A New Digital Signature Scheme from Layered Cellular Automata

Xing Zhang[1], Rongxing Lu[2], Hong Zhang[1], Chungen Xu[3]
(Corresponding author: Rongxing Lu)

School of Computer Sciences and Engineering, Nanjing University of Science and Technology[1]
No. 200, Xiaolingwei Street, Nanjing, Jiangsu 210094, P.R. China
School of Electrical and Electronic Engineering, Nanyang Technological University[2]
50 Nanyang Avenue, 639798 Singapore
School of Science, Nanjing University of Science and Technology[3]
(Email: rxlu@ntu.edu.sg)

(Received Apr. 25, 2014; revised and accepted Aug. 25 & Nov. 27, 2014)

Abstract

Cellular Automata (CA) is one of the important tools to design cryptographic algorithms, and in the past years many researchers have explored several cryptographic algorithms based on CA. However, most of reported CA-based cryptographic algorithms focus on the symmetric key encryption schemes and few CA-based asymmetric encryption scheme has been proposed, let alone the CA-based digital signature scheme. In this paper, to fill this gap, we present a new digital signature scheme based on the layered CA technique. Specifically, in the proposed layered CA-based digital signature scheme, we combine the transition rules of some one-dimensional (1D) reversible CAs to generate the rules of a two-dimensional (2D) CA, where the reverse of the 1D transition rules are kept as the private key and the 2D transition rules are set as the corresponding public key. Based on the hardness assumption of the layered CA reversibility (LCAR) problem, we formally prove the proposed scheme is semantically secure against chosen-message attack in the random oracle model.

Keywords: Digital signature, layered cellular automata, reversible cellular automata, T-shaped neighborhood

1 Introduction

Digital signature is an indispensable technique in modern information security system [20]. In particular, a digital signature is a mathematical tool for demonstrating the authenticity of a digital message. A valid signature can give a recipient reason to believe that the message was created by a known signer, such that the signer cannot deny having signed the message (i.e., authentication) and the message not changed in transit (i.e., integrity) [18, 21]. Therefore, digital signatures have been widely used for software distribution, financial transactions, and in other cases where it is important to detect forgery or tampering today.

Applying cellular automata (CA) tool to design cryptographic algorithms is a promising technique in modern cryptography. CA can be regarded as a discrete model that consists of a number of individual cells, each cell exists several states and will change its state based on the states of its neighboring cells by following a prescribed rule. In general, the overall structure can be viewed as a parallel processing device. However, the simple structure will produce complex patterns when iterated several times. Since most CA can be implanted on very fast hardware and software, as well as its inherent features like parallelism, locality and homogeneity, CA has become an important tool to design cryptographic algorithms.

While most of the investigations of CA-based cryptographic algorithms have been focused on traditional symmetric cryptosystems, few CA-based public key cryptosystem has been found in the literature [5, 11, 15, 24], and let alone the CA-based digital signature scheme. Therefore, there is a high desire to design a CA-based digital signature in the complement of existing RSA and ElGamal signatures [8, 19, 25].

In this paper, to fill this gap, we would like to present a new digital signature scheme based on the layered CA technique [2, 12, 23, 28]. Specifically, in the proposed CA-based digital signature scheme, we will combine the transition rules of some one-dimensional (1D) reversible CAs to generate the rules of a two-dimensional (2D) CA, where the reverse of the 1D transition rules are kept as the private key and the 2D transition rules are set as the corresponding public key. Based on the hardness assumption of the layered CA reversibility (LCAR) problem, we formally prove the proposed scheme is semantically secure against chosen-message attack in the random oracle model [3].

The remainder of this paper is organized as follows. In Section 2, some preliminaries are introduced, including some notations, the foundations of digital signature, and the concept of the cellular automata and its corresponding security assumption. In Section 3, we present our layered cellular automata based signature scheme, followed by security analysis in Section 4, and a simple example in Section 5. In Section 6, we analyze the strengths of the proposed signature scheme. Finally, we draw our conclusions in Section 7.

2 Preliminaries

2.1 Notations

Let $\mathbb{N} = \{1, 2, 3 \ldots\}$ be the set of positive integers. If x is a string , then $|x|$ denotes its length, while if $\mathbb{S}$ is a set then $|\mathbb{S}|$ denotes its cardinality. If $k \in \mathbb{N}$ then 1^k denotes the string k ones. If $\mathbb{S}$ is a set then $s \xleftarrow{R} \mathbb{S}$ denotes the operation of picking a random element s of $\mathbb{S}$ uniformly.

2.2 Foundations of Digital Signature

A digital signature (DS) scheme consists of three algorithms: Key Generation, Signature Generation, and Signature Verification.

- Key Generation (KG): On input of an unary string 1^k with security parameter k, KG outputs a public and private key pair (pk, sk). Here, KG is a randomized algorithm in digital signature.
- Signature Generation (SG): On input of a message m, the public and private key pair (pk, sk), SG outputs a signature σ of m with respective to the public key pk. Note that, SG here is referred as a deterministic algorithm [25]; when SG is randomized one, some random input should also be included [8].
- Signature Verification (SV): On input of a purported signature σ of m with respective to the public key pk, SV outputs “1” if (m, σ) is valid, and “0” otherwise. Note that, SV must be a deterministic algorithm in digital signature.

The above algorithms must satisfy the standard consistency constraint of the digital signature. That is, if a signature $\sigma \leftarrow SG(m, pk, sk)$ is generated, then we must have “1” $\leftarrow SV(\sigma, m, pk)$.

Security Model of Digital Signature. For digital signatures, the well-known strong security notion is existential forgery against adaptive chosen message attacks (EF-CMA) presented by Goldwasser [9] et al.

In the random oracle model [6], we consider the most powerful adversary $\mathcal{A}$ as follows: i) $\mathcal{A}$ is allowed to access to the signing oracle $\mathcal{O}_S$ and the random oracle $\mathcal{O}_H$; ii) $\mathcal{A}$ returns a new valid signature $\sigma^\star$ on message $m^\star$, with a natural restriction that the signature $\sigma^\star$ has not been obtained from the signing oracle $\mathcal{O}_S$ before.

