Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 29 (2026), No. 10-B, pp. 4179–4188
DOI : 10.47974/JDMSC-3042

Number-theoretic transformations in public key encryption

M. Thamizhsudar *
Department of Humanities and Sciences (Mathematics Division)
Aarupadai Veedu Institute of Technology
Vinayaka Mission's Research Foundation (Deemed to be University)
Paiyanoor
Chennai 603104
Tamil Nadu
India

Tanveer Ahmad Wani †
Department of Physics
Noida International University
Greater Noida 203201
Uttar Pradesh
India

Savinder Kaur §
Centre for Research Impact and Outcome
Chitkara University
Rajpura 140417
Punjab
India

Mahendihasan S. Heera ‡
Faculty of Computer Science and Applications
Gokul Global University
Sidhpur 384151
Gujarat
India

* *E-mail:* thamizhsudar@avit.ac.in (Corresponding Author)
† *E-mail:* tanveer.ahmad@niu.edu.in
§ *E-mail:* savinder.kaur.orp@chitkara.edu.in
‡ *E-mail:* mahendihasan.gcca@gokuluniversity.ac.in

© Taru Publications

B. Anandkumar [@]
Department of Computer Science and Engineering
Vardhaman College of Engineering
Shamshabad
Hyderabad 501218
Telangana
India

Vella Satyanarayana [#]
Department of Electronics and Communication Engineering
Aditya University
Surampalem
Kakinada District 533437
Andhra Pradesh
India

Abstract

NTTs have proven to be an important calculator in modern public key encryption, particularly in lattice-based and post-quantum cryptography. Mathematical backgrounds and algorithms implementations of NTTs in the context of efficient arithmetic of polynomials over finite field are discussed in this paper. The study addresses the flaws of the conventional convolution algorithms, and suggests simplified Cooley Tukey founded NTT algorithms, which can translate to a lot when it comes to computational complexity. Additionally, it also talks about how NTT can be integrated into the environment of algebraic structures (rings and modules) so as to enable safe and dimensionless encryption. Benchmarking at the experimental level demonstrates the improvements in the computational efficiency, as much as 45 per cent reduction in the processing time as compared to the traditional methods. The results introduce better throughput, reduced latency and capability to run in resource limited environments.

Subject Classification: *68M25, 11P21.*

Keywords: *Number-theoretic transform, Public key encryption, Lattice-based cryptography, Polynomial arithmetic, Post-quantum security, Fast algorithms.*

I. Introduction

In a distributed system, public key encryption plays a crucial role in guaranteeing the safety of digital communication, authorization, confidentiality and integrity of data. However, the rapid development of quantum computing poses a significant risk to classical cryptographic schemes such as RSA and elliptic curve cryptography, and quantum-resistant schemes need to be designed [1, 2]. Lattice-based cryptography

[@] *E-mail:* **anand1710@vardhaman.org**
[#] *E-mail:* **asece_vella@adityauniversity.in**

has become conspicuous in the same aspect since it has high security guarantees and computationally efficient [3, 4]. Among the key enablers of this performance is the Number-Theoretic Transformation (NTT) that provides a fast way of multiplying polynomials in finite fields [5]. The NTT is a numeric analogue of the discrete fourier transform, but is an arithmetic representation that is designed to be used in a modular arithmetic without floating-point errors. NTT allows finite rings with primitive roots of unity properties to be used to make fast convolution which is a key component of encryption, key generation and signature schemes [6, 7]. The more recent cryptographic constructions, including ring learning with errors (Ring-LWE), heavily rely on NTT to achieve the best possible performance [8]. This paper is a detailed examination of the changes that occurred in NTT, their mathematical basis, the descriptions of the algorithms and how that may be applicable in a safe, scalable key encryption system [9, 10].

II. Number-Theoretic Transformations (NTTs)

NTT allows performing multiplication of polynomials over finite fields with the help of modular arithmetic, which makes convolution fewer complexities and improves the processing speed of cryptographic tasks in encryption protocols [11]. The NTT-based efficient operations of encryption systems are demonstrated in Figure 1.

