AI-Driven Innovations in Cryptography: Enhancing Key Generation and Security

Ananda Priya B, Gnanachandra P, and *Seenivasan M*

Centre for Research and Post Graduate Studies, Ayya Nadar Janalki Ammal College, Sivakai-626124
DDE-Mathematics Wings, Annamalai University, Chidambaram, Tamilnadu, India

Abstract. In this paper, we introduce a novel approach for securing confidential data through a symmetric key cryptographic algorithm called the modified Hill Cipher by utilizing rhotrices. We provide a step-by-step procedure to implement this method and elucidate the process through an example. The modified Hill Cipher technique uses AI to generate key rhotrix and incorporates the use of rhotrices and rhotrix algebra to encrypt plain text and decrypt cipher text.

1 Introduction

In network security, one of the most significant challenge is how to safely transmit data across an open network. This can be achieved using cryptography, the process of encrypting information employed to secure data, which are hard to decipher. One of the most well-known classical techniques for encryption is the Hill cipher. In 1929, Lester S. Hill [3] invented this technique, rooted from the concepts of linear algebra, which is a polygraphic substitution cipher. This method works by matrix multiplication and inverse of a matrix. But this method may not provide adequate security as it is susceptible to message tracing by attackers. So, several experts have conducted various studies to enhance the security of the Hill cipher technique.

In [1], Sadeenia used random permutations of rows and columns of a matrix is employed to create a encryption key, resulting in enhanced security for the information. Yi-Shiung Yeh [2], proposed a enhancement in Hill cipher method by transforming matrices and using numerical system with various bases. Adinarayana Reddy K and et.al., [4] modified the hill cipher method by implementing circulant matrices. AbdAllah A. ElHabshy [5], introduced augumented hill cipher which is used with all block ciphers.In [6], Andysah Putera Utama Siahaan used Genetic algorithms offer an efficient approach to finding the key for optimization. In order to secure communication, several researchers modified the hill cipher technique [7].

The purpose of this paper is to provide a more secure Hill cipher cryptosystem that uses rhotrices. We employ rhotrices in our approach because they offer greater convenience than matrices. We provide a technique for creating key rhotrices, encrypting the message following many phases, then decrypting it again using the same steps. In addition, we provide an illustration of how the algorithm functions.

2 Modified Hill Cipher using Rhotrices : Proposed Al- gorithm

In this section, we lay out the algorithm for modified hill cipher technique employing rhotrices. In this approach, there are 3 stages which are outlined below:

Phase 1: Generating Key Rhotrix
Phase 2: Encryption
Phase 3: Decryption

Now, we present the step by step procedure for all these 3 phases.

A. Phase 1: Generating Key Rhotrix using AI

(i) Consider any rhotrix of size 3 with heart of the rhotrix always as 1. This rhotrix will act as an origin key rhotrix, A.

(ii) Since the origin key rhotrix has always 1 in the heart, it is always invertible. The next step is to form a rhotrix of size 3 with various transformation of the origin key A. So, we need to find the inverse, transpose, transpose of the inverted origin key rhotrix and coupled rhotrix of A and fix them in a such a manner to form a key rhotrix of size 7.

$$\left\langle \begin{matrix} & A & \\ A^{-1} & [AC] & [A^{-1}]^T \\ & A^T & \end{matrix} \right\rangle$$

For $\mathcal{A}=\left\langle \begin{matrix} & a_1 & \\ a_2 & 1 & a_3 \\ & a_4 & \end{matrix} \right\rangle$, the key rhotrix

$$\mathcal{K} = \left\langle \begin{matrix} & & & a_1 & & & \\ & & a_2 & 1 & a_3 & & \\ & \frac{-a_1}{e^2} & a_1 & a_4 & a_3 & \frac{-a_1}{e^2} & \\ \frac{-a_2}{e^2} & 1 & \frac{-a_3}{e^2} & 1 & \frac{-a_3}{e^2} & 1 & \frac{-a_2}{e^2} \\ & \frac{-a_4}{e^2} & a_2 & a_1 & a_4 & \frac{-a_4}{e^2} & \\ & & a_3 & 1 & a_2 & & \\ & & & a_4 & & & \end{matrix} \right\rangle$$

(iii) The key rhotrix K is always invertible, since the center part of rhotrix is the coupled rhotrix of origin key A and so the heart of the key rhotrix is heart of the origin key rhotrix which is 1.

