on any $m\times n$ matrix are called elementary row operations:

i. Interchange two rows. This corresponds to pre-multiplying the matrix by a Permutation Matrix.

ii. Multiply a rwo by a non-zero constant. This corresponds to pre-multiplying the matrix with matrix of the form $\begin{pmatrix} 1 & 0 & 0 & 0 & \dots & 0\\ 0 & 1 & 0 & 0 & \dots & 0\\ 0 & 0 & 1 & 0 & \dots & \vdots\\ 0 & 0 & \dots & \alpha & \dots & \vdots\\ 0 & 0 & \dots & \dots & \ddots & . \end{pmatrix}$ which is a diagonal matrix and only of the diagonal elements different from 1.

iii. Addition of two rows. This corresponds to pre-multiplying the matrix with matrix of the form $I_{n\times n}+A_{ij}$ where $A_{xy}=(a_{ij})_{n\times n}$ and $a_{ij}=1$ if $i=x, j=y$ and 0 otherwise.

**Note 4.** It is easy to observe that each of the operation above is invertible. It is easy to construct the corresponding inverse matrix for each of the operations. However, these operations dont commute in general.

**Lemma 1.** *If a matrix $A$ in a field $\mathbb{F}$ can be reduced to a matrix $A'$ by series of elementary row operations with corresponding pre-multiplication matrices as $P_1, P_2, \ldots, P_t$ then $A'=\left(\prod_{i=1}^t P_i\right)A$.*

**Note 5.** In all complexity calculations we consider addition and multiplication on the underlying field are $O(1)$ operations.

While matrix computation in general takes $O(n^3)$ steps, given the special nature of matrices corresponding to elementary row operations, multiplication by these matrices can be realized much more effeciently.

**Lemma 2.** *Given an $n\times n$ matrix $M$ in a field $\mathbb{F}$. There exists an $O(n)$ algorithm to compute the resulting matrix obtained by multiplying a matrix correponding to elementary row operations.*

---

**Definition 3. (Hermite Canonical Form)** *A Square matrix $M$ is said to be HCF if:*

*i. $M$ is upper triangular.*

*ii. Diagonal elements of $M$ are either 0 or 1.*

*iii. If diagonal element is 1 then all other entries in that column are 0.*

*iv. If diagonal element is 0, then all other entries in that row are 0.*

---

**Theorem 2. (Hermite Canonical Form)** *Any Square matrix whose elements are from a field, can be converted to* ***Hermite Canonical Form*** *using elementary row operations in $O(n^3)$ time.*

**Proof.** We shall specify the steps to construct the required canonical form with inductive justification:

Let $A=(a_{ij})_{n\times n}$ be the given matrix with elements from a Field.

If $a_{11}\neq 0$, then sweep out the first column using $a_{11}$ as pivot.

If $a_{11}=0$, then if $\exists i>1$ such that $a_{i1}\neq 0$, then interchange rows $i, 1$ and sweep out the first column using the new $a_{11}$ as the pivot. Otherwise, whole of first column is zero.

Suppose, that after $k-1$ steps, the $(k-1)\times(k-1)$ principal sub-matrix is in HCF.

If $a_{kk}\neq 0$ sweep out the $k^{\text{th}}$ column using $a_{kk}$.

If $a_{kk}=0$, and if $\exists l>k$ such that $a_{kl}>0$, If $\exists i<k$, such that $a_{ik}\neq 0$, but $a_{ii}=0$, then interchange $i, j$ rows and sweep out the $k^{\text{th}}$ column using the new $a_{pk}$ as the pivot.

After these operations the principle submatrix of the resulting matrix of order $k$ is in $HCF$.

These operations suggest the algorithm and use at most $O(n^3)$ field operations. □

---

**Note 6.** Above method clearly works for non square matrices as well.

**Theorem 3.** *Consider a system of equations $Ax = b$, where $A$ is an $n \times n$ matrix in a field $\mathbb{F}$. If an augmented matrix formed as $[A\colon I_{n\times n}\colon b]$ can be redued to its Hermite Canonical Form as $[H_{n\times n}\colon G_{n\times n}\colon d]$. Then following statements hold:*

*i. The #1s on the diagonal of $H$ is rank of $A$.*

*ii. $G$ is a generalized inverse of $A$.*

*iii. System of equations $Ax = b$ is consistent $\Leftrightarrow$ $d_i = 0$ whenever $h_{ii} = 0$.*

*iv. Any solution of the system of equations $Ax = b$ is of the form: $d + (1 - H)z, z \in \mathbb{F}^n$.*

*v. The columns of $H$ form a basis for Null Space of $A$.*

*vi. Let $1 \leqslant i_1 \leqslant i_2 \ldots \leqslant n$ be such that $h_{i_j,i_j} = 1$ and all $h_{ii} = 0$ and $B = [A_{*i_1}\colon A_{*i_2}\colon \ldots\colon A_{*i_r}]$ and $C = \begin{pmatrix} H_{i_1*} \\ H_{i_2*} \\ \vdots \\ H_{i_r*} \end{pmatrix}$ then $A = B \times C$.*

---

**Theorem 4.** *Let $A$ be an $n \times n$ non singular matrix. Then there is a permutation matrix P, s.t. all principal sub-matrices of $PA$ are non-singular.*

**Proof.** Steps are similar to Theorem:14. In each step collect all the permutations required in order. Then $P$ is the product of all those permutation matrices taken in order. □

---

**Theorem 5. (LU Decomposition)** *Let $A$ be an $n \times n$ non singular matrix, such that all the principal sub matrices of $A$ are non-singular, then we can write $A = LU$, where*

$$L = \begin{pmatrix} 1 & 0 & 0 & \ldots & 0 \\ l_{21} & 1 & 0 & \ldots & 0 \\ l_{31} & l_{32} & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ l_{n1} & l_{n2} & l_{n3} & \ldots & 1 \end{pmatrix}_{n\times n}$$

$$U = \begin{pmatrix} u_{11} & u_{12} & \ldots & u_{1n} \\ 0 & u_{22} & \ldots & u_{2n} \\ 0 & 0 & & \vdots \\ \vdots & \vdots & & \\ 0 & 0 & \ldots & u_{nn} \end{pmatrix}_{n\times n}$$

*and both $L, U$ can be computed in $O(n^3)$ time.*

**Proof.** Since $A$ is a such that all its principal sub matrices are non-singular we can solve for each of $l_{ij}$ and $u_{ij}$ by multiplying $L$ and $U$ and then comparing the result with $A$. □

---

# 3 LUP decomposition of Matrix and Applications

### 3.0.1 Efficient Methods for Multiplying Matrices and Polynomials

While general $n \times n$ matrix and $n^{\text{th}}$ order Polynomial multiplication takes $O(n^3)$ time. Efficient methods for multiplication exist. Earliest successful attempt can be traced back to Karatsuba's algorithm for matrix multiplication published around 1950. This method is believed to be motivation for research towards more efficient matrix multiplication as well and first successful efficient algorithm for matrix multiplication has been published by Strassen in 1967.

Karatsuba's Algorithm for Multiplication of two polynomials. Let $P(x), Q(x)$ be two degree-$2n$, $n \in \mathbb{N}$, polynomials expressed as: $P(x) = x^n P_1(x) + P_0(x)$, $Q(x) = x^n Q_1(x) + Q_0(x)$. The product of both the polynomials can be computed if we have $P_0Q_0, P_1Q_0 + P_0Q_1, P_1Q_1$ computed, which normally requires computing four multiplications. However by using extra addition which is cost effective than multiplication we can reduce the number of required multiplications to three, as follows: We can compute $P_0Q_0, P_1Q_1, (P_1 + P_0)(Q_1 + Q_0) - P_0Q_0 - P_1Q_1$ which precisely give the 3 expressions required for computing product with only 3 multiplications. This suggests a recursive divide and conquor algorithm of order complexity: $O\big(n^{\log_2^3}\big)$ as opposed to the normal method which require $O(n^2)$ time.

Strassen's algorithm for computing the product of a matrices is similar, however require computing more non-trivial expressions. To compute product of two matrices of order $2n \times 2n$, if we visualize the matrices as follows:

$$C_{2n \times 2n} = A_{2n \times 2n} B_{2n \times 2n}$$

$$\begin{pmatrix} C^0_{n\times n} & C^1_{n\times n} \\ C^2_{n\times n} & C^3_{n\times n} \end{pmatrix}_{2n\times 2n} = \begin{pmatrix} A^0_{n\times n} & A^1_{n\times n} \\ A^2_{n\times n} & A^3_{n\times n} \end{pmatrix}_{2n\times 2n} \begin{pmatrix} B^0_{n\times n} & B^1_{n\times n} \\ B^2_{n\times n} & B^3_{n\times n} \end{pmatrix}_{2n\times 2n}$$

In the above scheme, computing matrix $C$ requires computing $C^i_{n\times n}$ and it requires computing products of 8 sub-matrices: $A^0B^0$, $A^1B^2$, $A^0B^1$, $A^1B^3$, $A^2B^0$, $A^3B^2$, $A^2B^1$, $A^3B^3$. However, we can reduce this to compting 7 products, by introducing extra additions which are cheaper than computing products, as follows - compute the following 7 expressions:

$$\begin{aligned}
m_1 &= (A^1_{n\times n} - A^3_{n\times n})(B^0_{n\times n} + B^3_{n\times n}) \\
m_2 &= (A^0_{n\times n} + A^3_{n\times n})(B^0_{n\times n} + B^3_{n\times n}) \\
m_3 &= (A^0_{n\times n} - A^2_{n\times n})(B^0_{n\times n} + B^1_{n\times n}) \\
m_4 &= (A^0_{n\times n} + A^1_{n\times n})(B^3_{n\times n}) \\
m_5 &= (A^0_{n\times n})(B^1_{n\times n} - B^3_{n\times n}) \\
m_6 &= (A^3_{n\times n})(B^2_{n\times n} - B^0_{n\times n}) \\
m_7 &= (A^2_{n\times n} + A^1_{n\times n})(B^0_{n\times n})
\end{aligned}$$

which in total require computing 7 products of order $n \times n$. Then matrix $C$ can be computed as

$$C = \begin{pmatrix} m_1 + m_2 - m_4 + m_6 & m_4 + m_5 \\ m_6 + m_7 & m_2 - m_3 + m_5 - m_7 \end{pmatrix}_{2n\times 2n}$$

This procedure, clearly suggests a divide and conquer algorithm. If $T(n)$ denotes the time required for computing a matrix of order $n \times n$ then $T(n) = 7T\left(\frac{n}{2}\right) + 18\left(\frac{n^2}{4}\right)$. Since the above procedure requires computing 7 products and 18 sums of order $\frac{n}{2} \times \frac{n}{2}$. Then $T(n) = O\big(n^{\log_2^7}\big) = O(n^{\sim 2.81})$.

**Note 7.** In the above method clearly works only when the order of matrix is a power of 2. Given a matrix of arbitrary order, we embed the it in a larger square matrix whose order is a power of 2 to use the above method.

**Note 8.** Strassen's Algorithm being the first efficient method for multiplication of matrices, has created the interest in the question if multiplication can be performed better.

**Notation 3.** *Time complexity for matrix multiplication of* $n \times n$ *matrices is denoted as* $M(n) = O(n^{\omega})$. *If* $\omega = 2 + \varepsilon$, $\varepsilon$ *is called the exponent of linear algebra.*

### 3.0.2 Matrix Multiplication ⇔ Matrix Inversion

**Theorem 6.** *Matrix multiplication is no harder than Matrix Inversion.*

**Proof.** Suppose there exists and algorithm $\mathcal{A}$ to invert a Matrix. Then one can construct an algorithm $\mathcal{B}$ for multiplication as follows:

Let $X, Y$ be two $n \times n$ matrices to be multiplied,

construct $D = \begin{pmatrix} I_{n\times n} & X & Y \\ O_{n\times n} & I_{n\times n} & Y \\ O_{n\times n} & O_{n\times n} & I_{n\times n} \end{pmatrix}_{3n\times 3n}$. Then the matrix $D$ is invertible and inverse given as $D^{-1} = \begin{pmatrix} I_{n\times n} & -X & XY \\ O_{n\times n} & I_{n\times n} & -Y \\ O_{n\times n} & O_{n\times n} & I_{n\times n} \end{pmatrix}_{3n\times 3n}$. This can be verified by explicit computation.

