12

Deeper Investigating Adequate Secret Key Specifications for a Variable Length Cryptographic Cellular Automata Based Model

Gina M. B. Oliveira, Luiz G. A. Martins and Leonardo S. Alt
Universidade Federal de Uberlândia
Brazil

1. Introduction

Cellular automata (CA) are particularly well suited for cryptographic application and there are several previous studies in this topic (Benkiniouar & Benmohamed, 2004; Gutowitz, 1995; Kari, 1992; Nandi et al., 1994; Oliveira et al., 2004; 2008; Seredynski et al., 2003; Tomassini & Perrenoud, 2000; Wuensche, 2008; Wolfram, 1986). Since CA rule is simple, local and discrete, it can be executed in easily-constructed massively-parallel hardware at fast speeds. Basically, the CA-based cryptographic models can be divided into three classes: (*i*) models that use CA to generate binary sequences with good pseudo-random properties, which are used as cryptographic keys, but the effective ciphering process is made by another function (Benkiniouar & Benmohamed, 2004; Seredynski et al., 2003; Tomassini & Perrenoud, 2000; Wolfram, 1986); (*ii*) models based on additive, non-homogeneous and reversible CA, that use algebraic properties of this kind of rules to generate automata of maximum and/or known cycle (Kari, 1992; Nandi et al., 1994); and (*iii*) models based on irreversible CA, which uses the backward interaction of cellular automata in the ciphering process and the forward interaction to decipher (Gutowitz, 1995; Oliveira et al., 2004; 2008; 2010a; Wuensche, 2008), as the cryptographic model discussed here.

Gutowitz has previously proposed a cryptographic model based on backward evolution of irreversible CA (Gutowitz, 1995). A toggle CA rule transition is used as the secret key in his model. A pre-image of an arbitrary lattice is calculated adding extra bits in each side of the lattice. This increment is pointed as the major flaw in the model. CA backward interaction is also known as pre-image computation and an efficient reverse algorithm was proposed by Wuensche and Lesser (Wuensche & Lesser, 1992) for a periodic boundary condition, keeping the pre-image with the same size of the image. Such algorithm was evaluated as encryption method in (Oliveira et al., 2008). However, its usage has the disadvantage that there is no guarantee of pre-image existence for any given lattice and any given rule. The only rules with assurance of pre-image existence are not appropriate for ciphering because they do not exhibit a chaotic dynamics.

Thus, a new approach was proposed, which alternates the original reverse algorithm and the variation that uses extra bits, using the second only when the pre-image computation fails. This variation is similar to pre-image computation adopted in Gutowitz model (Gutowitz, 1995). Although this approach needs to add bits to the ciphertext when a failure occurs, it is expected that in practice few failures happen and the ciphertext length will be equal or close to the plaintext. In the resultant method, encryption always succeeds and the final length of the ciphertext is not fixed. This method was named Variable-Length Encryption Method (VLE) (Oliveira et al., 2010a).

Initial experiments were performed using small sets of radius 2 and radius 3 toggle rules. Subsequent experiments were performed using a representative rule set formed by all radius 2 right-toggle rules, totalizing 65536 rules. These rules represent 50% of the possible secret keys in radius 2 space, being that the other 50% are the all radius 2 left-toggle rules, which are dynamically equivalent to the set analyzed. Based on an exhaustive analysis of this rule space we concluded that, considering a cryptographic purpose, there are a lot of undesirable behavior rules in the complete set that must be avoided as secret keys; they represent approximate 5% of the rule space investigated.

It was employed an analysis based on several CA static parameters, trying to capture a pattern associating such parameters to underperforming rules. Static parameters like Z, Sensitivity, Absolute Activity, Activity Propagation and Neighborhood Dominance (Oliveira et al., 2001) were investigated to capture the pattern associated to underperforming rules. A database was generated associating rules performance in VLE ciphering with their parameters. A genetic algorithm-based data mining was performed to discover adequate key specifications based on CA parameters. We deeper investigate these specifications using them as to filter the set of radius 2 rules as to elaborate new radius 3 rules. By applying such methodology it were able to discover good secret key specifications for VLE. Using such specifications, ciphertext length is short, encryption process returns high entropy and VLE has a good protection against differential cryptanalysis.

This chapter is organized as follows. Section 2 presents some concepts related to cellular automata with emphasis on forecast behaviour CA parameters. Section 3 reviews some previous CA models related to the cryptographyc model discussed here. Section 4 details the major steps of VLE cryptographic model. Section 5 presents results of experiments performed to evaluate VLE´s ciphering quality and to find a good specification of toggle rules to be used as secret keys. Section 6 presents the major conclusions.

2. Cellular automata definitions

Cellular automata (CA) are discrete complex systems that possess both a dynamic and a computational nature. A cellular automaton consists of two parts: the cellular space and the transition rule. Cellular space is a regular lattice of N cells subjected to boundary conditions. A state is associated to each cell at time t. The transition rule τ yields the next state for each cell as a function of its neighbourhood. At each time step, all cells synchronously update their states according to τ. For one-dimensional (1D) CA, the neighbourhood size m is usually written as $m = 2R + 1$, where R is CA radius. In binary-state CA, the transition rule τ is given by a state transition table which lists each possible neighbourhood together with its output bit, that is, the updated value for the state of the central cell.

A special kind of CA rule is used as the secret key in the cryptographic model discussed here: they are toggle transition rules. A CA transition table is said to be a toggle rule if it is sensible in respect to a specific neighborhood cell, that is, if any modification of the state on this cell necessarily provokes a modification on the new state of the central cell, considering all possible rule neighborhoods. Considering one-dimensional radius-1 CA, the neighborhood is formed by three cells and they can be sensible in respect to the left cell, to the right cell, and to

the central cell. For example, elementary rules 00101101, 10011010, 10010110 are left-toggle, right-toggle and left-and-right-toggle rules, respectively.
The dynamics of a cellular automaton is associated with its transition rule. In order to help forecast the dynamic behavior of CA, several parameters have been proposed in the literature, as the precursor lambda parameter (Langton, 1990). Some forecast parameters had shown to be important in the analysis of the key space related to the cryptographic model discussed here. They are briefly discussed following.

2.1 Z

The definition of Z parameter derived from the pre-image calculus algorithm and it is composed by Z_{left} and Z_{right}. Let us assume that a part of a pre-image of an arbitrary lattice configuration is known and that we want to infer the missing cell states, successively, from left to right. Z_{left} is defined as the probability that the next cell to the right in the partial pre-image has a unique value, and it is directly calculated from the transition table, by counting the deterministic neighborhoods. Z_{right} is the converse, from right to left. Z parameter is the greater of Z_{left} and Z_{right}. A detailed explanation about this parameter is given in (Wuensche, 1998). In (Oliveira et al., 2001) an analysis of Z parameter is presented and the main conclusions are: (i) it is good discriminators of the chaotic behavior (ii) rules with a Z value close to 1 have high probability to be chaotic. The components Z_{left} and Z_{right} have shown very important in the specification of the rule transitions used as secret keys in the method discussed in (Oliveira et al., 2008).

2.2 Symmetry

During the evaluation of reverse algorithm described in (Oliveira et al., 2008), a characteristic that have shown important to secret key specification was the symmetry of the output binary sequence representing a transition rule. Let m be the number of cells considered in the neighborhood of a binary cellular automaton and the rule transition defined by k output bits ($k = 2^m$): $b_0, b_1, ..., b_{k-1}$. The symmetry level (S) is the number of pair of bits b_i and b_{k-i-1} ($0 \leq i \leq k/2 - 1$) that have the same value. It can be normalized divided by the total number of pairs ($k/2$). For example, considering radius-1 CA rules, the symmetry level of rules 10111101, 10110010 and 10001111 is 1, 0 and 0.25, respectively. The importance of this parameter in the specification of CA rules suitable for encryption was first noted due to the low number of lattices with at least one pre-image when rules with S equal to 1 are used. That is, the secret key can not be a palindrome. This parameter has also demonstrated to be relevant in another classical problem in the context of cellular automata field: to find rules able to classify the density of 1s of a given lattice, the density classification task (DCT). Such kind of pattern was observed when analyzing good rules suitable to perform DCT (de Oliveira et al., 2006). Recently, when working with the equivalent dynamical transformations of a CA rule (reflection, complementary and complementary-plus-reflection) we perceived that this symmetry is directly related to the bits used to perform the complementary transformation. In this sense, a rule with $S = 0$ does not have a complementary equivalent rule, because when we apply the transformation it returns to the same rule. For example, rule 01101001 of elementary space. Conversely, rules with $S = 1$ has a complementary rule formed by the complement of the output bits. For example, equivalent elementary rules 01100110 and 10011001.

