Analyzing Cryptographical Properties of 1D and 2D Cellular Automaton

Sindhuja G\textsuperscript{1}, Vinupriya P\textsuperscript{2}, Jyothirbindhu V\textsuperscript{3}, Mohaideen Pitchai K\textsuperscript{4} and Bhuvaneswari M\textsuperscript{5}

\textsuperscript{1,2,3,4,5}Student, Department of Computer Science and Engineering, National Engineering College, K. R. Nagar, Kovilpatti, India.

ABSTRACT

Cellular automata (CA) has been used in pseudorandom number generation for over a decade. Cryptography has become a basic requirement in this age of global electronic connectivity to secure data storage and transmission against the possibility of message eavesdropping and electronic fraud. In this paper, we analyze the cryptographical properties for both 1D and 2D cellular automaton rules. The cellular automaton rules are analyzed with respect to nonlinearity, algebraic degree, d-monomial test and balancedness.

1. INTRODUCTION

Several important computer simulation methods rely on random numbers, including Monte Carlo techniques, Brownian dynamics, and stochastic optimization methods such as simulated annealing. The quality of the results of these methods critically depends on the quality of the random sequence as measured by suitable statistical tests. Computational efficiency is also an important aspect when very long sequences of random numbers have to be produced. Random numbers are also needed in another important application area: built-in self-test devices for VLSI circuits [3].

In recent years, cellular automata (CA) have been found as an attractive modeling tool for various applications, such as, pattern recognition, image processing, data compression, encryption and specially VLSI design and test. However, for all such applications, a special class of CA, called as linear/additive CA, has been utilized. Since linear CA, limit the search space, we may not reach to the best result while searching for the solution to a problem. Nonlinear CA can be an alternative to linear/additive CA for achieving desired solutions in different applications. Several researchers have attempted to apply the pseudorandomness of CA to cryptography. The cryptanalysis of linear CA based cryptographic techniques show that nonlinearity is needed for Cryptographic applications. However, nonlinear CA shows high correlation. Hence, for cryptography Cellular Automata rules need to be nonlinear as well as satisfy additional properties.

In this paper, we analyze the CA by modeling its rule as a Boolean function relating output bits with input bits. Parameters like nonlinearity, balancedness and algebraic degree are known to be important for the cryptographic analysis of Boolean functions [5].

2. CONCEPT OF CELLULAR AUTOMATA

A cellular automaton consists of regular grid of cells in which each cell can have finite number of possible states. The state of a cell at a given time step is updated in parallel and determined by the previous states of surrounding
neighborhood of cells with the help of a specified transition rule. The state of each cell is updated simultaneously at discrete time steps based on the states in its neighborhood at the preceding time step.

There is one dimensional, two dimensional CA used to solve different type of problems. One dimensional CA (1D CA) consists of linear arrays of cells whereas in two dimensional CA (2D CA), cells are arranged in a rectangular or hexagonal grid with connections among the neighboring cells, which is depicted in figure 1 and 2 respectively.

A CA can be represented with five tuple, $C = \{L, N, Q, \delta, q_0\}$; where $L$ is the regular lattice of cells, $Q$ is the finite set of states, $q_0$ is called the initial state and $q_0 \in Q$, $N$ is a finite set (of size $n = |N|$) of neighborhood indices such that for all $r \in L$, for all $c \in N$, $r + c \in L$ and $\delta: Qn \rightarrow Q$ is the transition function [6].

2.1. Neighborhood Structure
The neighborhood of a cell, called the core cell (or central cell), made up of the core cell and those surrounding cells whose states determine the next state of the core cell. There are different neighborhood structures for cellular automata. The two most commonly used neighborhoods are Von Neumann and Moore neighborhood, shown in Figure 3.

Fig 1: Structure of 1D CA

Fig 2: Structure of 2D CA

Fig 3: Neighborhood model (a) Von Neumann, (b) Moore
Von Neumann neighborhood has five cells, consisting of the cell and its four immediate non-diagonal neighbors and has a radius of 1. The radius of a neighborhood is defined to be the maximum distance from the core cell, horizontally or vertically, to cells in the neighborhood.

Moore neighborhood has nine cells, consisting of the cell and its eight surrounding neighbors and has a radius of 1. Extended Moore neighborhood composed of the same cells as the Moore neighborhood, but the radius of neighborhood is increased to 2 [7].

2.2. Boundary Conditions

Most popular boundary conditions are null boundary and periodic boundary conditions which are used when a transition rule is applied to the boundary cells of CA. In null boundary conditions, the extreme cells are connected to logic 0-state and in periodic boundary conditions the extreme cells are adjacent to each other.