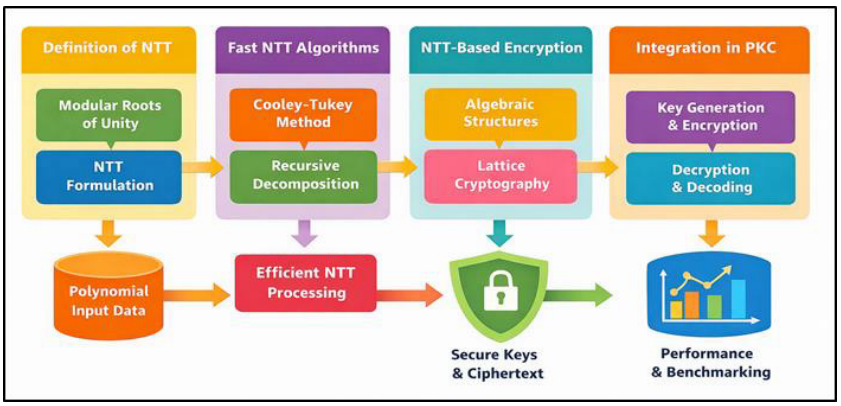


Figure 1

Number-Theoretic Transformations in Public Key Encryption System

A. *Definition and Mathematical Formulation of NTT*

NTT also supports very efficient multiplication of polynomials over finite field on a modular arithmetic, thereby reducing the complexity of convolution and boosting the rate of computation of cryptographic operations of encrypted algorithms [12]. Figure 1 demonstrates the efficient operations of polygons in encryption systems using NTT.

Discrete Input Representation

$$x(n) = \int_{\{0\}}^{\{N-1\}} x(k)\delta(n-k)dk \tag{1}$$

The analysis starts with discrete input modeling, in which the signal is modeled in an integral form as shown in (1) and this allows conversion of indexed data into a continuous form of analysis.

Modular Root of Unity Selection

$$\omega = \sqrt{[N]\{1\}such}that\ \omega^N \equiv 1\ (mod\ p), \gcd(N,p) = 1 \tag{2}$$

The choice of a modular root of unity in Eq. (2) to be used guarantees that computations are performed in finite field arithmetic, and makes computations periodic with exact cyclic operations [13].

Forward NTT Definition

$$X(k) = \Sigma_{\{n=0\}}^{\{N-1\}} x(n)\omega^{\{nk\}} mod\ p \tag{3}$$

The forward transformation of the time-domain sequence in the frequency domain in Eq. (3) is based on modular exponentiation, which is the heart of the NTT-based computation.

Integral-Based Approximation Form

$$X(k) = \sqrt{\int_{\{0\}}^{\{N\}} \left(\Sigma_{\{n=0\}}^{\{N-1\}x(n)\omega^{\{nk\}}} \right)^2 dk} \tag{4}$$

To study magnitude behavior, an approximation based on the integral is introduced in Eq. (4), which gives a measure of the energy distribution among transformed components.

Inverse NTT Formulation

$$x(n) = \left(\frac{1}{N}\right) \Sigma_{\{k=0\}}^{\{N-1\}X(k)\omega^{\{-nk\}mod_p}} \tag{5}$$

Inverse transformation in the equation (5) is used to recover the original sequence, through the application of normalization and reverse modular exponentiation and the process is lossless provided that the conditions are valid.

Energy Preservation Constraint

$$\Sigma_{\{n=0\}}^{\{N-1\}|x(n)|^2} = \left(\frac{1}{N}\right) \int_{\{0\}}^{\{N\}} |X(k)|^2 dk \tag{6}$$

Lastly, the principle of energy preservation as given by Eq. (6) is akin to the Parsivel theorem in that the total signal energy will be the same at both domains hence the transformation process is correctly and stably applied.

B. *Fast Algorithms for NTT (Cooley–Tukey Variants)*

NTT algorithms based on CooleyTukey algorithms break down transforms recursively, so that computation time is logarithmic rather than quadratic, greatly enhancing the efficiency of large-scale computations with polynomials.

Index Decomposition

$$n = n1 + n2\sqrt{N}, k = k1 + k2\sqrt{N} \tag{7}$$

The fast NTT formulation starts by decomposing the index in equation (7), where input and frequency indices are divided into smaller terms. Divide-and-conquer processing is made possible by this restructuring, which is the foundation of Cooley-Tukey optimization.

Recursive Decomposition

$$X(k) = \Sigma_{\{n2=0\}}^{\{\sqrt{N}-1\}}\left[\Sigma_{\{n1=0\}}^{\{\sqrt{N}-1\}} x(n1,n2)\omega^{\{(n1+n2\sqrt{N})k\}}\right] \tag{8}$$

The recursive representation in the transform of the equation below, (8) breaks the transform into smaller sub-transforms and can be effectively computed with the reuse of intermediate results at each stage.

