Cryptography using Automata Theory

Kuldeep Vayadande[*]*, Kirti Agarwal, Aadesh Kabra, Ketan Gangwal, Atharv Kinage*
VIT, Pune, Maharashtra, India

Abstract. Encryption and decryption are the two most crucial components of cryptography. Data protection is the main objective of both systems. We utilised encryption to transform plain text into ciphertext. Decryption, which works the other way around from encryption, is the process of converting encrypted text back into plain text. By using a finite state machine and the LU decomposition method, the created encryption solution ensures data secrecy for safe communication. In our suggested approach, we additionally employ lower and upper triangular matrices, which are obtained by decomposing a square matrix. During encryption, the key will be a lower triangular matrix modulated by a prime number, and during decryption, an upper triangular matrix modulated by a prime number. The tactic is beneficial. This tactic is helpful in sectors such as finance and military services where confidential material must be delivered.

1 Introduction

Cryptography relies heavily on encryption and decryption. Both tactics are primarily used to protect data. To convert plain text to encrypted text, we use an encryption technique. In the opposite direction of encryption, decryption is the process of transforming encrypted text into plain text.[1,2]. Someone who works with encryption and decoding is referred to as a "cryptographer." When a user possesses a certain piece of concealed knowledge, information is identified. A key is the hidden information that is transferred to the receiver. In cryptography, the encryption process is the process of safely converting data from one form to another. As a result, ciphertext is the value of encryption, because it cannot be read by unauthorised individuals. Encryption protects data stored on a computer system or transmitted over the internet. The most critical part of any encryption is the encryption key. The two types of keys are public and private keys. In the encryption and decryption operations, both keys are used. Public keys are available to everyone, but private keys must be kept private[3]. The key size is proportional to the encryption strength. As a result, breaking encrypted data gets increasingly harder as the key size increases. Decryption is the process of transforming encrypted text into text that our computer can read and comprehend. Decryption is the manual process of decrypting data using the necessary codes or keys. Without the secret key, decoding data is exceedingly difficult. We will get the original text after decoding. Automata theory has several applications in the field of cryptography. Deterministic finite automaton (DFA)is a field of Theory of Computation which is based on Automata. For every given string character input, a finite state machine creates a unique

[*] kuldeep.vayadande@gmail.com

encoded string. The words "deterministic" and "uniqueness of computation" are synonymous. On a comparable input symbol, non-deterministic finite automata allow for zero, one, or many transfers from one state to another. If S is a non-empty collection of k,j,l states, then the outcome is an element of S for deterministic automata and a subset of S for non-deterministic automata. A finite state machine is therefore an algorithm with a fixed count of states and transitions. Nowadays, this technique is widely used in cryptography to encode data and ensure data privacy.

a. Vernam Cipher and secret key cryptography
Vernam Cipher is a cryptographic encryption algorithm. It is one of the transposition methods used to convert plain strings to an encrypted string. In this strategy, we allot a number to each character in the normal-Text.

Method for obtaining a key:
In the Vernam cypher algorithm, we use a key to encrypt the strings, and the key's length should be the same as the string's length.

Algorithm for Encryption:
1. Assign a numerical value to each count in the string.
2. Add the two numbers together.
3. If the extra integer is larger than 26, deduct the integer from 26; else you are good to go.

b. Finite Automaton Public Key Cryptosystems:

Finite automata are the foundations of language-theoretic cryptosystems. The vast majority of cryptosystems based on language and word issues are either unsafe or fail to fulfil the cryptographic signature prosperity criteria. Automata-based cryptosystems are divided into three categories: transducers, cellular automata, and acceptors. In this study, we explore the benefits and drawbacks of popular finite automata-based cryptosystems such as FAPKC, Gysin, Wolfram, Kari, Dmsi's cryptosystems[4, 5].

Any cryptosystem must adhere to two fundamental principles: secrecy and authenticity. These two concepts provide a dilemma for the symmetric cryptosystem. The difficulty with confidentiality in symmetric cryptography is that, as we all know, a secret key is used to both convert and decode the communication. As a result, this key must be interchanged by both communication parties in some way, or they must depend on a third organisation, such as a key allocation centre, to allocate the key[6].

