Randomness with CA

Bruno Martin

Université Côte d'Azur, I3S-CNRS

Journée Aléa I3S

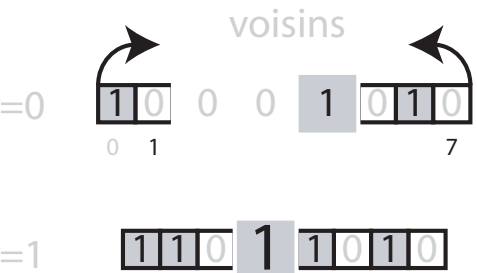
Contents

Cellular Automata

[Ulam and von Neumann, 1950] for self-reproduction.
Here: finite 1-dimensional binary CA:

Definition

A *CA* is a finite array of cells. Each cell is a FSM $C = (\mathbb{F}_2, f)$ where $\mathbb{F}_2$ is the set of states and f a mapping $f : \mathbb{F}_2^3 \to \mathbb{F}_2$.



Later, f will be a mapping $f : \mathbb{F}_2^4 \to \mathbb{F}_2$ or $f : \mathbb{F}_2^5 \to \mathbb{F}_2$.

Representation Forms for CA

$(x_{i-1}^t x_i^t x_{i+1}^t)$	111	110	101	100	011	010	001	000
x_i^{t+1}	0	1	0	1	1	0	1	0

Wolfram Numbering: Bin(90)= 01011010, truth table of f
Hexadecimal: `5A`, truth table of f
Boolean function: $x_{i-1}\text{XOR}\, x_{i+1}$
ANF: $x_{i-1} \oplus x_{i+1}$ or $1 + 3$ (with algebraic degree 1)

Representations generalize to rules of wider radius

Definition (ANF)

$f : \mathbb{F}_2^n \to \mathbb{F}_2$ is uniquely represented by a n-variable binary polynomial: $f(x) = \bigoplus_{u \in \mathbb{F}_2^n} a_u (\prod_{i=1}^n x_i^{u_i})$.
The *algebraic degree* of f is its *ANF degree*.

Walsh Transform & Randomness

Walsh transform $\widehat{f}$ of f is defined over $\mathbb{F}_2^n$ by

$$\widehat{f}(u) = \sum_{x \in \mathbb{F}_2^n} (-1)^{f(x) \oplus u \cdot x}$$

Used to test PRG.

[Yuen, 1977]: a truly random sequence has an asymptotically flat Walsh power spectrum.

Property: $\hat{f}(0) = E[f(x)] = 2^{n-1}$; tests if f *balanced*.

Correlation Testing

In crypto: study correlation-immunity (CI) of Boolean functions.

[Xiao and Massey, 1988] link together CI and WT.

Theorem

A function $f : \mathbb{F}_2^n \to \mathbb{F}_2$ *is* k*-correlation-immune (CI(*k*)) iff* $\hat{f}(u) = 0 \ \forall u = (u_0, \cdots, u_{n-1}) \neq 0$ *with* $w_H(u) \leq k$.

WT computes correlations between inputs and outputs.
Great interest: quasi-linear time computation

Definition

CI(k) + balanced = k-resilient (R(k))

Boolean Functions

Definition (equivalent BF)

f and g Boolean functions with n variables are *equivalent* iff

$$f(x) = g\left((x \cdot A) \oplus a\right) \oplus \left(x \cdot B^T\right) \oplus b, \quad \forall x \in \mathbb{F}_2^n \tag{1}$$

A is a non-singular binary $n \times n$ matrix, $b \in \mathbb{F}_2$, $a, B \in \mathbb{F}_2^n$.

Theorem (Siegenthaler bound)

For a R(k) BF with n variables ($0 \leq k < n-1$), there is an upper bound for its algebraic degree d:
$d \leq n-k-1$ if $k < n-1$ and $d = 1$ if $k = n-1$.

Radius 1 CA Rules

Siegenthaler's bound with $n = 3$ variables, $k = 1$-resiliency provides an algebraic degree $d \leq n - k - 1 = 1$.
Only linear functions are 1-resilient.

Theorem

There is no non-linear radius 1 CA rule which is resilient.

The same is obtained through rules exploration via WT [Martin, 2008].

What are the other ways to get randomness with CAs?