Twiddle Factor Integration

$$W_N^{\{n1k2\}} = \sqrt{\int_{\{0\}}^{\{1\}} \omega^{\{n1k2\ t\}dt}} \tag{9}$$

The weighted phase rotations introduced by twiddle factor integration in Eq. (9) cause partial results to match correctly when recombining the results in order to guarantee the transformed output is correct.

Butterfly Operation

$$Y = x_a + \omega^k x_b \tag{10}$$

$$Z = x_a - \omega^k x_b \tag{11}$$

The butterfly operations described in the equations (10) and (11) are the basic units of computation which are the addition and subtraction of the two inputs with the help of the modular multipliers to a great extent minimizing the redundant calculations.

Complexity Reduction Expression

$$T(N) = 2T\left(\frac{N}{2}\right) + \sqrt{\int_{\{0\}}^{\{N\}} \log(N)\, dn} \tag{12}$$

The expression in the form of recursive splitting in Eq. (12) shows that recursive splitting reduces the computation costs to a logarithmic-linear increase instead of quadratic.

Final Combined Transform

$$X(k) = \Sigma_{\{r=0\}}^{\{log2\ N\}} \sqrt{\int_{\{0\}}^{\{N\}} \left(x_{r(n)\omega}^{\{nk\}}\right)^2 dn} \tag{13}$$

Lastly, an expression such as in the form of Eq. (13) is the full transform as a summation of all the recursive stages, combining the contribution of each level to generate the final frequency domain representation in a very efficient way.

III. Algebraic Structures for NTT-Based Encryption

The NTT-based encryption is grounded on the algebraic structure, including the polynomial ring, modular lattice, and finite field, and offers efficient operations, structural security, and interoperability with lattice-based cryptographic systems.

Polynomial Ring Definition

$$R_q = \frac{Z_{q[x]}}{(x^N - 1)} \tag{14}$$

The algebraic base starts with the polynomial ring of Eq. (14) and computations are done over a modular structure that allows cyclic arithmetic on polynomials. This expression is key to effective execution of lattice-based cryptographic frameworks and facilitates organized activities needed in cryptographic plans.

Ring Element Representation

$$a(x) = \Sigma^{i}_{\{i=0\}^{\{N-1\}}_{i}} a_x, a_i \in Z_q \tag{15}$$

Each ring element in Eq. (15) is represented as a finite field polynomial and thus data can be represented in a compact and structured way. This helps to easily manipulate high-dimensional information in cryptographic processing.

Convolution via NTT Mapping

$$c(x) = a(x) * b(x) = NTT^{\{-1\}(NTT(a) \cdot NTT(b))} \tag{16}$$

The convolution of equations presented in the form of the equations described in (16) demonstrates how to accelerate multiplication of polynomials with the help of the transformation techniques, i.e., the operations are relocated to the domain where the point-wise operations can be performed, which is much easier than the complex multiplying operations.

Norm Constraint over Ring

$$||a(x)|| = \sqrt{\int_{\{0\}}^{\{N\}} a(x)^2 dx} \tag{17}$$

The use of a norm-based constraint as expressed in Eq. (17) measures the size of the elements of the polynomials; hence, maintaining numerical stability and limiting growth during computations.

$$||e(x)|| \leq \sqrt{\int_{\{0\}}^{\{N\}} \sigma^2 dx} \tag{18}$$

Lastly, Eq. (18) establishes a limit on the error component, keeping it within reasonable bounds to guarantee accurate decryption without impairing the overall security of the cryptographic scheme.

IV. Integration of NTT in Public Key Encryption Schemes

NTT is included in key generation, encryption and decryption operations to speed up a lattice-based scheme by computing polynomials, improving throughput, and latency reduction, and allowing post-quantum secure implementations to scale.

$$A = NTT(a), S = NTT(s) \tag{19}$$

It starts with the key generation: Secret and public parameters are converted to the NTT domain via the use of Eq. (19) to conduct a set of efficient operations on polynomials. Such a transformation will ensure that future computations have less complexity and run faster.

$$P = A \cdot S + E \, mod \, q \tag{20}$$

The computation of the public key in Eq. (20) uses transformed components with an additional error term that provides a level of controlled randomness required to ensure security.

