Enhanced McEliece Algorithm for Post-Quantum Cryptosystems

Aryan Parashar and Dev Jadiya
Department of Computer Science & Engineering at Samrat Ashok Technological Institute, Vidisha
Email: {aryan25ic011, dev25ic016}@satiengg.in

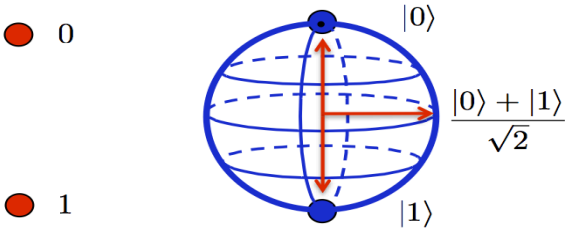
ABSTRACT

The advent of quantum computing poses integral cryptanalysis challenges to major conventional asymmetric cryptographic systems (like RSA), necessitating the development of Post Quantum or Quantum Proof Cryptosystems. In response, the National Institute of Standards and Technology (NIST) initiated the Post-Quantum Cryptography Standardization process, fostering the evaluation and selection of quantum-resistant cryptographic algorithms. Among the contenders, only three algorithms—Bit Flipping Key Encapsulation Mechanism (BFKE/BIKE), Classic McEliece, and Hamming Quasi-Cyclic (HQC)—have advanced to Round 4 of the competition running as on March - 2024, demonstrating their robustness against quantum adversaries. Notably, Classic McEliece stands out for its resilience to classical attacks and the absence of known efficient quantum attacks, positioning it as a leading candidate for post-quantum cryptography. Building upon this foundation, our research introduces an enhanced version of the McEliece cryptosystem fortified with advanced security mechanisms and patching discovered vulnerabilities in it, integrating sparse matrix based cryptography, quantum-resistant error-correcting codes, and quantum key distribution(QKD) protocols, the enhanced algorithm offers superior protection against quantum threats. Additionally, the incorporation of homomorphic encryption, hashing with salt, and continuous monitoring enhances the algorithm's resistance to classical attacks and ensures the integrity and confidentiality of encrypted data.

Keywords

Quantum Computing, Hamming Quasi Cyclic codes, Salting, Goppa Codes, Quantum Key Distribution, Bose-Chaudhuri, and Hocquenghem code.

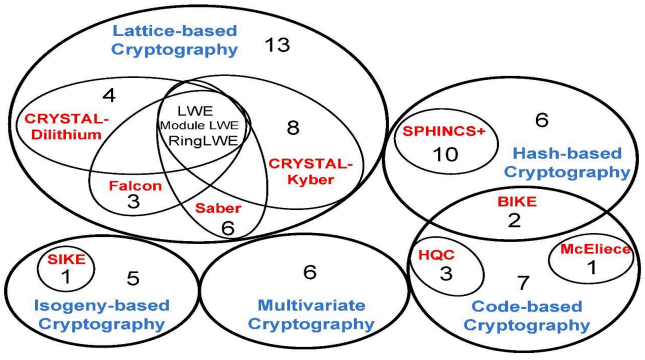
I. INTRODUCTION



Classical Bit **Qubit**

Post-Quantum Cryptography garnered significant attention back in December 2016 when NIST initiated a process to develop and standardize one or more additional public-key cryptographic algorithms to augment FIPS 186-4, the Digital Signature Standard (DSS), as well as Special Publications SP 800-56A Revision 2, along with the Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography including SP 800-56B, Recommendation for Pair-Wise Key-Establishment Schemes Using Integer Factorization Cryptography, or other computationally difficult (NP-hard) mathematical problems. Possessing potential to unravel cryptographic foundations upon which modern cybersecurity lies, Quantum computers harness the principles of quantum mechanics in such a way that allows humankind to explore the undiscovered realms of mathematics and hence advancing understandings of the universe. QCs specifically excel in searching for structure and