B. Phase 2: Encryption

(i) Convert the first 25 characters from the plain text to integer value by using ASCII values and arrange them in a rhotrix P of size 7.

(ii) Transform P by changing each integer value to binary form.

(iii) Execute the logical XOR operations between each entry with heart of the rhotrix.

(iv) After transforming P back to integer form, use the equation $C = K * P \bmod 256$ to encrypt P with key rhotrix K and obtain the cipher text.

(v) Convert each integer value in cipher text C into the binary form and merge all binary data as 200-bit.

(vi) Split the 200-bit binary block to 6-bit binary block and if necessary, add zero padding to the last binary data to finish the 6-bit block.

(vii) Use the Radix64 character to encrypt the final cipher text C.

C. Phase 3: Decryption

(i) Transform each character into an integer value using Radix64 decode cipher text C.

(ii) Change each integer value into the binary form of 6-bit block and merge all binary data into a single 200-bit block.

(iii) Split the 200-bit binary block into 8-bit block and then transform each 8-bit block into an integer value.

(iv) Perform the decryption process between the inverse of key rhotrix K and the cipher text C using the equation $[(K^{-1} \bmod 256)C] \bmod 256$

(v) Convert the temporary plain text P to binary form and perform logical XOR operations between each entry with heart of the rhotrix.

(vi) Convert the 8-bit block to integer value. Using ASCII values, the plain text P is fully recovered.

3 Algorithm Overview: An Example Implementation

In this section, we illustrate how the improved Hill Cipher technique using rhotrices functions with the aid of an prime example. The procedure has 3 phases, as stipulated in preceding section.

(i) **Key Rhotrix Generation:**

Let us randomly take the rhotrix of size 3, $\mathcal{A} =$ $\begin{matrix} & 1 & \\ 1 & 1 & 1 \\ & 1 & \end{matrix}$ as origin rhotrix

The Key rhotrix generated by A, is $\mathcal{K} =$

$$\begin{matrix} & & & 1 & & & \\ & & 1 & 1 & 1 & & \\ & -1 & 1 & 1 & 1 & -1 & \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ & -1 & 1 & 1 & 1 & -1 & \\ & & 1 & 1 & 1 & & \\ & & & 1 & & & \end{matrix}$$

For the purpose of decryption, we need $\mathcal{K}^{-1}=$

$$\begin{matrix} & & & -1 & & & \\ & & -1 & -1 & -1 & & \\ & 1 & -1 & -1 & -1 & 1 & \\ 1 & -1 & 1 & 1 & 1 & -1 & 1 \\ & 1 & -1 & -1 & -1 & 1 & \\ & & -1 & -1 & -1 & & \\ & & & -1 & & & \end{matrix}$$

(ii) **Encryption:**

Plain Text: HELLO
ASCII Values: 72 69 76 76 79
Plain Text in Rhotrix,

$$\mathcal{P} = \begin{matrix} & & & 72 & & & \\ & & 69 & 76 & 76 & & \\ & 79 & 45 & 45 & 45 & 45 & \\ 45 & 45 & 45 & 45 & 45 & 45 & 45 \\ & 45 & 45 & 45 & 45 & 45 & \\ & & 45 & 45 & 45 & & \\ & & & 45 & & & \end{matrix}$$

Convert each integer in the rhotrix with the corresponding binary form of 8-bit block. Then,

$$\mathcal{P} = \begin{matrix} & & & & 01001000 & & & \\ & & & 01000101 & 01001100 & 01001100 & & \\ & & 01001111 & 00101101 & 00101101 & 00101101 & 00101101 & \\ & 00101101 & 00101101 & 00101101 & 00101101 & 00101101 & 00101101 & 00101101 \\ & & 00101101 & 00101101 & 00101101 & 00101101 & 00101101 & \\ & & & 00101101 & 00101101 & 00101101 & & \\ & & & & 00101101 & & & \end{matrix}$$

Now, perform XOR operation between the corresponding entry and the heart of the rhotrix.