- Switch to non-uniform *hybrid* CA
- Increase the neighborhood for uniform CA

Linear Hybrid CA

HCA combine different rules.
LHCA combine linear rules (e.g. 90 and 150) with null boundary conditions.
LHCA are specified by the *rule vector* that tells which cells use rule 90 and which use rule 150. $M = [d_0, d_1, \ldots, d_{N-1}]$ s.t.

$$d_i = \begin{cases} 0 & \text{if cell } i \text{ uses rule 90} \\ 1 & \text{if cell } i \text{ uses rule 150} \end{cases}$$

New dynamics: $x_i^{t+1} = f_i(x_{i-1}^t x_i^t x_{i+1}^t) = x_{i-1}^t + d_i x_i^t + x_{i+1}^t \mod 2$

LHCA 90/150

In $x_i^{t+1} = f_i(x_{i-1}^t x_i^t x_{i+1}^t) = x_{i-1}^t + d_i x_i^t + x_{i+1}^t$, since f_i is linear
$\Rightarrow F$ its global function is also linear (endomorphism of $\mathbb{F}_2^N$).
There is a *HCA* matrix A s.t. $x^{t+1} = F(x^t) = A \cdot x^t$
(it plays the same role as an LFSR transition matrix)

$$A = \begin{pmatrix} d_0 & 1 & 0 & \cdots & \cdots & 0 & 0 \\ 1 & d_1 & 1 & \ddots & & & 0 \\ 0 & 1 & d_2 & \ddots & \ddots & & \vdots \\ \vdots & & & & 1 & d_{N-2} & 1 \\ 0 & 0 & \cdots & \cdots & 0 & 1 & d_{N-1} \end{pmatrix}$$

Δ denotes the characteristic polynomial, or HCA polynomial

Results on LHCA [Cattell and Muzio, 1998]

Theorem

Let $p \in \mathbb{F}_2[x]$ of degree n. Then p is a HCA polynomial iff for some solution q for y of the congruence

$$y^2 + (x^2 + x)p'y + 1 \equiv 0 \mod p \tag{2}$$

Euclid's algorithm on p and q results in n degree 1 quotients.

Theorem

If $p \in \mathbb{F}_2[x]$ irreducible of degree n, then eq. (2) has exactly two solutions, both of which result in n deg. 1 quotients.

$d^\circ 0$ coefs in the quotients give the d_i values. This only gives necessary conditions for HCA polynomials.

Corollary

If $p \in \mathbb{F}_2[x]$ irreducible, then p has exactly two HCA realizations with one being the reversal of the other.

Similarity Transform Between LHCA and LFSR

[Cattell and Muzio, 1998] provide a similarity tranform which provides explicit mappings between the states of a LHCA and the states of a LFSR.

Thus, we inherit of the work done on LFSR for LHCA, in particular for generating PRS with LFSR.

Similarity Transform Between LHCA and LFSR

[Cattell and Muzio, 1998] provide a similarity tranform which provides explicit mappings between the states of a LHCA and the states of a LFSR.

Thus, we inherit of the work done on LFSR for LHCA, in particular for generating PRS with LFSR.

But LHCA sequences are predictable (since they are linear). Massey-Berlekamp's algorithm is able to recover the characteristic polynomial of a LFSR from the binary sequence.

Cellular Programming Approach

[Sipper and Tomassini, 1996]: genetic algorithm for selecting the rules used in a radius 1 HCA.
Their fitness function depends upon Koza's *entropy*

$$E_h = -\sum_{j=1}^{k^h} p_{h_j} \log_2 p_{h_j}$$

- k = number of possible values per sequence position
- h a subsequence length
- p_{h_j} is a measured probability of occurrence of a sequence h_j in a PRS

Best rules: 90, 105, 150 and 165 (all linear).
Tests: χ^2, serial correlation coefficient, entropy and MC

HCA With More Neighbors, Genetic Algorithm

- [Seredynski et al., 2004]: generalization of the cellular programming approach to 5-variable updating functions.
- Use of both 3 and 5-variable rules in HCA.
- Best rules: 30, 86, 101 and 869020563, 1047380370, 1436194405, 1436965290, 1705400746, 1815843780, 2084275140 and 2592765285.
- Tests: statistical tests required by the FIPS 140-2 standard and the Marsaglia tests.

4-Variable Local Functions

There are $2^{16} = 65536$ 4-variable CA rules.

A BF in 4 variables is represented by an integer $\{0, ..., 65535\}$.

200 non-linear $R(1)$ quadratic functions (Siegenthaler bound).

Divided into 8 equivalence classes by [Lacharme et al., 2008].

4-Variable 1-Resilient Rules

f	ANF	card.
34680	$12+3+4$	12
6120	$4+12+13+23$	8
7140	$2+4+12+13$	48
11730	$1+3+4+12$	24
34740	$2+3+4+12+42$	48
39318	$1+2+3+4+34$	12
7128	$3+4+12+31+42+43$	24
11220	$2+3+12+31+42$	24
		200

Can we find more with 5-variable local functions ?