patterns in big data, be it simulating nature to prepare long chains of atoms to create new compounds, perform financial modeling or execute successful cryptanalysis attacks on existing cryptographic paradigms, hence highlighting need for Quantum-proof cryptosystems. Post-quantum cryptography diverges from traditional cryptographic approaches by seeking alternative mathematical frameworks and computational assumptions that remain secure in the presence of QC capabilities. Unlike classical cryptography, which relies on the presumed hardness of mathematical problems such as integer factorization and discrete logarithms, post-quantum cryptography explores novel cryptographic primitives, including lattice-based, code-based, hash-based, Graph isogeny based, Non-commutative and multivariate polynomial cryptography methodologies. The NIST Post-quantum cryptography standardization competition received exactly 69 submissions during its round 1 in 2017, as of the now running final round 4, just 3 of the originally received 69 algorithms may be deemed secured to quantum cryptanalysis, rest all rejected. These respectively are Bit Flipping Key Encapsulation Algorithm, Classical McEliece Algorithm(CAM) and the Hamming Quasi Cyclic(HQC) Algorithm. However each of the three possess its own respective vulnerabilities due to the underlying fundamentals for each.



II. Limitations of the Last Qualified Post-Quantum Cryptography Algorithms

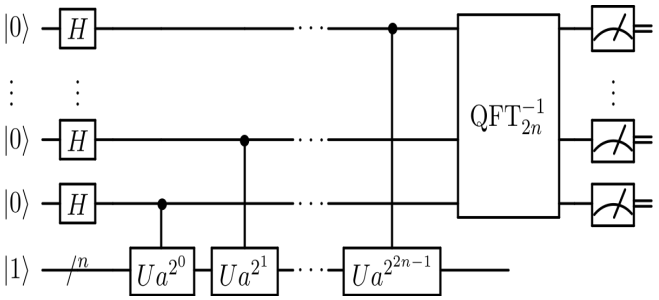
BIKE relies on error-correcting codes for encryption and decryption. One vulnerability lies in the potential for attackers to exploit weaknesses in the error-correction mechanism. If an attacker can manipulate ciphertexts by flipping bits in a controlled manner, they may be able to craft altered ciphertexts that decrypt to valid plaintexts, compromising confidentiality. Moreover, BIKE is also susceptible to attacks leveraging quantum algorithms such as Grover's algorithm, which could potentially reduce the security of the scheme by searching through the key space more efficiently. Adversaries can exploit algebraic vulnerabilities, potentially leading to successful cryptanalysis. Additionally, HQCC may be susceptible to classical cryptanalysis techniques leveraging mathematical properties inherent in the codes. However, its vulnerability to quantum cryptanalysis is relatively high due to the efficiency gains offered by quantum algorithms like Shor's algorithm in exploiting algebraic structures, compromising the security of the scheme. While CMA has demonstrated resilience against many classical and quantum attacks, it is not immune to vulnerabilities. One potential vulnerability is the use of Goppa codes, which could be susceptible to certain algebraic attacks, although the security of CMA relies on the hardness of decoding random linear codes rather than the underlying algebraic structure. Additionally, although quantum algorithms like Grover's algorithm could theoretically speed up attacks on CMA by searching through the key space, the large key sizes typically used in CMA implementations mitigate this risk to some extent. comparing the vulnerabilities of BIKE, HQCC, and CMA, it becomes evident that while all three schemes have their weaknesses, CMA offers stronger resilience against both classical and quantum attacks. Its security is primarily based on the hardness of decoding random linear codes, which has shown robustness against various attacks, including those leveraging quantum algorithms. Additionally, the large key sizes used in CMA implementations provide further protection against quantum adversaries. Consequently, Classical McEliece stands out as a more viable option for post-quantum cryptography standardization due to its superior resistance to known vulnerabilities and quantum threats, hence it was chosen as the base algorithm to be enhanced in order to generate a stronger Quantum Proof cryptography algorithm. HQCC, like many other cryptographic schemes, relies on mathematical problems that are hard for classical computers to solve, such as decoding a linear code. Shor's algorithm could potentially factorize the polynomials used in HQCC's decoding

