SPCA의 상태전이 행동분석†

조성진*· 최언숙**· 황윤희*· 권민정*· 임지미*

*부경대학교· ** 동명대학교

Analysis of state transition behavior of SPCA

S.J. Cho*· U.S. Choi**· Y.H. Hwang* · M.J. Kwon* · J.M. Lim*

*Pukyong National University ·** Tongmyong University

E-mail : sjcho@pknu.ac.kr

요 약

본 논문에서는 Self-Programmable Cellular Automata(SPCA)를 기반으로 하여 최대주기수열을 생성하는 방법을 제안한다. 최근 긴 주기의 수열을 생성하기 위해서 CA의 차원을 증가시키거나, 상태전이 단계마다 각 셀의 전이규칙을 바꾸는 PCA를 사용하여 왔다. 본 논문에서는 셀의 전이 규칙에 역동성을 부여하기 위하여 각 셀의 상태전이 규칙을 각 시간 단계마다 바꿈으로써 더 긴 주기의 수열을 생성하고 생성된 수열의 랜덤성을 높이고자 한다.

ABSTRACT

In this paper we propose the method of generating the maximum length sequence based on Self Programmable Cellular Automata. Recently there is a research trend such that increased CA dimensionality and PCA which applies different rules on the same cell at different time steps can make a sequence with a long period. By changing a cell's state transition rules to give the cell dynamic energy at each time step, we can make the period of a sequence longer and the randomness of a sequence higher.

키워드

셀룰라오토마타, PCA, SPCA, 전이규칙, 의사난수열생성

Ⅰ. 서 론

셀룰라 오토마타(이하 CA)는 1986년 말에 Wolfram에 의하여 처음으로 의사 난수열 생성기로 사용되었다. 이러한 CA는 국소적 상호작용에 의해 상태가 전이되고 간단하고 규칙적이며 작은 단위로 확장 연결할 수 있는 구조를 가지고 있어 VLSI 구현에 적합하며 LFSR에 비하여 난수성이 뛰어나다. 이제 CA는 의사난수생성기(PRNG)로 사용되는 일반적인 방법 중 하나가 되고 있다[1].

그 동안 가장 가까운 3-이웃에 의존하는 1차원 CA PRNG가 폭넓게 연구되어지다가[2-12] CA의 차원을 증가시키거나 Programmable CA(PCA)를 사용하여 더 긴 주기의 수열을 만들어내기 위한 연구가 많아지고 있다. 2차원 CA PRNG의 난수성이 1차원 CA PRNG의 난수성에 비해 뛰어나기 때문에 더 많은 주목을 받고 있다[13, 14]. 그러나 설계상에서의 복잡함과 조작에 있어서의 효율성을 고려하면 어떤 것이 더 좋다고 말하기 어렵다. 또한 그 크기가 큰 경우에도 2차원 CA PRNG에 비해 1차원 CA PRNG를 구현하기가 쉽다. 지금까지의 연구는 CA의 차원과 전이규칙의 조합을 이용하여 랜덤성을 높이는 것에 관심을 가졌으며 최근에는 셀의 전이규칙에 역동성을 부여하는 PCA가 더 효과적인 PRNG로 주목받고 있다.

본 논문에서는 각 시간단계마다 적용되는 셀의 상태전이 규칙을 바꾸는 self-programmable CA(SPCA)를 정의하고, 이를 이용하여 더 긴 주기의 수열을 만들어내고, 생성된 수열의 랜덤성을 높이고자 한다.

†이 논문은 2009년도 정부재원(교육과학기술부 인문사회연구역량강화사업비)으로 한국학술진흥재단의 지원을 받아 연구되었음(KRF-2009-371-B00008)

Ⅱ. 배경지식

일차원으로 배열되어 있는 각 셀들의 위치를 i, 각 시간단계를 t, 시간단계 t에서 i번째 셀의 상태를 $q_i(t)$로 표기하면 n개의 셀들을 선형으로 나열한 1차원 CA는 $\langle q_0(t), q_1(t), \cdots, q_n(t)\rangle$, $i\in\{0,1,2,\cdots,n\}$, $q_i\in\{0,1\}$로 표현할 수 있다. 각 시간단계마다 CA의 각 셀들은

현재 이웃에 적용되는 전이규칙에 따라 갱신된다.