Definition 1 (Unforgeability). *Let DS be a digital signature, and $\mathcal{A}$ be an EF-CMA adversary against DS. We consider the following random experiments, where k is the security parameter:*

$$\begin{aligned} &\text{Experiment } \ \mathbf{Exp}^{\textsf{EF-CMA}}_{\textsf{DS},\mathcal{A}}(k) \\ &\quad (pk, sk) \leftarrow KG(k), \\ &\quad (\sigma^\star, m^\star) \leftarrow \mathcal{A}^{\mathcal{O}_H, \mathcal{O}_S}(pk) \\ &\quad \text{return } \ SV(pk, \sigma^\star, m^\star) \end{aligned}$$

We define the success probability of $\mathcal{A}$ via

$$\mathbf{Succ}^{\textsf{EF-CMA}}_{\textsf{DS},\mathcal{A}}(k) = \Pr[\mathbf{Exp}^{\textsf{EF-CMA}}_{\textsf{DS},\mathcal{A}}(k) = 1]$$

Let $\tau \in \mathbb{N}, \epsilon \in [0, 1]$, we say that DS is (τ, ϵ)-secure if no EF-CMA adversary $\mathcal{A}$ running in time τ has a success $\mathbf{Succ}^{\textsf{EF-CMA}}_{\textsf{DS},\mathcal{A}}(k) \geq \epsilon$.

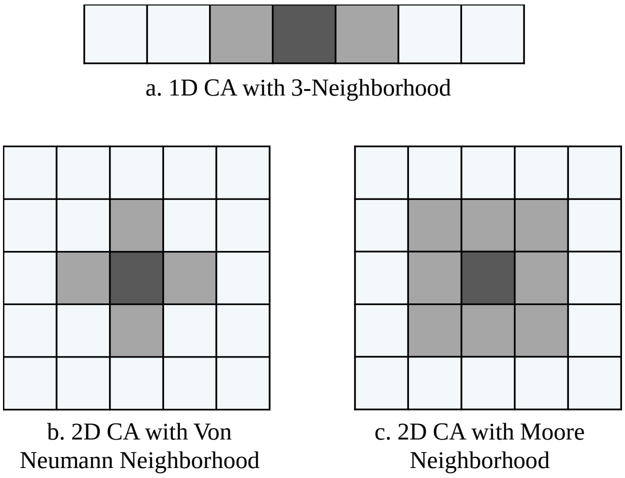
2.3 Concepts of Cellular Automata

2.3.1 Basis of Cellular Automata

A Cellular Automata (CA) is a discrete model in which space and time are discrete, and consists of grids of cells in which each cell can exist in a finite number of states. All cells change their states synchronously, according to a predefined transition rule that specifies the new state of each cell based on the old states of the cell and its neighboring cells. As CA exhibits some inherent features like parallelism, locality, simplicity, unpredictability and homogeneity, it is naturally efficient in its hardware and software implementations [28].

Formally, a CA is often defined by a quadruple $\{D, S, N, f\}$ with the dimension D , the state set S , the neighboring states set N, and the transition rule f.

- Dimension D: define the dimension of CA, which can be one-dimensional (1D) or two-dimensional (2D), and a d-dimensional ($d \in \mathbb{N}$) CA consists of a d-dimensional array of identical cells [15]. Most of existing studies of CA are focused on 1D and 2D CA [1, 4, 13, 14, 26, 27] as shown in Figure 1.
- State Set S: define a set of possible states of all cells in a CA, which is often defined as the sets, such as $S_1 = \{0, 1\}$, $S_2 = \{0, 1, 2, \ldots\}$, and $S_3 = \{white, black\}$.
- Neighboring States N: define a set of neighboring states based on the existing neighborhood structures. Currently, the most popular structures are 3-neighborhood, Von Neumann and Moore neighborhood [28], as shown in Figure 1.
- Transition Rule f: define a transition map $f : S \rightarrow S$ as the transition rule from one state to another.



a. 1D CA with 3-Neighborhood

b. 2D CA with Von Neumann Neighborhood

c. 2D CA with Moore Neighborhood

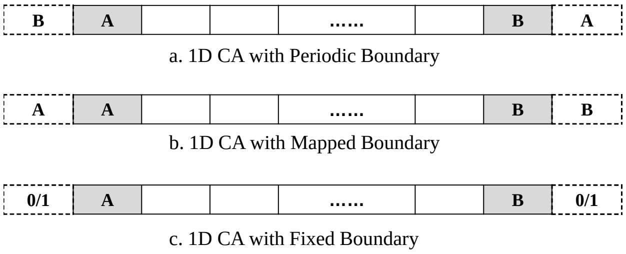
Figure 1: Typical neighborhood structure of CA with radius $r = 1$

Let $s_i^t \in S$ denote as the state of the i-th cell at t time step and $s_i^{t+1} \in S$ be the state of the i cell at $t+1$ time step. Then, the states of all cells in a CA at t time step can be denoted as $S^t = (s_0^t, s_1^t, \cdots, s_i^t, \cdots)$, also called a configuration. As the state of each cell at the next time step s_i^{t+1} is determined by the transition rule along with its current state s_i^t and states of its neighboring cells, we can represent s_i^{t+1} by the following formula:

$$s_i^{t+1} = f(s_{i-r}^t, \cdots, s_{i-1}^t, s_i^t, s_{i+1}^t, \cdots, s_{i+r}^t)$$

where r denotes its neighborhood radius.

