ORIGINAL RESEARCH ARTICLE UHD JOURNAL OF SCIENCE AND TECHNOLOGY

Optimization of Lattice-Based Cryptographic Key Generation using Genetic Algorithms for Post-Quantum Security

Tara Nawzad Ahmad Al Attar, Rebaz Nawzad Mohammed

Department of Computer Science, College of Science, University of Sulaimani, Sulaymaniyah, Iraq.

ABSTRACT

The progress of quantum computing has posed serious threats to classical cryptographic systems, necessitating much research into developing post-quantum cryptography (PQC). Of the schemes available in PQC, the strongest candidates appear to be lattice-based cryptography (LBC), which encompasses an ample security basis and good computation efficiency. However, practically implementing LBC is faced with key-generation and optimization difficulties, mainly because of its enormous key sizes and computational overhead. The research proposes a novel concept whereby genetic algorithms (GAs) are blended with LBC to increase the merits of key generation while guaranteeing security. Through the evolutionary capacity of GAs, the proposed method optimizes lattice-based keys through selection, crossover, and mutation to ensure high entropy and computationally feasible with experimental results indicating that the GA-based method can cut down memory requirements and computational complexity, making it favorable for resource-constrained environments such as the Internet of Things and embedded systems. The method thus suggested accelerates encryption speed and simultaneously strengthens the security of the optimized key structures. This study emphasizes evolutionary algorithms' potential to facilitate PQC advancement and provides a scalable and efficient framework for cryptographic systems.

Index Terms: Cryptographic Security, Genetic Algorithm, Key Optimization, Lattice-Based Cryptography, Post-Quantum Cryptography

1. INTRODUCTION

The launch of quantum computing has considerably challenged the standard cryptographic setups [1]. Existing classical cryptographic systems, such as RSA and Elliptic Curve Cryptography (ECC) [2], bank on mathematical complexities that are computationally very hard for standard computers but can be quite simply solved by quantum algorithms, such as Shor's algorithm [3]. This weakness has sped up the search for post-quantum computing (PQC) solutions and eventually led to the design of cryptosystems that are amenable to attack by quantum computers [4]. Lattice-based-cryptography (LBC) [5], with its robust security guarantees and great ease of computation has since been in the limelight. In this spirit, the National Institute of Standards and Technology (NIST) [6] has broadened its interests in forming global efforts to standardize PQC algorithms: Some, like lattices, are under scrutiny to become future standards for cryptographic schemes.

Lattice-based cryptography (LBC) is based on the extreme hardness of two computationally challenging lattice problems called the Shortest Vector Problem (SVP) and the Learning

Access this article online

DOI:10.21928/uhdjst.v9n1y2025.pp93-105

E-ISSN: 2521-4217

P-ISSN: 2521-4209

Corresponding author's e-mail: tara.ahmad@univsul.edu.iq

Received: 02-03-2025 Accepted: 27-03-2025 Published: 20-04-2025

with Errors (LWE) [7]. This kind of cryptography is considered a potent building block for digital signatures, key agreements, and encryption security for the quantum age [8]. However, the practical deployment of such systems faces challenges in key generation and optimization. Key sizes in large key lattice systems may yield troublesome memory sizing. At the same time, computational cost may restrict their applicability in low-resource environments such as the Internet of Things (IoT) and embedded systems [9].

LBC derives its security from the computational intractability of certain fundamental lattice problems, primarily the SVP and the LWE problem [7]. The SVP involves finding the shortest non-zero vector in a high-dimensional lattice, which is known to be NP-hard under specific norms [10]. The LWE problem, on the other hand, is based on solving linear equations with small random errors, making it resistant to both classical and quantum attacks. These problems serve as the basis for constructing secure encryption schemes, digital signatures, and key exchange protocols [11]. Unlike factorization-based cryptosystems, which are vulnerable to Shor's algorithm, the hardness of SVP and LWE remains intact even in the presence of quantum computers [12]. As a result, LBC has emerged as a leading candidate for PQC and is actively being standardized by NIST for future cryptographic applications.

To tackle these problems, genetic algorithms (GAs) are used in this study to address the optimization incorporating lattice-based cryptographic keys. GAs [13], which were taken from biology around 1970, are an efficient and effective technique for solving optimization problems in complex algebraic and other structures. GAs have, in so many words, implemented all the desirable features that were essential to hyperplanes' construction [14], such as characters, crossover/mutation, strategies, concealing, and non-linearities. Through the practice of GA, it is possible to refine lattice-based cryptographic keys by adopting an evolutionary multi-objective technique selection, crossover, and mutation. The GA and LBC concepts meshing enhance the key generation process by pruning the infeasible keyword combinations that do not guarantee high security.

Major Contributions:

- The paper introduces GAs in LBC modification, stating clear processes for obtaining security
- Our proposed change advances the efficiency of key generation, thus reducing resource waste for computation while guaranteeing security
- We mean improving key optimization, memory efficiency, and encryption speed and illustrate how the proposed method performs better for this particular purpose
- This article examines this technique within real-world cryptography systems resistant to quantum computer attacks, focusing on particularly resource-limited environments.

The remaining sections of this paper are as follows: Section 2 provides a comprehensive survey of PQC and lattice-based schemes and explanatory notes on optimization. Section 3 shows how GAs are combined with lattice-based key generation and explains the research strategy used. Section 4 describes an experimental study on the improvement method, including the performance measurement and the results obtained. Finally, Section 5 provides closure to the text and delves into the prospective directions of the study's development.

2. LITERATURE REVIEW

A serious threat to traditional cryptographic schemes by quantum computers has spurred the development of PQC – an arms race for creating cryptographic algorithms capable of withstanding quantum-level attacks. The quest for more robust encryption methods in the future of quantum computing has led the NIST to assume a leading role in standardizing such algorithms. It then seeks an extensive account of recent pasta and relevant methodologies of PQC in LBC, its applications, improved performance, and enhanced security against threats from quantum computers.

Quantum computers use quantum mechanical principles to perform tasks at exponentially faster speeds than classical ones. Notably, integer factorization, which is an example of the task, could be broken through the use of Shor's algorithm, thus making the most used cryptographic systems, such as RSA and ECC "dead." Although Shor's algorithm has only been deduced by theory, it is powerful or efficient enough to continue with any practical quantum computer for big-sized everyday applications. Thus, the emergence of PQC would keep it secure even against the quantum era. Standardization about resistance to quantum algorithms is yet crucial, and even candidates have been evaluated at NIST primarily based on complex mathematical problems, e.g., LBC [15].

The rapid advancements in quantum computing pose significant threats to traditional cryptographic systems, necessitating a transition toward PQC. UAVs, which rely on open wireless channels and have constrained computational resources, are particularly vulnerable to quantum-based attacks, making PQC integration crucial for securing their communications [16]. The NIST PQC standardization process has driven research toward hardware implementations, leading to the development of PQC hardware circuits and system design techniques to enhance security in various applications [17]. However, the migration of IP networks to PQC remains a complex challenge due to inconsistencies in migration steps, terminology, and security concerns, with present implementations being largely experimental [18]. A structured transition plan to PQC is essential, incorporating identification, protection, detection, and response phases to ensure a quantum-resilient cybersecurity infrastructure [19]. While hybrid cryptographic systems combining traditional and quantum-resistant techniques are being explored, significant challenges remain in practical deployment and regulatory alignment.

One of the candidates in the post-quantum computing (PQC) contest, LBC, has become a beautiful choice. LBC entails the solving of lattice-based problems such as the closest vector problem, the flexible LWE, and the ring LWE issues that are known to be classically and quantumly hard to solve [20]. LBC-based protocols are available for encryption, key exchange, digital signature, and even homomorphic encryption, and they can be used to secure various kinds of data and communication channels. Even though LBC is safe in practice, some real-world problems arise while trying to implement this scheme, where primary key and memory management structures may be inefficient, making it not convenient for use in areas where resources, such as embedded systems and IoT equipment, are few.

