A Fast Cryptosystem Using Reversible Cellular Automata

Said BOUCHKAREN
Department of Mathematics and Computer Science / LABTIC
National School of Applied Sciences of Tangier
AbdelMalek Essaadi University
B.P. 1818, Tangier Morocco

Saiida LAZAAR
Department of Mathematics and Computer Science / LABTIC
National School of Applied Sciences of Tangier
AbdelMalek Essaadi University
B.P. 1818,Tangier Morocco

***Abstract*—This article defines a new algorithm for a secret key cryptosystem using cellular automata which is a promising approach to cryptography. Our algorithm is based on cellular automata built on a set of reversible rules which have the ability to construct unpredictable secret keys using MARGOLUS neighborhood. To prove the feasibility of the algorithm, we present some tests of encryption, decryption and diffusion; a CPU time comparison with an encryption algorithm by blocks as for instance AES-256 is established. On the other hand, the security of the algorithm is proved and the implemented algorithm resists against a brute force attack.**

Keywords—AES; Cellular automata; Diffusion; Cryptosystem; MARGOLUS neighborhood

I. INTRODUCTION

Cryptographic algorithms are used to secure computer networks, electronic transactions or information exchanges. They are implemented in security protocols, electronic chips, etc. Otherwise, and since its appearance, the field of cryptography has experienced a great evolution, mainly in the design of many methods of encryption based on public or secret keys.

The methods of modern cryptography can be divided into two main categories: symmetric and asymmetric cryptography. Each cryptosystem uses keys to generate from the clear text a cipher text. The most known symmetric cryptography systems are DES, AES, RC4 and RC5, [1]. They are used in a secure communication protocols as TLS. Unlike symmetric cryptography, the asymmetric cryptography uses public and secret keys for encryption and decryption. The best known algorithm is RSA [1]; it is implemented in the SSL protocol.

Our work is part of a new approach to cryptosystems based on cellular automata (CA) which presents a promising approach to cryptography. CA gives a secret key for the encryption which cannot be predicted since it evolves a chaotic and complex system starting from an initial state [8-10]. A Brief history of CA can be found in [4].

In one dimension, many encryption concepts based on CA was studied, the most known ones belong to Wolfram [13-14]. In two dimension, some cryptosystems based on CA have been constructed for public and secret keys [9],[11],[15]; for example, CA were used to construct cryptosystem based on Vernam cipher and generated keys with a pseudo-random numbers sequence; CA were also used for block cipher. Concerning our work, we aim to design and to implement a novel two dimensional secret key cryptosystem based on reversible CA using MARGOLUS neighborhood [2]. We remind however that reversibility concept has been used for block encryption in one dimension; for more details, see [8].

This paper is organized as follows. The first section consists on a brief review of symmetric and asymmetric cryptographic methods based on secret and public keys. Section 2 outlines CA and section 3 focuses on MARGOLUS neighborhood. Section 4 describes our cryptosystem. To prove its feasibility, a simulation will be presented in section 5; it holds on the diffusion and performance tests where a comparison with the AES algorithm is established. Section 6 concludes the paper.

II. CELLULAR AUTOMATA AND REVERSIBILITY

A cellular automaton (CA) is a dynamic system defined by the following 4-tuple: dimension, set of finite states, neighborhood and set of rules. Dimension defines number of cells. Cells are updated accordingly to some rule. Such rule is based on the state of the cell and the neighborhood.

More precisely, let A be a cellular automaton defined by $A = \{S, Z^d, f, V\}$ where:

- S is a finite set of states,
- Z is the set of integers,
- d is the size of the automaton,
- Z^d is the space of the automaton,
- $f : S^n \to S$ is the rule (transition function), $n = card(V)\ where\ V = \{v_1, v_2, \dots, v_n\} \subseteq Z^d$ is the set of neighborhood, $(v_i \in Z^d, 1 \leq i \leq n)$.

We call configuration the allocation of the state of each automaton cell. To illustrate, we present the following test:

$A = \{S, Z^1, f, V\}$

- $S = \{0, 1\}$, two states,

- $V = \{-1, 0, 1\}$ (the neighbors of the cell i are i-1, i+0, i+1),
- f is defined by the following box:

V(∗)	7	6	5	4	3	2	1	0
$f(V)$	0	0	1	0	0	1	0	1

(*) Expressed in the basis 10 (5 ➔ s[i]=0, s[i-1]=1, s[i+1]=1).