However, depending on a third organisation jeopardises the secret key's confidentiality. In public key cryptography, each user must produce a pair of keys, one of which is kept hidden and is known as a private key, while the other is made public and is known as a public key. It is entirely up to the program whether the original communication can be encrypted using the giver's secret key or the beneficiary's public key.

c. Cellular automata and cryptography
Cellular automata are dynamic with distinct attributes. This algorithm consists of a sequence of cells, and which are updated in sequential manner with random time. Cellular Automata is a distinct computing paradigm that gives an easy, extensible, and effective platform for revitalising large systems and executing sophisticated computations based on evidence from the surroundings[11, 12]. CA is made up of two parts. 1) a collection of cells and 2) a set of regulations.

The primary goal of LCASE's (Lightweight Cellular Automata) design is to significantly improve both parties' necessities. The model, on the other hand, has several flaws to deal with traditional security concerns and different difficulties taken into account when developing the suggested algorithm are:
a. Fast performance with a low code density.
b. Impervious to attacks like traditional cryptanalysis as well as assault timing
c. Code-effective and simple implementation.

2 Problem Statement

With the growing influence of the internet in terms of communication and e-commerce, data security is becoming increasingly important. The data security can be provided by various encryption techniques. These techniques use various algorithms to make the encryption and decryption more secure. The goal of this project is to provide a novel encrypting system that makes use of a FSM and recurrence matrix.

3 Literature Review

The paper[1], by Ayush Mittal and Dr. Ravindra Kumar Gupta suggests an approach on research to develop a novel cryptographic technique based on the LU decomposition method and finite state machine.
In the paper[2], proposed by Kearns Michael and Valiant Leslie in the year 1994, the paper showed how a constant depth threshold circuit, DFA, has significant cryptographic and number theory implications.
Application of finite automata in cryptography[3], by A.A.Sharipbay, Zh.S.Saukhanova, G.B.Shayakhmetova and N.S.Saukhanov explain key ideas in symmetric and asymmetric cryptosystems. The applications of the finite automaton model in information encryption are also examined.
Ivone Amorim, Antonio Machiavelo, and Rogério Reis' work On Linear Finite Automata and Cryptography[4] focuses on formalisation and derives fundamental conclusions on the issue. It also proposes a different criterion for a specific type of automaton.
The authors S. Nandi, B.K. and Chaudhari offer a system in their study[5], by discussing the theory and applications of Cellular Automata for a class of block and stream ciphers. Analytically, the authors demonstrated that cellular automata using XNOR rules may create an alternating group.
In this Paper[6], The writers use computer theory tools for encryption and decryption. Encryption is done using an encryptor formed by using a Turing machine and Decryption using a decryptor.
Paper [7], was written by Umesh Prasad , Sahu Madhusmita. This study offers a novel block cipher encryption and decryption procedure based on the principles of nonlinear and linear cellular automata.
The article[8], by Harsh Bhasin, Ramesh Kumar, and Neha Kathuria provides a Cellular Automata and Genetic Method-based encryption algorithm. The approach has been deployed, and preliminary results show that it is capable of competing with AES.
The authors created a producer to produce numbers arbitrarily using a unidimensional type of extended automata.[9]
In the [10], the writers proposed a new graphic symmetric cryptosystem to encode coloured pictures defined by pixels and many other colours in their paper. This cryptosystem employs a pseudo random bit generator and is based on bi-dimensional cellular-automata. Study of Cellular Automata Applications by Kumaresan and Gopalan used numerous examples and

examined the fundamental ideas of several functions of automata which discusses their uses in this field.[11]
We reviewed a paper in which the authors researched about one dimensional cellular automata and used a genetic model to see protocols of cellular automata cells resulting in sequences matching for symmetric key cryptography.[12]

4 Methodology

The following algorithm works in a way such that no one can decrypt the ciphertext without knowing the private key and automata machine used. The following table shows the conversion of alphabets/characters to specific numerical values.