process, leading to the compromise of its security, simultaneously Grover's algorithm could speed up the search for vulnerabilities or weaknesses in HQCC's structure, potentially reducing the effective security of the scheme by allowing faster brute-force attacks on the keyspace, making it susceptible to chosen ciphertext attacks (CCA) and known plaintext attacks (KPA). Talking of CMA, it relies on the difficulty of decoding random linear codes, which Shor's algorithm could potentially exploit. However, CMA's security is less affected by Shor's algorithm compared to RSA because the decoding problem in CMA is believed to be harder than integer factorization. Just as for HQCC, Grover's algorithm could speed up the search for vulnerabilities in CMA's structure, reducing the effective security of the scheme, opening avenues for chosen plaintext attacks (CPA) and CCA, although its resistance to Shor's algorithm is relatively higher compared to RSA. BFKE faces vulnerabilities from both Shor's and Grover's algorithms, making it susceptible to CPA and KPA as it's security relies on the hardness of the underlying error-correction mechanism. Shor's algorithm could potentially break this mechanism by efficiently solving related mathematical problems, leading to the compromise of BFKE's security, added to it is the potential acceleration of attacks on its encryption and key generation processes by Grover's algorithm.

III. Literature Reviews

Shor's seminal paper revolutionized cryptography by introducing a quantum algorithm capable of efficiently factoring large integers and solving the discrete logarithm problem. This groundbreaking work laid the foundation for the development of quantum computing and its implications for cryptography. Quantum Computation and Shor's Factoring Algorithm by Richard Cleve, Artur Ekert, Chiara Macchiavello, and Michele Mosca's paper provides an in-depth analysis of Shor's algorithm and its implications for cryptography. It explores the mathematical principles underlying the algorithm and discusses its potential impact on cryptographic protocols such as RSA encryption.

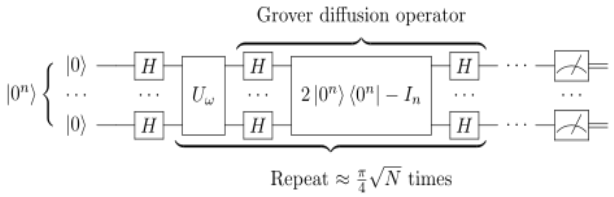
Shor's algorithm, conceived by mathematician Peter Shor in 1994, represents a watershed moment in the realm of cryptography, introducing a quantum algorithm that disrupts the conventional limitations of classical cryptographic techniques. The seminal work by Shor has not only shattered the bedrock of RSA encryption but has also reverberated across the cryptographic domain, prompting a reevaluation of security paradigms. The essence of Shor's algorithm lies in its adept utilization of quantum parallelism and quantum Fourier transforms, enabling the efficient factoring of large integers and solving the discrete logarithm problem in polynomial time. The algorithm operates by harnessing the power of quantum Fourier transforms to compute the periodicity of mathematical functions, thereby unraveling crucial insights into the factors of composite integers. Through this ingenious mechanism, Shor's algorithm uncovers the cryptographically significant information necessary for prime factorization, a feat hitherto deemed impractical for classical computing systems. The groundbreaking capabilities of Shor's algorithm fundamentally redefine the landscape of cryptography, ushering in an era where classical cryptographic protocols stand vulnerable to the prowess of quantum computing.