Questions We Address

- Which are the rule transforms preserving resiliency?
- Which are the 1-resilient radius 2 CA rules?
- Which are the rules preserving resiliency upon iteration?

Just $R(1)$ since there are only 8 $R(2)$-BF in 5-variable.

Theoretical Results

Assumptions:

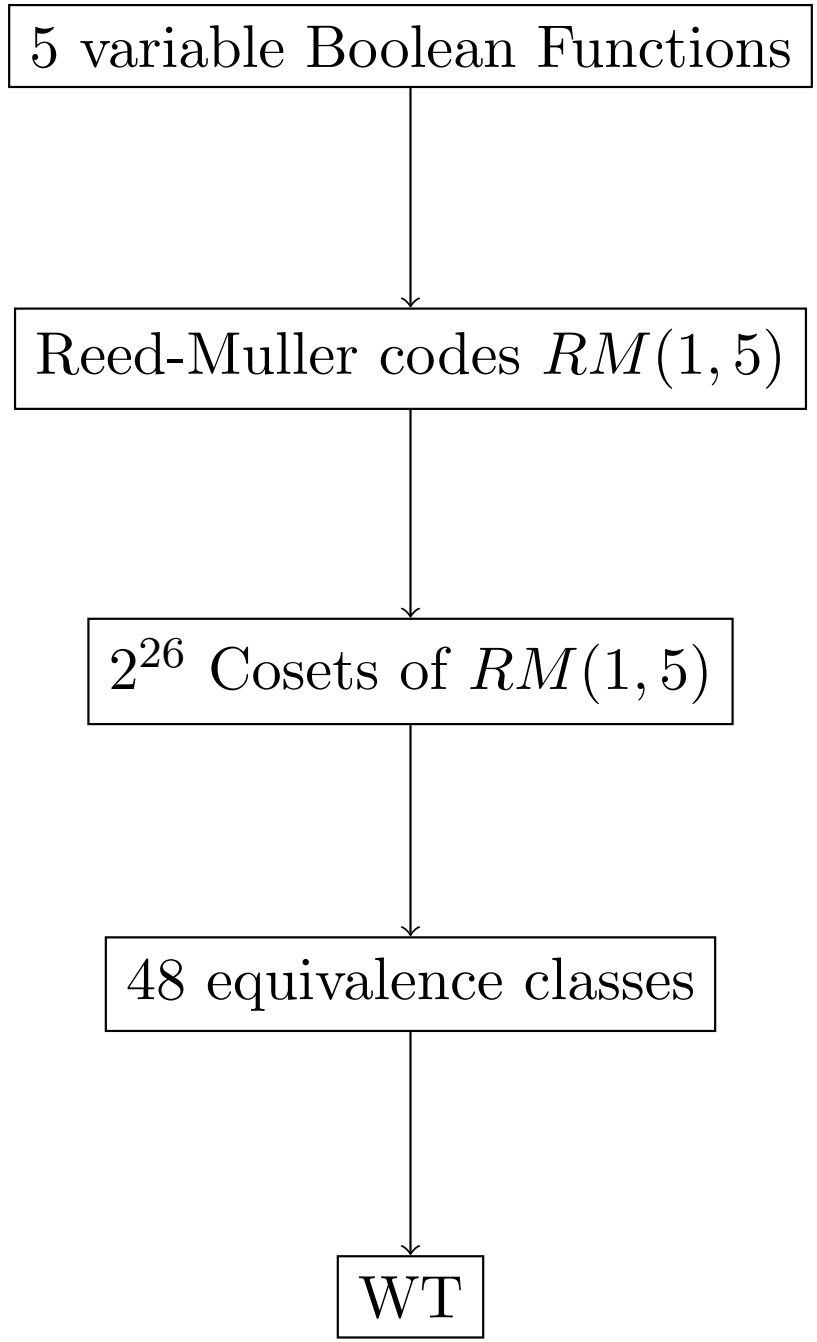
- $f : \mathbb{F}_2^{2m+1} \to \mathbb{F}_2$, local function of a CA
- $\forall t \in \mathbb{N}$, f^t denotes f's iterate

Results:

- f_R^t is 1-resilient iff f^t is 1-resilient.
- f_N^t is 1-resilient iff f^t is 1-resilient.

where:

- f_N negation of the truth table
- f_R reflection of the truth table (mirror image)


5 variable Boolean Functions
Reed-Muller codes $RM(1,5)$
2^{26} Cosets of $RM(1,5)$
48 equivalence classes
WT

1-Resilient, Radius 2-CA Rules

From [Braeken et al., 2008], we know the representatives of BF which are 1-resilient (skipping linear):

Representative	$\mathcal{N}_{CI(1)}$	$\mathcal{N}_{R(1)}$
12	4 840	4 120
123	16 640	11 520
123+14	216 000	133 984
123+14+25	69 120	24 960
123+145+23	1 029 120	537 600
123+145+23+24+35	233 472	96 960

Table: Number of functions satisfying $CI(1)$ and $R(1)$.

