

Research article

A Modified Hill Cipher Cryptosystem Using Columnar Transposition for end-to-end Communication

Mohammed Ibrahim Anyass¹, Mohammed Muniru Iddrisu², Peter Awon-natemi
Agbedemnab³*

¹Department of Mathematics, University of Technology and Applied Sciences, Ghana.

²Department of Mathematics, Faculty of Physical Sciences, University for Development Studies, P. O. BOX TL 1882, Ghana.

³Department of Information Systems and Technology, University of Technology and Applied sciences, Ghana

*Corresponding Author: Mohammed Ibrahim Anyass. Email: ibrahimmohammedanyass@gmail.com

Received: 12 July 2026; Accepted: 03 September 2026; Published: 14 September 2026

Citation: M. I. Anyass, M. M. Iddrisu, P. A. Agbedemnab, A Modified Hill Cipher Cryptosystem Using Columnar Transposition for end-to-end Communication. Ann. Commun. Math. 9 (2026), 10. <https://doi.org/10.62072/acm.2026.09042>

Abstract: Digital communication among individuals, institutions, and enterprises across internet-based platforms has grown substantially over the past decades. This sharp increase has greatly exposed sensitive information to adversarial attacks and to substantial data losses. Although many cryptosystems have been proposed and implemented to address these challenges, daily reports continue to reveal successful attacks. In this paper, we propose a symmetric-key cryptosystem to improve message security. The proposed system combines the classical Hill cipher with columnar transposition. It uses four distinct keys: two square key matrices and two additional keys termed 'key-within-key' (ω). The Hill cipher is enhanced by expanding the character space to include all printable ASCII characters, mitigating known-plaintext attacks. The key space is doubled by using the transpose of the key matrix as a second key. Encryption is performed in two rounds, each round employing two of the four keys; decryption follows the reverse process. Matlab simulation conducted was observed to display efficient results.

Mathematics Subject Classification: 16U10, 16U80, 16U99, 16S34

Keywords: Cryptosystem, Hill Cipher, Columnar Transposition, Security, Efficiency

1 Introduction

Throughout history, communication of information between individuals has been consistently recognized as valuable. Societies have always developed means of managing information that requires protection, employing various processes to protect it. The scientific discipline concerned with devising and applying methods to protect data or information is known as cryptography. Within this framework, an intended message is modified using predefined algorithms or procedures into a format that is readable only by a specific subset of communicating parties—referred to as legitimate users[1] Myriad of mechanisms have since been developed to convert legible messages into unintelligible ones. Caesar Cipher, transposition cipher,



affine cipher, and others are a few of these mechanisms. To promote effective and efficient communication, communication and cryptography became interconnected. The twentieth century saw an increase in the importance of cryptography, particularly during World Wars I and II, because messages could be sent safely between combat commanders without interception. [2].

Through the advancement of technology, traditional mechanisms to protect information became tractable. Cryptographers have therefore made several attempts to increase data security by strengthening data confidentiality, authenticity, integrity in order to promote efficient communication by introducing new mechanisms. These mechanisms are referred to as cryptosystems. The rise of digital communication has seen a proportional rise in cyber-attacks. It is therefore necessary to implement a system to guarantee the security and privacy of information transmitted over electronic communication channels against unauthorized reception or transmission interception so that digital communication can be protected.

1.1 Some Terms in Cryptography

- Plaintext: This is the plain/readable message that the sender intends to send to the receiver. Plaintext can be a text, audio or video.
- Ciphertext: This is the unintelligible/non-readable form that the message is transformed into using a cipher. A ciphertext is also known as a cryptogram.
- Encryption: This refers to the process of converting a data/message from a readable format to a scrambled piece of information.
- Decryption: This is the process of transforming a ciphertext into a plaintext. It is the reverse of encryption.
- Cryptographic Key: This consists of numerals, alphabets or alpha numeric characters. It may also contain special symbols which are used to encrypt and decrypt data.
- Cryptographic Algorithm: These are sequences of processes, or rules, used to encrypt or decrypt messages in a cryptosystem.

1.2 Types of Cryptosystems

A cryptosystem uses a key for both encryption and decryption. The same key may be used for both operations, or different keys may be employed. Based on this key structure, cryptosystems fall into two categories: symmetric-key cryptosystems and asymmetric-key cryptosystems.

- A symmetric-key cryptosystem employs a single key for both encryption and decryption. Both communicating parties must share this cryptographic key in advance. The sender uses the key and an encryption algorithm to transform the plaintext into ciphertext, while the receiver applies the same shared key along with a decryption algorithm to recover the original plaintext.
- An asymmetric-key cryptosystem, also known as public-key cryptography, utilizes two distinct keys: a public key and a private key. As the name implies, the public key can be freely shared with anyone and requires no confidentiality. In contrast, the private key must remain secret and is never disclosed. Encryption is performed using the public key, while decryption can only be accomplished with the corresponding private key.

Three sets—the key space $\mathcal{K}$, the message space $\mathcal{M}$, and the ciphertext space $\mathcal{C}$ —are connected to a cryptosystem. The two algorithms below exhibit the input/output traits:

- $\text{Enc}(k, m) \rightarrow c$: On input a secret key $k \in \mathcal{K}$ and a message $m \in \mathcal{M}$, it outputs a ciphertext $c \in \mathcal{C}$.
- $\text{Dec}(k, c) \rightarrow m$: On input a secret key $k \in \mathcal{K}$ and a ciphertext $c \in \mathcal{C}$, it outputs a message $m \in \mathcal{M}$.

