



## Optimizing Variance for Reliability Decryption in NTRUEncrypt

Daud, M. A.\* <sup>1,2</sup>, Kamarulhaili, H.\* <sup>1,5</sup>, Mandangan, A. <sup>3,5</sup>, and Asbullah, M. A. <sup>4,5</sup>

<sup>1</sup>*School of Mathematical Sciences, Universiti Sains Malaysia, Malaysia*

<sup>2</sup>*Preparatory Centre for Science and Technology, Universiti Malaysia Sabah, Malaysia*

<sup>3</sup>*Mathematics Visualization Research Group, Faculty of Sciences and Technology,  
Universiti Malaysia Sabah, Malaysia*

<sup>4</sup>*Centre for Foundation Studies in Sciences,  
Universiti Putra Malaysia, 43400 Serdang, Selangor*

<sup>5</sup>*Malaysia Cryptology Technology and Management Centre,  
Universiti Putra Malaysia, 43400 Serdang, Selangor*

*E-mail: [azlan.daud@ums.edu.my](mailto:azlan.daud@ums.edu.my)*

*[hailiza@usm.my](mailto:hailiza@usm.my)*

*\*Corresponding author*

*Received: 12 August 2024*

*Accepted: 5 August 2025*

### Abstract

This paper investigates factors influencing the polynomial coefficients of

$$a(x) = p \cdot r(x) * g(x) + m(x) * f(x),$$

ensuring they remain within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$  in NTRUEncrypt. The study highlights the significance of coefficient distribution and randomness. Through probability theory and statistical analysis, critical conditions for reliable decryption are identified. Findings demonstrate that selecting appropriate parameters ensures polynomial coefficients stay within the required bounds for successful decryption. By developing a detailed framework, the study focuses on optimizing parameter selection in NTRUEncrypt. The framework strengthens the theoretical and practical resilience of NTRUEncrypt against decryption failures, providing a deeper understanding of its decryption dynamics and enhancing its reliability in cryptographic applications.

**Keywords:** post quantum cryptography; ntruencrypt; decryption success; public key structures; coefficients probability distribution; coefficient variance.

## 1 Introduction

In cryptography, secure communication requires flawless encryption and decryption to preserve the integrity and readability of transmitted messages. A single bit error during decryption can render an entire message meaningless, underscoring the importance of reliability as a core requirement for cryptographic algorithms. As modern systems face growing demands for efficiency, robustness, and security, developing algorithms that minimize decryption failures while maintaining strong defenses against adversarial attacks becomes crucial.

Hoffstein *et al.* [4] introduced the NTRUEncrypt cryptosystem, outlining its polynomial ring structure and lattice-based security. Hosein [6] provides a detailed tutorial on the principles and design of NTRUEncrypt. Silverman and Whyte [13] showed that the probability of decryption failure in NTRUEncrypt can be kept below  $10^{-5}$ . They also outlined a theoretical framework that, when used with the recommended parameters, can push this probability below  $2^{-100}$ , highlighting the algorithm's strong resilience against errors. Yu *et al.* [15] were among the first to study how gaps in parameter selection can cause decryption errors in NTRUEncrypt. Their work shows that these gaps can undermine the scheme's reliability.

Attacks on NTRUEncrypt have exposed vulnerabilities where an NTRUEncrypt private key may fail to decrypt valid ciphertexts due to cyclic shifts facilitated by an oracle [11]. Such decryption failures have significant implications for NTRU-based schemes and padding methods. Additionally, limitations in average-case failure analysis for CCA2-secure schemes have been identified [7]. The introduction and demonstrated effectiveness of the Non-malleable, Almost-Equivalent Padding (NAEP) scheme underscore the importance of strengthening cryptographic protocols against evolving threats through rigorous security proofs [8]. In [9], the importance of selecting parameters in the NTRUEncrypt scheme to achieve 128-bit post-quantum security is emphasized, indirectly enhancing decryption reliability and success by ensuring robust cryptographic strength.

Past research on a few NTRU variants has shown that keeping polynomial coefficients within specific limits is essential for reliable decryption. In one study, the alternative NTRU variant HXDTRU was introduced, where the coefficients of  $A = pG \circ (\Phi \circ F) + (F \circ M) \circ F$  were analysed using the normal distribution to estimate the probability of staying within  $\left(-\frac{q}{2}, \frac{q}{2}\right)$  [1]. Another work on the BCTRU scheme [14] looked at the probability of successful decryption by checking whether all coefficients of  $A = pG * \phi * F + F * M * U$  fall within  $\left[-\frac{q-1}{2}, \frac{q-1}{2}\right]$ , and expressed this probability through statistical analysis. Both studies highlight that controlling coefficient variance plays a key role in improving decryption success rates, which is also a central focus of this paper.

The sections of this paper are organized as follows: Section 2 introduces foundational concepts of NTRUEncrypt, focusing on its structure and relevance within lattice-based cryptography. Section 3 explores relationships between statistical measures and coefficient distributions in the NTRUEncrypt. Section 4 examines the impact of variance on polynomial coefficients in NTRUEncrypt, specifically how ensuring all  $a_i$  coefficients remain within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right)$  leads to successful decryption. Finally, Section 5 concludes by summarizing our contributions and their implications for achieving reliable decryption through proper parameter selection.

## 2 $N^{th}$ – degree Truncated Polynomial Ring Units

Cryptography forms the backbone of secure digital communication, yet achieving flawless encryption and decryption remains a challenge. In [4], the original NTRUEncrypt scheme was introduced, setting out the polynomial ring structure and parameter choices that our work builds on. NTRUEncrypt is known for its efficiency and strong resistance to quantum attacks. Hülsing et al. [9] examined its role in post-quantum cryptography, showing that it compares well with other lattice-based approaches in both performance and security.

Unlike traditional cryptosystems that rely on factoring large integers or solving discrete logarithms, NTRUEncrypt derives its strength from the complexity of lattice problems, such as the Shortest Vector Problem (SVP) and other related computational challenges. Lattice problems like SVP are computationally hard, even for quantum computers, positioning NTRUEncrypt as a key contender in post-quantum cryptography. In [6], a clear explanation of NTRUEncrypt’s structure is provided, especially its use of ternary polynomials, which is directly relevant to our analysis of coefficient variance. The reliability of NTRUEncrypt depends on keeping decryption failures to a minimum, as these can be triggered by randomness in coefficients, noise, or poorly chosen parameters. Howgrave-Graham et al. [8] demonstrated that such failures can undermine the overall security of the scheme. Therefore, our work focuses on managing randomness, controlling noise, and selecting appropriate parameters by analyzing the impact on variance in polynomial coefficients.

### 2.1 NTRUEncrypt overview

NTRUEncrypt employs lattice-based cryptography principles to achieve secure data encryption. Its strength lies in the careful selection of parameters, which influence each stage of the cryptosystem, from key generation to decryption. A detailed tutorial on the construction of NTRUEncrypt, emphasizing its reliance on efficient polynomial arithmetic and the application of modular reductions in a truncated polynomial ring, is provided in [6].

Table 1 summarizes key parameters used in the key generation, encryption, and decryption processes of NTRUEncrypt, as established by [4] in their original presentation of the scheme.

Table 1: Description of parameters.

Parameter	Description	Channel
$N$	$\in \mathbb{Z}^+$ and sufficiently large to prevent lattice attacks	Public
$p$	$\in \mathbb{Z}^+$ and $\text{gcd}(p, q) = 1$	Public
$q$	$\in \mathbb{Z}^+$ , much larger than $p$ and $q > (6d + 1)p$	Public
$d$	$\{-1, 0, 1\}$	Public
$h(x)$	Public key polynomial	Public
$e(x)$	Ciphertext polynomial	Public
$f(x)$	Private key polynomial	Private
$F_p(x)$	Inverse of $f(x)$ modulo $p$	Private
$F_q(x)$	Inverse of $f(x)$ modulo $q$	Private
$g(x)$	Private key polynomial	Private
$r(x)$	Ephemeral key polynomial	Private
$m(x)$	Plaintext polynomial	Private

Parameters in Table 1 play a crucial role in determining the security and efficiency of the cryptosystem. Proper configuration is essential to prevent decryption errors, such as coefficient overflow or noise interference.