The global spread of IoT devices and the subsequent advancement in capabilities of mobile networks to 5G has necessitated the search and development of cryptosystems that will resist attacks facilitated by a quantum computer. IoT devices are of limited computational power, which requires lightweight cryptographic solutions that provide resistance to quantum attacks and are not stretched when it comes to performance [21]. Such a lightweight LBC instance is beneficial in such a scenario as it provides almost the same level of security as the ordinary LBC with approaches that decrease both memory usage and the level of computing resources in general [22]. In particular, a recent study provided efficient symmetric and asymmetric quantum-safe algorithms for an IoT environment, showing significant performance and memory cost improvements over the existing NIST-approved algorithms [21].

With its abundance of highly complex operations, the prospect of LBC has put a lot of pressure on its developers to improve performance [23]. Various forms of vectorization have been instrumental in enhancing lattice-based schemes, namely key exchange protocols. The section has improved and reduced time and energy consumption [24]. Finally, attention was focused on the problems associated with enlarging keys and using lattice-based systems. An example of such systems is NTRU systems, which can provide faster encryption and key generation than RSA but also suffer from the problem of the large sizes of the keys, which renders them unsuitable for many users [25].

LBC has the potential to address challenging security issues, but great care is still required to protect all such systems from the effects of quantum computers. There continue to be hazards associated with lattice structures and their operations, irrespective of whether they are hardware or software components. For that reason, research in this area is necessary to improve the security for LBC and come up with workable measures in the wake of new threats, such as those that may rely on stealth, as seen in side channel methodologies or physical access [26]. Moreover, the transformation of NIST's PQC faceless approach has to consider the security necessity along with the practicality of memory consumed, scheduler cycles utilized, and power consumed by the devices to ensure that the said tablets satisfy the limits of different hardware equipment [27].

Cryptographic systems that are lattice-based have arisen as the answer to the problem of post-quantum vulnerability. This security concern is determined by the specific algorithms encountered while encrypting information. Although the security model of such systems is rather strong, certain complexity is observed in their practical implementation in environments where resources are limited, such as the IOTs or Edge processing. Optimizing LBC production or incorporating other LBC optimal techniques, such as vectorization, has dramatically increased the ability of LBC to adapt to different appliances. Moreover, it may be expedient to evaluate the security, performance, and other parameters in scale as pioneers of this technology accept other changes that are inevitable toward quantum computers. NIST's attempts to invent Projected Quantum Computation Standalone systems for the future are significant for the following safety standards needed. A comparison of the most relevant related works is provided in Table 1 to highlight the differences and advantages of the proposed method.

3. MATERIALS AND METHODS

3.1. PQC and LBC

PQC [28] is a field of cryptographic research that aims to develop secure cryptographic algorithms resilient to attacks by quantum computers. Classical cryptographic schemes such as RSA and ECC rely on problems, such as integer factorization and discrete logarithms, which are efficiently solvable using Shor's algorithm on a sufficiently powerful quantum computer. PQC schemes are designed to counteract this vulnerability based on problems that remain computationally intractable even in the presence of quantum adversaries.

LBC has emerged as one of the most promising candidates for post-quantum secure cryptographic protocols. It relies on the computational hardness of lattice problems, such as the SVP and the LWE problem. Given a basis $B = \{b_1|, b_2|\ldots,b_n|\}$ of an n-dimensional lattice Λ, the SVP requires finding the shortest non-zero vector $v \in \Lambda$ which is computationally difficult even for quantum computers. The LWE problem, introduced by Regev, is formulated as follows: given a set of linear equations with small noise, recovering the original secret vector s is assumed to be hard. Mathematically, LWE can be defined as Eq(1).

$$A{\cdot}s + e = b \bmod q \tag{1}$$

Where $A \in \mathbb{Z}_q^{m\times n}$ is a randomly chosen matrix, $s \in \mathbb{Z}_q^n$ is the secret vector, and $e \in \mathbb{Z}_q^m$ is an error term sampled from a small noise distribution. Recovering s given (A,b) is believed to be computationally infeasible for large enough parameters. LBC provides efficient and scalable constructions for encryption, digital signatures, and key exchange protocols.

3.2. GA

GA [13] is an evolutionary optimization technique inspired by natural selection and genetics principles. It is widely employed to solve complex optimization problems where traditional mathematical or heuristic approaches may be inefficient. GA operates on a population of candidate solutions, evolving them iteratively through selection, crossover, and mutation operations to enhance solution quality. The fitness function, denoted as f(x), evaluates the suitability of each candidate solution x within the search space.

The algorithm begins with an initial population $P = \{x_1, x_2,\ldots,x_n\}$, where each x_i is a chromosome encoding a potential solution. Selection is performed to retain high-fitness individuals, commonly using roulette wheel selection, tournament selection, or rank-based selection. In roulette wheel selection, the probability of selecting an individual x_i is proportional to its fitness:

$$P(x_i) = \frac{f(x_i)}{\sum_{j=1}^{n} f(x_i)} \tag{2}$$

Crossover, or recombination, involves exchanging genetic material between selected parents to create offspring. A common approach is single-point crossover, where a random crossover point c is chosen, and genes are swapped between two parents to produce new chromosomes:

$$Child_1 = (Parent_1[: c], Parent_2\ [c:]) \tag{3}$$

$$Child_2 = (Parent_2[: c], Parent_1\ [c:]) \tag{4}$$

Mutation introduces small random alterations to maintain genetic diversity and prevent pre-mature convergence. If x_i is a binary-encoded chromosome, a bit-flip mutation at position j can be expressed as:

$$x_{ij} = 1\text{-}x_{ij} \tag{5}$$

Where x_{ij} is the j^{th} gene of chromosome x_i. The algorithm iterates through these operations until a termination criterion is met, such as achieving a pre-defined fitness threshold or reaching a maximum number of generations.

3.3. Proposed Method

This study presents a hybrid cryptographic framework hybrids GA and lattice-based encryption to enhance security and adaptability. The method consists of four main stages: key optimization, encryption, mutation, and decryption. Key optimization is performed using GA to generate an invertible encryption key matrix K over Z_{256}. The fitness function ensures the invertibility of K by verifying that its determinant satisfies $gcd(det(K),256) = 1$ and that its modular inverse exists in modulo 256. The fitness function is defined as:

$$F(K) = \begin{cases} \sum K, if\ \gcd(\det(k), 256) = 1 \\ -\infty, otherwise \end{cases} \tag{6}$$

Where $\sum K$ represents the sum of all elements in the key matrix. The GA iteratively refines K using selection, crossover, and mutation to obtain an optimal encryption key. Encryption transforms the plaintext message P into its numerical representation and computes the ciphertext C using matrix multiplication in modular arithmetic:

$$C = P \cdot K \ mod\ 256 \tag{7}$$

Where padding is applied to align with encryption parameters. To enhance security, a GA-based mutation process introduces controlled randomness by flipping bits in C, further obfuscating the encrypted text and increasing resistance against attacks. Decryption is performed using the modular inverse of the key matrix, ensuring correct retrieval of the original message:

$$P' = C \cdot K^{-1} \ mod\ 256 \tag{8}$$

Where P' ideally reconstructs the original plaintext. The invertibility of K ensures accurate recovery of P without information loss. Integrating GA-based optimization with lattice cryptography provides a robust mechanism resistant to conventional attacks. The introduced noise through mutation adds a layer of obfuscation. The proposed scheme offers a potential post-quantum alternative, leveraging the hardness of modular matrix inversion while maintaining computational efficiency.

To address the potential security vulnerabilities associated with using GAs for cryptographic key generation, we acknowledge that while GAs offer efficiency improvements, they may introduce risks. Specifically, the optimization process might lead to patterns or predictable sequences in the generated keys, which could be exploited by attackers. This concern arises because GAs are inherently stochastic, and if not carefully designed, they may converge to weak or vulnerable keys, thereby reducing the security of the encryption system. In addition, if the search space is not sufficiently large or diverse, or if the genetic operators (such as selection, crossover, and mutation) are not effectively applied, the randomness required for secure cryptographic keys could be compromised, making them more susceptible to cryptanalysis techniques such as brute-force or statistical attacks.