The figures below represent the symmetric and asymmetric cryptosystems respectively.

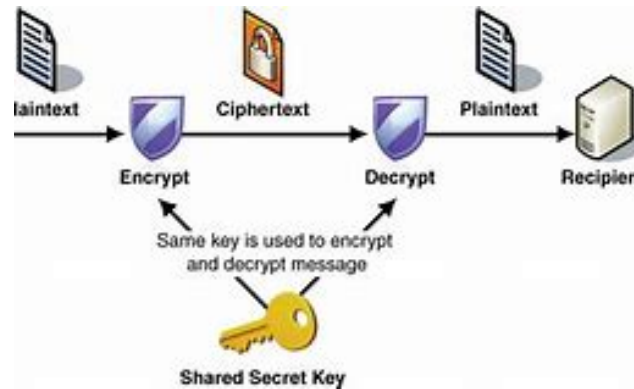


Figure 1: Symmetric-key Encryption and Decryption Process

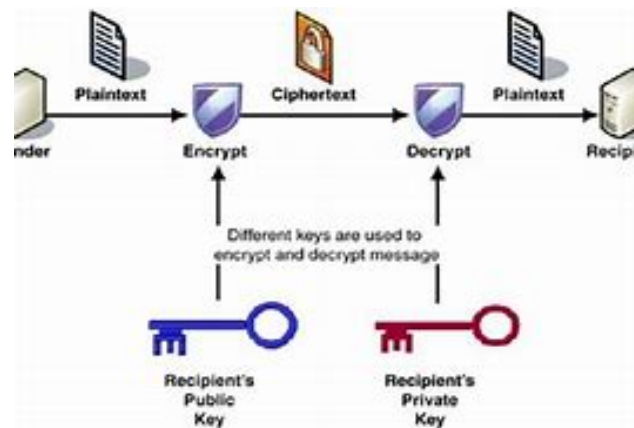


Figure 2: Asymmetric-key Encryption and Decryption Process

1.3 The Hill Cipher

The Hill Cipher is a classical polyalphabetic encryption technique introduced in 1929 by the mathematician Lester S. Hill. It is recognized as one of the first cryptosystems to incorporate concepts from abstract algebra into encryption processes. The algorithm utilizes an invertible $n \times n$ square matrix as its secret key, with all computations performed in arithmetic modulo 26 ($\mathbb{Z}_{26}$). In the encryption process, plaintext characters are first transformed into their numerical representations, ranging from 0 to 25, after which matrix multiplication is applied to produce the ciphertext. Although the Hill Cipher is computationally efficient and mathematically elegant, its inherent linearity renders it susceptible to cryptanalytic techniques, particularly known-plaintext attacks. This vulnerability has motivated extensive research aimed at enhancing the security of the cipher through various modifications and hybrid approaches designed to improve ciphertext robustness and protect the key matrix from cryptographic attacks. [3].

1.4 Columnar Transposition

A columnar transposition cipher is a type of transposition cipher that encrypts plaintext by arranging the message into rows of fixed length determined by the size of a chosen key. Once the plaintext has been organized into the matrix, the ciphertext is generated by reading the characters column by column. The sequence in which the columns are read is governed by the ordering of the key, thus providing the basis for the transposition process.

Matrices remain one of the first ever mathematical implementations in cryptography. Many modifications have been proposed to improve message security, cryptosystem robustness, avalanche effect, and to reduce computational time.

[13] proposed algorithm uses cyclic square matrices for encryption and decryption and also calculates the space complexity of the encrypted and decrypted data. The cyclic square matrix offers the advantage of accommodating a high density of characters within words. However, the absence of a comparative analysis limits the ability to assess the algorithm's relative performance against existing methods.

[5] Conducted a study to examine how matrices are used in cryptography. The orthogonal matrix implementing reflection was found, and essential matrices from the plane's equation were taken into account. The practical difficulty in locating the key matrix increased security. Encryption was made easier by using orthogonal matrix transformation properties. However, the system was not explained in terms of time and spatial complexity. Only alphabetic coding was allowed in the encoding system.

In [8] An encryption technique was developed by integrating a transpose matrix with the AES cryptosystem. The implementation aims to enhance confidentiality by varying the ciphertext for text-based security. When the proposed algorithm was tested, an increase in the avalanche effect was observed. However, this improvement came at the cost of increased computational time.

Furthermore, [11] proposed a new algorithm rule for digital encoding called Matrix Rotation and Conversion

Technique (MRBC). The proposed algorithm reduced the data size and form. It exhibited fast and also safer encryption process. MRBC was carried out in two stages. In stage one, an eight to seven (8-7) byte technique was implemented to convert each eight bytes of text into seven bytes thereby reducing size of data. In the second stage, matrix rotation technique was done to shift and reverse data and that included both row and column transposition.

In another research, a matrix was used to construct an encryption and decryption technique for use with android devices. The 3D character matrix is replaced in the plan with a 2D encryption value matrix. The algorithm was especially made for android devices. To achieve dynamic encryption values, the matrix was used to modify the position of the message characters. When compared to other current techniques, it was discovered that the scheme used less CPU for encryption and even less for decryption [15].

