# Algebra and Computation

*Instructor: Amit Kumar Sinhababu and Sumanta Ghosh*

Scribe: Soham Chatterjee

sohamchatterjee999@gmail.com
Website: sohamch08.github.io

2024

CHAPTER 1

# Introduction

CHAPTER 2

# Integer and Polynomial Addition

# CHAPTER 3

## Integer and Polynomial Multiplication

**Definition 3.1: Multiplication Time Function: $M(n)$**

The function $M : \mathbb{N} \to \mathbb{R}_+$ for any commutative ring $R[x]$ is called multiplication time function for if polynomials in $R[x]$ of degree less than $n$ can be multiplied using at most $M(n)$ operations in $R$.

Similarly we can define the function $M$ as above for multiplication time for $\mathbb{Z}$ if two integers of length $n$ bits can be multiplied using at most $M(n)$ operations

**Assumption 3.0.1.** *content...*

**Proof of Claim c:** ontent... ■

CHAPTER 4

# Polynomial Evaluation

## 4.1 Introduction

We will consider the following situation: $R$ is a commutative ring as always and $f \in R[x]$ where $\deg(f) = d$. We also have $k$ points $u_0, \ldots, u_{k-1} \in R$. Now we want to discuss here the fast algorithms of finding out $(f(u_0), \ldots, f(u_{k-1}))$. So we basically want the evaluation map

$$\begin{aligned} \varphi : R[x]/\langle m \rangle &\to R^n \\ f &\to (f(u_0), \ldots, f(u_{k-1})) \end{aligned}$$

which is a ring homomorphism. If $R$ is a field then $R[x]$ is a vector space over $R$ and the $\phi$ is an isomorphism. Formally we want to solve the following two problems with fast algorithms:

**Problem 4.1: Single Point evaluation**

Given $f \in R[x]$ with $\deg(f) = d$ and $\alpha \in R$ compute $f(\alpha)$

**Problem 4.2: Multi-Point evaluation**

Given $f \in R[x]$ with $\deg(f) = d$ and $u_o, \ldots, u_{n-1} \in R$ compute $f(u_0), \ldots, f(u_{n-1})$

## 4.2 Single Point Evaluation

### 4.2.1 Horner's Method

**Theorem 4.2.1** Horner's Method

Given a polynomial $f(x) = \sum_{i=0}^{d} a_i x^i$ where $a_i \in R$ for all $i \in [n]$ and a point $\alpha \in R$ using only $O(d)$ many additions and multiplications.

***Proof:*** Consider the following algorithm:

**Algorithm 1:** Horner's Method

**begin**

$p(x) = a_0 + x(a_1 + x(a_2 + (x(\cdots + x(a_n)\cdots))))$

Clearly we are using only $d$ many additions and $d$ many multiplications. So overall we need $2d = O(d)$ ring operations to evaluate the polynomial. The following lower bound results we obtain ■

This is the minimal number of additions and multiplications for any algorithm to evaluate a polynomial.

**Theorem 4.2.2** [OST13]

Any algorithm to evaluate an arbitrary degree $d$ polynomial $f \in R[x]$ at any point $\alpha \in R$ must use at least $n$ additions

**Theorem 4.2.3** [Pan66]

Any algorithm to evaluate an arbitrary degree $d$ polynomial $f \in R[x]$ at any point $\alpha \in R$ without initial conditioning of coefficients has at least $n$ multiplications and at least $n$ additions.

**Theorem 4.2.4** [Pan66],[Mot55]

Any degree $d$ real polynomial can be evaluated using $\left\lfloor \frac{d}{2} \right\rfloor + 2$ multiplications and $d$ additions.

## 4.3 Fast Multi-point Evaluation

CHAPTER 5

# Polynomial Interpolation

CHAPTER 6

# Bibliography


[Mot55] T. S. Motzkin. Evaluation of polynomials and evaluation of rational functions. *Bulletin of the American Mathematical Society*, 61:163, 1955.

[OST13] A. OSTROWSKI. On two problems in abstract algebra connected with horner's rule. In RICHARD von MISES, editor, *Studies in Mathematics and Mechanics*, pages 40–48. Academic Press, 2013.

[Pan66] Victor Y. Pan. Methods of computing values of polynomials. *Russian Mathematical Surveys*, 21:105–136, 1966.