So, we shall construct algorithm $\mathcal{B}$ as follows:

1. Accept two matrices $X_{n\times n}, Y_{n\times n}$.
2. Construct the matrix $D$ as illustrated above.
3. Call Algorithm $\mathcal{A}$ with input as $D$ to get $D^{-1}$.
4. Product of $X, Y$ is the right principal submatrix of $D^{-1}$ of order $n \times n$. □

**Lemma 3.** *If* $A$ *is an* $2n \times 2n$ *matrix. written as* $A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}_{2n\times 2n}$ *each* $A_{ij}$ *an* $n \times n$ *submatrix. If* $A$ *is non-singular then* $A_{11}$ *is non singular and the inverse of* $A$ *is given as:*

$$A^{-1} = \begin{pmatrix} A_{11}^{-1} + A_{11}^{-1}A_{12}S^{-1}A_{21}A_{11}^{-1} & -A_{11}^{-1}A_{12}S^{-1} \\ -S^{-1}A_{21}A_{11}^{-1} & S^{-1} \end{pmatrix}$$

*where* $S = A_{22} - A_{21}A_{11}^{-1}A_{12}$.

**Theorem 7.** *Matrix Inversion is no harder than Matrix multiplication.*

**Theorem 8.** *If* $A$ *is a matrix that is invertible, then* $A$ *can be written as* $A = LUP$ *where* $P$ *is a product of permutation matrices.*

**Proof.** From Theorem 17. If $A$ is invertible then there exist permutation matrices $Q_1, Q_2, \ldots, Q_t$ such that $A\left(\prod_{i=1}^{t} Q_i\right) = B$ is such that all principal sub-matrices of $B$ are invertible.

Let $Q = \prod_{i=1}^{t} Q_i$. Clearly $Q$ is invertible, since each of $Q_i$ is invertible. By Theorem 18. $B = AQ$ can be decomposed as $LU$. So, $AQ = LU$. So, choose $P = Q^{-1}$. □

We state the following theorem without proof.

**Theorem 9.** *$LUP$ decomposition of matrix* $A$ *can be computed in* $O(M(n))$ *time.*

$LUP$ decomposition of matrix can be used to compute determinant of the matrix efficiently.

### 3.0.3 Computing matrix determinant.

**Lemma 4.** *If we can decompose a matrix* $A = LUP$*,then* $\det(A) = (-1)^{\mathrm{Sgn}(P)}\det(U)$.

**Proof.** If $A = LUP$ then we have $\det(A) = \det(L)\det(U)\det(P)$.

Since $P$ is a product of permuation matrices, det $(P) = (-1)^{\mathrm{Sgn}(P)}$, which can be computed in $O(n)$ time. Since $L$ is a lower triangular matrices with all its diagonal elements 1, we have det $(L) = 1$. Hence, det $(A) = (-1)^{\mathrm{Sgn}(P)}\det(U)$. □

This suggests an algorithm to compute the determinant in $O(M(n))$ steps.

**Theorem 10.** *Determinant of an $n \times n$ matrix can be computed in $O(M(n))$ steps.*

**Proof.** Compute the determinant if matrix $A_{n \times n}$ as follows:

1. Compute LUP decomposition of matrix $A = LUP$
2. Compute det $(P) = (-1)^{\mathrm{Sgn}(P)}$.
3. Compute det $(U) = \prod_{i=1}^{n} u_{ii}$.
4. det $(A) = (-1)^{\mathrm{Sgn}(P)} \prod_{i=1}^{n} u_{ii}$.

Since, $LUP$ decomposition requries $O(M(n))$ steps and all other steps require $O(n)$ steps, determinant can be computed in $O(M(n))$ steps. □

Lemma 23. Suggests a very efficient method for computing inverse of a non singular matrix. If $A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}_{2n \times 2n}$ is a non singular matrix, and is upper triangular then $A_{21} = 0_{n \times n}$. $A_{11}$, $A_{22}$ are upper triangular and so is $A^{-1} = \begin{pmatrix} A_{11}^{-1} & -A_{11}^{-1} A_{12} A_{22}^{-1} \\ 0_{n \times n} & A_{22}^{-1} \end{pmatrix}_{2n \times 2n}$. This suggests a very simple divide and conquer algorithm for computing inverse of an upper triangular matrix:

```
InvertUTMatrix($A$):
1. $A_{11}^{-1} \leftarrow$ InvertUTMatrix($A_{11}$).
2. $A_{22}^{-1} \leftarrow$ InvertUTMatrix($A_{22}$).
3. $A^{-1} = \begin{pmatrix} A_{11}^{-1} & -A_{11}^{-1} A_{12} A_{22}^{-1} \\ 0 & A_{22}^{-1} \end{pmatrix}$.
```

If $T(n)$ is the number of steps required to compute the inverse of $n \times n$ upper triangular matrix. Then $T(n) = 2T\left(\frac{n}{2}\right) + 2\left(\frac{n}{2}\right)^3 + \left(\frac{n}{2}\right)^2$.

**A MATRIX MULTIPLICATION INEQUALITY IS ASSUMED HERE!** FILL THE DETAILS.

For a general matrix, we can use $LUP$ decomposition and use the above method to invert the matrices $U, L$. $P$ being a permutation matrix is invertible in $O(n)$ steps for an $n \times n$ matrix. So, Inversion of matrix can be done in $O(M(n))$ time.

**Theorem 11.** *Given a non singular matrix A. Inverse of A can be computed in $O(M(n))$ steps.*

So the methods above describe computing determinant and inverse of a matrix. Determinant of the matrix can be defined over a Ring but all the above operations assume a field. This restriction can be overcome in case of Integral Domains by embedding the elements of given matrix into the field of fractions. But the issue is growth of intermediate fractions, so even though total number of operations wont change, each operation may become very costly. So an alternative for integer matrices is to compute determinants in prime fields and then combine the results.

Given a matrix $M$ whose entries are integers, suppose there is an apriori known bound on det $(M) < \mathbf{\Delta}$. Let $p_1$, $p_2$, ..., $p_r$ be distinct primes such that $\mathbf{\Delta} < p_1 p_2 ... p_r$. Then compute $\mathbf{\Delta}_i$ = det $(M)(\mathrm{mod}\ p_i)$ and combine the results by using Chinese Reminder Theorem to obtain det $(M)(\mathrm{mod}\ p_1 p_2 ... p_r) = d$. But since $\mathbf{\Delta} < p_1 ... p_r$ we have $d =$ det $(M)$. Such a bound on determinant can be obtained using hardamard bound on determinant given as $\mathbf{\Delta} \leqslant \prod_{1 \leqslant i \leqslant n} \left(\sum_{1 \leqslant j \leqslant n} |m_{ij}|^2\right)^{\frac{1}{2}}$.

# 4 Upper Hessenberg Form

**Definition 4. (Characteristic Polynomial)** *Characteristic Polynomial of a matrix $A$ denoted as $Ch_A(x) := \det(xI - A)$.*

**Lemma 5.** *If $A$ is an $n \times n$ matrix then $\deg(Ch_A(x)) = n$.*

**Theorem 12. (Caley Hamilton)** *Every square matrix satistifies its Characteristic polynomial.*

**Definition 5. (Minimal Polynomial)** *A Minimal polynomial of matrix $A$ is least degree monic polynomial $m_A(x)$ such that $m_A(A) = O_{n\times n}$.*

---

**Lemma 6.** *If $A_{n\times n}$ is a matrix then $m_A(x)|Ch_A(x)$.*

**Proof.** Suppose $m_A(x)\nmid Ch_A(x)$ then by remainder theorem, we have $Ch_A(x) = m_A(x)q(x) + r(x)$ where $\deg(r) < \deg(m)$.
But, by definition we have $m_A(A) = 0_{n\times n}$. and Caley-Hamilton theorem $Ch_A(A) = 0$.
So, $Ch_A(A) = 0 = m_A(A)q(A) + r(A) = 0 + r(A)$
$\Rightarrow r(A) = 0$. Contradicting the minimality of $m_A(x)$. □

---

**Proposition 1.** *If $A$ is a matrix and $\rho(x)$ is an irreducible polynomial which divides $Ch_A(x)$ then $\rho(x)|m_A(x)$.*

**Proof.** Since $m_A(x)$ is a polynomial we have $(x - y)|m_A(x) - m_A(y)$.
So we can write $m_A(x) - m_A(y) = (x - y)k(x, y)$.
Substituting $x = \lambda I$ and $y = A$, we have $m_A(\lambda I) - m_A(A) = m_A(\lambda I)$ (by definition. of $m_A$)
$\Rightarrow m_A(\lambda I) = (\lambda I - A)k(\lambda I, A)$
$\Rightarrow m_A(\lambda)I = (\lambda I - A)k(\lambda I, A)$
$\Rightarrow \det(m_A(\lambda)I) = \det(\lambda I - A)\det(k(\lambda I, A))$
$\Rightarrow (m_A(\lambda))^n = Ch_A(\lambda)\det(k(\lambda I, A))$
So, if $\rho(\lambda)|Ch_A(\lambda)$ then $\rho(\lambda)|(m_A(\lambda))^n$. But since $\rho(x)$ is irreducible $\rho(x)|m_A(x)$. □

---

**Corollary 1.** *The distinct roots of characteristic polynomial and minimal polynomials of a matrix are the same they differ only in multiplicities.*

How to compute the minimal polynomial for a given matrix? It is clear to see that $I$, $A$, ..., $A^n$ are linearly dependent, since characteristic polynomial is of degree $n$. So find smallest $r$ such that $I$, $A$, ..., $A^r$ are linearly dependent, $r$ will be the deg of the minimal polynomial of $A$ and coefficients of minimal polynomial can be obtained as coefficients of linear dependence. This can be done using Guassian Elimination which takes $O(n^4)$ time. This however is not the best, there exist algorithms to compute minimal polynomial in $O(n^3)$ time.

How to compute the characteristic polynomial? Let $F_n[x]$ be the vector space of all polynomial over $F$ whose degree is atmost $n$. Then $F_n[x]$ has a dimension of $n + 1$ over $F$ with a basis $\{1, x, x^2, ..., x^n\}$. We shall give a more non trivial basis for $F_n[x]$.

Let $c_i$, $1 \leqslant i \leqslant n$ be $n$ distinct elements of $F$. We shall define polynomials, $p_i(x) = \frac{(x - c_1)(x - c_2)...(x - c_{i-1})...(x - c_{i+1})...(x - c_n)}{(c_i - c_1)(c_i - c_2)...(c_i - c_n)}$ for $1 \leqslant i \leqslant n$. Then we can observe that $p_i(c_j) = 1 \Leftrightarrow i = j$ and 0 otherwise.

**Lemma 7.** *$\{p_1(x), ...p_n(x)\}$ forms a basis for $F_n[x]$ over $F$.*

Any $f(x) \in F_n[x]$ can be written as $f(x) = \sum_{1\leqslant i\leqslant n} f(c_i)p_i(x)$. This observation can be used to compute the characteristic polynomial of matrix $M$. We compute $Ch_A(c_i) = \det(c_iI - A)$ for all $i$. Then write $Ch_A(x) = \sum_{1\leqslant i\leqslant n} Ch_A(c_i)p_i(x)$. Since computing $n \times n$ determinant in $F$ takes $O(M(n))$ time computing characteristic polynomial takes $O(nM(n))$ time.

**Definition 6.** *Let $A$, $B$ be two $n\times n$ matrices. They are said to be similar if there exists an invertible matrix $C$ such that $A = CBC^{-1}$.*

**Note 9.** Similarity is an equivalence relation.

**Lemma 8.** *If $A, B$ are similar then they have same characteristic polynomial.*

**Exercise 1.** Find two non-similar matrices which have same characteristic polynomial.

**Definition 7. (Upper Hessenberg Form)** *A matrix of the following form is said to be in upper hessenberg form.*

$$\begin{pmatrix} h_{11} & h_{12} & h_{13} & \dots & h_{1n-1} & h_{1n} \\ k_2 & h_{22} & h_{23} & \dots & h_{2n-1} & h_{2n} \\ 0 & k_3 & h_{33} & \dots & h_{3n-1} & h_{3n} \\ 0 & 0 & k_4 & \dots & h_{4n-1} & h_{4n} \\ \vdots & & & & & \vdots \\ 0 & 0 & 0 & \dots & k_n & h_{nn} \end{pmatrix}_{n\times n}$$

**Theorem 13.** *Given a matrix $A$ we can apply a series of similarity transforms on it to get it into Upper Hessenberg Form.*

# 5 Smith Canonical Form

**Definition 8. (Non-derogatory Matrix)** *A matrix $M$ is said to be non-derogatory if minimal polynomial of matrix $P_M(x)$ is same as characterisic polynomial of $M$.*

**Definition 9. (Elementary Jordan Matrix)** *A matrix of the following form is called Elementary Jordan Matrix.*

$$J_n(c) = \begin{pmatrix} c & 0 & 0 & 0 & \dots & 0 \\ 1 & c & 0 & 0 & \dots & 0 \\ 0 & 1 & c & 0 & \dots & 0 \\ & & & & \vdots & \\ 0 & 0 & 0 & \dots & 1 & c \end{pmatrix}_{n\times n}$$

**Note 10.** Elementary Jordan Matrix is Non-derogatory matrix and that can be observed by computing the successive powers of the matrix that minimal polynomial cannot have degree less than $n$.

