Unimodular matrix and bernoulli map on text encryption algorithm using python

by Puguh Prasetyo

Submission date: 29-Oct-2022 12:35PM (UTC+0700)
Submission ID: 1938500973
File name: and_bernoulli_map_on_text_encryption_algorithm_using_python.pdf (1.27M)
Word count: 3534
Character count: 18270

Al-Jabar: Jurnal Pendidikan Matematika
Vol. 12, No. 2, 2021, Hal 447 – 455
p-ISSN: 2086-5872, e-ISSN: 2540-7562

Unimodular matrix and bernoulli map on text encryption algorithm using python

Samsul Arifin[1]*, Indra Bayu Muktyas[2], Puguh Wahyu Prasetyo[3], Abdul Azis Abdillah[4]

[1] Bina Nusantara University, Indonesia
[2] Sekolah Tinggi Keguruan dan Ilmu Pendidikan Surya, Indonesia
[3] Universitas Ahmad Dahlan, Indonesia
[4] Politeknik Negeri Jakarta, Indonesia

Article Info
Submitted : 29 – 11 – 2021
Revised : 25 – 12 – 2021
Accepted : 26 – 12 – 2021
Published : 27 – 12 – 2021

*Correspondence:
samsul.arifin@binus.edu

Abstract
One of the encryption algorithms is the Hill Cipher. The square key matrix in the Hill Cipher method must have an inverse modulo. The unimodular matrix is one of the few matrices that must have an inverse. A unimodular matrix can be utilized as a key in the encryption process. This research aims to demonstrate that there is another approach to protect text message data. Symmetric cryptography is the sort of encryption utilized. A Bernoulli Map is used to create a unimodular matrix. To begin, the researchers use an identity matrix to generate a unimodular matrix. The Bernoulli Map series of real values in (0,1) is translated to integers between 0 and 255. The numbers are then inserted into the unimodular matrix's top triangular entries. To acquire the full matrix as the key, the researchers utilize Elementary Row Operations. The data is then encrypted using modulo matrix multiplication.

Keywords: Bernoulli Map; Hill cipher; Python; Unimodular Matrix

http://ejournal.radenintan.ac.id/index.php/al-jabar/index

Introduction

A computer network is a collection of computers connected to share resources, communicate, and access data. The goal of a computer network is to fulfill its goals, and any component of the network can request and deliver services. The client is the person who requests and receives the service, whereas the server is the person who delivers and sends the service. A client-server system is an architecture utilized in practically all computer network applications. Securing computer networks or data necessitates the use of appropriate procedures. In cryptography, this is a common strategy. Cryptography may be employed in a variety of ways to safeguard data. The Hill Cipher algorithm is one of the most often utilized methods. One of the implementations of Hill Chipper can be seen in (Nana & Prasetyo, 2021). The Hill Cipher is a symmetric key cryptographic technique with various data encryption benefits. The key matrix is constructed using the new binomial coefficient Newton to prevent a non-invertible key matrix. Plaintext may employ picture or text media and uses the same key for encryption and description. The key to encryption and decryption in the Hill Cipher technique is an m × m sized matrix. Multiplication between matrices and inverses the matrix is the basic matrix theory employed in Hill Cipher. Hill Cipher is a cryptographic algorithm that uses modulo arithmetic. This cryptography approach encrypts and decrypts using a square matrix as the key. When the Hill Cipher algorithm is used to send and receive data, security is assumed to be assured. Thanks to this algorithm, users of data delivery services no longer have to be concerned about attackers attempting to compromise security systems and steal data (Siahaan, 2018).

Text messaging is quite significant in this decade. When text messages are sent from one person to another, protecting them is frequently done at the same time. This necessitates better text message security, both in sending and keeping them. Encrypting text messages is one method. The Hill cipher is a well-known encryption algorithm. The Hill cipher has seen several changes in recent years. However, this approach only works with 3×3 or 4×4 finite key matrices. When the key matrix size exceeds 4, it is said that finding the inverse key is difficult or that finding the reversible key matrix would take a long time. A specific matrix, known as a unimodular matrix, can be used to overcome this problem (Arifin & Muktyas, 2018). Using basic row operations, the researchers can create a unimodular matrix. More information on the unimodular matrix can be accessed (Guo & Yang, 2013). It is unnecessary to utilize the complete matrix as a key. The researchers will create a unimodular matrix using a Bernoulli Map, which means the researchers will need fewer parameters.