3-이웃 CA의 상태전이함수는 아래와 같이 표현될 수 있다.

$$q_i(t+1) = f[q_{i-1}(t), q_i(t), q_{i+1}(t)]$$

여기서 함수 f는 CA의 전이규칙으로 결합논리를 갖는 국소전이 함수이다. 본 논문에서 사용되는 전이규칙은 규칙90 과 150으로 이 규칙에 대한 결합논리는 다음과 같다.

- rule 90 : $q_i(t+1) = q_{i-1}(t) \oplus q_{i+1}(t)$
- rule 150 : $q_i(t+1) = q_{i-1}(t) \oplus q_i(t) \oplus q_{i+1}(t)$

n 개의 셀로 이루어진 n-셀 CA의 전이규칙이 XOR 또는 XNOR로만 이루어졌을 때 다음상태를 구하는 함수는 $n \times n$ 상태전이 행렬 $T=(t_{ij})$로 다음과 같이 나타낼 수 있다.

- $t_{ij}=1$: i번째 셀의 다음 상태가 현재 j번째 셀에 영향을 받는 경우
- $t_{ij}=0$: 그 외의 경우

CA의 다음 상태는 현재 상태 벡터와 행렬의 곱으로 얻어진다. 만약 $f_t(x)$가 시간 t인 순간 CA의 상태를 나타낸다면, 시간 $t+1$ 순간의 상태와 $t+2$ 인 순간의 상태는 아래의 식으로 표현될 수 있다 :

$$f_{t+1}(x) = T \cdot f_t(x)$$
$$f_{t+2}(x) = T \cdot f_{t+1}(x) = T^2 \cdot f_t(x)$$

같은 방법으로 p 단계 후의 상태는 다음과 같다.

$$f_{t+p}(x) = T^p \cdot f_t(x)$$

이렇게 하여 만들어진 T의 특성다항식 $p(x)$는 $p(x)=|T+xI|$이다. $p(x)$가 인수분해 되지 않는 n차 다항식이고 x^m-1을 나눌 때, m의 최솟값이 2^n-1인 다항식을 n차 **원시 다항식**이라 한다[9]. 주어진 CA가 최대길이를 갖는 CA거 되기 위한 필요충분조건은 전이행렬의 특성다항식이 원시다항식이라는 것이다.

규칙 90과 규칙 150은 자기 자신에 의존하느냐 하지 않느냐에 대해서만 차이가 난다. 따라서 각 셀에 하나의 제어선을 연결함으로써 각 시간 단계마다 같은 셀에 규칙 90과 규칙 150을 각각 적용할 수 있다.

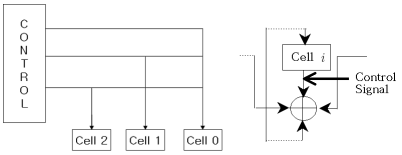

그림 2.1 3-셀 PCA 구조와 PCA 셀

따라서 k-셀 CA 구조는 2^k 개의 CA 형태를 구현할 수 있다. 여기서 그 스위치와 조절 프로그램을 적절히 조절함으로써 같은 구조에 서로 다른 CA를 적용할 수 있다. i번째 bit가 1(0) 상태이면 i번째 셀을 조절하는 스위치가 닫힌다(열린다). 이런 구조를 **Programmable CA(PCA)**라고 한다.

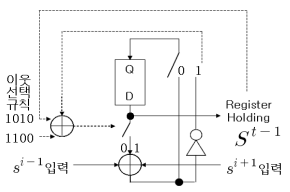

그림 2.2. SPCA 셀

그림 2.2는 90/150/165/105 규칙을 적용하는 **Self Programmable CA(SPCA)** 셀에 대한 구조이다. 일반적인 PCA와 마찬가지로 SPCA에는 셀의 상태를 갱신하기 위한 국소적인 상태 전이 이웃과 규칙을 정하기 위한 이웃을 선택하는 새로운 규칙이 있다. 상태 전이 이웃을 위해서 SPCA는 가장 가까운 3셀 이웃과 선형 규칙을 조합하여 사용하는데 이때 사용되는 선형 규칙은 그림의 점선에 의해 조절할 수 있다. SPCA의 복잡도는 규칙을 역동적으로 바꾸기 위해 이웃을 선택하는 부가적인 규칙 때문에 증가된다. 이렇게 이웃을 선택하는 새로운 규칙으로 선형규칙과 여원규칙 모두를 정의할 수 있다. 이웃을 선택하는 규칙의 output으로 셀의 상태를 조절하고 이미 설계된 두개의 상태전이 규칙들을 서로 바꾸게 한다. 150/105 또는 90/165와 같은 모든 여원 규칙 조합들을 사용하면 상태전이 이웃연결은 출력의 NOT gate에 의해서만 달라진다. 그와 동시에 150/105와 90/165의 상태 전이 이웃 연결은 그 셀 스스로에 의존하느냐 하지 않느냐의 여부에 따라 달라진다. 따라서 네 개의 규칙을 길이 n인 PCA에 적용하면 총 $4n$개의 서로 다른 hybrid CA 규칙 조합을 만들 수 있다.

