

FFT for Dummies

Pedro G. M. R. Alves

Senior GPU Engineer @ Zama¹

Aug 2025

¹The views expressed in this presentation are my own and do not necessarily reflect those of my employer.

Motivation

- (Large degree) Polynomial multiplication is an important operation,
- a naive approach can be $O(n^2)$,
- in the Fourier domain, its cost is linear,
- using the FFT, the total cost goes down to $O(n \log n)$.

Polynomials

What's a polynomial?

Polynomials

What's a polynomial?

$$p(x) = x^2 + 7x + 2$$

is a 2-degree polynomial.

Polynomials

What's a polynomial?

$$p(x) = x^2 + 7x + 2$$

is a 2-degree polynomial.

Let $b = \{2, 7, 1\}$. Then

$$p(x) = \sum_{i=0}^2 b_i x^i.$$

Polynomials

$$p(x) = \sum_{i=0}^N b_i x^i.$$

is an N -degree polynomial.

Irreducible Polynomials

Definition

Let F be a finite field and R a ring of polynomials with coefficients in F . A polynomial $P \in R$ is said to be **irreducible** over F if there do not exist polynomials $A, B \in R$ such that:

$$P = A \cdot B$$

That is, P cannot be factored as the product of two nontrivial polynomials in R .

You may think on irreducible polynomials are “polynomial primes”.

Cyclotomic Polynomials

Definition

Let P be an irreducible polynomial over a field F . P is the n th **cyclotomic polynomial** if it divides $X^n - 1$ and does not divide $X^k - 1$ for any $k < n$, with $k, n \in \mathbb{Z}$.

Special case: If n is a power of 2, the n th cyclotomic polynomial is given by:

$$\Phi_n(X) = X^{n/2} + 1$$

Complex Numbers

$$i^2 = -1$$

Complex Numbers

$$i^2 = -1$$

A **complex number** is written as:

$$z = a + ib \quad \text{with } a, b \in \mathbb{R}$$

- a : real part
- b : imaginary part

Complex Numbers

$$i^2 = -1$$

A **complex number** is written as:

$$z = a + ib \quad \text{with } a, b \in \mathbb{R}$$

- a : real part
- b : imaginary part

(a, b) is what matters

Complex Numbers

$$z = a + ib \quad \text{with } a, b \in \mathbb{R}, \quad i^2 = -1$$

Addition:

$$(a + ib) + (c + id) = (a + c) + i(b + d)$$

Multiplication:

$$(a + ib)(c + id) = (ac - bd) + i(ad + bc)$$

Associative Operations

- Changing the grouping of operands does **not** change the result.

$$(a \circ b) \circ c = a \circ (b \circ c)$$

- Example (Addition):

$$(2 + 3) + 4 = 2 + (3 + 4) = 9$$

- Example (Multiplication):

$$(2 \cdot 3) \cdot 4 = 2 \cdot (3 \cdot 4) = 24$$

Commutative Operations

- Changing the order of operands does **not** change the result.

$$a \circ b = b \circ a$$

- Example (Addition):

$$2 + 3 = 3 + 2 = 5$$

- Example (Multiplication):

$$2 \cdot 4 = 4 \cdot 2 = 8$$

Neutral Elements

Definition

An element \mathbb{O} is a **neutral element** (also called identity) for an operation \circ on a set S if:

$$\forall a \in S, \quad a \circ \mathbb{O} = a$$

Examples:

- **Addition** (+) on integers:

$$a + 0 = a \quad \Rightarrow \text{Neutral: } 0$$

- **Multiplication** (·) on real numbers:

$$a \cdot 1 = a \quad \Rightarrow \text{Neutral: } 1$$

Note: A set may have different neutral elements for different operations.

Inverse Elements

Definition

Given an operation \circ on a set S , an element $\mathbb{I} \in S$ is called the **inverse** of $a \in S$ if:

$$a \circ \mathbb{I} = \mathbb{O}$$

Examples:

- **Addition** on integers:

$$a + (-a) = 0 \quad \Rightarrow \text{Inverse of } a \text{ is } -a$$

- **Multiplication** on nonzero real numbers:

$$a \cdot \frac{1}{a} = 1 \quad \Rightarrow \text{Inverse of } a \text{ is } \frac{1}{a}$$

Note: A set where every element has an inverse (under some operation) forms a **group**.