$$\mathcal{P} = \begin{matrix} & & & & 01100101 & & \\ & & 01101000 & 01100001 & 01100001 & & \\ & 01100010 & 00000000 & 00000000 & 00000000 & 00000000 & \\ 00000000 & 00000000 & 00000000 & 00101101 & 00000000 & 00000000 & 00000000 \\ & 00000000 & 00000000 & 00000000 & 00000000 & 00000000 & \\ & & 00000000 & 00000000 & 00000000 & & \\ & & & 00000000 & & & \end{matrix}$$

Again, convert each binary number of 8-bit block to integer value.

$$\mathcal{P} = \begin{matrix} & & & 101 & & \\ & 104 & 97 & 97 & & \\ 98 & 0 & 0 & 0 & 0 & \\ 0 \;\; 0 & 0 & 45 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \\ & 0 & 0 & 0 & & \\ & & 0 & & & \end{matrix}$$

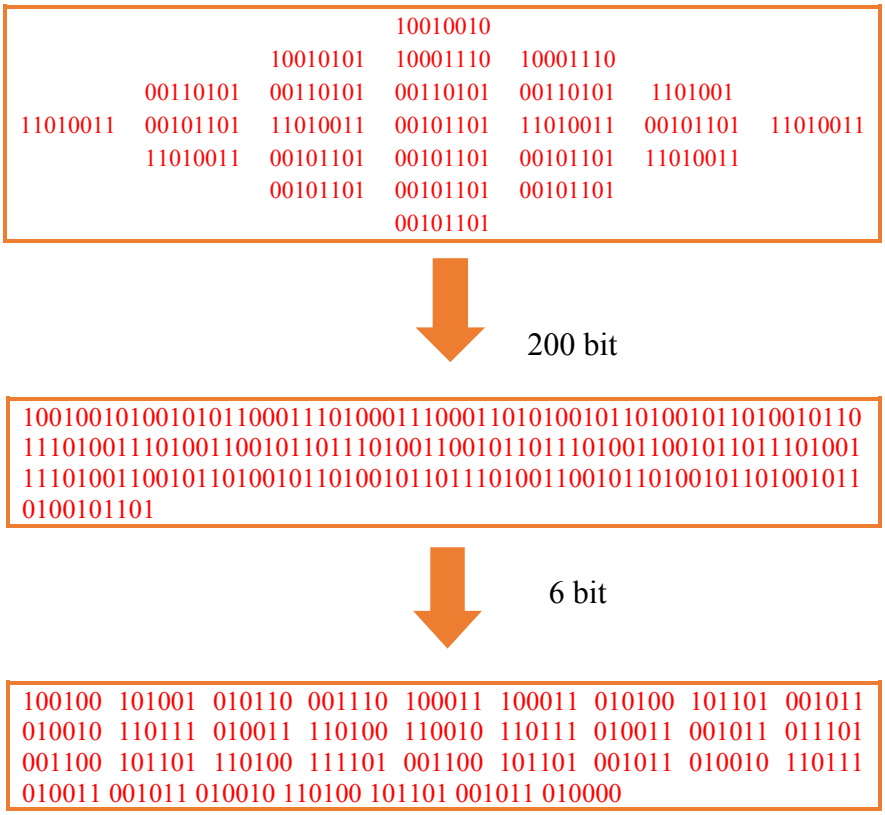
To acquire the cipher text C, perform the process of encryption by multiplying the final P with the generated key rhotrix K. (ie.)C = PK mod 256

$$\mathcal{P}K = \begin{matrix} & & 101 & & & & & & 1 & & \\ & 104 & 97 & 97 & & & & 1 & 1 & 1 & \\ 98 & 0 & 0 & 0 & 0 & & & 1 & 1 & 1 & -1 \\ 0 \;\; 0 & 0 & 45 & 0 & 0 & 0 & -1 \;\; 1 & -1 & 1 & -1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & & & 1 & 1 & 1 & -1 \\ & 0 & 0 & 0 & & & & 1 & 1 & 1 & \\ & & 0 & & & & & & 1 & & \end{matrix}$$

