

An Overview of Post-Quantum Cryptography Families

Wiam Abdelrahman Elmalih^{1,2*}, Johnson Ihieh Agbinya³

¹College of Computer Sciences and Information Technology, Sudan University of Science and Technology, Khartoum, Sudan

²College of Science, Northern Border University, Arar, Saudi Arabia

³School of Information Technology and Engineering, Melbourne Institute of Technology, Melbourne, Australia

Email: *wiam.elyass@nbu.edu.sa, jagbinya@mit.edu.au

How to cite this paper: Elmalih, W.A. and Agbinya, J.I. (2026) An Overview of Post-Quantum Cryptography Families. *Advances in Internet of Things*, 16, 77-105. <https://doi.org/10.4236/ait.2026.164006>

Received: August 19, 2026

Accepted: October 16, 2026

Published: October 19, 2026

Copyright © 2026 by author(s) and

Scientific Research Publishing Inc.

This work is licensed under the Creative

Commons Attribution International

License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Quantum computing is a distinct field of computational science that uses principles from quantum mechanics to tackle problems that traditional computing struggles with in speed, size, and complexity. Classical cryptography relies on the complexity of mathematical problems such as large-number factorization, but quantum systems can solve them much faster. This threat necessitates the development and implementation of cryptographic methods resistant to attacks from both classical and quantum systems, known as Post-Quantum Cryptography (PQC). Strong cryptographic algorithms are crucial to counter vulnerabilities in pre-quantum systems. This paper reviews the challenges and solutions for securing communication amid advances in quantum computing. It stresses the urgent need to shift from classical cryptographic systems, which are susceptible to quantum threats, to PQC. The discussion includes theoretical frameworks and practical implementations, covering quantum principles, Quantum Key Distribution (QKD), and standard PQC algorithms. Additionally, it briefly introduces Number Theoretic Transform (NTT) concepts, which enhance polynomial multiplication speed—an essential operation in PQC—by reducing computational complexity from $O(n^2)$ to $O(n \log n)$. The paper demonstrates the NTT/Inverse Number Theoretic Transform (INTT) hardware core in Python, focusing on its application to ML-KEM Polynomial arithmetic. Results indicate that NTT outperforms INTT in execution time and throughput. The paper concludes with an overview of security assumptions, cryptanalysis, and the current status of standardization.

Keywords

Quantum Cryptography, Quantum Key Distribution, Post-Quantum Cryptography, Lattice-Based Cryptography, Number Theoretic Transform

1. Introduction

The modern world relies heavily on telecommunications, with the Internet serving as a crucial medium for communication between parties [1]. Protecting the privacy of transmitted data is essential, but it also creates challenges for securing communication networks. Various techniques and algorithms aim to enhance the security of these systems, ensuring data confidentiality and integrity through cryptographic methods [2].

A cryptosystem encompasses the framework and implementation of cryptographic algorithms designed to safeguard information. It is based on mathematical problems that are difficult for classical computers to solve, though quantum computers can tackle certain complex issues much more efficiently. Unlike classical computers, which use bits that are either 0 or 1, quantum computers employ qubits, which can represent multiple states simultaneously. This capability speeds up specific calculations, making classical cryptographic protocols less effective against security threats [3]. Shor's and Grover's algorithms illustrate the impact of quantum algorithms.

In 1994, Peter Shor developed two quantum algorithms that can solve key problems underpinning current cryptographic systems: the Prime Factorization Problem and the Discrete Logarithm Problem. Unlike classical approaches, which require exponential time, Shor's algorithms operate in polynomial time by finding periods using the Quantum Fourier Transform. Although probabilistic, the chances of successful factorization improve with multiple runs [4].

Published in 1996, Grover's algorithm enables efficient searching of unordered databases in $O(n^{1/2})$ time on quantum systems. It utilizes quantum parallelism and interference through operations that modify the database to efficiently locate target elements. This algorithm can be repeated to increase the likelihood of finding an element and applies to various contexts, such as computing means or finding function inverses. It also helps quantum cryptanalysts break algorithms and search for keys in symmetric systems [4].

PQC ciphers are designed to replace classical cryptographic algorithms that are vulnerable to quantum attacks. Since 2016, the National Institute of Standards and Technology (NIST) has conducted an extensive standardization process for these ciphers. This paper reviews and implements NIST-standardized PQC ciphers, providing an overview of their historical development and standardization to assess their reliability. The paper is structured as follows: Section 2 presents background information on quantum cryptography. Section 3 examines Post-Quantum Cryptography, including the standardization process and established methods. Section 4 compares these methods and describes the strategies of major companies such as IBM, Google, and Microsoft in pursuing quantum capabilities. Section 5 details the implementation of the CRYSTALS-Kyber NTT and INTT algorithms, and Section 6 discusses the results. Section 7 provides the conclusions.

2. Quantum Cryptography

Quantum cryptography promises secure communication protocols that surpass

the limitations of classical cryptography. Quantum cryptography is one of the most critical areas of convergence between fundamental physics and information security. It uses quantum-physics principles to transfer or store data with complete security.

2.1. Core Principles of Quantum Mechanics

Quantum Superposition: Classical computing encodes information in bits, each representing either 0 or 1. In contrast, quantum computing encodes information in quantum bits (qubits). A qubit is a two-level quantum system with two orthogonal basis states, typically denoted as $|0\rangle$ and $|1\rangle$. Unlike classical bits, a qubit can exist in the states $|0\rangle$, $|1\rangle$, or any linear combination of them. This property, known as superposition, is represented as shown [5]:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (1)$$

here, α and β are complex probability amplitudes corresponding to each basis state, and they satisfy the normalization condition $|\alpha|^2 + |\beta|^2 = 1$

Quantum Entanglement is a phenomenon in which the quantum states of two or more objects become correlated, such that the state of one object influences the state of the others, even when separated by large distances. According to quantum theory, particles can exist in multiple states simultaneously and may become inextricably linked, or “entangled”, despite physical separation. Quantum entanglement is mathematically described using the formalism of quantum states in a Hilbert space, which encompasses all possible states of a quantum system. Entangled states are represented using notation that explicitly shows the relationship between the quantum states of the particles involved. The creation of n -qubit entangled states can be achieved using a Hadamard gate and $(n - 1)$ Controlled-NOT (C-NOT) gates. One of the most famous entangled states is the Bell state:

$$\begin{aligned} |00\rangle &\mapsto |\psi_{00}\rangle := \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \\ |01\rangle &\mapsto |\psi_{01}\rangle := \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle) \\ |10\rangle &\mapsto |\psi_{10}\rangle := \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle) \\ |11\rangle &\mapsto |\psi_{11}\rangle := \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle) \end{aligned} \quad (2)$$

Quantum repeaters cannot be constructed because quantum particle states cannot be duplicated. However, particles at either end of a network can be connected via a chain of linked particles, creating a repeater effect that forms the foundation of a quantum network. This connection enables the development of advanced protocols like quantum teleportation.

Quantum Teleportation: This technique transfers quantum information between two locations via a classical link, meaning the information cannot be transmitted faster than the speed of light. Despite its name, quantum teleportation does not involve the physical movement of information; rather, it involves transferring the

state of quantum information via entanglement.

Heisenberg Uncertainty Principle: Introduced in 1927 by German physicist Werner Heisenberg, this principle highlights the limitations in precisely knowing certain pairs of physical quantities, such as position and momentum. It establishes a fundamental relationship, stating that both variables cannot be known with absolute precision simultaneously. Mathematically, it sets a lower bound on the uncertainties of complementary variables.

Photon Polarization Principle: Polarization is a phenomenon associated with the wave nature of electromagnetic radiation, arising from interactions between electric and magnetic fields. Light waves can be classified as unpolarized, with vibrations occurring in multiple planes, or polarized, with vibrations limited to a single plane. Polarization describes the orientation of electric field vectors perpendicular to the direction of wave propagation. A single photon's polarization represents a two-state quantum system, or qubit, with horizontal and vertical polarizations as the logical states. Other polarization states are combinations of these two. The typical polarization-state pairs are shown in **Table 1** below.

Table 1. Representation of light polarization as a two-level system or qubit.

Basis	Representation	Bit 0	Bit 1	State
Rectilinear	$\oplus$	$ 0\rangle$ 0° 	$ 1\rangle$ 90° 	$ 0\rangle = (1, 0)$ $ 1\rangle = (0, 1)$
Diagonal	$\otimes$	$ +\rangle$ 45° 	$ -\rangle$ -45° 	$ +45^\circ\rangle = \frac{1}{\sqrt{2}}(0\rangle + 1\rangle)$ $ -45^\circ\rangle = \frac{1}{\sqrt{2}}(0\rangle - 1\rangle)$
Circular	O	$ R\rangle$ 	$ L\rangle$ 	$ L\rangle = \frac{1}{\sqrt{2}}(0\rangle + i 1\rangle)$ $ R\rangle = \frac{1}{\sqrt{2}}(0\rangle - i 1\rangle)$

The No-cloning theorem is a principle in quantum mechanics stating that identical copies of arbitrary unknown quantum states cannot be made. Introduced by Wootters, Zurek, and Dieks in 1982, it has significant implications for quantum computing. While orthogonal states like $\{|0\rangle, |1\rangle\}$ can be copied as with classical bits, replicating an arbitrary single-qubit state is impossible.

2.2. Quantum Key Distribution (QKD)

A significant challenge in classical cryptography is safe key generation and exchange, particularly in unsecured environments. When two parties communicate, they must share information over public channels, exposing them to potential attacks. A key advancement in quantum cryptography is Quantum Key Distribution (QKD) [6].

QKD uses quantum mechanics to generate a secret key over a quantum channel, ensuring security based on the laws of physics. The randomness of quantum states and measurements facilitates key generation, and the no-cloning theorem guarantees uniqueness. The Uncertainty Principle further protects QKD against interception attempts [7]. QKD protocols can be categorized into [8]:

- **Continuous Variable (CV) QKD:** Information is encoded through position and momentum quadratures of a quantized electromagnetic field in an infinite-dimensional Hilbert space, exemplified by the Gaussian modulation protocol.
- **Discrete Variable (DV) QKD:** This approach uses finite-dimensional discrete variables like phase and polarization, including protocols such as BB84, B92, and SARG04.
- **Distributed Phase Reference (DPR) QKD:** In DPR protocols, a single photon is not required for encoding; in fact, Key information is encoded using the phase difference or arrival times of photons, employing weak coherent pulses. DPR protocols include Differential Phase Shift (DPS) QKD and Coherent One-Way (COW) QKD.