Boundary Conditions. Though a CA is an infinite system, it should be finite-dimensional in practical applications. As a result, it is crucial to define the boundary conditions of a CA. Currently, the boundary conditions, including periodic boundary condition, mapped boundary condition, and fixed boundary [30], are mostly considered in CA systems, as shown in Figure 2. Since the periodic boundary comes closest to simulate an infinite lattice, it has been widely suggested in many CA systems.



a. 1D CA with Periodic Boundary

b. 1D CA with Mapped Boundary

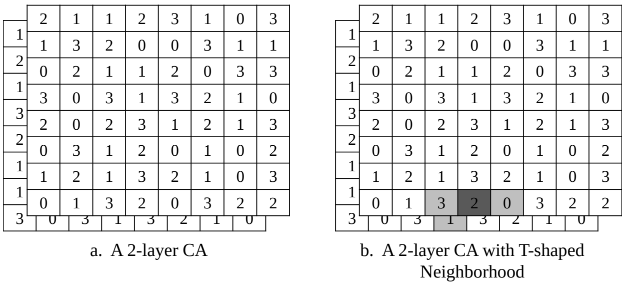
c. 1D CA with Fixed Boundary

Figure 2: 1D CA with different boundaries, where “A” is the leftmost cell state, “B” is the rightmost cell state, and “0/1” is the fixed cell state.

Reversibility. If the transition rule of a CA is reversible, we say the CA is Reversible CA (RCA)[16]. Otherwise, the CA is called irreversible. In specific, a CA is reversible, if and only if each configuration has only one succeeding state and one preceding state. Due to the reversibility, many RCA systems have been designed for symmetric cryptosystems [5, 7, 17, 26, 29], where the same transition rule serves as the secret key applying into both encryption and decryption operations. However, irreversible CA, due to irreversibility, is not as popular as the RCA in designing CA-based cryptosystems.

2.3.2 Layered Cellular Automata

Layered CA (LCA) is a special CA, which can be regarded as a highly parallel system consisting of layers and each layer is formed by rows of 1D CA, as shown in Figure 3. The stacked structure of LCA enables the cells in LCA to hold more complex and various neighborhoods, which has brought more theoretical interest [2, 12, 23, 28]. Jaberi et al. [12] use two-layer CA to imitate Pseudo-Neumann neighborhood structure and generate trackable random numbers. Kishore et al. [28] propose a block encryption scheme by using a 8-layer CA, which observably possesses better confusion and diffusion properties compared with the well-known AES [22].



2	1	1	2	3	1	0	3
1	3	2	0	0	3	1	1
0	2	1	1	2	0	3	3
3	0	3	1	3	2	1	0
2	0	2	3	1	2	1	3
0	3	1	2	0	1	0	2
1	2	1	3	2	1	0	3
0	1	3	2	0	3	2	2

a. A 2-layer CA

2	1	1	2	3	1	0	3
1	3	2	0	0	3	1	1
0	2	1	1	2	0	3	3
3	0	3	1	3	2	1	0
2	0	2	3	1	2	1	3
0	3	1	2	0	1	0	2
1	2	1	3	2	1	0	3
0	1	3	2	0	3	2	2

b. A 2-layer CA with T-shaped Neighborhood

Figure 3: 2-layer CA with 1-radius T-shaped neighborhood.

A 2-layer CA with 1-radius T-shaped neighborhood is shown in Figure 3, where the state of each cell is changed based on not only itself and its left and right neighbors, but also the cell at the same position in the other layer. This neighborhood structure unites two layers of the CA as a unified system, which can effectively improve the diffusion property. In this work, we will present a new digital signature scheme based on the LCA with T-shaped neighborhood structure.

Security Assumption. In order to construct a new digital signature scheme from the layered cellular automata, we should define the computational hard problem on which we can rely. Kari [15] has proven that the reversibility of 2D CA is undecidable, i.e., there does not exist any efficient algorithm that can decide whether a given two-dimensional transition rule is reversible or not [16]. Thus, for a given 2D transition rule, we can't decide its reversibility, and then we also can't compute its reverse. The 2D transition rule of a 2-layer CA can be constructed from several 1D reversible transition rules [5]. Given these 1D reversible transition rules, we can generate the 2D transition rule and the 2-layer CA is reversible. However, only given the 2D transition rule, we cannot gain these 1D reversible transition rules, and the 2-layer CA is irreversible. We call it as LCA Reversibility (LCAR) Problem.

Definition 2 (LCAR Problem). *Let $f_i : S_1 \rightarrow S_1$, $i \in \{1, 2, 3, \ldots, n\}$, be n reversible transition rules of 1D RCAs, where S_1 is state set of these RCAs. Define $f_{ca} : S_2 \rightarrow S_2$ be the transition rule of a 2-layer CA, where S_2 is state set of the 2-layer CA, and f_{ca} is constructed by the compound operations of transition rules f_i, $i \in \{1, 2, 3, \ldots, n\}$, i.e., $f_{ca} = f_1 \circ f_2 \circ \cdots \circ f_n$, and then we set $S_2 = S_1$. Given any configuration $S_2^t = (s_0^t, s_1^t, \cdots, s_i^t, \cdots)$ of time t of the 2-layer CA, where $s_i^t \in S_2$, we evolve S_2^t by the transition rule f_{ca}, and obtain $S_2^{t+1} = f_{ca}(S_2^t)$. The LCAR problem is that for given (S_2^{t+1}, f_{ca}), computing S_2^t is impossible.*

Base on the above problem, we give the security assumption of LCA as follows.

Definition 3 (LCAR Assumption). *Given any configuration $S_2^t = (s_0^t, s_1^t, \cdots, s_i^t, \cdots)$ of a 2-Layer CA, we evolve S_2^t by the transition rule f_{ca} and obtain $S_2^{t+1} = f_{ca}(S_2^t)$. Let $\mathcal{A}$ be a probabilistic polynomial-time (PPT) adversary, which takes (S_2^{t+1}, f_{ca}) as input and outputs S_2^t. We consider the following random experiment on LCAR problem of LCA:*

$$\begin{aligned}
&\text{Experiment } \mathbf{Exp}_{\mathcal{A}}^{LCAR} \\
&\quad S_2^{t+1} \leftarrow f_{ca}(S_2^t), \\
&\quad S_2^{\star} \leftarrow \mathcal{A}(S_2^{t+1}, f_{ca}) \\
&\quad \text{if } S_2^{\star} = S_2^t, \text{return 1; and return 0 otherwise}
\end{aligned}$$

We define the success probability of $\mathcal{A}$ via

$$\mathbf{Succ}_{\mathcal{A}}^{LCAR} = \Pr[\mathbf{Exp}_{\mathcal{A}}^{LCAR} = 1]$$

Let $\tau \in \mathbb{N}, \epsilon \in [0, 1]$, we say that LCAR is (τ, ϵ)-secure if no PPT adversary $\mathcal{A}$ running in time τ has a success $\mathbf{Succ}_{\mathcal{A}}^{LCAR} \geq \epsilon$.