Key generation, encryption, and decryption are core parts of NTRUEncrypt. In [5] these steps are explained in detail, describing the use of polynomial rings and convolution operations.

---

**Algorithm 1:** NTRUEncrypt Key Generation Algorithm.

---

**Input:** Selection of public parameters  $(N, p, q, d)$  in NTRUEncrypt, where  $p$  is a small prime, subject to the constraints  $q > (6d + 1)p$  and  $\gcd(N, q) = \gcd(p, q) = 1$ .

**1 Step:**

- i) Choose private key polynomial  $f(x) \in \mathcal{T}(d + 1, d)$  that is invertible in  $R_p$  and  $R_q$ .
- ii) Choose private key polynomial  $g(x) \in \mathcal{T}(d, d)$ .
- iii) Compute  $F_p(x)$ , where  $F_p(x) = f^{-1} \pmod{p}$  in  $R_p$ .
- iv) Compute  $F_q(x)$ , where  $F_q(x) = f^{-1} \pmod{q}$  in  $R_q$ .
- v) Publish the public key polynomial  $h(x) \equiv F_q(x) * g(x) \pmod{q}$  on an open channel.

**Output:** Public key polynomial  $h(x)$  and private key polynomials  $(f(x), g(x), F_p(x), F_q(x))$ .

---



---

**Algorithm 2:** NTRUEncrypt Encryption Algorithm.

---

**Input:** Public key polynomial  $h(x)$  and sender’s message polynomial  $m(x) \in R_p$ .

**1 Step:**

- i) Select the plaintext polynomial  $m(x) \in R_p$ .
- ii) Choose a random polynomial  $r(x) \in \mathcal{T}(d, d)$  as the ephemeral key.
- iii) Encrypt the message polynomial  $m(x)$  using the recipient’s public key polynomial by computing,

$$e(x) \equiv p \cdot r(x) * h(x) + m(x) \pmod{q}.$$

- iv) Send the ciphertext polynomial  $e(x)$  to the recipient.

**Output:** Ciphertext polynomial  $e(x)$ .

---

A significant challenge with NTRUEncrypt lies in its nature as a public key cryptosystem. The sender cannot verify whether decryption will succeed, as verifying success requires access to the private key polynomial  $f(x)$ , a critical component in the NTRUEncrypt for decrypting ciphertext and recovering the original message [5]. To ensure successful decryption, parameters within the NTRUEncrypt must be carefully chosen to minimize the probability of decryption failure. Scholten and Vercauteren [12] developed a probabilistic analysis that identifies safe parameter bounds, helping reduce the likelihood of such failures. Enhancing and modifying the original NTRU-Encrypt involves addressing and reducing the likelihood of decryption failures, as the system’s practicality depends on maintaining a low probability of such occurrences.

**Algorithm 3:** NTRUEncrypt Decryption Algorithm.

**Input:** Ciphertext polynomial  $e(x)$  and private key polynomials  $(f(x), F_p(x))$ , where  $F_p(x) * f(x) \equiv 1 \pmod{p}$ .

**1 Step:**

- i) Compute  $f(x) * e(x) \pmod{q}$ .
- ii) Centerlift the polynomial  $f(x) * e(x) \pmod{q}$  to  $a(x) \in R$ .
- iii) Compute  $b(x) \equiv F_p(x) * a(x) \pmod{p}$ .
- iv) Centerlift the polynomial  $b(x)$  to recover the message polynomial  $m(x) \in R_p$ .

**Output:** Message polynomial  $m(x)$ .

One crucial step in the decryption process is centerlifting, which adjusts a polynomial to ensure its coefficients fall within the correct range. The centerlifting process is essential for accurately recovering the plaintext polynomial. The process is represented as,

$$a(x) = p \cdot r(x) * g(x) + m(x) * f(x). \tag{1}$$

Decryption fails if centerlifting is not performed correctly. Noise, denoted as  $\epsilon$ , may interfere with the polynomial  $a(x)$  during centerlifting, resulting in,

$$a'(x) = (p \cdot r(x) * g(x) + m(x) * f(x)) + \epsilon \cdot q. \tag{2}$$

Consequently, the recovered message becomes,

$$m(x) + \epsilon \cdot q \cdot f^{-1}(x) \pmod{p}. \tag{3}$$

Centerlifting is critical for ensuring decryption accuracy, as errors introduced during the centerlifting process can corrupt the recovered plaintext.

Successful decryption in NTRUEncrypt relies on carefully chosen parameters that keep coefficients within acceptable bounds and enable accurate recovery of plaintext. The following lemmas provide the theoretical foundation for ensuring coefficients remain within required bounds, facilitating accurate plaintext recovery and guiding the selection of secure and efficient parameters.

Lemma 2.1 establishes the conditions necessary to ensure decryption proceeds without errors caused by coefficients exceeding acceptable limits.

**Lemma 2.1.** [5] Let  $f(x)$  and  $e(x)$  be polynomials, and  $p, q, d \in \mathbb{Z}^+$ . If the receiver computes the polynomial  $f(x) * e(x) \pmod{q}$  and centerlift it to  $a(x)$ , where,

$$a(x) = \sum_{i=0}^{N-1} a_i x^i \quad \text{and} \quad a_i < \frac{q}{2}, \tag{4}$$

then the selected parameters  $(p, q, d)$  for NTRUEncrypt satisfy the condition  $q > (6d + 1)p$ .

Lemma 2.2 extends the principle of decryption reliability by demonstrating that, under appropriate conditions, the plaintext can always be recovered exactly.

**Lemma 2.2.** [5] If the parameters  $(p, q, d)$  for NTRUEncrypt satisfy  $q > (6d + 1)p$ , and the magnitude of each coefficient  $a_i$  in  $p \cdot r(x) * g(x) + f(x) * m(x)$  is strictly less than  $\frac{q}{2}$ , then the polynomial  $b(x)$  computed by the receiver in Algorithm 3 will exactly match the sender's plaintext  $m(x)$ .

Ternary polynomials are those with coefficients restricted to  $-1, 0,$  or  $1$ . Sparsity parameters  $d_g$  and  $d_r$  indicate the number of non-zero coefficients in  $g(x)$  and  $r(x)$ , respectively. A higher sparsity parameter corresponds to more non-zero coefficients. Parameters  $(N, p, q, d)$  must satisfy specific conditions to ensure the encryption scheme’s security and correctness. In particular,  $q$  must be sufficiently large to prevent coefficient overflow during polynomial operations. The condition  $q > (6d + 1)p$  ensures that  $q$  is large enough relative to  $p$  and  $d$ .

A magnitude condition ensures that polynomial coefficients after encryption remain within a specific range, avoiding decryption errors. Specifically, the requirement that the magnitude of each coefficient  $a_i$  in the expression  $p \cdot r(x) * g(x) + f(x) * m(x)$  is strictly less than  $\frac{q}{2}$  is critical for the decryption process to function correctly and to prevent overflow errors [5]. To provide insight into the impact of parameter selection, the following section includes a numerical example illustrating decryption failure caused by coefficient overflow.

**2.1.1 Numerical example demonstrating decryption failure in NTRUEncrypt**

This section presents a numerical example of decryption failure in NTRUEncrypt due to coefficient overflow. The example demonstrates how inappropriate parameter selection can lead to decryption errors and highlights the importance of carefully choosing parameters to ensure reliable decryption. Key generation, encryption, and decryption processes are central to NTRUEncrypt. The following example offers a simplified step-by-step explanation adapted from [5], who provide a formalization of these algorithms in their foundational work on lattice-based cryptography.