In another work, there was an introduction of Radix64 for encoding and decoding of data along with a simplified approach of block-level data encryption using the logical XOR and shift operations. The results showed that, based on the real experiment conducted, the security of the encryption increased to 53% avalanche effect percentage [10]

Also, [14] suggested a secure modification to the Hill cipher algorithm. Their suggested approach employed a secret prime circulant matrix key and a public key matrix so that the coefficient matrix determinant was zero. Only the inverse matrix multiplication fused with the original Hill cipher was performed in their algorithm.

Again, [6] presented a unique encryption matrix construction approach to address the issue of problematic encryption matrix construction. A stochastic sequence is first produced using the Gauss-Markov technique. Then, using the tensor creation method, multiply the selected matrices with determinant $D=1$ by the elements of the sequence that were converted into integers. As a result, we can build various high order encryption matrices and apply them in practical settings. The suggested technique can create encryption matrices with high security, good random properties, and autocorrelation to fulfill the requirements of cryptography, according to the encryption example and theoretical analysis.

[4] also proposed a rectangular key matrix to be used for encryption through playfair cipher algorithm. The playfair cipher algorithm made the recall of the rectangular key matrix very easy and also provided safe key delivery when shared with other users. The use of rectangular key matrix instead of the usual square matrix presented a much more challenging linear equation to solve and as such presented a much longer and intensely scrambled ciphertext.

In the work of [9] a Myszkowski transposition was added to the Hill cipher algorithm in a study to create an improved system. This was done in order to help the researchers resolve the known plaintext attack problem the Hill cipher method housed. To gauge the performance of the suggested algorithm,

performance evaluation was done. The effectiveness of the strategy was demonstrated by avalanche effect results and runtime.

[12] proposed a hybrid cryptosystem which employed the classical Caesar cipher and columnar transposition to encrypt messages. In their proposed system, the researchers expanded the Caesar cipher which was formally only limited to the alphabetic letters to now comprise the ASCII characters of numbers, letters and symbols. A key matrix trace value limited to modulo 94 whose entries are driven from recursive random number generation equation is used to generate the key for the Caesar cipher. The outcome of the matrix depends on the seed number. The second part of the encryption process uses the columnar transposition with an arbitrary random column selection.

[7] Instead of using a randomly generated matrix, the authors proposed a predictable method for generating a Hill cipher key matrix. Their approach involved creating a magic rectangle of order $m \times n$ ($MR_{m \times n}$), from which a key sub-matrix would be derived. The $MR_{m \times n}$ is then converted into a square matrix of order $m \times m$ ($SM_{m \times m}$). A key of order k is accepted as input, and depending on the encoding method employed, the modulus value is extracted from each element of the square matrix. The proposed algorithm generates the key matrix either with or without row and column overlap..

In [16]A cryptosystem was proposed to secure computer-based examination data. The authors combined the Hill cipher algorithm with the Caesar cipher algorithm to strengthen data security. A random seed value is selected and used to encrypt the data with the Caesar cipher encryption algorithm. The ciphertext resulting from this first encryption then serves as the plaintext for the second encryption process. The key matrix for the Hill cipher encryption is derived from the same seed value using a simple arithmetic calculation.

2 Materials and methods

2.1 Proposed Cryptosystem

The proposed cryptosystem is a symmetric-key cryptosystem that implements the same set of cipher keys for both encryption and decryption. The proposed scheme uses an improved Hill Cipher and columnar transposition in the encryption and decryption process.

2.2 Proposed Cryptosystem Encryption Algorithm

Round One of Encryption Algorithm

- Pick an invertible key matrix $n \times n$ say $k_{(i,j)}$ over Z_m where m is the modulus, i and j represent the row and column, respectively.

$$k = \begin{bmatrix} k_{11} & k_{12} & \dots & k_{1n} \\ k_{21} & k_{22} & \dots & k_{2n} \\ \dots & \dots & \dots & \dots \\ k_{n1} & k_{n2} & \dots & k_{nn} \end{bmatrix}$$

- Dissociate the plaintext (P) into blocks depending on the key matrix size, i.e. (2 x 2), (3 x 3), etc. For $P = [p_1, p_2, \dots, p_n] \in P$
- Represent the plaintext characters with their corresponding numerical vector values as in the character value table.
- Multiply the invertible key matrix $K = (k_{i,j})$ by the numerical vector of consecutive plaintext characters modulo 95. The result is a column matrix of integer values, which is then converted to the corresponding characters to produce the ciphertext C_1 .

$$C_1 = [k_{(i,j)}][P]$$

$$C_1 = \begin{bmatrix} k_{11} & k_{12} & \dots & k_{1n} \\ k_{21} & k_{22} & \dots & k_{2n} \\ \dots & \dots & \dots & \dots \\ k_{n1} & k_{n2} & \dots & k_{nn} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \dots \\ p_n \end{bmatrix} \text{ mod } m$$

Columnar Transposition Cipher Encryption Algorithm

- The key of the columnar transposition is generated from the $n \times n$ key matrix $k_{(i,j)}$ for which reason it is called the key-within-key (ω). ω is generated by taking the main diagonal elements and the anti-diagonal elements and merging them. For a matrix (3 x 3), $\omega = [k_{11}k_{22}k_{33}k_{13}k_{22}k_{31}]$
- The ciphertext (C_1) serves as the plaintext for the columnar transposition cipher.
- Distribute (C_1) which is now the plaintext horizontally, including space, from left to right, and continuously according to the size of ω until no more plaintext is left. A padding is called if the last block of text is incomplete.
- Re-arrange ω with its columns in ascending order and write out the text columnwise.
- The text written from the columnar transposition becomes the second ciphertext (C_2).