3 Proposed LCA-based Digital Signature Scheme

In this section, we propose our digital signature scheme based on CA with T-shaped neighborhood, which mainly consists of three algorithms, namely: Key Generation (KG), Signature Generation (SG), and Signature Verification(SV).

- **KG**: Given a security parameter $k \in \mathbb{N}$. First, we define some 1D RCAs with periodic boundary, labeled CA_1, CA_2,..., CA_n, and $CA_i = (1, S_1, N_r, f_i)$, for $i \in \{1, 2, \ldots, n\}$, where state set $S_1 = \{0, 1, 2, 3\}$ and N_r is the neighboring state set with radius r. The transition rule $f_i : S_1 \rightarrow S_1$, for $1 \leq i \leq n$ of the reversible CA_i is reversible, and the reverse rule of f_i, denoted f_i^{-1}, is also the map between the state set S_1, i.e. $f_i^{-1} : S_1 \rightarrow S_1$, for $1 \leq i \leq n$.

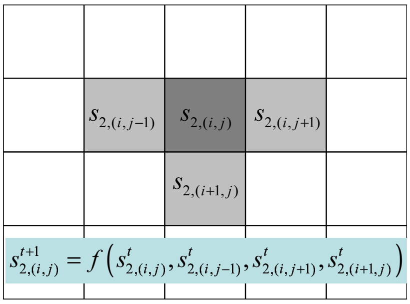

$s_{2,(i,j-1)}$
$s_{2,(i,j)}$
$s_{2,(i,j+1)}$
$s_{2,(i+1,j)}$
$s_{2,(i,j)}^{t+1} = f\left(s_{2,(i,j)}^t, s_{2,(i,j-1)}^t, s_{2,(i,j+1)}^t, s_{2,(i+1,j)}^t\right)$

Figure 4: 2D CA with 1-radius T-shaped neighborhood, where $s_{2,(i,j)}^t$ denotes the state of the cell at i-th row j-column.

Next, we define a 2-layer CA also with periodic boundary, denoted by $CA' = (2, S_2, N', f_{ca})$, where the state set $S_2 = \{0, 1, 2, 3\} = S_1$ and the transition rule f_{ca} is constructed by the compound operations of transition rules f_i, $i \in \{1, 2, \ldots, n\}$, i.e.,

$$f_{ca} = f_1 \circ f_2 \circ \cdots \circ f_n$$

The neighborhood structure of CA' is set as T-shaped neighborhood with radius $r' = 1$. In addition, we set each layer in the 2-layer CA has 64 cells, and then the 2-layer CA has total 128 cells. Specifically, we use f_i to generate the 2D transition rule f_{ca} by a 2D CA with 1-radius T-shaped neighborhood structure, as shown in Figure 4. We define the states of the cells in the CA are come from S_2, and let

$$S_{2,(i,j)}^t = (s_{2,(i,j-1)}^t, s_{2,(i,j)}^t, s_{2,(i,j+1)}^t, s_{2,(i+1,j)}^t)$$

denote the configuration of the cell at i-th row j-column at time t, where $s_{2,(i,j)}^t \in S_2$. As each cell has four possible states and together with three neighbors, there are total $4^4 = 256$ possible configurations of each cell. Take each configuration as the input of the compound operations $f_{ca} = f_1 \circ f_2 \circ \cdots \circ f_n$, the corresponding output is the new state of the central cell. This procedure can be presented as follows:

we first compute

$$s_{1,(i,j)}^{t+n} = f_n(f_{n-1}(\cdots(f_2(f_1(S_{1,(i,j)}^t)))))$$

and then let $S_{2,(i,j)}^t = S_{1,(i,j)}^t$ and $s_{2,(i,j)}^{t+1} = s_{1,(i,j)}^{t+n}$, where $s_{1,(i,j)}^{t+n} \in S_1, s_{2,(i,j)}^{t+1} \in S_2$. Since

$$f_{ca} = f_1 \circ f_2 \circ \cdots \circ f_n$$

we have

$$s_{2,(i,j)}^{t+1} = f_{ca}(S_{2,(i,j)}^t)$$

The map $f_{ca} : S^t_{2,(i,j)} \rightarrow s^{t+1}_{2,(i,j)}$ is the 2D transition rule we need. We define a function $F_{ca} : S_1 \rightarrow S_1$, where $F_{ca} = f_n^{-1} \circ \cdots \circ f_2^{-1} \circ f_1^{-1}$ and set the private key as $sk = F_{ca}$.

It is obvious that $f_{ca} = (F_{ca})^{-1}$, i.e., given the f_i^{-1}, for $i = 1, 2, \cdots, n$, f_{ca} is reversible and we can compute its reverse. However, if we only know the 2D transition rule f_{ca}, we cannot decide its reversibility, to say nothing of computing its reverse. Therefore, we set the f_{ca} as the corresponding public key pk, i.e., $pk = f_{ca}$.

Let $\mathbb{S}$ denote the message space, owing to the 2-layer CA with 128 cells, the size of $\mathbb{S}$ is $|S_1|^{128} = 4^{128} = 2^{256}$, where $|S_1|$ is the cardinality of the state set S_1, the message space is large enough to against exhaustive attack. Besides, we define a secure one-way hash function H, where $H : \{0, 1\}^* \rightarrow \mathbb{S}$.

- **SG**: On input of a message m, we compute R_1 and R_2, where

$$R_1 = H(m) \in \mathbb{S}$$
$$R_2 = F^k_{ca}(R_1) \in \mathbb{S}$$

i.e. $H(m)$ is evolved by the transition rules in the private key $sk = F_{ca}$ for k times, where $k \in \mathbb{N}$ is a large security parameter. Then we set the signature of m as $\sigma = R_2$.