---

**Step 1: Key Generation Process.**

---

**Input:** Selection of public parameters  $(N = 7, p = 3, q = 11, d = 2)$  in NTRUEncrypt, where  $\gcd(N, q) = \gcd(p, q) = 1$  and  $q$  does not satisfy the condition  $q > (6d + 1)p$ .

**1 Step:**

- i) Define the private key polynomial  $f(x) = x^6 - x^4 + x^3 + x^2 - 1$ .
- ii) Define the private key polynomial  $g(x) = x^6 + x^4 - x^2 - x$ .
- iii) Compute  $F_p(x) = f^{-1} \equiv x^6 + 2x^5 + x^3 + x^2 + x + 1 \pmod{3}$ .
- iv) Compute  $F_q(x) = f^{-1} \equiv 6x^6 + x^4 + 10x^3 + 2x^2 + 7x + 8 \pmod{11}$ .
- v) Compute the public key polynomial,

$$h(x) = F_q(x) * g(x) \equiv 9x^6 + 2x^5 + 7x^4 + 9x^3 + 6x^2 \pmod{11}.$$

**Output:** Public key polynomial  $h(x)$  and private key polynomials  $(f(x), g(x), F_p(x), F_q(x))$ .

---

**Step 2: Encryption Process.**

**Input:** Public key polynomial  $h(x) \equiv 9x^6 + 2x^5 + 7x^4 + 9x^3 + 6x^2 \pmod{11}$  and sender's message polynomial  $m(x) = -x^5 + x^3 + x^2 - x + 1$ .

**1 Steps:**

- i) Define the ephemeral key  $r(x) = x^6 - x^5 + x - 1$ .
- ii) Encrypt the message polynomial  $m(x)$  using the recipient's public key polynomial by computing,

$$e(x) \equiv p \cdot g(x) * r(x) + m(x) \pmod{q}$$

$$= x^6 + 8x^5 + 7x^4 + 7x^3 + 11x^2 + x + 10.$$

- iii) Send the ciphertext polynomial  $e(x)$  to the recipient.

**Output:** Ciphertext polynomial  $e(x)$ .

**Step 3: Decryption Process.**

**Input:** Ciphertext polynomial  $e(x)$  and private key polynomials  $(f(x), F_p(x))$ , where  $F_p(x) * f(x) \equiv 1 \pmod{p}$ .

**1 Step:**

- i) Compute:

$$f(x) * e(x) \equiv p \cdot g(x) * r(x) + f(x) * m(x) \pmod{q}$$

$$= x^6 + 10x^5 + 3x^4 + 10x^3 + 10x^2 + x + 10.$$

- ii) Centerlift  $f(x) * e(x)$  to  $a(x) \in R$ ,

$$a(x) = x^6 - x^5 + 3x^4 - x^3 - x^2 + x - 1.$$

- iii) Compute  $b(x) \equiv F_p(x) * a(x) \pmod{p}$ ,

$$b(x) = 2x^5 + 2x^3 + 2x^2 + 1.$$

- iv) Centerlift  $b(x)$  to obtain the message polynomial,

$$m'(x) = -x^5 - x^3 - x^2 + 1.$$

**Output:** Decrypted message polynomial  $m'(x)$  does not match the original  $m(x)$ .

The recovered message  $m'(x) = -x^5 - x^3 - x^2 + 1$  does not match the original message,  $m(x) = -x^5 + x^3 + x^2 - x + 1$ . The discrepancy between the recovered and original messages is caused by coefficient overflow, where the coefficients of  $a(x)$  lie outside the interval  $(-5, 5]$ , leading to incorrect centerlifting.

Lemma 2.2 states that the magnitude of each coefficient  $a_i$  in  $p \cdot r(x) * g(x) + f(x) * m(x)$  must be strictly less than  $\frac{q}{2}$ . We find,

$$a(x) = p \cdot r(x) * g(x) + f(x) * m(x) = x^6 + 10x^5 - 8x^4 - x^3 - x^2 + x - 1. \tag{5}$$

Here, the coefficients  $a_4 = -8$  and  $a_5 = 10$  fall outside the interval  $(-5, 5]$ , which produces coeffi-

cient overflow.

The example above highlights the critical importance of selecting an appropriate modulus  $q$  in NTRUEncrypt. A modulus  $q = 11$  is insufficient for the given parameters, leading to decryption failure due to coefficient overflow. To ensure reliable decryption,  $q$  must satisfy the condition  $q > (6d + 1)p$ , where  $d$  represents the number of nonzero coefficients in  $f(x)$ ,  $g(x)$ , and  $r(x)$ . Meeting the condition  $q > (6d + 1)p$  ensures that the modulus is large enough to accommodate noise growth during encryption, enabling successful recovery of the original message.

Building on the foundational structure and key parameters discussed earlier, the reliability of NTRUEncrypt extends beyond theoretical design. The success of decryption depends on factors such as the behavior of polynomial coefficients  $a_i$ , the influence of noise, and the careful tuning of parameters. The interplay between polynomial coefficient behavior, noise impact, and parameter adjustments is essential for maintaining accuracy and robustness in the cryptosystem.

In the following section, we examine how statistical tools, such as the normal distribution and cumulative distribution function (CDF), provide insights into the probability of successful decryption and overall system reliability.

## 2.2 Factors influencing decryption success in NTRUEncrypt

In the context of NTRUEncrypt, several factors significantly influence the probability of decryption success. Bindel and Schanck [2] demonstrated that the distribution and randomness of coefficients directly affect the likelihood of decryption failures, highlighting the importance of careful parameter selection. Proper coefficient distribution within the NTRUEncrypt framework ensures that all coefficients  $a_i$  of the computed polynomial  $a(x)$  remain within the required interval  $\left[-\frac{q-1}{2}, \frac{q-1}{2}\right]$ , thereby avoiding overflow and preserving decryption correctness. This is confirmed by the data in Table 2, where all  $a_i$  values from successful decryption samples fall within the expected range.

Table 2: Sample polynomial pairs  $m(x)$  and  $a(x)$  under parameters  $N = 7, p = 3, q = 41, d = 2$ .

<b>Private Key</b> $f(x) = x^6 - x^4 + x^3 + x^2 - 1$
<b>Inverse of <math>f(x)</math> mod <math>p</math>:</b> $F_p(x) = x^6 + 2x^5 + x^3 + x^2 + x + 1$
<b>Inverse of <math>f(x)</math> mod <math>q</math>:</b> $F_q(x) = 8x^6 + 26x^5 + 31x^4 + 21x^3 + 40x^2 + 2x + 37$
<b>Polynomial</b> $g(x) = x^6 + x^4 - x^2 - x$
<b>Public Key</b> $h(x) = 19x^6 + 38x^5 + 6x^4 + 32x^3 + 24x^2 + 37x + 8$
<b>Random Polynomial</b> $r(x) = x^6 - x^5 + x - 1$