**Definition 10. (Companion matrix of Polynomial)** *Companion matrix of the polynomial $p(x) = x^n + a_{n-1}x^{n-1} + \dots + a_1x + a_0$ denoted by $C(p)$ is the matrix of the following form:*

$$\begin{pmatrix} 0 & 0 & \dots & 0 & -a_0 \\ 1 & 0 & \dots & 0 & -a_1 \\ \vdots & & & & \\ 0 & 0 & \dots & 1 & -a_{n-1} \end{pmatrix}_{n\times n}$$

**Note 11.** It is easy to observe that characteristic polynomial of $C(p)$ is $p(x)$.

**Notation 4.** *Henceforth we shall denote characteristic polynomial of a matrix $M$ as $Ch_M(x)$.*

**Definition 11. (Direct Sum)** *The direct sum of two matrices square matrices $A_{n\times n}, B_{m\times m}$ is the matrix $\left(\begin{smallmatrix} A & 0 \\ 0 & B \end{smallmatrix}\right)_{(m+n)\times(m+n)}$ and is denoted as $A\oplus B$.*

**Lemma 9.** $\det(A \oplus B) = \det(A)\det(B)$

We shall state the following result without proof.

**Theorem 14. (Jordan Canonical Form)** *Every matrix is similar to a direct sum of elementary jordan matrices, which is unique upto a permutation of elementary jordan blocks.*

**Notation 5.** *We shall denote $F$ for a field and $F[x]$ for ring of polynomials over field $F$. We shall denote the set of $n \times n$ matrices in ring $R$ as $M_{n \times n}(R)$.*

**Definition 12. (Rational Canonical Form)** *Let $M = \oplus_{k=1}^{s} C(d_k)$, where $d_1, d_2, \ldots, d_s$ are non-constant monic polynomials in $F[x]$ such that $d_i | d_{i+1}$. Then $M$ is said to be in Rational Canonical Form.*

**Definition 13. (Invariant Factors)** *If a matrix $A$ is $\sim$ to $M = \oplus_{k=1}^{s} C(d_k)$ then the polynomials $d_i$ are called invariant factors of $A$.*

Following results show why Rational Canonical Form is useful in computational prespective.

**Lemma 10.** *Minimal polynomaial of a matrix $M = \oplus_{j=1}^{s} C(d_j)$ is* $\mathrm{LCM}(d_1, \ldots d_s)$.

---

**Theorem 15.** *If $d_1, d_2, \ldots d_s$ are invariant factors of $A$, then*

*i.* $m_A(x) = d_s(x)$

*ii.* $Ch_A(x) = d_1 d_2 \ldots d_s$

**Proof.** □

---

**Definition 14.** *A matrix $P$ in $M_{n \times n}(F[x])$ is said to be unit if there exist a matrix $Q \in M_{n \times n}(F[x])$ such that $PQ = I$.*

**Lemma 11.** *If matrices $P, Q$ are units then so is $PQ$.*

**Lemma 12.** *If matrix $P \in M_{n \times n}(F[x])$ is unit then* $\det(P) \in F$.

In the following discussion, following operations on matrices are treated as elementary row operations:

i. interchange two rows.

ii. multiply a row by an element of $F[x]$ and add it to another row.

Both the above operations correspond to multiplication of original matrix by another matrix in $M_{n \times n}(F[x])$.

**Lemma 13.** *Matrices corresponding to elementary row operations are units.*

**Definition 15. (Unimodular Matrix)** *Products of elementary row matrices and elementary column matrices are called* ***Unimodular****.*

We shall state the following theorem without proof.

**Theorem 16.** *A matrix is a unit iff it is unimodular.*

**Definition 16.** *Let $A, B \in M_{n \times n}(F[x])$. Then $A$ is said to be equivalent to $B$, if there are units $P, Q$ such that $PAQ = B$.*

**Definition 17.** *Let $A \in M_{n\times n}(F[x])$ then for $1 \leqslant k \leqslant n$ let $d_k(A)$ denote the gcd of all $k \times k$ minors of $A$. Then $d_k(A)$ is called the $k^{\text{th}}$ determinant divisor of $A$.*

**Lemma 14.** *If two matrices in $M_{n\times n}(F[x])$ are similar then they have the same determinantal divisors.*

**Note 12.** The above notion can also be exnted to rectangular matrices.

**Note 13.** $\gcd(f_1, \dots f_n) \neq 0 \Leftrightarrow$ atleast one of $f_i$ is non-zero.

**Definition 18. (Determinant Rank)** *Determinant rank of a matrix $A$ denoted by $\rho(A)$ is the largest integer $r$ for which $d_r(A) \neq 0$.*

---

**Theorem 17.** *If $A$ is a matrix in $M_{n\times n}(F[x])$ then for $1 \leqslant k \leqslant \rho(A)$ $d_k(A)|d_{k+1}(A)$.*

**Proof.** □

---

**Theorem 18. (Smith Canonical Form)** *Every non-zero matrix $A \in M_{n\times n}(F[x])$ with $r = \rho(A)$ is equivalent to matrix of the following form*

$$\begin{pmatrix} f_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & f_2 & 0 & \dots & 0 & 0 \\ 0 & 0 & f_3 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \dots & f_r & \dots & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 \\ \vdots & & & & & \vdots \\ 0 & 0 & 0 & 0 & \dots & 0 \end{pmatrix}_{n\times n}$$

*where $f_i \in F[x]$ and $f_i | f_{i+1}$ this is called Smith Canonical Form of the matrix.*

**Theorem 19.** *Let $A \in M_{n\times n}(F[x])$ and $r = \rho(A)$ then the polynomials $f_1$, $f_2$, $\dots f_r$ in the smith canonical form of $A$ are called the invariant factors of $A$.*

---

**Lemma 15.** *If $d \in F[x]$ then the smith canonical form of $xI - c(d)$ is $\operatorname{diag}(1, 1, \dots 1, d)$.*

**Proof.** □

---

**Theorem 20.** *For every matrix $A \in M_{n\times n}(F[x])$ smith canonical form is unique.*

**Proof.** □

---

Following theorem gives connection between Rational Canonical Form and Smith Canonical Form.

**Theorem 21.** *Let $B \in M_{n\times n}(F)$. If the invariant factors of $B$ are $d_1, \dots d_s$ then smith canonical form of $xI - B$ is equal to $\operatorname{diag}(1, 1, \dots, 1, d_1, \dots d_s)$.*

**Theorem 22.** *Let $A, B \in M_{n\times n}(F)$. Then $A$ is similar to $B$ iff $xI - A$ is equivalent to $xI - B$ iff $xI - A$ and $xI - B$ have the same smith canonical form.*

So to test similarity of two matrices $A, B \in M_{n\times n}(F)$ we resort to following steps:

1. Obtain $U, V$ such that $U(xI - A)V = \operatorname{diag}(p_1(x), ..., p_n(x))$
2. Obtain $R, Q$ such that $R(xI - B)W = \operatorname{diag}(q_1(x), ..., q_n(x))$
3. If $p_i = q_i$ for $1 \leqslant i \leqslant n$ then $A \sim B$.
4. Suppose similarity holds. Then $P(xI - A)Q = xI - B$ where $P = R^{-1}U$ and $Q = VW^{-1}$ then $T$ computed as right value of $Q(x)$ at $B$ and then $T^{-1}$ is left value of $P(x)$ at $B$ is such that $TAT^{-1} = B$.

## 6 Fast Fourier Transform

This part is mostly covered from "The Design and Analysis of Computer Algorithms" by Aho, Hopcroft and Ullman.

### 6.1 Bit-Complexity of Fast Fourier Transform

### 6.2 Schonhage-Strassen algorithm

## 7 Computational complexity of Fundamental Integer operations.

This part is mostly covered from "The Design and Analysis of Computer Algorithms" by Aho, Hopcroft and Ullman.

## 8 Greatest Common Divisor

**Definition 19. (GCD)** *If $R$ is a euclidean ring and $a, b \in R$ then $g$ is called the* $\operatorname{GCD}(a, b)$ *if $g|a$, $g|b$ and if for any $h \in R, h|a, h|b$ then $h|g$.*

### 8.1 Euclidean GCD Algorithm

### 8.2 Half-GCD Algorithm

## 9 Polynomial Factoring

### 9.1 Berlekamp's Method

Fix a prime finite field $\mathbb{F}_p$. Let $f(x) \in \mathbb{F}_p[x]$ and $\deg(f) = n$. Without loss of generality we shall consider $f$ as a square-free polynomial, for otherwise we can make it square free as follows.

**Definition 20. (Derivative)** *If $f(x) \in R[x]$ where $R$ is a ring, and $f(x) = \sum a_i x^i$ then we shall define derivative of $f$ as $f'(x) = \sum i a_i x^{i-1}$.*

**Notation 6.** *Going further we assume $p \in \mathbb{N}$ is a prime number and $\mathbb{F}_p$ is the finite field of order $p$.*

**Lemma 16.** *$f \in \mathbb{F}_p[x]$ is square-free iff $\gcd(f, f') = 1$.*

**Lemma 17.** *If $f \in \mathbb{F}_p[x]$ is not square-free then $\frac{f(x)}{g(x)}$ is square-free, where $g(x) = \gcd(f, f')$.*

**Note 14.** There is a procedure to check if given polynomial is irreducible, Ref. Mc.William-sloane Ch.4.

To factor the polynomial $f(x)$, the basic intuition is that we take the gcd of $f(x)$ with "some" polynomial, such that gcd turns out to be non-trivial.

Over $\mathbb{F}_p$ we know that $x^p \equiv x \pmod p$, $\forall x \in \mathbb{F}_p$. Also let $g(x) = (x^p - x)' = p x^{p-1} - 1 = -1$ in $\mathbb{F}_p[x]$. So $f(x) = x^p - x$ is square-free. Hence $f(x) = (x^p - x) = \prod_{i \in \mathbb{F}_p} (x - i)$. So for any $v(x) \in \mathbb{F}_p[x]$ we have $(v(x))^p - v(x) = \prod_{i \in \mathbb{F}_p} (v(x) - i)$. And this is a square free splitting as $\gcd((v(x) - i), (v(x) - j)) = 1$ for $i \neq j$. So for a given square-free $f(x)$, if we can find $v(x)$ such that $(v(x))^p \equiv (v(x)) \bmod f(x)$ then any irreducible factor of $f(x)$ will divide exactly one factor $v(x) - k$. So we shall try to find such $v(x)$. We shall start by assuming $v(x)$ with required property and derive the necessary condition.

Let $f(x)$ be a square-free polynomial in $\mathbb{F}_p[x]$ and let $v(x)$ be a polynomial such that $(v(x))^p - v(x) \equiv 0 \pmod{f(x)}$. Let $v(x) = \sum_{0 \leqslant i \leqslant t} (a_i x^i)$ then in $\mathbb{F}_p[x]$ we have $(v(x))^p = \sum_{0 \leqslant i \leqslant t} (a_i x^{ip})$. Then $(v(x))^p \pmod{f(x)} = \sum_{0 \leqslant i \leqslant t} a_i (x^{pi} \bmod f(x))$. Each of Let $x^{pi} \bmod f(x)$ is of degree less than $n = \deg(f(x))$, so we shall each of $(x^{pi} \bmod f(x))$ as $\left(\sum_{0 \leqslant j < n} b_{i,j} x^j\right)$. Then we can easily see following holds:

**Lemma 18.** *Choose the degree of $v(x)$ to be $n - 1$. Then taking the notations above and defining $B = (b_{i,j})_{n \times n}$. $(v(x))^p - v(x) \equiv 0 \pmod{f(x)}$ iff $\tilde{v} B = \tilde{v}$, where $\tilde{v} = (a_i)_{0 \leqslant i < n}$ is an $n$ vector of coefficients of $v(x)$.*

So, $\tilde{v}$ is in the null space of $B - I$. Since $B - I$ is an $n \times n$ matrix we can compute the basis of null space using sweep-out method in Sec.1 in $O(n^3)$ steps. Let the basis of $B - I$ be $\bigcup_{i<r} \{v^{[i]}(x)\}$. Where $r = \#$of irreducible factors of $f(x)$.