Table I Character conversion Table[1]

alphabet/ symbol	numerical value	alphabet/ symbol	numerical value
@	0	P	16
A	1	Q	17
B	2	R	18
C	3	S	19
D	4	T	20
E	5	U	21
F	6	V	22
G	7	W	23
H	8	X	24
I	9	Y	25
J	10	Z	26
K	11	[	27
L	12	\	28
M	13	]	29
N	14	space	30
O	15		

Encryption :
1. Take the String input as plain text. Encode each character according to the table. Split the plain text into x number of characters, and then position them into a matrix of order x, where x is greater than 0. This square matrix is called a plain matrix.
2. Get the input for the machine by adding all the numbers of the plain matrix and converting the final result into binary form. This input is the secret key too.
3. Create a Finite state machine. For this project we have used the “Mealy machine”.
4. The recurrence matrix will be used to build the key matrix.
5. The formula for calculating the encrypted text matrix is: Encrypted matrix at ti+1th state = Encrypted matrix at ti th state (Key) (result at ti+1th state)(mod p).
6. Using the above formula calculate the cipher text matrix for all plain text matrices, at each stage.
7. Finally, reconvert the numbers of the last encrypted matrix into alphabets or characters. Lastly, send this encrypted text to the receiver

Decryption :
1. The encrypted text, matrix, private key and FSM are sent to the recipient.
2. Conversion of each symbol into corresponding value using the above table.
3. The decrypted matrix is calculated by considering the inverse of the key matrix and doing calculations based on this key matrix. This decrypted matrix is obtained at each step.

4. The decryption is done using the formula: Decrypted Matrix at ti+1th state = [inverse of {(Key matrix)(result at q(i+1)th state)}](mod p).
5. Finally, we convert the resultant matrix into string using the Character Conversion Table.
6. Thus, the original message is obtained at the end.

Instancing Encryption:
1. Take a plain text, for example 'ENCRYPT DECRYPT'.
2. Convert them to numerical form using alphabet to numerical conversion table.
3. Organise it in a matrix.

$$P = \begin{bmatrix} 5 & 14 & 3 & 18 \\ 25 & 16 & 20 & 30 \\ 4 & 5 & 3 & 18 \\ 25 & 16 & 20 & 30 \end{bmatrix}$$

4. Add the plain text's numerical values and convert it into binary form. So we get 11111100 as the secret key.
5. If we give this as an input to the Mealy machine, the output considering residue mod 5 can be given as:

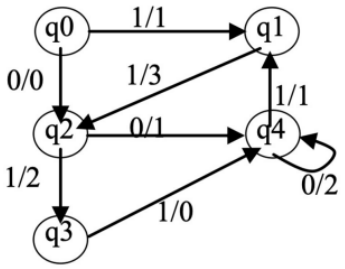

Figure 1 - Finite State Machine

This algorithm uses the Mealy machine.
6. Consider a key matrix and use formula 1 to compute the cypher matrix for each state.
7. This would result as follows.

Table 2. Cipher matrix at each state

S N	In put	Previo-us State	Prese-nt State	Out put	Cipher matrix
1	1	q_0	q_1	1	$\begin{bmatrix} 6 & 9 & 5 & 19 \\ 10 & 21 & 1 & 5 \\ 4 & 24 & 2 & 7 \\ 10 & 21 & 1 & 5 \end{bmatrix}$
2	1	q_1	q_2	3	$\begin{bmatrix} 20 & 22 & 30 & 26 \\ 9 & 25 & 6 & 27 \\ 7 & 12 & 22 & 18 \\ 9 & 25 & 6 & 27 \end{bmatrix}$
3	1	q_2	q_3	2	$\begin{bmatrix} 10 & 23 & 28 & 9 \\ 30 & 12 & 17 & 10 \\ 19 & 1 & 7 & 3 \\ 30 & 12 & 17 & 10 \end{bmatrix}$
4	1	q_3	q_4	0	$\begin{bmatrix} 10 & 23 & 28 & 9 \\ 30 & 12 & 17 & 10 \\ 19 & 1 & 7 & 3 \\ 30 & 12 & 17 & 10 \end{bmatrix}$
5	1	q_4	q_1	1	$\begin{bmatrix} 16 & 11 & 5 & 20 \\ 20 & 0 & 24 & 20 \\ 19 & 24 & 5 & 13 \\ 20 & 0 & 24 & 20 \end{bmatrix}$
6	1	q_1	q_2	3	$\begin{bmatrix} 28 & 25 & 8 & 29 \\ 30 & 5 & 15 & 2 \\ 2 & 20 & 30 & 0 \\ 30 & 5 & 15 & 2 \end{bmatrix}$
7	0	q_2	q_4	1	$\begin{bmatrix} 8 & 5 & 28 & 6 \\ 19 & 28 & 14 & 29 \\ 15 & 3 & 5 & 22 \\ 19 & 28 & 14 & 29 \end{bmatrix}$
8	0	q_4	q_4	2	$\begin{bmatrix} 24 & 16 & 16 & 0 \\ 1 & 1 & 14 & 12 \\ 1 & 22 & 6 & 13 \\ 1 & 1 & 14 & 12 \end{bmatrix}$