- **SV**: For the message m and a purported signature σ, together with the public key $pk = f_{ca}$ and security parameter k, we compute R'_1, where

$$R'_1 = f^k_{ca}(\sigma) = f^k_{ca}(R_2) = (f_1 \circ f_2 \circ \cdots \circ f_n)^k(R_2)$$

Then, we check the following equality

$$R'_1 = H(m)$$

If it does hold, output '1', the signature σ will be accepted, and rejected otherwise.

4 Security Analysis

In this section, we will formally prove our proposed signature scheme satisfies the requirements stated in Section 2.2.

Theorem 1. *Let $\mathcal{A}$ be an adversary which can produce an existential forgery under chosen-message attacks [10] within a time τ and success probability ϵ, after q_H and q_S queries to the hash function H (modeled as random oracle $\mathcal{O}_H$) and the signing oracle $\mathcal{O}_S$ respectively. The LCAR problem can be resolved with another probability ϵ' within time τ', where $\epsilon' \approx \frac{1}{q_h+q_s+1}\epsilon$, $\tau' = \tau + (q_h + q_s + 1) \cdot \Theta$ with Θ the time for an $f^k_{ca}(\cdot)$ computation.*

Proof. We define a sequence of games $\mathbf{G_1}, \mathbf{G_2}, \cdots$, of modified attack games starting from the actual game $\mathbf{G_0}$. Then, with these incremental games, we reduce a LCAR problem instance (i.e. given f_{ca} and S_2^{t+1}, where $S_2^{t+1} = f_{ca}(S_2^t)$, compute S_2^t) to an attack against the proposed signature scheme. We show that the adversary $\mathcal{A}$ can help us to resolve the LCAR problem.

GAME $\mathbf{G_0}$. This is an actual game, in the random oracle model [3]. The adversary $\mathcal{A}$ is allowed to access a random oracle $\mathcal{O}_H$ and a signing oracle $\mathcal{O}_S$. Moreover, the public key $pk = f_{ca}$ is also available to $\mathcal{A}$.

To break the signature scheme, the adversary $\mathcal{A}$ outputs its forgery $(\sigma^\star, m^\star)$ that $m^\star$ has not been asked for $\mathcal{O}_S$, one then checks whether it is a valid signature or not. Note that the adversary $\mathcal{A}$ asks q_s queries to the signing oracle $\mathcal{O}_S$ and q_h queries to the random oracle $\mathcal{O}_H$, at most $q_s + q_h + 1$ queries are asked to the random oracle during this game, since each signing query may make such a new query, and the last verification step does too. We set $\mathbf{Forge}_0$ denotes the event that the forged signature is valid, and set the same notation $\mathbf{Forge}_n$ in any game $\mathbf{G_n}$. By definition, the success probability ϵ can be represented as follows:

$$\epsilon = \mathbf{Succ}^{\mathsf{EF\text{-}CMA}}_{\mathsf{DS},\mathcal{A}} = \Pr[\mathbf{Forge}_0]$$

GAME $\mathbf{G_1}$. In this game, we will simulate the hash oracle $\mathcal{O}_H$ by maintaining a hash list $L_{\mathcal{H}}$.

For a new hash query $\mathcal{O}_H(m)$, we randomly choose a new random element $r \in \mathbb{S}$, create and append a record $(m, h = f^k_{ca}(r), r)$ in $L_{\mathcal{H}}$, and respond with $H(m) = h$.

In order to implant the challenge $S_2^{t+1} = f_{ca}(S_2^t)$ into the hash answer, for some query on m^*, we insert a record $(m^*, h^* = f^{k-1}_{ca}(S_2^{t+1}), \sqcup)$ into the list $L_{\mathcal{H}}$, and respond with $H(m^*) = h^*$.

From the above simulation, we can see this game is indistinguishable from the actual attack. Consequently,

$$\Pr[\mathbf{Forge}_1] = \Pr[\mathbf{Forge}_0]$$

GAME $\mathbf{G_2}$. In this game, we simulate the signing oracle $\mathcal{O}_S$. For a signing query $\mathcal{O}_S(m)$, if $m \neq m^*$, we first look up the record $(m, h = f^k_{ca}(r), r)$ in $L_{\mathcal{H}}$, and return $\sigma = r$ as the signature to the adversary $\mathcal{A}$. In the eye of $\mathcal{A}$, $\sigma = r$ is valid, as it satisfies the verification equation.

However, if $m = m^*$, the corresponding record in $L_{\mathcal{H}}$ is $(m^*, h^* = f^{k-1}_{ca}(S_2^{t+1}), \sqcup)$, we cannot return a valid value. Thus, we have to terminate the game and report failure.

Unless the signing query fails, this game is indistinguishable from the previous game. Therefore,

$$\Pr[\mathbf{Forge}_2] = (1 - \frac{1}{q_h + q_s + 1})^{q_s} \Pr[\mathbf{Forge}_1]$$

GAME $\mathbf{G_3}$. In this game, we take a close look at the valid forgery $(\sigma^\star, m^\star)$. If $m^\star \neq m^*$, we have to terminate the game again, as the returned is irrelevant to the implanted challenge. However, $m^\star = m^*$, we can convert the adversary $\mathcal{A}$'s capability to solve the challenge " given f_{ca} and S_2^{t+1}, where $S_2^{t+1} = f_{ca}(S_2^t)$, compute S_2^t" as follows.