Furthermore, using the Python 2.7.14 programming language, the proposed approach will be implemented in many text messages. Some of the reasons the researchers utilize Python are as follows. Python is a fairly simple language to learn, it is free as an open-source, and it is used widely by any programmer in our whole world. Hence, an implementation using Phyton is claimed to have a big impact on the community. This application is also really simple to obtain. Python is available on a variety of operating systems. Python has several applications across various skill sets and disciplines (Arifin et al., 2021; Oliphant, 2007).

Researches on communication security have always been an interesting topic. Recently, online activities cannot be neglected in the new normal era. That is why communication security is still being investigated to better and better. Some researches on communication security can be seen in (Muktyas et al., 2021; Nana & Prasetyo, 2021). In (Muktyas et al., 2021), the security on a message of image form has been developed.

Moreover, this research uses Python to implement a text message encryption technique using a unimodular matrix and a Bernoulli Map, as explained above. In Session 2, the researchers will talk about the research process and review the theory employed, followed by the Python code. Session 3 will cover the implementation of the algorithm and some analysis, and Session 4 will wrap up this work.

The Research Methods

Cryptography is the study of encryption techniques in which plaintext is converted to ciphertext by encrypting it with a secret key. Someone without a decryption key will be unable to solve this document. The ciphertext will be decrypted using an agreed-upon key, and the original data will be returned. In the not-too-distant future, the chances of someone without a decryption key recovering the original text are extremely slim. In classical cryptography, symmetric encryption is utilized, in which the decryption key is the same as the encryption key. Asymmetric encryption techniques are necessary for public-key cryptography when the decryption key is different from the encryption key. Because asymmetric encryption employs very big numbers, encryption, decryption, and key generation for asymmetric encryption techniques need more intense computer than symmetric encryption. Hill Cipher was designed by Lester S. Hill in 1929. This cryptographic approach was developed to build encryption that could not be cracked via frequency analysis. Because Hill Cipher employs matrix multiplication for encryption and decryption, it does not replace identical alphabets in plaintext with the same alphabet in the ciphertext. Because the text to be processed will be separated into blocks of a

specific size, the Hill Encryption, a polyalphabetic cipher, may be classified as a block cipher. In the encryption and decryption process, each character in a block impacts the other characters, ensuring that the same character is not mapped to the same character.

Hill Cipher is one of the traditional cryptographic algorithms that cryptanalysts find extremely difficult to crack if they simply have access to the ciphertext file. However, this approach can be quickly solved if the cryptanalyst possesses a ciphertext file and a portion of the plaintext file. A known-plaintext attack is the name of this cryptanalysis approach. Modulo arithmetic to the matrix is the foundation of the Hill Cipher algorithm. Hill Cipher employs matrix multiplication and inverse matrices methods in its implementation. The matrix n x n, where n is the block size, is the key to Hill Cipher. Because the K-1 matrix is the key required to decode, the K matrix that produces this key must be an invertible matrix with inverse K-1. As a result, the key must have an inverse (Siahaan, 2018; W, 2006).

In the year, Lester Hill found a way to encrypt a plaintext using the Linear Equation System. Before encrypting a plaintext, Hill Cipher divides it into blocks first. The ciphertext is generated by solving a System of Linear Equations consisting of n equations and n variables. The Linear Equation System can be solved using the matrix multiplication concept. Note that the concept of Hill Cipher belongs to asymmetrical cryptography, which means that the resulting key must have an inverse, which can be guaranteed by a unimodular matrix as the key matrix. The phrase "unimodular matrix" or "Nice Matrix" was coined by Hanson (1982). A square matrix A is called a unimodular matrix if it occurs det(A) = -1 or det(A) = 1. Some examples of unimodular machines are the upper triangle matrix and the lower triangle matrix, where all the main diagonal entries are 1 or -1. This statement follows the following theorem (Anton & Rorres, 2013; Muktyas et al., 2021).