III. PN 수열 생성

시간 t인 순간 n-셀 CA의 상태를 벡터 S^t로 표기하자. 각각의 시간이 단계적으로 진행되는 동안 CA의 상태는 연속적으로 $\langle S^0, S^1, S^2, \cdots \rangle$을 만들어 낸다. 이러한 방법으로 CA를 이용하여 반복되는 PRNG를 만들 수 있다.

$f(x)$를 $n(n \geq 2)$차 원시다항식이라 하고 T를 $f(x)$에 대응하는 90/150 CA의 전이행렬이라 하자. 다음 점화식을 만족하는 수열 $\{S^t\}$를 생각하자.

$$S^1 = TS^0 + S^0$$
$$S^t = TS^{t-1} + PS^{t-2} \ (t \geq 2) \qquad (1)$$

여기서 $P=T^u$, $u \in \{0,1,2,\cdots,2^n-2\}$이다. 그러면 이 점화식에 의해 생성된 수열 $\{S^t\}$는 다음과 같은 성질을 갖는다.

<정리 3.1> $\{S^t\}$의 한 주기 내에서 생성되는 항의 최대개수는 $2^{2n}-1$이다. □

정리 3.1에 의해 한 주기 내에서 생성된 $2^{2n}-1$개의 항들 중에는 같은 항이 반복되어 나타나기도 한다. 즉, $2^{2n}-1$보다 짧은 주기를 갖는 수열이 포함될 수 있다. 그런 수열들의 주기는 모두 $2^{2n}-1$의 약수이며 그중 가장 주기가 긴 최대주기 수열의 주기는 $2^{2n}-1$이 됨을 알 수 있다.

식 (1)에 의해서 생성되는 수열 $\{S^t\}$의 최대 주기가 $2^{2n}-1$ 이므로 $\langle S^0, S^1, S^2, \cdots \rangle$에 포함되면서 더 짧은 주기를 갖는 수열을 나타내기 위한 적절한 표현방법이 필요하다. 따라서

$$\{S^t\} = \{S^0, S^1, \cdots, S^{w-1}, S^0, S^1, \cdots, S^{w-1}, S^0, S^1, \cdots, S^{w-1}, \cdots\}$$

가 생성되었다면 이것을 $S^t := [S^0, S^1, \cdots, S^{w-1}]$로 쓰기로 하자.

이제 생성된 수열 $\{S^t\}$ 중에서 최대주기를 갖는 수열의 모양과 그 생성원리에 대해 살펴보자.

<정리 3.2> n-셀 PCA에 의해 생성된 수열 $\{S^t\}$가

$$\begin{aligned}&[S^0, S^1, \cdots, S^{2^n},\\ &PS^0, PS^1, \cdots, PS^{2^n},\\ &P^2S^0, P^2S^1, \cdots, P^2S^{2^n}, \cdots,\\ &\quad\vdots\\ &P^{2^n-2}S^0, P^{2^n-2}S^1, \cdots, P^{2^n-2}S^{2^n}]\end{aligned} \tag{2}$$

이면 $\{S^t\}$는 최대주기수열이다. 여기서 P의 주기는 2^n-1이다. □

<정리 3.3> 식(2)형태의 최대주기수열에서 $S^{2^n}=S^0$이다. □

<정리 3.4> 점화식 (1)에 의해 생성된 수열 $\{S^t\}$의 S^{2^n}번째 항의 모양은 다음과 같다.

$$S^{2^n} = (T^{2^n} + T^{2^n-1} + PT^{2^n-2} + P^2T^{2^n-4} + \cdots + P^{2^{n-2}}T^{2^{n-1}} + P^{2^{n-1}})S^0$$