Since $\sigma^\star$ is valid, i.e., $f_{ca}^k(\sigma^\star) = f_{ca}^{k-1}(S_2^{t+1})$, we calculate $S_2^t = \sigma^\star$ as the challenge. Therefore, we have

$$\mathbf{Succ}_{\mathcal{A}}^{\mathsf{LCAR}} = \Pr[\mathbf{Forge}_3]$$

By observing $\mathbf{G_3}$ and $\mathbf{G_2}$, we can see $\mathbf{G_3}$ won't terminate unless $m^\star = m^*$. Therefore, we have

$$\Pr[\mathbf{Forge}_3] = \frac{1}{1+q_h}\Pr[\mathbf{Forge}_2]$$

As mentioned in **GAME $\mathbf{G_0}$**, the success probability of attacking the signature scheme is

$$\mathbf{Succ}_{\mathsf{DS},\mathcal{A}}^{\mathsf{EF\text{-}CMA}} = Pr[\mathbf{Forge}_0] = \epsilon.$$

By combining all above games, we have

$$\begin{aligned}\epsilon' =& \mathbf{Succ}_{\mathcal{A}}^{\mathsf{LCAR}} = \Pr[\mathbf{Forge}_3]\\ =& \frac{1}{1+q_h}\Pr[\mathbf{Forge}_2]\\ =& \frac{1}{1+q_h}\left(1-\frac{1}{q_h+q_s+1}\right)^{q_s}\Pr[\mathbf{Forge}_1]\\ \approx& \frac{1}{q_h+q_s+1}\cdot\epsilon\end{aligned}$$

Besides, there are total $q_h + q_s + 1$ operations of computing $f_{ca}^k(\cdot)$ in the above games, thus it costs $(q_h + q_s + 1)\Theta$. Plus the time τ of running the adversary $\mathcal{A}$, the time τ' of resolving LCAR problem is bounded by $\tau + (q_h + q_s + 1)\Theta$ in the end. Thus, this completes the proof. □

5 A Simple Example

In this section, we give a simple example to demonstrate the feasibility of our proposed signature scheme. We use the transition rules of three 1D RCAs to generate the rules of a 2D CA with T-shaped neighborhood. And set the reverse rules of the 1D transition rules as the private key and the constructed 2D rules as the corresponding public key. Then we use the private key to generate signature and the public key to verify whether the signature is valid or not.

- **KG**: First, we choose a security parameter $k = 3$, in fact, k should be a large number, but here we only choose a small one to simply demonstrate our scheme. We define three 1D RCAs, labeled CA_1, CA_2 and CA_3, and $CA_i = (1, S_1, N_r, f_i)$, for $i \in \{1, 2, 3\}$, where state set $S_1 = \{0, 1, 2, 3\}$ and N_r is the neighboring state set with radius $r = 1/2$. The reversible transition rule $f_i : S_1 \to S_1$, $i \in \{1, 2, 3\}$ is shown in Table 1.

  It can be proved that rule f_1, f_2 and f_3 all self-reversible, i.e. the reverse of f_i is itself, $f_i^{-1} = f_i$. Since the radius of 1D RCA is 1/2, there is only one neighbor in its neighborhood besides itself, for each cell in 1D RCA, we set the right one besides it as its neighbor. In addition, we specify a direction for each f_i, it is shown in Figure 5.

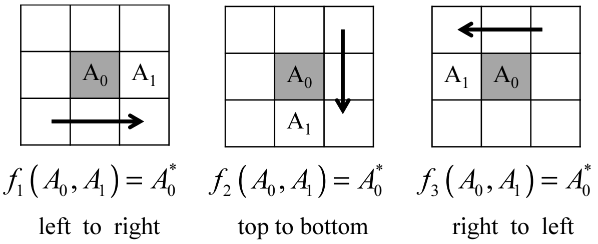


Figure 5: The neighborhood and direction of three 1D rules, where A_0 is the state of central cell and A_1 is the state of its neighbor, A_0^* is the new state of central cell.

  Then we define a 2-layer CA also with periodic boundary, denoted by $CA' = (2, S_2, N', f_{ca})$, where $S_2 = \{0, 1, 2, 3\}$ and the neighborhood structure of CA' is set as T-shaped neighborhood with radius $r' = 1$. The transition rule f_{ca} constructed by the compound operations of transition rules f_i, $i \in \{1, 2, 3\}$, i.e.

$$f_{ca} = f_1 \circ f_2 \circ f_3$$

  Specifically, we first define a $S_{2,(i,j)}^t$, where

$$S_{2,(i,j)}^t = (s_{2,(i,j-1)}^t, s_{2,(i,j)}^t, s_{2,(i,j+1)}^t, s_{2,(i+1,j)}^t)$$

  denotes the configuration of the cell at i-th row j-column at time t, and $s_{2,(i,j)}^t \in S_2$. Take each possible configuration $S_{2,(i,j)}^t$ as the input of the compound operations $f_{ca} = f_1 \circ f_2 \circ f_3$, the corresponding output is the new state of the central cell.

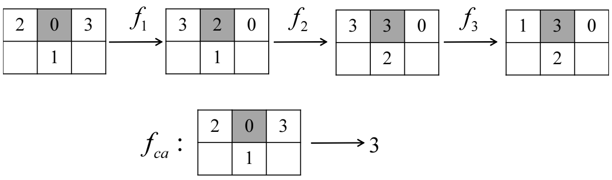


Figure 6: A example of generating a 2D transition rule

  There is a concrete example of the rule generation process in Figure 6, and $f_{ca} : 2031 \to 3$ is a 1-radius 2D rule. Table 2 shown some 2D rules generated in this algorithm.

  Next, we define a function $F_{ca} : S_1 \to S_1$, where $F_{ca} = f_3^{-1} \circ f_2^{-1} \circ f_1^{-1}$, and set it as the private key

Table 1: The reversible transition rules of three 1D RCAs

s_i^{t+1} (s_i^t, s_{i+1}^t)	f_1	f_2	f_3
00	1	0	2
01	0	1	1
02	3	0	3
03	2	2	0
10	0	1	3
11	2	0	0
12	2	2	1
13	3	3	2
20	3	3	0
21	1	3	3
22	1	1	2
23	0	0	1
30	2	2	1
31	3	2	2
32	0	3	0
33	1	1	3

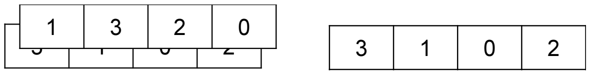
Table 2: Part of the generated 1-radius 2D rules, $f_{ca}: S^t_{2,(i,j)} \rightarrow s^{t+1}_{2,(i,j)}$

$S^t_{2,(i,j)} \rightarrow s^{t+1}_{2,(i,j)}$	$S^t_{2,(i,j)} \rightarrow s^{t+1}_{2,(i,j)}$	$S^t_{2,(i,j)} \rightarrow s^{t+1}_{2,(i,j)}$
0133 → 3	1232 → 3	3221 → 2
1321 → 2	2330 → 3	2222 → 3
3200 → 0	1313 → 2	2210 → 0
2012 → 0	3121 → 3	2121 → 2
2311 → 0	3202 → 1	1122 → 1
3103 → 1	2033 → 0	1202 → 0
1022 → 1	0313 → 1	2012 → 0
0230 → 0	3121 → 3	1012 → 2

of the signature scheme, i.e. $sk = F_{ca}$, set f_{ca} as the public key pk, i.e. $pk = f_{ca}$.