2.3. One-dimensional generators

Wolfram first proposed CA as Pseudo Random Number Generators (PRNG). In particular, he extensively studied the bit sequences generated by the 1−d, r = 1, rule 30. The rule number for one-dimensional, r = 1 rules represents in decimal format the binary number encoding the rule table. Thus, \( f(1 1 1) = 0, f(1 1 0) = 0, f(1 0 1) = 0, f(1 0 0) = 1, f(0 1 1) = 1, f(0 1 0) = 1, f(0 0 1) = 1, f(0 0 0) = 0 \), is denoted rule 30. In the previous representation, the groups of three bits between parentheses represent all the possible neighborhood states and the single bit after the equal sign is the resulting output bit at the next time step. The rule number is obtained by multiplying each output bit by the corresponding power of two and adding the results. Position 0 corresponds to the (0 0 0) neighborhood. There exist \( 2^8 = 256 \) possible rules with \( r = 1 \). In Boolean form, rule 30 can be written as:

\[
s_i(t+1) = s_{i,j}(t) \oplus (s_i(t) \lor s_{i+1,j}(t)),
\]

where \( s_i(t) \) of the state of cell \( i \) at time \( t \). The formula gives the state of cell \( i \) at time step \( t +1 \) as a Boolean function of the states of the neighboring cells at time \( t \) [1].

2.4. Two dimensional generators

In 2D Nine Neighborhood CA the next state of a particular cell is affected by the current state of itself and eight cells in its nearest neighborhood. Such dependencies are accounted by various rules.

Let \( s_{i,j}(t) \) be the state of the cell at row \( i \) and column \( j \), at time \( t \). Its state at the next time step, \( s_{i,j}(t + 1) \) is then computed as follows:

\[
S_{i,j}(t + 1) = X \oplus (C \cdot S_{i,j}(t)) \oplus (N \cdot S_{i-1,j}(t)) \\
\oplus (W \cdot S_{i,j-1}(t)) \oplus (S \cdot S_{i+1,j}(t)) \\
\oplus (E \cdot S_{i,j+1}(t)),
\]
where $\oplus$ and $\cdot$ (dot) are the operations XOR and AND, respectively, and X, C (center), N (north), S (south), W (west), and E (east) are binary variables. C, N, S, W and E denote whether the respective neighboring cell state is taken into account (a value of 1) or not (a value of 0) [1].

3. PROPERTIES OF CRYPTOGRAPHICALLY ROBUST CELLULAR AUTOMATA RULES

This section describes the fundamental properties that should be possessed by cryptographic primitives. Here we have explored the tests for cryptographic properties like d-

3.1. Nonlinearity

Nonlinearity of a boolean function $f$ of $n$ variables is the minimum hamming distance from $f$ to the set of affine functions with $n$ variables [2]. If $A_n$ be the set of all $n$ bit affine boolean function, then

$$nl(f) = \min_{h \in A_n} d(f, h)$$

An affine function is a linear function or the complement of a linear function [8].

Example: $f(x_1, x_2, x_3) = x_1 \oplus x_2 \oplus 1$

The Hamming distance between two functions is the number of places where their truth table representations disagree [8].

Example: The hamming distance between 1011101 and 1001001 is 2.

If the pseudo random numbers are linear in nature, then we can easily predict the upcoming pseudo random numbers. This will lead to the loss of cryptographic security. Hence, a good pseudo random number should be nonlinear in nature. Thus, nonlinearity is one of the important cryptographic properties. Pseudo random numbers with greater nonlinearity are considered as good pseudo random numbers.

3.2. Balancedness

A Boolean function of $n$ variables is said to be

Balanced if for exactly $2^n - 1$ assignments the function $f$ will evaluate to 0 and for exactly $2^n - 1$ assignments the function $f$ will evaluate to 1[8].

3.3. Algebraic Degree

The maximum number of literals in any conjunction of ANF of a Boolean function is called its degree. For example

$f(x_1, x_2) = x_1 \oplus x_2 \oplus x_2$ has algebraic degree 2. Linear functions have algebraic degree 1 [8].

3.4. d-Monomial test

It is a statistical test for pseudo randomness. If a Boolean function of $n$ Boolean variables is a good pseudo random sequence generator, then it will have $\frac{1}{2} \binom{n}{d}$ d-degree monomials. It is first introduced by [9]. It is a simple procedure to test pseudo randomness of a sequence. Beside its simplicity it has gained a huge acceptance in
cryptography community. d-monomial test has been used in [10] to test the pseudo randomness of the sequence generated by the CA.

For example, consider the function of rule 60, 
\( f(x_1, x_2, x_3) = x_1 \oplus x_2 \), it has 2, 1-degree monomials and 0, 2 degree monomial. The ideal number of 1, 2 and 3 degree monomials of 1D CA would be 
\[
\frac{1}{2} \begin{pmatrix} 3 \\ 1 \end{pmatrix} = 1.5, \frac{1}{2} \begin{pmatrix} 3 \\ 2 \end{pmatrix} = 1.5, \frac{1}{2} \begin{pmatrix} 3 \\ 3 \end{pmatrix} = 0.5
\]

It turns out that it has 2, 1-degree monomials more and 1 2-degree monomial less, hence it is expected to be non-pseudo random. On the other hand, the function of rule 120 \( f(x_1, x_2, x_3) = x_1 \oplus x_2 \oplus x_3 \) is expected to be a good pseudo random generator.