(Theorem 1.)(Anton & Rorres, 2013) *Let A is a triangle matrix. Then* $\det(A) = a_{11}a_{22}\dots a_{nn}$.

The following lemma describes the steps needed in generating a unimodular matrix. Note that Lemma 2 below will be the starting point for writing Python program code to build a unimodular matrix (Hanson, 1982; Arifin & Muktyas, 2021).

Lemma 2. (Hanson, 1982) *A unimodular matrix* A_n *can be constructed in the following ways:*

1. *First, make a diagonal matrix with the diagonal entry* $a_{ii} = 1$ *or* $a_{ii} = -1$.
2. *Second, fill in any random integers at each entry with* $i < j$. *Afterward, it has formed a top triangular matrix whose determinants are 1 or -1. This is a unimodular matrix.*
3. *Third, to be a complete matrix, use elementary row operations (ERO) "add a row with multiply of another row."*

Based on Lemma 2 above, the researchers will select the value of a_{11} as 1 in step one, as it will be applied to the positive integer value in this case. The researchers will use the Bernoulli Map function in the second stage to convert it into integers between 0 and 255. In the third step, the researchers are still using the Bernoulli Map function so that the researchers can apply "add one line by multiplying another row." The derivative Bernoulli Map function is expressed as real numbers between 0 and 1. Then, the researchers change from real to an integer by selecting the first three digits after the decimal point and modulating with 256 (Delmi et al., 2020; Rosen, 2019).

In 1976, May was the first to present the logistics map (Devaney, 2018). Despite the fact that one of the chaotic functions looks to be straightforward. The recursive formula is

$$x_{n+1} = rx_n \ mod \ 1.$$

One of the chaotic functions utilized in cryptography applications is the Bernoulli Map. This method creates rows of real numbers, which may then be utilized to create a keystream. The Bernoulli Map method, which was created to discover the keystream plant, was utilized to advance to the next phase by restricting the number of integers raised to eight. This algorithm creates real numbers sequence that must be transformed into integers with a range of 0 to 255 as a keystream. The method begins by luting the key rows generated from Equation (2.1). After these keys are multiplied by 10000 and rounded down (floor) to form an integer row, the key is mapped in a range between 0 and 255 (Maulida, 2018). Bernoulli Maps produce a random series of numbers with a dispersion between 0 and 1 when $r > 1.(0.1)$. There are several methods for constructing a row of integers with values ranging from 0 to 255. One is to modulate the results by 256 using three first digits following the decimal point of x given by the Bernoulli Map. The first three digits after the decimal point were employed in this inquiry. This paper offers four algorithms, each of which uses a unimodular matrix as the key. The matrix is created using Bernoulli Maps. Algorithms 1 and 2 are used in the encryption and decryption process. The four algorithms adopted from (Pareek et al., 2006; Ye & Ma, 2013) with modification can be seen in Table 1.

Table 1. The Four Algorithms The Researchers Propose

Algorithm 1: Generating Bernoulli Map Sequence	Algorithm 1: Generating Unimodular Matrix (Key)
1. x = x0 #x0 as initial value 2. loop to 1000 times first to make sensitive sequence of Bernoulli Map 3. barisan = [] 4. for i in range(n): 5. x = r * x mod 1 # r > 1 6. barisan[i] = x*1000%256 # take 3 first digit after decimal point 7. return barisan	1. K=I_n 2. for i < j: K_ij = random numbers generated by Bernoulli Map 3. Use ERO add a row with multiply of another row modulo m to complete the matrix K.