To mitigate these risks, we have incorporated mechanisms within our GA-based key generation process to preserve randomness and avoid pre-mature convergence. This includes ensuring that the search space remains large and diverse, and that proper randomization techniques are used. We also recommend conducting further security evaluations, such as testing against known cryptographic attacks (e.g., differential and linear cryptanalysis) and assessing the strength of GA-generated keys in realistic environments. While GAs can enhance the efficiency of cryptographic key generation, it is essential to carefully design and test the approach to prevent the introduction of vulnerabilities and ensure its applicability in secure cryptographic systems. The flowchart of the complete process is illustrated in Figure 1, providing a visual representation of the step-by-step workflow.

3.4. Computational Complexity Analysis

The computational complexity of the proposed method, "Optimization of Lattice-Based Cryptographic Key Generation Using Genetic Algorithms for Post-Quantum Security," is evaluated by considering the key stages: key optimization using GAs, encryption, mutation, and decryption. In the key optimization stage, the GA iterates over populations of candidate solutions. Selection evaluates each matrix based on the fitness function, which requires computing the determinant and verifying the invertibility condition. The time complexity of this operation is O (n^3). Crossover combines matrices with complexity O (n^2), and mutation flips bits in the matrix, also requiring O (n^2). Evaluating the fitness function involves determinant and modular inverse calculations, leading to a total complexity of O ($n^3 + n^2$) per generation.

For encryption, the process involves matrix multiplication between the plaintext and the key matrix, with a complexity of O (n^2). Similarly, the mutation step in the ciphertext involves flipping bits, which also takes O (n^2). Decryption requires matrix multiplication with the inverse of the key matrix, along with modular inverse computation, resulting in a complexity of O ($n^3 + n^2$).

The overall time complexity of the method is dominated by the GA key optimization and can be expressed as O ($g \cdot p \cdot (n^3 + n^2)$), where g is the number of generations, p is the population size, and n is the matrix dimension. The total complexity also includes the O ($n^3 + n^2$) cost of encryption, mutation, and decryption. While the GA introduces computational overhead due to its iterative nature, it provides a flexible approach to solving NP-hard optimization problems, such as LBC, which is resistant to quantum attacks. This hybrid approach offers strong security guarantees while maintaining efficiency within the constraints of the cryptographic operations. The overall computational complexity of the proposed method is dominated by the key optimization process using the GA. The total complexity of the method can be expressed as:

TABLE 1: Comparison of the related works

Study	PQC algorithm	Security focus	Key feature	Performance (Speed)	Memory usage	Platform	Optimizations/ Enhancements
Farooq *et al*. [15]	NIST PQC finalists	Post-Quantum Security	Key generation, encapsulation, decapsulation	Not specified	Not specified	General-purpose computing	Comparative analysis of efficiency
Nejatollahi *et al*. [20]	Lattice-based cryptography	Encryption, Digital Signatures, Key Exchange, Homomorphic Encryption	Strong foundational properties for PQC	Not specified	High for classical systems	General-purpose and emerging systems	Software and hardware implementation challenges
Asif [22]	Lightweight LBC (LW-LBC)	IoT Security	Lightweight lattice-based schemes for IoT	×70 faster (symmetric), ×10 faster (asymmetric)	×6000 less memory than NIST algorithms	Resource-constrained IoT devices	Optimized for energy efficiency and speed
Kaushik *et al*. [21]	Symmetric and Asymmetric Post-Quantum Algorithms	IoT and 5G Networks	Data stream encryption for IoT and 5G	Symmetric: ×70 faster, Asymmetric: ×10 faster	×6000 less memory than NIST algorithms	IoT and 5G environments	Focused on quantum security and efficiency
Koteshwara *et al*. [23]	CRYSTALS-Kyber KEM SHA3	Key Exchange (PQC)	Polynomial multiplication optimization	52% improvement over traditional methods	Not specified	High-performance computing	Vectorization improvements
Alkim *et al*. [24]	Ring-LWE-based Key Exchange	Post-Quantum Security for TLS	Improved error distribution and reconciliation	×8 speedup in portable C, ×27 in optimized Intel CPU version	Reduced communication overhead	High-performance TLS implementations	Enhanced error distribution and optimization
Gagnidze *et al*. [25]	NTRU-based Cryptosystem	Post-Quantum Security	Faster encryption, key generation	Faster encryption and key generation than RSA	Larger key sizes, variable signature sizes	General-purpose systems	Optimization for faster encryption
Liu *et al*. [26]	Lattice-based Cryptosystems	Edge Computing Security	Efficient lattice implementations for microcontrollers	Not specified	Optimized for 8 and 32-bit systems	Embedded systems and microcontrollers	Efficient for constrained devices
Roma *et al*. [27]	Various PQC Candidates	Post-Quantum Security	Energy consumption evaluation of PQC schemes	Not specified	Categorized by security level	Intel Core i7-6700 CPU	Focused on energy efficiency and optimization

PQC: Post-quantum cryptography, NIST: National Institute of Standards and Technology, LBC: Lattice-based cryptography, LWE: Learning with errors, IoT: Internet of Things

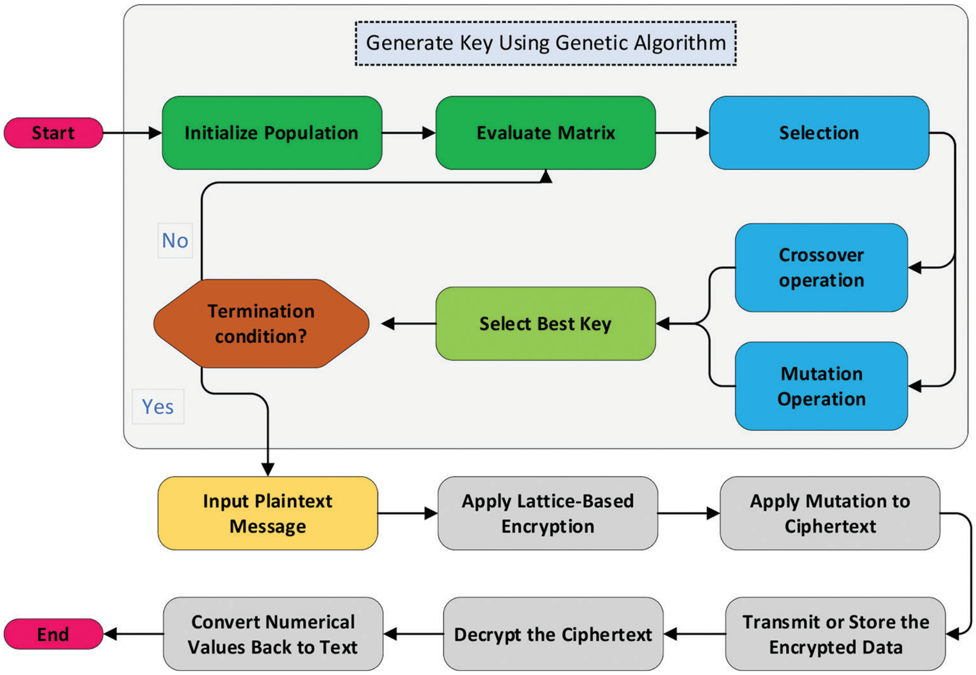

Fig. 1. Flowchart of the proposed method.

$O(g \cdot p \cdot (n^3 + n^2)) + O(n^3 + n^2)$. where g is the number of generations in the GA, p is the population size, and n is the dimension of the key matrix.

4. EXPERIMENTAL EVALUATION

The experimental evaluation of the proposed method was conducted with a population size of 50 and a maximum iteration count of 50. The implementation was implemented in Python and executed on a Core i7-13650HX processor with 24GB of RAM. To optimize the efficiency of the GA in cryptographic key generation, we used the face-centered central composite design method [29], which tests parameters such as population size, mutation rate, and number of generations at three levels: low, medium, and high. This systematic approach allows for a thorough exploration of the parameter space to identify optimal settings. The following tables and figures comprehensively assess the method's performance in encryption, decryption, and mutation processes.