Problem: How can we find the BF in the equivalence class?

$R(1)$, Radius 2-CA Rules

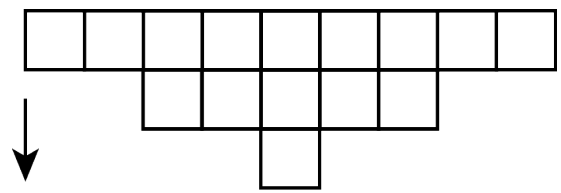
- Representative $R(x_1, x_2, x_3, x_4, x_5)$= coset leader.
- Consider elements of the form $R(x_1, x_2, x_3, x_4, x_5) \oplus (ax_1) \oplus (bx_2) \oplus (cx_3) \oplus (dx_4) \oplus (ex_5) \oplus h$ for a, b, c, d, e, h Boolean, spanning the 2^6 elements of the coset.
- Compute the WT on all elements of the coset
- Select balanced BF
- Select among the balanced BF those with $CI(1)$

$R(1)$-BF on 5 Variables

Coset	1-resilient functions
12	3c3c3cc3 3c3cc33c 3cc33c3c 3cc3c3c3 5a5a5aa5 5a5aa55a 5aa55a5a 5aa5a5a5 66666699 66669966 66996666 66999999 69696996 69699669 69966969 69969696 96696969 96699696 96966996 96969669 99666666 99669999 99996699 99999966 a55a5a5a a55aa5a5 a5a55aa5 a5a5a55a c33c3c3c c33cc3c3 c3c33cc3 c3c3c33c
123	66696996 66699669 66966969 66969696 69666699 69669966 69996666 69999999 96666666 96669999 96996699 96999966 99696969 99699696 99966996 99969669
123+14	66695aa5 6669a55a 66965a5a 6696a5a5 696655aa 6966aa55 969955aa 9699aa55 99695a5a 9969a5a5 99965aa5 9996a55a
123+14+25	$\varnothing$
123+145+23	1eb4663c 1eb499c3 e14b663c e14b99c3
123+145+23+24+35	$\varnothing$

Testing the Iterates

Extension to BF on 9 variables (2 iterations of the local f).



Select from previous rules, those preserving $R(1)$.

(iteration does not preserve resiliency)

$R(1)$- Rules After 2 Iterations

Coset 12	0x3C3C3CC3	yes	0x3C3CC33C	no	0x3CC33C3C	no
	0x3CC3C3C3	yes	0x5A5A5AA5	yes	0x5A5AA55A	yes
	0x5AA55A5A	yes	0x5AA5A5A5	yes	0x66666699	yes
	0x66669966	yes	0x66996666	yes	0x66999999	yes
	0x69696996	yes	0x69699669	yes	0x69966969	yes
	0x69969696	yes	0x96696969	yes	0x96699696	yes
	0x96966996	yes	0x96969669	yes	0x99666666	yes
	0x99669999	yes	0x99996699	yes	0x99999966	yes
	0xA55A5A5A	yes	0xA55AA5A5	yes	0xA5A55AA5	yes
	0xA5A5A55A	yes	0xC33C3C3C	yes	0xC33CC3C3	no
	0xC3C33CC3	no	0xC3C3C33C	yes		
Coset 123	0x66696996	yes	0x66699669	yes	0x66966969	yes
	0x66969696	yes	0x69666699	yes	0x69669966	yes
	0x69996666	yes	0x69999999	yes	0x96666666	yes
	0x96669999	yes	0x96996699	yes	0x96999966	yes
	0x99696969	yes	0x99699696	yes	0x99966996	yes
	0x99969669	yes				
Coset 123+14	0x66695AA5	yes	0x6669A55A	yes	0x66965A5A	yes
	0x6696A5A5	yes	0x696655AA	yes	0x6966AA55	yes
	0x969955AA	yes	0x9699AA55	yes	0x99695A5A	yes
	0x9969A5A5	yes	0x99965AA5	yes	0x9996A55A	yes
Coset 123+145+23	0x1EB4663C	no	0x1EB499C3	no	0x2D7855F0	no
	0x2D78AA0F	no	0x44EE3C66	no	0x44EEC399	no
	0x4B1ECC69	no	0x77220FAA	no	0x7722F055	no
	0x88DD0FAA	no	0x88DDF055	no	0xB4E13396	no
	0xBB113C66	no	0xBB11C399	no	0xD28755F0	no
	0xD287AA0F	no	0xE14B663C	no	0xE14B99C3	no

PRNG Testing

Two tests :

1. Randomness preservation:
   Is the randomness quality of a PRS preserved through CA iteration?
2. Random Number Generation [Shackleford et al., 2002]:
   Is the CA able to generate a good PRS?