Besides, we define a secure one-way hash function H, where $H: \{0,1\}^* \rightarrow \mathbb{S}$, $\mathbb{S}$ denotes the message space of the signature scheme.

- **SG**: For a message $m = 01100101$ and security parameter $k = 3$, we first compute $R_1 = H(m) = 13203102$, and arrange the 13203102 into a 2-layer CA, which has four cells in each layer, as shown in Figure 7. Then, for each cell in the 2-layer CA, taking its configuration as the input of the function $F_{ca} = f_3^{-1} \circ f_2^{-1} \circ f_1^{-1} = f_3 \circ f_2 \circ f_1$, and evolved $k = 3$ times, the corresponding output is its new state, all new states make up of the signature. So, the signature of m is $\sigma = F^3_{ca}(R_1) = 12322113$.



(a). The structure of 2-layer CA

(b). The second layer

Figure 7: 2-layer CA

- **SV**: For the message $m = 01100101$ and a signature $\sigma = 12322113$, together with the public key $pk = f_{ca}$ and security parameter $k = 3$, we compute R'_1, where

$$R'_1 = f^3_{ca}(\sigma) = f^3_{ca}(12322113) = 13203102.$$

It is obvious that the following equality is hold, so the signature $\sigma = 12322113$ is valid.

$$R'_1 = H(m).$$

6 Comparison

The simple example in the last section has shown the feasibility of the proposed signature scheme. In this section, we will exhibit its strengths by giving a comparison between the proposed scheme and RSA signature algorithm.

Table 3: The key space and timing analysis between the proposed scheme and RSA, where 1D and 2D denote the 1D and 2D CAs used in **KG** algorithm, respectively

	State number	Radius		Key space	Time (ms)
		1D	2D		
The proposed scheme	2	$r = 1$	$r' = 1$	2^{16}	
			$r' = 2$	2^{128}	15.6
			$r' = 3$	2^{1024}	218.4
	4	$r = \frac{1}{2}$	$r' = 1$	2^{512}	31
			$r' = 2$	2^{32768}	4274
RSA				2^{1024}	4708

Since the number and radius of the CAs in our proposed signature scheme are not appointed, we can achieve different size key space by changing the radius and state

Table 4: Average execution time for the proposed scheme and RSA-1024

	proposed scheme	RSA-1024
Signature	3.91 ms	4.26 ms
Verification	2.35 ms	2.77 ms

number. In the proposed scheme, we use n reversible rules of some 1D CAs with r-radius to generate the transition rule of a 4-state and r'-radius 2D CA, so there will be $4^{4^{(3r'+1)}}$ possible rules generated as the public key, i.e. the size of the key space is $4^{4^{(3r'+1)}}$. Of course, the state number can be changed according to the concrete applications. Table 3 shows the different key space size and the time of generating the public key with the state number and radius r and r' changed. It's observed that we can get a larger key space in less time when compared with RSA-1024.

Because the key space of the RSA-1024 algorithm is 2^{1024}, as well as the plaintext space and the ciphertext space, here we set $n = 4$, $r' = 3$ and the state number is 2 such that the key space of the proposed scheme is $2^{2^{(3r'+1)}} = 2^{1024}$. Now, we randomly choose 100 messages from the plaintext space and sign them to get the signatures, and then verify these signatures. All the signature and verification processes are executed by the RSA-1024 and the proposed scheme on an Intel Core 2 Duo 2.0 GHZ, in C++ platform. The average execution time of the 100 signature and verification processes are calculated separately and the results are tabulated in Table 4. It's obvious that the time taken by our proposed signature scheme is less than RSA-1024 algorithm, which obviously shows the efficiency of our proposed signature scheme.

7 Conclusions

In this paper, we have formally defined the digital signature, then proposed a new CA-based digital signature scheme based on the hardness assumption of the LCAR problem. We use the transition rules of some 1D RCAs to construct the transition rules of a 2D CA, as the reversibility of 2D CA is undecidable, we set the constructed 2D transition rules as the public key, the rules of 1D RCAs as the private key. And we have formally shown the proposed signature scheme is semantically secure against chosen-message attacks in the oracle model. Moreover, the proposed scheme is developed with a simple example, and analysis of the key space and efficiency are also carried out along with RSA-1024 algorithm, the results show that the proposed signature scheme is more efficient than RSA-1024.

References

[1] A. A. Abdo, S. Lian, I. A. Ismail, M. Amin, and H. Diab, "A cryptosystem based on elementary cellular automata," *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 1, pp. 136–147, 2013.

[2] R. Ayanzadeh, K. Hassani, Y. Moghaddas, H. Gheiby, and S. Setayeshi, "Multi-layer cellular automata for generating normal random numbers," in *IEEE 18th Iranian Conference on Electrical Engineering (ICEE'10)*, pp. 495–500, 2010.

[3] M. Bellare and P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," in *Proceedings of the 1st ACM Conference on Computer and Communications Security*, pp. 62–73, 1993.

[4] Z. Cinkir, H. Akin, and I. Siap, "Reversibility of 1D cellular automata with periodic boundary over finite fields $\mathbb{Z}_p$," *Journal of Statistical Physics*, vol. 143, no. 4, pp. 807–823, 2011.