Table 2 presents the optimized key matrices derived from the proposed method. The optimized key structure is critical to encryption security, ensuring randomness and unpredictability while maintaining computational efficiency. The displayed key values highlight the capability of the GA in optimizing cryptographic parameters. The key matrix exhibits a well-distributed pattern, reinforcing the robustness of the encryption scheme. These optimized keys are instrumental in generating secure ciphertext, thereby preventing unauthorized decryption attempts. The structured yet dynamic nature of the key suggests that the GA is effective in producing high-entropy keys, which is crucial for strong encryption. The results confirm that the proposed approach generates optimized encryption keys that enhance security and balance complexity and computational feasibility. By ensuring a high degree of randomness, the key optimization process significantly improves the overall encryption robustness, making it resistant to cryptanalysis.

TABLE 2: Optimized Keys achieved by the proposed method

Optimized key:	[[175 216 252 239] [131 239 239 213] [239 252 213 251] [187 239 251 216]]

Fig. 2 illustrates the convergence behavior of the GA in optimizing the key generation process for lattice-based cryptographic schemes. The x-axis represents the number of generations, while the y-axis denotes the best fitness value observed during the optimization process. As the number of generations increases, the fitness value improves consistently, demonstrating the effectiveness of the GA in evolving optimal cryptographic keys. The initial fluctuations in the fitness values indicate the exploration phase of the algorithm, where diverse candidate solutions are evaluated. As the generations progress, the curve exhibits a steady upward trend, reflecting the exploitation phase where the algorithm refines solutions to achieve higher fitness values. The smooth yet incremental improvement in fitness confirms the stability of the GA and its ability to converge toward an optimized key structure. This result validates the efficacy of using GA-based optimization in post-quantum cryptographic key generation by reducing computational overhead while maintaining high-security standards.

Table 3 presents the encryption and mutation results for short ciphertexts containing between 1 and 100 characters.

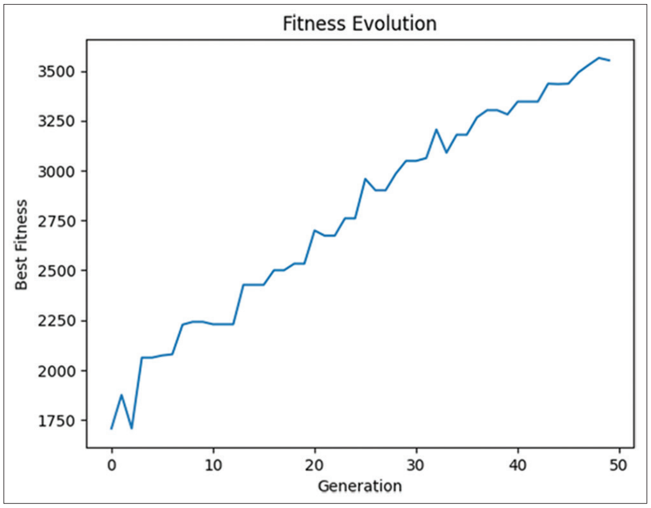

Fig. 2. The convergence curve is attained by the proposed method.

TABLE 3: Short ciphertext and mutation analysis 1–100 characters

Original:	Moonlit dreams drift upon the waves, lost in endless time.
Decrypted (GA-based key):	Moonlit dreams drift upon the waves, lost in endless time.
Original Ciphertext:	[[107, 143, 162, 227], [59, 55, 59, 237], [216, 201, 2, 21], [72, 49, 85, 50], [159, 235, 217, 173], [100, 28, 192, 177], [214, 248, 164, 206], [95, 203, 106, 112], [218, 35, 70, 20], [198, 117, 112, 153], [13, 206, 71, 175], [77, 239, 93, 35], [225, 252, 11, 166], [226, 107, 8, 79], [213, 138, 94, 241]]
Mutated Ciphertext:	[[107, 143, 163, 227], [58, 55, 58, 237], [217, 200, 3, 21], [73, 49, 85, 50], [158, 235, 216, 173], [101, 28, 193, 177], [215, 248, 165, 206], [94, 203, 107, 113], [219, 34, 70, 21], [199, 117, 112, 153], [12, 207, 71, 174], [77, 238, 93, 34], [225, 252, 10, 167], [226, 106, 9, 79], [213, 138, 95, 241]]

The original plaintext and its corresponding decrypted output confirm the accuracy of the GA-based key recovery. The exact match between the decrypted text and the original message indicates that the proposed decryption approach successfully reconstructs the original input without introducing errors. Furthermore, the analysis of original and mutated ciphertexts reveals subtle alterations applied to the encoded text, a critical feature in enhancing security. These controlled mutations introduce diversity in the ciphertext representation, ensuring that even minor variations in the key can lead to significantly different encrypted outputs. This characteristic is essential in preventing pattern-based cryptanalysis while maintaining the ability to decrypt the message when the correct key is used accurately. The results validate the proposed approach's effectiveness in ensuring secure communication with a robust decryption mechanism.

Table 4 extends the ciphertext analysis to medium-length messages, ranging from 101 to 200 characters. Similar to the previous case, the decrypted text precisely matches the original, further establishing the reliability of the GA-based decryption mechanism. The analysis of the ciphertext mutations reveals systematic modifications applied to the encrypted message. These modifications serve the dual purpose of enhancing security while preserving the integrity of the decryption process. By introducing slight perturbations in the ciphertext structure, the mutation process ensures that even if an attacker gains partial knowledge of the encryption method, reconstructing the exact plaintext without the key remains computationally infeasible. The results highlight the robustness of the proposed method in handling larger messages while maintaining high decryption accuracy. This adaptability to varying message lengths is crucial to practical cryptographic implementations, making the method viable for real-world applications requiring secure communication.

Table 5 evaluates the performance of the proposed approach on long ciphertexts containing between 201 and 300 characters. The results reinforce the algorithm's ability to decrypt lengthy messages accurately. The decrypted output remains consistent with the original, demonstrating the method's scalability and reliability. The analysis of the mutated ciphertexts further substantiates the security-enhancing capabilities of the proposed approach. The systematic variations introduced in the encrypted text ensure that even slight modifications to the input lead to substantial changes in the ciphertext, a desirable property in cryptographic systems. These controlled mutations make it challenging for adversaries to exploit patterns or predict key variations, strengthening encryption security. The successful

TABLE 4: Medium-length ciphertext and mutation evolution 101–200 characters

Original:	Through silent nights and golden dawns, the stars still weave their tales, whispering love to the restless tides.
Decrypted (GA-based key):	Through silent nights and golden dawns, the stars still weave their tales, whispering love to the restless tides.
Original Ciphertext:	[[39, 209, 119, 98], [40, 33, 61, 230], [243, 170, 30, 214], [200, 78, 14, 102], [80, 109, 201, 146], [230, 22, 163, 232], [250, 69, 216, 161], [165, 154, 143, 223], [136, 241, 62, 151], [127, 93, 161, 133], [47, 68, 177, 219], [206, 62, 251, 44], [22, 104, 127, 158], [63, 156, 108, 199], [201, 89, 214, 205], [175, 96, 200, 103], [148, 87, 173, 110], [138, 10, 162, 29], [238, 181, 149, 193], [0, 139, 230, 182], [194, 180, 17, 232], [178, 185, 245, 85], [245, 167, 207, 115], [133, 208, 160, 189], [160, 155, 45, 233], [148, 239, 99, 253], [18, 49, 237, 76], [63, 189, 130, 74], [18, 16, 40, 242]]
Mutated Ciphertext:	[[38, 209, 118, 98], [41, 32, 61, 230], [243, 171, 31, 214], [200, 78, 14, 102], [80, 109, 201, 146], [231, 22, 163, 233], [251, 68, 217, 161], [165, 154, 142, 223], [136, 240, 63, 150], [126, 93, 161, 133], [46, 68, 176, 218], [207, 62, 250, 45], [22, 105, 126, 159], [63, 157, 108, 199], [200, 88, 214, 205], [175, 97, 200, 103], [149, 87, 173, 110], [138, 11, 162, 28], [238, 181, 148, 193], [1, 139, 231, 183], [194, 180, 16, 232], [179, 184, 245, 85], [245, 167, 206, 115], [132, 209, 160, 189], [161, 154, 45, 233], [149, 239, 99, 253], [19, 48, 237, 76], [62, 188, 130, 74], [19, 16, 40, 243]]

