An Introduction to Number Theoretic Transform in Post-Quantum Lattice Cryptography
EE5163 Advanced Digital Signal Processing: Final Report

Lo-Chun, Chou
R13922136 [1]

[1]Department of Computer Science, National Taiwan University

June 10, 2025

Abstract

In the domain of post-quantum cryptography (PQC), lattice-based cryptography stands out as one of the most promising approaches due to its balanced performance between security and efficiency. A key computational task in these schemes is polynomial multiplication, where Number Theoretic Transform (NTT) is particularly effective for this task. NTT achieves quasilinear time complexity $O(n \log n)$, similar to the Fast Fourier Transform (FFT), but operates entirely over integers, thus avoiding rounding errors from floating-point arithmetic. In this report, we explore the foundations of NTT, drawing from some recent research papers, and provide an overview of its principles, implementation strategies, and applications in lattice-based cryptographic systems.

1 Introduction

1.1 Motivation

While both Number Theoretic Transform (NTT) and Fast Fourier Transform (FFT) are used in polynomial multiplication, NTT is a relative new approach, which is often not covered in the textbooks. This lack of accessible resources is also noted in the reference work *A Beginner's Guide to Number Theoretic Transforms (NTTs)*, where the authors mentioned that the lack of guidance and tutorial available in one place is the main obstacle for learning NTT[**?**], which is also the reason why they wrote the material.

While the beginner's guide offers a solid introduction to the basic ideas of NTT, it occasionally assumes familiarity with advanced mathematical concepts. Therefore, we expand upon its foundation by including additional mathematical background, clarifying essential concepts such as rings, primitive roots, and modular arithmetic, so all mathematical backgrounds that are needed but unfamiliar to a non-math-major would be covered.

Furthermore, we conclude with a brief example introducing the applications of NTT in real-world scenarios, so that we can move beyond the theoretical part, gaining a full picture of the concept of NTT.

1.2 Polynomial Multiplication in Cryptographic Systems

1.3 Report Structure

2 Preliminaries: Mathematical Foundations of the Number Theoretic Transform

2.1 Notations and Definitions

To unify the notations and definitions, in this report, we assume that if given two polynomials $G(x)$ and $H(x)$, then they are of degree $n-1$ if not specified.

2.2 Rings

To properly define primitive roots and illustrate the structure of the Number Theoretic Transform (NTT), it is necessary to first introduce the concepts of rings, ideals, and quotient rings.

These algebraic structures are needed because they provide the foundation for polynomial multiplication in NTT, since in order to do polynomial multiplication for polynomials $G(x)$ and $H(x)$, both of them must belong to the same quotient ring. Due to the fact that quotient rings are constructed on the notions of rings and ideals, we begin by briefly reviewing these foundational concepts.

Definition 2.1 (Ring, [2], pp. 85–86)**.** *A set R is called a **ring** if it has two binary operations, written as addition and multiplication, satisfying the following axioms for all $a, b, c \in R$:*

1. *$a + b = b + a$.*

2. *$a + (b + c) = (a + b) + c$.*

3. *An element 0 in R exists such that $0 + a = a$ for all a.*

4. *For each $a \in R$ an element $-a \in R$ exists such that $a + (-a) = 0$.*

5. *$a(bc) = (ab)c$.*

6. *An element 1 in R exists such that $1 \cdot a = a = a \cdot 1$ for all a.*

7. *$a(b + c) = ab + ac$ and $(b + c)a = ba + ca$.*

Definition 2.2 (Ideal, [**?**], p. 1)**.** *A subset A of a ring R (commutative, with 1) is said to be an **ideal** if*

> *1. $0 \in A$ and $a, b \in A \Rightarrow a + b, -a \in A$ (so A is an additive subgroup of R).*
>
> *2. $(a \in A, r \in R)\ ra \in A$.*

Intuitively, an ideal is a special subset of a ring, which includes the zero element and is closed under addition, also, it would satisfy that every multiple of an element in A is also in A.

Definition 2.3 (Quotient Ring, [**?**], p. 3)**.** *Let A be an ideal in the ring R. The **quotient ring** R/A is defined as follows:*

$$Set := \{r + A \mid r \in R\} \qquad (additive\ cosets)$$
$$0 := A$$
$$1 := 1 + A$$
$$(r + A) + (s + A) := (r + s) + A$$
$$(r + A)(s + A) := (rs) + A$$

where $r, s \in R$.

To be simple, a quotient ring R/A is formed by grouping elements of the ring R into disjoint sets called cosets, where each coset has the form $r + A = \{r + a \mid r \in R\}$. These cosets partition the ring in a way that reflects the structure of the ideal A.

The quotient rings and we treat every element in the same coset as "equivalent", just like in modular arithmetic. For example, in the integers, $\mathbb{Z}/n\mathbb{Z}$ represents the ring of integers modulo n, where all numbers differing by a multiple of n are considered the same. In the same way, a quotient ring allows us to "mod out" an ideal A and treat every element in A as zero, simplifying the structure of the ring and allowing new algebraic manipulations.

2.3   Modular Arithmetic and Roots of Unity

In this subsection, we first introduce the concept of residue class modulo n, in order to be used in defining $\mathbb{Z}_n$, then we further define the order of an integer under modulo n, so that primitive root can be defined. The reason why primitive root is needed is that it is used to construct the NTT matrix, which we would talk about the details in the next subsection.