Based on the architecture of protocols, DV-QKD protocols can be regrouped into two categories:

- **Prepare-and-measure protocols:** These measure unknown quantum states, helping detect eavesdropping and assess intercepted data.
- **Entanglement-based protocols:** These involve linked quantum states where the measurement of one affects the other, alerting involved parties if an eavesdropper interferes with a trusted node. **Figure 1** represents the classification of QKD protocols.

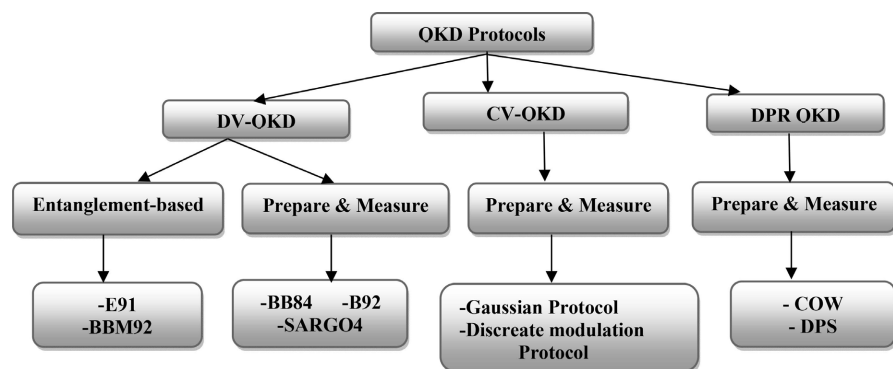


Figure 1. Classification of QKD protocols.

Each QKD protocol has its own advantages and disadvantages, so the best choice depends on the specific use case. A QKD protocol works in two main phases. First is the quantum transmission phase, where the sender and receiver exchange and measure quantum states. Next is the post-processing phase, where the bit strings from the quantum phase are turned into a pair of secure keys. For QKD to work, it requires two channels: a quantum channel and a classical channel, used to ex-

change messages. In the quantum channel, a beam of photons is sent, with each photon encoded by its polarization [7]. **Figure 2** shows a basic QKD process.

QKD provides information-theoretic security; however, it depends on authenticated classical channels during post-processing. Without authentication, these channels are vulnerable to man-in-the-middle attacks. Conventional approaches require communicating parties to pre-share symmetric keys through physical meetings, which is not feasible for multi-user QKD networks. A comprehensive security analysis of QKD must explicitly consider authentication overhead; discrepancies between theoretical proofs and practical implementation security, including side-channel countermeasures; endpoint security assumptions about device trust and key storage; and the potential for hybrid approaches that combine QKD with post-quantum cryptographic authentication.

QKD is already practical and can be used with today's technology, even without quantum computers. However, QKD only handles key distribution and does not replace the encryption systems we rely on to keep our communication secure.

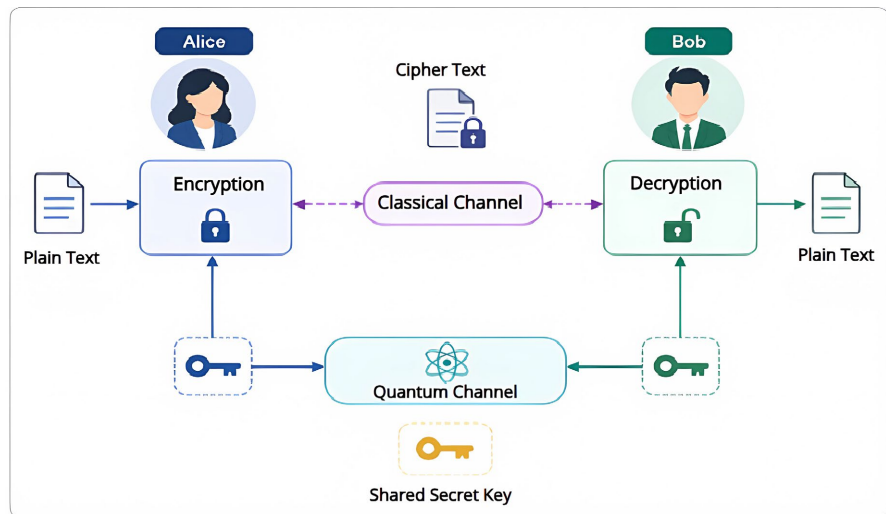


Figure 2. Basic key distribution protocol process [Generated by AI].

3. Post-Quantum Cryptography

If large-scale quantum computers are built, they could break many currently deployed public-key cryptosystems. Such a development would significantly compromise the confidentiality and integrity of digital communications across the Internet and other platforms. Consequently, research has intensified into public-key cryptosystems that remain secure against adversaries equipped with both quantum and classical computers. This area of study is commonly known as Post-Quantum Cryptography (PQC), or quantum-resistant cryptography. The primary objective of PQC is to develop cryptographic systems that are secure against both quantum and classical computational threats and can interoperate with existing communication protocols and networks. Focusing on PQC matters even though large-scale quantum computers are not yet available. Here are a few reasons why [9]:

- Preparation: By developing PQC algorithms now, we can be prepared for future risks posed by quantum computers. Since cryptographic systems protect data for many years, it is important to plan for possible quantum attacks.
- Longevity: PQC algorithms protect against both classical and quantum attacks. They will stay secure even as classical computers become more powerful.
- Adoption: Developing, testing, and adopting new cryptographic systems takes time.

3.1. NIST Post-Quantum Cryptography Standardization

The National Institute of Standards and Technology (NIST) launched a process to select quantum-resistant public-key cryptographic algorithms, aiming to establish robust standards to counter quantum-computer attacks. This initiative focuses on algorithms for digital signatures, public-key encryption, and key establishment, building on NIST's previous standards like DES, AES, and SHA.

In February 2016, NIST called for global contributions to develop post-quantum cryptography standards. By November 2017, 82 draft algorithms had been submitted, and 69 were accepted for the initial standardization phase. After thorough reviews, 26 candidates advanced in January 2019: 17 key establishment mechanisms and 9 digital signatures, chosen for their potential based on analysis and community feedback [10].

The second round included further evaluation, resulting in seven finalists and eight alternates selected in July 2020. NIST planned to standardize a select few finalists and alternates after a fourth round. This third round lasted about 18 months, during which NIST conducted detailed assessments of the theoretical and empirical security evidence, along with performance benchmarking.

In July 2022, NIST identified four final algorithms: Crystals-Kyber for key establishment and Crystals-Dilithium, Falcon, and Sphincs+ for digital signatures, with Crystals-Dilithium recommended as the primary choice. The process also set four alternate KEM candidates—BIKE, Classic McEliece, HQC, and SIKE—for further evaluation [10]. The fourth round began at the end of 2022 and focused on security validation. During this phase, SIKE was deemed insecure, and feedback collected from the cryptographic community proved insightful.

In August 2024, NIST completed the PQC standardization process by issuing three Federal Information Processing Standards (FIPS) specifying particular algorithms. The approved standards are the Module Lattice-Based Key Encapsulation Mechanism (ML-KEM) [11] from CRYSTALS-Kyber for key encapsulation, the Module Lattice-Based Digital Signature Algorithm (ML-DSA) [12] from CRYSTALS-Dilithium for digital signatures, and the Stateless Hash-Based Digital Signature Algorithm (SLH-DSA) [13] from SPHINCS+ for hash-based digital signatures. These algorithms are authorized for use in both federal and commercial environments. A fourth algorithm, the FFNTT-based NTRU Lattice Digital Signature Algorithm (FN-DSA), related to Falcon, remains in draft status with final approval anticipated in late 2026 or early 2027.

By March 2025, NIST announced HQC for standardization, aiming to finalize it in 2027 [14]. The PQC standardization process can be summarized in **Figure 3**.

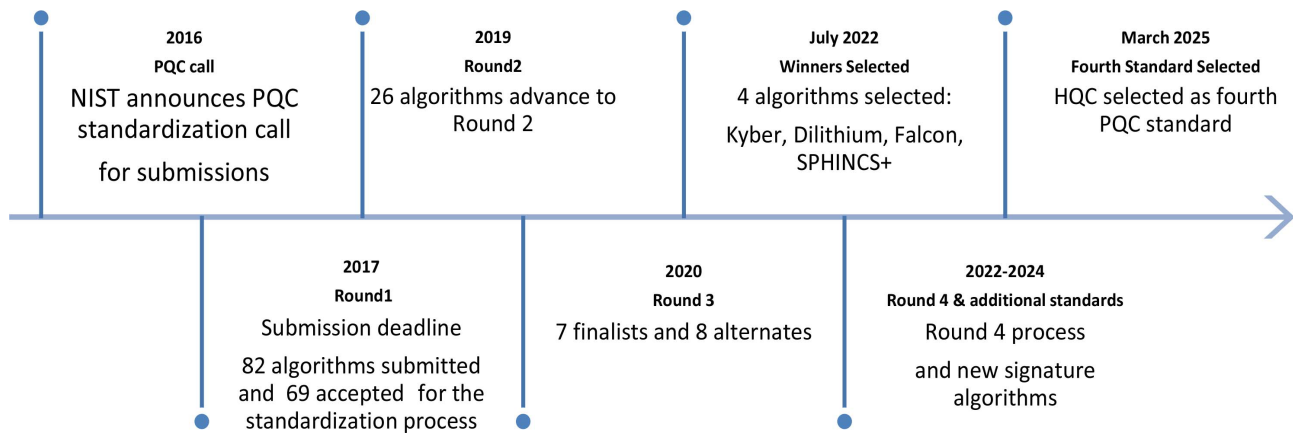


Figure 3. PQC standardization process of NIST.

3.2. Post-Quantum Cryptography Families

Cryptographic systems are categorized by their security foundations. PQC relies on complex mathematical challenges to ensure resilience against classical and quantum threats. Noteworthy families include lattice-based, isogeny-based, multivariate, code-based, and hash-based cryptography [15].

This review examines cryptographic families evaluated by the NIST Post-Quantum Cryptography Standardization Project. The comparison is based on criteria including the underlying mathematical problems and their anticipated resistance to quantum attacks; key, signature, and ciphertext sizes; and computational efficiency for key generation, encryption, signing, decryption, and verification. Additional considerations include memory requirements, practical deployment, and the strength of security proofs. To clarify algorithm status, three categories are defined: 1) officially standardized schemes with finalized NIST FIPS designations which are suitable for practical implementation; 2) selected but not yet finalized candidates which remain in draft status; and 3) obsolete or compromised schemes such as Rainbow and SIKE, which were eliminated following successful cryptanalytic attacks. **Figure 4** shows the cryptosystem families and the algorithms used.