No.	$m(x)$	$a(x)$
1	$-x^5 + x^3 + x^2 - x + 1$	$x^6 + 10x^5 - 8x^4 - x^3 - x^2 + x - 1$
2	$x^5 + x^3 + x^2 - x + 1$	$x^6 + 8x^5 - 6x^4 - x^3 - 3x^2 + 3x + 1$
3	$-x^5 - x^3 + x^2 - x + 1$	$-x^6 + 8x^5 - 8x^4 + x^3 - 3x^2 + x + 1$
4	$x^5 - x^3 + x^2 - x + 1$	$-x^6 + 6x^5 - 6x^4 + x^3 - 5x^2 + 3x + 3$
5	$x^6 - x^5 - x^3 + x^2 - x + 1$	$-2x^6 + 9x^5 - 8x^4 - 2x^2 + 2x + 1$
6	$x^6 + x^5 - x^3 + x^2 - x + 1$	$-2x^6 + 7x^5 - 6x^4 - 4x^2 + 4x + 3$
7	$x^6 - x^5 + x^3 + x^2 - x + 1$	$11x^5 - 8x^4 - 2x^3 + 2x - 1$
8	$x^6 - x^5 + x^3 - x^2 - x + 1$	$2x^6 + 9x^5 - 10x^4 - 2x^3 + 2x^2 - 1$
9	$x^6 - x^5 - x^3 - x^2 - x + 1$	$7x^5 - 10x^4 + 1$
10	$x^6 + x^5 - x^3 - x^2 - x + 1$	$5x^5 - 8x^4 - 2x^2 + 2x + 3$
11	$-x^6 + x^5 - x^3 - x^2 - x + 1$	$2x^6 + 3x^5 - 8x^4 + 2x^3 - 4x^2 + 3$
12	$-x^6 - x^5 - x^3 - x^2 - x + 1$	$2x^6 + 5x^5 - 10x^4 + 2x^3 - 2x^2 - 2x + 1$
13	$-x^6 - x^5 + x^3 - x^2 - x + 1$	$4x^6 + 7x^5 - 10x^4 - 2x - 1$
14	$-x^6 - x^5 + x^4 + x^3 - x^2 - x + 1$	$5x^6 + 7x^5 - 11x^4 + x^3 - 3x$
15	$-x^6 - x^5 - x^4 + x^3 - x^2 - x + 1$	$3x^6 + 7x^5 - 9x^4 - x^3 - x - 2$

The histogram in Figure 1 further supports this observation, showing that the distribution of  $a_i$  values is centered around zero with low variance, indicating a stable and well-balanced structure that favors successful decryption. The bell-shaped curve overlaying the histogram resembles a normal distribution, suggesting that the coefficients are evenly spread around the center, with extreme values occurring only rarely.

Applying probability theory and statistical analysis, as demonstrated by Silverman and Whyte [13], provides a solid framework for understanding and quantifying the likelihood of successful decryption in NTRUEncrypt. In their work, this approach was used to examine uncertainties and variations that may affect decryption outcomes.

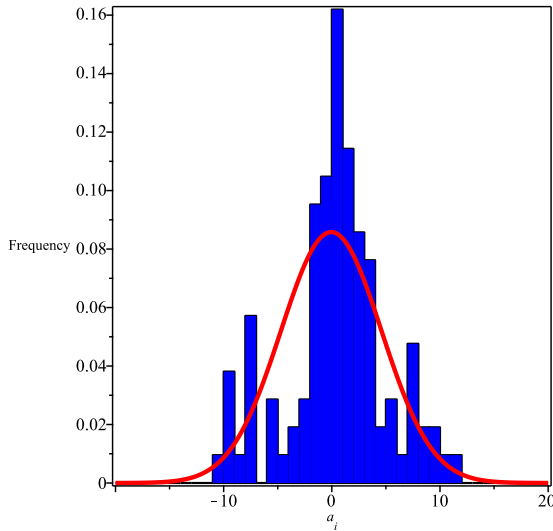


Figure 1: Histogram of  $a_i$  coefficients for  $q = 41$ .

### 2.2.1 Normal distribution in NTRUEncrypt

Understanding the distribution of polynomial coefficients in NTRUEncrypt is essential, particularly in the context of the normal (Gaussian) distribution. Assuming a coefficient follows a normal (Gaussian) distribution, which is symmetric around the mean, data points are more likely to occur closer to the mean than farther away, forming a bell-shaped curve. Since the normal (Gaussian) distribution is centered around its mean, the mean  $\mu$  of the coefficient distribution is crucial for assessing the likelihood of successful decryption [5]. The central limit theorem further supports the idea that the sum of many independent and identically distributed variables tends to form a normal (Gaussian) distribution, which applies to the coefficients in an NTRUEncrypt polynomial [13]. This assumption is supported by Figure 1, which shows the coefficient distribution centered around zero and closely resembling the shape of a normal distribution.

In [2], it was shown that even after a message is successfully decrypted, it’s still possible for an attacker to increase the chance of a later decryption failure by carefully choosing what to send next. This idea of "failure boosting" highlights why we pay close attention to how our coefficients are distributed, making sure they stay within safe limits to keep decryption reliable.

### 2.2.2 Cumulative distribution function (CDF) in NTRUEncrypt

The cumulative distribution function (CDF) measures the probability that coefficients remain within an acceptable range. For a coefficient  $a_i$ , the CDF  $F(x)$  is defined as,

$$F(x) = P(a_i \leq x) \text{ for } x \in \left( -\frac{q-1}{2}, \frac{q-1}{2} \right]. \tag{6}$$

In [5], the standard interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$  for coefficients in NTRUEncrypt is described. Based on this, we define the CDF  $F(x)$  to express the probability that a coefficient  $a_i$  lies within this range. In [8], a similar probabilistic view was used to assess how often coefficients remained within safe limits for successful decryption.

### 3 Distribution of Coefficients in NTRU Polynomial

In Section 3, we explore the relationships between statistical measures and the distribution of coefficients in the NTRUEncrypt. By examining key lemmas, theorems, and propositions, we identify critical factors that determine the probability of all  $a_i$  coefficients falling within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$ .

In NTRUEncrypt, understanding the behavior of polynomial coefficients is essential for reliable decryption. Our analysis focuses on the interplay between coefficient distribution and randomness, emphasizing their roles in influencing the predictability and reliability of decryption outcomes. Guided by Propositions 3.1, Propositions 3.2, Propositions 3.3, Propositions 3.4, and Theorem 3.1, we identify key elements that ensure decryption success within the original NTRUEncrypt framework. This section provides a detailed understanding of the factors that enhance the probability of all  $a_i$  coefficients remaining within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$ .

The terms  $g_t$  and  $r_u$  represent the product of individual coefficients from  $g(x)$  and  $r(x)$ , respectively. The product  $g_t r_u$  equals  $\pm 1$  when both  $g_t$  and  $r_u$  are non-zero and have either the same or opposite signs. The probability  $P(g_t r_u = \pm 1)$  quantifies how often non-zero products occur in the polynomial  $p \cdot r(x) * g(x) + f(x) * m(x)$ . Understanding the frequency of non-zero products in  $p \cdot r(x) * g(x) + f(x) * m(x)$  is crucial for ensuring correct decryption and preventing overflow errors.

In NTRUEncrypt, sparsity parameters  $d_g$  and  $d_r$  are algebraically related to the private key  $d$ . Specifically,  $g(x) \in \mathcal{T}(d, d)$  and  $r(x) \in \mathcal{T}(d, d)$ , where  $\mathcal{T}(d_1, d_2)$  denotes a ternary polynomial with  $d_1$  positive and  $d_2$  negative coefficients.