## 9.2 Cantor-Zassenhaus randomized algorithm

Let $v$ be a random linear combination of the basis vectors of the null space of $B - I$. Compute $\gcd\left((v(x))^{\frac{p-1}{2}} - 1, f(x)\right) = \gcd\left((v(x))^{\frac{p-1}{2}} - 1 (\bmod f(x)), f(x)\right)$ (here exponentiation can be done in $O(\log p)$ steps using square-multiply algorithm). We can argue that with probability atleast $\frac{4}{9}$ this gives a factor of $f(x)$.

We can analyze the performance by seeing in what all ways can gcd computation above can turn out to be trivial?

i. If $p_i(x) \Big| (v(x))^{\frac{p-1}{2}} - 1$ for all $i$. In this case gcd is $(v(x))^{\frac{p-1}{2}} - 1$

ii. If $p_i(x) \nmid (v(x))^{\frac{p-1}{2}} - 1$ for all $i$. In this case gcd is 1.

Consider the case i. Given a $p_i(x)$ and $v(x)$ there is a unique $s_i$ such that $p_i(x) | v(x) - s_i$. Then $(v(x))^{\frac{p-1}{2}} \equiv 1 (\bmod p(x))$ and $(v(x)) \equiv s_i (\bmod p_i(x))$. Both these conditions imply $s_i^{\frac{p-1}{2}} \equiv 1 (\bmod p)$ which happens iff $s_i$ is a Quadratic residue in $\mathbb{F}_p$.

Assumption: If $v(x)$ is chosen randomly from the null space of $B - I$ and $s_i$ is unifomly chosen from $\mathbb{F}_p$ Since there are $\frac{p-1}{2}$ quadratic residues in $\mathbb{F}_p$, the probability that $s_i$ is a Quadratic residue is $\frac{p-1}{2p}$. So probability for trivial case i. is $\left(\frac{p-1}{2p}\right)^r$.

Similarly, $\gcd\left((v(x))^{\frac{p-1}{2}} - 1, f(x)\right) = 1$ iff $s_i$ are all Quadratic Non Residues and this happens with probabilty $\left(\frac{p+1}{2p}\right)^r$. So, combining above cases probability that $\gcd\left((v(x))^{\frac{p-1}{2}} - 1, f(x)\right)$ is non-trivial is atleast $1 - \left(\frac{p-1}{2p}\right)^r - \left(\frac{p+1}{2p}\right)^r > \left(\frac{4}{9}\right)$.

## 9.3 Factorization over $\mathbb{Z}[x]$

Let $u(x)\in\mathbb{Z}[x]$ and $u(x)=u_0+u_1x+\ldots+u_nx^n$, without loss of generality assume $\gcd(u_0,u_1,\ldots,u_n)=1$ and $u(x)$ is square-free.

**Example 1.** $f(x)=x^8+x^6-3x^4-3x^3+8x^2+2x-5$ can be factored in $\mathbb{Z}_{13}$ as well as $\mathbb{Z}_2$ but is irreducible in $\mathbb{Z}$.

**Note 15.** There exist polynomials which have consistent degree factorization in $\mathbb{Z}_p$ for every prime $p$ but not in $\mathbb{Z}$.

**Note 16.** It is easy to be convinced that almost all polynomials over integers are irreducible.

**Lemma 19.** *Suppose $p$ is a big enough prime, such that coefficient in any true factorization of $u(x)=v(x)w(x)$ be in the range $\left(\frac{-p}{2},\frac{p}{2}\right)$ then factoring in $\mathbb{Z}_p$ we obtain true factorization over $\mathbb{Z}$.*

However there are two problems: In general it is difficult to get a good bound on $p$, $p$ may be too large. So, solution is to use a method called **"Hensel Lifting"**

### 9.3.1 Hensel Lifting

**Lemma 20.** *Let $u(x),v(x),a(x)$ and $b(x)$ be such that:*

*i.* $u(x)=v(x)w(x)$ *in* $\mathbb{Z}_q$.

*ii.* $a(x)v(x)+b(x)w(x)\equiv1$ *in* $\mathbb{Z}_p$.

*iii.* $c.l(v)\equiv1(\operatorname{mod} r)$ ($l(v)$ is the leading coefficient of $v(x)$)

*iv.* $\deg(u)=\deg(v)+\deg(w)$

*v.* $r=\gcd(p,q)$

*then there exist polynomials $V(x),W(x)$ such that :*

*i.* $u(x)=V(x)W(x)$ *in* $\mathbb{Z}_{qr}$.

*ii.* $V(x)=v(x)$ *in* $\mathbb{Z}_q$, $W(x)=w(x)$ *in* $\mathbb{Z}_q$

*iii.* $l(v)=l(V)$, $\deg(v)=\deg(V)$, $\deg(w)=\deg(W)$.

**Proof.** □

---

**Lemma 21. (Hensel)** *Let $u(x)$, $v_e(x)$, $w_e(x)$, $a(x)$, $b(x)$ be such that $u(x)=v_e(x)w_e(x) \bmod p^2$ and $a(x)v_e(x)+b(x)w_e(x)\equiv1(\operatorname{mod} p)$ where $p$ is a prime and $v_e(x)$ is monic $\deg(a)\leqslant\deg(w_e)$ and $\deg(b)<\deg(v_e)$ and $\deg(u)=\deg(v_e)+\deg(w_e)$. Then, there are polynomials $v_{e+1}(x)$ and $w_{e+1}(x)$ satisfying the same conditions $e$ increased by 1. Further $v_{e+1}(x)$ and $w_{e+1}(x)$ are unique modulo $p^{e+1}$.*

Using this lemma once we have factorization in $\mathbb{Z}_p$ we can lift the factorization to $\mathbb{Z}_{p^2},\mathbb{Z}_{p^3},\ldots$ There is an improvement suggested to this by Zessenhaus using which we can lift the factorization much faster.

---

**Lemma 22. (Zessenhaus)** *Suppose $u(x)\equiv v(x)w(x)$ in $\mathbb{Z}_q$ and $a(x)v(x)+b(x)w(x)=1$ in $\mathbb{Z}_p$ where $p=\gcd(p,q)=r$. Then there are polynomials $V(x)$, $W(x)$, $A(x)$, $B(x)$ such that $u(x)=V(x)W(x)$ in $\mathbb{Z}_{qr}$ and $A(x)V(x)+B(x)W(x)\equiv1(\operatorname{mod} pr)$.*

**Proof.** □

# 10 Ideals and Varieties

In this section essentially we will be studying Hilbert Basis Theorem, Grobner Basis and Buchberger's Algorithm for computing Grobner's Basis[1].

## 10.1 Multivariate Polynomials

Moving from uni-variate to bi-variate or multi-variate polynomial rings requires a change in intuition and in this lecture we shall focus on the aspects that differentiate them. For example, we know that a polynomial of degree $n$ can have atmost $n$ roots in any field, but however this need not be the case for multivariate polynomials as the following example illustrates.

**Example 2.** Consider $f \in \mathbb{C}[x, y]$ and $f(x, y) = y - x$ then any element of the set $\{(c, c) | c \in \mathbb{C}\}$ is a root of $f$ and the set is infinite. So this is an example of a polynomial with finite degree with infinite number roots.

**Definition 21. (Monomial)** *Given the inderminates of the polynomial ring as $x_i, 1 \leqslant i \leqslant n$ any formal product of the form $x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_n^{\alpha_n}$ for $\alpha_i \geqslant 0$ is called a Monomial.*

**Definition 22. (Polynomial)** *If $\mathbb{F}$ is a field an element of the ring $\mathbb{F}[x_1, \ldots x_n]$ with $x_i$ as indeterminates is a set of finite formal sum of the form $\sum_\alpha a_\alpha x^\alpha$ where $\alpha \in \mathbb{N}^n$. An element of the ring is called Polynomial.*

**Note 17.** If $\alpha \in \mathbb{N}^n$ we shall define $|\alpha| = \sum_{1 \leqslant i \leqslant n} \alpha_i$.

**Note 18.** If $f \in \mathbb{F}[x_1, \ldots, x_n]$ then $\deg(f) = \max(\{|\alpha|, a_\alpha \neq 0\})$

**Definition 23. (Monomial Ordering)** *Given $x_i, 1 \leqslant i \leqslant n$ as inderminates, Monomial ordering is a linear ordering $(>)$ on the set of monomials formed with those inderminates satsisfying certain regularity properties:*

*i. If $\alpha, \beta, \gamma$ are monomials such that $\alpha > \beta$ then $\alpha\gamma > \beta\gamma$.*

*ii. $>$ is a well-ordering on set of Monomials.*

**Note 19.** It is obvious to see that ordering of monomials formed with $n$ indeterminates transforms into ordering on $\mathbb{N}^n$. Going forward we shall interchange between these ideas.

**Example 3. (Lex)** Given $\alpha, \beta \in \mathbb{N}^n$ we say that $\alpha >_{\text{Lex}} \beta$ if the left most non-zero entry of $\alpha - \beta$ is positive.

**Example 4. (Graded Lex)** Given $\alpha, \beta \in \mathbb{N}^n$ we say that $\alpha >_{\text{GrLex}} \beta$ if $|\alpha| > |\beta|$ or if $|\alpha| = |\beta|$ then $\alpha >_{\text{Lex}} \beta$.

**Exercise 2.** Prove that Lex and Graded Lex orderings are Monomial Orderings.

**Definition 24.** *Let $f(x) = \sum_\alpha a_\alpha x^\alpha \in \mathbb{F}[x_1, \ldots, x_n]$ and $>$ be a monomial ordering on $x_i, 1 \leqslant i \leqslant n$ and $f \neq 0$. Then we define the following:*

*i.* $\text{multideg}(f) = max \ \{\alpha \in \mathbb{N}^n | a_\alpha \neq 0\}$

*ii.* $\text{leadingCoefficient}(f) = a_{\text{multideg}(f)}$

*iii.* $\text{leadingMonomial}(f) = x^{\text{multideg}(f)}$

*iv.* $\text{leadingTerm}(f) = a_{\text{multideg}(f)} x^{\text{multideg}(f)}$

[1] Grobner is Buchberger's Thesis advisor..

**Proposition 2.** *Given $f, g \in \mathbb{F}[x_1, \ldots, x_n]$ then we have the following properties:*

*i.* $\operatorname{multideg}(fg) = \operatorname{multideg}(f) + \operatorname{multideg}(g)$

*ii.* $f + g \neq 0 \Rightarrow \operatorname{multideg}(f + g) \leqslant \max\,(\operatorname{multideg}(f), \operatorname{multideg}(g))$ *and if* $\operatorname{multideg}(f) \neq \operatorname{multideg}(g)$ *then equalitiy holds, however this is not a necessary condition.*

## 10.2 Division algorithm in $\mathbb{F}[x_1, \ldots, x_n]$

In this we primarily aim at investigating membership of $f \in \mathbb{F}[x_1, \ldots, x_n]$ in an ideal generated by some elements of the ring. More precisely, given $f_1, \ldots f_s \in \mathbb{F}[x_1, \ldots, x_n]$ we would like to know if there exist $a_i, 1 \leqslant i \leqslant s$ such that $f = \sum_{1 \leqslant i \leqslant s} a_i f_i + r$ where $r, a_i \in \mathbb{F}[x_1, \ldots, x_n]$.

**Note 20.** In such a decomposition of $f$ shown above the coefficients of $f_i$ generally depend on the chosen Monomial ordering.

**Theorem 23.** *Let $(f_1, \ldots, f_s)$ be an ordered tuple of polynomials in $\mathbb{F}[x_1, \ldots, x_n]$ then every $f \in \mathbb{F}[x_1, \ldots, x_n]$ can be written as $f = \sum_{1 \leqslant i \leqslant s} a_i f_i + r$, where $a_i, r \in \mathbb{F}[x_1, \ldots, x_n]$ and either $r = 0$ or $r$ is such that it is divisible by none of* $\operatorname{leadingTerm}(f_i)$.

**Note 21.** $r = 0$ is sufficient to show that $f \in <f_1, \ldots, f_s>$ but not necessary. This is evident from the following example.

**Example 5.** Let $f(x, y) = xy^2 - x$ and $f_1(x, y) = xy + 1$ and $f_2(x, y) = y^2 - 1$ considering the monomial ordering as $>_{\text{Lex}}$ we have $xy^2 - x = y(xy + 1) + 0(y^2 - 1) + (-x - y)$ if we considering ordering on $f_i$ as $f_1 > f_2$, else we get $xy^2 - x = x(y^2 - 1) + 0(xy + 1) + 0$.