TABLE 5: Long Ciphertext with GA-based decryption and mutations 201–300 characters

Original:	Beneath the silver glow of distant constellations, the ocean sings its endless song, where fleeting echoes of love and longing intertwine, carried on the breath of time, forever dancing between the shores of memory and fate.
Decrypted (GA-based key):	Beneath the silver glow of distant constellations, the ocean sings its endless song, where fleeting echoes of love and longing intertwine, carried on the breath of time, forever dancing between the shores of memory and fate.
Original Ciphertext:	[[70, 142, 208, 185], [35, 100, 176, 11], [47, 68, 177, 219], [94, 137, 201, 46], [126, 79, 119, 109], [26, 197, 84, 220], [255, 190, 42, 63], [199, 180, 88, 74], [191, 9, 69, 30], [68, 218, 97, 200], [254, 235, 55, 227], [2, 221, 108, 78], [189, 232, 164, 57], [188, 44, 32, 41], [69, 1, 78, 97], [26, 107, 20, 82], [213, 204, 84, 208], [204, 8, 52, 147], [249, 199, 135, 2], [87, 66, 242, 116], [120, 210, 205, 68], [164, 196, 40, 19], [255, 53, 37, 231], [138, 35, 96, 196], [154, 142, 33, 138], [225, 166, 54, 74], [217, 182, 62, 2], [3, 225, 249, 102], [20, 234, 46, 107], [34, 128, 40, 102], [7, 197, 156, 188], [122, 128, 4, 93], [31, 246, 99, 109], [146, 7, 128, 18], [192, 57, 49, 207], [198, 133, 183, 199], [44, 181, 61, 135], [214, 248, 164, 206], [47, 254, 156, 49], [242, 15, 119, 124], [71, 233, 95, 61], [145, 126, 239, 192], [195, 9, 227, 94], [142, 238, 184, 97], [133, 15, 103, 242], [60, 75, 236, 212], [14, 139, 213, 23], [67, 176, 89, 142], [214, 248, 164, 206], [192, 100, 243, 108], [131, 223, 124, 138], [71, 233, 95, 61], [10, 48, 29, 83], [148, 54, 234, 83], [192, 230, 70, 86], [144, 130, 235, 234]]
Mutated Ciphertext:	[[71, 142, 209, 185], [34, 100, 177, 10], [46, 69, 177, 218], [94, 137, 201, 47], [126, 78, 119, 108], [26, 197, 85, 220], [255, 191, 43, 62], [198, 180, 88, 75], [190, 9, 68, 31], [68, 218, 97, 200], [254, 234, 55, 226], [3, 221, 109, 79], [189, 233, 165, 57], [188, 45, 33, 40], [68, 1, 78, 97], [27, 107, 20, 82], [213, 205, 84, 208], [204, 9, 52, 147], [249, 199, 135, 3], [86, 67, 243, 117], [121, 210, 204, 68], [164, 196, 40, 18], [255, 53, 37, 230], [138, 35, 96, 196], [154, 142, 33, 138], [225, 166, 55, 74], [216, 182, 62, 3], [2, 225, 249, 103], [21, 235, 47, 106], [34, 129, 41, 102], [6, 197, 157, 189], [123, 129, 4, 92], [31, 247, 99, 109], [147, 6, 128, 19], [192, 56, 49, 206], [198, 132, 183, 199], [45, 180, 61, 134], [215, 248, 165, 206], [46, 255, 157, 48], [242, 14, 118, 125], [71, 233, 94, 61], [145, 126, 239, 193], [194, 8, 226, 94], [142, 239, 184, 96], [133, 15, 103, 242], [60, 75, 237, 213], [14, 138, 212, 23], [66, 176, 88, 142], [215, 248, 165, 206], [192, 100, 243, 109], [130, 223, 125, 138], [70, 233, 95, 60], [11, 48, 28, 82], [148, 54, 235, 83], [192, 230, 70, 87], [144, 131, 234, 235]]

decryption of long messages while preserving security properties establishes the proposed method as a robust solution for high-volume data encryption scenarios, such as secure messaging and data storage.

Table 6 presents another analysis of long ciphertexts, reaffirming the consistency of the proposed method in handling extensive encryption and decryption tasks. The decrypted output aligns perfectly with the original message, confirming the reliability of the GA-based approach. The mutations applied to the ciphertext exhibit controlled randomness, ensuring that each encryption process yields a unique representation of the plaintext. This diversity in ciphertext generation enhances security by preventing pattern recognition and statistical attacks. The ability to maintain decryption accuracy despite introducing systematic variations in the ciphertext underlines the robustness of the encryption framework. These findings further validate the scalability and efficiency of the proposed approach, making it a viable candidate for real-world cryptographic applications involving long-form textual data.

Table 7 extends the evaluation to an additional long ciphertext, reinforcing the established trends observed in previous analyses. The decrypted message matches the original text, highlighting the precision and effectiveness of the GA-based decryption mechanism. The mutation process continues to demonstrate its role in enhancing security by

TABLE 6: Long ciphertext with GA-based decryption and mutations 201–300 characters

Original:	The moon, a silent guardian over restless seas, casts silver light upon forgotten echoes. The wind whispers across endless waves, carrying secrets of lovers lost to time. Each tide is a story, each ripples a song, weaving eternity’s tale in a fluid motion.
Decrypted (GA-based key):	The moon, a silent guardian over restless seas, casts silver light upon forgotten echoes. The wind whispers across endless waves, carrying secrets of lovers lost to time. Each tide is a story, each ripples a song, weaving eternity’s tale in fluid motion.
Original Ciphertext:	[[79, 68, 49, 251], [75, 143, 34, 195], [99, 92, 69, 207], [243, 170, 30, 214], [171, 197, 49, 126], [24, 219, 161, 22], [124, 79, 209, 150], [212, 172, 61, 6], [130, 55, 184, 201], [220, 125, 206, 55], [33, 103, 202, 246], [156, 101, 213, 98], [105, 151, 110, 195], [13, 35, 182, 86], [199, 36, 73, 57], [104, 89, 174, 151], [203, 199, 3, 180], [111, 73, 199, 197], [50, 18, 224, 67], [80, 111, 27, 25], [46, 137, 66, 225], [49, 42, 203, 130], [54, 120, 4, 174], [55, 67, 66, 48], [43, 197, 241, 174], [0, 139, 230, 182], [30, 186, 233, 230], [123, 30, 152, 92], [29, 48, 56, 164], [249, 199, 135, 2], [67, 254, 222, 212], [253, 203, 112, 204], [188, 3, 170, 157], [238, 193, 70, 131], [200, 214, 210, 205], [233, 40, 186, 168], [26, 94, 2, 3], [3, 225, 249, 102], [200, 112, 14, 1], [198, 117, 112, 153], [13, 145, 105, 80], [226, 107, 8, 79], [220, 21, 165, 41], [112, 133, 209, 230], [42, 162, 114, 237], [100, 108, 112, 253], [234, 176, 78, 133], [154, 251, 84, 147], [11, 142, 242, 133], [139, 60, 0, 39], [90, 116, 97, 6], [89, 26, 88, 122], [84, 111, 58, 153], [18, 16, 221, 128], [208, 128, 9, 176], [198, 245, 83, 151], [143, 208, 181, 132], [175, 156, 101, 159], [204, 38, 131, 174], [93, 197, 77, 90], [114, 55, 55, 147], [89, 241, 194, 192], [252, 34, 58, 88]]
Mutated Ciphertext:	[[79, 68, 48, 250], [74, 143, 34, 195], [99, 93, 68, 206], [243, 170, 30, 215], [171, 197, 48, 126], [24, 218, 161, 23], [124, 79, 209, 150], [212, 173, 60, 6], [130, 55, 185, 201], [220, 125, 206, 55], [32, 103, 203, 247], [156, 100, 213, 98], [104, 151, 110, 194], [13, 34, 183, 87], [199, 37, 73, 56], [105, 88, 175, 151], [203, 199, 3, 180], [111, 73, 199, 197], [51, 18, 224, 66], [81, 111, 26, 24], [46, 136, 67, 224], [49, 43, 203, 131], [55, 120, 5, 174], [55, 67, 66, 48], [42, 197, 240, 174], [0, 138, 230, 183], [31, 186, 233, 231], [123, 30, 153, 93], [28, 49, 56, 164], [249, 199, 134, 3], [67, 255, 223, 213], [252, 202, 112, 205], [189, 3, 171, 157], [239, 193, 71, 130], [201, 215, 210, 205], [232, 41, 186, 169], [27, 95, 2, 3], [2, 225, 248, 103], [200, 113, 14, 0], [199, 116, 112, 152], [13, 145, 105, 80], [227, 106, 8, 79], [221, 21, 165, 40], [112, 133, 208, 231], [43, 163, 114, 236], [101, 108, 113, 253], [235, 176, 79, 133], [154, 250, 84, 146], [10, 142, 243, 133], [138, 61, 0, 39], [90, 117, 97, 6], [89, 26, 88, 122], [85, 110, 58, 152], [19, 17, 220, 128], [209, 129, 9, 177], [198, 245, 83, 151], [142, 209, 180, 133], [174, 157, 101, 158], [205, 39, 130, 175], [92, 196, 77, 90], [115, 55, 54, 147], [89, 240, 194, 193], [253, 35, 59, 89]]

