# Using Irreducible Polynomials For Random Number Generation

Tamás Herendi

Sándor Roland Major

DEBRECENI EGYETEM

# Contents



DEBRECENI EGYETEM

# Introduction

- Pseudorandom number generation (PRNG) is an important component of many practical applications:
  - Simulations
  - Monte Carlo methods
  - Key generation
  - Stream cyphers
  - Asymmetric cryptosystems
  - Authentication protocols

DEBRECENI EGYETEM

# Introduction

- The goal is the construction of uniformly distributed linear recurrence sequences (LRS) modulo powers of 2, with theoretically arbitrarily large period lengths

DEBRECENI EGYETEM

# Algorithm for creating LRS

- Requires an irreducible polynomial over $\mathbb{F}_2$
- The degree of this polynomial is directly related to the resulting period length
- Modified version of a previous algorithm by Tamás Herendi, optimized to be less computationally expensive

DEBRECENI EGYETEM

# Algorithm for creating LRS

- Choose an integer $k$, and find an irreducible polynomial $q(x) \in \mathbb{F}_2[x]$
- Calculate

$$p(x) \equiv (x^2 - 1)q(x) \bmod 2 \quad \text{and}$$
$$p'(x) \equiv (x - 1)q(x) \bmod 2 \ ,$$

- The four candidate polynomials:

$$p_1(x) = p(x)$$
$$p_2(x) = p(x) - 2$$
$$p_3(x) = p(x) - 2x$$
$$p_4(x) = p(x) - 2x - 2$$

DEBRECENI EGYETEM

# Algorithm for creating LRS

- For $i \in \{1,2,3,4\}$, $j \in$ {0,1,…,k+2}, let $a_{ij}$ denote the coefficient of $x^j$ in the polynomial $p_i(x)$
- Calculate $S_i = \sum_{j=0}^{k+1} -a_{ij}$ for each candidate
- Keep the two candidates that satisfy $S_i \equiv 1 \mod 4.$
- Denote them as $c_1$ and $c_2$

DEBRECENI EGYETEM

# Algorithm for creating LRS

- Let $\varrho = ord(q)$ be the order of $q(x)$, i.e. the smallest positive integer such that $q(x) | x^{\varrho} - 1$
- We need the candidate which satisfies $c_i(x) \nmid x^{2\varrho} - 1$
- To find it, calculate $r(x) \equiv x^{\varrho} \mod (2, p(x))$

- Then, find the candidate which satisfies $1 \not\equiv r(x)^2 \mod (4, c_i(x))$

DEBRECENI EGYETEM

# Algorithm for creating LRS

- Denote the remaining candidate with $c(x)$, this will be the characteristic polynomial of the LRS we want to create
- Let $b_j,\ j \in \{0, 1, \dots k+2\}$ be the coefficients of $x^j$ in $c(x)$
- Then our final recurrence relation is:

$$u_{n+k+2} = -b_{k+1}u_{n+k+1} - b_k u_{n+k} \dots - b_0 u_n$$

- Note: choosing suitable initial values for the sequence is not trivial either

DEBRECENI EGYETEM

# Q-Transform

- A transformation that under certain conditions can create an infinite series of irreducible polynomials
- Let $q$ be a prime power, $\mathbb{F}$ be a field, and $\mathbb{F}_q$ be a finite field with $q$ elements
- Let $\mathrm{p} \in \mathbb{F}[x]$ be a polynomial of degree $d$. The Q-transform of $\mathrm{p}$ is:

$$\tilde{p}(x) = x^d p(x + x^{-1})$$

DEBRECENI EGYETEM

# Q-Transform

- Let the reciprocal of p be $p^*(x) = x^d p(x^{-1})$. If $p = p^*$, then $p$ is called self-reciprocal
- If $\mathrm{p} \in \mathbb{F}[x]$, then $\tilde{p}$ is self-reciprocal, and $\deg(\tilde{p}) = 2\mathrm{d}$
- Let $\mathrm{q} \in \mathbb{F}_2[x]$ be an irreducible polynomial in the form
$$q(x) = x^d + a_{d-1}x^{d-1} + \cdots + a_1 x + 1$$
- Then $\tilde{q}$ is irreducible if and only if $a_1 = 1$. Furthermore, the coefficient of the linear term of $\tilde{q}$ is 1.

