Traversing a *n*-cube without Balanced Hamiltonian Cycle to Generate Pseudorandom Numbers

J.-F. Couchot, P.-C. Heam, C. Guyeux, Q. Wang, and J. M. Bahi

FEMTO-ST Institute, University of Franche-Comté, France
College of Automation, Guangdong University of Technology, China

2014/09/25

Pseudo Random Number Generation

- Fields of Applications:
  - Security: hash function, steganography, cryptography
  - Time Synchronization: GPS
  - Numerical simulations: Monte-Carlo algorithms

- Some requirements:
  - For cryptography: cryptographically secure
  - Successful pass on PRNG batteries of tests: NIST[1], DieHARD[2]

[1] E. Barker and A. Roginsky. Draft NIST special publication 800-131 recommendation for the transitioning of cryptographic algorithms and key sizes, 2010.

[2] G. Marsaglia. DieHARD: a battery of tests of randomness. *http://stat.fsu.edu/ geo/diehard.html*, 1996

Pseudo Random Number Generation

- Fields of Applications:
  - Security: hash function, steganography, cryptography
  - Time Synchronization: GPS
  - Numerical simulations: Monte-Carlo algorithms
  - Simulation of Chaotic systems: protein dynamics e.g.
- Some requirements:
  - For cryptography: cryptographically secure
  - Successful pass on PRNG batteries of tests: NIST[1], DieHARD[2]
  - Should have chaotic properties

[1] E. Barker and A. Roginsky. Draft NIST special publication 800-131 recommendation for the transitioning of cryptographic algorithms and key sizes, 2010.

[2] G. Marsaglia. DieHARD: a battery of tests of randomness. *http://stat.fsu.edu/ geo/diehard.html*, 1996

Chaotic PRNG

Motivation

Automatically generating a large class of PRNGs with chaos and statistical properties

Previous work

To provide a PRNG with the properties of Devaney's chaos and of succeeding NIST test: a (non-chaotic) PRNG + iterating a Boolean maps[a]:

- with strongly connected iteration graph
- with doubly stochastic Markov probability matrix

---

[a] J. Bahi, J.-F. Couchot, C. Guyeux, and A. Richard. On the link between strongly connected iteration graphs and chaotic Boolean discrete-time dynamical systems, *Fundamentals of Computation Theory*, volume 6914 of *Lecture Notes in Computer Science*, pages 126–137. Springer Berlin Heidelberg, 2011.

Problematic

A (coarse) two steps approach

1. Sufficient conditions to retrieve Boolean maps whose graphs are strongly connected are given
2. Further filter those whose Markov matrix is doubly stochastic

Drawback

Delaying the second requirement to a final step whereas this is a necessary condition

Content of this work

A completely new approach to generate Boolean functions, whose Markov matrix is doubly stochastic and whose graph of iterations is strongly connected (denoted as DSSC Matrix)

Outline

Outline

Boolean Map

- Boolean algebra on $\mathbb{B} = \{0, 1\}$ with the classical operators: ., $+$, $\bar{\ }$, disjunctive union $\oplus$
- For $n \in \mathbb{N}^*$, a *Boolean map* f: a function

$$\mathbb{B} \to \mathbb{B}, x = (x_1, \ldots, x_n) \mapsto f(x) = (f_1(x), \ldots, f_n(x))$$

- Dynamics:
  - $s = (s_t)_{t \in \mathbb{N}}$: sequence of indices in $[\![1; n]\!]$ called “strategy”.
  - At the t^{th} iteration: only the s_t-th component is “iterated”

$$\begin{array}{rcl} x^{t+1} & = & F_f(s_t, x^t) \\ & \text{where} & \\ F_f & : & [\![1; n]\!] \times \mathbb{B}^n \to \mathbb{B}^n \\ F_f(i, x) & = & (x_1, \ldots, x_{i-1}, f_i(x), x_{i+1}, \ldots, x_n) \end{array}$$

Iteration Graph and Markov Matrix

Iteration Graph

The *iteration graph* $\Gamma(f)$: directed graph s. t.