**Proposition 3.1.** For ternary polynomials  $g(x) = \sum_{t=0}^{N-1} g_t x^t$  and  $r(x) = \sum_{u=0}^{N-1} r_u x^u$ , let  $d_g$  and  $d_r$  represent sparsity parameters, where  $g(x) \in \mathcal{T}(d, d)$  and  $r(x) \in \mathcal{T}(d, d)$ , and let  $N$  denote the degree of the polynomials. In the context of NTRUEncrypt, if the parameters  $(N, p, q, d)$  satisfy  $q > (6d + 1)p$  and the magnitude of each coefficient in  $p \cdot r(x) * g(x) + f(x) * m(x)$  is strictly less than  $\frac{q}{2}$ , then the probability of  $g_t r_u = \pm 1$  for  $t, u = 0, 1, \dots, N - 1$  is given by,

$$P(g_t r_u = \pm 1) = \frac{4d_g d_r}{N^2}. \tag{7}$$

*Proof.* To establish Proposition 3.1, we first revisit the expression for  $a(x)$  in the original NTRUEncrypt decryption process,

$$a(x) = p \cdot g(x) * r(x) + f(x) * m(x). \tag{8}$$

Polynomial  $a(x)$ , with coefficients  $a_i$  for  $i = 0, 1, \dots, N - 1$ , is derived from the convolution of ternary polynomials  $f(x)$ ,  $g(x)$ ,  $r(x)$ , and the message polynomial  $m(x)$ . Focusing on the coefficients associated with  $g(x)$  and  $r(x)$ , ternary polynomial  $g(x)$  has coefficients  $g_t$ , and  $r(x)$  has coefficients  $r_u$ , both considered pairwise independent random variables. The coefficient  $a_i$  can be expressed as,

$$a_i = p \sum_{t+u=i \pmod N} g_t r_u + \sum_{v+j=i \pmod N} f_v m_j. \tag{9}$$

Following [5], we define ternary polynomials  $g(x)$  and  $r(x)$  as,

$$g(x) \in \mathcal{T}(d_g, d_g), \tag{10}$$

$$r(x) \in \mathcal{T}(d_r, d_r). \tag{11}$$

The coefficients of ternary polynomials follow specific probability distributions. For  $g(x)$ , the probabilities are

$$P(g_t = 1) = \frac{d_g}{N}, \tag{12}$$

$$P(g_t = -1) = \frac{d_g}{N}, \tag{13}$$

$$P(g_t = 0) = 1 - \frac{2d_g}{N}, \tag{14}$$

and for  $r(x)$ , the probabilities are

$$P(r_u = 1) = \frac{d_r}{N}, \tag{15}$$

$$P(r_u = -1) = \frac{d_r}{N}, \tag{16}$$

$$P(r_u = 0) = 1 - \frac{2d_r}{N}. \tag{17}$$

The probability of non-zero coefficients being  $-1$  or  $1$  for  $g_t$  is  $P(g_t = \pm 1) = \frac{2d_g}{N}$ , and for  $r_u$ , it is  $P(r_u = \pm 1) = \frac{2d_r}{N}$ . Assuming pairwise independence of all  $g_t$  and  $r_u$  for  $t, u = 0, 1, \dots, N - 1$ , the probability of  $g_t r_u = \pm 1$  can be calculated as,

$$P(g_t r_u = \pm 1) = \frac{4d_g d_r}{N^2}. \tag{18}$$

This concludes the proof. □

In the transition from Proposition 3.1 to Proposition 3.2, we further analyze the implications of sparsity parameters  $d_g$  and  $d_r$  for the ternary polynomials  $g(x)$  and  $r(x)$ , respectively. The norm of the polynomial  $p \cdot r(x) * g(x) + f(x) * m(x)$  is influenced by the presence of zero products. A higher probability of zero products  $g_t r_u = 0$  results in a lower overall polynomial norm, which is advantageous for ensuring that the polynomial’s coefficients remain within the required range for correct decryption.

**Proposition 3.2.** For ternary polynomials  $g(x) = \sum_{t=0}^{N-1} g_t x^t$  and  $r(x) = \sum_{u=0}^{N-1} r_u x^u$ , let  $d_g$  and  $d_r$  represent the sparsity parameters, where  $g(x) \in \mathcal{T}(d, d)$  and  $r(x) \in \mathcal{T}(d, d)$ , and let  $N$  be the degree of the polynomials. In the context of NTRUEncrypt, if parameters  $(N, p, q, d)$  satisfy  $q > (6d + 1)p$  and the magnitude

of each coefficient in  $p \cdot r(x) * g(x) + f(x) * m(x)$  is strictly less than  $\frac{q}{2}$ , then the probability of  $g_t r_u = 0$  for  $t, u = 0, 1, \dots, N - 1$  is given by,

$$P(g_t r_u = 0) = 1 - \frac{4d_g d_r}{N^2}. \tag{19}$$

*Proof.* To establish Proposition 3.2, we build on the result from Proposition 3.1. Let  $P_1$  denote the probability that  $g_t r_u = 0$  and  $P_2$  the probability that  $g_t r_u = \pm 1$ . Then, the relationship is given by,

$$P_1 + P_2 = 1. \tag{20}$$

From Proposition 3.1, we know,

$$P_2 = \frac{4d_g d_r}{N^2}. \tag{21}$$

Substituting (21) into (20), we obtain,

$$P_1 + \frac{4d_g d_r}{N^2} = 1. \tag{22}$$

Solving for  $P_1$ , we find,

$$P_1 = 1 - \frac{4d_g d_r}{N^2}. \tag{23}$$

This concludes the proof. □

Proposition 3.3 focuses on the probability of  $f_v m_j$  falling within the range  $\mu \in \left(-\frac{p}{2}, \frac{p}{2}\right)$  for ternary polynomials  $f(x)$  and  $m(x)$ . Proposition 3.3 provides a formula to calculate the probability of  $f_v m_j$  lying within the specified range, aiding in the analysis of the statistical properties of the individual coefficients' products  $f_v m_j$ . By linking parameters  $d_f, \alpha_m$ , and  $N$ , Proposition 3.3 sheds light on the probabilistic behavior of polynomial coefficients in NTRUEncrypt.

**Proposition 3.3.** Let  $d_f$  denote the sparsity parameter of ternary polynomial  $f(x) = \sum_{v=0}^{N-1} f_v x^v$ ,  $\alpha_m$  represent the number of coefficients  $m_j = \mu \in \left(-\frac{p}{2}, \frac{p}{2}\right)$ , and  $N$  be the degree of the polynomials. If NTRUEncrypt with parameters  $(N, p, q, d)$  satisfies  $q > (6d + 1)p$ , the magnitude of each coefficient in  $p \cdot r(x) * g(x) + f(x) * m(x)$  is strictly less than  $\frac{q}{2}$ , and  $m(x) \in R_p$ , then the probability of

$$f_v m_j = \mu \in \left(-\frac{p}{2}, \frac{p}{2}\right), \text{ for } i, j = 0, 1, \dots, N - 1,$$

is given by,

$$P \left[ f_v m_j = \mu \mid \mu \in \left(-\frac{p}{2}, \frac{p}{2}\right) \right] = \frac{\alpha_m}{N}. \tag{24}$$

*Proof.* To establish Proposition 3.3, we first revisit the expression for  $a(x) = \sum_{i=0}^{N-1} a_i x^i$  in the original NTRUEncrypt decryption process,

$$a(x) = p \cdot g(x) * r(x) + f(x) * m(x). \tag{25}$$

Polynomial  $a(x)$ , with coefficients  $a_i$  for  $i = 0, 1, \dots, N - 1$ , is derived from the convolution of ternary polynomials  $f(x)$  and  $g(x)$  with  $r(x)$  and the message polynomial  $m(x)$ . We focus on the term  $f_v m_j$ . Note that the ternary polynomial  $f(x)$  has coefficients  $f_v$ , which are considered pairwise independent random variables. Coefficients of  $m(x)$  are represented by  $\mu$  and are chosen from the interval  $(-\frac{p}{2}, \frac{p}{2})$ , as described by [5]. Independence of coefficients  $f_v$  in  $f(x)$  and the range  $(-\frac{p}{2}, \frac{p}{2})$  for coefficients  $\mu$  in  $m(x)$  allow us to analyze the coefficients  $f_v m_j$  in  $a_i$ .