Round Two of Encryption Algorithm

- Procedures 1 - 4 of 2.2 are followed using the transpose of the key matrix $k_{(i,j)}$ as the second and new key (k_1) for the improved Hill Cipher (modulo 95) with C_2 as the plaintext to produce the third ciphertext (C_3).

$$C_3 = [k_1][C_2]$$

$$k_1 = \begin{bmatrix} k_{11} & k_{21} & \dots & k_{n1} \\ k_{12} & k_{22} & \dots & k_{n2} \\ \dots & \dots & \dots & \dots \\ k_{1n} & k_{2n} & \dots & k_{nn} \end{bmatrix}$$

$$C_3 = \begin{bmatrix} k_{11} & k_{21} & \dots & k_{n1} \\ k_{12} & k_{22} & \dots & k_{n2} \\ \dots & \dots & \dots & \dots \\ k_{1n} & k_{2n} & \dots & k_{nn} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \dots \\ p_n \end{bmatrix} \bmod m$$

- Again, the ciphertext (C_3) obtained from the immediate previous step is subjected to a second columnar transposition with ω_2 to obtain the final ciphertext called the master ciphertext (C_4), which is sent through the unsafe channel.

2.3 Proposed Cryptosystem Decryption Algorithm

The ciphertext is decrypted back to the plaintext by performing the decryption algorithm twice. First we convert the ciphertext (C_4) into the ciphertext (C_3), from (C_3) to (C_2), from (C_2) to (C_1) and then from (C_1) back to the original plaintext (message). The following equations are considered for the first and second Hill cipher cryptosystem in the decryption process, respectively.

$$C_x = [k_1]^{-1}[C_{x+1}] \bmod 95 \quad (1)$$

$$P = [k]^{-1}[C_1] \bmod 95 \quad (2)$$

Where $[k]^{-1}$ is the multiplicative inverse of the key k . $[k_1]^{-1}$ is the multiplicative inverse of the key k_1 and x is the ciphertext step to be determined. To decrypt a message, the following procedure is followed sequentially.

Round 1 of Decryption Algorithm

- Compute the inverses of the keys k and k_1 .
- For performing columnar transposition on the master ciphertext (C_4) to retrieve the ciphertext (C_3), calculate the length of the column (μ) using the formula;

$$\mu = \frac{\text{length of ciphertext}}{\text{length of key}}$$

- Write out the message in columns and re-arrange according to the order of ω_2 to obtain primary ciphertext (C_3).
- Dissociate the secondary ciphertext (C_3) into blocks of two, three, four consecutive letters depending on the size of the key matrix i.e. (2 x 2), (3 x 3) etc.
- Represent the characters of the ciphertext (C_3) with their corresponding numerical vector values.
- Multiply the inverse key matrix $[k_1]^{-1}$ and the numerical vector matrix of consecutive ciphertext characters modulo 95 using eqn 1. This gives a column matrix of integer values that is transformed into the corresponding characters to obtain the primary ciphertext (C_2).

Round 2 of Decryption Algorithm

Steps 2 through to 6 of the decryption algorithm are repeated with ω for the columnar transposition and the inverse key matrix $[k]^{-1}$ for the Hill Cipher to decrypt the primary ciphertext (C_2) to (C_1) and from (C_1) to the original plaintext (P) using eqn 2.

3 Illustration and Analysis of Proposed Cryptosystem

3.1 Encryption

Suppose that we randomly chose a key matrix $k = \begin{bmatrix} 4 & 7 \\ 3 & 5 \end{bmatrix}$ such that $[k]^{-1}$ exists. To encrypt the plaintext (P) "possible"

$$\omega = [4573]$$

$$k_1 = [k]^T = \begin{bmatrix} 4 & 3 \\ 7 & 5 \end{bmatrix}$$

$$\text{and } \omega_2 = [4537]$$

$$(P, \text{ o}) = \begin{bmatrix} 4 & 7 \\ 3 & 5 \end{bmatrix} \begin{bmatrix} 80 \\ 79 \end{bmatrix} \bmod 95 = \begin{bmatrix} 18 \\ 65 \end{bmatrix} \rightarrow (2, \text{ a})$$

$$(s, \text{ s}) = \begin{bmatrix} 4 & 7 \\ 3 & 5 \end{bmatrix} \begin{bmatrix} 83 \\ 83 \end{bmatrix} \bmod 95 = \begin{bmatrix} 58 \\ 94 \end{bmatrix} \rightarrow (Z, \text{ ~})$$

$$(i, \text{ b}) = \begin{bmatrix} 4 & 7 \\ 3 & 5 \end{bmatrix} \begin{bmatrix} 73 \\ 66 \end{bmatrix} \bmod 95 = \begin{bmatrix} 89 \\ 74 \end{bmatrix} \rightarrow (y, \text{ j})$$

$$(l, \text{ e}) = \begin{bmatrix} 4 & 7 \\ 3 & 5 \end{bmatrix} \begin{bmatrix} 76 \\ 69 \end{bmatrix} \bmod 95 = \begin{bmatrix} 27 \\ 3 \end{bmatrix} \rightarrow (, \text{ #})$$

Primary ciphertext (C_1) = "2aZyj;#"

The first columnar transposition key ω for this stage is [4573]. Performing the columnar transposition

using (C_1) as plaintext produces ciphertext (C_2) as follows.