$f(x)$가 n차 원시다항식이고 $f(T)=\mathbb{O}$을 만족하는 T에 대하여 $T^i \oplus T^j = T^k\ (0 \le i, j \le 2^n-2,\ i \ne j)$을 만족하는 $k(\in\{0, 1, 2, \cdots, 2^n-2\})$가 존재한다. T^k는 정칙이므로 T^k의 모든 열들은 일차독립이다. 또한 이 결과를 이용하면 $f(x)$가 n차 원시다항식이고 $f(T)=\mathbb{O}$을 만족하는 T에 대하여 $T^kS^0 \ne \mathbb{O}$임을 알 수 있다. (단, $S^0 \ne \mathbb{O}$ 이고 $k\in\{0, 1, 2, \cdots, 2^n-2\}$). 점화식 (1)에 의하여 생성되는 $\{S^t\}$의 항은 $i, j \in \{1, 2, \cdots, 2^n-1\}$인 i, j에 대하여 $T^iS^0+T^jS^0$, $\mathbb{O}+T^jS^0$, $T^iS^0+\mathbb{O}$의 형태를 갖는다. 따라서 최대주기수열 $\{S^t\}$에 $\mathbb{O}$가 나타나는 규칙을 발견할 수 있다.

<정리 3.5> 식(2) 형태의 최대주기수열 $\{S^t\}$의 원소들 $S^0, S^1, \cdots, S^{2^n}$에는 $\mathbb{O}$가 유일하게 존재한다. □

이제 P를 적당히 선택함으로써 최대주기수열을 생성하고자 한다. 정리 3.4의 $\mathbb{O}$ 존재성을 이용하면 최대주기수열을 생성하는 P에 대해 알 수 있다.

<정리 3.6> $P=T^2$인 경우의 $\{S^t\}$는 식(2) 형태의 최대주기수열이 될 수 없다. □

지금까지의 내용을 바탕으로 하여 n-셀 CA가 $2^{2n}-1$ 주기를 갖도록 하는 원시다항식과 행렬 P를 찾아보자.

<예제 3.7> (a) 2-셀 CA

정리 3.4에 의하면 $S^4=(T^2P+T+P^2+I)S^0=S^0$

이므로 $T^2P+T+P^2=\mathbb{O}$가 되어야 한다. 여기서 $P := T$로 두면 $T^3+T+T^2=T(T^2+T+I)=\mathbb{O}$

가 되어야 하므로 이 조건을 만족하는 2차 원시다항식은 $f(x)=x^2+x+1$이다.

(b) 4-셀 CA

같은 방법으로

$$S^{2^4}=(T^{2^4}+T^{2^4-1}+PT^{2^4-2}+P^2T^{2^4-4}+P^{2^2}T^{2^3}+P^{2^3})S^0$$

이므로 $T^{2^4-1}=I$, $S^{2^4}=S^0$를 이용하면

$$PT^{14}+P^2T^{12}+P^4T^8+P^8=T \tag{3}$$

이다. 이제 우리는 P를 적절히 선택함으로써 이것을 만족하는 P와 원시다항식을 찾으면 된다.

(i) $f(x)=x^4+x+1$ 인 경우 :

$T^4+T+I=\mathbb{O}$이다. 식(3)을 만족하는 P를 찾기 위해 $P := T$, $P := T^2$, $\cdots$ 순서로 찾아 나간다. 우선$P := T$인 경우를 보자. 그러면 식(3)의 오른쪽은 다음과 같이 된다.

$$\begin{aligned}&T^{15}+T^{14}+T^{12}+T^8\\ &=T^{14}+T^{12}+T^8+I\\ &=T^6(T^8+T^2+I)+T^{12}+T^6+I\\ &=T^4(T^8+T^2+I)\\ &=T^4+I\\ &=T\end{aligned}$$

그러므로 $P=T$로 두면 생성되는 $\{S^t\}$는 주기가 $2^{2\cdot 4}-1=255$인 최대주기수열이 된다. 따라서 $P=T$는 최대주기를 생성하는 P 중 하나이다.

(ii) $f(x)=x^4+x^3+1$ 인 경우 :

(i)과 같은 방법으로 최대주기수열을 생성하는 P를 찾으면 $P=T^4, T^8, T^{11}, T^{13}, T^{14}$ 이다.