TABLE 7: Long ciphertext with GA-based decryption and mutations 201–300 characters

Original:	In the vast expanse of the night sky, constellations tell the myths of ages past. The ocean listens, its surface shimmering with celestial reflections, while the wind carries whispers of dreams yet to be fulfilled. Beneath these heavens, the heart longs, ever entwined with the call of destiny.
Decrypted (GA-based key):	In the vast expanse of the night sky, constellations tell the myths of ages past. The ocean listens, its surface shimmering with celestial reflections, while the wind carries whispers of dreams yet to be fulfilled. Beneath these heavens, the heart longs, ever entwined with the call of destiny.
Original Ciphertext:	[[205, 22, 234, 237], [217, 181, 253, 17], [212, 69, 189, 250], [222, 15, 191, 203], [182, 121, 14, 104], [175, 174, 218, 191], [1, 61, 37, 81], [80, 109, 201, 146], [1, 168, 135, 144], [246, 21, 100, 109], [30, 168, 48, 85], [99, 188, 149, 171], [71, 222, 68, 68], [107, 140, 185, 139], [120, 72, 172, 240], [145, 91, 160, 226], [65, 12, 87, 149], [206, 241, 57, 183], [213, 71, 190, 243], [76, 143, 58, 230], [54, 120, 4, 174], [93, 201, 184, 72], [192, 239, 225, 218], [168, 167, 194, 210], [214, 50, 241, 178], [232, 148, 204, 1], [58, 247, 252, 118], [145, 94, 93, 56], [132, 196, 88, 31], [219, 149, 222, 171], [154, 142, 33, 138], [88, 183, 167, 242], [184, 173, 218, 102], [227, 204, 64, 78], [185, 154, 182, 251], [216, 13, 137, 69], [37, 209, 242, 140], [127, 93, 161, 133], [124, 176, 125, 180], [175, 96, 200, 103], [55, 67, 66, 48], [143, 25, 85, 206], [198, 133, 183, 199], [177, 46, 22, 194], [0, 139, 230, 182], [30, 186, 233, 230], [255, 190, 42, 63], [203, 186, 23, 138], [187, 79, 200, 216], [13, 145, 105, 80], [177, 202, 103, 113], [239, 213, 129, 135], [166, 7, 19, 255], [252, 112, 184, 178], [123, 229, 66, 176], [32, 196, 164, 20], [235, 138, 97, 26], [62, 19, 188, 71], [12, 128, 186, 21], [184, 72, 172, 48], [74, 195, 219, 27], [81, 86, 206, 37], [144, 50, 116, 241], [200, 231, 239, 145], [39, 243, 173, 249], [142, 51, 79, 245], [104, 18, 35, 244], [136, 193, 178, 182], [188, 232, 188, 52], [179, 27, 198, 212], [13, 21, 153, 184], [61, 171, 131, 30], [90, 98, 183, 164], [129, 106, 14, 157]]
Mutated Ciphertext:	[[204, 23, 234, 236], [217, 180, 253, 17], [212, 69, 189, 250], [223, 14, 190, 202], [183, 120, 15, 104], [174, 174, 219, 190], [0, 60, 37, 80], [80, 108, 201, 147], [0, 168, 134, 144], [247, 21, 101, 108], [30, 169, 49, 84], [98, 188, 149, 171], [70, 223, 69, 68], [106, 140, 185, 139], [120, 73, 172, 241], [145, 91, 160, 227], [64, 12, 87, 149], [206, 240, 57, 182], [212, 70, 190, 242], [77, 142, 58, 230], [54, 121, 5, 174], [92, 201, 185, 72], [192, 239, 224, 218], [169, 167, 194, 211], [215, 51, 241, 178], [232, 148, 205, 1], [59, 247, 253, 118], [144, 95, 93, 57], [132, 197, 88, 31], [219, 148, 222, 170], [155, 143, 33, 139], [88, 183, 166, 243], [185, 172, 219, 103], [226, 205, 64, 78], [184, 154, 182, 250], [217, 12, 137, 69], [37, 209, 243, 140], [127, 92, 160, 132], [125, 176, 125, 181], [175, 96, 201, 102], [55, 67, 66, 48], [143, 25, 84, 206], [199, 133, 183, 198], [177, 46, 22, 194], [1, 138, 230, 182], [30, 187, 233, 230], [254, 190, 43, 63], [203, 187, 22, 139], [187, 78, 200, 217], [13, 144, 104, 81], [177, 202, 102, 113], [238, 213, 129, 135], [167, 7, 18, 255], [253, 112, 185, 179], [122, 228, 66, 177], [33, 197, 165, 20], [235, 138, 97, 26], [62, 18, 189, 71], [12, 129, 187, 21], [185, 72, 173, 48], [74, 195, 219, 26], [81, 87, 207, 37], [145, 51, 116, 240], [200, 230, 239, 144], [39, 242, 173, 249], [143, 51, 78, 245], [105, 19, 34, 245], [136, 193, 178, 182], [188, 233, 189, 53], [179, 27, 198, 212], [13, 20, 152, 185], [60, 171, 130, 30], [91, 98, 183, 164], [128, 107, 15, 157]]

introducing structured variations in the encrypted output. These mutations contribute to the overall unpredictability of the ciphertext, making it increasingly resistant to cryptographic attacks. The results further support the applicability of the proposed method in diverse encryption scenarios where security and accuracy are paramount. The method's ability to adapt to different message lengths while preserving encryption integrity ensures its practical utility in securing sensitive communications.

Table 8 examines the performance of the proposed approach on very long ciphertexts exceeding 301 characters. The results affirm the scalability and reliability of the encryption-decryption framework in handling extensive textual data. The decrypted output accurately reconstructs the original message, demonstrating the robustness of the GA-based key generation and recovery process. The mutations applied to the ciphertext introduce calculated variations, reinforcing encryption security. These systematic alterations ensure that even minor differences in the key result in significantly distinct ciphertext outputs, a property essential for preventing decryption by unauthorized entities. The findings confirm the proposed method is well-suited for high-security applications, where message integrity

TABLE 8: Very long ciphertext with GA-based decryption and mutations of 301+characters