$$\mathcal{C} = \begin{matrix} & & & 146 & & \\ & 149 & 142 & 142 & & \\ 53 & 45 & 45 & 45 & -45 & \\ -45 & 45 & -45 & 45 & -45 & 45 & -45 \\ -45 & 45 & 45 & 45 & -45 & & \\ & 45 & 45 & 45 & & & \\ & & 45 & & & & \end{matrix} \quad \text{mod } 256$$

$$\mathcal{C} = \begin{matrix} & & & 146 & & \\ & 149 & 142 & 142 & & \\ 53 & 45 & 45 & 45 & 211 & \\ 211 & 45 & 211 & 45 & 211 & 45 & 211 \\ 211 & 45 & 45 & 45 & 211 & & \\ & 45 & 45 & 45 & & & \\ & & 45 & & & & \end{matrix}$$

Convert all the integers above into 8 bit binary blocks. Combine all those into single 200 bit block and split them into 6- bit blocks. If there is some remaining binary data which does not form a 6-bit block, then zero padding must be done.



Now to acquire the final cipher text output, use Radix 64 to convert the 6-bit binary data to characters. The Cipher Text C to be transferred is

kpWOjjUtLS3T0y3TLdMt09MtLS3TLS0tLQ

(iii) **Decryption:**

To begin the process of decryption, convert the cipher text to the binary data using RADIX-64. The Cipher Text is:

kpWOjjUtLS3T0y3TLdMt09MtLS3TLS0tLQ

6-Bit Binary Blocks for each character in the cipher text is:

100100 101001 010110 001110 100011 100011 010100 101101 001011 010010
110111 010011 110100 110010 110111 010011 001011 011101 001100 101101
110100 111101 001100 101101 001011 010010 110111 010011 001011 010010
110100 101101 001011 010000

On combining all 6-bit block into single 200-bit block and then splitting them into 8-bit blocks. We get,

10010010 10010101 10001110 10001110 00110101 00101101 00101101 00101101
11010011 11010011 00101101 11010011 00101101 11010011 00101101 11010011
11010011 00101101 00101101 00101101 11010011 00101101 00101101 00101101
00101101

Convert each 8-bit binary data into a integer value and arrange them in a rhotrix.

$$\mathcal{C} = \begin{array}{ccccccc} & & & 146 & & & \\ & & 149 & 142 & 142 & & \\ & 53 & 45 & 45 & 45 & 211 & \\ 211 & 45 & 211 & 45 & 211 & 45 & 211 \\ & 211 & 45 & 45 & 45 & 211 & \\ & & 45 & 45 & 45 & & \\ & & & 45 & & & \end{array}$$

By using the origin key rhotrix A, generate the key rhotrix K. Now, perform $[K^{-1}mod\ 256\ \bullet \mathcal{C}]$mod 256 for the decryption process of the cipher text to obtain the plain text. $\mathcal{P} = [K^{-1}mod\ 256\ \bullet \mathcal{C}]$mod 256

$[K^{-1}mod\ 256\ \bullet \mathcal{C}]$ =

$$\begin{array}{ccccccc} & & & 255 & & & \\ & & 255 & 255 & 255 & & \\ & 1 & 255 & 255 & 255 & 1 & \\ 1 & 255 & 1 & 1 & 1 & 255 & 1 \\ & 1 & 255 & 255 & 255 & 1 & \\ & & 255 & 255 & 255 & & \\ & & & 255 & & & \end{array} \begin{array}{ccccccc} & & & 146 & & & \\ & & 149 & 142 & 142 & & \\ & 53 & 45 & 45 & 45 & 211 & \\ 211 & 45 & 211 & 45 & 211 & 45 & 211 \\ & 211 & 45 & 45 & 45 & 211 & \\ & & 45 & 45 & 45 & & \\ & & & 45 & & & \end{array}$$

$$\mathcal{P} = \begin{array}{ccccccc} & & & 11621 & & & \\ & & 11624 & 11617 & 11617 & & \\ & 98 & 11520 & 11520 & 11520 & 256 & \\ 256 & 11520 & 256 & 45 & 256 & 11520 & 256 \\ & 256 & 11520 & 11520 & 11520 & 256 & \\ & & 11520 & 11520 & 11520 & & \\ & & & 11520 & & & \end{array} \text{ mod } 256$$