Ciphertext (C_1)

4	5	7	3
2	a	Z	~
y	j	;	#

Re-arranging the columns in ascending order

Ciphertext (C_2)

3	4	5	7
~	2	a	Z
#	y	j	;

Picking the entries in Ciphertext (C_2) in column major order. The primary ciphertext (C_2) is “~#2yajZ;” This completes round one of the encryption process.

The primary ciphertext (C_2) of the round one encryption algorithm is fed as plaintext into the round two encryption algorithm with keys k_1 and ω_2 .

$$\begin{aligned} (\sim, \#) &= \begin{bmatrix} 4 & 3 \\ 7 & 5 \end{bmatrix} \begin{bmatrix} 94 \\ 3 \end{bmatrix} \bmod 95 = \begin{bmatrix} 5 \\ 8 \end{bmatrix} \rightarrow (\%, \quad) \\ (2, y) &= \begin{bmatrix} 4 & 3 \\ 7 & 5 \end{bmatrix} \begin{bmatrix} 18 \\ 89 \end{bmatrix} \bmod 95 = \begin{bmatrix} 54 \\ 1 \end{bmatrix} \rightarrow (V, \quad!) \\ (a, j) &= \begin{bmatrix} 4 & 3 \\ 7 & 5 \end{bmatrix} \begin{bmatrix} 65 \\ 74 \end{bmatrix} \bmod 95 = \begin{bmatrix} 7 \\ 65 \end{bmatrix} \rightarrow (', \quad a) \\ (Z, ;) &= \begin{bmatrix} 4 & 3 \\ 7 & 5 \end{bmatrix} \begin{bmatrix} 58 \\ 27 \end{bmatrix} \bmod 95 = \begin{bmatrix} 28 \\ 66 \end{bmatrix} \rightarrow (<, \quad b) \end{aligned}$$

The primary ciphertext $C_3 = “\%(V! 'a<b”$. The next operation involves applying columnar transposition on C_3 with ω_2 which is $[4 \ 5 \ 3 \ 7]$

Ciphertext C_3

4	5	3	7
%	(	V	!
'	a	<	b

Re-arranging the columns in ascending order

Ciphertext (C_4)
 3 4 5 7
 V % (!
 < ' a b

The master ciphertext (C_4) = “V<%’(a!b”. The master ciphertext is the final ciphertext that is sent across the insecure channel to the receiver.

3.2 Decryption

$$k = \begin{bmatrix} 4 & 7 \\ 3 & 5 \end{bmatrix} \text{ and } k_1 = [k]^T = \begin{bmatrix} 4 & 3 \\ 7 & 5 \end{bmatrix}$$

computing the inverses $[k]^{-1}$ and $[k_1]^{-1}$ in modulo 95.

$$[k]^{-1} = \begin{bmatrix} 90 & 7 \\ 3 & 91 \end{bmatrix} \text{ and } [k_1]^{-1} = \begin{bmatrix} 90 & 3 \\ 7 & 91 \end{bmatrix}$$

Also, Message length = 8, length of $\omega = 4$ where the message length here represents length of ciphertext (c_4).

$$\mu = \frac{8}{4} = 2$$

Ciphertext (c_4) = “V<%’(a!b”

Writing out (c_4) in columns and rearranging to the original order of ω_2 gives;

Ciphertext (C_4)
 3 4 5 7
 V % (!
 < ' a b

Ciphertext (C_3)
 4 5 3 7
 % (V !
 ' a < b

Re-writing the decrypted ciphertext (C_4) row-wise gives the ciphertext (C_3) “%(V!’a<b”

The dissociation C_3 into two consecutive letters gives (% , (), (V , !), (' , a), and (< , b).

Represent the characters in their respective vector values and perform Hill cipher decryption with the inverse of the key k_1 and C_3 to determine the second ciphertext (C_2) using eqn 2.

Substituting $x = 2$ into eqn 2 gives the equation $C_2 = [k_1]^{-1}[C_3] \bmod 95$.

The primary ciphertext C_2 is retrieved as;

$$(\%, \ () = \begin{bmatrix} 90 & 3 \\ 7 & 91 \end{bmatrix} \begin{bmatrix} 5 \\ 8 \end{bmatrix} \bmod 95 = \begin{bmatrix} 94 \\ 3 \end{bmatrix} \rightarrow (\tilde{}, \ \#)$$

$$(V, \ !) = \begin{bmatrix} 90 & 3 \\ 7 & 91 \end{bmatrix} \begin{bmatrix} 54 \\ 1 \end{bmatrix} \bmod 95 = \begin{bmatrix} 18 \\ 89 \end{bmatrix} \rightarrow (2, \ y)$$

$$(\cdot, a) = \begin{bmatrix} 90 & 3 \\ 7 & 91 \end{bmatrix} \begin{bmatrix} 7 \\ 65 \end{bmatrix} \bmod 95 = \begin{bmatrix} 65 \\ 74 \end{bmatrix} \rightarrow (a, j)$$

$$(<, b) = \begin{bmatrix} 90 & 3 \\ 7 & 91 \end{bmatrix} \begin{bmatrix} 58 \\ 27 \end{bmatrix} \bmod 95 = \begin{bmatrix} 28 \\ 66 \end{bmatrix} \rightarrow (Z, \cdot)$$

The ciphertext $(C_3) = \text{"\%(V! 'a<b"}$ is decrypted back to the primary ciphertext $(C_2) = \text{"\#2yajZ;"}$

Converting ciphertext (C_2) to ciphertext C_1 is done by performing columnar transposition on C_2 with ω .