- the set of vertices: $\mathbb{B}^n$
- the set of edges: $(x, F_f(i,x)) \in \Gamma(f)$, $x \in \mathbb{B}^n$, $i \in [\![1;n]\!]$

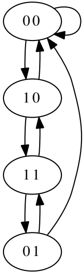
Markov Matrix

Matrix M:

$$M_{ij} = \frac{1}{n} \text{ if } i \neq j \text{ and } (i,j) \in \Gamma(f)$$
$$M_{ij} = 0 \text{ if } i \neq j \text{ and } (i,j) \notin \Gamma(f)$$
$$M_{ii} = 1 - \sum_{j=1, j\neq i}^{n} M_{ij}$$

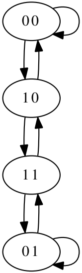
Iteration Graph and Markov Matrix (cont'd)

$g(x_1, x_2) = (\overline{x_1}, x_1\overline{x_2})$, $h(x_1, x_2) = (\overline{x_1}, x_1\overline{x_2} + \overline{x_1}x_2)$



$$\frac{1}{2}\begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

(a) $\Gamma(g)$, M_g



$$\frac{1}{2}\begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

(b) $\Gamma(h)$, M_h

Our PRNG

Mixing Time

The smallest iteration number that is sufficient to obtain a deviation lesser ε between rows of M and a given distribution.

PRNG $\chi_{\textit{14Secrypt}}$

Input: a function f, an iteration number b, a *Random* PRNG, an initial configuration x^0 (n bits)
Output: a configuration x (n bits)
$x \leftarrow x^0$;
for $i = 0, \ldots, b-1$ **do**
  $s \leftarrow Random(n)$;
  $x \leftarrow F_f(s, x)$;
end
return x;

- From x^0: a random walk in $\Gamma(f)$ thanks to *Random* of length b

Outline

1. Introduction

2. Preliminaries

3. Generation of DSSC Matrices

4. On Removing Hamiltonian Cycles

5. Experiments

6. Conclusion

A typical CLPFD

From Theory

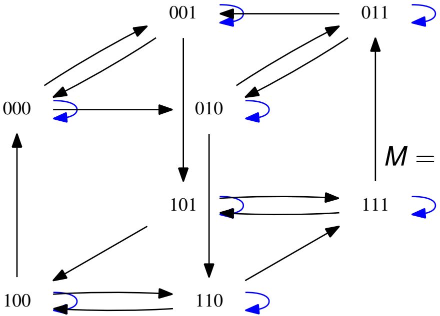
Find all the $2^n \times 2^n$ matrices $M = \frac{1}{n}.\hat{M}$ such that:

1. $\hat{M}_{ij} = 0$ if j is not a neighbor of i
2. $0 \leq \hat{M}_{ii} \leq n$: the number of loops around i is lesser than n
3. Otherwise $\hat{M}_{ij} = 1$ if the edge from i to j is kept and 0 otherwise
4. For any index of line i, $1 \leq i \leq 2^n$, $n = \sum_{1 \leq j \leq 2^n} \hat{M}_{ij}$: the matrix is right stochastic
5. For any index of column j, $1 \leq j \leq 2^n$, $n = \sum_{1 \leq i \leq 2^n} \hat{M}_{ij}$: the matrix is left stochastic
6. All the values of $\sum_{1 \leq k \leq 2^n} \hat{M}^k$ are strictly positive: the induced graph is strongly connected

A typical CLPFD (cont'd)

To Practice

- Definitively not efficient enough: a *generate and test* approach
- $f^*(x_1, x_2, x_3) = (x_2 \oplus x_3, \overline{x_1}\,\overline{x_3} + x_1\overline{x_2}, \overline{x_1}\,\overline{x_3} + x_1 x_2)$: function with the smallest MT, $n = 3$

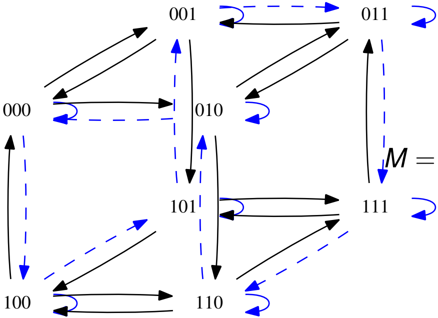


$$M = \frac{1}{3}\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

A typical CLPFD (cont'd)