2.3 Absolute activity

This parameter came from the analysis of lambda parameter (or Activity) Langton (1990); it measures only the rule's activity level counting how many transitions lead to state 1, regardless of the states of the neighbors. The goal of Absolute Activity parameter was to improve the measure of activity verifying the transitions that lead to a different state of the central cell's state or it's neighbors (Oliveira et al., 2001). It quantifies how much change is entailed by the rule, in the state of the central cell, in relation to the current state of the central cell, and the states of the pair of cells which are equally apart from the centre (Oliveira et al., 2001).

2.4 Neighborhood dominance

It verifies whether the new value of the central cell "follows" the state that appears the most in the neighborhood (Oliveira et al., 2001). For example, in transition 100 $\rightarrow$ 1 there is no neighborhood dominance, although the central cell changes. Besides, a relative weight is associated to each neighborhood in such way that greater is the homogeneity, greater is the weight of dominance associated to this neigborhood. For example, in elementary space, the neighborhoods 000 and 111 have weigth 3, and the others, 1 (Oliveira et al., 2001).

2.5 Core entropy

Let τ be the CA transition rule, R the radius, $m = 2R + 1$ the neighborhood size and $n = 2^m$ the number of neighborhoods (size of the rule). In a toggle rule, we know that if the neighborhood T_i has output X, for $0 \leq i < n$:

- If it is right toggle:
  - If i is even, $T_{i+1} \rightarrow \overline{X}$;
  - If i is odd, $T_{i-1} \rightarrow \overline{X}$;
- If it is left toggle:
  - If i is $< n/2$, $T_{i+n/2} \rightarrow \overline{X}$;
  - If i is $\geq n/2$, $T_{i-n/2} \rightarrow \overline{X}$;

So, if we know half of the rule, we can generate the rule. A default rule generator is the sequence of the cells in even positions if the rule is right toggle, and the first half of the rule, if it is left toggle. This generator rule is called rule core. The Core Entropy is the spatial entropy associated to this core (Oliveira et al., 2010b).

3. Previous CA-based cryptographic models

A classification scheme of the previous approaches used in CA-based cryptographic models has been presented in the introductory section of this work. In the present section the methods which use irreversible CA are revised. In the first sub-section we revise the methods that use toggle rules, which return a strong and fixed increase in the ciphertext length in relation to the original plaintext. In the second sub-section we revise the methods that investigated the employment of the reverse algorithm proposed by (Wuensche & Lesser, 1992) as a ciphering process aiming to keep the ciphertext in the same length of the related plaintext. A comparative analysis of two similar proposes described in (Oliveira et al., 2008) and (Wuensche, 2008) are also presented.

3.1 Toggle rules-based methods

Gutowitz proposed and patented the first cryptographic model based on backward evolution of irreversible CA (Gutowitz, 1995). His model uses toggle transition rules (section 2) as secret keys. Gutowitz used irreversible CA with toggle rules - either to the right or to the left - for encrypting. Toggle rules turn possible to calculate a pre-image of any lattice starting from a random partial pre-image. Using this method, a pre-image of any lattice is calculated adding R extra bits in each side of the lattice, where R is the radius of the rule. Consider an initial lattice of N cells and a right-toggle transition rule with radius 1: the pre-image will have $N + 2$ cells. Suppose that two initial bits X and Y are randomly chosen to start the pre-image calculation and they are positioned in the left border of the lattice. The state of the next right cell (the third one) is deterministically obtained because the only two possible transitions are $XY0 \rightarrow T$ and $XY1 \rightarrow \overline{T}$. One can check the value of first cell in the initial lattice to see if it is T or $\overline{T}$ and to determine if the third cell in the pre-image is 0 or 1. Once the state of the third cell is determined, the next step is to determine the state of the forth cell and the other cells successively up to completing the entire pre-image. The right-toggle property of the transition rule guarantees that the entire $N + 2$ pre-image cells can be obtained, step-by-step, in a deterministic way. The same process can be done for left-toggle rules and in this case the initial cells must be positioned in the rightmost side. A fundamental characteristic in Gutowitz's model to guarantee pre-image for any lattice is the usage of a non periodic boundary condition. When the pre-image is obtained the extra bits are not discarded and they are incorporated to the lattice. Therefore, it is always possible to obtain a pre-image for any initial choice of bits. As the length of initial bits is $2R$, it is possible to obtain 2^{2R} different pre-images for each lattice. Therefore, the cellular automaton is irreversible. This pre-image calculus is specific to the ciphering method and it can be used only when applying toggle rules.

A toggle transition rule τ is used as the secret key in Gutowitz's cryptographic model. The plaintext is the initial lattice and P pre-images are successively calculated, starting with random initial bits ($2R$ cells) at each step. The ciphertext is given by the last pre-image obtained after P steps. The decryption process is based on the fact that the receptor agent knows both τ and P. By starting from the ciphertext the receptor just needs to apply the transition rule τ forward by P steps and the final lattice will be the plaintext. Let P be the number of pre-image steps, N the length of the original lattice and R the CA radius. The method adds $2R$ bits to each pre-image calculated and the size of the final lattice is given by $N + 2RP$. For example, if 50 pre-images were calculated using a radius-2 rule starting from a plaintext of 128 bits, the size of the ciphertext would be 328. Clearly, it is not a negligible increment and it is pointed as the major flaw in Gutowitz's model.

A high degree of similarity between ciphertexts when the plaintext is submitted to a little perturbation was identified as another flaw in Gutowitz's model. The original model was altered by using bidirectional toggle CA rules (to the right and to the left simultaneously) in (Oliveira et al., 2004). The experiments with this model show that the similarity flaw was solved with such a modification and it is protected against differential cryptanalysis (Oliveira et al., 2004). However, the ciphertext increase in relation to the plaintext length remains in this model.

3.2 Reverse algorithm-based methods

An algorithm for a generic pre-image computation was proposed in (Wuensche & Lesser, 1992). This algorithm is known as reverse algorithm and it is based on the idea of partial neighborhoods. The complete description of this method can be found in (Wuensche & Lesser, 1992) and a summary in (Oliveira et al., 2008). Before starting the calculus, R cells are added to each side of the lattice corresponding to the pre-image, where R is the CA radius and the algorithm starts the calculus of the pre-image, randomly initializing the $2R$ leftmost cells. This procedure is similar to the start of the pre-image calculus used in Gutowitz's model. However, in a periodic CA boundary condition, the last cells need to be validated. The pre-image calculus is concluded verifying if the initial bits can be equal to the final $2R$ rightmost ones. If so, the extra bits added to each neighborhood side are discarded returning the pre-image to the same size of the original lattice. If no, the process returns again to use the last option stored in a stack, which contains the not evaluated possible values until this point of computation. The reverse algorithm finds all the possible pre-images for any arbitrary lattice and any arbitrary rule transition.

The reverse algorithm was evaluated as an encryption method in (Oliveira et al., 2008). An advantage of this method is that it can be applied to any kind of CA rule. So, it is not a specialized method as the one used in Gutowitz's model that works only with toggle rules. However, its application as a ciphering process has the disadvantage that there is no guarantee of pre-image existence for any given lattice and any given rule transition. On the other hand, it is well-known that for a fixed lattice the large majority of cellular automata rule space has a lot of lattice configurations with no pre-images; they are known as Garden-of-Eden states (Wuensche & Lesser, 1992). Therefore, the major challenge to apply the reverse algorithm as a viable cipher method is to find a manner to guarantee the existence of at least one pre-image for any possible initial lattice representing a possible plaintext to be encrypted.