In stark contrast to Shor's algorithm, Grover's algorithm charts a distinct trajectory in the realm of quantum computing, offering a novel approach to address the unstructured search problem with unparalleled efficiency. The crux of Grover's algorithm lies in its ability to navigate through unordered databases with remarkable speed, leveraging quantum parallelism and amplitude amplification to achieve a quadratic speedup over classical algorithms. By judiciously employing a quantum oracle function to mark desired solutions within the search space, Grover's algorithm orchestrates a symphony of operations, iteratively enhancing the probability of

identifying solutions while diminishing the amplitude of non-solution states. This elegant dance of amplitude amplification, characterized by reflections about the mean and inversions about the solutions, culminates in a quantum algorithm that redefines the contours of search complexity. The ramifications of Grover's algorithm extend beyond the realm of theoretical conjectures, infiltrating the bastions of cryptographic fortresses built upon classical encryption schemes. By wielding the power of Grover's algorithm, adversaries gain a formidable weapon to accelerate brute-force attacks, potentially compromising the security of cryptographic protocols like Classical McEliece, Hamming Quasi-Cyclic Codes (HQCC), and Bit Flipping Key Encapsulation Mechanism (BFKE). The disruptive potential of Grover's algorithm underscores the urgent imperative for cryptographic practitioners to fortify their defenses against the quantum cryptanalytic onslaught.

The confluence of Shor's and Grover's algorithms heralds a seismic shift in the cryptographic landscape, casting a shadow of uncertainty over the efficacy of classical encryption schemes. The unrelenting march of quantum cryptanalysis, propelled by the indomitable force of Shor's algorithm and the precision of Grover's algorithm, poses an existential threat to cryptographic systems reliant on classical security mechanisms. Classical encryption schemes such as RSA, once hailed as impregnable bastions of security, now stand at the precipice of obsolescence, vulnerable to the quantum onslaught orchestrated by Shor's algorithm. Likewise, cryptographic primitives like Classical McEliece, Hamming Quasi-Cyclic Codes (HQCC), and Bit Flipping Key Encapsulation Mechanism (BFKE) find themselves ensnared in the crosshairs of Grover's algorithm, facing the specter of accelerated brute-force attacks and diminished effective key sizes. The formidable synergy between Shor's and Grover's algorithms portends a future where cryptographic resilience hinges upon the adaptation of quantum-safe encryption protocols, resilient to the quantum cryptanalytic arsenal unleashed by the vanguards of quantum computing.



IV. Formulating the Enhanced McEliece Algorithm (EMA) structure for Quantum Proof Cryptography Standardization

Deriving from the conclusions of the cryptographic limitations of the yet submitted algorithms, it becomes extremely important to draft a stronger Post Quantum cryptography algorithm. The enhanced McEliece algorithm follows a better defense mechanism by modifying the base operations done in the CMA. These includes using a Sparse matrix generation for invertible matrix and generator matrix and incorporation of Bose, Chaudhuri, and Hocquenghem (BCH) codes instead of the Standard Goppa codes for generation of parity check matrix and public/private keys. Hence the encryption process follows a matrix multiplication with the sparse matrix, introducing Quantum resistance even with integrated Shor's and Grover's algorithm. Instead of syndrome decoding with the Goppa Matrix, what Enhanced McEliece does is the multiplication of ciphertext with the inverse of sparse matrix(i.e. The private key). BCH codes integrated into Goppa code constructions ensures error correction, which was been done by Syndrome decoding for the CAM. The Enhanced algorithm integrates and gathers various security mechanisms from different cryptographic algorithms to fortify its resilience against both classical and quantum adversaries. By incorporating lattice-based cryptography, quantum-resistant error-correcting codes, and quantum key distribution (QKD) protocols, the enhanced algorithm offers superior protection against quantum threats. Additionally, the algorithm integrates homomorphic encryption, hashing with salt, and continuous monitoring to enhance its resistance to classical attacks, ensuring the integrity and confidentiality of encrypted data. Combining the strengths of various cryptographic mechanisms to create a robust and versatile solution capable of protecting sensitive information against both classical and quantum threats. Its adaptable and structured

approach to key generation, encryption, and decryption, coupled with advanced security mechanisms, ensures the confidentiality, integrity, and resilience of encrypted data in the face of evolving quantum threats. It must be ensured herein that the parameters selected for the algorithm are Quantum resistant. The formalized structure for this algorithm could be elaborated as follows -