Definition 2.4 (Residue Class Modulo n, [2], p. 30)**.** *If a is an integer, its equivalent class [a] with respect to congruence modulo n is called its **residue class modulo** n,*

and we write $\bar{a} = [a]$ for convenience:

$$\bar{a} = [a] = \{x \in \mathbb{Z} \mid x \equiv a \pmod{n}\}. \tag{1}$$

Intuitively saying, after we selected an integer n to be the modulus, we can divide all the integers into n classes, each class is a set of integers that having the same remainder when divided by n.

Therefore, for any integer n, there are n residue classes modulo n, which are $\bar{0}, \bar{1}, \bar{2}, \ldots, \overline{n-1}$, if we collect all of these residue classes as a set, then we get the set of integers modulo n, which is defined as follows:

Definition 2.5 (Integers Modulo n, [2], p. 30)**.** *The set of all residue classes modulo n is denoted*

$$\mathbb{Z}_n = \{\bar{0}, \bar{1}, \bar{2}, \ldots, \overline{n-1}\}. \tag{2}$$

*and is called the set of **integers modulo** n.*

Before introducing the definition of primitive root, we first define the order of an integer modulo n as follows:

Definition 2.6 (Order of an Integer Modulo n , [1], p. 147)**.** *Let $n > 1$ and $\gcd(a, n) = 1$. The **order of** a **modulo** n is the smallest positive integer k such that*

$$a^k \equiv 1 \pmod{n}. \tag{3}$$

Using the definition of order, we can then define the primitive root as follows:

Definition 2.7 (Primitive Root, [1], p. 150)**.** *If $\gcd(a, n) = 1$ and a is of order $\varphi(n)$ modulo n, then a is called a **primitive root** of the integer n. Equivalently, a is a primitive root modulo n if*

$$a^{\varphi(n)} \equiv 1 \pmod{n} \quad but \quad a^k \not\equiv 1 \pmod{n} \quad \forall \, 1 \leq k < \varphi(n). \tag{4}$$

Note that the primitive root may not be unique, and since we use the primitive root to construct the NTT matrix, the NTT of a polynomial may not be unique and would depend on the choice of the primitive root. [**?**, pp. 5–6]

In the context of NTT, we choose the modulus n to be a prime number, let it be denoted as q, and under the assumption that our polynomials $G(x)$ and $H(x)$ are of degree $n-1$, we try to find primitive roots ω that would satisfy:

$$\omega^n \equiv 1 \pmod{q} \quad \text{and} \quad \omega^k \not\equiv 1 \pmod{q} \quad \forall \, 1 \leq k < n. \tag{5}$$

Which means that if we're trying to do convolution of polynomials of degree 3, then we should find a primitive root that would be congruent to 1 when raised to the power of 4, under some choice of modulus q.

And such ω is called the **primitive n-th root of unity** in the ring $\mathbb{Z}_q$. [**?**, p. 4]

3   Limitations of NTT

3.1   Parameter Restrictions and Ring Compatibility

3.2   Negative Wrapped Convolution and Modulus Conditions

3.3   Hardware Bottlenecks and Communication Overheads

In hardware implementations of the Number Theoretic Transform (NTT), one major source of inefficiency arises from the intricate data communication between adjacent computation stages.
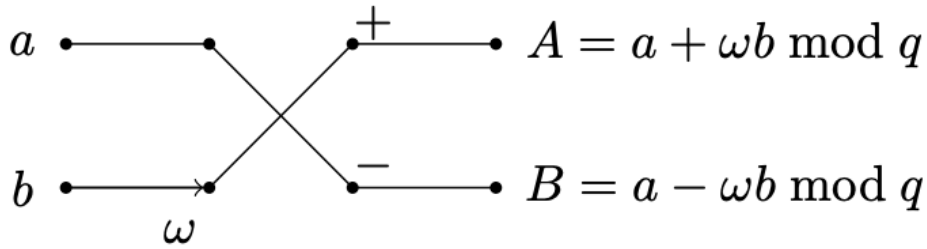
(a) Cooley-Tukey butterfly

Figure 1: Butterfly Diagram of the Cooley–Tukey NTT Algorithm. Source: `https://k.rypto.cafe/assets/images/misc/ct_butterfly.png`

As visualized in the butterfly diagram of the Cooley–Tukey NTT algorithm, multiple stages are required to process even small-length inputs, and each stage introduces data dependencies that result in extensive data movement. This leads to increased wiring complexity and higher area and energy overhead when deploying NTT on ASIC or FPGA-based accelerators. For in-memory computing architectures, these inefficiencies translate into additional shift operations to align data with memory bitlines, further increasing latency and hardware cost.

3.4 Handling NTT-Unfriendly Rings and Algorithmic Alternatives

4 Applications of NTT in Lattice-Based Cryptography

4.1 NTT in Post-Quantum Cryptographic Schemes

4.2 Error-Resilient and Hardware-Efficient Implementations

4.3 Optimized NTT Variants for Real-World Schemes

5 Conclusion and Future Directions

5.1 Summary of Key Concepts

5.2 Challenges and Open Research Problems

References

[1] D.M. Burton. *Elementary Number Theory*. Tata McGraw-Hill Publishing Company Limited, 2006.

[2] W. Keith Nicholson. *Introduction to Abstract Algebra*. Wiley Publishing, 4th edition, 2012.

[3] Ardianto Satriawan, Rella Mareta, and Hanho Lee. A complete beginner guide to the number theoretic transform (ntt), 2024. Last accessed 9 June 2025.