The first attempt to solve this problem was to use Z parameter (Wuensche, 1998) in rule specification (Macêdo et al., 2008). It was already known that as higher is the Z value associated to a CA transition rule as higher is the probability of the cellular automation exhibits a chaotic behavior (Oliveira et al., 2001) and as higher is the probability of a pre-image existence for any lattice (Wuensche, 1998). Initially, the set of CA rules with Z equal to 1 was chosen as potential rules to use as secret keys in a cryptographic model based on the reverse algorithm. However, the first analysis of some simple $Z = 1$ rules (elementary rules and some other radius 2 binary rules manually constructed) have shown that this specification would not be enough to guarantee of pre-image existence. When this first $Z = 1$ set of rules was applied to encrypt random texts with usually cipher-block sizes (32, 64, 128 bits), it was not possible to cipher a lot of tested initial lattices. Aiming to discover the ideal specification for a 100% of pre-image existence rule, a simple genetic algorithm (GA) was implemented to find CA rules with a desirable characteristic (for example, as specific value of Z parameter). The implemented GA aids the process of generating rule sets to be evaluated as secret keys but it is not properly a part of the cipher method. Using GA it was possible to evaluate samples of rules with different specifications and several parameters related to the CA context have been evaluated as the dynamical forecast parameters Z (Wuensche, 1998), lambda (Langton, 1990), sensitivity (Binder, 1994), neighborhood dominance (Oliveira et al., 2001) and so on. The parameters that have presented more dependence to the 100% of pre-image existence problem was the components of Z known as Z_{right} and Z_{left} and the symmetry level parameter S explained in section 2. When a set of rules generated by GA with specification Z equal to 1 were applied to cipher some lattices samples with fixed length (32, 64 and 128 bits), a wide range of performance was obtained, from 0% to 100% of encrypted lattices. However, a clear characteristic have emerged from the analysis of these rules: the worst performance was obtained applying $Z_{left} = Z_{right} = 1$ rules and $S = 1$ rules.

Using this information, the set of $Z = 1$ rules could be improved, avoiding these characteristics. Due to Z definition (Wuensche, 1998), considering a rule with Z equal to 1, either Z_{right} or Z_{left} necessarily must be equal to 1. However, the other component is independent and can assume a value from 0 to 1. As we mentioned before, one component must be equal to 1 and the other must be different of 1 to avoid the worst performance. On the other hand, the symmetry level equal to 1 also has to be avoided. Following this observation, a set of rules was generated with the opposite characterization: rules with a low level of symmetry (< 0.25) and with one of the components of Z (Z_{left} or Z_{right}) equal to 1 and the other with a low value (< 0.25). Some initial experiments with this set have return in a first moment a mistaken conclusion that the "ideal rule specification" for a 100% of pre-image existence was found since they returned 100% of performance when tested in several samples of 32, 64 and 128 bits (Macêdo et al., 2008). However, an undesirable characteristic has been latter discovered when the dynamical behavior of some rules of this "ideal" set were analyzed: they are not chaotic rules as expected for $Z = 1$ rules; conversely, they are fixed-point behavior rules (with a spatial shift) (Oliveira et al., 2008) Some experiments using an entropy measure to characterize the difference between two ciphertexts obtained encrypting to very similar plaintexts have confirmed this observation for the majority of the rule set, because the associated entropy of this ciphering was very low. So, they cannot be used as secret keys since they are not able hide the original information, a primordial pre-requisite for any encryption process (Oliveira et al., 2008). Therefore, a trade-off was established when specifying a transition rule to be used with the reverse algorithm: if the rule is perfect in respect to the existence of pre-images, it does not have a chaotic behavior; if the rule is perfect in respect to the chaoticity, it can not be able to calculate the pre-image for a large range of possible plaintexts.

A new round of experiments were performed in (Oliveira et al., 2008) using $Z = 1$ rules with an intermediate level both for symmetry and Z_{left}/Z_{right} balance: $0.25 < S < 0.5$ and $0.25 < Z_{left} < 0.5$ and $Z_{right} = 1$. A sample of 100 radius-2 CA rules were evaluated calculating 128 consecutive pre-images starting from 1000 random lattices of 512 bits. All the rules were able to calculate the 128 consecutive pre-images for all 512-bits lattices. Besides, the average of the mean entropy obtained for all the rules was high (0.8857) indicating that they exhibit a chaotic behavior. An important final observation is that although it was possible to specify rules with a high probability to find at least one pre-image for any lattice and with a good perturbation spread, even the better rules evaluated can fail when the pre-image computation is applied to some lattice.

The major conclusion of the analysis in (Oliveira et al., 2008) is that the simple adoption of the reverse algorithm is not recommended because the possible rules with 100% guarantee of pre-image existence are not appropriate for ciphering. Two alternative approaches have been emerged. The first is the application of a variation of the reverse algorithm in which extra bits are added to the lattice in each cipher step, similar to the process adopted in Gutowitz's cryptographic model. Using this approach the 100% guarantee of pre-image existence was obtained with a good level of entropy. However, this approach is affected by the same disadvantage of Gutowitz's model: a considerable and fixed increase in the ciphertext in relation to the original one. The second propose in (Oliveira et al., 2008) is a method based on the original reverse algorithm adopting a contour procedure to apply when the pre-image calculus fail. As the secret key specification previous discussed gives a low probability to this failure occurrence, we expect to rarely use this contour procedure but it guarantees the possibility to cipher any plaintext. This method, which is the key point of the present work, alternates the original reverse algorithm and the variation that uses extra bits in the ciphering process, using the second only when the pre-image calculus fails. Although this approach needs to add some bits to the ciphertext when a failure occurs, it is expected that in practice few failures happens and the ciphertext length will be equal or close to the plaintext. The general idea of this second approach was proposed in (Oliveira et al., 2008) but it has been redefined, implemented and tested first in (Oliveira et al., 2010a). Sections 4 and 5 details the method and presents experiments performed to evaluate and to improve its and the properly specification of the secret keys to have a reasonable final length.

In a certain sense, the method proposed in (Wuensche, 2008) is very similar to the initial method proposed in (Oliveira et al., 2008). It is also based on the employment of the reverse algorithm to encrypt a plaintext through successive pre-image computations, obtaining a lattice correspondent to the ciphertext. Deciphering method is performed by CA forward evolution. Therefore, in essence the idea is the same one applied in the original method described in (Oliveira et al., 2008), even so the methods have been proposed in an independent way. The problem of finding rules with 100% guarantee of pre-image existence was also addressed in (Wuensche, 2008). Aiming to guarantee a good performance of the algorithm in the ciphering process, the author indicates that keys must be chain-rules with $Z_{left} = 1$ and $0.5 < Z_{right} < 1$. The treatment given to failure occurrences when performing pre-image calculus is an important point to discern the two works in (Oliveira et al., 2008) and (Wuensche, 2008). The author in (Wuensche, 2008) said that: "*for big binary systems, like 1600, the state-space is so huge,* 2^{1600}*, that to stumble on an unencryptable state would be very unlikely, but if it were to happen, simply construct a different chain rule*". However, there is a practical viability of any cipher method only if this method assures the encryption of any plaintext and that the suggested secret key discarding in the case of failure cannot be adopted in a communication system. Moreover, even with the adoption of lattices with high length (as 1600 bits suggested in (Wuensche, 2008)) the use of rules with $Z_{right}/Z_{left} > 0.5$ do not avoid the occurrence of rules with a high number of Garden-of-Eden states. Systematic experiments were not presented in (Wuensche, 2008) to evaluate this possibility. We performed several experiments using different groups of rules with radius 3 and $Z_{left} = 1$. Approximate 200 rules have $0.75 < Z_{right} < 1$ and some of them do not return a good performance attempting to calculate 20 consecutive pre-images. Therefore, even using a large lattice and a rule within the specification suggested in (Wuensche, 2008), it is clear that a failure can happen during the ciphering process. As example of rule of such undesirable performance we can point rule 569A99965999596AA965666AA666A699. Thus, the alternative method discussed in (Oliveira et al., 2008) to deal with situations in which the reverse algorithm did not obtain the pre-image is a necessary approach. It is the key point of the method investigated in this work.

4. Variable length encryption method

Since the main conclusion of the analysis in (Oliveira et al., 2008) is that the simple adoption of the reverse algorithm is not possible, an alternative method was investigated in (Oliveira et al., 2010a). It is based on reverse algorithm adopting an alternative procedure to apply when the pre-image computation fails (Oliveira et al., 2008). It is expected that with an appropriate key specification there is a low probability to this failure occurrence. The alternative procedure adds extra bits only when the pre-image is not possible to calculate. Therefore, it is expected to rarely use this procedure but it guarantees the possibility to cipher any plaintext. For practical reasons related to encryption speed, it can be better to limit the method to operate with only toggle rules. The method works as it alternates rounds of pre-image computation performed by reverse algorithm (a variation of Gutowitz's model for periodic conditions) with few or none steps of pre-image computation performed by Gutowitz's model. Ciphering is made by computing P consecutive pre-images starting from a lattice of size N corresponding to the plaintext. The secret key is a radius-R CA rule τ generated with an appropriate specification based CA static parameters.