As shown in the table we have 8 states according to the key matrix binary conversion. We gave this binary string as input to the Mealy machine of modulus 5 and the output of the Mealy machine is used as the power of matrix in encryption and decryption algorithm. The last column of the cipher matrix is calculated at each state by using the formula: Encrypted matrix at ti+1th state = Encrypted matrix at ti th state (Key) (result at ti+1th state)(mod p). Now this cipher(encrypted) matrix is used as the input matrix for the next state and with the output of the Mealy machine at that specific state we calculate the cipher matrix at that specific state. The cipher matrix at last state is the final encrypted matrix which is used to encrypt the string. Therefore , Reconverting the resultant matrix into character string with the help of the character conversion table , we get the following encrypted string.

The encrypted string is : XPP@AANLAVFMAANL

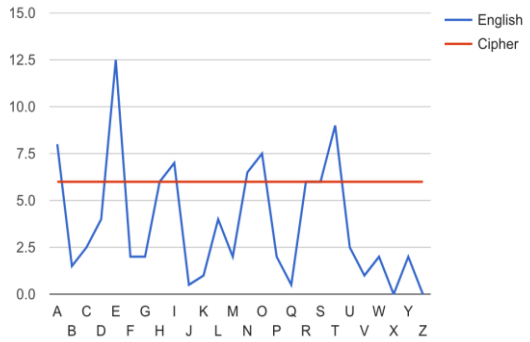

Figure 2

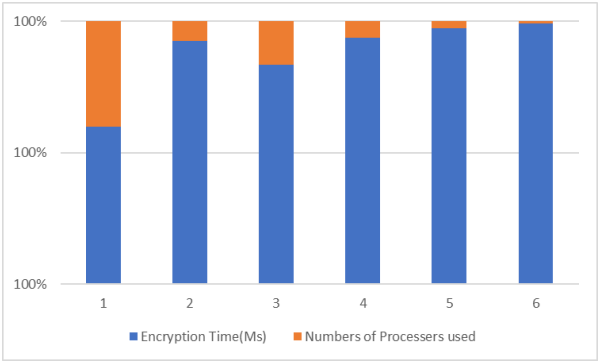

Figure 3

Instancing Decryption :
1. Using the alphabet to numerical conversion table, change each symbol of the final ciphered text into a numeric value.
2. Arrange the numeric values into the square matrix.
3. Formula for calculating the matrix : Encrypted matrix at state = Encrypted matrix at ti+1th state * [inverse of {(Key)(output at ti+1 th state)}](mod p), where we take p = 31.
4. The final matrix obtained after performing all the Decryption steps is :

$$\begin{bmatrix} 5 & 14 & 3 & 18 \\ 25 & 16 & 20 & 30 \\ 4 & 5 & 3 & 18 \\ 25 & 16 & 20 & 30 \end{bmatrix}$$

5. Re-convert each character of the above matrix into numeric using the above table. We receive the original message i.e. - ENCRYPT DECRYPT.

5 Future Scope

Although the fate of cryptography is unknown, one thing is certain: cryptography will continue to evolve and progress. Cryptography can be used in various sectors -Banking and Military services. Hence, data encryption and decryption becomes important. The algorithm

of the encryption technique can be made even stronger to prevent malicious attacks. The proposed system works well for encrypting and decrypting short text messages. The system can be optimised for long text messages. The number of special characters considered in the proposed system are limited. This number of special characters can be increased so that the system becomes more efficient.

6 Conclusion

The proposed approach is based on several matrix operations and a FSM. The selected FSM, recurrence matrix, other matrix operations and secret key all contribute to the security. Although extracting original information from encrypted text is challenging, a method exists. This study proposes two optimization targets to define greater algorithm efficiency: the fewest ciphertext pairings and the minimum computational complexity. We provide the related optimum key recovery algorithms with varied optimization orders of these two goals.