Writing out C_2 in columns and rearranging to the original order of ω gives;

Ciphertext C_2

```
3  4  5  7
~  2  a  Z
#  y  j  ;
```

Ciphertext (C_1)

```
4  5  7  3
2  a  Z  ~
y  j  ;  #
```

Re-writing the decrypted ciphertext (C_2) row-wise gives the ciphertext $(C_1) = \text{"2aZ~yj;#\text{"}$

Finally, Hill cipher decryption is applied to the ciphertext C_1 to recover the original message, i.e., the plaintext P , utilizing the inverse of the key k in conjunction with C_1 . This process proceeds by dissociating C_1 into consecutive digraphs.

$C_1 = (2, a), (Z, \cdot), (y, j)$ and $(;, \#)$.

Converting each ciphertext letter to its ASCII value and multiplying by $[k]^{-1}$ the inverse of k in modulo 95.

$$(2, a) = \begin{bmatrix} 90 & 7 \\ 3 & 91 \end{bmatrix} \begin{bmatrix} 18 \\ 65 \end{bmatrix} \bmod 95 = \begin{bmatrix} 80 \\ 79 \end{bmatrix} \rightarrow (P, o)$$

$$(Z, \cdot) = \begin{bmatrix} 90 & 7 \\ 3 & 91 \end{bmatrix} \begin{bmatrix} 58 \\ 94 \end{bmatrix} \bmod 95 = \begin{bmatrix} 83 \\ 83 \end{bmatrix} \rightarrow (s, s)$$

$$(y, j) = \begin{bmatrix} 90 & 7 \\ 3 & 91 \end{bmatrix} \begin{bmatrix} 89 \\ 74 \end{bmatrix} \bmod 95 = \begin{bmatrix} 73 \\ 66 \end{bmatrix} \rightarrow (i, b)$$

$$(;, \#) = \begin{bmatrix} 90 & 7 \\ 3 & 91 \end{bmatrix} \begin{bmatrix} 27 \\ 3 \end{bmatrix} \bmod 95 = \begin{bmatrix} 76 \\ 69 \end{bmatrix} \rightarrow (l, e)$$

The ciphertext $C_1 = \text{"2aZ yj;#\text{"}$ is successfully decrypted back to the original message (plaintext) "possible"

3.3 Performance Evaluation of the Proposed Cryptosystem

The proposed cryptosystem has been well evaluated to understand its performance standard. The parameters considered in the evaluation include; Execution time (encryption and decryption time) and security analysis (key sensitivity analysis and key space analysis).

3.3.1 Key Sensitivity Analysis

A good cryptosystem should be highly sensitive to the slightest change in key. This stipulates that a bit change in the cryptographic key should result in a detectable change in the ciphertext generated compared to the original key ciphertext. The proposed scheme produced an appreciable change in ciphertext when tested. The key k in the illustrative example produced the master ciphertext “V<%’(a!b”.

A bit change in the key (from 4 to 3) in the matrix entry reproduced the ciphertext “Da&E(a!b” which is different from the master ciphertext by 50%. Table 1 below represents the keys and the ciphertexts produced by the corresponding plaintexts.

Table 1: key sensitivity analysis results 1

Plaintext	Key	Ciphertext
possible	[4 7;3 5]	V<%’(a!b
possible	[3 7;3 5]	Da&E(a!b
Save the boy	[4 7;3 5]	RE+*A!{]py?6
Save the boy	[4 7;3 0]	iU+BAA*]IZ{6

From Table 1, it can be seen that the ciphertext produced by the original and the slightly altered keys to the plaintext are recognizably different. This informs how sensitive the proposed cryptosystem is. Conversely, in trying to obtain the plaintext, the inverse of key k_1 , when altered just a bit, showed a text other than the plaintext.

Table 2 below represents the inverse key, the slightly altered key and the plaintext produced with the formally generated ciphertexts.

Table 2: key sensitivity analysis results 2

Ciphertext	Inverse Key	Plaintext
V<%’(a!b	[90 7;3 91]	possible
Da&E(a!b	[91 7;3 91]	#oNscb(e
RE+*A!{]py?6	[90 7;3 91]	Save the boy
RE+*A!{]py?6	[90 7;3 93]	SgvH ehS]o<

This shows that, a slightest change in the inverse key generates a text that is completely different from the original plaintext. The same applies to the key-within-key. When there is a bit change, it also greatly influences the outcome of the ciphertext.

3.3.2 Key Space Analysis

The security of a good cryptosystem is again dependent on the key space. Key space refers to the set of all possible keys. In the proposed cryptosystem, multiple keys are used to ensure tight security. The use of multiple keys is directly proportional to the key space. The more numbers of keys used, the larger the key space and the larger the key space, the more resistant the cryptosystem to brute-force attack. For the Hill Cipher, the key space refers to all invertible $d \times d$ matrices over Z_m for an already determined modulo m and is represented by

$$\text{key Space}(d, \mathbb{Z}_m) = \prod_i \left(p_i^{(n_i-1)d^2} \prod_{k=0}^{d-1} (p_i^d - p_i^k) \right)$$

Where p is prime, $n, d \in \mathbb{Z}^+$ [17]

For a 2×2 single key matrix mod 95, computations from the above formula gave the key space of 59,097,600. Doubling the key space because the scheme uses two key matrices gives a key space of 118,195,200 which is an increase in key space. Also, the key-within-key used for a 2×2 key matrix is of length four (4) where each four digit key has $95^4 = 81,450,625$ possibilities. These two keys together provide a possible key combination of $95^4 \times 95^4 = 95^8 = 6,634,204,312,890,625$ for the key-within-key alone which is quite a remarkable figure that can withstand a brute force attack for a longer time. This indicates how large the key space for a 3×3 key matrix modulo 95 would be. Each of the four keys has to be figured out in order to break the cipher which will take a longer time to achieve. The scheme therefore would be able to withstand brute force attack.