Evaluations made with the Diehard test suite.

Diehard

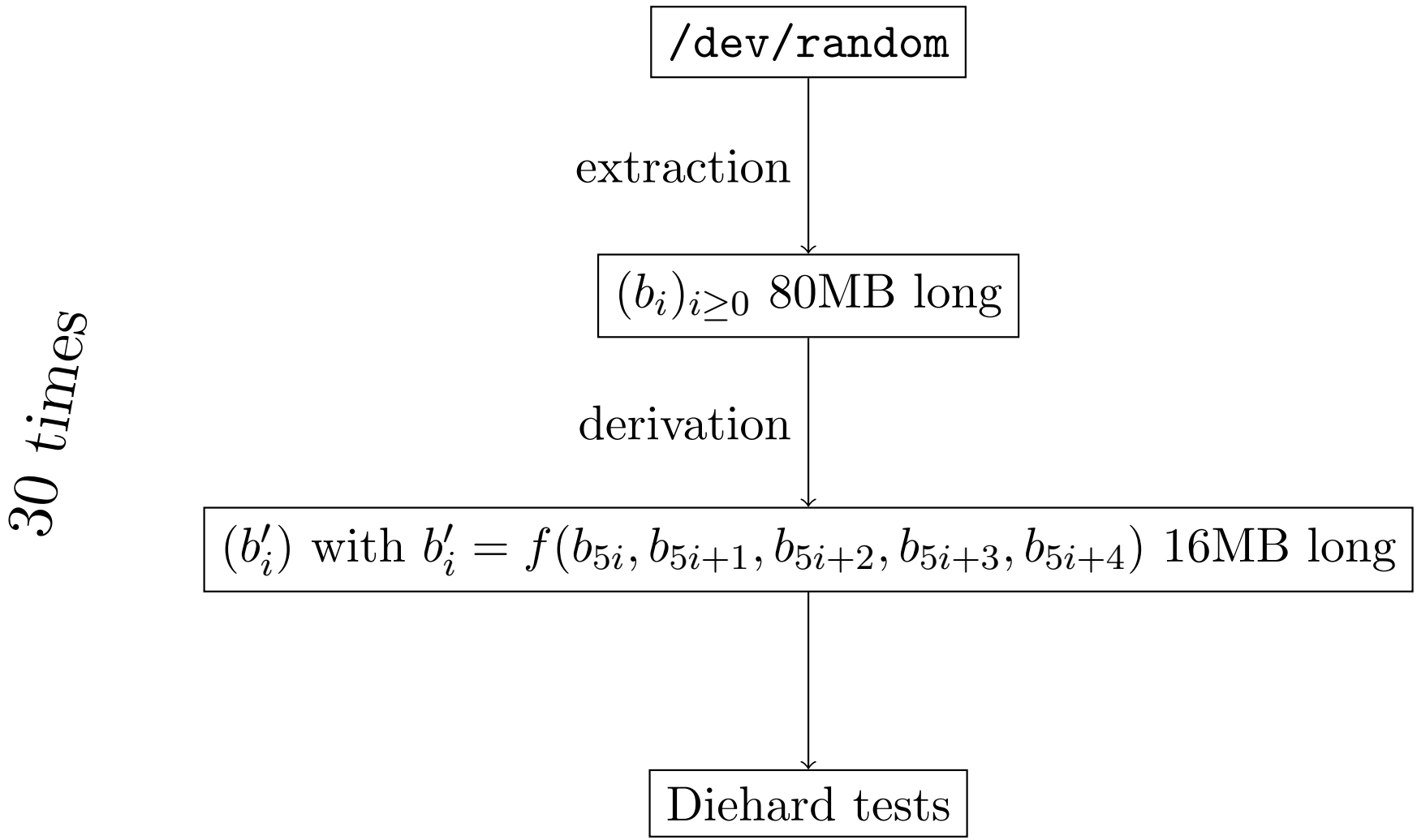
by G. Marsaglia - Florida State University
recommended by CSRC/CSD of NIST
U.S. National Institute of Standards and Technology