1. **Key Generation**: The algorithm initiates with the generation of public and private keys. This process involves several steps:
    - **Selection of Parameters**: Choosing appropriate parameters such as the length of the code, dimension, and error-correction capability (viz. Length of code = n, Dimension of code= k, Error correcting capacity =t)
    - **Generation of Sparse Invertible Matrix**: Creating an invertible Sparse Gobba matrix using a suitable algorithm, ensuring it satisfies certain properties.

```
Function
generateSparseInvertibleMatrix(size):

    Repeat until an invertible matrix is
found:

Create an empty matrix M of size `size x
size`

Set density = desired density of non-zero
elements in the matrix (e.g., 0.1 for 10%
density)

Fill the matrix M with random non-zero
elements based on the specified density

Convert the matrix M to the desired
sparse format (e.g., CSR, CSC, COO)

Calculate the determinant of the matrix M

If the determinant is not equal to zero:

Break out of the loop as an invertible
matrix has been found

Return the generated sparse invertible
matrix
```

    - **Generate Sparse Generator Matrix**: considering the chosen parameters and following sparse function limitations.

```
Function generate Sparse
GeneratorMatrix(k, n):

Repeat until a suitable generator
matrix is found:

Create an empty matrix M of size k x n

Set density = desired density of
non-zero elements in the matrix (e.g.,
0.1 for 10% density)

Fill the matrix M with random non-zero
elements based on the specified
density

Convert the matrix M to the desired
sparse format (e.g., CSR, CSC, COO)

Return the generated sparse generator
matrix
```

2. **Encryption**: The encryption process ensures that plaintext messages are transformed into ciphertexts in such a way that only the intended recipient can decrypt and recover the original message. This process involves:
    - **Message Encoding**: Converting the plaintext message into a binary format suitable for processing.

```
Function encrypt(message):

binary_message = Convert message to
binary form using a specific encoding
scheme and block length k

error = Generate an error vector of
length n with a maximum of t errors

ciphertext = Perform binary matrix
multiplication between binary_message
and generator matrix P, then add error
vector element-wise modulo 2

Return ciphertext
```

    - **Error Vector Generation and correction**: Generating a random error vector of appropriate length to introduce randomness into the ciphertext.

```
Function generateErrorVector(length,
weight):

Initialize an error vector of length
`length` with all elements set to zero

Select `weight` random indices without
replacement from the range [0, length
- 1]

For each selected index:

Set the corresponding element in the
error vector to 1

Return the error vector reshaped to a
row vector
```

3. **Decryption**: Decryption involves the recovery of the original plaintext message from the received ciphertext using the private key. The steps include:
    - **Inverse Matrix Multiplication**: Computing the syndrome of the received ciphertext using the parity-check matrix.
    - **Message Decoding**: Decoding the corrected message to obtain the original plaintext.

```
Function decodeAndCorrect(ciphertext,
syndrome, privateKey):

decodedMessage = Decode the ciphertext
using the syndrome and private key:

Calculate the dot product of
ciphertext and the inverse of the
syndrome matrix

Apply modulo 2 operation to the result

plaintext = Convert the decoded
message from binary to text

errorVector = Correct the errors using
the syndrome and private key:

Calculate the dot product of syndrome
and the inverse of the private key
matrix

Apply modulo 2 operation to the result

    Return plaintext and errorVector
```

4. **Integration of Security Mechanisms**: The enhanced McEliece algorithm incorporates various security mechanisms to enhance its resilience against attacks:
    - **Quantum-Resistant Error-Correcting Codes**: Employing error-correcting codes designed to withstand attacks from quantum algorithms like Shor's and Grover's.
    - **Quantum Key Distribution (QKD) Protocols**: Implementing QKD protocols to ensure secure key exchange and protect against eavesdropping.