The coefficient  $a_i$  can be expressed as,

$$a_i = p \sum_{t+u=i \pmod N} g_t r_u + \sum_{v+j=i \pmod N} f_v m_j. \tag{26}$$

Following [5], we define ternary polynomial  $f(x)$  as,

$$f(x) \in \mathcal{T}(d + 1, d). \tag{27}$$

Ternary polynomials follow specific probability distributions. Specifically, for  $f(x)$ ,

$$P(f_v = 1) = \frac{d_f + 1}{N}, \tag{28}$$

$$P(f_v = -1) = \frac{d_f}{N}, \tag{29}$$

$$P(f_v = 0) = 1 - \frac{2d_f + 1}{N}. \tag{30}$$

As described by Hoffstein et al. [5], the polynomial  $m(x)$  has coefficients  $m_j$  constrained within the interval  $\mu \in (-\frac{p}{2}, \frac{p}{2})$  and here  $\alpha_m$  represents the number of coefficients  $m_j$  equal to  $\mu$ . Since  $N$  is the degree of the highest exponent in the polynomial, the probability distribution for  $m_j = \mu$  is

$$P(m_j = \mu) = \frac{\alpha_m}{N}. \tag{31}$$

The probability of non-zero coefficients being  $-1$  or  $1$  out of  $N$  for  $f_v$  is  $P(f_v = \pm 1) = \frac{2d_f + 1}{N}$ . Assuming all  $f_v$  and  $m_j$  are pairwise independent random variables for  $i, j = 0, 1, \dots, N - 1$  and  $\mu = -\frac{p-1}{2}, \dots, \frac{p-1}{2}$ , the probability of  $f_v m_j = \mu$  is computed as,

$$P(f_v m_j = \mu) = \frac{(2d_f + 1)\alpha_m}{N^2} + \frac{\alpha_m}{N} - \frac{(2d_f + 1)\alpha_m}{N^2} \tag{32}$$

$$= \frac{\alpha_m}{N}. \tag{33}$$

This concludes the proof. □

Proposition 3.4 addresses the relationship between coefficient variance and parameters of ternary polynomials, specifically sparsity parameters  $(d_f, \alpha_m, d_g)$ , polynomial degree  $N$ , and modulus values  $(p, q)$ . By analyzing how parameters  $(d_f, \alpha_m, d_g, N, p, q)$  influence coefficient variance, Proposition 3.4 demonstrates the importance of controlling parameters  $(d_f, \alpha_m, d_g, N, p, q)$  to improve the probability of all  $a_i$  coefficients falling within the interval  $(-\frac{q-1}{2}, \frac{q-1}{2})$ .

**Proposition 3.4.** Let  $d_f, \alpha_m, N$ , and  $\text{Var}(a_i)$  denote the sparsity parameter of ternary polynomial

$f(x) = \sum_{v=0}^{N-1} f_v x^v$ , the number of coefficients  $m_j = \mu \in \left(-\frac{p}{2}, \frac{p}{2}\right)$ , the degree of the polynomial  $N$ , and the coefficient variance of  $a_i$ , respectively. If NTRUEncrypt with parameters  $(N, p, q, d)$  satisfies  $q > (6d+1)p$ , the magnitude of each coefficient in  $p \cdot r(x) * g(x) + f(x) * m(x)$  is strictly less than  $\frac{q}{2}$ , and  $m(x) \in R_p$ , then the coefficient variance of terms  $a_i = p \cdot g(x) * r(x) + f(x) * m(x)$  with  $\mu \in \left(-\frac{p}{2}, \frac{p}{2}\right)$  and  $i = 0, 1, \dots, N-1$  is given by,

$$\text{Var}(a_i) \Big|_{\mu \in \left(-\frac{p}{2}, \frac{p}{2}\right)} = \frac{8d_g d_r p^2}{N} + \frac{\alpha_m (p-1)^2}{2}. \tag{34}$$

*Proof.* Using probabilities associated with coefficients  $a_i$  from Propositions 3.1, 3.2 and 3.3 for  $g_t r_u$  and  $f_v m_j$ , we have

$$P(g_t r_u = \pm 1) = \frac{4d_g d_r}{N^2}, \tag{35}$$

$$P(g_t r_u = 0) = 1 - \frac{4d_g d_r}{N^2}, \tag{36}$$

$$P(f_v m_j = \mu) = \frac{\alpha_m}{N}, \tag{37}$$

respectively.

Using probabilities  $P(g_t r_u = \pm 1)$ ,  $P(g_t r_u = 0)$  and  $P(f_v m_j = \mu)$ , along with properties of variance, we calculate the coefficient variances of terms in  $a_i$ ,

$$\text{Var}(g_t r_u) = E((g_t r_u)^2) - E(g_t r_u)^2 \tag{38}$$

$$= \left(\frac{4d_g d_r}{N^2} \cdot 1^2 + \frac{4d_g d_r}{N^2} \cdot (-1)^2 + \left(1 - \frac{4d_g d_r}{N^2}\right) \cdot 0^2\right) \tag{39}$$

$$= \frac{8d_g d_r}{N^2}, \tag{40}$$

and

$$\text{Var}(f_v m_j) = E((f_v m_j)^2) - E(f_v m_j)^2 \tag{41}$$

$$= \frac{\alpha_m}{N} \left(\frac{-(p-1)^2}{2} + \frac{(p-1)^2}{2}\right) \tag{42}$$

$$= \frac{\alpha_m}{N} \left(\frac{p^2 - 2p + 1 + p^2 - 2p + 1}{4}\right) \tag{43}$$

$$= \frac{\alpha_m}{N} \left(\frac{2p^2 - 4p + 2}{4}\right) \tag{44}$$

$$= \frac{\alpha_m (p-1)^2}{2N}. \tag{45}$$

For two independent random variables  $A$  and  $B$ , recall that  $\text{Var}(A + B) = \text{Var}(A) + \text{Var}(B)$ . Assuming independence of all products  $g_t r_u$  and  $f_v m_j$ , we compute,

$$\text{Var}\left(\sum_{t+u=i \pmod N} g_t r_u\right) = N \frac{8d_g d_r}{N^2} = \frac{8d_g d_r}{N}, \tag{46}$$

and

$$\text{Var} \left( \sum_{v+j=i \pmod N} f_v m_j \right) = N \frac{\alpha_m (p-1)^2}{2N} = \frac{\alpha_m (p-1)^2}{2}. \tag{47}$$

Finally, using  $\text{Var}(pX) = p^2 \text{Var}(X)$ , for coefficients  $a_i$  resulting from summation of terms involving products of coefficients from  $g_t, r_u, f_v$ , and  $m_j$ , we conclude:

$$\text{Var}(a_i) = p^2 \text{Var} \left( \sum_{t+u=i \pmod N} g_t r_u \right) + \text{Var} \left( \sum_{v+j=i \pmod N} f_v m_j \right) \tag{48}$$

$$= \frac{8d_g d_r p^2}{N} + \frac{\alpha_m (p-1)^2}{2}. \tag{49}$$

Thus, the coefficient variance  $\text{Var}(a_i)$ , governing the probability of  $a_i$  in NTRUencrypt, where  $|a_i| < \frac{q}{2}$ , concludes the proof. □

Proposition 3.4 demonstrates that, in NTRUencrypt, the likelihood of all  $a_i$  coefficients falling within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$  heavily depends on the variance of the coefficients of  $a(x)$ ,

denoted as  $\text{Var}(a_i)$ . When the coefficients of  $a(x) = \sum_{i=0}^{N-1} a_i x^i$  follow a normal distribution, the

probability of all  $a_i$  coefficients falling within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$  is very high. A high probability ensures a successful decryption process by keeping the coefficients within the required range, as referenced in Lemma 2.2.

Next, we introduce Theorem 3.1, which investigates the probability of all  $a_i$  coefficients falling within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$ . Assuming  $a_i$ , random coefficients in  $a(x)$ , are independent random variables following a normal distribution, Theorem 3.1 establishes a relationship between the probability of each coefficient  $a_i$  falling within the required range for decryption and the parameters of the NTRUencrypt system. Understanding the relationship between coefficient variance and its impact on reliability and the success rate of all  $a_i$  coefficients falling within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$  is crucial.