<정리 3.8> 점화식 (1)에 의해 생성되는 수열의 항들 $S^0, S^1, \cdots, S^{2^n}$ 중 $S^k=S^{k+1}$, $(1 \le k \le 2^n-1)$이며 P가 다음을 만족하면 $\{S^t\}$는 식(2) 형태의 최대주기수열이다.

$$\begin{aligned}&S^{(k+2)+(2^n+1)} = PS^{k+2} \bmod f(x)\\ &P=T^{2^i}(0 \le i \le n-1),\ f(T)=\mathbb{O}\end{aligned}$$ □

정리 3.8에 의하여 최대주기수열을 생성하는 P는 n차 원시다항식의 켤레근들 중 하나임을 알 수 있다. 따라서 최대주기수열을 생성하기 위한 P를 선택하는데 걸리는 시간은 $O(n)$이다.

<정리 3.9> 주기가 $2^{2n}-1$인 $\{S^t\}$의 각 열은 최대주기를 갖는 어떤 $2n$-셀 90/150 CA에 의해서 생성된 수열과 같다. □

이제 정리 3.9에 의하여 2차의 원시다항식을 이용하여 4차의 원시다항식으로 만들 수 있는 긴 주기의 PN 수열을 생성할 수 있다. 차수가 높은 경우에는 아주 의미 있는 결과라고 할 수 있다.

<예제 3.10> $f(x)=x^2+x+1$이고 $T=\begin{pmatrix}0&1\\1&1\end{pmatrix}$라 하자. 여기서 $P=T$, $S^0=(0, 1)^t$로 두고 계산한 결과는

$S^0=(0,1)^t$ $S^6=S^0=(0, 1)^t$ $S^{12}=TS^0=(1, 1)^t$

$S^1=T^2S^0=(1, 0)^t$ $S^7=S^0=(0, 1)^t$ $S^{13}=\mathbb{O}=(0, 0)^t$

$S^2=T^2S^0=(1, 0)^t$ $S^8=\mathbb{O}=(0, 0)^t$ $S^{14}=T^2S^0=(1, 0)^t$

$S^3=\mathbb{O}=(0, 0)^t$ $S^9=TS^0=(1, 1)^t$ $S^{15}=S^0=(0, 1)^t$

$S^4=S^0=(0, 1)^t$ $S^{10}=T^2S^0=(1, 0)^t$

$S^5=TS^0=(1, 1)^t$ $S^{11}=TS^0=(1, 1)^t$

이다. 이렇게 생성된 벡터들을 옆으로 나열하면

011001000111101

100011110101100

이 되는데 이것은 4차의 원시다항식 x^4+x^3+1 으로 생성할 수 있는 주기 15인 수열이다.

<예제 3.11> $T=<0, 1, 1>$라 하면 특성다항식은 $f(x)=x^3+x+1$이다.

1. $P=T^3$: 주기 7인 수열 $\{S^t\}$ 생성

$S^0=S^0$

$S^1=TS^0+S^0=(T+I)S^0=PS^0$

$S^2=TS^1+PS^0=T^6S^0=P^2S^0$

$S^3=TS^2+PS^1=T^2S^2=P^3S^0$

$S^4=TS^3+PS^2=T^5S^0=P^4S^0$

$S^5 = TS^4 + PS^3 = TS^0 = P^5 S^0$
$S^6 = TS^5 + PS^4 = T^4 S^0 = P^6 S^0$
$S^7 = P^7 S^0 = S^0$

2. $P = T$: 주기 $2^{2 \cdot 3} - 1 = 63$인 수열 $\{S^t\}$ 생성

위와 같은 방법을 사용하여 $P = T^i$로 두었을 때 최대 주기를 생성여부를 알아보았다.

i	1	2	4
$x^3 + x + 1$	○		
$x^3 + x^2 + 1$			○

i	1	2	4	8
$x^4 + x + 1$	○			○
$x^4 + x^3 + 1$			○	○

i	1	2	4	8	16
$x^5 + x^2 + 1$				○	
$x^5 + x^3 + 1$					○
$x^5 + x^4 + x^3 + x^2 + 1$			○	○	
$x^5 + x^3 + x^2 + x + 1$	○				○
$x^5 + x^4 + x^3 + x + 1$	○				
$x^5 + x^4 + x^2 + x + 1$			○		

i	1	2	4	8	16	32
$x^6 + x + 1$	○					○
$x^6 + x^5 + 1$			○	○		
$x^6 + x^5 + x^4 + x + 1$			○			○
$x^6 + x^5 + x^2 + x + 1$	○			○		
$x^6 + x^5 + x^3 + x^2 + 1$			○			○
$x^6 + x^4 + x^3 + x + 1$	○			○		