3.3.3 Execution Time Analysis

Although numerous parameters influence the performance of a cryptosystem, execution time is second only to security in importance. Execution time—the total duration needed to encrypt or decrypt a message should be the only performance metric traded against security. Simulations performed to compare the proposed cryptosystem with the classical Hill cipher and the combined Hill–Caesar cipher [16] showed that the proposed cryptosystem performed better. The tables below detail the execution times (in seconds) for each encryption scheme.

Table 3: Encryption Execution Time

Input File size(KB)	Encryption Execution Time (Seconds)		
	Hill Cipher	Caesar and Hill Cipher	Proposed Cryptosystem
12.5	0.00015	0.00454	0.00214
15	0.00401	0.67686	6.79127
20	0.02886	0.81310	43.8209

From Table 3 the simulation results showed that the proposed cryptosystem performed competitively well when compared with the other schemes with lower file sizes but fairly well as the file size increases.

This is expected because of the cascade nature of the proposed cryptosystem and the high time can be traded with the higher security provided by the cryptosystem. Table 4 on the other hand illustrates the

Table 4: Decryption Execution Time

Input File size (KB)	Decryption Execution Time (Seconds)		
	Hill Cipher	Caesar and Hill Cipher	Proposed Cryptosystem
12.5	0.401332	0.594168	0.43131
15	0.472326	0.741636	0.46529
20	0.529689	0.833512	0.59792

simulation result from the decryption execution time. It can be concluded from the experiment that the proposed system despite its multiple decryption algorithms performed very well in the decryption process even better than the Caesar with Hill cipher scheme.

The figures below show the graphical representations of encryption and decryption times respectively.

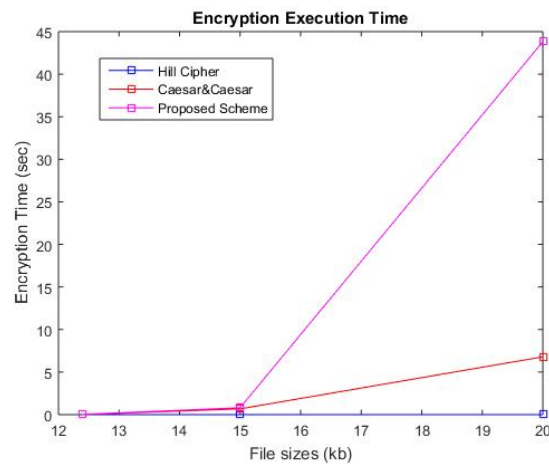


Figure 3: Encryption Execution Time (EET)

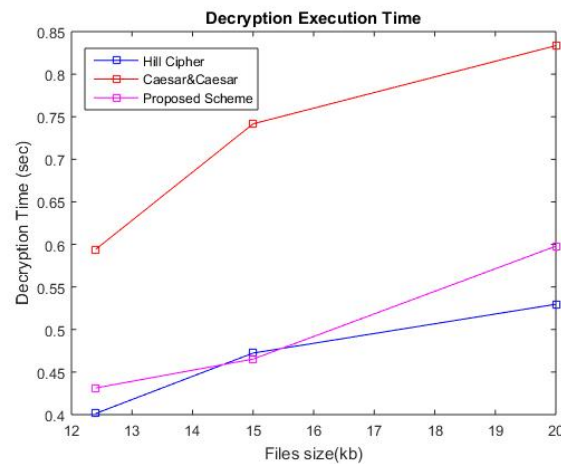


Figure 4: Decryption Execution Time (EET)

4 Conclusion

This paper presented an enhanced variant of the Hill cipher cryptosystem. By integrating the improved Hill cipher with columnar transposition, the proposed scheme significantly strengthens message security for confidential communication. Expanding the traditional character set from 26 English letters to include the printable ASCII set (letters, numbers, and symbols) substantially increases the cryptosystem resistance to brute-force and known-plaintext attacks. Furthermore, the enlarged key space contributes directly to robustness. Collectively, these modifications yield a cryptosystem that not only secures data exchange more effectively than classical approaches but also achieves improved communication efficiency.

Author Contributions: *Mohammed Ibrahim Anyass:* Conceptualization, Formal analysis, Methodology, Writing—Original Draft, Writing—Review and Editing, Visualization, Software, Data curation, Investigation, Results analysis, Resources. *Mohammed Muniru Iddrisu:* Formal analysis, Methodology, Writing—Review and Editing, Visualization, Supervision, Validation. *Peter Awonnatemi Agbedem nab:* Formal analysis, Methodology, Writing—Review and Editing, Visualization, Supervision, Validation. All authors have read and approved the final version of the manuscript for publication.

Acknowledgement: The authors are grateful to the reviewers for all their constructive feedback to refine this work to the appropriate form.