Let the configuration at time$t$:

1	0	1	0	0	**1**	0	0	0	1

At time t +1 the configuration will be:

0	1	1	0	0	**1**	0	1	0	0

According to the definition table of f , we have in binary 00100101 which are worth 37 in decimal, we are talking about the rule 37. In total, there are $2^8 = 256$ rules. For an automaton of dimension d where the set of states is $S = \{s_1, s_2, \dots, s_n\}$, we can use n^d rules.

In this work, we use a reversible cellular automaton according to this definition: the automaton A with the evolution function f is reversible if there is an automaton B with the evolution function g such that for each configuration, $f(g(c)) = c$. For more details, we can refer to the works published in [3-6].-For the one-dimensional CA, it is possible to check if the automaton is reversible [7]. For the CA of other dimensions, the reversibility is undecidable [6].

III. MARGOLUS NEIGHBORHOODS

We construct two-dimensional CA by blocks based on MARGOLUS neighborhood. We remind that if we consider two-dimensional grid then it is possible to define different kinds of neighborhoods; the most common ones are Von Neumann (related to four neighborhoods), Moore (associated to nine neighborhoods) or MARGOLUS [4],[9],[12]. To justify our choice, MARGOLUS neighborhood allows creation of reversible CA and this reversibility gives more security to the encryption procedure.

For MARGOLUS neighborhood, the automaton cell is divided into blocks of n cells (four cells for example), we applied the transition function to each block. To create reversibility, we use the reversible transition function that can be built by creating a mapping between the set of states.

To illustrate, let A be a cellular automaton of dimension 2, with two states S = {0, 1}, we construct a block of four cells as indicated in Figure 1. The arrow indicates the conventional sense.

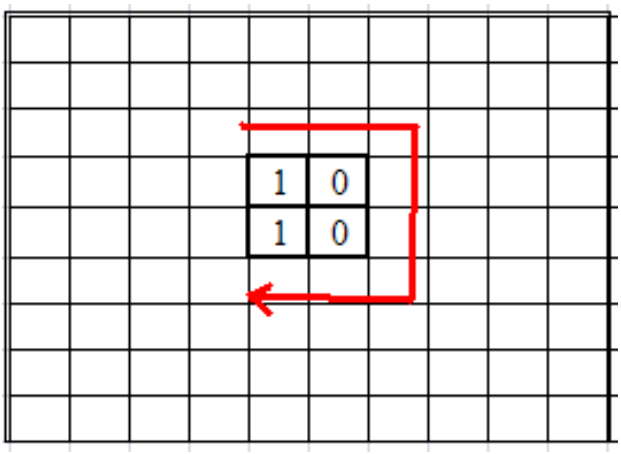

Fig. 1. Bits interpretation Sense

Four cells imply the use of four bits, thus the minimum value is 0000 (in binary) and the maximum value is 1111 (in binary and 15 in decimal). Each cell may contain a value of the set D = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15}. Therefore, to construct a reversible rule, we just need to take a bijective application $f: D \to D$.

Figure 2 explains the concept of the MARGOLUS neighborhood. First, we take the original automaton « A », then we partition into blocks of n cells (4 cells in this case). Second, we partition starting with the first index (even partitioning (B) on Figure 2).

Finally, we redo the odd partitioning starting with an odd index (C), see Figure 2.

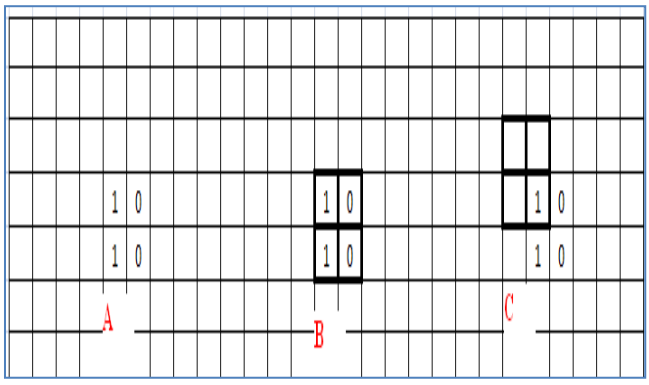

Fig. 2. Partitioning of a 2-D cellular automaton in MARGOLUS neighborhood basis.

IV. PROPOSED CRYPTO-SYSTEM

The proposed cryptosystem is based on two-dimensional reversible CA using MARGOLUS neighborhood, its secret key is the transition rule of the cellular automaton.