3.2.1. Lattice-Based Cryptography Methods

Lattice-based cryptography relies on complex mathematical structures called lattices to develop encryption methods resilient to quantum attacks. These lattices consist of regularly spaced points in a multi-dimensional space. A vector indicates a specific point, with the origin represented by all zeros. Vectors can be categorized as “far” (larger norm) or “short” (smaller norm) based on their distance from the origin. A basis, a select group of vectors, can represent all points in the lattice by ensuring no two points lie along the same line from the origin [1].

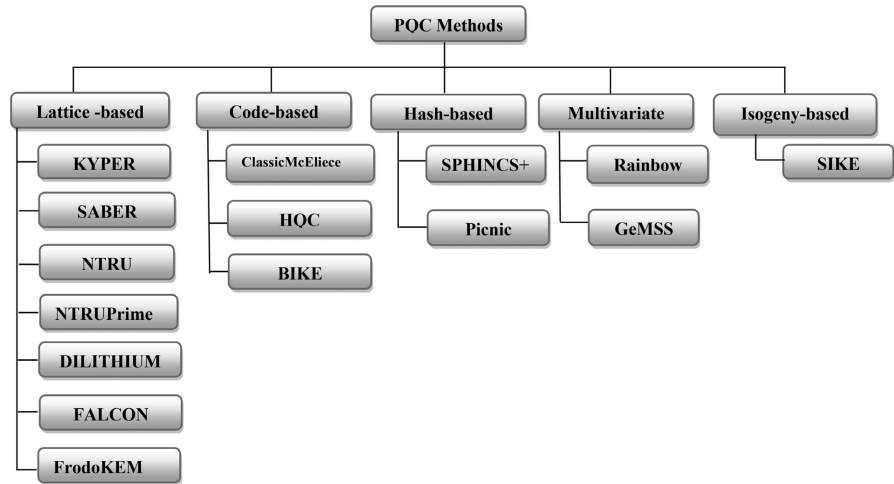


Figure 4. Post-quantum cryptography methods.

Definition 1. A set of vectors $\{v_1, v_2, \dots, v_m\} \in \mathbb{R}^n$ is linearly independent if the equation

$$\alpha_1 v_1 + \alpha_2 v_2 + \dots + \alpha_m v_m = 0, \text{ where } \alpha_i \in \mathbb{R} \quad (3)$$

accepts only the trivial solution $\alpha_1 = \alpha_2 = \dots = \alpha_m = 0$.

Definition 2. Given n linearly independent vectors $\{b_1, b_2, \dots, b_m\} \in \mathbb{R}^n$, the lattice generated by them is defined as the set

$$\mathcal{L}(b_1, b_2, \dots, b_m) = \left\{ \sum_i x_i b_i : x_i \in \mathbb{Z}, 1 \leq i \leq n \right\} \quad (4)$$

The set of vectors $\{b_1, b_2, \dots, b_m\}$ is called the basis of the lattice L .

Let B denote the $n \times m$ matrix with columns $b_1, b_2, \dots, b_m$, then a lattice can be written as

$$\mathcal{L}(B) = \mathcal{L}(b_1, b_2, \dots, b_m) = \{Bx \mid x \in \mathbb{Z}^n\} \quad (5)$$

Matrix B is called a basis matrix. A basis for L is any set of linearly independent vectors that spans L .

Definition 3. The rank of a lattice L is defined as the number of elements in any basis of L , which equals the dimension of the vector subspace spanned by L .

We call m the rank of the lattice, and n the dimension. If $n = m$, the lattice L is called a full-rank lattice.

Lattice L may be generated by various bases, and this concept is generalized using an unimodular matrix.

Definition 4. A matrix $U \in \mathbb{Z}^{n \times n}$ is called unimodular if $\det U = \pm 1$.

Definition 5. Let L be a lattice with dimension n and $B = \{b_1, b_2, \dots, b_m\}$ a basis of the lattice. We define the fundamental parallelepiped of L to be the set

$$\mathcal{P}(B) = \{Bx \mid x \in \mathbb{R}^n, 0 \leq x_i < 1\} \quad (6)$$

Definition 6. Let $L = L(B)$ be a lattice of rank n , and let B be a basis of L . The determinant of L , denoted by $\det(L)$, is the n -dimensional volume of $\mathcal{P}(B)$. The determinant can be written as

$$\det(\mathcal{L}(B)) = \text{vol}(\mathcal{P}) \quad \text{and also} \quad \det(\mathcal{L}) = \sqrt{\det(B^T B)} \quad (7)$$

Definition 7. For any lattice $L = L(B)$, we define the minimum distance of L as the smallest distance between any two lattice points:

$$\lambda_1(\mathcal{L}) = \inf \{\|x - y\| : x, y \in \mathcal{L}, x \neq y\}. \quad (8)$$

Lattice problems are particularly vital because they resist efficient solutions, even on quantum computers. This ensures secure protocols against future quantum attacks [16].

A. Shortest Vector Problem (SVP)

The primary computational challenge in lattice theory is the Shortest Vector Problem (SVP). Because a lattice L is discrete, it has a nonzero element $v \in L$ of minimum length under, say, the Euclidean norm. The fundamental parameter of a lattice is the length of its shortest nonzero vector, denoted as λ_1 .

Definition 8. (Shortest Vector Problem). Given a basis $B = \{b_1, b_2, \dots, b_m\}$ of lattice $L(B)$, find the nonzero vector, i.e., find $v \in L(B)$ for which $\|v\| = \lambda_1(L(B))$.

Figure 5 shows the vector closest to the origin.

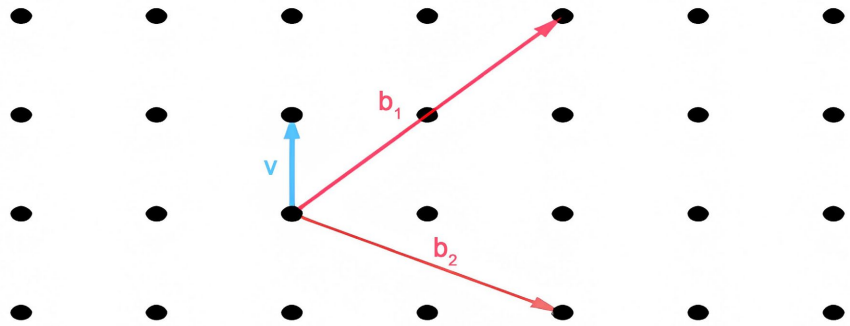


Figure 5. The shortest vector problem [17].

B. Closest Vector Problem (CVP)

CVP is a computational problem in lattices that is closely related to the Shortest Vector Problem (SVP). In the CVP, a lattice L and a target point x are given, and the objective is to identify the lattice point closest to x . This problem is highly challenging because it is NP-hard. Moreover, it is considered more difficult than the SVP, which seeks the shortest non-zero lattice vector. Let L denote a lattice and let t be a fixed point in $\mathbb{R}^n$. The distance is defined as follows:

$$d(t, \mathcal{L}) : \min_{x \in \mathcal{L}} \|x - t\|. \quad (9)$$

Definition 9. (Closest Vector Problem). Given a basis $B = \{b_1, b_2, \dots, b_m\}$ of a lattice $L(B)$ and a target vector t , which may not belong to the lattice, find the lattice point $v \in L(B)$ closest to t .

Therefore, the requested nonzero vector $v \in L(B)$ is the one for which $\|t - v\|$ is minimal, i.e., $\|v\| = d(t, L)$, as represented in Figure 6.

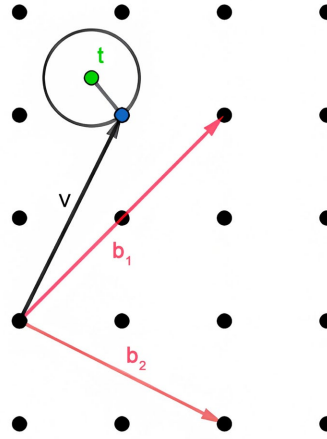


Figure 6. The closest vector problem [17].

C. Shortest Independent Vectors Problem (SIVP)

The primary objective is to identify a new basis that generates the same lattice while minimizing the length of the longest vector. **Figure 7** illustrates this problem.

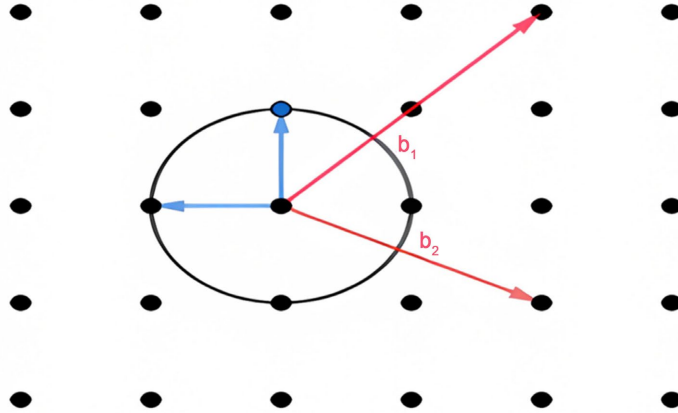


Figure 7. Shortest Independent Vector Problem (SIVP) [17].

Definition 10. *Shortest Independent Vectors Problem (SIVP).* Given a basis $B = \{b_1, b_2, \dots, b_m\}$ of a lattice $L(B)$, find n linearly independent vectors $\{v_1, v_2, \dots, v_n\}$ that belong to the lattice such that $\|v_i\| \leq \lambda_n$ for $1 \leq i \leq n$.

D. Learning with Error (LWE) Problem

LWE has emerged as a fundamental element in efficient lattice-based protocols and is recognized as a significant milestone in the theoretical basis of PQC. This cryptosystem is believed to be as hard as the most difficult lattice problems. It generalizes the Learning Parity with Noise (LPN) problem to larger moduli. Specifically, the LWE problem considers a secret vector $s = (s_1, s_2, \dots, s_n) \in \mathbb{Z}^n$ with integer coefficients and m linear equations, such that:

$$\begin{aligned}
a_{11}s_1 + a_{12}s_2 + \cdots + a_{1n}s_n &\approx b_1 \\
a_{21}s_1 + a_{22}s_2 + \cdots + a_{2n}s_n &\approx b_2 \\
&\vdots \\
a_{m1}s_1 + a_{m2}s_2 + \cdots + a_{mn}s_n &\approx b_m
\end{aligned} \tag{10}$$

By adding noise to linear equations, the LWE problem complicates tasks like Gaussian elimination, rendering the resulting equations ineffective in providing useful information. Error accumulation makes it difficult to arrive at the correct outcome.