Many different tests to *measure* the quality of the randomness

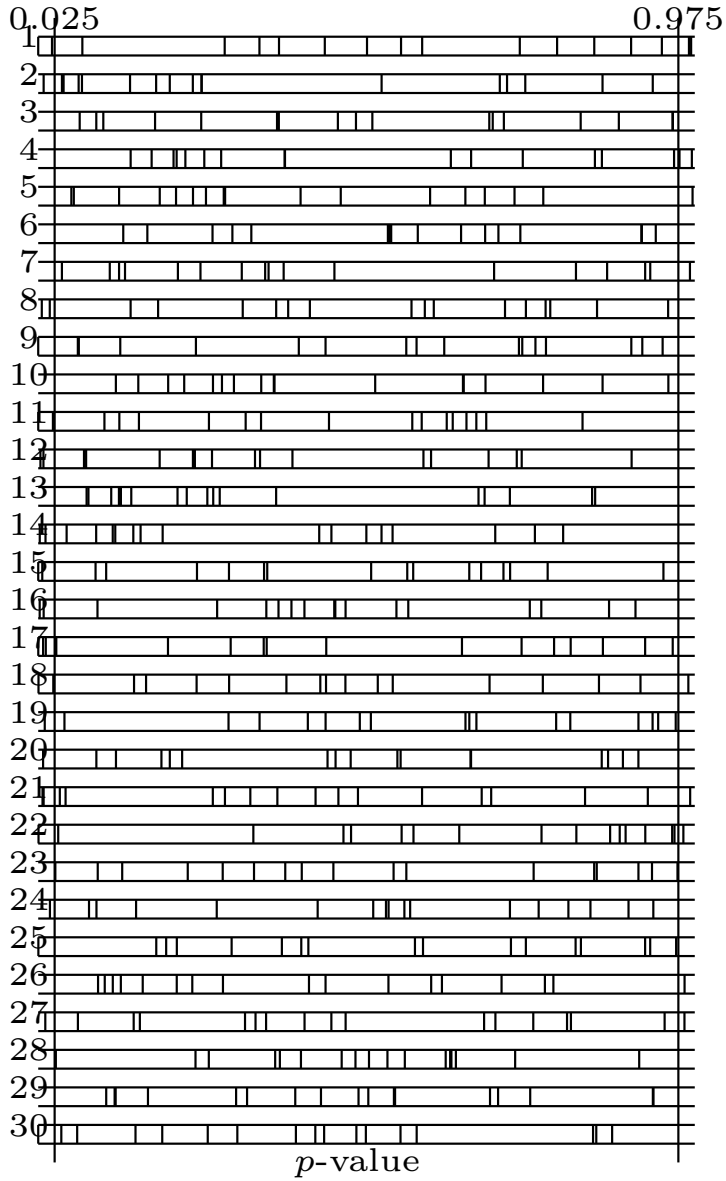
Based on Kolmogorov-Smirnov normality test

Provide indicators which should be uniformly distributed on $[0, 1]$ if the input sequence is made of truly independents bits.