Algorithm 3: Encryption	Algorithm 3: Decryption
1. Convert input text to a matrix P_(n) 2. Generate a key matrix K_m using Algorithm 2. If n is not divided by m, add more dummy character j to P such that m|(n+j) 3. Reshape the matrix P_(n+j) to a matrix P_(m, (n+j)/m) 4. C_(m, (n+j)/m) = K_m * P_(m, (n+j)/m) mod x 5. Reshape the matrix C_(m, (n+j)/m) to a matrix C_(n+j). 6. Convert the matrix C_(n+j) to encrypted text.	1. Convert input text to a matrix C_(n) 2. Generate a key matrix K_m using Algorithm 2. If n is not divided by m, add more dummy character j to P such that m|(n+j) 3. Find the invers matrix K_m^(-1) using ERO mod x 4. Reshape the matrix C_(n+j) to a matrix C_(m, (n+j)/m) 5. P_(m, (n+j)/m) = K_m^(-1) * C_(m, (n+j)/m) mod x 6. Reshape the matrix P_(m, (n+j)/m) to a matrix P_(n+j). 7. Convert the matrix P_(n+j) to decrypted text.

If the researchers use the process in (Muktyas et al., 2021), one of the disadvantages is the existence of a keyspace for a small password 1, which is based only on the size of the image. As of this writing, password key 1 is expanded. This is still acceptable because the decrypted message reads like the original one. Look at the following example. Suppose the researchers want to encrypt a *P* statement that reads as follows:

kelompok kami berisi samsul arifin dan indra bayu muktyas

In the sentence above, the researchers change first into Barisan or matrix line, which can be seen in Figure 1 below.

```
[107 101 108 111 109 112 111 107  32 107  97 109 105  32  98 101 114 105
 115 105  32 115  97 109 115 117 108  32  97 114 105 102 105 110  32 100
  97 110  32 105 110 100 114  97  32  98  97 121 117  32 109 117 107 116
 121  97 115  32  32  10]
```

Figure 1. The Row Matrix P Corresponding to the Statement P

Note that the length of P is 58 characters. Then the researchers add P with the integer n in such a way up to $5|58 + n$. Note that in this case, the value of n is 2, and 58+2 is 60. As a result the set containing factors of 60 is $\{2, 3, 4, 5, 6, 10, 12, 15, 20, 60, 30\}$. Next, create a key matrix sized according to the user input, for example, 5. This is password 1. As a result, the key matrix built is a $5x5$ unimodular matrix, obtained from algorithm 2. Furthermore, suppose password 2 is 211101, then based on the given 1 and 2 passwords, the generated $5x5$ unimodular matrix can be seen in Figure 2.

```
[[1.0000e+00 1.1400e+02 4.4800e+02 7.4700e+02 9.1500e+02]
 [3.3100e+02 3.7735e+04 3.8859e+04 2.7486e+04 2.8760e+04]
 [2.9200e+02 3.3288e+04 2.0817e+04 5.3616e+04 4.8102e+04]
 [1.3900e+02 1.5846e+04 7.2720e+03 4.8834e+04 1.7782e+04]
 [5.4400e+02 7.0160e+03 2.3712e+04 2.1368e+04 2.7610e+03]]
```

Figure 2. The Unimodular Matrix K_{5x5} (Key Matrix)

Note that the size of the P matrix is adjusted to the K matrix so that it can be multiplied, bringing the size of the P matrix to 5x12. Next is to do the encryption process that multiplies the matrix of $K_{5\times5}$ and $P_{5\times12}$, which produces a matrix C measuring 5×12, and if made into a matrix of rows, will be as follows: in Figure 3.

```
[23111  2615 18303 46451 16636 23994 36134 53987 14849 54983 17230 46872
 32439 51234  9942 12414 17972 48933 54825 52072 25259 53380 52352 43500
 29125 22321 15826 28258  5583 12514 48883  6464  2787 17866 48420  5476
 37375 52699 24222 36643 11393 49518 34977 27199 42698 16839 49171 31299
 32501 47592  1941 24461 30091 17852 22017 54025 47971 45784 23152 33378]
```

Figure 3. Matrix $C_{5\times12}$, as the Product of the Matrix $K_{5\times5}$ and $P_{5\times12}$

The matrix $C_{5\times12}$ above indicates that the encryption process is complete. The next process is to decrypt, which is to multiply between the inverses of matrix K and C. Note that the unimodular matrix used guarantees that every encryption process can be decrypted again because every unimodular matrix must have an inverse.