Definition 11. (LWE) Let n, q be positive integers, and x be a probability distribution on $\mathbb{Z}$ and s be a secret vector in $\mathbb{Z}_q^n$. We denote by $L_{s,x}$ the probability distribution on $\mathbb{Z}_q^n \times \mathbb{Z}_q$ obtained by choosing $a \in \mathbb{Z}_q^n$ uniformly at random, choosing $e \in \mathbb{Z}$. According to x , and considering it in $\mathbb{Z}_q$ and returning

$$(a, b) = (a, \langle a, s \rangle + e) \in \mathbb{Z}_q^n \times \mathbb{Z}_q \tag{11}$$

Decision-LWE is the problem of deciding whether pairs $(a, b) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ are sampled according to $L_{s,x}$ or the uniform distribution on $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

Search-LWE is the problem of recovering s from

$$(a, b) = (a, \langle a, s \rangle + e) \in \mathbb{Z}_q^n \times \mathbb{Z}_q \text{ sampled according to } L_{s,x}.$$

The solution is asked to lie in a subset $S \subset \mathbb{Z}_q^m$. Where S can be a subset such that $S = \{0, 1\}$ or, in general, the set of all solutions $S = [-B, \dots, B]^m$, and each coordinate takes an absolute value bounded by some number $B \ll q/2$. This is the Short Integer Solution (SIS) problem. The LWE problem is regarded as an analog of the SIS problem. To further improve efficiency while preserving security, the M-LWE problem generalizes LWE by introducing a ring structure. For a polynomial ring $R = \mathbb{Z}[X]/(X^n + 1)$ and modulus q , M-LWE is defined over the module R_q^k as follows [18]:

$$\text{M-LWE}_{n,k,q,x} : \text{Given } (A, b = AS + e) \tag{12}$$

with $A \in R_q^{k \times k}$, $s \in R_q^k$, $e \leftarrow x^k$ distinguishing this distribution from uniform.

This refinement underpins NIST's standardized schemes: the Module-Lattice Key Encapsulation Mechanism (ML-KEM) and the Module-Lattice Digital Signature Algorithm (ML-DSA). These constructions balance rigorous theoretical guarantees with computational efficiency, rendering them suitable for practical deployment and ensuring they inherit the robust hardness assumptions of classical lattice problems [18].

E. Short Integer Solution (SIS) Problem

The Short Integer Solution (SIS) problem requires identifying a nonzero short vector in a specific class of random lattices, known as random q -ary lattices.

Definition 12. (The Short Integer Solution problem) parameterized by positive integers n, m , and q , denoted by $\text{SIS}_{n,m,q,\beta}$. Given m uniformly random vector $a_i \in \mathbb{Z}_q^n$ forming the column of a matrix $A \in \mathbb{Z}_q^{n \times m}$, find a nonzero integer vector $z \in \mathbb{Z}^m$ of norm $\|z\| \leq \beta$, such that:

$$f_A(z) := Az = \sum_i a_i z_i = 0 \in \mathbb{Z}_q^n. \tag{13}$$

In terms of a lattice problem, the SIS problem states that, given a lattice generated by A , find a nonzero integer vector $z \in \mathbb{Z}^n$ such that:

$$A \cdot z = 0 \bmod q \quad \text{where } z \neq 0 \quad \text{and } z \in [-B, B]^m \quad (14)$$

Lattice-based cryptography is a promising approach to PQC for its security, flexibility, and resistance to quantum attacks. A distinguishing characteristic of lattice-based cryptography is its reliance on worst-case security, whereas most other cryptosystems are based on average-case security. Several lattice-based algorithms are leading candidates, each offering distinct advantages in terms of security, computational efficiency, and practical usability.

CRYSTAL-KYBER is part of the **Cryptographic Suite for Algebraic Lattices** (CRYSTAL) family of primitives. In late 2022, the National Institute of Standards and Technology (NIST) standardized the CRYSTALS-Kyber algorithm, citing its strong security, efficient hardware and software implementations across multiple platforms, and compatibility with most existing internet protocols and cryptographic applications [15]. KYBER has been proposed as a public-key algorithm offering security levels comparable to Advanced Encryption Standard (AES) schemes. It functions as a Key Encapsulation Mechanism (KEM) and is based on the computational hardness of the Module-Learning-With-Errors (MLWE) problem. A KEM consists of three algorithms: Key Generation, Encapsulation, and Decapsulation. In a standard key-establishment procedure, the recipient generates a public-private key pair. The sender then uses the recipient's public key to perform the encapsulation operation, which produces both a ciphertext and a shared secret. The recipient applies the corresponding private key to decapsulate the ciphertext and independently derive the same shared secret. This shared secret may be used directly or processed through a key-derivation function to generate a symmetric session key, enabling efficient authenticated encryption and secure communication. Consequently, Kyber is not primarily intended for encrypting application data. Its main function is to provide a quantum-resistant way to establish cryptographic keying material, after which symmetric-key cryptography can efficiently secure the exchanged data.

SABER is a finalist in the Public-Key Encryption (PKE) category for PQC, with security based on the hardness of the MLWR problem. Developed by Banerjee, Peikert, and Rosen at EUROCRYPT 2012, SABER achieves Indistinguishability under Chosen-Plaintext Attack (IND-CPA). The scheme employs a hierarchical structure of algorithms. First, it constructs a CPA-secure public-key encryption scheme based on the MLWR problem. Subsequently, a post-quantum adaptation of the Fujisaki-Okamoto transform constructs a Chosen-Ciphertext Attack (CCA)-secure KEM scheme [19].

CRYSTALS-Dilithium is a digital signature scheme that relies on the complexity of the Module Short Integer Solution (MSIS) and MLWE problems. As part of the CRYSTALS suite, NIST selected it as a finalist in the post-quantum cryptography standardization process in July 2020. The scheme operates within a polynomial ring defined by two integers, n and q , with a matrix of polynomials used in

its operations. Security levels can be adjusted by varying the matrix dimensions, enabling optimization across different levels. Dilithium offers four parameter sets: weak, medium, recommended, and very high, with the recommended level featuring 2.7 KB signatures and 1.5 KB public keys.

NTRU (**N**-th degree **T**runcated **p**olynomial **R**ing **U**nit), introduced in 1998 by Hoffstein, Pipher, and Silverman, represents the first practical public-key encryption scheme utilizing lattices over polynomial rings. It supports both encryption (NTRUencrypt) and digital signatures (NTRUSign). Its security hinges on the difficult shortest vector problem in lattice reduction, for which no known polynomial-time solution exists. In 2009, the IEEE approved NTRU for standardization. Although NIST did not select NTRU as one of the top four quantum-resistant algorithms, it offers advantages over competitors such as KYBER and Saber, including a compact ciphertext and potentially faster encryption and decryption [20].

Falcon (**F**ast-**F**ourier **L**attice-based **C**ompact Signatures over **N**TRU) is a cryptographic signature algorithm that was submitted to the NIST Post-Quantum Cryptography Project on November 30, 2017. It is based on NTRU and employs lattice-based techniques to provide a robust solution for digital signing in the context of quantum computing. The algorithm utilizes a technique based on the work of Gentry, Peikert, and Vaikuntanathan, combined with Fast Fourier sampling, to generate signatures. Falcon's core elements are polynomials with integer coefficients, usually of degree that is a power of two, such as 512 or 1024. It performs calculations in the field of $\mathbb{Q}[x]/(\phi)$, operating under a monic polynomial of degree represented as ϕ , which takes the form $\phi = x^n + 1$. Falcon's structure includes three key processes: key generation, signature generation, and verification [21].

In post-quantum cryptography, schemes such as CRYSTALS-Kyber, NTRU, and SABER address key encapsulation and secure key exchange, while Falcon and CRYSTALS-Dilithium serve as digital signature methods. NIST selected CRYSTALS-Kyber as the standard KEM for its security and efficiency. Dilithium offers effective signature creation with simple implementation, whereas Falcon yields smaller signature sizes but incurs greater complexity. NTRU is one of the earliest lattice-based systems, while SABER offers an efficient alternative based on Module-LWR that avoids the need for the NTT.

3.2.2. Code-Based Cryptography Methods

Code-based cryptography leverages the intractability of decoding general linear codes, a challenge that quantum computers also find insurmountable. This approach is based on the difficulty of decoding error-correcting codes known as Goppa codes, which remain unbroken and are linear codes with fast-decoding algorithms. They are used in a wide range of applications, including hard drives, satellite communication, and fault-tolerant quantum computation.

Definition 13. Let $g(z) = g_0 + g_1z + g_2z^2 + \dots + g_rz^r \in \mathbb{F}_{q^m}[z]$, and let

$L = \{\alpha_1, \alpha_2, \dots, \alpha_n\} \subseteq \mathbb{F}_{q^m}$ such that $g(\alpha_i) \neq 0$, all $\alpha_i \in L$. Then, the code defined by

$$\left\{ c = (c_1, c_2, \dots, c_n) \in \mathbb{F}_q^n : \sum_{i=1}^n \frac{c_i}{z - \alpha_i} \equiv 0 \pmod{g(z)} \right\} \quad (15)$$

is called Goppa code with parameters $g(z)$ and L ; denoted by $\Gamma(L, g(z))$.

The general problems in coding theory that underlie the security of code-based cryptosystems are listed below.

Problem 1. General Decoding. Given $G \in \mathbb{F}_q^{k \times n}$ of rank k , $t \in [0, n]$, $y \in \mathbb{F}_q^n$ where $y = c + e$ with $c = mG$ for some $m \in \mathbb{F}_q^k$ and $|e| = t$, find e .

Problem 2. Syndrome Decoding. Given $H \in \mathbb{F}_q^{(n-k) \times n}$ of rank $n - k$, $t \in [0, n]$, $s \in \mathbb{F}_q^{n-k}$ where $He^T = s^T$ with $|e| = t$, find e .