Suppose that it started to calculate pre-images using reverse algorithm and the secret key τ and it fails in the *K-th* pre-image such that $K \leq P$. In such situation the ciphering process uses the modified reverse algorithm with extra bits to calculate the *K-th* pre-image. Thus, the *K-th* pre-image will have $N + 2R$ cells. Ciphering returns again using the original reverse algorithm to calculate the remaining pre-images. If all the subsequent pre-images computation succeeds the final ciphertext will have a size of $N + 2R$. If the pre-image computation fails again, the ciphering process changes and adds $2R$ more bits to the lattice. If the process fails in F pre-images ($F \leq P$) the final lattice will have $N + 2FR$. Starting from a lattice of N cells, the size of the ciphertext after P pre-images computation is given by $N \leq \textit{ciphertext size} \leq N + 2PR$. Therefore, it is a variable-length encryption model, named *VLE*. However we expected that in practice the ciphertext size will be next to N due to the characteristics of the rule used as secrete key. It is important to note that the ciphertext obtained using Gutowitz's model will have exactly $N + 2PR$ bits – the worst and improbable situation in the proposed method. For example, if $N = 512$, $P = 64$ and $R = 5$ the size of the ciphertext using VLE will be situated between 512 and 1152 bits. However, we expected that in practice the ciphertext size will be next to 512 due to the characteristics of the rule used as secret key. On the other hand, the ciphertext obtained using Gutowitz model will be exactly 1152 bits.

Deciphering will be executed applying the forward interaction of cellular automata rules. By starting from the ciphertext the recipient needs to apply the transition rule τ forward by P steps and the final lattice will be the plaintext. He also needs to know in which pre-images failures happened to recover the original text. An improvement of the method in relation to the one proposed in (Oliveira et al., 2008) is the usage of a non-retroceding method in a case of failure. Tests performed in (Oliveira et al., 2008) have used the following retroceding strategy: when calculating a sequence of 10 consecutive pre-images to cipher a plaintext, suppose that a failure occurs in the 8th pre-image, for example. In this case, the algorithm returns to the last lattice in which there is a possibility to have another pre-image. So, when we try to calculate 10 consecutive pre-images starting from a given lattice and this process fail in the end we can affirm that all the possible backward trajectories were evaluated and it is really impossible to find 10 consecutive pre-images starting from the given lattice. Using this retroceding procedure the number of lattices possible to be ciphered by the original reverse algorithm is increased. However, during the experiments, we perceived that this procedure is high-time consuming as expected and with low actual return to the final performance of the rules. Besides, with the increase of the block size, this inefficient behavior is more probable. So for the simplicity and the speedy of the method, we decided to implement the variable-length ciphertext method using a non-retroceding method. When the pre-image computation fail, the method changes to the computation using extra bits without trying to backtrack to find another pre-image with lower order.

CA backward interaction-based ciphering method is more adjusted to encryption as a block-cipher method using a restricted lattice length instead of encryption as a stream cipher method with an arbitrarily large lattice. This argument is motivated by the fact that the pre-image computation is an algorithm of considerable computational cost. A major advantage of the pre-image calculus adopted is that the process never retrocedes when a pre-image is calculating and it always succeeds, although there is an increase of cells during the process. Using a parallel hardware in Gutowitz's model, it is possible to calculate simultaneously different pre-images to reduce significantly the final time of processing. Unfortunately, using the reverse algorithm with periodic boundary conditions this kind of parallelism is not possible. Using the simplification to apply only toggle rules (as in Gutowitz's model), it speeds up the pre-image computation, since that will be removed the ambiguities and there is no possibility of the algorithm to come back in the computation before arriving at the end of the lattice. However, various pre-image initializations are possible and typically only one will succeed. Using a stream cipher approach, since reverse calculus is essentially sequential, the idea to parallel the calculus of a unique pre-image starting from a specific initialization is probably unfeasible. On the other hand, if a block cipher approach is used, a parallel architecture can be used to perform the blocks ciphering in a distributed way increasing the throughput of the entire encryption method. Therefore, we claim that the employment of the reverse algorithm-based ciphering is more adequate as a block-cipher method and we suggested the following block sizes: 256 or 512 bits. Although the reverse algorithm-based block-cipher method can be applied using any operation mode already investigated in the literature for other block-cipher method (ECB, CBC, PCBC, OFB, CFB or CTR) (Stallings, 2003), we strongly suggest the use of counter operation mode (CTR) to increase the security and the throughput of the entire encryption process.

5. Experiments

Using VLE we have the guarantee that ciphering is possible even if an unexpected Garden-of-Eden state occurs. However a short length ciphertext depends on the secret key specification. Some initial experiments were performed to analyze method's performance and to evaluate rules specification proposed in (Oliveira et al., 2008) and (Wuensche, 2008). In these experiments small groups of radius 2 and radius 3 rules were used. Section 5.1 presents these experiments. Subsequently, a deeper investigation about an appropriate rule specification using a more representative key set was carried out. Aiming to perform a more exhaustive analysis, some experiments were conducted using the complete set of radius 2 right-toggle rules: it was used all possible radius 2 transition rules with $Z_{left} = 1$. Sections 5.2 and 5.3 report these experiments.

5.1 Small sets of radius 2 and 3 rules

A computational environment was implemented to analyze method's performance and to evaluate rules specification, starting from the information presented in (Oliveira et al., 2008) which uses Z_{left}, Z_{right} and S. The extreme parameters values ($Z_{right} \leq 0.05$, $Z_{right} \geq 0.95$, $S \leq 0.05$ and $S \geq 0.95$) were excluded due to rules in these ranges present low performance in relation to either pre-images existence or entropy. Two groups of rules with $Z_{left} = 1$ were established, each one containing 500 rules for each radius analyzed (2 and 3). The first group was formed by rules with Z_{right} specified in the range $0.25 < Z_{right} \leq 0.5$ as suggested in (Oliveira et al., 2008); it was named Group 0. The second group was formed by rules in the range $0.5 < Z_{right} < 0.95$ as suggested in (Wuensche, 2008); it was named Group 1. Each group was used in the ciphering process using samples of 256 bits plaintexts: 100 initial lattices randomly generated with a Gaussian distribution. The number of pre-images steps (P) was defined according to the rule radius (R) (considering the block size of 256 bits): it was used $P = 128$ for radius 2 rules and $P = 85$ for radius 3 ones. The results for each group of rules are presented in Table 1. The objectives of this investigation are: (i) to determine which is the average length of the final lattices (ciphertexts), related to the average number of failures occurred during the ciphering; and (*ii*) to determine which is the difference associated to two ciphertexts obtained starting from two very similar plaintexts, to evaluate the encryption quality and specially its protection against a differential cryptanalysis-like attack.

(i) Calculating ciphertexts final length: applying the method described in Section 4, any rule with $Z_{left} = 1$ is able to complete the ciphering process starting from any initial lattice, independently of the number of pre-image previously established. However, the final length of the ciphertext can be between N (best case) and $N + 2PR$ (worst case). We want to evaluate if the expected final length is in fact close to the best case and which is the behavior of each group in such evaluation. Table 1 presents group performances: the average length of the final lattice or ciphertext (L_{mean}) and the average number of failures occurred during ciphering process (F_{mean}). Table 1 shows L_{mean} and F_{mean} for each group of rules and for each radius size R. Each average was computed considering 50000 tests: 500 rules applied over 100 initial lattices.

(ii) Comparing ciphertexts generated by pairs of similar plaintexts: cryptanalysis methods try to find the plaintext after getting the ciphertext without knowing the secret key. The differential cryptanalysis is based on the analysis of some pairs of ciphertexts generated by similar plaintexts. Although the origins of differential cryptanalysis are related to studies in how to break ciphertexts encrypted by DES algorithm (Biham & Shamir, 1991), Sen et al. (2002) have used the same idea to analyze their CA cryptosystem named CAC and they compared their results with the results obtained with DES and AES cryptosystem (Sen et al., 2002). In this analysis, several pairs of plaintext (X, X') are used, which differ one of the other by a fixed and small difference D. Each pair (X, X') is used to generate a pair of ciphertexts (Y, Y') which differ one of the other by a difference D'. D and D' are obtained by applying the XOR operations between the pair members. For each pair (Y, Y'), the number of 1s in D' is counted, which corresponds to the number of different bits between Y and Y', and the standard deviation of this measure is calculated over all the analyzed pairs. As higher the deviation standard in D', as higher is the probability of the ciphertext be broken by differential cryptanalysis. An algorithm with standard deviation below 10% is said to be protected against differential cryptanalysis (Sen et al., 2002). Difference D between two plaintexts X and X' was fixed in only one bit in any arbitrary position over the lattice (single bit perturbation) and the value of D' was determined for each plaintext evaluated. D' was calculated to each pair (Y,Y') to obtain the standard deviation (σ) for each group set. The total number of tests is 200000: 50000 tests for each group and for each fixed radius. A standard deviation of 4.47% was obtained considering all the 200000 tests. Therefore, the proposed CA cryptographic model can be considered secure in relation to differential cryptanalysis. Considering groups 0 and 1, we obtained 4.81% and 4.13%, respectively. Table 1 shows the standard deviation (σ) for each group of rules (0 or 1) and for each radius size (2 or 3).