**Theorem 3.1.** Let  $\sigma(a_i)$  denote the standard deviation of  $a_i$ , where  $a_i = \sum_{i=0}^{N-1} a_i x^i$  represents independent random variables following a normal distribution  $\mathcal{N}(0, \sigma^2(a_i))$ . If NTRUencrypt parameters  $(N, p, q, d)$  satisfy  $q > (6d + 1)p$  and all coefficients  $a_i$  satisfy the condition  $|a_i| < \frac{q}{2}$  for  $i = 0, 1, \dots, N - 1$ , then the probability  $P \left( |a_i| \leq \frac{q-1}{2} \right)$  is given by,

$$P \left( |a_i| \leq \frac{q-1}{2} \right) = 2\Phi \left( \frac{q-1}{2\sqrt{\frac{8d_g d_r p^2}{N} + \frac{\alpha_m (p-1)^2}{2}}} \right), \tag{50}$$

where  $\Phi$  is the cumulative distribution function (CDF) of the standard normal distribution and  $i = 0, 1, \dots, N - 1$ .

*Proof.* Assume that  $a_i$  are independent random variables distributed according to  $\mathcal{N}(0, \sigma^2(a_i))$ . The probability that  $a_i$  falls within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$ , as referenced in Theorem 3.1, is given by,

$$P\left(-\frac{q-1}{2} \leq a_i \leq \frac{q-1}{2}\right) = P\left(|a_i| \leq \frac{q-1}{2}\right). \tag{51}$$

For  $\mathcal{N}(0, \sigma^2(a_i))$ , the probability that  $a_i$  falls within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$  can be expressed using the CDF of the standard normal distribution  $\Phi$ ,

$$P\left(|a_i| \leq \frac{q-1}{2}\right) = 2\Phi\left(\frac{q-1}{2\sigma(a_i)}\right). \tag{52}$$

From Proposition 3.4, the variance  $\text{Var}(a_i)$  is the square of the standard deviation  $\sigma(a_i)$ , which gives,

$$\sigma(a_i) = \sqrt{\frac{8d_g d_r p^2}{N} + \frac{\alpha_m(p-1)^2}{2}}. \tag{53}$$

Substituting  $\sigma(a_i)$  into (52), the probability that  $|a_i| \leq \frac{q-1}{2}$  for successful decryption becomes,

$$P\left(|a_i| \leq \frac{q-1}{2}\right) = 2\Phi\left(\frac{q-1}{2\sqrt{\frac{8d_g d_r p^2}{N} + \frac{\alpha_m(p-1)^2}{2}}}\right), \tag{54}$$

where  $i = 0, 1, \dots, N - 1$ . □

Theorem 3.1 establishes the probability that all coefficients  $a_i$  in NTRUEncrypt remain within the required interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$ . Ensuring that all coefficients stay within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$  is essential for decryption reliability, as coefficients outside the interval can result in decryption failure.

Section 3 examined factors influencing the distribution of coefficients in NTRUEncrypt and emphasized the importance of controlling coefficient variance. In Section 4, we move into a discussion of how coefficient variance directly impacts decryption success by analyzing its effect on the likelihood of coefficients remaining within the required range. This discussion builds upon the statistical foundations established in Section 3, focusing on theoretical approaches to optimizing the performance and security of NTRUEncrypt.

### 4 Impact of Coefficient Variance on Decryption Success in NTRUEncrypt

In NTRUEncrypt, understanding the distribution and behavior of polynomial coefficients is essential to ensure that all  $a_i$  coefficients remain within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$ . Coefficient

variance, which measures the spread of  $a_i$  coefficients in the polynomial  $a(x)$ , plays a critical role in determining the reliability and accuracy of the decryption process. Specifically, the variance of coefficients in  $a(x)$  is a key factor in assessing the probability that all  $a_i$  coefficients remain within the required interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$ .

This section examines the role of coefficient variance in NTRUEncrypt from three perspectives. Subsection 4.1 emphasizes the importance of minimizing variance to improve decryption reliability and reduce errors. Subsection 4.2 explores the mathematical relationship between variance and the probability of coefficients remaining within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$ , highlighting the impact of parameter selection. Subsection 4.3 addresses the security implications of high variance, including vulnerabilities such as cyclic shift attacks, and discusses strategies to mitigate risks like decryption failures and potential private key exposure through careful parameter tuning.

#### 4.1 Significance of coefficient variance in NTRUEncrypt

Coefficient variance quantifies how much the coefficients  $a_i$  deviate from their mean value. In NTRUEncrypt, lower coefficient variance indicates that the coefficients are more tightly clustered around the mean. The tighter grouping of coefficients around the mean, resulting from lower coefficient variance, is critical for keeping the coefficients within the required range for successful decryption, specifically  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$ .

A lower coefficient variance directly correlates with higher reliability in the decryption process. When coefficients are less spread out, the probability of any coefficient falling outside the acceptable range decreases, ensuring the decryption operation proceeds correctly. Coefficient variance also impacts the precision of decryption. A lower coefficient variance results in computed values that are more consistent and closer to their expected values. Consistency is crucial for the decryption algorithm to function accurately and produce the correct plaintext. In NTRUEncrypt systems, maintaining low coefficient variance enhances security by minimizing the spread of coefficients, thereby reducing the risk of outliers that could lead to decryption errors or vulnerabilities. With an understanding of the importance of minimizing coefficient variance, the next subsection discusses the mathematical foundations linking variance and parameter selection to decryption success.

#### 4.2 Impact of coefficient distribution on decryption success in NTRUEncrypt

Theorem 3.1 emphasizes the importance of controlling the distribution of polynomial coefficients in NTRUEncrypt to ensure that all  $a_i$  coefficients fall within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$ . By modeling the  $a_i$  coefficients of the polynomial  $a(x)$  as independent random variables with a normal distribution  $\mathcal{N}(0, \sigma^2(a_i))$ , following the approach outlined by [5], we can calculate the probability that each coefficient remains within the required range  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$  for successful decryption.

Theorem 3.1 provides a formula to determine the probability that coefficients  $a_i$  fall within

the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$ , linking the probability of  $a_i$  coefficients staying within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$  to the coefficient variance  $\text{Var}(a_i)$ . A smaller coefficient variance increases the likelihood of coefficients remaining within the required bounds, as coefficients are more tightly clustered around the mean. Smaller coefficient variance also reduces the chances of outliers causing decryption errors in NTRUEncrypt.

The probability formula is as follows,

$$P\left(|a_i| \leq \frac{q-1}{2}\right) = 2\Phi\left(\frac{q-1}{2\sqrt{\frac{8d_g d_r p^2}{N} + \frac{\alpha_m(p-1)^2}{2}}}\right), \tag{55}$$

where  $\Phi$  represents the cumulative distribution function (CDF) of the standard normal distribution.  $P\left(|a_i| \leq \frac{q-1}{2}\right)$  demonstrates how the parameters of NTRUEncrypt influence the likelihood of  $a_i$  coefficients remaining within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$ . Understanding the relationship between parameters and the probability of coefficients staying within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$  allows for better selection of  $N, p, q, d_g, d_r,$  and  $\alpha_m$  to optimize performance and ensure reliable decryption in NTRUEncrypt.

Building on the probabilistic framework linking parameter selection to decryption success, the next subsection discusses how high coefficient variance can lead to security risks, particularly vulnerabilities to cyclic shift attacks.

### 4.3 Coefficient variance in NTRUEncrypt security

Coefficient variance affects the security and reliability of NTRUEncrypt. A high coefficient variance increases the likelihood of coefficients falling outside the required range  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$ , potentially causing decryption failures. Beyond decryption errors, large coefficient variance can introduce vulnerabilities that attackers might exploit.

One example of such a vulnerability is cyclic shifts. In cyclic shift attacks, an oracle (a helper providing feedback) is used to identify cases where the private key fails to correctly decrypt valid ciphertexts. Failures in decryption often occur when coefficients deviate significantly from expected values [11]. Weak padding methods worsen the problem of cyclic shift vulnerabilities by increasing the likelihood of decryption errors and even exposing the private key [10].