The problem remains exponentially hard in the general case, even after extensive research in coding theory, information theory, and computational complexity. This persistent difficulty supports strong confidence in the long-term security of code-based cryptography, which is underpinned by robust mathematical foundations and an extensive history of cryptanalysis. Code-based cryptography offers security guarantees established through decades of theoretical and practical analysis. Notably, the original McEliece cryptosystem has withstood attacks for nearly five decades of intensive cryptanalytic scrutiny [18].

Classic McEliece is a code-based KEM that utilizes binary Goppa codes within the Niederreiter variant of the McEliece cryptosystem. These codes, combined with established cryptographic techniques, achieve IND-CCA2 security and resist chosen-ciphertext attacks. The original McEliece cryptosystem was developed to guarantee One-Wayness under Chosen-Plaintext Attacks (OW-CPA), ensuring that if an adversary randomly selects a codeword, it cannot efficiently recover it from the ciphertext and public key. McEliece's security level has remained stable over time. The cryptosystem is based on three core algorithms: a probabilistic key-generation algorithm that outputs a public-private key pair, a probabilistic encryption algorithm, and a deterministic decryption algorithm. In a McEliece deployment, all users employ the same security parameters: n , t , and k . The recommended parameter values are $n = 1024$, $t = 38$, and $k = 644$ [22].

HQC (Hamming Quasi-Cyclic) is a KEM that employs quasi-cyclic codes without embedding a trapdoor in the code. This approach leverages the distinctive properties of quasi-cyclic codes and provides a direct security reduction to decoding random linear codes. Unlike similar candidates, the security proof for HQC relies solely on assumptions related to the decisional Quasi-Cyclic Syndrome Decoding (QCSD) problem [23]. HQC's IND-CPA security relies on the computational hardness of QCSD and the associated parity issue. Applying the Fujisaki-Okamoto (FO) transformation to a CPA-secure Public-Key Encryption (PKE) scheme enables HQC to achieve IND-CCA2 KEM security. Although the quasi-cyclic structure of HQC yields relatively small public keys and ciphertexts, they

remain larger than those of structured lattice-based KEMs. NIST has determined that HQC complements ML-KEM effectively, as it is based on a distinct security foundation while maintaining sufficient performance for general applications. NIST further asserts that the HQC submission satisfies the requirements for a low Decryption Failure Rate (DFR), thereby ensuring IND-CCA2 security [14].

BIKE (Bit-Flipping Key Encapsulation) is a KEM based on binary linear Quasi-Cyclic Moderate-Density Parity-Check (QC-MDPC) codes, a class of error-correcting codes that enable efficient encryption, decryption, and key exchange. Although the BIKE cryptosystem was originally developed for ephemeral key use, it is now claimed to support static-key applications as well.

Definition 14. A quasi-cyclic code is a linear code for which cyclically shifting a codeword a fixed number $n_0 \neq 0$ of positions either to the right or to the left results in another codeword. For $n_0 = 1$, a quasi-cyclic code is a cyclic code. The integer n_0 is called the shifting constraint.

Several standard methods represent a quasi-cyclic parity-check code. For a code with a shifting constraint n_0 and positive integers c and t , where $c < t$, let $H_i \in \mathbb{F}_2^{c \times n_0}$ for $i \in [t]$. The nullspace of is then a quasi-cyclic parity-check code [24].

$$H = \begin{bmatrix} H_1 & H_2 & \cdots & H_t \\ H_t & H_1 & \cdots & H_{t-1} \\ \vdots & \vdots & \ddots & \vdots \\ H_2 & H_3 & \cdots & H_1 \end{bmatrix} \in \mathbb{F}_q^{t \times t n_0} \quad (16)$$

The secret key comprises a parity-check matrix $H_{r \times 2r}$, for a Quasi-Cyclic Moderate-Density Parity-Check (QC-MDPC) code, where r denotes a prime number. Each row of matrix H possesses a specified weight. The secret key is denoted by (h_0, h_1) , and the public key is derived from it. During encryption, the message is interpreted as an error vector e of weight t , and the ciphertext is computed accordingly. Decryption requires multiplying the ciphertext by h_0 to obtain the syndrome, which is subsequently decoded using a Black-Grey-Flip (BGF) decoder to recover either the error vector or the original message [9]. The security of the BIKE algorithm relies on the computational intractability of decoding randomly generated error-correcting codes without knowledge of the private key, which poses significant challenges for attackers, including those equipped with quantum computers, attempting to recover the shared key.

3.2.3. Hash-Based Cryptography Methods

Hash functions are mathematical tools that convert input data of any size into a fixed-length output, called a hash value. A key characteristic of these functions is that they are one-way, making it nearly impossible to retrieve the original data from the hash. This feature is crucial for maintaining data integrity and identifying any alterations. Additionally, hash functions exhibit collision resistance, meaning it's difficult to find two distinct inputs that yield the same hash, which is vital for cryp-

tographic security [22].

In the late 1970s, hash-based signature schemes emerged, initially as one-time signature systems developed by Lamport and later extended by Ralph Merkle. He created the MerKel signature schemes and proposed the MerKel tree structure. The security of these systems is straightforward to assess and relies on the hash function used. However, users must keep track of how many signatures have been created with a specific key. Over the past 40 years, advancements have enhanced the efficiency and theoretical foundations of hash-based signatures, culminating in XMSS, which is nearing standardization as the first signature scheme designed to withstand quantum threats. These schemes combine one-time signatures with a Merkle tree structure to sign messages effectively [25]. Hash-based signature schemes integrate a One-Time Signature scheme (OTS) with a Merkle tree structure. Because each OTS key securely signs only one message, multiple keys are organized into a Merkle tree to enable signing multiple messages.

SPHINCS, developed by a team of researchers, is a hash-based signature scheme that does not rely on stateful components, making it the first to propose parameters against quantum attacks. It builds on components of XMSS but uses larger keys and signatures, thereby removing the need for state [26]. The scheme incorporates elements from HORST and WOTS++ (Winternitz OTS), which traditionally require pre-generated keys to calculate the Merkle root, limiting key management. In contrast, SPHINCS can handle many more keys through two methods: a hypertree for authentication and a random key-path addressing scheme.

SPHINCS+ improves this framework with better speed, security, and smaller signatures. While it retains similar principles, it differs by using FORS for signing, implementing a publicly verified leaf-index selection, and introducing tweakable hash functions, thereby streamlining the security analysis of hash-based signatures [22].

3.2.4. Isogeny-Based Cryptography Methods

Isogeny-based cryptography combines elliptic curves and isogenies, showing promise as a secure cryptographic approach. An elliptic curve consists of points that satisfy a specific equation over a finite field and form a group under operations such as point addition. Elliptic curves, which are projective curves of genus 1 with a specified base point, are fundamental in cryptography, especially in Elliptic Curve Cryptography (ECC), as shown in **Figure 8** [27].

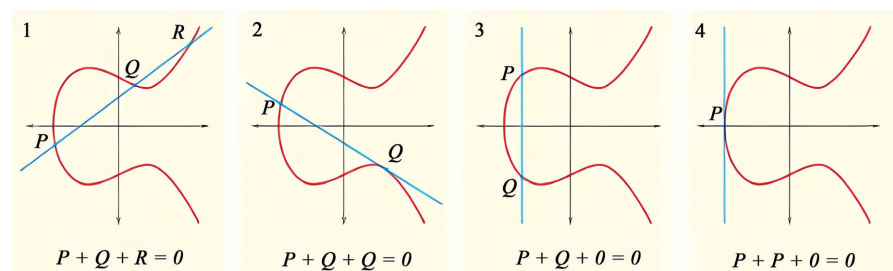


Figure 8. Elliptic curve cryptography lines [27].

Isogenies are algebraic maps between elliptic curves over finite fields that serve as group homomorphisms, as illustrated in **Figure 9**. These maps define relationships between curves, and their degrees quantify the size of the isogeny, a property particularly important for separable isogenies [27].

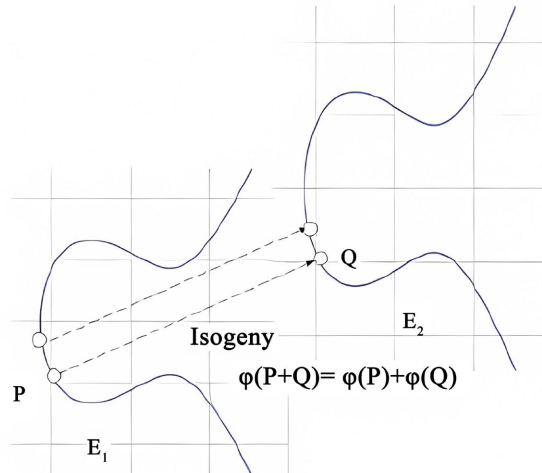


Figure 9. Isogeny [26].

Supersingular elliptic curve isogeny cryptography leverages these properties to implement key-exchange protocols like Diffie-Hellman, offering a quantum-resistant alternative with smaller key sizes than many other post-quantum cryptographic systems. The primary schemes in this domain are Supersingular Isogeny Diffie-Hellman (SIDH) and its commutative variant (CSIDH), while Supersingular Isogeny Key Encapsulation (SIKE) has been submitted to the NIST Post-Quantum Cryptography Competition [10].

SIKE is a key encapsulation method chosen in the third round of NIST's post-quantum standardization process. It builds on the SIDH key exchange protocol from 2011, utilizing arithmetic on elliptic curves and computing isogenies between them. The security of both SIDH and SIKE depends on the difficulty of identifying a specific isogeny between two curves, which is distinct from the discrete logarithm problem on a single curve [28].

One key advantage of these methods is their small public keys, making them among the most compact quantum-resistant algorithms. However, their relatively slow performance has limited their effectiveness as practical post-quantum candidates.

3.2.5. Multivariate Cryptography Methods

Multivariate cryptography uses asymmetric encryption based on multivariate polynomials over finite fields. It's commonly used to create digital signatures and is known for producing the shortest signatures among Post-Quantum Cryptography (PQC) methods. The security of this approach hinges on the complexity of solving nonlinear systems of equations, particularly multivariate quadratic polynomial systems [25]. There are primarily two categories of multivariate digital signature

schemes. The first includes BigField schemes like HFEv and the NIST candidate GeMSS. The second category comprises SingleField schemes, including UOV and Rainbow [29]. While some multivariate encryption and authentication schemes have proven weak, a few signature schemes have withstood cryptanalysis. Notably, UOV (Unbalanced Oil and Vinegar) and SFLASH v2 are two resilient schemes that have demonstrated effective implementation and were recognized in 2002 by the European project NESSIE as viable digital signature algorithms.