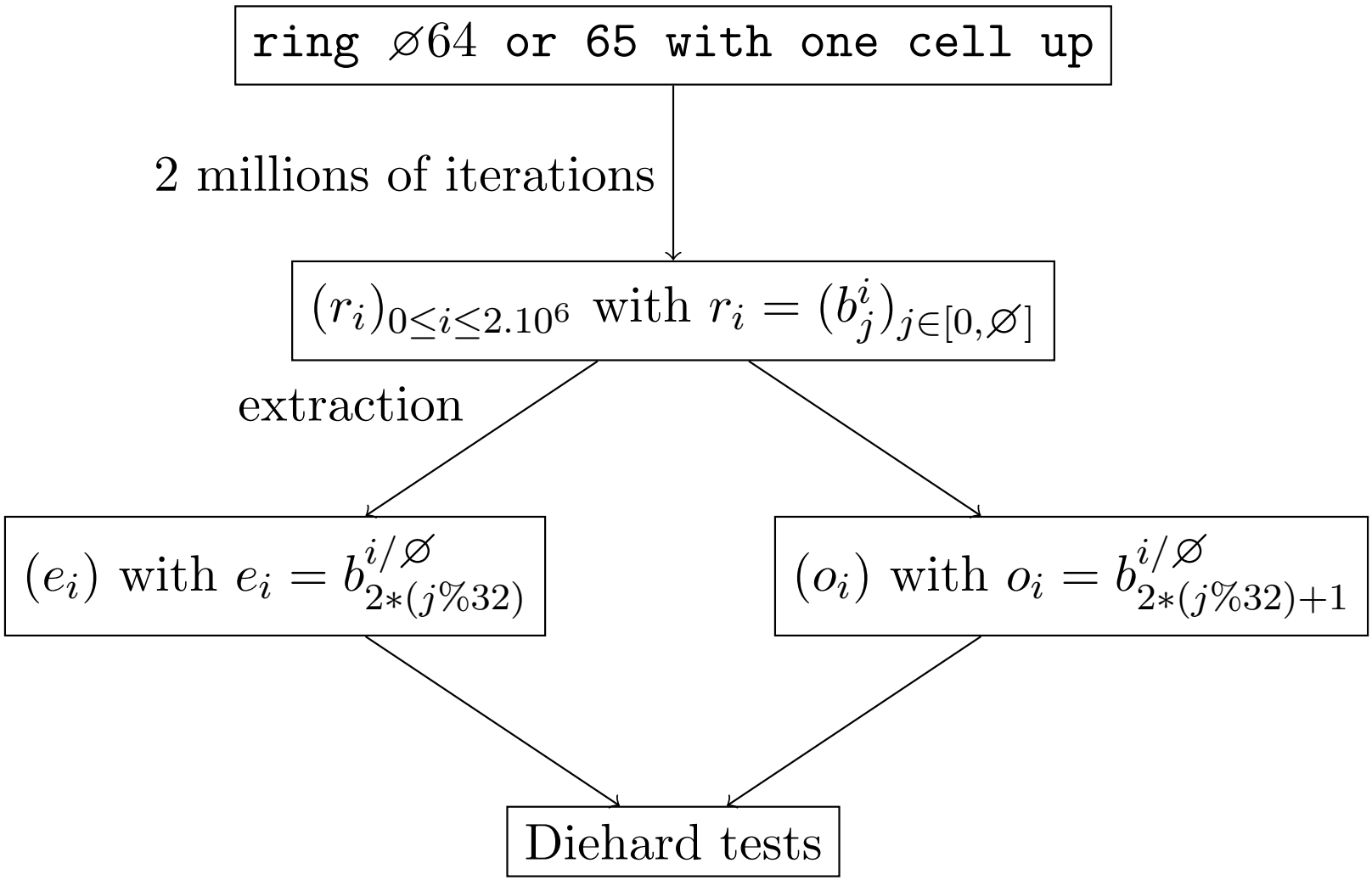
Randomness preservation



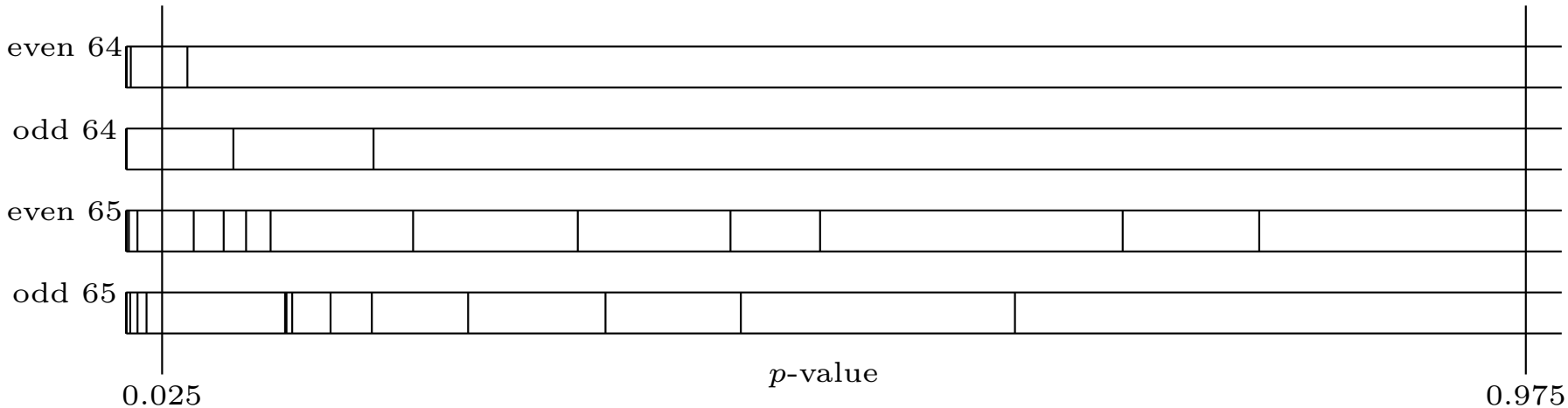
Randomness preservation - results



Random Number Generation

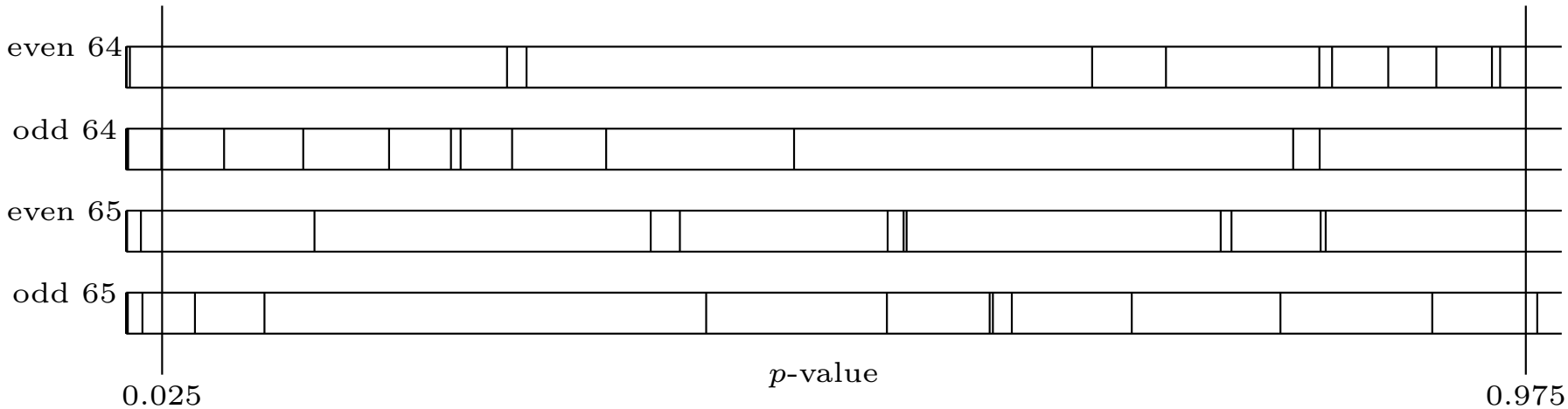


Random Number Generation



p-values distribution for the ring CA with rule `0x3C3C3CC3`. p-values between the two lines (at 0.025 and 0.975) mean that the corresponding statistical test was successful, which is not the case for even 64 and odd 64 (all the p-values are almost zero) and barely for even 65 and odd 65.

Random Number Generation



Distribution of the p-values for the ring CA with rule `0x69999999`. p-values between the two lines (at 0.025 and 0.975) mean that the corresponding statistical test was successful.

Further work - Conclusion

- Approach for getting good BF for PRG
- Complete the search of radius-2 rules
- Classify all CA rules up to radius 2
- Provide a tool to find good BF in many variables

- Thank you


Braeken, A., Borissov, Y., Nikova, S., and Preneel, B. (2008).
Classification of boolean functions of 6 variables or less with respect to cryptographic properties.
Technical report, IACR248.


Cattell, K. and Muzio, J. (1998).
An explicit similarity transform between CA and LFSR matrices.
Finite fields and their applications, 4:239–251.


Formenti, E., Imai, K., Martin, B., and Yunès, J. (2014).
Advances on random sequence generation by uniform cellular automata.