References

1. M. Kearns, L. Valiant "*Cryptographic Limitations on Learning Boolean Formulae and Finite Automata*". Journal of the ACM, Volume **41**, Issue 1Jan. (1994) pp 67–95.
2. A.A.Sharipbay, Zh.S.Sarukhanova, G.B.Shayakhmetova, N.Z.Soukhanov "*Application of finite automata in cryptography*" ICEMIS '19: Proceedings of the 5th International Conference on Engineering and MISJune (2019) Article No.: 20 Pages 1–3.
3. I. Amorim, A. Maquiavelo, R. Reis "*On Linear Finite Automata and Cryptography*", Faculdade De Ciências Universidade Do Porto, August (2011)
4. Z. Saqib, M. Ahmad Shahid and M. Umair "*Encryption and Decryption Using Automata Theory*", International Journal Of Multidisciplinary Sciences And Engineering, Vol. **6**, No. 4, (2015)
5. S. Panda, M. Sahu and U. Rout "*Encryption and Decryption algorithm using two dimensional*", International Journal of Communication Networks and Security ,Vol. **1**, Article 5, (2011)
6. H. Bhasin, R. Kumar, N. Kathuria "*Cryptography Using Cellular Automata*", National Conference on Advances in Computational Intelligence, (2011)
7. Z. Xuelong, L. Qianmu, X. Manwu and L. Fengyu , "*A Symmetric Cryptography based on Extended cellular automata"* by IEEE International Conference on Systems, Man and Cybernetics Volume: **1** (2005)
8. A.Mara, L.Encinas , A.Encinas, A. Rey, and G. S´anchez *Graphic Cryptography with Pseudorandom Bit Generators and Cellular Automata* Gonzalo Knowledge-Based Intelligent Information and Engineering Systems, 7th International Conference , volume **2773** LNAI (2007)
9. G. Kumaresan, N.P. Gopalan , *An Analytical Study of Cellular Automata and its applications in Cryptography*, I. J. Computer Network and Information Security, Vol. **12**, 45-54, (2017)
10. M. Szaban, F. Seredynski and P. Bouvry , *Evolving Collective Behaviour of Cellular Automata for Cryptography* by MELECON 2006 - 2006 IEEE Mediterranean Electrotechnical Conference, 1-4244-0087-2, (2006)
11. V. Kuldeep, R. Pokarne, M. Phaldesai, T. Bhuruk, T. Patil, and P. Kumar. "*Simulation Of Conway's Game Of Life Using Cellular Automata*." International Research Journal of Engineering and Technology (IRJET) **9**, no. 01 (2022): 2395-0056.

12. V. Kuldeep, R. Mandhana, K. Paralkar, D. Pawal, S. Deshpande, and V. Sonkusale. "*Pattern Matching in File System.*" International Journal of Computer Applications 975: 8887. (2022)
13. K.Vayadande, N.Bhavar, S.Chauhan, S. Kulkarni, A.Thorat, Cellular automata Image Encryption. (2022)
14. K. Vayadande, H. More, O. More, S. Muley, A. Pathak, V. Talanikar, "*Pac Man: Game Development using PDA and OOP*", International Research Journal (2022)
15. K. Vayadande, P. Sheth, A. Shelke, V. Patil, S. Shevate, C. Sawakare, "*Simulation and Testing*", International Journal of Computer Sciences and Engineering, Vol.**10**, Issue.1, pp.13-17, 2022.
16. R. Gurav, S. Suryawanshi, P. Narkhede, S. Patil, S. Hukare, K. Vayadande, "*Universal Turing machine simulator*", International Journal of Advance Research, Ideas and Innovations in Technology, ISSN: 2454-132X, (Volume **8**, Issue 1 - V8I1-1268)(2022)
17. K. Vayadande, K. Patel, N. Punde, S.Patil, S. Nikam, S.Pathrabe, "*Non-Deterministic Finite Automata to Deterministic Finite Automata Conversion by Subset Construction Method using Python*," International Journal of Computer Sciences and Engineering, Vol.**10**, Issue.1, pp.1-5, 2022.
18. K. Vayadande and S. Pate and N. Agarwal and D. Navale and A. Nawale and P. Parakh, "*Modulo Calculator Using Tkinter Library*", EasyChair Preprint no. 7578, EasyChair, 2022.