Definition 15. (*Multivariate quadratic maps*) The central object in Multivariate Quadratic cryptography is the multivariate quadratic map. A multivariate quadratic map $\mathcal{P}$ with m components and n variables is a sequence $\mathcal{P}_1(x), \dots, \mathcal{P}_m(x)$ of m multivariate quadratic polynomials in n variables $x = (x_1, \dots, x_n)$, with coefficients in a finite field $\mathbb{F}_q$. To evaluate the map $\mathcal{P}$ at a value $a \in \mathbb{F}_q^n$, simply evaluate each of its component polynomials in a to get a vector $b = (b_1 = \mathcal{P}_1(a), \dots, b_m = \mathcal{P}_m(a))$ of m output elements; denote this by $\mathcal{P}(a) = b$ [30].

Rainbow, created in 2004 by Jintai Ding and Dieter Schmidt, is a digital signature scheme derived from Jacques Patarin's Oil-Vinegar (UOV) model. It features a trapdoor function represented by a multivariate quadratic map with several layers of nested subspaces. The version presented in the NIST PQC project features a 2-layer variant with specific dimensions and equations connecting inputs and outputs. Although Rainbow offers rapid signature generation and verification, its key generation is slower [29].

In 2022, a key-recovery attack exploited SIKE's torsion-point information, enabling full key recovery within hours on a single core and prompting NIST to withdraw the scheme from consideration. Similarly, Ward Beullens applied a rectangular MinRank attack to Rainbow, rapidly recovering the private key and resulting in its removal from the standardization process. These incidents highlight significant risks in post-quantum cryptanalysis, demonstrating that structured algebraic problems may harbor vulnerabilities and rendering such methods unsuitable for practical deployment, despite prior assumptions of quantum resistance.

4. Discussion

Quantum computing has advanced significantly in recent years, particularly as leading technology companies develop machines with capacities exceeding 1000 qubits. IBM, for example, plans to build a supercomputer with more than 4000 qubits by 2025, emphasizing improvements in circuit quality to support 5000 gates per circuit for large-scale workloads. Google aims to achieve an error-corrected quantum computer by 2029, building upon its 53-qubit Sycamore processor. Microsoft is pursuing utility-scale quantum computing through a three-tiered approach—Foundational, Resilient, and Scale—with a primary focus on topological qubits. In February 2025, Microsoft introduced the Majorana 1 processor, designed to scale to 1 million qubits by leveraging hardware-protected qubits [31].

Quantum computers are advancing rapidly and will become widely adopted in the future. With quantum technology progressing swiftly, post-quantum cryp-

tographic algorithms are also evolving. The NIST has recently announced four standardized post-quantum encryption algorithms, crucial for securing digital communications against future quantum threats. The analysis indicates that lattice-based schemes are ideal for this transition. ML-KEM is recommended for key encapsulation because of its strong security and efficiency, while ML-DSA offers balanced performance for general use. Falcon excels in bandwidth-limited settings but necessitates careful implementation, and SLH-DSA remains valuable as a high-assurance option despite its slower speed and larger signatures. While post-quantum cryptography continues to improve in efficiency, its inherently large parameter sizes remain a significant challenge. Choosing the right algorithm will depend on security levels, parameter size, execution speed, and specific application needs. **Table 2** presents the first standardized quantum-resistant cryptographic algorithms, highlighting these considerations and correlating security levels with the difficulty of breaking traditional encryption methods such as AES [32].

Table 2. NIST PQC KEM & digital signature algorithms, security levels, and parameter sizes [18] [32].

NIST PQC KEM algorithms									
Algorithm	Scheme	Parameters	Security	KeyGen (ms)	Enc (ms)	Dec (ms)	Private Key (B)	Public Key (B)	Ciphertext (B)
Kyber	Lattice-Based	Kyber512	1	0.253	0.070	0.084	1632	800	768
		Kyber768	3	0.354	0.095	0.118	2400	1184	1088
		Kyber1024	5	0.483	0.138	0.159	3168	1568	1568
HQC	Code-based	HQC-128	1	2.84	4.12	8.95	2305	2249	4433
		HQC-192	3	5.67	8.34	18.2	4586	4522	8978
		HQC-256	5	11.2	16.8	35.4	7317	7245	14,421
NIST PQC digital signature algorithms									
Algorithm	Scheme	Parameters	Security	KeyGen (ms)	Signing (ms)	verifying (ms)	Private Key (B)	Public Key (B)	Signature (B)
Dilithium	Lattice-Based	Dilithium 2	1	0.253	0.840	0.267	2528	1312	2420
		Dilithium 3	3	0.392	1.205	0.398	4000	1952	3293
		Dilithium 5	5	0.652	1.998	0.584	4864	2592	4595
Falcon	Lattice-Based	Falcon 512	1	8.64	0.168	0.036	1281	897	666
		Falcon 1024	5	27.45	0.344	0.073	2305	1793	1280
SPHINCS+	Hash-based	SPHINCS+-128s	1	0.032	120.5	1.45	64	32	7856
		SPHINCS+-192s	3	0.048	285.2	2.98	96	48	16,224
		SPHINCS+-256s	5	0.064	652.8	5.12	128	64	29,792

All three algorithm families align with NIST security levels (1, 3, 5), showing proportional increases in key and ciphertext/signature sizes as security strengthens. Kyber stands out for its computational efficiency, operating in sub-millisecond times across all levels, highlighting the strengths of lattice-based designs for modern hardware. In contrast, HQC features significantly larger public keys, complicating its application in bandwidth-limited environments. Dilithium delivers competitive signing performance, maintaining verification times under 1.2 milliseconds at high security levels, indicating that post-quantum signatures can

match or exceed traditional methods. However, SPHINCS+ signature generation takes considerably longer, ranging from 120 ms to 650 ms, due to its complex computations.

Lattice-based algorithms offer an effective balance of efficiency, minimal communication overhead, and strong security foundations, making them suitable for deployment across a wide range of computing environments [18]. With Kyber based on the MLWE problem, it resists simple key-guessing attacks and remains secure against even powerful quantum threats. Yet, transitioning to quantum-safe standards poses significant challenges [33]:

- A lack of visibility in cryptographic use.
- Larger key and signature sizes affect performance.
- The need for hybrid systems.
- Limitations posed by Legacy devices.
- The requirement for coordinated updates across various technology layers.

5. Simulation

Lattice-based methods are recognized for their strong security and computational efficiency, making them ideal for a range of cryptographic applications. Among these methods, CRYSTALS-Kyber stands out as the main KEM chosen during the PQC standardization process. It provides strong security, straightforward computation, and is suitable for resource-constrained environments. Kyber uses efficient polynomial operations enabled by the NTT, which is essential in lattice cryptography. Its predictable structure and small ciphertext sizes enhance its performance with hardware accelerators, making it a top option for fast cryptographic solutions [34].

NTT plays a key role in post-quantum cryptography and homomorphic encryption by streamlining polynomial multiplication. Instead of multiplying polynomials directly, NTT simplifies the process to pointwise multiplication. However, this method requires modular arithmetic in $\mathbb{Z}_q$, which poses challenges for many platforms. Traditional modular multiplication techniques, such as trial division, can be inefficient and difficult to implement in hardware.

In NTT, polynomials are converted from their standard coefficient form to the NTT domain via a forward NTT process that uses butterfly operations to combine coefficients with precomputed constants. Once transformed, polynomials can be reverted to their original form through inverse NTT, which involves a different set of butterfly operations and a final scaling step to adjust the output. Montgomery multiplication performs efficient modular arithmetic during scaling. The method leverages bit manipulation techniques to optimize performance, particularly by using parameters that streamline hardware implementation. The simulation's operation is as follows:

- 1) Load 256 input coefficients into coefficient memory.
- 2) Start Forward NTT.
- 3) Perform butterfly operations stage by stage.
- 4) Store-and-forward NTT result.

- 5) Run inverse NTT on the result.
- 6) Apply final inverse scaling.
- 7) Read output coefficients.
- 8) Compare output with original input to ensure correctness.

The forward and inverse NTT processes efficiently transform between polynomial representations, with careful attention to storage order and scaling adjustments.

Forward NTT

The forward NTT transforms polynomials into the NTT domain using an in-place, iterative algorithm. The reference implementation performs the NTT as follows:

- 1) Require polynomial $a = (a_0, a_1, \dots, a_{n-1})$ as $a(x) \in \mathbb{Z}_q[X]/(X^n + 1)$, $\psi \in \mathbb{Z}_q$ is the N -th primitive root of unity
- 2) Precompute a vector of the first n powers of such a root stored in bit-reverse order.
- 3) Starting with length = 128, work down to length = 2.
- 4) For each length, apply butterfly operations on pairs of coefficients.
- 5) The result is in bit-reversed order.

Inverse NTT

The INTT transforms from the NTT domain back to coefficient representation as follows:

- 1) Processes in the reverse order of the forward NTT.
- 2) Uses the inverse roots of unity (stored in the same zetas array)
- 3) Applies a final scaling factor to each coefficient (Montgomery constant).
- 4) Returns the result in standard order.

Figure 10 illustrates the NTT and INTT algorithm flowchart.

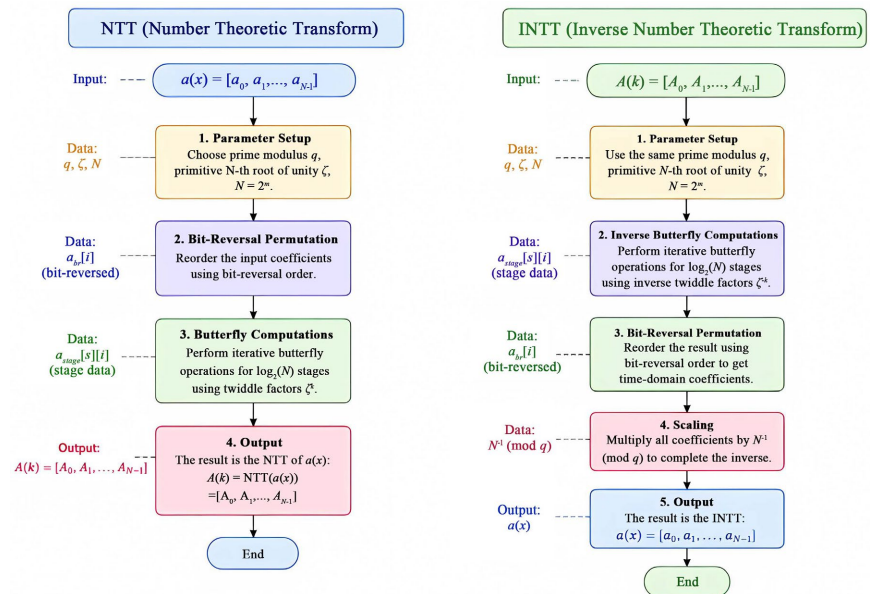


Figure 10. NTT and INTT flowchart [Generated by AI].