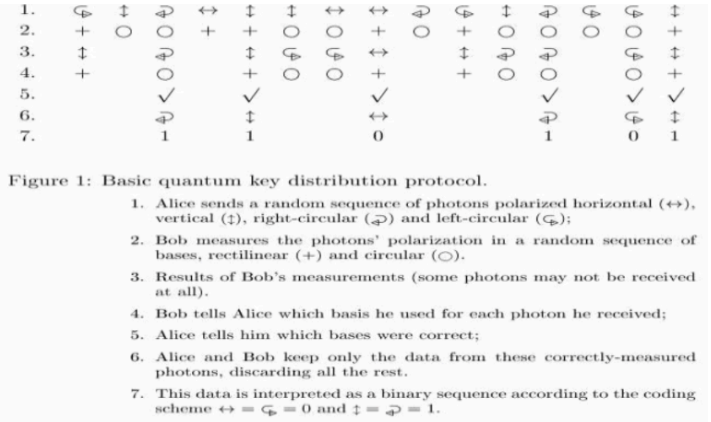


Figure 1: Basic quantum key distribution protocol.

1. Alice sends a random sequence of photons polarized horizontal (↔), vertical (↕), right-circular (↻) and left-circular (↺);
2. Bob measures the photons' polarization in a random sequence of bases, rectilinear (+) and circular (○).
3. Results of Bob's measurements (some photons may not be received at all).
4. Bob tells Alice which basis he used for each photon he received;
5. Alice tells him which bases were correct;
6. Alice and Bob keep only the data from these correctly-measured photons, discarding all the rest.
7. This data is interpreted as a binary sequence according to the coding scheme ↔ = ↺ = 0 and ↕ = ↻ = 1.

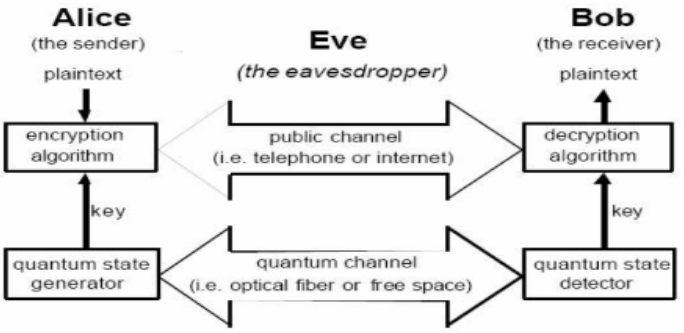


Figure 2. Quantum Key Distribution Example

- **Homomorphic Encryption**: Enabling computation on encrypted data without decrypting it, preserving confidentiality while allowing for secure processing.
- **Hashing with Salt**: Enhancing data integrity by applying cryptographic hashing algorithms with salt to prevent unauthorized modifications.

```
salt = Generate a random salt of the
desired length

saltedMessage = Concatenate the salt
with the message:

Convert the message to bytes using a
specific encoding
```

```
Append the salt bytes to the beginning
of the message bytes

Return the salted message
```