Algebraic structures

Group:

- A set with one operation (e.g., addition or multiplication) that is:
 - associative,
 - has a neutral element,
 - each element has an inverse.
- If the operation is also commutative, the group is called **abelian**.
- **Example:** Integers with addition

Ring:

- A set that is an abelian group under $+$,
- equipped with an associative multiplication \cdot ,
- has a multiplicative identity (optional in some definitions),
- satisfies distributivity:

$$a \cdot (b + c) = a \cdot b + a \cdot c$$

- **Example:** Integers

Algebraic structures

Field:

- A set where:
 - $(F, +)$ is an abelian group with additive identity \mathbb{O} ,
 - $(F \setminus \{\mathbb{O}\}, \times)$ is an abelian group with multiplicative identity \mathbb{I}
 - i.e. every non-zero element has a multiplicative inverse.
 - Multiplication is distributive over addition:

$$a \cdot (b + c) = a \cdot b + a \cdot c$$

- **Examples:** Rational numbers \mathbb{Q} , real numbers \mathbb{R}

Finite Field:

- A field with a finite number of elements
- Also called a **Galois Field**, denoted \mathbb{F}_p or $\text{GF}(p^k)$
- Commonly used in cryptography and FFT over finite domains

Primitive Roots Modulo a Prime

Let p be a prime number. A number $g \in \mathbb{Z}_p$ is called a **primitive root modulo p** if:

$$\{g^1, g^2, \dots, g^{p-1}\} \equiv \{1, 2, \dots, p-1\} \pmod{p}$$

Primitive Roots Modulo a Prime

Let p be a prime number. A number $g \in \mathbb{Z}_p$ is called a **primitive root modulo p** if:

$$\{g^1, g^2, \dots, g^{p-1}\} \equiv \{1, 2, \dots, p-1\} \pmod{p}$$

Example: $g = 3$ is a primitive root modulo 7 because:

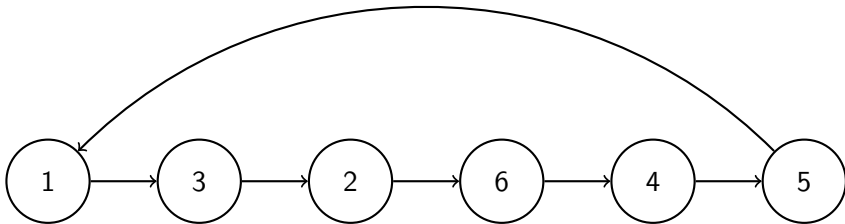
$$3^1 = 3, \quad 3^2 = 2, \quad 3^3 = 6, \quad 3^4 = 4, \quad 3^5 = 5, \quad 3^6 = 1 \pmod{7}$$

Visualizing a Cyclic Subgroup

Example: The multiplicative group $\mathbb{F}_7^\times = \{1, 2, 3, 4, 5, 6\}$

The element $3 \in \mathbb{F}_7^\times$ is a **primitive root**, because:

$$\langle 3 \rangle = \{3^0, 3^1, 3^2, 3^3, 3^4, 3^5\} \pmod{7} = \{1, 3, 2, 6, 4, 5\}$$



The group is cyclic of order 6, generated by 3.

Primitive n th Root of Unity

Let R be a ring (or field).

An element $\omega \in R$ is called an **n th root of unity** if:

$$\underbrace{\omega \cdot \omega \cdot \dots \cdot \omega}_{n \text{ times}} = \omega^n = 1 \in R$$

It is called a **primitive n th root of unity** if:

- $\omega^n = 1$, and
- $\omega^k \neq 1$ for all $1 \leq k < n$

Remarks:

- The powers $\omega^0, \omega^1, \dots, \omega^{n-1}$ form a cyclic subgroup of order n .
- Such roots exist in many settings: complex numbers, finite fields, modular arithmetic, etc.