Algorithm 1 shows the NTT operation using the Cooley-Tukey butterfly. **Algorithm 2** shows the INTT operation using the Gentleman-Sande butterfly.

Algorithm 1. The Cooley-Tukey NTT.

Input: An n -element vector $a = a_0, a_1, \dots, a_{n-1}$ where $a_i \in [q-1]$, n (power of 2)
 modulus q ($q \equiv 1 \pmod{2n}$)

Input: Zeta (precomputed table of $2n$ -th roots of unity ψ , in bit-reversed order)

Output: $\hat{f} \leftarrow \text{NTT}(a)$

$t \leftarrow n/2; m \leftarrow 1$

while ($m < n$) **do**

$k \leftarrow 0$

for ($i \leftarrow 0; i < m; i \leftarrow i+1$)

$S \leftarrow \text{zeta}[m+i]$

for ($j \leftarrow k; j < k+t; j \leftarrow j+1$)

$U \leftarrow a[j]$

$V \leftarrow a[j+t] \cdot S \pmod{q}$

$a[j] \leftarrow U + V \pmod{q}$

$a[j+t] \leftarrow U - V \pmod{q}$

$k \leftarrow k + 2t$

$t \leftarrow t/2; m \leftarrow 2m$

return $\hat{f}$

Algorithm 2. The gentleman-sande INTT.

Input: An n -element vector $\hat{f} = f_0, f_1, \dots, f_{n-1}$ where $f_i \in [q-1]$, n (power of 2)
 modulus q ($q \equiv 1 \pmod{2n}$)

Input: zeta_inv (precomputed table of $2n$ -th roots of unity ψ^{-1} , in bit-reversed order), $n^{-1} \pmod{q}$

Output: $f \leftarrow \text{INTT}(\hat{f})$

$t \leftarrow 1; m \leftarrow n/2$

while ($m > 0$) **do**

$k \leftarrow 0$

for ($i \leftarrow 0; i < m; i \leftarrow i+1$)

$S \leftarrow \text{zeta_inv}[m+i]$

for ($j \leftarrow k; j < k+t; j \leftarrow j+1$)

$U \leftarrow \hat{f}[j]$

$V \leftarrow \hat{f}[j+t]$

$\hat{f}[j] \leftarrow U + V \pmod{q}$

$W \leftarrow U - V \pmod{q}$

```


$$\hat{f}[j+t] \leftarrow W \cdot S \pmod{q}$$


$$k \leftarrow k + 2t$$


$$t \leftarrow 2t; m \leftarrow m/2$$

for ( $i \leftarrow 0; i < m; i \leftarrow i+1$ )
    
$$f[i] \leftarrow \hat{f}[i] \cdot n^{-1} \pmod{q}$$

return  $f$ 

```

6. Results

Figure 11 shows the CPU performance benchmark for NTT and INTT, implemented on a computer with a Core (TM) i7-1360P processor at 2.20 GHz and 16GB of RAM.

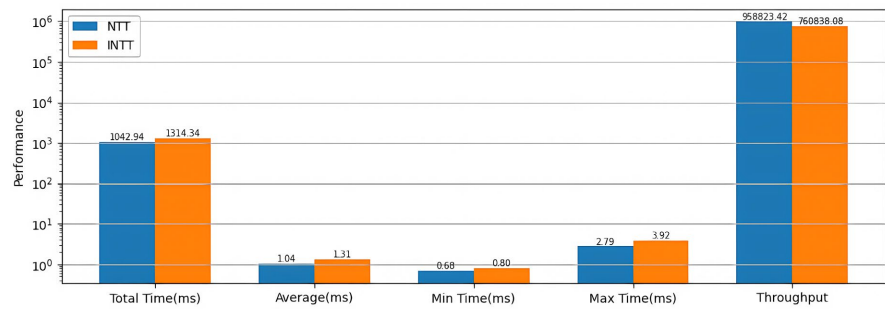


Figure 11. CPU performance benchmark.

The total time taken for the NTT and INTT indicates that INTT takes longer to execute on the same CPU. INTT also has lower throughput. **Table 3** compares CPU performance benchmarks (NTT and INV_NTT) and shows significant differences.

Table 3. Comparison of CPU performance benchmarks.

Metric	NTT	INV_NTT	Difference	Ratio %
Total Time (ms)	1042.94	1314.34	271.4	≈26.0%
Average Time (ms)	1.04	1.31	0.27	≈25.96%
Min Time (ms)	0.68	0.80	0.12	≈17.65%
Max Time (ms)	2.79	3.92	1.13	≈40.5%
Throughput	968823.42	760838.08	207985.34	≈27.34%

NTT clearly outperforms INV_NTT on all time scales. Productivity in NTT is about 27% higher, and INTT suffers from the worst execution time, at 40.5% higher, indicating greater performance fluctuation.

Figure 12 shows the results of the GPU performance benchmark for NTT and INTT, implemented on the Server Colab T4 GPU.

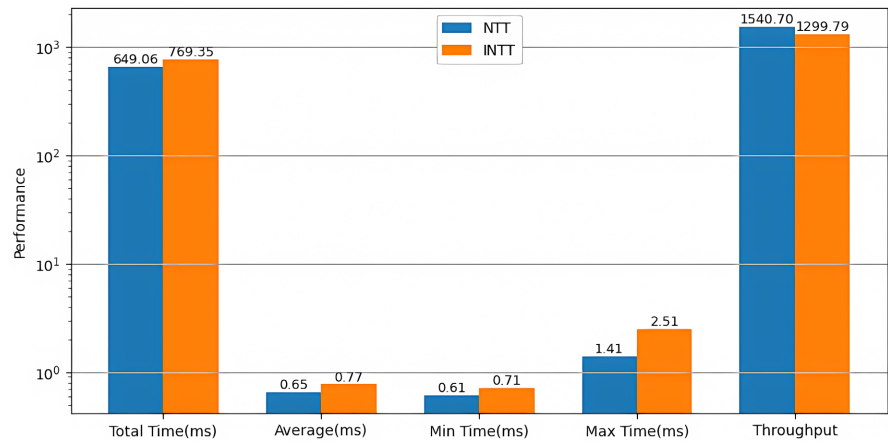


Figure 12. GPU performance benchmark.

Table 4 compares GPU performance benchmarks (NTT and INV_NTT) and shows significant differences.

Table 4. Comparison of GPU performance benchmarks.

Metric	NTT	INV_NTT	Difference	Ratio %
Total Time (ms)	649.06	769.35	120.29	≈18.53%
Average Time (ms)	0.65	0.77	0.12	≈18.46%
Min Time (ms)	0.61	0.71	0.10	≈16.39%
Max Time (ms)	1.41	2.51	1.10	≈78.0%
Throughput	1540.70	1299.79	240.91	≈18.53%

The results demonstrate that NTT outperforms INTT across all measured metrics:

- 18.5% lower total execution time.
- 18.5% lower average execution time.
- 18.5% higher throughput.
- Much lower worst-case latency (about 78% better).
- More stable execution behavior (0.8 ms).

NTT clearly outperforms INV_NTT in terms of execution speed, stability, and productivity. If the application requires high performance and reliability, NTT is the best choice because it delivers faster performance and less volatility. The expected reason for NTT to outperform INTT is that the INTT algorithm includes additional operations compared to NTT, such as:

- Employing transactions ζ^{-1} instead of ζ .
- The final strike in $n^{-1} \bmod q$.
- More Montgomery Reduction operations in some applications.

Therefore, the INTT time is naturally greater, although the theoretical complexity of the two algorithms is the same: $O(n \log n)$.

7. Conclusions

PQC is becoming the main line of defense against the threats posed by large-scale quantum computing to classical cryptographic systems. Popular public-key methods such as RSA and Elliptic-Curve Cryptography (ECC) are expected to be compromised by quantum attackers, highlighting the need for a proactive shift in cryptography. Although practical quantum computers are still in development, various stakeholders, including governments and standard-setting bodies, are already creating strategies to protect sensitive information and vital infrastructure.

This paper reviews the latest innovations in quantum-resistant cryptographic algorithms, such as lattice-based, code-based, Hash-based, and multivariate polynomial methods. It also examines current standardization efforts led by organizations such as NIST, which are vital to establishing reliable quantum-safe cryptographic standards.

The conclusion emphasizes that advancing post-quantum cryptography is essential for securing communication networks against the growth of quantum computing technology. Based on recent findings and NIST's PQC standardization process, lattice-based cryptography, specifically CRYSTALS-Kyber (for encryption) and CRYSTALS-Dilithium (for digital signatures), stands out as the most suitable approach for its scalability, robustness, and hardware adaptability.

However, several challenges remain. Although NIST has standardized four algorithms, comparative analysis of other methods, particularly multivariate, code-based, and isogeny-based approaches, remains limited. Additionally, many algorithms lack hardware implementations, which could slow down their adoption. Also, while post-quantum algorithms offer enhanced security, they may be more resource-intensive, creating issues for small IoT devices. Implementing PQC in IoT devices presents multiple challenges. Many IoT devices possess limited processing power and memory, which complicates the use of PQC algorithms originally designed for more capable systems. These algorithms often require increased energy, thereby reducing battery life in devices with constrained power supplies. Bandwidth limitations are also significant, as low-power networks typically offer reduced transmission speeds, and the larger key sizes of some PQC methods can extend communication times and increase the risk of transmission failures. Furthermore, many IoT devices lack robust firmware update mechanisms, making the transition to new cryptographic methods difficult. Migrating from classical to post-quantum techniques adds complexity, as it requires running both algorithms simultaneously and increases demands on processing, memory, and energy resources. Together, these factors pose substantial obstacles to implementing PQC on resource-constrained devices.

Finally, transitioning to hybrid post-quantum/classical methods requires coordinated updates across systems.