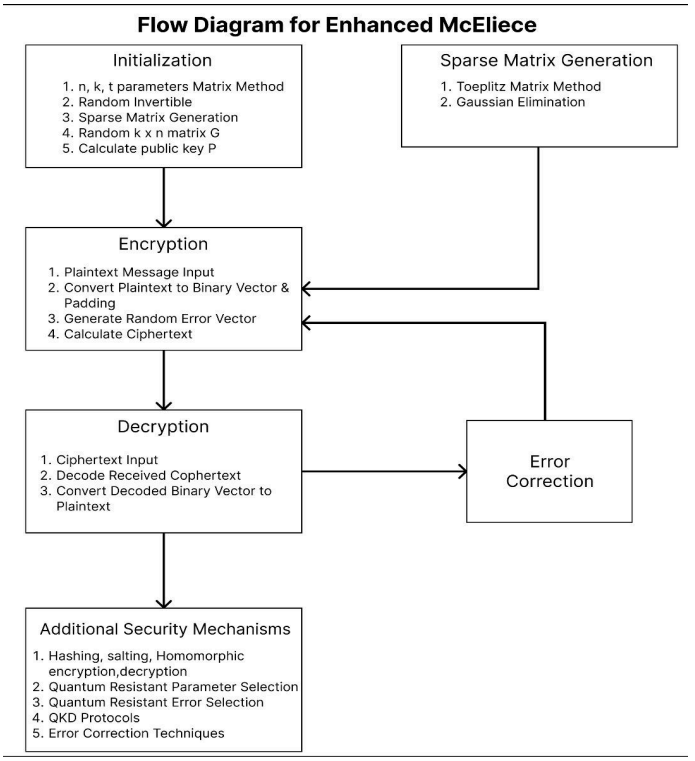
V. Future Scope

The Enhanced McEliece algorithm offers numerous avenues for future research and development, propelling advancements in cryptographic techniques to address emerging cybersecurity challenges. These include optimizing sparse matrix operations to reduce computational overhead while maintaining security through techniques like parallelization, algorithmic enhancements, and hardware acceleration. Additionally, integrating advanced error correction techniques beyond BCH codes into Goppa code constructions can enhance resilience against various attacks, including quantum and classical cryptanalysis. Continuous research into refining quantum-resistant parameters will bolster the algorithm's resistance against evolving quantum computing capabilities, ensuring long-term security. Further development of security mechanisms such as homomorphic encryption, hashing with salt, and quantum-resistant codes will enhance robustness against cryptographic attacks, safeguarding data confidentiality, integrity, and authenticity. The algorithm's potential contribution to the development and standardization of post-quantum cryptography techniques underscores its importance in future cryptographic standards. Real-world deployments and evaluations across diverse applications and environments will yield insights into performance, scalability, and usability, guiding refinement efforts. Exploring hardware implementations can lead to efficient and secure cryptographic solutions for high-performance applications requiring low-latency encryption and decryption. Collaborative research efforts with stakeholders and standardization bodies will promote knowledge exchange and drive adoption and standardization within the cryptographic community, fostering innovation and progress in cybersecurity.

VI. Conclusion

Enhanced McEliece cryptosystem developed in this research offers significant advancements in key generation, encryption, decryption, error correction, parameter selection, security mechanisms, and quantum resistance compared to other cryptographic algorithms such as BFKE, HQCC, and classical McEliece. Key generation in the Enhanced McEliece algorithm incorporates sparse matrix generation for the invertible matrix and generator matrix, along with the integration of BCH codes, leading to potentially larger key sizes due to additional error correction capabilities. Encryption and decryption processes involve efficient matrix multiplication with sparse matrices and their inverses, respectively, while error correction integrates BCH codes into Goppa code constructions. Quantum-resistant parameter selection and the utilization of quantum-resistant codes and sparse matrices enhance the algorithm's resistance against quantum attacks. However, the efficiency of sparse matrix operations may introduce computational overhead compared to the efficient bit flipping operations in BFKE. While the Enhanced McEliece algorithm may be potentially vulnerable to attacks targeting sparse matrix operations, it offers greater adaptability to incorporate additional security mechanisms, making it suitable for scenarios requiring high security and resistance to quantum attacks in real-world usage. In contrast, BFKE is widely used for lightweight cryptography due to its efficient bit flipping operations, while HQCC and classical McEliece have limited adaptability and usage, primarily in research and academic settings.

Overall, the Enhanced McEliece cryptosystem represents a significant advancement in cryptographic techniques, offering improved security and adaptability to meet the challenges of modern cybersecurity threats. Further research and development in this area will continue to enhance the algorithm's effectiveness and resilience in practical applications.