DEBRECENI EGYETEM

# Q-Transform

- Let $\tilde{p}^{(k)}$ denote applying the Q-transform $k$ times to the polynomial $p$
- Let q ∈ $\mathbb{F}_2[x]$ be an irreducible polynomial in the form
$$q(x) = x^d + a_{d-1}x^{d-1} + \cdots + x + 1$$
- Then $\tilde{q}^{(k)}$ is irreducible for all $k \in \mathbb{N}$
- $ord(\tilde{q})|2^d + 1$ (in practise, it is often equal to this limit)
- Conjecture: if $d$ is even, then $2^{d/2} + 1 < \mathrm{ord}(\tilde{q})$

DEBRECENI EGYETEM

# Statistical testing

- Two irreducible polynomials of large degree were created, one using a brute force method and one using Q-transformations.
- The pseudorandom sequences generated using these polynomials were tested using the NIST statistical test suite.

DEBRECENI EGYETEM

# Statistical testing

- NIST test suite: 15 tests to examine the properties of pseudorandom bit sequences
  - Frequency test
  - Runs test
  - DFT (Spectral) test
  - Template matching test
  - Maurer's ”Universal Statistical” test
  - Linear complexity test
  - Etc.

DEBRECENI EGYETEM

# Statistical testing

- The two polynomials tested are denoted $t_1$ and $t_2$
- $t_1$ generated using irreducibility testing methods, implemented using NTL (Number Theory Library)
- $\deg(t_1) = 216091$, because $2^{216091} - 1$ is a Mersenne prime
- This simplifies Step 4 of the previously shown algorithm

DEBRECENI EGYETEM

# Statistical testing

- $t_2$ generated using iterated Q-transformation
- Let $q$ be a self-reciprocal irreducible monic polynomial, with $\deg(q) = d$
- Run the previous algorithm, using $q$ as input. Let $p$ be the candidate polynomial that remains after Step 4. Determine $s, r \in \mathbb{Z}[x]$ such that $p = sq + r$, and $\deg(r) < \deg(q)$
- Compute $t = s\tilde{q}^{(n)} + r$

DEBRECENI EGYETEM

# Statistical testing

- The self-reciprocal irreducible monic polynomial used:

$$q_2 = x^{14} + x^{13} + x^{12} + x^{11} + x^{10} + x^9 + x^7 + x^5 + x^4 + x^3 + x^2 + x + 1$$

- Using the previous method, $p_2 = s_2 q_2 + r_2$ was determined and $t_2$ was set as $t_2 = s_2 \tilde{q_2}^{(14)} + r_2$.
- $\deg(t_2) = 229378$

DEBRECENI EGYETEM

# Statistical testing

- Using $t_1$ and $t_2$, two LRSs were created to generate the pseudorandom sequences to be tested, denoted L1 and L2 respectively.
- Both LRSs generate 64-bit words.
- Following the recommendations in the documentation of the NIST test suite, 16MB ($2^{21}$ words) of test data were generated using L1 and L2 each.

DEBRECENI EGYETEM

# Statistical testing

- For each of these two streams, the NIST suite split the data into 100 bitstreams. The testing software provides a detailed output of the tests, as well as a summary showing the number of bitstreams that passed each test.
- The minimum pass rate for a test is considered to be 96 out of a sample size of 100
- The full report can be found at https://arato.inf.unideb.hu/major.sandor/statistical_results/

DEBRECENI EGYETEM

# Statistical testing

TABLE I
NIST TEST RESULTS OF $L_1$ GENERATOR

| Statistical Test | P-value | Proportion |
|---|---|---|
| Frequency | 0.779188 | 100/100 |
| Runs | 0.514124 | 100/100 |
| FFT | 0.924076 | 99/100 |
| OverlappingTemplate | 0.012650 | 96/100 |
| Universal | 0.935716 | 97/100 |
| LinearComplexity | 0.699313 | 99/100 |

TABLE II
NIST TEST RESULTS OF $L_2$ GENERATOR

| Statistical Test | P-value | Proportion |
|---|---|---|
| Frequency | 0.955835 | 100/100 |
| Runs | 0.108791 | 98/100 |
| FFT | 0.678686 | 98/100 |
| OverlappingTemplate | 0.035174 | 97/100 |
| Universal | 0.249284 | 100/100 |
| LinearComplexity | 0.719747 | 100/100 |

DEBRECENI EGYETEM

# Thank you for your attention!

The research was supported by the SETIT Project (no. 2018-1.2.1-NKP-2018-00004), which has been implemented with the support provided by the National Research, Development and Innovation Fund of Hungary, financed under the 2018-1.2.1-NKP funding scheme.