Funding Statement: The author(s) received no specific funding for this study.

Data Availability Statement: Data are available from the corresponding author upon reasonable request.

Ethics Approval: Not applicable.

Use of Generative-AI tools declaration: The authors declare that no Artificial Intelligence (AI) tools were used in the creation of this article.

Conflicts of Interest: The authors have no competing interests to disclose.

References

1. K. M. Martin, *Everyday Cryptography: Fundamental Principles and Applications*, 2nd ed., Oxford University Press, Oxford, UK, **2017**. DOI: <https://doi.org/10.1093/oso/9780198788003.001.0001>
2. O. O. Khalifa, M. R. Islam, S. Khan and M. S. Shebani, Communications cryptography, *2004 RF and Microwave Conference (RFM 2004)*, IEEE, Subang, Selangor, Malaysia, **2004**, 220–223. DOI: <https://doi.org/10.1109/RFM.2004.1411111>
3. R. G. Barrieta, A. S. Canlas, D. M. A. Cortez and K. E. Mata, Modified Hill Cipher Algorithm using Myszowski Transposition to address Known-Plaintext attack, *International Journal for Research in Applied Science & Engineering Technology* **10(4)** (2022), 3242–3249. DOI: <https://doi.org/10.22214/ijraset.2022.41970>
4. T. Alawiyah, A. B. Hikmah, W. Wiguna, M. Kusmira, H. Sutisna and B. K. Simpony, Generation of Rectangular Matrix Key for Hill Cipher Algorithm Using Playfair Cipher, *Journal of Physics: Conference Series* **1641(1)** (2020), 012094. DOI: <https://doi.org/10.1088/1742-6596/1641/1/012094>
5. A. O. Ayo-Aderele, S. O. Edeki, V. O. Udjor and O. Ugbenue, A note on matrix algebra for cryptography settings and applications, *Journal of Physics: Conference Series* **2199(1)** (2022), 012016. DOI: <https://doi.org/10.1088/1742-6596/2199/1/012016>
6. Z. Ajin and X. Xiuchun, Digital Image Encryption Based on Gauss-Markov Random Encryption Matrix, *2011 International Conference on Multimedia Technology*, **2011**, 6311–6313.
7. K. Mani and M. Viswambari, Generation of Key Matrix for Hill Cipher using Magic Rectangle, *Advances in Computational Sciences and Technology* **10(5)** (2017), 1081–1090.
8. E. B. P. Manurung, O. S. Sitompul and S. Suherman, Applying transpose matrix on advanced encryption standard (AES) for database content, *Journal of Physics: Conference Series* **978(1)** (2018), 012073. DOI: <https://doi.org/10.1088/1742-6596/978/1/012073>
9. R. G. Barrieta, A. S. Canlas, D. M. A. Cortez and K. E. Mata, Modified Hill Cipher Algorithm using Myszowski Transposition to address Known-Plaintext attack, *International Journal for Research in Applied Science & Engineering Technology* **10(4)** (2022), 3242–3249. DOI: <https://doi.org/10.22214/ijraset.2022.41970>
10. J. R. Paragas, A. M. Sison and R. P. Medina, An Improved Hill Cipher Algorithm using CBC and Hexadecimal S-Box, *2019 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE)*, IEEE, **2019**, 77–81. DOI: <https://doi.org/10.1109/ECICE47484.2019.8942717>
11. S. K. Reddy and R. S. Kishore, A New Digital Encryption Scheme: Matrix Rotations and Bytes Conversion Encryption Algorithm, *International Journal of Engineering Research & Technology* **3(7)** (2014). DOI: <https://doi.org/10.17577/IJERTV3IS070490>
12. G. Srikantaswamy and H. D. Phaneendra, Improved Caesar Cipher with Random Number Generation Technique and Multistage Encryption, *International Journal on Cryptography and Information Security* **2(4)** (2012), 39–49.
13. K. Thiagarajan, P. Balasubramanian, J. Nagaraj and J. Padmashree, Encryption and decryption algorithm using algebraic matrix approach, *Journal of Physics: Conference Series* **1000(1)** (2018), 012148. DOI: <https://doi.org/10.1088/1742-6596/1000/1/012148>
14. A. R. Vishnuvardhan, V. M. Viswanatham, A. Reddy and K. A. V. N. Krishna, A Modified Hill Cipher Based on Circulant Matrices, *Procedia Technology* **4** (2012), 114–118. DOI: <https://doi.org/10.1016/j.protcy.2012.05.016>
15. P. Baviskar Dinesh, N. Patil Sidhhant and K. Pawar Onkar, Android Based Message Encryption/Decryption Using Matrix, *International Journal of Research in Engineering and Technology* **4(1)** (2015), 315–320. DOI: <https://doi.org/10.15623/ijret.2015.0401047>
16. A. S. I. Yuda, N. Heryana and A. Solehudin, Combination of Hill Cipher Algorithm and Caesar Cipher Algorithm for Exam Data Security, *Buana Information Technology and Computer Sciences (BIT and CS)* **1(2)** (2020), 42–45. DOI: <https://doi.org/10.36805/bit-cs.v1i2.1072>

17. J. Overbey, W. N. Traves and J. Wojdylo, On the keyspace of the Hill cipher, *Cryptologia* **29(1)** (2005), 59–72.
DOI: <https://doi.org/10.1080/0161-110591893771>

Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of Techno Sky Publications and/or the editor(s). Techno Sky Publications and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.