## 10.3 Hilbert's Nullstellensatz

**Definition 25. (Affine Variety)** *If $f_i \in \mathbb{F}[x_1, \ldots, x_n]$ for $1 \leqslant i \leqslant s$ then the set of common solutions of $f_i$ is called an Affine Variety. More precisely, an Affine Variety is defined as $V(f_1, \ldots, f_s) = \{(a_1, \ldots, a_n) \in \mathbb{F}^n \colon f_i(a_1, \ldots, a_n) = 0, \forall 1 \leqslant i \leqslant s\}$*

**Example 6.** $V(x^2 + y^2 - 4)$ is a circle, $V(x^2 + y^2 + 1) = \phi$ etc.,.

**Proposition 3.** *If $V, W$ are affine varieties then so are $V \bigcap W$ and $V \bigcup W$.*

**Proof.** If $V = V(f_1, \ldots, f_s)$ and $W = V(g_1, \ldots, g_t)$ then $V \bigcap W = V(\{f_i g_j\})$ and $V \bigcup W = V(\{f_i\} \bigcup \{g_j\})$. □

An affine variety may be an infinite set, so compact representation of affine variety may be useful in some cases. To that goal we aim to describe the set concisely using some parameters. We call this parametrization.

**Example 7.** $V(x^2 + y^2 - 1) = \{(\cos(t), \sin(t)) \colon t \geqslant 0\} = \left\{\left(\frac{1 - t^2}{1 + t^2}, \frac{2t}{1 + t^2}\right)\right\} \bigcup \{(-1, 0)\}$. In this case $t$ is the parameter. And the second representation is called **Rational Parametrization**.

Suppose a parametric representation of an affine variety is given, can we find the defining polynomials? This requires systematic elimination of parameters, studied as Elimination Theory.

**Notation 7.** *If $f_i \in \mathbb{F}[x_1, \ldots, x_n]$ are elements of the multinomial ring, then the ideal generated by $f_i$ denoted as $<f_1, \ldots f_s>$.*

**Definition 26.** *If $I \subset \mathbb{F}[x_1, \ldots x_n]$ is an ideal and $I = <f_1, \ldots f_s>$ then $I$ is said to be finitely generated by $\{f_1, \ldots f_s\}$ and is called a basis of $I$.*

**Proposition 4.** *If $<f_1,...,f_s>=<g_1,...g_s>$ then $V(f_1,...f_s)=V(g_1,...g_s)$*

**Proof.** Work out! □

**Definition 27.** *Let $V$ be an affine variety, then $I(V)=\{f\colon f(a_1,...a_n)=0,\forall(a_1,...a_n)\in V\}$*

**Proposition 5.** *$I(V)$ is an ideal in $\mathbb{F}[x_1,...x_n]$.*

**Lemma 23.** *If $f_1,...f_s\in\mathbb{F}[x_1,...,x_n]$ then $<f_1,...f_s>\subset I(V(f_1,...,f_s))$ and equality need not occur.*

**Proof.** □

**Proposition 6.** *Let $V,W$ be affine varieties, then*

*i.* $V\subset W\Leftrightarrow I(V)\supset I(W)$

*ii.* $V=W\Leftrightarrow I(V)=I(W)$

**Proof.** □

**Theorem 24. (Weak Nullstellensatz)** *Let $\mathbb{F}$ be an algebraically closed field and let $\mathcal{I}$ be an ideal of $\mathbb{F}[x_1,...,x_n]$ satisfying $V(\mathcal{I})=\phi$, then $\mathcal{I}=\mathbb{F}[x_1,...,x_n]$.*

**Theorem 25. (Hilbert's Nullstellensatz)** *Let $\mathbb{F}$ be an algebracially closed field. If $f_1$, ..., $f_s\in\mathbb{F}[x_1,...,x_n]$ are such that $f\in I(V(f_1,...,f_s))$ then there is an $m\geqslant 1$ such that $f^m\in<f_1,...,f_s>$.*

**Definition 28.** *If $\mathcal{I}$ is an ideal, then $\sqrt{\mathcal{I}}=\{f\colon\exists m,f^m\in\mathcal{I}\}$*

**Definition 29. (Radical Ideal)** *An ideal $I$ such that if $f^m\in I\Rightarrow f\in I$ is called a Radical Ideal.*

**Proposition 7.** *If $I$ is an ideal then $\sqrt{I}$ is a Radical ideal.*

**Theorem 26. (Strong Nullstellensatz)** *Let $\mathbb{F}$ be an algebraically closed field. If $I$ is an ideal of $\mathbb{F}[x_1,...,x_n]$ then $I(V(I))=\sqrt{I}$*

**Definition 30.** *An ideal $I\subset\mathbb{F}[x_1,...,x_n]$ is a monomial ideal if $I=<x^\alpha\colon\alpha\in A>$ and $A\subset\mathbb{N}^n$ and $A$ need not be finite.*

**Exercise 3.** Verify that not all ideals are monomial ideals. (hint: $\{x^4y^2,x^3y^4,x^2y^5\}$)

**Lemma 24.** *Let $I=<x^\alpha\colon\alpha\in A>$ be a monomial ideal, then a monomial $x^\beta\in I\Leftrightarrow x^\alpha|x^\beta,\alpha\in A$.*

**Proof.** □

**Definition 31.** *Let $I\neq\{0\}$ be an ideal then the set of leading terms of elements of $I$ denoted by* $\mathrm{LT}(I)=\{\mathrm{LT}(f)\colon f\in I\}$

**Note 22.** $\mathrm{LT}(I)$ above is a monomial ideal.

**Note 23.** $<\mathrm{LT}(I)>$ clearly depends on monomial ordering.

**Note 24.** $<\mathrm{LT}(I)>=<\mathrm{LM}(I)>$

**Lemma 25.** *Let $I$ be a monomial ideal and $f\in\mathbb{F}[x_1,...,x_n]$ then the following are equivalent:*

*i.* $f\in I$

*ii. every term of $f$ is in $I$*

*iii. $f$ is an $\mathbb{F}$ linear combination of monomials of $I$.*

**Proof.** □

**Corollary 2.** *Two monomial ideals are equal* $\Leftrightarrow$ *they have the same monomials.*

**Theorem 27. (Dickson's Lemma)** *Let* $I = <x^{\alpha}, \alpha \in A>$ *be a monomial ideal, then* $I = <x^{\alpha_1}, \ldots, x^{\alpha_s}>$ *for* $s \geqslant 1$ *and* $\alpha_1, \ldots, \alpha_s \in A$.

**Proof.** □

**Corollary 3.** *Let* $>$ *be a linear order on* $\mathbb{Z}^n_{\geqslant 0}$ *satisfying the following condition: if* $\alpha > \beta$ *then* $\alpha + \gamma > \beta + \gamma$ *then* $>$ *is a well ordering* $\Leftrightarrow$ $\alpha \geqslant 0_n$ $\forall \alpha \in \mathbb{Z}^n_{\geqslant 0}$.

**Proof.** □

**Proposition 8.** *Suppose I is an ideal, then*

*a)* $<\mathrm{LT}(I)>$ *is a monomial ideal.*

*b) There are* $g_1, \ldots, g_s$ *such that* $<\mathrm{LT}(I)> = <\mathrm{LT}(g_1), \ldots, \mathrm{LT}(g_s)>$

**Proof.** □

**Note 25.** $g_1, \ldots, g_s$ form a basis for $I$ and has special properties. Such a basis for $I$ is called Grobner Basis for $I$. We shall study further properties of such a basis in the following section.

**Theorem 28. (Hilbert Basis Theorem)** *Every ideal I in* $\mathbb{F}[x_1, \ldots, x_n]$ *is finitely generated i.e.* $I = <g_1, \ldots, g_s>$ *for some* $g_i \in I$.

## 10.4 Grobner Basis

**Definition 32. (Grobner Basis)** *Fix a monomial order. A finite subset* $G = \{g_1, \ldots, g_s\}$ *of an ideal* $I \neq \{0\}$ *is said to be Grobner basis if* $<\mathrm{LT}(g_1), \ldots, \mathrm{LT}(g_n)> = <\mathrm{LT}(I)>$.

**Corollary 4.** *Fix a monomial order, then every ideal* $I \neq \phi$ *has a Grobner basis. Further, any Grobner basis for an ideal I is a basis for I.*

**Proposition 9.** $V(I)$ *is an affine variety. In particular if* $I = <f_1, \ldots, f_s>$ *then* $V(I) = V(f_1, \ldots, f_s)$

### 10.4.1 Properties of Grobner Basis

**Proposition 10.** *Let* $G = \{g_1, \ldots, g_s\}$ *be a* $G-$*basis for ideal I and let* $f \in \mathbb{F}[x_1, \ldots, x_n]$. *Then there is a unique* $r \in \mathbb{F}[x_1, \ldots, x_n]$ *s.t*

*a) No terms of r is divisible by any of* $\mathrm{LT}(g_1), \ldots, \mathrm{LT}(g_s)$

*b) There is a* $g \in I$ *s.t* $f = g + r$.

*In particular, r is the reminder when f is divided by G no matter how the elements of G are ordered.*

**Proof.** □

**Definition 33.** *Let* $S = (f_1, \ldots f_s)$ *be a tpi[;e pf distinct polynomials and* $f \in \mathbb{F}[x_1, \ldots, x_n]$ *then* $f^s$ *is the reminder when f is divided by S.*

**Corollary 5.** *For a* $G-$*basis, we can ignore the ordering of the elements of G during division.*

**Definition 34.** *Suppose* $f, g \neq 0$

*i. If multideg*$(f) = \alpha$ *and multideg*$(g) = \beta$ *then define* $\gamma = (\gamma_1, \ldots, \gamma_n)$ *where* $\gamma_i = \max(\alpha_i, \beta_i)$ *then* $x^{\gamma} = \mathrm{LCM}(x^{\alpha}, x^{\beta}) = \mathrm{LCM}(\mathrm{LM}(f), \mathrm{LM}(g))$.

*ii.* $S-$*polynomial of* $f, g$ *denoted* $S(f, g) = \frac{x^{\gamma}}{\mathrm{LT}(f)} f - \frac{x^{\gamma}}{\mathrm{LT}(g)} g$

**Note 26.** $S-$polynomials cancel out the leading terms.

**Lemma 26.** *Let $f_1, \ldots, f_s$ be such that multidegree$(f_i) = \delta$. Define $g = \sum_{i=1}^{s} c_i f_i$ and suppose multidegree$(g) < \delta$. Then $g = \sum c_{ij}' S(f_i . f_j)$ where $c_{ij}' \in \mathbb{F}$ and each multidegree$(S(f_i, f_j)) < \delta$ cancellation of the leading term in $\sum c_i f_i$ are accounted for by the S polynomials.*

**Theorem 29. (Buchberger Criterion)** *Let $I$ be an ideal and $G = \{g_1, \ldots g_s\}$ be a basis for $I$. Then $G$ is a Grobner basis $\Leftrightarrow$ $S(g_i, g_j) \operatorname{mod} G = 0$ $\forall i, j$ and $G$ taken in any order.*

---

**Algorithm (Buchberger's Algorithm)**

---

**Lemma 27.** *Let $G$ be a $G-$base for $I$. Let $p \in G$ be such that $\mathrm{LT}(p) \in <\mathrm{LT}(G - \{p\})>$ then $<\mathrm{LT}(G - \{p\})> = <\mathrm{LT}(I)>$.*

**Definition 35. (Minimal G-basis)**

- $\mathrm{LC}(p) = 1$ *for all* $p \in G$
- *For all* $p \in G$, $\mathrm{LT}(p) \in <\mathrm{LT}(G - \{p\})>$.

**Proposition 11.** *If $G, \tilde{G}$ are two minimal $G-$basis of $I$ then $\mathrm{LT}(G) = \mathrm{LT}(\tilde{G})$.*

**Definition 36.** *Reduced $G-$basis for ideal $I$*

- $\mathrm{LC}(p) = 1$, $p \in G$
- *$\forall p \in G$,no monomial of $p$ lies in $<\mathrm{LT}(G - \{p\})>$.*

**Theorem 30.** *Let $I = \{0\}$ and fix a monomial order. Then $I$ has a unique reduced $G-$basis.*

**Proof.** Start with a minimal $G-$ basis for $I$. Let $g \in G$ and let $g' = g \bmod (G - \{g\})$. Define $G' = (G - \{g\}) \bigcup \{g'\}$. $\mathrm{LT}(g) = \mathrm{LT}(g') \Rightarrow \mathrm{LT}(G) = \mathrm{LT}(G') \Rightarrow G'$ is also minmal basis.