In the future, researchers should focus on designing lightweight post-quantum cryptographic solutions for constrained IoT devices, as well as developing hybrid schemes that effectively integrate symmetric, asymmetric, and post-quantum cryp-

tographic approaches while ensuring energy efficiency.

Acknowledgements

I sincerely thank my supervisor, Johnson Ihieh Agbinya, for his invaluable guidance, support, and mentorship throughout this research. His insights and expertise significantly influenced the direction and outcomes of this study. During the preparation of this study, the authors used [ChatGPT, GPT-5.5] to generate two images. The authors have reviewed and edited the output and take full responsibility for the content of this publication.

Author Contributions

The authors contributed to this manuscript as detailed below: Wiam Abdelrahman Elmalih, the first and corresponding author, drafted and organized all sections of the article, from the abstract to the conclusion. She also handled the practical aspects of the study, including analysis, interpretation, and visualization of the results. Johnson Ihieh Agbinya, the second author, contributed through supervision, guidance, and by providing critical revisions and final edits to the manuscript. All authors reviewed and approved the published version of the manuscript.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Bavdekar, R., Chopde, E.J., Bhatia, A., Tiwari, K., *et al.* (2022) Post Quantum Cryptography: Techniques, Challenges, Standardization, and Directions for Future Research. arXiv: 2202.02826. <https://doi.org/10.48550/arXiv.2202.02826>
- [2] Panhwar, M.A., Khuhro, S.A., Mazhar, T., Deng, Z.L., *et al.* (2021) Quantum Cryptography: A Way of Improving Security of Information. *International Journal of Mathematics & Computer Science*, **16**, 9-12.
- [3] Rubia, J., Babitha, R., Nithila, E., Shibi, S. and Rosi, A. (2024) A Survey about Post Quantum Cryptography Methods. *EAI Endorsed Transactions on Internet of Things*, **10**, 1-9. <https://doi.org/10.4108/eetiot.5099>
- [4] Bavdekar, R., Jayant Chopde, E., Agrawal, A., Bhatia, A. and Tiwari, K. (2023) Post Quantum Cryptography: A Review of Techniques, Challenges and Standardizations. 2023 *International Conference on Information Networking (ICOIN)*, Bangkok, 11-14 January 2023, 146-151. <https://doi.org/10.1109/icoi56518.2023.10048976>
- [5] Nath, A., Maity, S., Banerjee, S. and Roy, R. (2024) Quantum Key Distribution (QKD) for Symmetric Key Transfer. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, **10**, 270-280. <https://doi.org/10.32628/cseit24103105>
- [6] Sabani, M.E., Savvas, I.K., Poulakis, D., Garani, G. and Makris, G.C. (2023) Evaluation and Comparison of Lattice-Based Cryptosystems for a Secure Quantum Computing Era. *Electronics*, **12**, Article 2643. <https://doi.org/10.3390/electronics12122643>
- [7] Sabani, M., Savvas, I., Poulakis, D. and Makris, G. (2022) Quantum Key Distribution:

- Basic Protocols and Threats. *Proceedings of the 26th Pan-Hellenic Conference on Informatics*, Athens, 25-27 November 2022, 383-388.
<https://doi.org/10.1145/3575879.3576022>
- [8] Malpani, P., Kumar, S. and Pathak, A. (2024) Implementation of Coherent One Way Protocol for Quantum Key Distribution up to an Effective Distance of 145 km. *Optical and Quantum Electronics*, **56**, Article No. 1369.
<https://doi.org/10.1007/s11082-024-06934-2>
 - [9] Dam, D.T., Tran, T.H., Hoang, V.P., Pham, C.K. and Hoang, T.T. (2023) A Survey of Post-Quantum Cryptography: Start of a New Race. *Cryptography*, **7**, Article 40.
<https://doi.org/10.3390/cryptography7030040>.
 - [10] Alagic, G., Cooper, D., Dang, Q., Dang, T., *et al.* (2022) Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process. NIST IR 8413-upd1. National Institute of Standards and Technology.
 - [11] National Institute of Standards and Technology (2024) Module-Lattice-Based Key Encapsulation Mechanism Standard. NIST FIPS 203. U.S. Department of Commerce, Federal Information Processing Standards Publication (FIPS).
<https://doi.org/10.6028/NIST.FIPS.203>
 - [12] National Institute of Standards and Technology (2024) Module-Lattice-Based Digital Signature Standard. NIST FIPS 204. U.S. Department of Commerce, Federal Information Processing Standards Publication (FIPS). <https://doi.org/10.6028/NIST.FIPS.204>
 - [13] National Institute of Standards and Technology (2024) Stateless Hash-Based Digital Signature Standard. NIST FIPS 205. U.S. Department of Commerce, Federal Information Processing Standards Publication (FIPS). <https://doi.org/10.6028/NIST.FIPS.205>
 - [14] Alagic, G., Bfros, M., Ciadoux, P., *et al.* (2025) Status Report on the Fourth Round of the NIST Post-Quantum Cryptography Standardization Process. NIST IR 8545.
<https://doi.org/10.6028/NIST.IR.8545>
 - [15] Li, S., Chen, Y., Chen, L., Liao, J., Kuang, C., Li, K., *et al.* (2023) Post-Quantum Security: Opportunities and Challenges. *Sensors*, **23**, Article 8744.
<https://doi.org/10.3390/s23218744>
 - [16] Pouly, A., Shen, Y. (2026) Solving the Shortest Vector Problem in $2^{0.63269n+O(n)}$ Time on Random Lattices. In: Daemen, J. and Thomé, E., *Advances in Cryptology—EUROCRYPT 2026*, Springer, 92-123.
 - [17] Sabani, M.E., Savvas, I.K. and Garani, G. (2024) Learning with Errors: A Lattice-Based Keystone of Post-Quantum Cryptography. *Signals*, **5**, 216-243.
<https://doi.org/10.3390/signals5020012>
 - [18] Chhetri, G., Somvanshi, S., Hebli, P., Brotee, S. and Das, S. (2025) Post-Quantum Cryptography and Quantum-Safe Security: A Comprehensive Survey. arXiv: 2510.10436.
<https://doi.org/10.48550/arXiv.2510.10436>
 - [19] Sinha Roy, S. (2019) SaberX4: High-Throughput Software Implementation of Saber Key Encapsulation Mechanism. 2019 *IEEE 37th International Conference on Computer Design (ICCD)*, Abu Dhabi, 17-20 November 2019, 321-324.
<https://doi.org/10.1109/iccd46524.2019.00050>
 - [20] Hoffstein, J., Pipher, J. and Silverman, J.H. (1998) NTRU: A Ring-Based Public Key Cryptosystem. In: *Lecture Notes in Computer Science*, Springer, 267-288.
<https://doi.org/10.1007/bfb0054868>
 - [21] Peng, K.X. and Du, S.Y. (2024) Falcon Anti-Quantum Attack Scheme Based on 8-Qubits Channel RSP, PREPRINT (Version 1).
<https://doi.org/10.21203/rs.3.rs-5316539/v1>

- [22] Srivastava, V., Baksi, A. and Debnath, S.K. (2023) An Overview of Hash Based Signatures. IACR Cryptology ePrint Archive. <https://eprint.iacr.org/2023/411>
- [23] Paiva, T.B. and Terada, R. (2020) A Timing Attack on the HQC Encryption Scheme. In: *Lecture Notes in Computer Science*, Springer, 551-573. https://doi.org/10.1007/978-3-030-38471-5_22
- [24] Matthews, G.L. and McMillon, E. (2025) A Combinatorial Approach to Avoiding Weak Keys in the BIKE Cryptosystem. *Designs, Codes and Cryptography*, **93**, 3413-3436. <https://doi.org/10.1007/s10623-025-01643-7>
- [25] Bernstein, D.J., Dobraunig, C., Schwabe, P., *et al.* (2017) SPHINCS+ Submission to the NIST Post-Quantum Cryptography Project. <https://sphincs.org/data/sphincs%2B-specification.pdf>
- [26] Hülsing, A. and Gazdag, S.L. (2015) Hash-Based Signatures: An Outline for a New Standard. *NIST Workshop on Cybersecurity in a Post-Quantum World*, Gaithersburg, 2-3 April 2015, 1-12.
- [27] Varnita, L.S.M.S., Subramanyam, K., Narasimha, T.P., Bhandari, H., *et al.* (2024) A Study on Isogeny Based Cryptography. 2024 *International Conference on Electronics, Computing, Communication and Control Technology (ICECCC)*, Bengaluru, 2-3 May 2024, 1-6. <https://doi.org/10.1109/iceccc61767.2024.10593878>
- [28] Seo, H., Anastasova, M., Jalali, A. and Azarderakhsh, R. (2021) Supersingular Isogeny Key Encapsulation (SIKE) Round 2 on ARM Cortex-M4. *IEEE Transactions on Computers*, **70**, 1705-1718. <https://doi.org/10.1109/tc.2020.3023045>
- [29] Petzoldt, A. (2020) Efficient Key Generation for Rainbow. In: *Lecture Notes in Computer Science*, Springer, 92-107. https://doi.org/10.1007/978-3-030-44223-1_6
- [30] Beullens, W. (2021) Improved Cryptanalysis of UOV and Rainbow. In: *Lecture Notes in Computer Science*, Springer, 348-373. https://doi.org/10.1007/978-3-030-77870-5_13
- [31] Swayne, M. (2025) Quantum Computing Roadmaps: A Look at the Maps and Predictions of Major Players. The Quantum Insider. <https://thequantuminsider.com/2025/05/16/quantum-computing-roadmaps-a-look-at-the-maps-and-predictions-of-major-quantum-players/>
- [32] Chang, S.Y. and Khan, Q. (2026) Post-quantum Cryptography in Networking Protocols: Challenges, Solutions, and Future Directions. *Cryptography*, **10**, Article 12. <https://doi.org/10.3390/cryptography10010012>
- [33] Internet Pros Team (2026) Post-Quantum Cryptography: How PQC Is Protecting the Digital World from Quantum Threats in 2026. Internet Pros. <https://internet-pros.com/blog/post-quantum-cryptography-pqc-2026/>
- [34] Sonbul, O.S., Rashid, M. and Jaffar, A.Y. (2025) Accelerating Crystals-Kyber: High-Speed NTT Design with Optimized Pipelining and Modular Reduction. *Electronics*, **14**, Article 2122. <https://doi.org/10.3390/electronics14112122>