A. Description of the algorithm

Encryption step

- Divide the unencrypted data into blocks of $4T^2$-bits (for example 1024-bits)
- Arrange the $4T^2$-bits in a square matrix M of dimension 2 with a size of 2Tx2T
- Generate a reversible transition rule R (Rr is the inverse rule of R)
- Generate a key K={R, Rr}
- For i←1 until 6

  M=PRule(K,M)
  M=IRule(K,M)
  End

- Group the bits of the matrix M to obtain the encrypted data.

Decryption step

As already mentioned, the proposed system is symmetric; for the decryption, we use the same key as in the encryption step.

- Divide the encrypted data into blocks of $4T^2$-bits (for example 1024-bits)
- Arrange the $4T^2$-bits in a square matrix M of two dimension with a size of 2Tx2T

  ***For** i←1**util** 6*
  M=IRule(K,M)
  M=PRule(K,M)
  End

- Group the bits of the matrix M to obtain the unencrypted data.

B. Transformation PRule() and IRule()

Consider the following data presented in Figure 3:

Fig. 3. Illustration data

PRule(), IRule() are the functions that evolve the cellular automaton, they use the MARGOLUS neighborhood described above.

PRule() applies on the even blocks (the red blocks in the graph), in opposite, IRule() applies on the odd blocks (the green blocks in the graph).

C. Security of the algorithm

The security of this algorithm is based on the choice of CA parameters: we can increase or decrease the neighborhood size or/and the number of states.

Let N be the neighborhood size and S the number of states. The maximum possible number of keys is $(S^N)!$ which making impossible to apply a brute force attack with an exhaustive search of keys; for example, if S =2 and $N = 8$, then we have to try $2^8!$=256! $\approx 8.10^{506}$ combinations that are impossible to achieve within a reasonable time.

V. SIMULATION

For the simulation, we used a cellular automaton by blocks with the following data:

- Number of blocks per cell: 4 (2X2)
- Cardinal of the set of states : 2 (0 and 1)
- Transition rule:
  R={0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15} and its inverse Rr={3,2,5,8,9,7,0,4,1,13,6,14,10,15,11,12}

For the system parameters, we used the following data:

- Message = « Crypto-system based on the CAs. »
- K={R, Rr}
- 256-bits in each data block.

To make the encrypted data readable, we used the *Base64* encoding.

The clear message encoded by *Base64* is:
« Q3J5cHRvLXN5c3Q/bWUgYmFzPyBzdXIgbGVzIEFDcy4 = »
The encrypted message is:
« isFbk69OTBjYrZdRdOb46riq+5NlrrYZN/uuYE5sQF8= ».

A. Diffusion application

The diffusion calculates the influence to change the bits in the clear message (plain text) onto the encrypted message keeping constant the key. Figure 4 gives diffusion tests of CA cryptosystem and AES algorithm by taking a plain text of size 256-bits with a key of 256-bits. It is observed that diffusion levels obtained by CA cryptosystem are better than AES.

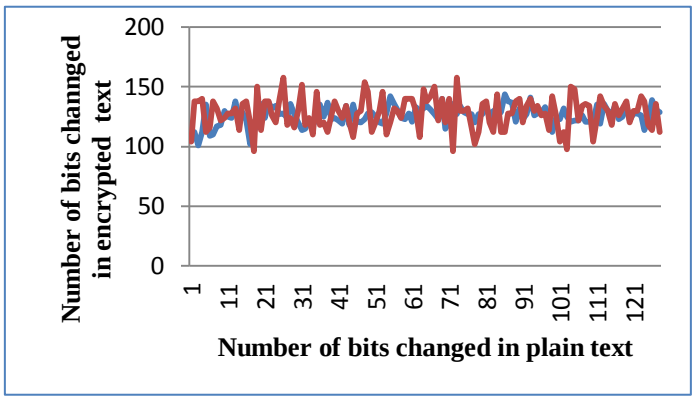

Fig. 4. Diffusion tests: Red curve corresponds to CA cryptosystem and blue curve corresponds to AES algorithm.