Now we shall argue for **Uniqueness.** Suppose $G, G'$ are two reduced $G-$basis. Being reduced they are minimal and hence $\mathrm{LT}(G) = \mathrm{LT}(G')$. So for $g \in G$, there is $g' \in G'$ such that $\mathrm{LT}(g) = \mathrm{LT}(g')$. $g - g' \in I$, so $(g - g') \equiv 0 (\operatorname{mod} G)$ and also $(g - g') = 0 (\operatorname{mod} G')$. In $g - g'$ the leading terms of $g$, $g'$ cancel each other and another monomial of $g, g'$ is not divisible by any of the leading terms of $G$, so $(g - g') \operatorname{mod} G = 0$. Forcing $g = g'$. □

**Note 27.** Reduced Grobner basis is useful for Equality testing of ideals. Given two ideals $< f_!, \ldots f_n >$, $< g_1, \ldots, g_m >$ are equal iff they have the same Reduced Grobner basis.

**Definition 37.** *Let $I = <f_1, \ldots, f_s>$ and ideal of $\mathbb{F}[x_1, \ldots, x_n]$. The $l^{\text{th}}$ elimination ideal $I_l$ is defined to be $I_l = I \bigcap \mathbb{F}[x_1, \ldots, x_n]$.*

**Theorem 31. (Elmination)** *Let $I \subset \mathbb{F}[x_1, \ldots, x_n]$ to be an ideal and let $G$ be a $G-$basis for $I$ w.r.t $>_{\text{lex}}$ with $x_1 > \ldots > x_n$. Then for every $0 \leqslant l \leqslant n-1$, the set $G_l = G \bigcap \mathbb{F}[x_{l+1}, \ldots, x_n]$ is a $G-$basis for $I_l$.*

**Proof.** For any $f \in I_l$, there is a $g \in G_l$, such that $\mathrm{LT}(g) | \mathrm{LT}(f)$. Since $I_l \subset I$, $f \in I$. Now $G$ is a $G-$basis for $I$, so there is some $g \in G$, such that $\mathrm{LT}(g) | \mathrm{LT}(f)$. Since $f \in I_l$, $\mathrm{LT}(f)$ does not depend on $x_1, \ldots, x_l$. Also since $\mathrm{LT}(g) | \mathrm{LT}(f)$, $\mathrm{LT}(g)$ does not depend on $x_1, \ldots, x_l$. Because $>_{\text{lex}}$ is used, the other terms of $g$ also cannot depend on $x_1, \ldots, x_l$. So $g \in G_l$. □

**Note 28. Extension Step.** $V(I) = \{(a_1, \ldots, a_n) \in \mathbb{F}^n$: $f(a_1, \ldots, a_n) = 0, \forall f \in I\}$ then $(a_{l+1}, \ldots, a_n) \in V(I_l)$ is a partial solution. Suppose $I_{l-1} = <g_1, \ldots, g_s>$, then we want to solve $x_l$ in the equations, $g_1(x_l, \ldots, x_n) = \ldots = g_s(x_l, \ldots, x_n) = 0$. Natural approach to solve for $x_l$ in the gcd of the uni-variate polynomials in $x_l$.

**Theorem 32.** **(Extension)** *Let $I = < f_1, ..., f_s >$ be an ideal of $\mathbb{C}[x_1, ..., x_n]$ and $I_1$ be its first elimination ideal, for $1 \leqslant i \leqslant s$ let $f_i = g_i(x_2, ..., x_n)x_1^{N_i} + \text{terms in } x_i \text{ having deg } < N_i$ with $g_i \neq 0$. Suppose we have a partial solution $(a_2, ..., a_n) \in V(I_1)$. If $(a_2, ..., a_n) \in V(g_1, ..., g_s)$ then there is an $a \in \mathbb{C}$, such that $(a_1, ..., a_n) \in V(I)$. The condition $(a_2, ..., a_n) \notin V(g_1, ..., g_s)$ is equivalent to saying that $f_1, ..., f_s$ do not simultaniously vanish at $(a_2, ..., a_l)$.*

# 11 Lattices

Major references for this series of lectures are:

i. Lecture notes by Oded Regev.

ii. Lecture notes by Daniel Micercineio.

## 11.1 Introduction

Lattice is a periodic arrangement of points in $\mathbb{R}^n$. It can be considered geometry of integers.

**Definition 38.** *Given n linearly independent vectors $B = \{ b_1, ..., b_n\}$ in $\mathbb{R}^m$ as a vector space $[\mathbb{R}^m: R]$ the lattice generated by them is defined as $\mathcal{L}(B) = \{\sum x_i b_i : x_i \in \mathbb{Z}\}$. The set $B$ is called a basis for lattice $\mathcal{L}(B)$.*

**Note 29.** It is easy to note that a lattice is a discrete addtive sub group of $\mathbb{R}^n$. However not all discrete additive subgroups are lattices. Lattices are not dense.

**Note 30.** In the above definition if $m = n$ then the lattice is called full rank lattice.

**Definition 39.** **(Span)** $\text{Span}(\mathcal{L}(B)) = \{By: y \in \mathbb{R}^n\}$ *is a subspace of* $\mathbb{R}^n$.

**Definition 40.** **(Fundamental Parallopiped)** $\mathcal{P}(B) = \{Bx: x \in \mathbb{R}^n, 0 \leqslant x_i < 1\}$ *is called fundamental parallopiped.*

**Lemma 28.** *Let $\Delta$ be a lattice of rank n and let $B = \{b_1, ..., b_n\} \in \Delta$ be linearly independent, then $b_1, ..., b_n$ is a basis $\Leftrightarrow \mathcal{P}(B) \bigcap \Delta = \{0\}$.*

**Proof.** Suppose $b_1, ..., b_n \in \Delta$ is a basis, consider $\sum x_i b_i \in \mathcal{P}(B)$ so that $0 \leqslant x_i < 1$. If $y \in \Delta$, then $y$ is an integer combination of $b_1, ..., b_n$ i.e. $y = \sum y_i b_i$. Then $\sum x_i b_i = \sum y_i b_i \Rightarrow \sum (x_i - y_i) b_i = 0 \Rightarrow x_i = y_i$.

Arguing the other way, suppose $\mathcal{P}(B) \bigcap \Delta = \{0\}$. Suppose, $Bx \in \Delta$ for some $x \in \mathbb{R}^n$, we have to argue that $x$ is a vector of integers, consider $\lfloor x \rfloor = (\lfloor x_1 \rfloor, ..., \lfloor x_n \rfloor)$ then $B(\lfloor x \rfloor) \in \Delta$, but then also $B(x - \lfloor x \rfloor) \in \Delta$ which is a contradiction. □

**Definition 41.** *An $n \times n$ matrix $U$ with integer entries is said to be unimodular if* $\det(U) = \pm 1$.

**Lemma 29.** *Two basis $B_1, B_2$ for a given lattice are equivalent $\Leftrightarrow B_2 = B_1 U$ for some unimodular matrix $U$.*

**Lemma 30.** *A matrix is unimodular $\Leftrightarrow$ it is a product of sequence of elementary column matrices.*

**Lemma 31.** *Two bases are equivalent $\Leftrightarrow$ one can be obtained from the other using sequence of elementary column operations.*

**Definition 42.** *Let $\Delta = \mathcal{L}(B)$ be a lattice of rank n. Then* $\det(\Delta)$ *is the volume of $\mathcal{P}(B)$ ie.e* $\det(\Delta) = \sqrt{(\det(B^T B))}$. *If $\Delta$ is a full rank then* $\det(\Delta) = |\det(B)|$.

**Proposition 12.** $\det(\Delta)$ *is independent of basis.*

**Note 31.** $\det(\Delta)$ is inversely proporitional to the density of the lattice. If one takes a large ball $K$, then the number of lattice points in $K$ approxmately $\frac{\text{Vol}(K)}{\det(\Delta)}$.

## 11.2 Some Important problems in Lattices

**Shortest Vector Problem.** Given basis $B$ of a lattice, find a vector of shortest possible length in $\mathcal{L}(B)$.

**Closest Vector Problem.** Given a basis $B$ and a vector $y \in \mathbb{R}^n$, find $x \in \mathcal{L}(B)$ s.t. $d(x, y) = \|x - y\|$ is the minimum possible.

**Successive Minima.** For $i \geqslant 1$, $\lambda_i(\Delta) = \inf \{r \mid B(0, r) \text{ contains atleast } i \text{ linearly ind.vectors}\}$

**Theorem 33.** *Let $B$ be basis of a rank $n$ lattice and $B' = \{b_i'\}$ be its Gram schmidt orthogonalization. Then $\lambda_1(B) \geqslant \min_{1 \leqslant i \leqslant n} (\|b_i'\|) > 0$.*

**Proof.** Let $x \in \mathbb{Z}^n$, we show $\|Bx\| \geqslant \min (\|b_i'\|) > 0$.

Let $j$ be largest s.t. $x_j \neq 0$.

$|<Bx, b_j'>| = |<\sum x_i b_i, b_j'| = |x_i| \|b_j'\|^2$ (1)

By Cauchy schwartz inequality we have $|<Bx, b_j'>| \leqslant \|Bx\| \| \|b_j'\| \|$. (2)

From both the above we have: $\|Bx\| \geqslant |x_j| \|b_j'\| \geqslant \|b_j'\| > \min (\{\|b_j\|\})$. □

**Theorem 34.** *Let $\Delta$ be a lttice. Then there is some $\varepsilon > 0$ such that for any unequal $z_1, z_2 \in \Delta$ $\|z_1 - z_2\| \geqslant \varepsilon$.*

**Corollary 6.** *Successive minima are achieved.*

## 11.3 Minkowski's Theorems

**Theorem 35. (Blitchfeldt)** *For any full rank lattice $\Delta$ and measurable set $S \subset \mathbb{R}^n$ with $\operatorname{Vol}(S) > \det (\Delta)$, there exists two points $z_1, z_2 \in S$ such that $z_1 \neq z_2$ and $z_1 - z_2 \in \Delta$.*

**Proof.** Define $S_x = S \bigcap (x + \mathcal{P}(\Delta))$. $\sum_x \operatorname{Vol}(S_x) = \operatorname{Vol}(S)$.

We define $S_x' = S_x - \{x\} \subset \mathcal{P}(\Delta)$. Then we have $\sum_x \operatorname{Vol}(S_x') = \sum_x \operatorname{Vol}(S_x) > \det (\Delta) = \operatorname{Vol}(\mathcal{P}(\Delta))$.

So, there has to be $x \neq y$ such that $S_x' \bigcap S_y' \neq \phi$ and $x, y \in \Delta$, let $z \in S_x' \bigcap S_y'$. So, there are $w_1$, $w_2$ such that $w_1 \in S_x$ and $w_2 \in S_y$ and $z = w_1 - x$ and $z = w_2 - y$. So $w_1, w_2 \in S$, $w_1 - x = w_2 - y$

$\Rightarrow w_1 - w_2 = x - y \in \Delta$. □

**Theorem 36. (Minkowski's Convex Body Theorem)** *Let $\Delta$ be a full rank lattice of rank $n$. Then for any centrally symmetric convex sets $S$, if $\operatorname{Vol}(S) > 2^n \det(\Delta)$, then $S$ contains a non-zero lattice point.*

**Proof.** Define $\hat{S} = \frac{S}{2} = \{x: 2x \in S\}$. Then $\operatorname{Vol}\big(\hat{S}\big) = 2^{-n} \operatorname{Vol}(S) > \det (\Delta)$. So, there are $z_1, z_2 \in \hat{S}$, $z_1 \neq z_2$ s.t $z_1 - z_2 \in \Delta \Rightarrow 2z_1, 2z_2 \in S$ and $-2z_2 \in S$ and $2z_1 - 2z_2 \in S$. Choose $\lambda = \frac{1}{2}$ so by convexity of $S$, $z_1 - z_2 \in S$. □

**Proposition 13.** *The volume of an $n -$ dimensional ball of radius $r$ is atmost $\left(\frac{2r}{\sqrt{n}}\right)^n$.*

**Proof.** $B(0, r)$ contains a cube of side $\left(\frac{2r}{\sqrt{n}}\right)$. □

**Theorem 37. (Minkowski's First Theorem)** *For any full rank lattice $\Delta$ of rank $n$, $\lambda_1(\Delta) \leqslant \sqrt{n} (\det (\Delta))^{\frac{1}{n}}$.*

**Proof.** By defn. of open ball $B(0, \lambda_1)$ does not contain any non-zero lattice point. So $\left(\frac{2\lambda_1}{\sqrt{n}}\right)^n \leqslant \operatorname{Vol}(B(0, \lambda_1)) \leqslant 2^n \det(\Delta)$. Hence the result follows. □

**Theorem 38. (Minkowski's Second Theorem)** *For any full rank lattice $\Delta$ of rank $n$ we have $\left(\prod_{i=1}^n \lambda_i(\Delta)\right)^{\frac{1}{n}} \leqslant \sqrt{n} (\det (\Delta))^{\frac{1}{n}}$.*

**Proof.** Let $x_1, \ldots, x_n \in \Delta$ be linearly independent vectors achieving successive minima, i.e. $\|x_i\| = \lambda_i$. Let $x_1', \ldots, x_n'$ be their Gram Schmidt Orthogonalization. Consider the open ellipsoid $T$, with axes $x_1', \ldots, x_n'$ and lengths $\lambda_1, \ldots, \lambda_n$ and $T = \left\{ y \in \mathbb{R}^n : \sum_{i=1}^n \left( \frac{<y, x_i'>}{\|x_i'\| \lambda_i} \right)^2 < 1 \right\}$. Expressing $y$ in terms of $x_i'$ as $\sum r_i x_i'$. Then $\left( \frac{<y_i, x_i'>}{\|x_i'\| \lambda_i} \right)^2 = \left( \frac{<\sum y_i x_i', x_i'>}{\|x_i'\| \lambda_i} \right)^2 = \left( \frac{y_i \|x_i\|^2}{\|x_i\| \lambda_i} \right)^2 = \frac{y_i^2}{\lambda_i^2} \|x_i'\|^2$.