The Results of the Research and the Discussion

The researchers used Intel® Xeon® CPU E5-2650 v2 @2.60GHz 2.60 GHz, RAM 32GB, Windows 10 64-bit. The proposed algorithm was successfully implemented on text lorem ipsum. The Python program code can be seen in Figure 4.

```python
import numpy as np
from functools import reduce
import time

def factors(n):
    return set(reduce(list.__add__,
                ([i, n//i] for i in range(1, int(n**0.5) + 1) if n % i == 0)))


#OBE
def r_ij(m, baris_i, baris_j, r):
    return m[baris_i] + r*m[baris_j]

#########################

#barisan logistic map
def log(x0, banyak):
    x = x0
    for i in range(1000):
        x = 3.9 * x * (1 - x)     # 3.9 bisa diganti bil. pada [3.7, 4]

    barisan = np.zeros(banyak, dtype=np.uint16)
    for i in range(banyak):
        x = 3.9 * x * (1 - x)     # 3.9 bisa diganti bil. pada [3.7, 4]
        barisan[i] = x*1000%55000
    return barisan
#########################

def kunci(n, x0):
    # matriks segitiga atas
    banyak = int(n * (n - 1)/2)
    barisan = log(x0, banyak + n - 1)


    msa = np.eye(n)

    indeks = 0
    for i in range(n):
        for j in range(i+1,n):
            msa[i,j] = barisan[indeks]
            indeks += 1

```

Figure 4. Python Code Display on Windows OS

Figure 5 contains the text we used as a test was dummy text taken from a famous dummy text that is usually used from lipsum.com site that contained five paragraphs, which is as follows:

```
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer at felis blandit, vulputate odio ut, scelerisque
nisi. Ut rutrum nec eros id porttitor. Nam quis eros dignissim, maximus orci non, molestie urna. Suspendisse
pretium metus a metus finibus, non vehicula nunc sagittis. Pellentesque pulvinar condimentum ante eget tincidunt.
Nam et odio sodales, rhoncus nibh quis, auctor turpis. Sed finibus volutpat velit iaculis faucibus. Curabitur in
venenatis diam, quis porta ante.

Phasellus convallis quam mi, vitae tincidunt purus mollis vitae. Sed sagittis facilisis ipsum at convallis.
Vestibulum ultrices pellentesque libero, eget pretium leo vulputate aliquet. Mauris a ullamcorper odio. Ut sodales
tortor in ultricies mattis. Suspendisse ut interdum velit, ac hendrerit orci. Donec eros arcu, finibus eget
lobortis et, mattis et erat. Aenean eleifend maximus augue, ut iaculis nulla euismod ac. Suspendisse ornare eu
risus at rhoncus. Sed laoreet vehicula lectus sed ullamcorper. Nulla malesuada purus sed tempor finibus.
Pellentesque pretium pulvinar porttitor. Suspendisse consectetur ante at risus commodo vestibulum. Duis rutrum
velit non nisl varius laoreet. Suspendisse euismod sem pharetra massa condimentum placerat.

In hac habitasse platea dictumst. Duis sit amet dictum erat. Proin elementum turpis nec mauris faucibus vehicula.
Praesent eu lectus varius, volutpat sapien eu, placerat nisl. Donec vestibulum congue sapien, in viverra mauris
luctus eget. Fusce eu lacinia diam, ultrices tincidunt sem. Aenean blandit, lorem eu rutrum finibus, eros turpis
luctus nulla, vitae fermentum erat felis id leo.

Nam eleifend tincidunt dignissim. Maecenas sit amet est porta, auctor augue nec, tincidunt est. Nam viverra neque
commodo felis laoreet maximus. Curabitur imperdiet mi blandit nulla vestibulum, sed mattis ligula semper. Etiam at
magna lorem. Sed fermentum leo odio, nec viverra diam feugiat ut. Ut tristique quam sed aliquet interdum. Donec
eget ultricies mauris. Morbi a sem a nisi fringilla sagittis. Pellentesque porttitor lectus rhoncus, porttitor
odio sed, egestas nisl. Cras non lorem ornare, varius leo non, porta nulla. Cras sed magna vitae erat sagittis
rhoncus faucibus ut nisi. Suspendisse eu nisi nec erat feugiat volutpat. Phasellus lobortis ex sed dolor eleifend
semper. Cras elit massa, rhoncus ut tincidunt ac, commodo sed enim.

Morbi eleifend tincidunt libero, et consectetur nunc tempus id. Lorem ipsum dolor sit amet, consectetur adipiscing
elit. Proin nec lacus vel quam mattis rutrum. Curabitur vel convallis augue. Nullam eu ipsum nunc. Aenean sed nisl
sollicitudin, viverra justo vitae, tincidunt lorem. Vivamus dui libero, convallis et placerat eget, cursus vitae
ipsum. In at nisi auctor, efficitur metus vel, accumsan tellus. Suspendisse quis venenatis eros.
```