The ideal number of 1, 2, 3, 4 and 5 degree monomials of 2D CA would be
\[
\begin{align*}
\frac{1}{2} \begin{pmatrix} 5 \\ 1 \end{pmatrix} &= 2.5, \\
\frac{1}{2} \begin{pmatrix} 5 \\ 2 \end{pmatrix} &= 5, \\
\frac{1}{2} \begin{pmatrix} 5 \\ 3 \end{pmatrix} &= 5, \\
\frac{1}{2} \begin{pmatrix} 5 \\ 4 \end{pmatrix} &= 2.5, \\
\frac{1}{2} \begin{pmatrix} 5 \\ 5 \end{pmatrix} &= 0.5
\end{align*}
\]

For example, consider the cell (2, 2), then the function of rule 15214748, 
\[
f(x_{1,2}, x_{2,1}, x_{2,2}, x_{2,3}, x_{3,2}) = (x_{1,2} \&\& x_{2,2}) \oplus (x_{1,2} \&\& x_{2,3}) \oplus (x_{2,1} \&\& x_{2,2}) \oplus (x_{2,1} \&\& x_{2,3}) \oplus (x_{2,2} \&\& x_{3,2}) \oplus (x_{1,2} \&\& x_{2,1} \&\& x_{2,3}) \oplus (x_{1,2} \&\& x_{2,1} \&\& x_{2,3}) \oplus (x_{1,2} \&\& x_{2,2} \&\& x_{2,3}) \oplus (x_{1,2} \&\& x_{2,3} \&\& x_{3,2}) \oplus (x_{2,1} \&\& x_{2,3} \&\& x_{3,2}) \oplus (x_{1,2} \&\& x_{2,1} \&\& x_{2,2} \&\& x_{2,3}) \oplus x_{2,2} \oplus x_{2,3}
\]
is expected to be a good pseudorandom generator.

4. RESULT

The main objective of the work is to find out the rule that satisfy the cryptographic properties. Our tool is deriving the linearity, balancedness, algebraic degree and d-monomial test. In order to identify the best rules, it is necessary to feed the inputs like range of inputs, type of cellular automata.

<table>
<thead>
<tr>
<th>RULE</th>
<th>NON-LINEARITY</th>
<th>BALANCEDNESS</th>
<th>ALGEBRAIC DEGREE</th>
<th>D-MONOMIAL TEST</th>
</tr>
</thead>
<tbody>
<tr>
<td>102</td>
<td>2</td>
<td>TRUE</td>
<td>2</td>
<td>GOOD</td>
</tr>
<tr>
<td>103</td>
<td>2</td>
<td>TRUE</td>
<td>2</td>
<td>GOOD</td>
</tr>
<tr>
<td>120</td>
<td>2</td>
<td>TRUE</td>
<td>2</td>
<td>GOOD</td>
</tr>
<tr>
<td>135</td>
<td>2</td>
<td>TRUE</td>
<td>2</td>
<td>GOOD</td>
</tr>
<tr>
<td>147</td>
<td>2</td>
<td>TRUE</td>
<td>2</td>
<td>GOOD</td>
</tr>
<tr>
<td>148</td>
<td>2</td>
<td>TRUE</td>
<td>2</td>
<td>GOOD</td>
</tr>
</tbody>
</table>

Fig 4: 1D CA rules which are satisfying the above four cryptographical properties
The following figure 4 shows that the properties of 1D CA rules. If the type CA is one dimensional, there is possibility of 256 rules. Based on this analysis, we could identify the best possible rule that will generate longest period Pseudo Random Numbers. The figure 5 portrays the cryptographical properties analysis of 2D CA rules.

<table>
<thead>
<tr>
<th>RULE</th>
<th>NON-LINEARITY</th>
<th>BALANCEDNESS</th>
<th>ALGEBRAIC DEGREE</th>
<th>D-MONOMIAL TEST</th>
</tr>
</thead>
<tbody>
<tr>
<td>142290000</td>
<td>TRUE</td>
<td>10</td>
<td>4</td>
<td>GOOD</td>
</tr>
<tr>
<td>14239786</td>
<td>TRUE</td>
<td>10</td>
<td>4</td>
<td>GOOD</td>
</tr>
</tbody>
</table>

Fig 5: 2D CA rules between rule 142290000 and rule 142300000 which are satisfying the above four cryptographical properties

5. CONCLUSION

We have described about the cryptographical properties of pseudo random numbers such as nonlinearity, balancedness, algebraic degree and d-monomial test. This will produce good pseudo random numbers, which can be used for secure communication.

REFERENCE


[4] Marco Tomassini, Mathieu Perrenoud, "Cryptography with cellular automata”, Institute of Computer Science, University of Lausanne, 1015 Lausanne, Switzerland Received 7 December 2000; received in revised form 29 March 2001; accepted 4 July 2001.

[5] Marco Tomassini a, Moshe Sipper b, Mosé Zolla a, Mathieu Perrenoud a a, “Generating high-quality random numbers in parallel by cellular automata” Institute of Computer Science, University of Lausanne, 1015 Lausanne, Switzerland Logic Systems Laboratory, Swiss Federal Institute of Technology, CH-1015 Lausanne, Switzerland Accepted 17 March 1999