Original:	Through the endless corridors of time, where memories drift, such as echoes upon the wind and the soul seeks the light of understanding. Each footstep is a verse in the poetry of existence; each breath is a note in the universe's symphony. Love lingers in the silent embrace of the cosmos, waiting to be found anew.
Decrypted (GA-based key):	Through the endless corridors of time, where memories drift, such as echoes upon the wind and the soul seeks the light of understanding. Each footstep is a verse in the poetry of existence; each breath is a note in the universe's symphony. Love lingers in the silent embrace of the cosmos, waiting to be found anew.
Original Ciphertext:	[[39, 209, 119, 98], [40, 33, 61, 230], [47, 68, 177, 219], [149, 46, 54, 253], [161, 169, 216, 187], [46, 207, 181, 190], [218, 166, 217, 64], [32, 102, 113, 226], [226, 107, 8, 79], [92, 229, 205, 175], [252, 142, 140, 31], [145, 242, 43, 144], [5, 189, 22, 166], [159, 198, 14, 57], [165, 226, 122, 161], [59, 198, 101, 82], [244, 215, 34, 186], [77, 7, 158, 107], [70, 62, 241, 16], [27, 247, 91, 148], [38, 39, 208, 166], [124, 64, 224, 56], [168, 232, 12, 137], [244, 19, 187, 222], [192, 164, 113, 181], [65, 48, 144, 121], [242, 111, 27, 103], [77, 16, 96, 109], [106, 157, 213, 151], [239, 6, 115, 29], [226, 130, 15, 204], [6, 216, 250, 123], [24, 43, 172, 17], [71, 90, 46, 101], [251, 214, 42, 192], [175, 127, 207, 69], [149, 57, 97, 133], [133, 37, 144, 30], [173, 47, 109, 23], [47, 68, 177, 219], [100, 217, 166, 210], [206, 72, 164, 35], [13, 100, 249, 33], [179, 112, 68, 170], [63, 190, 151, 224], [111, 52, 145, 140], [60, 134, 144, 254], [242, 15, 119, 124], [99, 92, 69, 207], [18, 236, 116, 1], [173, 47, 109, 23], [47, 68, 177, 219], [27, 219, 172, 105], [250, 160, 42, 101], [71, 233, 95, 61], [47, 68, 177, 219], [126, 240, 237, 68], [133, 37, 144, 30], [83, 62, 191, 245], [27, 247, 91, 148], [164, 141, 30, 162], [1, 250, 2, 141], [150, 225, 127, 66], [163, 145, 82, 245], [175, 174, 218, 191], [248, 248, 92, 9], [146, 242, 39, 215], [229, 112, 204, 121], [238, 30, 215, 214], [126, 152, 218, 243], [226, 238, 173, 6], [7, 156, 65, 150], [246, 88, 186, 107], [61, 112, 135, 57], [15, 181, 101, 247], [126, 135, 17, 52], [252, 34, 156, 167], [35, 186, 22, 191]]
Mutated Ciphertext:	[[39, 209, 119, 98], [40, 32, 60, 231], [46, 69, 177, 218], [148, 46, 54, 253], [161, 169, 217, 187], [47, 206, 181, 191], [219, 167, 217, 64], [32, 103, 112, 226], [227, 106, 8, 79], [92, 228, 204, 174], [253, 142, 141, 31], [145, 242, 43, 144], [4, 189, 22, 167], [159, 199, 15, 57], [164, 226, 123, 160], [58, 199, 101, 83], [244, 214, 35, 186], [77, 7, 159, 106], [71, 62, 240, 16], [27, 247, 90, 148], [38, 39, 209, 166], [125, 64, 225, 57], [169, 233, 13, 136], [245, 18, 186, 223], [193, 164, 113, 181], [64, 49, 144, 121], [242, 111, 27, 103], [76, 16, 96, 109], [107, 156, 212, 151], [238, 6, 115, 29], [227, 130, 14, 205], [6, 217, 250, 122], [25, 42, 172, 16], [70, 91, 46, 100], [250, 214, 43, 193], [175, 126, 206, 69], [148, 56, 96, 132], [133, 36, 144, 30], [173, 47, 109, 23], [46, 68, 176, 219], [101, 216, 166, 210], [207, 73, 165, 34], [12, 100, 249, 32], [178, 113, 69, 171], [62, 190, 151, 225], [111, 52, 144, 140], [61, 135, 144, 255], [243, 14, 119, 124], [98, 92, 69, 207], [18, 237, 116, 0], [172, 47, 108, 22], [47, 69, 177, 219], [27, 218, 173, 104], [251, 161, 42, 101], [71, 232, 94, 61], [47, 68, 176, 218], [126, 241, 236, 68], [132, 37, 145, 30], [83, 62, 191, 244], [27, 247, 90, 149], [164, 140, 31, 163], [0, 251, 3, 140], [150, 224, 127, 67], [162, 144, 82, 244], [174, 175, 219, 191], [249, 249, 92, 8], [146, 242, 39, 214], [228, 113, 204, 120], [238, 31, 214, 215], [126, 152, 218, 242], [227, 239, 172, 7], [6, 157, 64, 150], [246, 89, 186, 107], [60, 113, 135, 56], [15, 180, 100, 246], [126, 134, 16, 53], [253, 35, 157, 166], [34, 187, 22, 191]]

TABLE 9: Performance evaluation of encryption and decryption times for different instances

Instance No.	Genetic optimization-based algorithm		Particle warm optimization-based algorithm	
	Encryption time	Decryption time	Encryption time	Decryption time
1	0.00010895729064941406	0.00018334388732910156	0.00015895729064941406	0.00025334388732910156
2	0.00022983551025390625	0.0003142356872558594	0.00028983551025390625	0.0003742356872558594
3	0.00018024444580078125	0.0002307891845703125	0.00023024444580078125	0.0002907891845703125
4	0.0002686977386474609	0.0004341602325439453	0.0003186977386474609	0.0004941602325439453
5	0.0002577304840087891	0.00046706199645996094	0.0003077304840087891	0.00052706199645996094
6	0.0002918243408203125	0.0005147457122802734	0.0003418243408203125	0.0005747457122802734
Key generation time	0.4535059928894043 s		0.5535059928894043 s	

and confidentiality are critical. By successfully encrypting, decrypting, and mutating long messages, the technique establishes itself as a versatile solution for modern cryptographic challenges.

Table 9 presents a comparative analysis of the encryption and decryption times for different instances using a genetic optimization-based algorithm and a particle swarm optimization-based algorithm. The results indicate that the particle swarm optimization-based algorithm consistently exhibits slightly higher encryption and decryption times across all instances. For instance, in the first case, the encryption time for the genetic optimization-based approach is 0.000108957 s, whereas for the particle swarm optimization-based approach, it increases to 0.000158957 s. Similarly, the decryption time follows the same trend, with 0.000183343 s for the genetic optimization-based algorithm and 0.000253343 s for the particle swarm-based approach. This pattern persists across all instances, with increasing encryption and decryption times as the instance number grows. The key generation time is also reported, with the genetic optimization-based algorithm achieving 0.453506 s, whereas the particle swarm optimization-based algorithm requires 0.553506 s, highlighting a notable computational overhead. These findings suggest that while both optimization techniques enable secure encryption and decryption, the genetic optimization-based method demonstrates slightly superior efficiency in terms of execution time. This difference in computational performance may have implications for real-time security applications, where minimizing encryption and decryption times is crucial for maintaining system responsiveness.

In conclusion, the results presented in the tables and figures provide a comprehensive evaluation of the proposed method's performance in encryption, decryption, and mutation processes. The method demonstrates high accuracy in decryption, robust security enhancements through ciphertext mutations, and scalability across varying message lengths. The effectiveness of the GA in optimizing encryption keys ensures a high degree of security while maintaining computational efficiency. These findings establish the proposed approach as a reliable and practical cryptographic solution suitable for diverse applications requiring secure communication and data protection.

5. CONCLUSION AND FUTURE WORKS

The rapid advancement of quantum computing necessitates the development of robust post-quantum cryptographic solutions to replace traditional cryptographic methods that are vulnerable to quantum attacks. LBC has emerged as a leading candidate due to its resistance to quantum algorithms such as Shor's. However, the practical implementation of LBC is hindered by challenges related to key size, memory requirements, and computational complexity. In this study, we introduced a GA-based optimization framework that enhances the key generation process in lattice-based encryption. The results demonstrate that the proposed GA-driven approach optimizes key structures, reducing computational overhead while maintaining high-security standards. By refining the selection, crossover, and mutation mechanisms, the GA ensures the generation of highly secure and computationally efficient encryption keys.