Polynomial Multiplication: The Schoolbook Way

Let two polynomials:

$$A(x) = \sum_{i=0}^{n-1} a_i x^i, \quad B(x) = \sum_{j=0}^{n-1} b_j x^j$$

Their product is:

$$C(x) = A(x) \cdot B(x) = \sum_{k=0}^{2n-2} c_k x^k$$

where each coefficient c_k is given by:

$$c_k = \sum_{i=0}^k a_i b_{k-i}$$

Complexity: $O(n^2)$

Polynomial Multiplication: The Schoolbook Way

Let two polynomials:

$$A(x) = \sum_{i=0}^{n-1} a_i x^i, \quad B(x) = \sum_{j=0}^{n-1} b_j x^j$$

Their product is:

$$C(x) = A(x) \cdot B(x) = \sum_{k=0}^{2n-2} c_k x^k$$

where each coefficient c_k is given by:

$$c_k = \sum_{i=0}^k a_i b_{k-i}$$

Complexity: $O(n^2)$ and that's **bad** = (

What is a Transform?

A **transform** is a mathematical process that converts a signal from one form to another — usually to reveal some hidden structure.

DFT as a Matrix–Vector Multiplication

Let ω be a primitive n th root of unity. The Discrete Fourier Transform (DFT) of a vector $\mathbf{y} = [y_0, y_1, \dots, y_{n-1}]^T$ is computed as:

$$\begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega & \omega^2 & \dots & \omega^{n-1} \\ 1 & \omega^2 & \omega^4 & \dots & \omega^{2(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{n-1} & \omega^{2(n-1)} & \dots & \omega^{(n-1)^2} \end{bmatrix} \cdot \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{n-1} \end{bmatrix}$$

DFT as a Matrix–Vector Multiplication

Let ω be a primitive n th root of unity. The Discrete Fourier Transform (DFT) of a vector $\mathbf{y} = [y_0, y_1, \dots, y_{n-1}]^T$ is computed as:

$$\begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega & \omega^2 & \dots & \omega^{n-1} \\ 1 & \omega^2 & \omega^4 & \dots & \omega^{2(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{n-1} & \omega^{2(n-1)} & \dots & \omega^{(n-1)^2} \end{bmatrix} \cdot \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{n-1} \end{bmatrix}$$

Each output X_j corresponds to evaluating the polynomial

$$p(X) = \sum_{k=0}^{n-1} y_k \cdot X^k$$

on ω^j for $j = 0, \dots, n-1$. In the complex numbers, $\omega = e^{2\pi i/n}$.

DFT Example: Input $[1, 2, 3, 4]^T$

Let $\omega = e^{2\pi i/4} = i$. Given input vector:

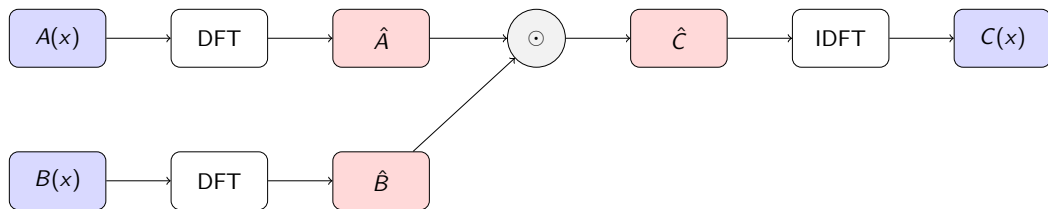
$$\mathbf{x} = [1, 2, 3, 4]^T$$

we compute the DFT:

$$\mathbf{X} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 10 \\ -2 + 2i \\ -2 \\ -2 - 2i \end{bmatrix}$$

- $X_0 = 1 + 2 + 3 + 4 = 10$
- $X_1 = 1 + 2i - 3 - 4i = -2 + 2i$
- $X_2 = 1 - 2 + 3 - 4 = -2$
- $X_3 = 1 - 2i - 3 + 4i = -2 - 2i$

Polynomial Multiplication via DFT



DFT / IDFT: complexity $\mathcal{O}(n^2)$

Pointwise multiplication: complexity $\mathcal{O}(n)$

Transform \rightarrow Multiply \rightarrow Inverse Transform
convolution becomes element-wise.