$$\mathcal{P} = \begin{array}{ccccccc} & & & 101 & & & \\ & & 104 & 97 & 97 & & \\ & 98 & 0 & 0 & 0 & 0 & \\ 0 & 0 & 0 & 45 & 0 & 0 & 0 \\ & 0 & 0 & 0 & 0 & 0 & \\ & & 0 & 0 & 0 & & \\ & & & 0 & & & \end{array}$$

$$\mathcal{P} = \begin{matrix} & & & 01100101 & & \\ & 01101000 & 01100001 & 01100001 & & \\ 01100010 & 00000000 & 00000000 & 00000000 & 00000000 & \\ 00000000 & 00000000 & 00101101 & 00000000 & 00000000 & 00000000 \\ 00000000 & 00000000 & 00000000 & 00000000 & 00000000 & \\ & 00000000 & 00000000 & 00000000 & & \\ & & 00000000 & & & \end{matrix}$$

$$\mathcal{P} = \begin{matrix} & & & 01001000 & & \\ & 01000101 & 01001100 & 01001100 & & \\ 01001111 & 00101101 & 00101101 & 00101101 & 00101101 & \\ 00101101 & 00101101 & 00101101 & 00101101 & 00101101 & 00101101 \\ 00101101 & 00101101 & 00101101 & 00101101 & 00101101 & \\ & 00101101 & 00101101 & 00101101 & & \\ & & 00101101 & & & \end{matrix}$$

$$\mathcal{P} = \begin{matrix} & & & 72 & & \\ & 69 & 76 & 76 & & \\ 79 & 45 & 45 & 45 & 45 & \\ 45 & 45 & 45 & 45 & 45 & 45 \\ 45 & 45 & 45 & 45 & 45 & \\ & 45 & 45 & 45 & & \\ & & 45 & & & \end{matrix}$$

72 69 76 76 79 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45 45

H E L L O -

4 Comparative Study between the Modified Hill Cipher and Proposed Method

The algorithm proposed in this paper uses rhotrices which have the advantage of allowing a message that has 25 characters to be inputted and which cannot be done with matrices. Computations such as multiplication and inverses are easier and quicker with rhotrices than with matrices. Moreover, generating keys from an origin rhotrix with large numbers can lead to a better level of encryption. To increase security even further, empty cells may be filled with different types of symbols that enhance the encryption and provide a more secure cipher text.

5. Conclusion

The algorithm for modified hill cipher using rhotrices is discussed, which provides an effective method for encrypting and decrypting messages using rhotrices. Also, we have provided an example in this paper illustrating how well the modified Hill Cipher algorithm works to encrypt a message. Moreover, a comparison between the modified hill cipher using matrices and rhotrices has been studied.

References

1. S. Saeednia, How to Make the Hill Cipher Secure, Cryptologia Journal, **24**: 353-360 (2000)
2. YS Yeh, TC Wu, CC Chang, WC Yang, A New Cryptosystem Using Matrix Transformation, 25th IEEE International Carnahan Conference on Security Technology, pp.131-138 (1991)
3. Lester S. Hill, Cryptography in An Algebraic Alphabet, The American Mathematical Monthly, **36:6**, 306-312 (1929)
4. K. Adinarayana Reddy, B. Vishnuvardhan, Madhuviswanatham, A.V.N. Krishna, A Modified Hill Cipher Based on Circulant Matrices, Procedia Technology, **4**, Pages 114-118 (2012)
5. Alhabshy, Abdallah. Augmented Hill Cipher. International Journal of Network Security, **21**, 812-818 (2019)
6. Andysah Putera Utama Siahaan, Genetic Algorithm in Hill Cipher Encryption, American International Journal of Research in Science, Technology, Engineering and Mathematics, **15(1)**, pp. 84-89 (2016)
7. Paragas, Jessie & Sison, Ariel & Medina, Ruji, Hill Cipher Modification: A Simplified Approach, 821-825 (2019)
8. V. U. K. Sastry, & N. R. Shankar, Modified Hill Cipher with Interlacing and Iteration. Journal of Computer Science, **3(11)**, 854–859 (2007)