In [3], it is shown that careful selection of parameters such as  $p, q,$  and  $N$  helps control coefficient variance. Their study demonstrated that keeping variance low improves the reliability of NTRUEncrypt and increases the likelihood of successful decryption.

Controlling coefficient variance is crucial for the security and reliability of NTRUEncrypt. Reducing variance lowers the risk of decryption errors and mitigates vulnerabilities. By selecting optimal parameters, NTRUEncrypt can achieve better performance and enhanced security in practical applications.

## 5 Conclusion

This section concludes by emphasizing the vital role that coefficient variance plays in ensuring successful decryption in NTRUEncrypt. By linking statistical analysis to practical cryptographic performance, the discussion highlights how optimizing coefficient variance can significantly enhance decryption reliability. In [5], it is shown that controlling coefficient variance is critical to keep all  $a_i$  coefficients within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$ , a condition necessary for reliable decryption in NTRUEncrypt. Lower coefficient variance provides tighter control over coefficients, reducing the likelihood of outliers and increasing the probability of successful and reliable decryption. By optimizing coefficient variance and its effect on controlling  $a_i$  coefficients, the likelihood of decryption success in NTRUEncrypt can be significantly improved.

Ensuring that all  $a_i$  coefficients remain within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$  is crucial for decryption reliability. Theorem 3.1 establishes a mathematical relationship between coefficient distribution and decryption reliability, reinforcing the need to minimize coefficient variance across all polynomial coefficients. The mathematical framework outlined in Theorem 3.1 serves as a foundation for improving parameter selection and optimizing the performance of the NTRUEncrypt cryptographic scheme.

Continuous research into the statistical properties of NTRUEncrypt is necessary to uncover effective methods for coefficient variance reduction. Thorough testing protocols are crucial for assessing how methods for generating polynomial coefficients with minimal coefficient variance impact the likelihood of all  $a_i$  coefficients falling within the interval  $\left(-\frac{q-1}{2}, \frac{q-1}{2}\right]$  across different parameter sets. Reducing coefficient variance increases the probability of decryption success. Future research could focus on developing adaptive algorithms, such as machine-learning-based approaches, to dynamically generate polynomials with minimal coefficient variance.

The findings about the critical role of coefficient variance in the decryption process and its impact on achieving decryption success in NTRUEncrypt's encryption and decryption mechanisms highlight its importance. By linking statistical analysis to practical cryptographic performance, future research can focus on an optimizing parameter selection and improving NTRUEncrypt's ability to resist attacks, errors, and decryption failures.

**Acknowledgement** The authors would like to express their gratitude to the Ministry of Education, Malaysia, for their invaluable support and research funding. This research project has been made possible through the generous support of the Fundamental Research Grant Scheme (FRGS), under the grant number FRGS/1/2020/STG06/USM/01/1.

**Conflicts of Interest** The authors declare that they have no conflicts of interest to disclose. There are no financial or personal affiliations between the authors and individuals or organizations.

## References

- [1] N. M. G. Al-Saidi & H. R. Yassein (2017). A new alternative to NTRU cryptosystem based on highly dimensional algebra with dense lattice structure. *Malaysian Journal of Mathematical Sciences*, 11(S), 29–43.
- [2] N. Bindel & J. M. Schanck (2020). Decryption failure is more likely after success. In *Post-Quantum Cryptography: 11th International Conference, PQCrypto 2020, Paris, France, 15–17 April 2020, Proceedings*, volume 12100 of *Lecture Notes in Computer Science* pp. 206–225. Springer, Cham. [https://doi.org/10.1007/978-3-030-44223-1\\_12](https://doi.org/10.1007/978-3-030-44223-1_12).
- [3] J. Hoffstein, J. Pipher, J. M. Schanck, J. H. Silverman, W. Whyte & Z. Zhang (2017). Choosing parameters for NTRUEncrypt. In *Topics in Cryptology – CT-RSA 2017*, pp. 3–18. Springer International Publishing, Cham. <https://eprint.iacr.org/2015/708>.
- [4] J. Hoffstein, J. Pipher & J. H. Silverman (1998). NTRU: A ring-based public key cryptosystem. In *Algorithmic Number Theory (ANTS III)*, volume 1423 of *Lecture Notes in Computer Science* pp. 267–288. Springer, Berlin. <https://doi.org/10.1007/BFb0054868>.
- [5] J. Hoffstein, J. Pipher & J. H. Silverman (2014). *Lattices and Cryptography*, pp. 373–470. Undergraduate Texts in Mathematics,. Springer, New York. [https://doi.org/10.1007/978-1-4939-1711-2\\_7](https://doi.org/10.1007/978-1-4939-1711-2_7).
- [6] H. Hosein. Introduction to NTRU public key cryptosystem 2021. Lecture notes, last modified 27 May 2021, <https://hadipourh.github.io/course-cryptanalysis/Slides/Main/NTRU.pdf>.
- [7] N. Howgrave-Graham, P. Q. Nguyen, D. Pointcheval, J. Proos, J. H. Silverman, A. Singer & W. Whyte (2003). The impact of decryption failures on the security of NTRU encryption. In *Advances in Cryptology – CRYPTO 2003: 23rd Annual International Cryptology Conference, Santa Barbara, California, 17-21 August 2003, Proceedings*, volume 2729 of *Lecture Notes in Computer Science* pp. 226–246. Springer, Berlin, Heidelberg. [https://doi.org/10.1007/978-3-540-45146-4\\_14](https://doi.org/10.1007/978-3-540-45146-4_14).
- [8] N. Howgrave-Graham, J. H. Silverman, A. Singer & W. Whyte. NAEP: Provable security in the presence of decryption failures. *Cryptology ePrint Archive*, Paper 2003/172 2003. <https://eprint.iacr.org/2003/172>.
- [9] A. Hülsing, J. Rijneveld, J. Schanck & P. Schwabe (2017). High-speed key encapsulation from NTRU. In *Cryptographic Hardware and Embedded Systems – CHES 2017: 19th International Conference, Taipei, Taiwan, 25–28 September 2017, Proceedings*, volume 10529 of *Lecture Notes in Computer Science* pp. 232–252. Springer, Berlin. [https://doi.org/10.1007/978-3-319-66787-4\\_12](https://doi.org/10.1007/978-3-319-66787-4_12).
- [10] P. Q. Nguyen & D. Pointcheval (2002). Analysis and improvements of NTRU encryption paddings. In *Advances in Cryptology – CRYPTO 2002: 22nd Annual International Cryptology Conference Santa Barbara, California, USA, August 18–22, 2002. Proceedings*, volume 2442 of *Lecture Notes in Computer Science* pp. 210–225. Springer, Berlin. [https://doi.org/10.1007/3-540-45708-9\\_14](https://doi.org/10.1007/3-540-45708-9_14).
- [11] J. Proos. Imperfect decryption and an attack on the NTRU encryption scheme. *Cryptology ePrint Archive*, Paper 2003/002 2003.
- [12] J. Scholten & F. Vercauteren (2003). An introduction to elliptic and hyperelliptic curve cryptography and the NTRU cryptosystem. *State of the Art in Applied Cryptography*, COSIC, 3, 1–24.

- [13] J. Silverman & W. Whyte. Estimating decryption failure probabilities for NTRUEncrypt. Technical report NTRU Cryptosystems No. 18, Version 1 2003. <http://www.ntru.com/cryptolab/articles.htm>.
- [14] H. R. Yassein & N. M. G. Al-Saidi (2019). An innovative bicartisian algebra for designing of highly performed NTRU like cryptosystem. *Malaysian Journal of Mathematical Sciences*, 13(S), 77–91.
- [15] W. Yu, D. He & S. Zhu (2005). Study on NTRU decryption failures. In *Third International Conference on Information Technology and Applications (ICITA'05)*, 4–7 July 2005, Sydney, Australia, volume 2 pp. 454–459. IEEE, Piscataway, New Jersey. <https://doi.org/10.1109/ICITA.2005.266>.