Additionally, to ensure that the final difference D' obtained between Y and Y' generated from two similar plaintexts X and X' does not keep any pattern which eventually could help a cryptanalyst using a differential cryptanalysis-like attack, a second measure was applied: we calculated the entropy associated to each D'. A measure of spatial entropy was applied on D' to evaluate the existence of some undesirable regularity on this difference. Entropy above 0.75 assures a random difference enough to affirm that ciphertexts Y and Y' do not maintain any similarity, even so they started from similar plaintexts. Entropy below 0.5 indicates a strong pattern in difference D'. Entropy values between 0.5 and 0.75 had been considered fuzzy, since it cannot guarantee the existence of an ordered or random pattern. Therefore, if any cryptography method is applied to similar texts returning an average of spatial entropy (E_{mean}) above 0.75 (with a low standard deviation), it indicates a good protection to differential cryptanalysis. Furthermore, it assures that the ciphering adds a high level of entropy during the process, a necessary characteristic in any encryption method.

R	N	P	Group	Z_{right}	L_{mean}	F_{mean}	E_{mean}	$\sigma_{mean}(\%)$
2	256	128	0	$0.25 < Z_{right} \leq 0.5$	256.19	0.047	0.89	5.29
			1	$0.5 < Z_{right} < 0.95$	269.80	3.449	0.87	4.32
3	256	85	0	$0.25 < Z_{right} \leq 0.5$	267.07	1.845	0.87	4.33
			1	$0.5 < Z_{right} < 0.95$	325.50	11.583	0.85	3.94

Table 1. Mean values obtained for all rules and for the 500 worst rules in each set in first experiment.

Looking over radius 2 rules in Table 1, rules of group 1 ($Z_{right} > 0.5$) have high entropy and a low standard deviation, confirming that these rules quickly spread any perturbation. However, they produce a significant increase in lattice size, confirming that these rules have high probability of failures during pre-image computation. The analysis of radius 2 rules indicated that group 0 presented the best results, i.e. the rules with Z_{right} between 0.25 and 0.5 also have high entropy and low standard deviation in the ciphertext pairs analysis and the number of expected failures in pre-images is very low (< 0.5), turning the texts ciphered starting from plaintexts with 256 bits very close to the original size. Analyzing the results using the radius 3 rules, one can observe that although the rules still presented good measures related to the pairs of ciphertexts analysis (high entropy and low standard deviation), the final size of the ciphertexts tend to increase when comparing with radius 2 rules, even that the number of pre-images used is smaller when using radius 3 rules. Rules with Z_{right} between 0.25 and 0.5 (Group 0) presented again the best performance in relation to the average number of failures. However, the number of expected failures is above 1 and the expected final length is almost 5% longer than the original lattice size: 267.07. Group 1 presented a high average number of failures (above 11 failures in 85 pre-image steps), returning ciphertexts 27% longer than the original size.

Therefore, Group 0 presented the best performance, despite having increased the average number of failures from radius 2 to 3; the number of failures is acceptable for both radius and the final lattices have an average size close the minimum. They represent the rules with $0.25 < Z_{right} \leq 0.5$ as suggested in (Oliveira et al., 2008). Group 1 is formed by the rules with $0.5 < Z_{right} \leq 0.95$; they represent the rules with $Z_{right} > 0.5$ as suggested in (Wuensche, 2008) and presented a non-satisfactory performance: although they returned good entropy values, they also returned the largest values of ciphertext length. Therefore, they were considered inappropriate to use in a block-ciphering system, at least in the block size investigated here: 256 bits. Similar experiments were performed in (Oliveira et al., 2010a) with other block sizes: as the block size increases, the adequacy of these rules also tends to increase in both groups. However, as a general conclusion, rules using Group 0 specification returns better performance than rules like Group 1. Despite of this comparative analysis between groups'specification, our expectative about the usage of the variable length method had been confirmed: it has a good quality of ciphering entropy and the ciphertext length is close to the original block size, specially using rules specified according to (Oliveira et al., 2008). However, a lot of open questions remained since the experiments were performed based on very limited samples of rules.

5.2 Investigating the complete radius 2 rule set with a variable number of pre-image steps

Subsequent experiments were conducted using the complete set of radius 2 right-toggle rules; all of them have $Z_{left} = 1$ and $0 \leq Z_{right} \leq 1$. Radius 2 toggle rules are defined by only 16 bits since the other 16 bits are deterministically defined. This set is composed by 65536 (2^{16}) rules, being that all of them have Z_{right} =1. These rules represent 50% of the possible keys in radius 2 space (restricted to use only toggle rules for faster encryption), being that the other 50% are the left-toggle rules ($Z_{left} = 1$), which are dynamically equivalent to the set analyzed. We employed the VLE environment to cipher one hundred 256-bits plaintexts using each right-toggle rule of the complete set. In previous experiments described in Section 5.1 the number of consecutive pre-images employed during ciphering was fixed: $P = 128$ for radius 2 rules and 256-bits plaintexts. Here, the number of consecutive pre-image steps was dynamically defined between 16 and 128, depending on the results obtained during ciphering. In that way, the ideal number of consecutive pre-images employed during ciphering was also experimentally investigated. Based on an exhaustive analysis of radius 2 right-toggle rule space we analyze the effects of using as secret keys the following sets of rules: *(i) Fullset* - the complete set of rules in which the unique restriction is the right-toggle property ($Z_{left} = 1$ is a consequence); *(ii)* $Subset_A$ - the restricted rule set defined by the specification $0.25 < Z_{right} \leq 0.5$ and $S = 1$ as proposed in (Oliveira et al., 2008). Fullset is formed by 65536 rules while $Subset_A$ is composed by only 21019 rules (32.1% of Fullset rules). Therefore, using $Subset_A$ specification, the reduction of possible keys is severe and the number of available secret keys is reduced to approximate $\frac{1}{3}$.

The objectives of this investigation are the same of those related on last subsection. The first objective is to calculate ciphertexts final length: Table 2 presents the performance of each set/subset: the average length of the final lattice or ciphertext (L_{mean}) and the average number of failures occurred during ciphering process (F_{mean}). Each average result was computed considering the application of each rule of the set to cipher 100 lattices of 256 bits. The second objective is to compare ciphertexts generated by pairs of similar plaintexts (X, X') aiming to verify if the method is secure against differential cryptanalysis-like attacks. D' was calculated to each pair of ciphertexts (Y,Y') to obtain the standard deviation (σ) for each rule set. Spatial entropy (E) (Oliveira et al., 2008) was calculated on D' to evaluate the existence of some undesirable regularity on this difference. E above 0.75 indicates a random difference enough to expect that ciphertexts Y and Y' do not maintain any similarity. Table 2 shows σ_{mean} and E_{mean} for each set of rules. We also used entropy E to determine when stop pre-image computation (P). D' entropy was calculated in some P steps to determine in which one the ciphering will be stopped. D' entropy (E) is first calculated in $P = 16$. If E is above 0.75 the pre-image computation stops and the 16th pre-image is the ciphertext. Otherwise, this process is repeated to $P = 32$, 64 and 128. If until $P = 128$ entropy E does not meet 0.75, the ciphertext is the 128th pre-image. A mean standard deviation (σ_{mean}) below 5% was obtained for all the sets of rules. Therefore, the proposed CA cryptographic model can be considered secure in relation to differential cryptanalysis. The mean number of faults (F_{mean}) during ciphering process is very low for all set - below 0.1 – which returns a mean ciphertext size very close to the original size 256 bits.