Ⅳ. 결론

본 논문에서 우리는 셀의 상태전이 규칙을 각 단계마다 다르게 적용하도록 변형함으로써 n-셀 90/150 CA를 이용하여 $2n$-셀 90/150 CA에 의해 생성된 수열과 같은 주기를 갖는 수열을 생성하는 알고리즘을 제안하였다. 이 관계식을 이용하여 생성된 수열이 모두 최대 주기수열이 되는 것은 아니지만 n-셀 90/150 CA를 이용하여 $2^{2n} - 1$ 주기를 갖는 수열이 생성될 수 있음을 발견한 것과 생성된 수열 중 최대주기를 갖는 수열의 알고리즘 및 그 원리의 타당성을 보인 것의 의의는 크다고 사료된다.

참고문헌

1. P. Sarkar, "A brief history of cellular automata", ACM Comput. Surveys, vol. 32, no. 1, pp. 80-107, 2000.
2. I. Kokolakis, I. Andreadis, and Ph. Tsalids, "Comparison between cellular automata and linear feedback shift registers based pseudo-random number generators", Microprocessors and Microsyst., vol. 20, pp. 643-658, 1997.
3. M. Matsumoto, "Simple cellular automata as pseudorandom m-sequence generators for built-in-self-test", ACM Trans. Modeling and Comput. Simul., vol. 8, no. 1, pp. 31-42, 1998.
4. M. Mihaljevic, "Security examination of a cellular automata based pseudorandom bit generator using an algebraic replica approach", in Proc. Applied Algebra, Algorithms and Error Correcting Codes, vol. 1255, Lecture notes in Computer Sciences, pp. 250-262, 1997.
5. M. Mihaljevic and H. Imai, "A family of fast keystream generators based on programmable linear cellular automata over GF(q) and time-variant table", IEICE Trans. Fundamentals, vol. E82-A, no. 1, pp. 32-39, 1999.
6. M. Tomassini, M. Sipper, M. Zolla, and M. Perrenoud, "Generating high-quality random numbers in parallel by cellular automata", Future Gen. Comput. Syst., vol. 16, pp. 291-305, 1999.
7. P. D. Hortensius, R. D. Mcleod, and H. C. Card, "Parallel random number generation for VLSI system using cellular automata", IEEE Trans. Comput., vol. 38, pp. 1466-1473, 1989.
8. P. D. Hortensius, R. D. Mcleod, W. Pries, D. M. Miller, and H. C. Card, "Cellular automata-based pseudorandom number generators for built-in self-test", IEEE Trans. Computer-Aided Design, vol. 8, pp. 842-859, 1989.
9. P. P. Chaudhuri, D. R. Chowdhury, S. Nandi, and S. Chattopadhyay, Additive Cellular Automata: Theory and Applications. Los Alamitos, CA: IEEE CS Press, vol. 1., 1997.
10. P. H. Bardell, "Analysis of cellular automata used as pseudorandom pattern generators", in Proc. Int. Test Conf., pp. 762-768, 1990.
11. S. Nandi, B. K. Kar, and P. P. Chaudhuri, "Theory and applications of cellular automata in cryptography", IEEE Trans. Comput., vol. 43, pp. 1346-1357, 1994.
12. S. Wolfram, "Cryptography with cellular automata", in Proc. CRTPTO 85-Advances in Cryptography, vol. 218, Lecture Notes in Computer Science, pp. 429-432, 1985.
13. D. R. Chowdhury, I. S. Gupta, and P. P. Chaudhuri, "A class of two-dimensional cellular automata and applications in random pattern testing", J. Elect. Testing : Theory and Applic., vol. 5, pp. 65-80, 1994.
14. M. Tomassini, M. Sipper, and M. Perrenoud, "On the generation of high-quality random numbers by two-dimensional cellular automata", IEEE Trans. Comput., vol. 49, pp. 1146-1151, 2000.
15. S. Wolfram, Statistical mechanics of cellular automata, Rev. Mod. Phys., vol. 55, No. 3, pp.

601-644, 1983.