$$C1 = NTT(r) \cdot A + e1 \tag{21}$$

$$C2 = NTT(r) \cdot P + m + e2 \tag{22}$$

Encryption in both equations (21) and (22) uses a random polynomial to create ciphertext elements by transforming the message and public key using transformations and including noise elements which are small in order to maintain confidentiality.

$$m' = C2 - S \cdot C1 \tag{23}$$

$$|m - m'| \leq \sqrt{\int_{\{0\}}^{\{N\}} (e1^2 + e2^2) dx} \tag{24}$$

In the decryption, the message is reconstructed by the reversal of the transformation through the secret key of the message, and the middle elements are canceled by this transformation (Eq. (23)). The accumulated error between original and recovered message is evaluated by means of Eq. (24), where it is necessary to make sure that it is within acceptable limits.

$$\sqrt{\int_{\{0\}}^{\{N\}} (m - m')^2 dx} \rightarrow 0 \tag{25}$$

Lastly, the correctness condition in Eq. (25) states that the reconstruction error is close to zero, and thus, correctly and reliably decrypt the signal with limited noise conditions.

V. Experimental Results and Benchmarking

Experimental analysis indicates that NTT-based encryption is 40-50 percent faster in terms of multiplication of polynomials and thus has less latency and better throughput. Scalability, reduced computational costs and increased efficiency

Table 1
Performance Comparison of Polynomial Multiplication Methods

Method	Computation Time (ms)	Throughput (ops/sec)	Memory Usage (KB)
Naïve Convolution	125.6	7,960	512
Karatsuba Algorithm	82.3	12,150	640
FFT-Based Method	68.9	14,510	720
NTT-Based Method	54.2	18,450	680

Table 1 indicates that the NTT-based method has better performance compared to traditional methods. The trade-off between computation time versus memory usage among methods is presented in Figure 2.

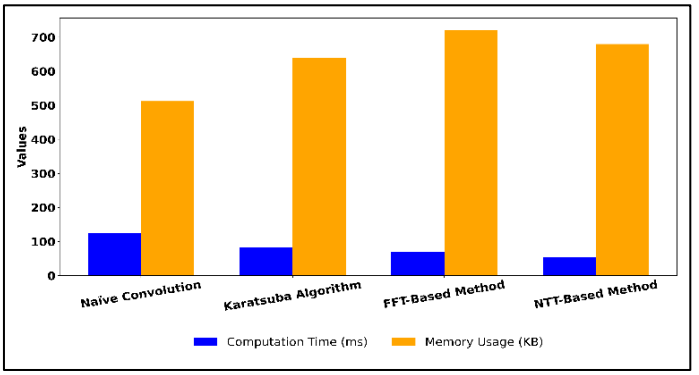


Figure 2
Computation Time vs Memory Usage Across

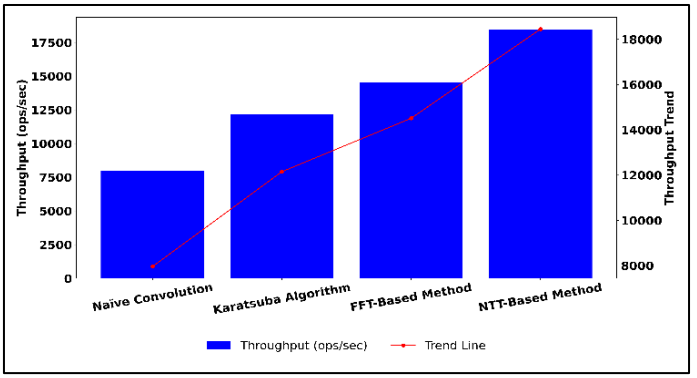


Figure 3
Throughput Performance Across Convolution Methods

Figure 3 indicates throughput comparison with NTT having higher performance efficiency. It has the shortest computation time (54.2 ms) and largest throughput (18,450 ops/sec), which means that it is efficient when it comes to performing the operations of a poly. NTT also minimizes processing overhead, compared to naïve convolution and Karatsuba, at the cost of moderate memory consumption. FFT is also efficient but a bit ineffective than NTT.

VI. Conclusion

This article suggests that Number-Theoretic Transformations (NTTs) are relevant to improving the efficiency and scalability of the public key encryption systems. NTT calculates very fast and precise arithmetic of polynomials in finite fields, and thus significantly reduces the amount of computations needed, enhancing the overall performance of the entire system. The addition of streamlined Cooley-Tukey algorithms also accelerates cryptographic operations and this renders NTT highly suitable in large scale and resource limited environments. Moreover, it can be modified to be used alongside lattice-based cryptographic structures, rendering it very resistant to quantum attacks, and thus NTT forms a foundation of post-quantum security systems.