$Subset_A$ returned the smaller ciphertexts, indicating that this specification indeed reduce the final size as observed in last subsection. When considering the mean entropy of D' (E_{mean}), all sets returned high values, above 0.87, indicating that the rules are able to add a high entropy during ciphering. Therefore, considering only the mean values of each set analyzed, all of them returned good values on the measures analyzed and there was no need to reduce the set of keys. However, the worst performing rules in $Fullset$ indicate the existence of secret keys not appropriate for ciphering purpose. Such rules are highlighted in the right side of Table 2, in which only the 500 worst performing rules in each set are considered. The mean number of faults (F_{mean}) is above 5 in $Fullset$, indicating that these rules return ciphertexts with size superior to 276 bits in average. Moreover, F_{mean} represents the mean value size for each rule considering all the 100 lattices used to test it. However, if we consider the worst result in such lattices, we can find ciphertexts with a considerable size: column F_{max} shows the mean of the maximum ciphertext size obtained considering the 500 worst rules in each set. This metric highlights the existence of secret keys in $Fullset$ returning ciphertexts with size superior to 290 bits. It was possible to notice that there are more than 200 rules in $Fullset$ returning ciphertext lengths between 280 and 740 bits in the worst case and more than 270 bits in average. There are only 17 rules in $Subset_A$ that returns ciphertext length above 300 bits in the worst case and only 97 rules with mean value between 260 and 280 bits. Therefore, considering the final ciphertext length, rules of $Subset_A$ presented much better performance both in mean values considering the entire rule set and in mean and maximum values considering the worst rules (Table 2). Naturally, this better performance is a consequence of the low number of fails. No significant difference is clear in mean entropy considering all rules in Table 2. Only when observing 500 worst performing rules considering entropy values differences are highlighted. The mean entropy in D' (E_{mean}) is bellow 0.5 in $Fullset$ and $Subset_A$, indicating that they do not perform an actual encryption of the plaintexts in average. E_{min} represents the worst entropy found for each rule considering all the 100 lattices. E_{min} is below 0.1 for $Fullset$ and $Subset_A$. It indicates the existence of lattices that are not encrypted by some few rules. The most probable behavior of such CA rules is that they only shift the initial lattices, not performing an actual encryption of plaintexts. Although this behavior is a minor occurrence in the entire set of CA rules it cannot be allowed in a cryptosystem.

		All rules				500 worst rules			
Set	Number of rules	L_{mean}	F_{mean}	E_{mean}	$\sigma_{mean}(\%)$	F_{mean}	F_{max}	E_{mean}	E_{min}
$Fullset$	65536	256.38	0.095	0.879	3.66	5.851	13.448	0.170	0.036
$Subset_A$	21019	256.10	0.025	0.872	3.65	0.932	3.212	0.396	0.037

Table 2. Mean values obtained for all rules and for the 500 worst rules in each set.

Concluding our analysis: (*i*) There are a lot of undesirable behavior rules in $Fullset$ – considering a cryptographic purpose - that must be avoided as secret keys. Therefore the entire rule space formed by all radius 2 toggle rules (totalizing 130816 rules including left-toggle and right-toggle and the existence of 256 left-and-right-toggle) cannot be applied as secret keys in VLE method. Approximate 6000 rules must be avoided (3500 due to low entropy and 2500 due to long ciphertext length). (*ii*) The specification proposed in (Oliveira et al., 2008) and evaluated in last section try to filter such undesirable behavior keys. However, its application is not so effective. The specification in (Oliveira et al., 2008) was proposed with the major goal of reducing ciphertext lengths, what is really achieved. But, the reduction imposed in key space is high (approximate 66%) and it could not avoid a great number of low entropy rules.

In respect to the number of pre-images P used during ciphering, the average of the maximum number of P used for each radius 2 rule of the $Fullset$ was of 33.10 pre-images (considering the 100 plaintexts of 256-bits analyzed). Thus, we conclude that a fixed number of pre-images $P < 64$ could be enough to cipher 256-bits plaintexts although $P = 128$ was used in the first experiments using VLE (Oliveira et al., 2008; 2010a).

5.3 Investigating the complete radius 2 rules set with a fixed number of pre-image steps

A new experiment was performed using the complete radius 2 rules set to cipher samples of 256-bits plaintexts employing a fixed and predefined number of pre-image steps (P). Different values of P were considered in this analysis: 32, 40, 48, 56 and 64. Using $P = 64$, the best result related to perturbation spread was obtained as expected (that is, the highest entropy) but additionally the number of fails increased (when compared with $P = 32$). On the other hand, using $P = 32$, it was possible to observe that this number of steps was not enough to spread perturbations when using a considerable portion of rules of the entire set. Considering the trade-off between perturbation spread and number of fails we conclude that the best evaluated value of P was 48. VLE environment were used to cipher one hundred 256-bits plaintexts by calculating 48 consecutive pre-image steps. We obtained $L_{mean} = 257.90$ and $F_{mean} = 0.476$, which returns a mean ciphertext size very close to the original size (256 bits). We obtained $\sigma_{mean} = 3.83\%$ and $E_{mean} = 0.876$ for the complete set of right-toggle rules. Since σ_{mean} is below 10%, the proposed CA cryptographic model can be considered secure in relation to differential cryptanalysis. E_{mean} is above 0.85 indicating that rules were able to add a high entropy during ciphering, using $P = 48$. However, the analysis of the worst performing rules in the set indicates the existence of secret keys not adequate for ciphering purpose. The entire rule space formed by all radius 2 right-toggle rules is not appropriate to be applied as secret keys in VLE method. Approximate 4000 rules (6% of the key space) must be avoided: 3200 due to low entropy and 800 due to long ciphertext length.

Aiming to better understand the relation between CA static parameters and underperforming rules, several parameters was used trying to identify a pattern to filter such rules; they are: Z_{right}, Symmetry (S), Absolute Activity (AA), Neighborhood Dominance (ND), Core Entropy (CE), complementary-plus-reflection Symmetry ($BWLR$), reflection Symmetry (LR), Sensitivity and Activity Propagation (AP) (Oliveira et al., 2001; 2010b). These nine parameters were calculated for all the 65536 radius 2 rules. A database was elaborated in which each register corresponds to one right-toggle transition rule and the fields are composed by the values of the nine parameters calculated for the rule, and the values of the metrics calculated when the transition rule is applied to cipher 100 plaintexts: F_{mean} and E_{mean} (results obtained with 48 pre-image steps). We first analyze the fields F_{mean} and E_{mean} of each register of the database to characterize the underperformed rules in specific classes. Field Class was added to the database with the classification of each register in one of these classes: (1) 886 rules with low mean entropy (< 0.5); (2) 750 rules with large mean ciphertext length (> 300 bits); and (3) 63900 remaining rules. As a pattern associating parameters with the worst performing rules in ciphering was not possible to recognize by a simple visual inspection, it was applied a data mining process (Fidelis et al., 2000). A standard GA was elaborated to mining IF-THEN rules associating the parameters (antecedent) with the performance class (consequent) (Oliveira et al., 2010c). The antecedent part of such rules is composed by a conjunction of conditions of the type: IF <parameter> <operator> <value>, being that the operator can be $<$ (minor), $\geq$ (larger or equal) or $\neq$ (different). The consequent part is always in the format THEN <class> = C, being that C can be "(1) Low Entropy", "(2) Large Ciphertext" or "(3) Adequate". The classification rule that was indeed intended to be mined is Class 3, because it represents appropriate transition rules to be used in cryptography. However, the other two classification rules (classes 1 and 2) were also important to achieve our goal, because they better characterize low entropy transition rules and long ciphertext ones, given us some important information to prune the rules returned by GA for Class 3. After several executions and post-processing pruning procedures we have found the best rules following:

$Rule_{ND}$: IF $S \neq 1$ AND $ND < 0.54$ AND $CE > 0.645$ AND $Z_{right} \neq 1$ AND $Z_{right} > 0.35$ THEN $Class = Adequate$
$Rule_{AA}$: IF $S \neq 1$ AND $AA \geq 0.46$ AND $CE > 0.645$ AND $Z_{right} \neq 1$ AND $Z_{right} > 0.35$ THEN $Class = Adequate$
Both rules employ four of the nine CA parameters to characterize adequate secret keys, being that tree of them are Symmetry (S), Core Entropy (CE) and Z_{right} parameters. Additionally, the first uses Neighborhood Dominance (ND) parameter and the second one uses Absolute Activity (AA) parameter, then we named them as $Rule_{ND}$ and $Rule_{AA}$. We applied each one of them as a filter in the complete radius 2 right-toggle CA rule set to remove all secret keys that non attending its conditions. The filtered set obtained applying $Rule_{ND}$ has 41556 right-toggle rules and it was named $Subset_{ND}$. The filtered set obtained applying $Rule_{AA}$ was named $Subset_{AA}$ and it has 47263 right-toggle rules. By using the environment implemented based on VLE, we employed them to cipher one hundred 256-bits plaintexts, by calculating 48 consecutive pre-image steps, as performed to the complete rule set. Table 3 shows the results obtained with $Subset_{ND}$ and $Subset_{AA}$ rules (considering all the remaining rules in each set), including mean values obtained with each set. Results of the complete set (65536 rules), named as $Fullset$, are also presented in this table.