Figure 5. The Dummy Text

Figure 6 is about the encryption result of the text with key 1: 10 and key 2: 22021985 as follows.

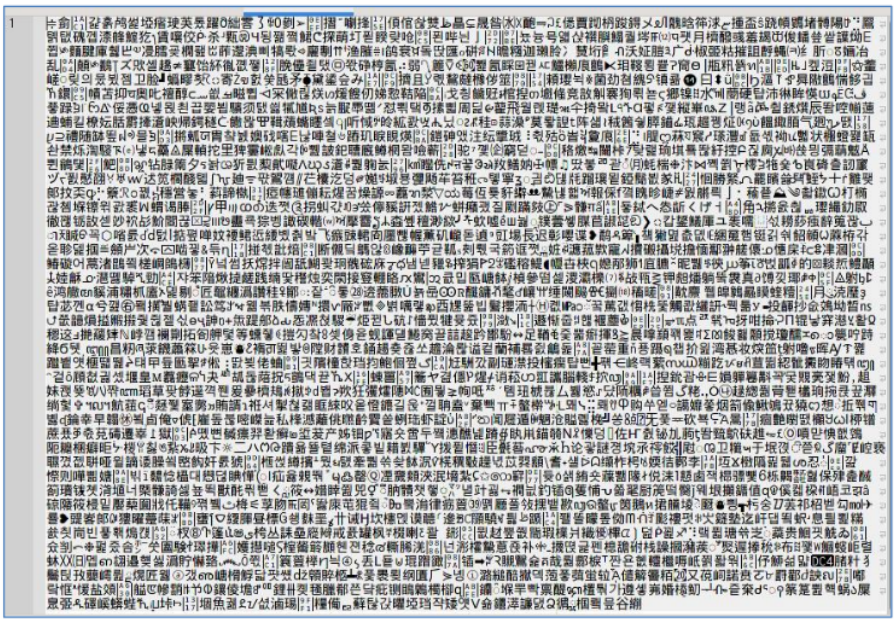

Figure 6. The Encryption Result

The characters are not well-read. It's good for encryption. Then after decrypting, the file is as follows in Figure 7. The decryption process can form back the ciphertext to be the plaintext, and the proposed algorithm guarantees this process