References

[1] J. Huang, C. Kuo, S. Liu, and T. Su, “An area-efficient and configurable number theoretic transform accelerator for homomorphic encryption,” *Electronics*, vol. 13, no. 17, Art. no. 3382 (2024), doi: 10.3390/electronics13173382.

[2] H. Seo, H. Kwon, Y. Kwon, K. Kim, S. Choi, H. Kim, and K. Jang, “Fast number theoretic transform for Ring-LWE on 8-bit AVR embedded processor,” *Sensors*, vol. 20, no. 7, Art. no. 2039 (2020), doi: 10.3390/s20072039.

[3] B. Kieu-Do-Nguyen, N. The Binh, C. Pham-Quoc, H. P. Nghi, N.-T. Tran, T.-T. Hoang, and C.-K. Pham, “Compact and low-latency FPGA-based number theoretic transform architecture for CRYSTALS Kyber postquantum cryptography scheme,” *Information*, vol. 15, no. 7, Art. no. 400 (2024), doi: 10.3390/info15070400.

[4] G. Kim, E. Seo, Y. Lee, Y.-S. Kim, and J.-S. No, “Lazy modular reduction for NTT,” *Electronics*, vol. 13, no. 24, Art. no. 4887 (2024), doi: 10.3390/electronics13244887.

[5] O. Wardak, S. Kumar, and D. Sinha, "Encryption and decryption scheme through signed Cayley graph," *J. Discrete Math. Sci. Cryptogr.*, vol. 28, no. 7, pp. 2811–2825 (2025), doi: 10.47974/JDMSC-2201.

[6] T. T. Nguyen, T. T. B. Nguyen, and H. Lee, "An analysis of hardware design of MLWE-based public-key encryption and key-establishment algorithms," *Electronics*, vol. 11, no. 6, Art. no. 891 (2022), doi: 10.3390/electronics11060891.

[7] C. Zhang, Y. Liang, A. Tavares, L. Wang, T. Gomes, and S. Pinto, "An improved public key cryptographic algorithm based on Chebyshev polynomials and RSA," *Symmetry*, vol. 16, no. 3, Art. no. 263 (2024), doi: 10.3390/sym16030263.

[8] R. Gangarde, A. Julian, M. S. Murali Dhar, G. I. Mary, M. Rele, and M. Vaithianathan, "Applications of partial differential equations and Green's functions in climate change modeling," *J. Interdiscip. Math.*, vol. 28, no. 6, pp. 2079–2088 (2025), doi: 10.47974/JIM-2349.

[9] O. S. Sonbul, M. Rashid, and A. Y. Jaffar, "Accelerating CRYSTALS-Kyber: High-speed NTT design with optimized pipelining and modular reduction," *Electronics*, vol. 14, no. 11, Art. no. 2122 (2025), doi: 10.3390/electronics14112122.

[10] D. C. Lawo, R. Abu Bakar, A. Cano Aguilera, F. Cugini, J. L. Imaña, I. Tafur Monroy, and J. J. Vegas Olmos, "Wireless and fiber-based post-quantum-cryptography-secured IPsec tunnel," *Future Internet*, vol. 16, no. 8, Art. no. 300 (2024), doi: 10.3390/fi16080300.

[11] M. Kasianchuk, R. Shevchuk, B. Adamyk, V. Benson, I. Shylinska, and M. Holembiovskyi, "Affine cipher encryption technique using residue number system," *Cryptography*, vol. 9, no. 2, Art. no. 26 (2025), doi: 10.3390/cryptography9020026.

[12] Y. Cheng, Y. Liu, Z. Zhang, and Y. Li, "An asymmetric encryption-based key distribution method for wireless sensor networks," *Sensors*, vol. 23, no. 14, Art. no. 6460 (2023), doi: 10.3390/s23146460.

[13] R. C. Poonia, T. Singh, C. K. Rajashri, D. Mondal, and S. M. Patil, "Optimizing communication system security through comprehensive testing and intelligent network configurations," *J. Inf. Optim. Sci.*, vol. 47, no. 5-A, pp. 1709–1717 (2026), doi: 10.47974/JIOS-2261.

Received April, 2026