Set	Number of Rules	Complete rule sets				500 worst performing rules			
		L_{mean}	F_{mean}	E_{mean}	σ_{mean}	F_{mean}	F_{max}	E_{mean}	E_{min}
$Fullset$	65536	257.903	0.476	0.876	3.83%	24.607	33.212	0.125	0.036
$Subset_{ND}$	41556	257.442	0.361	0.889	3.67%	10.045	16.608	0.839	0.625
$Subset_{AA}$	47263	257.488	0.372	0.889	3.68%	10.479	17.220	0.837	0.575

Table 3. Mean values of the complete rule sets and the 500 worst performing rules.

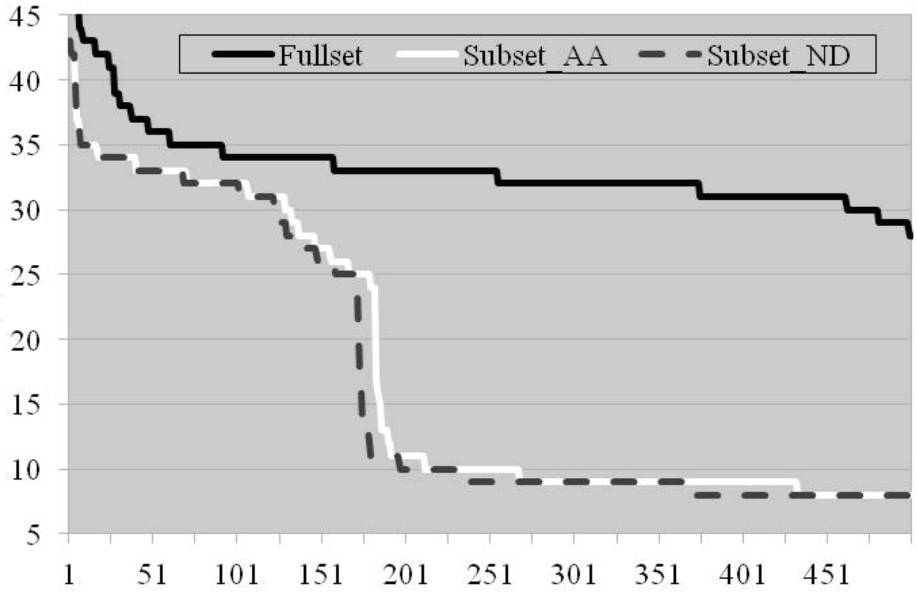

Fig. 1. 500 longest ciphertexts found in each set/subset.

Considering the totality of the rules, $Subset_{ND}$ and $Subset_{AA}$ results are better than those obtained using the entire radius 2 set, but the differences are not so expressive. However, when the worst performing rules are analyzed the advantage of such filters is evidenced. The mean values obtained considered only the 500 worst performing rules in Table 3 (F_{mean}, F_{max},

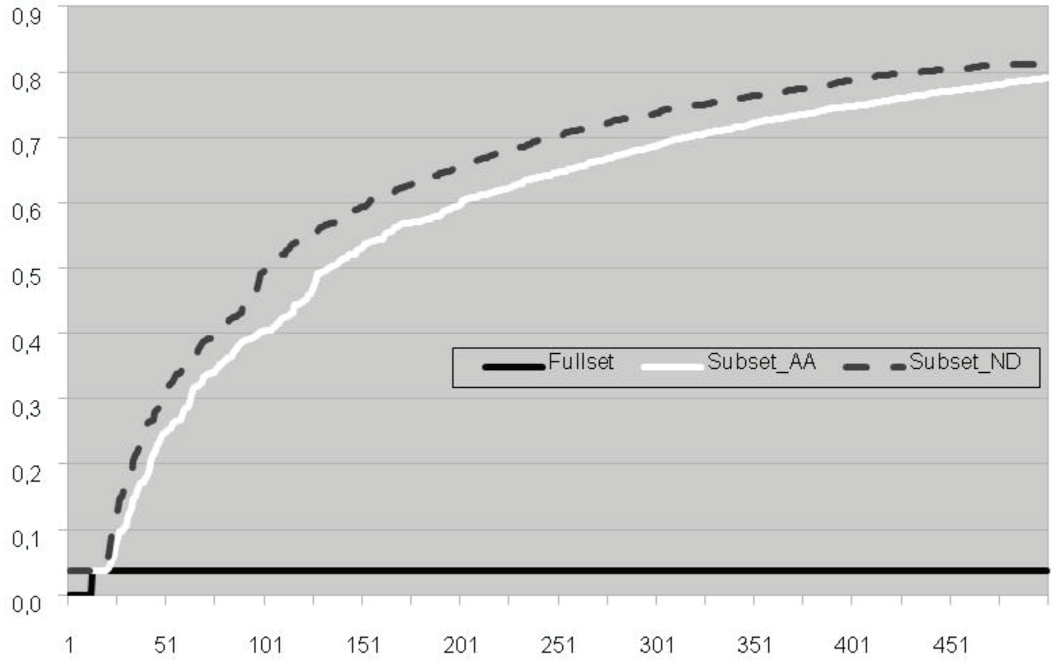

Fig. 2. 500 lowest entropy found in each set/subset.

E_{mean} and E_{min}). The number of fails exhibits a significant decay in both subsets and the longest ciphertexts were also smoothed. But the most expressive results are related to low entropy rules: the mean entropy of the worst rules improved from 0.125 to acceptable values (above 0.8) in both subsets. These results are highlighted in figures 1 and 2. Figure 1 shows the 500 longest ciphertexts while Figure 2 shows the 500 lowest entropy found in each set/subset. Comparing $Subset_{ND}$ and $Subset_{AA}$, the reduction of rules is more severe in $Subset_{ND}$ (41556 against 47263). Besides, it is possible to see that $Subset_{ND}$ returned results slightly better than $Subset_{AA}$, but this improvement does not justify such reduction imposed by $Rule_{ND}$. However, we decided to kept $Rule_{ND}$ as one of the best filters because the generalization of Neighborhood Dominance (ND) parameter from radius 2 to bigger radius is better than Absolute Activity (AA) parameter, as discussed in (Oliveira et al., 2000). Some experiments using radius 3 toggle rules specified using a filter rule based on ND are presented in (Oliveira et al., 2010c). The rule based on ND presented here is better than this previous filter rule which was obtained in preliminary data mining analysis. The tests performed using radius 3 rules returned adequate mean values. Moreover, no rule returns low entropy in any ciphered plaintext (Oliveira et al., 2010c).

6. Conclusions

The variable-length encryption method named VLE is a cryptographic model based on the backward interaction of cellular automata toggle rules. The general idea of this method was proposed in a previous work (Oliveira et al., 2008). This method alternates during the ciphering process the employment of the original reverse algorithm (Wuensche & Lesser, 1992) with a variation inspired in Gutowitz's model (Gutowitz, 1995), which adds extra bits when a pre-image is calculated. The plaintext is encrypted using this method to calculate P consecutive pre-images using a CA toggle rule τ as the secret key. As the number of failures when calculating consecutive pre-images using the reverse algorithm is not fixed, the final ciphertext length is variable. However, in practice, it is expected to obtain ciphertext

length close to original plaintext. Starting from the ciphertext, the recipient needs to apply the transition rule τ forward by P steps and the final lattice will be the plaintext.
We performed three series of experiments. The first one uses a small number of radius 2 and radius 3 rules and a number of pre-image steps fixed in 128 as suggested in previous works. The second experiment uses a variable number of pre-image steps (from 16 to 128) and a more representative set of radius 2 rules: the complete set of all possible radius 2 rules with Z_{left} equal to 1. The third one uses a fixed number of pre-image steps (P equal to 48), pruned based on the results of the second experiment in the same representative set of radius 2 rules. The average of standard deviation found is 3.83%, showing this method is very robust to a differential cryptanalysis-like attack being much lower than the upper bound limit suggested in (Biham & Shamir, 1991): 10%. Comparing with the results presented in (Sen et al., 2002) the superiority of VLE is clear in such criteria: 12%, 7% and 5% returned by DES, AES and CAC respectively, being the last one a CA-based method. Besides, the absence of an ordered pattern when ciphering similar plaintexts was evidenced by the mean entropy found: 0.876. Using VLE we have the guarantee that ciphering is possible even if an unexpected Garden-of-Eden state occurs. However a short length ciphertext depends on the secret key specification. The properly specification of the rules/key was deeper investigated in the present work using all the 65536 radius 2 right-toggle rules. Initially, we investigated specifications based on previous works (Oliveira et al., 2008) and (Wuensche, 2008), which uses static rule parameters Z and S. Our experimental results showed that none of these previous specifications are satisfactory. Thus, in a subsequent phase, we employed an analysis based on several CA static parameters. Therefore, we were able to find two good specifications of rules to be used as valid secret keys. The first specification is based on Neighborhood Dominance parameter while the second is based on Absolute Activity. These specifications had shown to be good to filter the complete set of radius 2 rules.