[5] A. Clarridge and K. Salomaa, "A cryptosystem based on the composition of reversible cellular automata," in *Language and Automata Theory and Applications*, pp. 314–325, Springer, 2009.

[6] J. Coron, J. Patarin, and Y. Seurin, "The random oracle model and the ideal cipher model are equivalent," in *Advances in Cryptology (CRYPTO'08)*, pp. 1–20, Springer, 2008.

[7] D. Das and A. Ray, "A parallel encryption algorithm for block ciphers based on reversible programmable cellular automata," *arXiv preprint arXiv:1006.2822*, 2010.

[8] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," in *Advances in Cryptology*, pp. 10–18, Springer, 1985.

[9] S. Goldwasser and S. Micali, "Probabilistic encryption," *Journal of Computer and System Sciences*, vol. 28, no. 2, pp. 270–299, 1984.

[10] S. Goldwasser, S. Micali, and R. L. Rivest, "A digital signature scheme secure against adaptive chosen-message attacks," *SIAM Journal on Computing*, vol. 17, no. 2, pp. 281–308, 1988.

[11] H. Gutowitz, "Cryptography with dynamical systems," in *Cellular Automata and Cooperative Systems*, pp. 237–274, Springer, 1993.

[12] A. Jaberi, R. Ayanzadeh, and A. Z. Mousavi, "Two-layer cellular automata based cryptography," *Trends in Applied Sciences Research*, vol. 7, no. 1, pp. 68–77, 2012.

[13] N. Jamil, R. Mahmood, M. R. Zába, Z. A. Zukamaen, and N. I. Udzir, "An observation of cryptographic properties of 256 one-dimensional cellular automata rules," in *Informatics Engineering and Information Science*, pp. 409–420, Springer, 2011.

[14] J. Jin, "An image encryption based on elementary cellular automata," *Optics and Lasers in Engineering*, vol. 50, no. 12, pp. 1836–1843, 2012.

[15] J. Kari, *Cryptosystems Based on Reversible Cellular Automata*, Manuscript, Apr. 16, 1992. (`http://users.utu.fi/jkari/CACryptoScanned.pdf`)

[16] J. Kari, "Reversibility and surjectivity problems of cellular automata," *Journal of Computer and System Sciences*, vol. 48, no. 1, pp. 149–182, 1994.

[17] J. Kari, "Undecidable properties on the dynamics of reversible one-dimensional cellular automata," in *Proceedings of the First Symposium on Cellular Automata'Journées Automates Cellulaires'*, pp. 3–14, 2008.

[18] R. S. Katti and R. G. Kavasseri, "Nonce generation for the digital signature standard," *International Journal of Network Security*, vol. 11, no. 1, pp. 20–29, 2010.

[19] C. Yu Liu, C. C. Lee, and T. C. Lin, "Cryptanalysis of an efficient deniable authentication protocol based on generalized elgamal signature scheme," *International Journal of Network Security*, vol. 12, no. 1, pp. 58–60, 2011.

[20] R. Lu and Z. Cao, "A directed signature scheme based on rsa assumption.," *International Journal of Network Security*, vol. 2, no. 3, pp. 182–186, 2006.

[21] N. A. Moldovyan, "Blind signature protocols from digital signature standards," *International Journal of Network Security*, vol. 13, no. 1, pp. 22–30, 2011.

[22] NIST, *Advanced Encryption Standard*, Federal Information Processing Standard, FIPS-197, vol. 12, 2001.

[23] C. S. Rao, S. R. Attada, M. J. Rao, and K. N. Rao, "Implementation of object oriented encryption system using layered cellular automata.," *International Journal of Engineering Science & Technology*, vol. 3, no. 7, 2011.

[24] P. R. Piedras, "Cellular automaton public-key cryptosystem," *Complex Systems*, vol. 1, pp. 51–57, 1987.

[25] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[26] M. Seredynski and P. Bouvry, "Block encryption using reversible cellular automata," in *Cellular Automata*, pp. 785–792, Springer, 2004.

[27] S. Ho Shin, D. S. Kim, and K. Y. Yoo, "A 2-dimensional cellular automata pseudorandom number generator with non-linear neighborhood relationship," in *Networked Digital Technologies*, pp. 355–368, Springer, 2012.

[28] J. N. Rao, and A. C. Singh, "A novel encryption system using layered cellular automata," *International Journal of Engineering Research and Applications*, vol. 2, no. 6, pp. 912–917, 2012.

[29] X. Wang and D. Luan, "A novel image encryption algorithm using chaos and reversible cellular automata," *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 11, pp. 3075–3085, 2013.

[30] S. Wolfram, *A New Kind of Science*, Wolfram Media Champaign, 2002.

Xing Zhang received the B.S.degree from Xuchang University, China, in 2010. From 2010 to now, she is working her Ph.D. degree in Computer Application from Nanjing University of Science and Technology (NUST), Jiangsu, China. During the period from November 2013 to May 2014, she was also a visiting Ph.D. student at the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. Her research interests include information security and cryptography, and the encryption scheme based on cellular automata.

Rongxing Lu received the Ph.D degree in computer science from Shanghai Jiao Tong University, Shanghai, China in 2006 and the Ph.D. degree (awarded Canada Governor General Gold Medal) in electrical and computer engineering from the University of Waterloo, Waterloo, Ontario, Canada, in 2012. Since May 2013, he has been with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, as an Assistant Professor. His research interests include computer, network and communication security, applied cryptography, security and privacy analysis for vehicular network, eHealthcare system, and smart grid communications. He won the IEEE Communications Society (ComSoc) Asia Pacific (AP) Outstanding Young Researcher Award in 2013.

Hong Zhang is a professor in the Department of Computer Science, Nanjing University of Science and Technology. His current interests are in the areas of theory and technology of information security, data mining and network fault diagnosis.

Chungen Xu received the M.S. degree from East China Normal University, Shanghai, China, in 1996 and the Ph.D degree from Nanjing University of Science and Technology in 2003. He is a professor in the Department of Applied Mathematics, School of Sciences, Nanjing University of Science and Technology. His current interests are in the areas of computer and network security, cryptography and coding.