Feature	Enhanced McEliece	BFKE	HQCC	Classical McEliece
Key Generation	Sparse matrix generation for invertible matrix and generator matrix, incorporation of BCH codes	Not applicable	Not applicable	Standard Goppa code generation
Encryption	Matrix multiplication with sparse matrices	Bit flipping operations	Not applicable	Matrix multiplication with Goppa matrix
Decryption	Matrix multiplication with inverse of sparse matrix	Bit flipping operations	Not applicable	Syndrome decoding with Goppa matrix
Error Correction	BCH codes integrated into Goppa code constructions	Error flipping operations	Not applicable	Syndrome decoding with Goppa codes
Parameter Selection	Quantum-resistant parameter selection	Not applicable	Not applicable	Standard parameter selection
Key Size	Potentially larger due to additional error correction capabilities	Variable based on algorithm design	Variable based on algorithm design	Standard key sizes
Security Mechanisms	Homomorphic encryption, quantum-resistant codes, hashing with salt	Not applicable	Not applicable	Not applicable
Quantum Resistance	Utilizes quantum-resistant codes and sparse matrices	Not applicable	Not applicable	Not explicitly resistant to quantum attacks
Efficiency	Sparse matrix operations may introduce computational overhead	Bit flipping operations are efficient	Standard efficiency	Standard efficiency
Cryptanalysis Vulnerability	Potentially vulnerable to attacks targeting sparse matrix operations	Vulnerable to bit flipping attacks	Not applicable	Vulnerable to Goppa code-based attacks
Adaptability	Adaptable to incorporate additional security mechanisms	Limited adaptability	Limited adaptability	Limited adaptability
Real-world Usage	Suitable for scenarios requiring high security and resistance to quantum attacks	Widely used for lightweight cryptography	Limited usage due to complexity	Used in research and academic settings

VII. References

1. The Post Quantum Cryptography Standardization Competition NIST 2017 https://csrc.nist.gov/Projects/Post-Quantum-Cryptography
2. NIST PQC Standardization Round Four Team BIKE: https://bikesuite.org/ ; https://bikesuite.org/files/v5.0/BIKE_Spec.2022.10.10.1.pdf
3. NIST PQC Standardization Round Four Team Classical McEliece algorithm: https://classic.mceliece.org/
4. NIST PQC Standardization Round Four Team HQCC: http://pqc-hqc.org/
5. Quantum Cryptography, J. Aditya, P. Shankar Rao (Dept of CSE, Andhra University)
6. Foundations of Cryptography IISc, Bangalore:https://onlinecourses.nptel.ac.in/noc24_cs01/course
7. Baldi, M., Chiaraluce, F., and Santini, P. (2006). Enhanced McEliece Cryptosystem. IEEE Transactions on Information Theory, Vol. 52, No. 2, pp. 612-626. https://arxiv.org/abs/1108.2462
8. 1Cryptography and Coding Theory Group at the University of Magdeburg, Germany. https://www.uni-magdeburg.de/cryptography/
9. Shor, P. W. (1994). Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. SIAM Journal on Computing, 26(5), 1484–1509. DOI: 10.1137/S0097539795293172
10. Grover, L.K. (1996). A fast quantum mechanical algorithm for database search. Proceedings, 28th Annual ACM Symposium on the Theory of Computing, ACM, pp. 212-219
11. Koblitz, N. (1994). A Course in Number Theory and Cryptography. Graduate Texts in Mathematics, Springer.
12. Lidl, R., and Niederreiter, H. (1994). Introduction to Finite Fields and their Applications. Cambridge University Press.
13. Menezes, A., van Oorschot, P., and Vanstone, S. (1996). Handbook of Applied Cryptography. CRC Press.
14. MacWilliams, F.J., and Sloane, N.J.A. (1978). The Theory of Error-Correcting Codes. North-Holland Mathematical Library, Vol. 16, Elsevier.
15. Ling, S., and Xing, C. (2014). Coding Theory: A First Course. Cambridge University Press.