In Calude, C. S., Freivalds, R., and Iwama, K., editors, *Computing with New Resources - Essays Dedicated to Jozef Gruska on the Occasion of His 80th Birthday*, volume 8808 of *Lecture Notes in Computer Science*, pages 56–70. Springer.


Lacharme, P., Martin, B., and Solé, P. (2008).
Pseudo-random sequences, boolean functions and cellular automata.
In *Proceedings of Boolean Functions and Cryptographic Applications*.


Martin, B. (2008).
A Walsh exploration of elementary CA rules.
Journal of Cellular Automata, 3(2):145–156.


Seredynski, F., Bouvry, P., and Zomaya, A. Y. (2004).
Cellular automata computations and secret key cryptography.
Parallel Comput., 30(5-6):753–766.


Shackleford, B., Tanaka, M., Carter, R. J., and Snider, G. (2002).
FPGA implementation of neighborhood-of-four cellular automata random number generators.
In *Proceedings of the 2002 ACM/SIGDA tenth international symposium on Field-programmable gate arrays*, FPGA'02, pages 106–112. ACM.


Sipper, M. and Tomassini, M. (1996).
Co-evolving parallel random number generators.

In *Parallel Problem Solving from Nature – PPSN IV*, pages 950–959, Berlin. Springer Verlag.

Xiao, G.-Z. and Massey, J. L. (1988).

A spectral characterization of correlation-immune combining functions.
IEEE Trans. on Information Theory, 34(3):569–.

Yuen, C.-K. (1977).

Testing random number generators by Walsh transform.
IEEE Trans. Computers, 26(4):329–333.