B. *Performance test*

We compare our CA cryptosystem with a block algorithm as for instance AES-256. This is illustrated in Figure 5. The comparison focuses on the performance tests (CPU time in ms). The encryption using CA is faster than AES algorithm

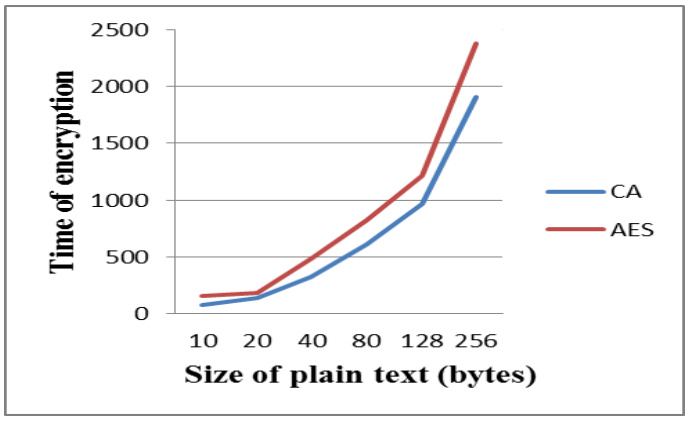

Fig. 5. Comparison of CPU times between CA cryptosystem (blue curve) and AES algorithm (red curve).

VI. CONCLUSION

In this paper, we presented a brief review of symmetric and asymmetric cryptographic methods; we introduced the concept of cellular automata as a promising approach to cryptography allowing creation of unpredictable cryptosystem keys born through a chaotic and complex system. We outlined also the neighborhood concept that is linked to the creation of cellular automaton and we described Margolus neighborhood which we selected to build our two-dimensional reversible CA. We constructed a fast secret key cryptosystem and we demonstrated its feasibility.

We realized a test of diffusion and a CPU time comparison with a block cipher algorithm; the comparison shows that the proposed algorithm is faster than AES-256. The security is proved and the implemented algorithm resists against a brute force attack. We remind that some new research works focus on network sensor security and use encryption algorithms to improve the security of data and energy consumption [16-18]; in this context, we aim to implement our CA cryptosystem as a lightweight security protocol.

REFERENCES

[1] S. Bruce Schneier, “Applied Cryptography”, John Wiley & Sons New York, 1996.

[2] T. Toffoli and N. Margolus, “Invertible Cellular automata”, Physica D 45, 1990.

[3] S. Wolfram, “A New Kind of science”, Wolfram Media, 2002.

[4] P. Sarkar, “A Brief History of Cellular Automata”, ACM Computing Surveys, 2000.

[5] J. L. Schiff, “Cellular automata: A discrete view of the world”, Wiley, 2008.

[6] J. Kari. “Reversible cellular automata”, Springer Berlin Heidelberg, 2005

[7] J. Kari, “Reversibility and Surjectivity problems of cellular automata”, Journal of Computer and system sciences 48, 1994.

[8] Marcin Seredynski and Pascal Bouvry, “Block Encryption Using Reversible Cellular Automata”, Lecture Notes in Computer Science, Volume 3305, pp 785-792, 2004.

[9] [9] Sambhu Prasad Panda, Madhusmita Sahu, Umesh Prasad Rout and Surendra Kumar Nanda, “Encryption and Decryption algorithm using two dimensional cellular automata rules in Cryptography”, International Journal of Communication Network & Security, Volume-1, Issue-1, 2011.

[10] Somanath Tripathy and Sukumar Nand, “LCASE: Lightweight Cellular Automata-based Symmetric-key Encryption” International Journal of Network Security, Vol.8, No.2, PP.243-252, Mar. 2009.

[11] Franciszek Seredynski, Pascal Bouvry, Albert Y. Zomaya, “Cellular automata computations and secret key cryptography Parallel Computing”, 2004.

[12] Joaquin Cerda, Rafael Gadea and Guillermo Paya. “Implementing a Margolus Neighborhodd Cellular Automata on a FPGA”, Lecture Notes in Computer Science, Volume 2687, pp 121-128, 2003.

[13] Stephen Wolfram, “Cryptography with cellular automata”, Lecture Notes in Computer Science. Volume 218, pp 429-432, 1986.

[14] Stephen Wolfram, “Cellular automata and complexity”, Addison-Wesley, 1994.

[15] Norman H, Packard and Stephen Wolfram, “Two dimensional cellular automata”, Journal of statistical Physics, 1985.

[16] Alvaro Araujo, Javier Blesa, Elena Romero and Daniel Villanueva, “Security in cognitive wireless sensor networks. Challenges and open problems”, EURASIP Journal on Wireless Communications and Networking, 2012.

[17] Nabil Ali Alrajeh, S. Khan and Bilal Shams, “Intrusion Detection Systems in Wireless Sensor Networks: A Review”, International Journal of Distributed Sensor Networks, 2013.

[18] Alexandros Fragkiadakis, Vangelis Angelakis and Elias Z. Tragos, “Securing Cognitive Wireless Sensor Networks: A Survey”, 2014.