7. Acknowledgements

GMBO thanks CNPq and FAPEMIG support. The authors thanks to Giordano Ferreira who has participated in the initial steps of this work, specially in the development of the genetic algorithm to perform data mining.

8. References

Benkiniouar, M. & Benmohamed, M. (2004). Cellular Automata for Cryptosystem, In: *Proceedings of IEE Conference Information and Communication Technologies: From Theory to Applications*, 423-424.

Biham, E. & Shamir, A. (1991). Differential Cryptanalysis of DES-like Cryptosystems, *Proc. of Advances in Cryptology (Crypto'90)*. 2-21.

Binder, P. (1994). Parametric Ordering of Complex Systems, *Physics Rev. E.*

de Oliveira, P.P.B.; Bortot & Oliveira, G.M.B. (2006). The best currently known cellular automata rules for density classification and the evolutionary mechanisms that led to them, *Neurocomputing*, Amsterdam, 70:35-43.

Fidelis, M.; Lopes, H. & Freitas, A. (2000). Discovery comprehensible classification rules with a genetic algorithm, *C.Evolutionary Computation, CEC-2000*, USA.

Gutowitz, H. (1995). Cryptography with Dynamical Systems, In: *Cellular Automata and Cooperative Phenomena*, E. Eds: Goles and N. Boccara, 1:237-274, Dordrecht: Kluwer Academic Press.

Kari, J. (1992). Cryptosystem based on reversible cellular automata, Personal communication. *Apud in (Seredynski, Bouvry and Zomaya, 2003).*

Langton, C.G. (1990). Computation at the Edge of Chaos: Phase Transitions and Emergent Computation, *Physica D*, 42:12-37.

Macêdo, H.B.; Lima, M.J.L & Oliveira, G.M.B. (2008). Searching for a Cryptographic Model Based on the Pre-Image Calculus of Cellular Automata, In: *Proceedings of 10th Brazilian Symposium on Neural Networks (SBRN'2008)*, IEEE Computer Society, 153-158.

Nandi, S.; Kar, B., & Chaudhuri, P. (1994). Theory and Applications of CA Automata in Cryptography, In: *IEEE Transactions on Computers*, 43:1346-1357.

Oliveira, G., de Oliveira, P. e Omar, N. (2000). Guidelines for Dynamics-Based Parameterization of One-Dimensional Cellular Automata Rule Spaces, *Complexity*, 6(2):63-71.

Oliveira, G.M.B.; de Oliveira, P. & Omar, N. (2001). Definition and applications of a five-parameter characterization of 1D cellular automata rule space, *Artificial Life*, 7(3):277-301.

Oliveira, G.M.B.; Coelho, A.R. & Monteiro, L.H.A. (2004). Cellular Automata Cryptographic Model Based on Bi-Directional Toggle Rules, In: *International Journal of Modern Physics C*, 15:1061-1068.

Oliveira, G.M.B.; Macêdo, H.B.; Branquinho, A.A.B. & Lima, M.J.L. (2008). A cryptographic model based on the pre-image calculus of cellular automata, In: *Automata-2008: Theory and Applications of Cellular Automata*, ed: A. Adamatzky, R. Alonso-Sanz, A. Lawniczak, G. Juarez Martinez, K. Morita, T. Worsch, Luniver Press, 139-155.

Oliveira, G.M.B.; Martins, L.G.A.; Alt, L.S. & Ferreira, G.B.: (2010a). Investigating a Cellular Automata-Based Cryptographic Model with a Variable-Length Ciphertext, In: *CSC'10 - International Conference on Scientific Computing*, Las Vegas, 2010.

Oliveira, G.M.B.; Martins, L.G.A.; Alt, L.S. & Ferreira, G.B.: (2010b). Exhaustive Evaluation of Radius 2 Toggle Rules for a Variable-Length Cellular Automata Cryptographic Model, In:*International Conference on Cellular Automata for Research and Industry*, 2010, Ascoli Piceno, Lecture Notes in Computer Science (LNCS), 2010, v. 6350. p. 1-10.

Oliveira, G.M.B.; Martins, L.G.A.; Ferreira, G.B. Alt, L.S.: (2010c). Secret Key Specification for a Variable-Length Cryptographic Cellular Automata-Based Model, In: *11th International Conference on Parallel Problem Solving From Nature (PPSN2010)*, 2010, Krakow, Lecture Notes in Computer Science (LNCS), 2010, v. 6239. p. 1-10.

Sen, S.; Shaw, C.; Chowdhuri, D.; Ganguly, N. & Chaudhuri, P. (2002). *Cellular Automata based Cryptosystem (CAC)*, LNCS, 2513: 303-314.

Seredynski, F.; Bouvry, P. & Zomaya, A.Y. (2003). Secret key cryptography with cellular automata, In: *Proceedings of Workshop on Nature Inspired Distributed Computing (IPDPS2003: International Parallel & Distributed Processing Symposium)*, 149-155.

Stallings, W. (2003). *Cryptography and Network Security: Principles and Practice*, ISBN:0-13-091429-0, New Jersey: Prentice Hall.

Tomassini, M. & Perrenoud, M. (1986). Stream Ciphers with One and Two-Dimensional Cellular Automata, In: *Proceedings of Parallel Problem Solving from Nature (PPSN VI)*, LNCS (Springer-Verlag), 1917:722-731.

Wuensche, A. & Lesser, M. (1992). *Global Dynamics of Cellular Automata*, ISBN: 0-201-55740-1, New Mexico: Addison-Wesley.

Wuensche, A. (1998). Classifying Cellular Automata Automatically: Finding gliders, filtering, and relating space-time patterns, attractor basins and the Z parameter, *Complexity*, 4 (3):47-66.

Wuensche, A. (2008). Encryption using cellular automata chain-rules, In: *Automata-2008: Theory and Applications of Cellular Automata*, ed: A. Adamatzky, R. Alonso-Sanz, A. Lawniczak, G. Juarez Martinez, K. Morita, T. Worsch, Luniver Press, 126-138.

Wolfram, S. (1986). Cryptography with cellular automata, In: *Proceedings of International Cryptology Conference (Crypto'85)*, LNCS (Springer-Verlag), 218:429-432.

Cellular Automata - Innovative Modelling for Science and Engineering
Edited by Dr. Alejandro Salcido

ISBN 978-953-307-172-5
Hard cover, 426 pages
Publisher InTech
Published online 11, April, 2011
Published in print edition April, 2011

Modelling and simulation are disciplines of major importance for science and engineering. There is no science without models, and simulation has nowadays become a very useful tool, sometimes unavoidable, for development of both science and engineering. The main attractive feature of cellular automata is that, in spite of their conceptual simplicity which allows an easiness of implementation for computer simulation, as a detailed and complete mathematical analysis in principle, they are able to exhibit a wide variety of amazingly complex behaviour. This feature of cellular automata has attracted the researchers' attention from a wide variety of divergent fields of the exact disciplines of science and engineering, but also of the social sciences, and sometimes beyond. The collective complex behaviour of numerous systems, which emerge from the interaction of a multitude of simple individuals, is being conveniently modelled and simulated with cellular automata for very different purposes. In this book, a number of innovative applications of cellular automata models in the fields of Quantum Computing, Materials Science, Cryptography and Coding, and Robotics and Image Processing are presented.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Gina M. B. Oliveira, Luiz G. A. Martins and Leonardo S. Alt (2011). Deeper Investigating Adequate Secret Key Specifications for a Variable Length Cryptographic Cellular Automata Based Model, Cellular Automata - Innovative Modelling for Science and Engineering, Dr. Alejandro Salcido (Ed.), ISBN: 978-953-307-172-5, InTech, Available from: http://www.intechopen.com/books/cellular-automata-innovative-modelling-for-science-and-engineering/deeper-investigating-adequate-secret-key-specifications-for-a-variable-length-cryptographic-cellular

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821