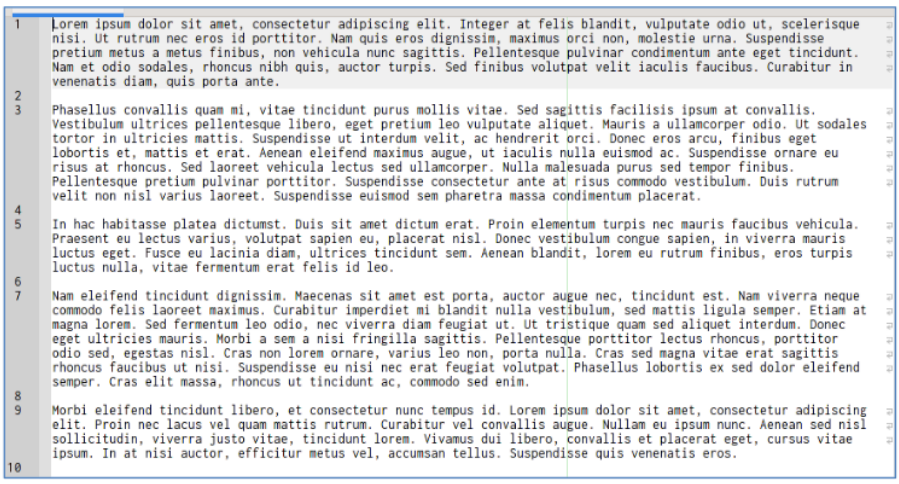

Figure 7. The Decryption Result

The full program code and example can be accessed at the following link: https://github.com/muktyas/text-encryption-chaos-unimodular.

Research relevant to what the researchers do can be found in (Irsan & Antoro, 2019). The objective of this research is to develop a method for encrypting text. By using the MS map as a keystream generator, the technique takes advantage of the chaotic map's characteristics. The proposed algorithm's ciphertext is extremely difficult to crack using brute-force and known-

plaintext attacks. The decryption process has been tested with three secret keys (all of which are extremely similar to the secret key used in the encrypted process). The decrypted text is identical to plaintext only if the secret input key is identical to that used in the encrypted process.

On the other hand, Krasimir Kordov's research is equally pertinent to our situation (Kordov, 2021). In this article, a new encryption technique for secure text message communication is given. The suggested cryptographic method is based on a pseudorandom generator that uses two chaotic maps to generate random numbers. An extensive cryptographic analysis is used to determine the security level. The findings of statistical testing, keyspace analysis, frequency analysis, common correlation analysis, entropy analysis, key sensitivity analysis, and speed performance are included in evaluating the proposed cryptographic system.

Hill Cipher is one of the methods used in cryptography in general. The square key matrix in the Hill Cipher algorithm must have an inverse modulo. The unimodular matrix is one of the unusual matrices that must have an inverse. A unimodular matrix can be utilized as a key in the encryption process. This unimodular matrix notion fills a research void because of its unique character, which is always invertible.

Additionally, the cryptography employed is symmetric cryptography. The approach is demonstrated by utilizing the Bernoulli Map function to generate a unimodular matrix. The researchers show an alternative to safeguarding text data.

Conclusion and Suggestion

The conventional Hill cipher uses a small key matrix $K_n, n = 4$. Furthermore, if $n > 4$, the entire matrix K_n is used as the key using the usual Hill cipher. In this research, we use a Bernoulli Map as a key to building a Unimodular matrix that can solve the problem. K_n, yet only two arguments are required (password 1 and password 2). According to the results of the experiments based on the matrix property, the encrypted text messages tested were difficult to decipher in normal languages.

Acknowledgment

The researchers would like to express their gratitude to the reviewers for their insightful comments, recommendations, and ideas, which have helped shape this article into something that deserves to be published.

References

Anton, H., & Rorres, C. (2013). *Elementary linear algebra: Applications version*. John Wiley & Sons.

Arifin, S., Bayu Muktyas, I., & Iswara Sukmawati, K. (2021). Product of two groups integers modulo m,n and their factor groups using Python. *Journal of Physics: Conference Series*, *1778*(1).

Arifin, S., & Muktyas, I. B. (2018). Membangkitkan Suatu Matriks Unimodular Dengan Python. *Jurnal Derivat: Jurnal Matematika Dan Pendidikan Matematika*, *5*(2), 1–10.

Arifin, S., & Muktyas, I. B. (2021). Generate a system of linear Equation through unimodular matrix using Python and Latex. *AIP Conference Proceedings*, *2331*.

Delmi, A., Suryadi, S., & Satria, Y. (2020). Digital image steganography by using edge adaptive based chaos cryptography. *Journal of Physics: Conference Series*, *1442*(1).

Devaney, R. (2018). *An Introduction To Chaotic Dynamical Systems*. CRC Press.