What is the FFT?

Fast Fourier Transform (FFT) is an efficient algorithm to compute the Discrete Fourier Transform (DFT).

DFT complexity: $\mathcal{O}(n^2)$

FFT complexity: $\mathcal{O}(n \log n)$

Key idea:

- It recursively breaks the transform into smaller DFTs.
- Works best when n is a power of 2.

Why it matters:

- Makes the Fourier domain relevant to accelerate polynomial multiplications.

Intuition behind the FFT

FFT uses a **divide and conquer** strategy.

Given a polynomial:

$$P(x) = \sum_{k=0}^{n-1} a_k x^k$$

Split the input:

- **Even part:** $P_{\text{even}}(x) = a_0 + a_2x + a_4x^2 + \dots$
- **Odd part:** $P_{\text{odd}}(x) = a_1 + a_3x + a_5x^2 + \dots$

Recursive idea:

$$P(x) = P_{\text{even}}(x^2) + x \cdot P_{\text{odd}}(x^2)$$

FFT Gives Free Reduction Modulo $\Phi_n(X)$

In many lattice-based crypto and FHE schemes, we work in:

$$R_q = \mathbb{Z}_q[X]/(\Phi_n(X)) \quad \text{with } \Phi_n(X) = X^n + 1$$

R_q is the ring of polynomials modulus $\Phi_n(X)$ with coefficients in the finite field \mathbb{Z}_q .

Result:

- Polynomial multiplication in Fourier domain is already **modulo** $\Phi_n(X)$.
- No explicit reduction step is needed — it's **built into the transform**.
- This makes multiplication in R_q fast and efficient.

This is why FFT is essential in efficient FHE.

Number Theoretic Transform (NTT)

Let $q = k \cdot N + 1$ be a prime and r a primitive root of q . The **NTT** is a generalization of the FFT that works over the finite field \mathbb{Z}_q .

Same exact algorithm as FFT:

- $\omega_N \equiv r^k \pmod{q}$.
- NTT domain has the same properties regarding polynomial multiplication as the Fourier domain.

Key difference:

- Operates over \mathbb{Z}_q instead of \mathbb{C}
- Requires $q \equiv 1 \pmod{2n}$ so that primitive roots of unity exist in \mathbb{Z}_q

Pros:

- Doesn't add floating-point errors in integer operations.
- May be faster if integer instructions are faster.

Discrete Galois Transform (DGT)

The **DGT** is another generalization of the FFT that works on the Galois field $\mathbb{Z}_q[i]$.

A **Galois integer** is written as:

$$z = a + ib \quad \text{with } a, b \in \mathbb{Z}_q$$

- An input N -degree polynomial is folded resulting in an $(N/2)$ -degree input.
- Can better fit the hardware if enough bandwidth is available.

FFT x NTT x DGT?




Which one is the best?

FFT x NTT x DGT?

Which one is the best?
It depends.



References I

-  Alves, Pedro Geraldo Morelli Rodrigues (2016). “Computação sobre dados cifrados em GPGPUs”. Portuguese. M.Sc. Dissertation. Campinas, Brazil: Universidade Estadual de Campinas (Unicamp).
-  Bailey, D. H. (1989). “FFTs in external or hierarchical memory”. In: *Supercomputing '89: Proceedings of the 1989 ACM/IEEE Conference on Supercomputing*, pp. 234–242. DOI: [10.1145/76263.76288](https://doi.org/10.1145/76263.76288).
-  Govindaraju, Naga K. et al. (2008). “High performance discrete Fourier transforms on graphics processors”. In: *SC '08: Proceedings of the 2008 ACM/IEEE Conference on Supercomputing*, pp. 1–12. DOI: [10.1109/SC.2008.5213922](https://doi.org/10.1109/SC.2008.5213922).