To Practice

- Definitively not efficient enough: a *generate and test* approach
- $f^*(x_1, x_2, x_3) = (x_2 \oplus x_3, \overline{x_1 x_3} + x_1\overline{x_2}, \overline{x_1 x_3} + x_1 x_2)$: function with the smallest MT, $n = 3$
- f^*: the 3-cube in which the *Hamiltonian cycle* $000, 100, 101, 001, 011, 111, 110, 010, 000$ has been removed



$$M = \frac{1}{3}\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

Outline

Theoretical Aspects

Theorem

The Markov Matrix M resulting from the n-cube in which an Hamiltonian cycle is removed, is doubly stochastic

Theorem

The iteration graph issued from the n-cube where an Hamiltonian cycle is removed is strongly connected

We are then left

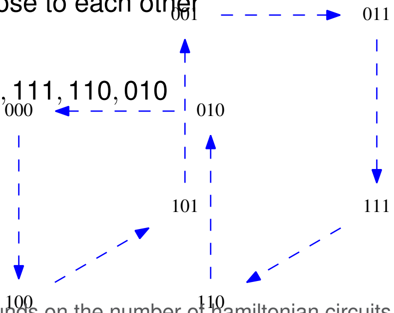
- To focus on the generation of Hamiltonian cycles in the n-cube, *i.e.*,
- To find cyclic Gray codes: sequences of 2^n codewords (n-bits strings) where two successive elements differ in only one bit position and and where the last codeword differs in only one bit position from the first one

Cyclic Balanced Gray Codes

- Lower bound[3] of number of Gray codes in $\mathbb{B}^n$:
  $\left(\frac{n*\log 2}{e \log\log n} * (1 - o(1))\right)^{2^n}$ (more than 10^{13} when n is 6).
- Restriction to balanced codes: the number of edges that modify the bit i in $\Gamma(f)$ have to be close to each other

Study of previous code

- $L^* = 000, 100, 101, 001, 011, 111, 110, 010$
- Its transition sequence:
  $S = 3, 1, 3, 2, 3, 1, 3, 2$



[3] T. Feder and C. Subi. Nearly tight bounds on the number of hamiltonian circuits of the hypercube and generalizations.
Inf. Process. Lett., 109(5):267–272, February 2009.

Generation of Balanced Gray Codes

- Algorithm [4]: inductive construction of n-bits Gray code given a $n-2$-bit Gray code

- *Let l be an even positive integer. Find $u_1, u_2, \ldots, u_{l-2}, v$ (maybe empty) subsequences of S_{n-2} such that S_{n-2} is the concatenation of $s_{i_1}, u_0, s_{i_2}, u_1, s_{i_3}, u_2, \ldots, s_{i_l-1}, u_{l-2}, s_{i_l}, v$ where $i_1 = 1$, $i_2 = 2$, and $u_0 = \emptyset$ (the empty sequence).*

- $\rightsquigarrow \#_n = \sum_{l'=1}^{2^{n-3}} \binom{2^{n-2}-2}{2l'-2}$ distinct u subsequences

n	4	5	6	7	8
$\#_n$	1	31	8191	5.3e8	2.3e18
$\#'_n$	1	15	3003	1.4e8	4.5e17

- A first simplification $\rightsquigarrow \#'_n$

[4] A. J. van Zanten and I. N. Suparta. Totally balanced and exponentially balanced gray codes. *Discrete Analysis and Operational Research*, 11:81–98, 2004.

Outline

1. Introduction
2. Preliminaries
3. Generation of DSSC Matrices
4. On Removing Hamiltonian Cycles
5. Experiments
6. Conclusion

Experiments

For each $n = 4, 5, 6, 7, 8$

- Generation of Balanced Gray Codes $\rightsquigarrow$ functions f to iterate
- Selection of the function $f*$ minimizing the mixing time b
- Reproduced in the paper
- Evaluation through NIST and DieHARD
- $\rightsquigarrow$ all the generators pass the NIST and the DieHARD batteries of tests

Outline

Conclusion & Future Work

Summary

- Goal: description of a method to compute a large class of truly chaotic PRNGs
- The chaotic iterated map inside the generator: built by removing from a n-cube an Hamiltonian path, *i.e.*, a balanced Gray code
- Statistical properties: established for $n = 4, 5, 6, 7, 8$ through NIST and DieHARD batteries

Open Problems

- Our proposal: remove from the n-cube an Hamiltonian path that is a balanced Gray code. Can we prove that this solution is the one that minimizes the mixing time?
- Lack of constructive method to build balanced Gray Code with large n. Can we propose a new algorithm?

Thanks

:-)