The findings of this study indicate that integrating GAs into PQC methodologies can significantly improve performance and resource efficiency. The optimized key generation process enhances encryption speed. It reduces memory footprint, making LBC more viable for real-world applications, particularly in IoT and embedded systems with limited computational resources. Moreover, the GA-based approach introduces an additional layer of randomness and robustness, ensuring that cryptographic keys maintain high entropy and security resilience. However, GAs have inherent limitations. As a heuristic approach, GAs do not guarantee global optimality and rely on extensive parameter tuning (e.g., population size, and mutation rates) to achieve desirable results.

For future research, further refinements can be made by exploring hybrid optimization techniques that integrate GAs with other metaheuristic approaches, such as particle swarm optimization or simulated annealing. In addition, implementing deep learning-driven GAs may further enhance the adaptability and performance of the proposed framework. Expanding the scope of experimental evaluations to real-world cryptographic applications, including secure communication protocols and blockchain security, will provide deeper insights into the practicality and scalability of the method. Furthermore, investigating hardware acceleration techniques, such as FPGA or GPU-based implementations, may improve computational efficiency for large-scale cryptographic operations. The proposed GA-optimized LBC framework can be further refined by addressing these avenues to meet the evolving demands of secure communication in the post-quantum era.

REFERENCES

[1] F. Bova, A. Goldfarb and R. G. Melko. "Commercial applications of quantum computing". *EPJ Quantum Technology*, vol. 8, no. 1, p. 2, 2021.

[2] M. R. Khan, K. Upreti, M. I. Alam, H. Khan, S. T. Siddiqui, M. Haque and J. Parashar. "Analysis of elliptic curve cryptography & RSA". *Journal of ICT Standardization*, vol. 11, no. 4, pp. 355-378, 2023.

[3] F. Kappel and A. V. "Kuntsevich. An implementation of Shor's r-algorithm". *Computational Optimization and Applications*, vol. 15, pp. 193-205, 2000.

[4] T. N. A. Al Attar, M. A. Mohammed and R. N. Mohammed. "Exploring post-quantum cryptography: Evaluating algorithm resilience against global quantum threats". *UHD Journal of Science and Technology*, vol. 9, no. 1, pp. 18-28, 2025.

[5] A. Khalid, S. McCarthy, M. O'Neil and W. Liu. "Lattice-based Cryptography for IoT in a Quantum World: Are We Ready?" In: *2019 IEEE 8th International Workshop on Advances in Sensors and Interfaces* (*IWASI*). IEEE, 2019.

[6] North American Numbering Plan. National Institute of Standards and Technology (NIST). Available on: https://www.nationalnanpa.com/. [Last accessed on 2025 Apr 12].

[7] X. Wang, G. Xu and Y. Yu. "Lattice-based cryptography: A survey". *Chinese Annals of Mathematics, Series B*, vol. 44, no. 6, pp. 945-960, 2023.

[8] J. Yao and V. Zimmer. "Cryptography". In: *Building Secure Firmware: Armoring the Foundation of the Platform*. Springer, Cham, pp. 767-823, 2020.

[9] R. Chaudhary, G. S. Aujla, Member, N. Kumar and S. Zeadally. "Lattice-based public key cryptosystem for internet of things environment: Challenges and solutions". *IEEE Internet of Things Journal*, vol. 6, no. 3. pp. 4897-4909, 2018.

[10] M. Yasuda. "A Survey of Solving SVP Algorithms and Recent Strategies for Solving the SVP Challenge". In: *International Symposium on Mathematics, Quantum Theory, and Cryptography: Proceedings of MQC 2019*. Springer, Singapore, 2021.

[11] R. Lindner and C. Peikert. "Better key sizes (and attacks) for LWE-based encryption". In: *Topics in Cryptology-CT-RSA 2011: The Cryptographers' Track at the RSA Conference 2011*, *Proceedings*. Springer, San Francisco, CA, USA, 2011.

[12] S. Farooq, A. Altaf, F. Iqbal, E. B. Thompson, D. L. R. Vargas, I. d. l. T. Díez, and I. Ashraf, "Resilience Optimization of Post-Quantum Cryptography Key Encapsulation Algorithms," *Sensors*, vol. 23, no. 12, p. 5379, 2023.

[13] J. H. Holland. "Genetic algorithms". *Scientific American*, vol. 267, no. 1, pp. 66-73, 1992.

[14] J. H. Holland. "Building blocks, cohort genetic algorithms, and hyperplane-defined functions". *Evolutionary Computation*, vol. 8, no. 4, pp. 373-391, 2000.

[15] S. Farooq, A. Altaf, F. Iqbal, E. B. Thompson, D. L. R. Vargas, I. de la Torre Díez and I. Ashraf. "Resilience optimization of post-quantum cryptography key encapsulation algorithms". *Sensors* (*Basel*), vol. 23, no. 12, p. 5379, 2023.

[16] M. A. Khan, S. Javaid, S. A. H. Mohsan, M. Tanveer and I. Ullah. "Future-proofing security for UAVs with post-quantum cryptography: A review". *IEEE Open Journal of the Communications Society*, vol. 5, pp. 6849-6871, 2024.

[17] J. Xie, W. Zhao, H. Lee, D. B. Roy and X. Zhang. "Hardware circuits and systems design for post-quantum cryptography-A tutorial brief". *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 71, no. 3, pp. 1670-1676, 2024.

[18] C. Näther, D. Herzinger, S. L. Gazdag, J. P. Steghöfer, S. Daum and D. Loebenberger. "Migrating software systems towards post-quantum cryptography-a systematic literature review". *IEEE Access*, vol. 12, pp. 132107-132126, 2024.

[19] A. Aydeger, E. Zeydan, A. K. Yadav, K. T. Hemachandra and M. Liyanage. "Towards a Quantum-resilient Future: Strategies for Transitioning to Post-quantum Cryptography". In: *2024 15th International Conference on Network of the Future* (*NoF*). IEEE, 2024.

[20] H. Nejatollahi, N. Dutt, S. Ray, F. Regazzoni, I. Banerjee and R. Cammarota. "Post-quantum lattice-based cryptography implementations: A survey". *ACM Computing Surveys*, vol. 51, no. 6, pp. 1-41, 2019.

[21] A. Kaushik, L. S. S. Vadlamani, M. M. Hussain, M. Sahay, R. Singh, A. K. Singh, S. Indu, P. Goswami and N. G. V. Kousik. "Post quantum public and private key cryptography optimized for IoT security". *Wireless Personal Communications*, vol. 129, no. 2, pp. 893-909, 2023.

[22] R. Asif. "Post-quantum cryptosystems for Internet-of-Things: A survey on lattice-based algorithms". *IoT*, vol. 2, no. 1, pp. 71-91, 2021.

[23] S. Koteshwara, M. Kumar and P. Pattnaik. "Performance Optimization of Lattice post-Quantum Cryptographic Algorithms on Many-core Processors". In: *2020 IEEE International Symposium on Performance Analysis of Systems and Software* (*ISPASS*). IEEE, 2020.

[24] E. Alkim, L. Ducas, T. Pöppelmann and P. Schwabe. "Post-quantum Key {Exchange-A} New Hope". In: *25th USENIX Security Symposium* (*USENIX Security 16*). 2016.

[25] A. Gagnidze, M. Iavich, and G. Iashvili. "Analysis of post quantum cryptography use in practice". *Bulletin of the Georgian National Academy of Sciences*, vol. 11, no. 2, pp. 29-36, 2017.

[26] Z. Liu, K. K. R. Choo and J. Grossschadl. "Securing edge devices in the post-quantum internet of things using lattice-based cryptography". *IEEE Communications Magazine*, vol. 56, no. 2, pp. 158-162, 2018.

[27] C. A. Roma, C. E. A. Tai and M. A. Hasan. "Energy efficiency analysis of post-quantum cryptographic algorithms". *IEEE Access*, vol. 9, pp. 71295-71317, 2021.

[28] M. Kumar and P. Pattnaik. "Post Quantum Cryptography (pqc)-an Overview". In: *2020 IEEE High Performance Extreme Computing Conference* (*HPEC*). IEEE, 2020.

[29] M. Balachandran, S. Devanathan, R. Muraleekrishnan annd S. S. Bhagawan. "Optimizing properties of nanoclay-nitrile rubber (NBR) composites using face centred central composite design". *Materials & Design*, vol. 35, pp. 854-862, 2012.