So $\mathrm{Vol}(T) = (\prod_{i=1}^n \lambda_i) \mathrm{Vol}(B(0,1)) \geqslant \prod_{i=1}^n \lambda_i \left( \frac{2}{\sqrt{n}} \right)^n$.

We claim that $T$ does not contain non-zero lattice point if $0 \neq y \in \Delta$, Let $K$ be maxmimal s.t. $\|y\| > \lambda_k$ and $\|y\| < \lambda_{k+1}$. We have $\mathrm{Span}(x_1, \ldots, x_k) = \mathrm{Span}(x_1', \ldots, x_k')$, $y \in \mathrm{Span}(x_1, \ldots, x_k)$ as otherwise $x_1, \ldots, x_k, y$ are $k+1$ linearly independent vectors of length $< \lambda_{k+1}$, which violates the definition of $\lambda_{k+1}$. So $y \in \mathrm{Span}(x_1', \ldots, x_k')$.

Consider $\sum_{i=1}^n \left( \frac{<y, x_i'>}{\|x_i\| \lambda_i} \right)^2 = \sum_{i=1}^k \left( \frac{<y, x_i'>}{\|x_i\| \lambda_i} \right) \geqslant \frac{1}{\lambda_k^2} \sum_i^k \left( \frac{<y, x_i'>}{\|x_i\|} \right)^2 = \|y\|^2$.

So, $\mathrm{Vol}(T) = \prod \lambda_i \mathrm{Vol}(B(0,1)) \geqslant \prod \lambda_i \left( \frac{2}{\sqrt{n}} \right)^n$. Hence, it follows that $\mathrm{Vol}(T) \leqslant 2^n \det(\Delta)$. □

## 11.4 Dual Lattices

**Definition 43.** *$\Delta$ be a full rank lattice. Dual of this lattice denoted by $\Delta^*$ is defined as the set $\{y \in \mathbb{R}^n : <x, y> \in \mathbb{Z}, \forall x \in \Delta\}$.*

**Note 32.** for a non full rank lattice $\Delta$ we define $\Delta^* = \{y \in \mathrm{Span}(\Delta) : <x, y> \in \mathbb{Z}, \forall x \in \Delta\}$.

Fix an $x \in \Delta$, say $x = (x_1, \ldots, x_n)$. then the points in the dual satisfy $\sum y_i x_i \in \mathbb{Z}$. So we have sheets of hyperplanes paraterized on the integers. Each of these hyperplanes are separated by a distance of $\frac{1}{\|x\|}$.

**Definition 44.** *Let $B_{m \times n}$ be a basis, dual basis $D = (d_1, \ldots, d_n)$ is defined as the unique basis such that :*

*i.* $\mathrm{Span}(D) = \mathrm{Span}(B)$

*ii.* $B^T D = I_n$

**Note 33.** If $m = n$ we have $D = (B^T)^{-1}$ and if $n < m$ then $D = B(B^T B)^{-1}$.

**Proposition 14.** *If $D$ is a dual basis of $B$, then $\mathcal{L}(B)^* = \mathcal{L}(D)$. In particular $\mathcal{L}(B)^*$ is a lattice and $D$ is a basis for it.*

**Proof.** Claim: $\mathcal{L}(B)^* \subset \mathcal{L}(D)$. Let $y \in \mathcal{L}(B)^*$.

So, $y \in \mathrm{Span}(B) = \mathrm{Span}(D)$. $y = \sum a_i d_i$ for $a_i \in \mathbb{R}$. Now $<y, b_i> \in \mathbb{Z}$.

So $<\sum a_i d_i, b_j> = a_j \in \mathbb{Z} \Rightarrow y \in \mathcal{L}(D)$.

Arguing the otherway, Claim: $\mathcal{L}(D) \subset \mathcal{L}(B)^*$

Consider $d_j \in D$. Let $x \in \mathcal{L}(B)$, so $x = \sum a_i b_j$. Then $<d_j, \sum a_i b_i> = a_j \in \mathbb{Z} \Rightarrow d_j \in \mathcal{L}(B)$. With $a_i \in \mathbb{Z} \Rightarrow D \subset \mathcal{L}(B)^*$. Since $\mathcal{L}(B)^*$ is closed under addition, any integer combination of elements of $D$ is also in $\mathcal{L}(B)^* \Rightarrow \mathcal{L}(D) \subset \mathcal{L}(B)^*$. □

**Lemma 32.** *For any lattice $\Delta$, we have $(\Delta^*)^* = \Delta$.*

**Proof.** $B$ is a basis for $\Delta$, then $D = B(B^T B)^{-1}$ is a basis for $\Delta^*$ and $D(D^T D)^{-1}$ is a basis for $(\Delta^*)^*$. But on simplification we have $B = D(D^T D)^{-1}$. Hence the result. □

**Lemma 33.** $\det(\Delta^*) = \frac{1}{\det(\Delta)}$

**Proof.** $\det(\Delta^*) = \sqrt{\det(D^T D)}$ on further simplification gives $\frac{1}{\det(\Delta)}$. □

**Note 34.** Because of above lemma dual lattice is sometimes called reciprocal lattice.

**Lemma 34.** *For any rank $n$ lattice, $\lambda_1(\Delta)\lambda_2(\Delta^*) \leqslant n$.*

**Proof.** We have $\lambda_1(\Delta) \leqslant \sqrt{n}(\det(\Delta))^{\frac{1}{n}}$ and $\lambda_1(\Delta^*) \leqslant \sqrt{n}(\det(\Delta^*))^{\frac{1}{n}}$. Both together put we have the required result. □

**Lemma 35.** *For any rank $n$ lattice $\Delta$, $\lambda_1(\Delta) = \lambda_1(\Delta^*) \geqslant 1$*

## 11.5 LLL - Lenstra Lovasz Algorithm

This algorithm is a major achievement in Lattice theory, it is easy to note that given a basis for a lattice, if we consider the GSO of the lattice the the lattice generated by GSO is not identical to that original lattice in general. However, having almost orthogonal basis vectors will be helpful in various ways. In this section we formalize the notion of almost orthognonality and derive the LLL algorithm.

**Note 35.** Given a basis $\{b_1, ..., b_n\}$ for $\Delta$, let $\{b_1', ..., b_n'\}$ be its GSO then $b_i' = b_i - \sum_{j=1}^{i-1} (\mu_{ij} b_i')$ then $\mu_{ij} = \frac{<b_i, b_j'>}{<b_j', b_j'>}$.

**Definition 45.** *Let $\Delta$ be a full rank lattice with basis $B = \{b_1, ..., b_n\}$. Then $B$ is said to be $\delta - L^3 -$ reduced basis if:*

*a)* $|\mu_{ij}| \leqslant \frac{1}{2}$ *for* $1 \leqslant j < i \leqslant n$.

*b)* $|b_i'|^2 \geqslant (\delta - \mu_{ii-1}^2)|b_{i-1}'|^2$.

**Note 36.** In the following discusstion we shall assume $\delta = \frac{3}{4}$.

**Note 37.** We shall denote the GSO$(B) = B'$. Then we have $\det(B') = \det(B) = \det(\Delta)$. If $B$ is $L^3 -$ reduced we have $|\mu_{ij}| \leqslant \frac{1}{2}$.

**Theorem 39.** *Let $b_1, ..., b_n$ be an $L^3 -$ reduced basis of a lattice $\Delta$. Then*

*a)* $\det(\Delta) \leqslant \prod_{i=1}^n |b_i| \leqslant 2^{\frac{n(n-1)}{4}}(\det(\Delta))$

*b)* $|b_j| \leqslant 2^{\frac{i-1}{2}}|b_i'|$

*c)* $|b_1| \leqslant 2^{\frac{n-1}{4}}\det(\Delta)^{\frac{1}{n}}$

*d)* $\forall x \in L, x \neq 0$ *we have* $|b_1| \leqslant 2^{\frac{n-1}{2}}|x|$ *and hence* $|b_1| \leqslant 2^{\frac{n-1}{2}}\lambda_1(\Delta)$

*e) for any linearly independent vectors $x_1, ..., x_t \in \Delta$ we have $|b_j| \leqslant 2^{\frac{n-1}{2}}\max(|x_1|, ..., |x_\tau|)$, $\forall 1 \leqslant j \leqslant t$.*

**Proof.**

a) $\det(B)^2 = \det(B')^2 = \det(B')\det\left(B'^T\right) = \det\left(BB'^T\right) = \text{diag}\,[\|b_1'\|^2, ..., \|b_n'\|^2] = \prod_{i=1}^n |b_i'|^2 \leqslant \prod |b_i|^2$. (Because we have $|b_i|^2 = |b_i'|^2 + \sum_{1 \leqslant i \leqslant j} (\mu_{ij}^2 \|b_j'\|^2)$.

Since the $b_i$ are $L^3 -$ reduced $<b_i', b_i'> \geqslant \left(\frac{3}{4} - \mu_{ii-1}^2\right) < b_{i-1}', b_{i-1}' > \geqslant \frac{\|b_{i-1}'\|^2}{2}$.

So we have $|b_i'|^2 \leqslant 2^{i-j}|b_i|^2$. And $|b_i|^2 = <b_i, b_i> = <b_i', b_i'> + \sum_{1 \leqslant j \leqslant i} \mu_{ij}^2 < b_i', b_j' >$

$\leqslant |b_i|^2\left(1 + \frac{1}{4}(2^{i-1}...)\right) = |b_i'|^2\left(\frac{2^{i-1}+1}{2}\right)$

b) $|b_j|^2 \leqslant \left(\frac{2^{j-1}+1}{2}\right)|b_j'|^2 \leqslant \left(\frac{2^{j-1}+1}{2}\right)|b_j'|^2 \leqslant \left(\frac{2^{j-1}+1}{2}\right)2^{i-j}|b_i'|^2 \leqslant 2^{i-1}|b_i'|^2$.

c)

d) for every $x \in L$, $x \neq 0$ $|b_i| \leqslant 2^{\frac{n-1}{2}}|x|$. Hence we have $|b_1| \leqslant 2^{\frac{n-1}{2}}\lambda_1(\Delta)$. Let $x \in \Delta$ then $x = \sum_{1 \leqslant j < i} r_j b_j$ and $r_i \neq 0$. Hence the result. □

**Algorithm ($L^3$ – Algorithm)**

**Input:** Given basis $\{b_1, ..., b_n\}$
**Output:** $L^3$ – reduced basis $\{b_1, ..., b_n\}$

```
1. Suppose $b_1, ..., b_{k-1}$ is $L^3$ – reduced (initally true for $k = 2$)
2. Compute $\mu_{k\,k-1} = \frac{<b_k, b'_{k-1}>}{<b'_{k-1}, b'_{k-1}>}$
3. If $\left(|\mu_{k\,k-1}| > \frac{1}{2}\right)$
       $q \leftarrow \lfloor \mu_{k\,k-1} \rfloor$
       $b_k \leftarrow b_k - q b_{k-1}$
4. If $\left(|b'_k|^2 \geqslant \left(\frac{3}{4} - \mu_{k\,k-1}^2\right)|b_{k-1}|^2\right)$
       for $j \leftarrow k-2$ to 1
           compute $\mu_{kj}$
           if $\left(|\mu_{kj}| > \frac{1}{2}\right)$
               $q \leftarrow |\mu_{kj}|$
               $b_k = b_k - qb$
   else
       swap $b_k, b_{k-1}$
```

## 11.6 Nearest Plane Algorithm

## 11.7 Some applications of Lattices to Cryptography

RSA is used with low public exponent, suppose $e = 3$ and there are 3 users whose RSA modulii are $N_1, N_2, N_3$. Suppose a message $m$ is to be sent to 3 users.

$$\begin{aligned} y_1 &\equiv m^3 (\operatorname{mod} N_1) \\ y_2 &\equiv m^3 (\operatorname{mod} N_2) \\ y_3 &\equiv m^3 (\operatorname{mod} N_3) \end{aligned}$$

Then Chinese Reminder Theorem gives us the solution for $m$ uniquely. Hence, it is not secure.