Guo, X., & Yang, G. (2013). The probability of rectangular unimodular matrices over Fq[x]. *Linear Algebra and Its Applications*, *438*(6), 2675–2682.

Hanson, R. (1982). Integer matrices whose inverses contain only integers. *The Two-Year College Mathematics Journal*, *13*(1), 18–21.

Irsan, M. Y. T., & Antoro, S. C. (2019). Text encryption algorithm based on chaotic map. *Journal of Physics: Conference Series*, *1341*(6).

Kordov, K. (2021). Text Encryption Algorithm For Secure Communication. *International Journal of Applied Mathematics*, *34*(4), 705–719.

Maulida, S. (2018). *Pengamanan Citra Biner dengan Algoritma Data Encryption Standard (DES) dan Bernoulli Map*. UNIVERSITAS JEMBER.

Muktyas, I. B., Sulistiawati, & Arifin, S. (2021). Digital image encryption algorithm through unimodular matrix and logistic map using Python. *AIP Conference Proceedings*, *2331*.

Nana, N., & Prasetyo, P. W. (2021). An implementation of Hill cipher and 3x3x3 rubik's cube to enhance communication security. *Bulletin of Applied Mathematics and Mathematics Education*, *1*(2), 75–92.

Oliphant, T. E. (2007). Python for scientific computing. *Computing in Science & Engineering*, *9*(3), 10–20.

Pareek, N. K., Patidar, V., & Sud, K. K. (2006). Image encryption using chaotic logistic map. *Image and Vision Computing*, *24*(9), 926–934.

Rosen, K. H. (2019). *Discrete Mathematics and its Applications* (8th ed.). McGraw-Hill Education.

Siahaan, A. P. U. (2018). *Application of Hill Cipher Algorithm in Securing Text Messages*. 55–59. https://doi.org/10.31227/osf.io/n2kdb

W, S. (2006). *Cryptography and Network Security: Principles and Practices* (4th ed.). Pearson Prentice Hall.

Ye, R., & Ma, Y. (2013). A Secure and Robust Image Encryption Scheme Based on Mixture of Multiple Generalized Bernoulli Shift Maps and Arnold Maps. *International Journal of Computer Network and Information Security*, *5*(7), 21–33.

Unimodular matrix and bernoulli map on text encryption algorithm using python

ORIGINALITY REPORT

20%	10%	9%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.sciencegate.app Internet Source	7%
2	Samsul Arifin, Indra Bayu Muktyas. "Generate a system of linear equation through unimodular matrix using Python and Latex", AIP Publishing, 2021 Publication	3%
3	repository.uin-malang.ac.id Internet Source	2%
4	Submitted to Griffith College Dublin Student Paper	2%
5	Nurhayati, Abdul Meizar, Frinto Tambunan, Erwin Ginting. "Optimizing the Complexity of Time in the Process of Multiplying Matrices in the Hill Cipher Algorithm Using the Strassen Algorithm", 2019 7th International Conference on Cyber and IT Service Management (CITSM), 2019 Publication	2%

6	Shengming Jiang. "On Securing Underwater Acoustic Networks: A Survey", IEEE Communications Surveys & Tutorials, 2018 Publication	1 %
7	2020.eac4amitans.eu Internet Source	1 %
8	Siham OleiwiTuama, Sahar A. Kadum, Zahraa Hussein. "Text Encryption Approach Using DNA Computation and Hyperchaotic System", 2021 2nd Information Technology To Enhance e-learning and Other Application (IT-ELA), 2021 Publication	1 %
9	Submitted to Universitas Amikom Student Paper	1 %
10	Submitted to Curie High School Student Paper	1 %
11	dlc.dlib.indiana.edu Internet Source	1 %
12	www.coursehero.com Internet Source	1 %
13	Anvit Negi, Devansh Saxena, Kriti Suneja. "High Level Synthesis of Chaos based Text Encryption Using Modified Hill Cipher Algorithm", 2020 IEEE 17th India Council International Conference (INDICON), 2020	1 %

Publication

Exclude quotes	Off	Exclude matches	< 1%
Exclude bibliography	On		