Suppose $g_1, ..., g_k$ be polynomials of degree atmost $d$, where $g_i \in \mathbb{Z}_{N_i}[x]$. Where $N_1, ..., N_k$ are different modulli. Suppose there is a unique $m$, such that $m < \min\{N_i\}$ and $g_i(M) = C_i(\operatorname{mod} N_i)$. If $k \geqslant d$; then given the tuples $(g_i, C_i, N_i)_{i=1}^k$, it is possible to find $M$.

Let $h_i(M) = g_i(M) - c_i$. Then $h_i(M) = 0(\text{mod } N_i)$. Consider the system of polynomial equations $h_i(x) = 0(\text{mod } N_i)$. Using Chinese Reminder Theorem, we get the polynomail $h(x)$ such that $h(x) = 0$ (mod $N_1...N_k$). So, we are looking for a root of a polynomial of degree $d$. $M < \min\ (N_i) \leqslant (N_1...N_k)^{\frac{1}{k}} \leqslant (N_1...N_k)^{\frac{1}{d}}$. Let $N = N_1...N_k$. We are looking for a root $M$ of a polynomial of degree $d$ to the system $h(x) \equiv 0(\operatorname{mod} N)$ where $M < N^{\frac{1}{d}}$.

**Coppersmith's idea.** $f(x) \equiv 0(\operatorname{mod} N)$. Let $f(x) = x^d + a_{d-1}x^{d-1} + ... + a_1 x + a_0$. Suppose there is a $B$ such that $|a_i B^i| < \frac{N}{d+1}$ $(*)$. $|f(B)| < N$. Then a root $M$ of $f(x) \operatorname{mod} M$ with $M < B$ is a root of $f(x)$ over the integers. Finding all the integer roots of $f(x)$ will yield $M$.

In general $(*)$ will not hold. Then comes the crutial idea: is it possible to get polynomial $g(x)$ which has all the roots of $f(x)$ (and may be more) and $*$ holds for $g(x)$.

Consider $Z_1 = \{N, Nx, ...Nx^{d-1}, f(x)\}$, any linear combination of polynomials in $Z_1$ will have all the roots of $f(x)(\operatorname{mod} N)$. Consider the following matrix $B_1$.

$$B_1 = \begin{pmatrix} N & 0 & ... & a_0 \\ 0 & BN & ... & Ba_1 \\ 0 & 0 & & \vdots \\ \vdots & \vdots & & B^{d-1}a_{d-1} \\ 0 & 0 & & B^d \end{pmatrix}$$

Then $\det(\mathcal{L}(B_1)) = N^d \times B^{\frac{d(d+1)}{2}}$. Using $L^3$, we find an integer combination columns of $B_1$ to be a vector $v$, such that $\|v\| \leqslant O(\lambda_1(L_1)) \leqslant \left(\det(L_1)^{\frac{1}{d-n}}\right) = O\left(N.\frac{B^{\frac{d}{2}}}{N^{\frac{1}{d+1}}}\right)$.

If $B \leqslant c_1(d) N^{\frac{2}{d(d+1)}}$ then $\|v\| \leqslant \frac{N}{d+1}$. Suppose $v = (v_0, \ldots, v_{d+1})^T$ consider the polynomial $g(x) = b_d x^d + b_{d-1} x^{d-1} + \ldots b_1 x + b_0$. Obtained by integer combination of polynomials in $Z_1$ which correspond to the coodinates of $v$. Then $|b_1 B^i| < \frac{N}{d+1}$ and then find all the roots of $g(x)$ to get roots of $f(x)$. But, $B < c(d) N^{\frac{2}{d(d+1)}}$ is not a good bound, so we try to construct alternate lattice $L' = \{N, Nx, \ldots, Nx^{d-1}\} \bigcup \{f(x), x f(x), \ldots, x^{d-1} f(x)\}$.

$$B_2 = \left(\begin{array}{cccc} B_1 & 0 & & 0 \\ & a_0 & \ddots & 0 \\ & Ba_1 & \ddots & 0 \\ & \vdots & & Ba_0 \\ 0 & B^d & & \vdots \\ & & & \vdots \\ & & & B^{2d}-1 \end{array}\right)$$

with this lattice one can work with $B < c_2(d) N^{\frac{1}{2d-1}}$.

## 11.8 Some complexity theoretic aspects of CVP and SVP

**Decisional CVP.** Given a lattice basis $B$, target $t$ and a rational $r \in Q$, determine whether $\operatorname{dist}(t, \mathcal{L}(B)) \leqslant r$ or not.

**Optimization CVP.** Given a lattice basis $B$ and $t$, determine $r = \operatorname{dist}(t, L(B))$.

**Search CVP.** Given $B$ and $t$, determine $v \in L(B)$ such that $\operatorname{dist}(v, t) = \operatorname{dist}(t, L(B))$.

**Note 38.** It is easy to note that order of difficultly of following problems is dec CVP $\leqslant$ opt CVP $\leqslant$ search CVP.

**Theorem 40.** *If there is an oracle to solve decision CVP, then there is an algorithm to solve search CVP in polynomial time, using oracle for decision CVP.*

**Proof.** We have the input as $(B, t)$. We first determine $r = \operatorname{dist}(t, \mathcal{L}(B))$.

An upper bound for $r$ is $R = \sum_{i=1}^n \|b_i\|$ $t = \sum a_i b_i$ and let $x = \sum \lfloor a_i \rceil b_i$. Then $\operatorname{dist}(v, t) \leqslant \operatorname{dist}(x, t) \leqslant \sum \|b_i\|$. Also $r$ is a square root of integer, so $R^2$ is an integer in the rage $[1, R^2]$, so performing a binary search over these $R^2$ values in 2 logR time finds $r$. □

**Note 39.** It is sufficient to find a vector $x \in \mathcal{L}(B)$ which is closest to $w + t$ for some $v \in \mathcal{L}(B)$.

The idea is to sparsify the lattice; from $(b_!, b_2, \ldots, b_n)$, $t$ consider $(2b_1, b_2, \ldots, b_n)$, invoke the algorithm on $((2b_1, b_2, \ldots, b_n), t, r)$, if yes we proceed with $(2b_1, 2b_2, \ldots, b_n)$, $t$ else proceed with $(2b_1, \ldots, b_n), t - b_1$. We iterate this $K = n + \log r$ times to get basis $(2^k b_1, \ldots, b_n)$ for a sub lattice and a new target $t_1 = v_1 + t, v_1 \in \mathcal{L}(B)$. Perform this on each component to obtain the basis $(2^k b_1, \ldots, 2^k b_n)$ for a sublattice and a new target $s = v + t, v \in \mathcal{L}(B)$. $\lambda_1(\mathcal{L}(B')) \geqslant 2^k.2^n.r$. The next closest vector to $s$ is at a distance $r.2^n - r > 2^{n-1}$. Now applying Babai nearest plane algorithm, we obtain a vector in $\mathcal{L}(B')$ which is at distance atmost $2^{n-1}$ from $r$.

**Theorem 41.** *Dec-CVP is NP-complete*

**Proof.** We consider an instance $(B, t, r)$. Any $x \in \mathcal{L}(B)$ such that, $\|x - t\| \leqslant r$ is an $\mathrm{NP}-$witness.

All entries of $x$ are atmost $\|t\| + r$ in absolute value.

NP − hard: SubSet(SS) $\leqslant$ decCVP

SS: is there a subset $A \subset \{1, \ldots, n\}$ such that $\sum_{i \in A} a_i = S$. Instance of dec-CVP is constructed as the following:

$$B = \begin{pmatrix} a_1 & a_2 & \ldots & a_n \\ 2 & 0 & & 0 \\ 0 & 2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 2 \end{pmatrix}_{(n+1) \times n} \qquad t = \begin{pmatrix} s \\ 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}_{(n+1) \times 1} \qquad r = \sqrt{n}$$

Suppose the subset sum instance is an yes, then dec-CVP instalce is also an yes. □

**GapCVP$_\gamma$:input** $(B, t, r)$. Returns yes, if $\text{dist}(t, \mathcal{L}(B)) \leqslant r$, no if $\text{dist}(t, \mathcal{L}(B)) > \gamma r$.
**GapSVP$_\gamma$: input** $(B, r)$. Returns yes, if $\lambda_1(\mathcal{L}(B)) \leqslant r$, no if $\lambda_1(\mathcal{L}(B)) > \gamma r$.

**Note 40.** $\gamma = 1$ transforms the above problems into usual decision versions.

**Theorem 42.** *For any $\gamma \geqslant 1$, given access to an oracle for* $\text{GapCVP}_\gamma$*, it is possible to solve* $\text{GapSVP}_\gamma$ *in polynomial time.*

**Proof.** Instance of $\text{GapSVP}_\gamma$ is $(B, r)$. Construct $B_i = (b_1, \ldots, b_{i-1}, 2b_i, b_{i+1}, \ldots, b_n)$, $t_i = b_i$ and $\text{dist} = r$ and $(B_i, t_i, r)$ are $n$ instances of $\text{GapCVP}_\gamma$.

**Algorithm:** Invoke the oracle on all the $n$ instances, if the oracle returns yes on any instance then return yes, o.w return no.

**Correctness:** Suppose $(B, r)$ is an yes instance of gap-SVP$_\gamma$. So $\lambda_1(\mathcal{L}(B)) \leqslant r$. Let $v$ be the shortest vector in $\mathcal{L}(B)$ so that $\|v\| \leqslant r$. Write $v = \sum a_i b_i$. Not all of $a_i$ can ve even, for otherwise $\frac{v}{2}$ would be a shorter vector in $\mathcal{L}(B)$. Suppose $a_i$ is odd, then $\|v - b_i + b_i\| \leqslant r$ forcing $v + b_i \in \mathcal{L}(B)$. So the $i$ call returns yes.

Suppose $(B, r)$ is a no instance then $\lambda_1(\mathcal{L}(B)) \geqslant \gamma r$. For a non zero vector $v$,$\|v\| \geqslant \gamma r$. For each, $v \in \mathcal{L}(B_i), v - b_i \in \mathcal{L}(B)$ and $v - b_i \neq 0$. So $\|v - b_i\| \geqslant \gamma r$ i.e. $\text{dist}(b_i, \mathcal{L}(B_i)) \geqslant \gamma r$ □

## 11.9 Odds and Ends - Some applications of Lattices

**Theorem 43.** *If $p$ is a prime and $p \equiv 1 \pmod 4$ then $p$ can be written as sum of two squares.*

**Proof.** $p \equiv 1 (\text{mod}\, 4) \Rightarrow \frac{p-1}{2}$ is even and so $(-1)^{\frac{p-1}{2}} \equiv 1 (\text{mod}\, p)$. So $-1$ is quadratic residue mod $p$.

Let $i$ be a root of $-1$ so that $p | i^2 + 1$. Consider $B = \left(\begin{smallmatrix} 1 & 0 \\ i & p \end{smallmatrix}\right)$ and $\mathcal{L}(B)$ is a lattice. By Minkowski's theorem, there exists a vector whose length is less than $\sqrt{2 \det(B)}$. $v = Bx$ for some $x = (x_1, x_2)^T \in \mathbb{Z}^2$. $0 < \|Bx\|_2^2 < 2 \det(B) = 2p$. Taking $v \neq 0, 0 < x_1^2 + (i x_1 + p x_2)^2 < 2p$. So $x \equiv 0 (\text{mod}\, p)$ and $0 < x < 2p \Rightarrow x = p$. □

### 11.9.1 Hermite Canonical Form