Deterministically broken Periodicity of Linear Congruential Generators using Quasi-Crystals

Louis-Sébastien Guimond[*] Jiří Patera[†]
Zuzana Masáková[‡] Edita Pelantová [§]

CRM-2624

August 1999

[*]Centre de Recherches Mathématiques, Université de Montréal, c.p. 6128, succ. centre-ville, Montréal, Qué, Canada, H3C-3J7.guimond@CRM.UMontreal.CA

[†]Centre de Recherches Mathématiques, Université de Montréal, c.p. 6128, succ. centre-ville, Montréal, Qué, Canada, H3C-3J7.patera@CRM.UMontreal.CA

[‡]Department of Mathematics, Czech Technical University, Faculty of Nuclear Science and Physical Engineering, Trojanova 13, Prague 2, 120 00, Czech Republic.masakova@km1.fjfi.cvut.cz

[§]Department of Mathematics, Czech Technical University, Faculty of Nuclear Science and Physical Engineering, Trojanova 13, Prague 2, 120 00, Czech Republic.pelantova@km1.fjfi.cvut.cz

Abstract

We describe the design of a family of aperiodic pseudorandom number generator (APRNG). These deterministic generators are based on linear congruential generators (LCGs) and, unlike any other deterministic PRNG, lead to nonperiodic pseudorandom sequences. An APRNG consists of several LCGs whose combination, controlled by a quasicrystal, forms an infinite aperiodic sequence of pseudorandom numbers.

Keyworks : aperiodic pseudo-random number generators, cryptography, design of algorithm, linear congruential generators, pseudo-random number generators, quasi-crystals.

Résumé

Nous décrivons la conception d'une famille de générateurs apériodique de nombres pseudo-aléatoires (GANPA). Ces générateurs déterministes utilisent des générateurs congruents linéaires (GCLs) et, contrairement à tout autre GNPA, engendrent des suites pseudo-aléatoires apériodiques. Un GANPA est formé de GCLs dont la combinaison, déterminée à l'aide d'un quasicristal, forme une suite infinie et apériodique de nombres pseudo-aléatoires.

The aperiodic pseudo-random number generator (APRNG) is a deterministic PRNG based on linear congruential generators (LCGs) which, unlike any deterministic PRNG, leads to non-periodic pseudo-random sequence. The APRNG is of interest, for example, in computer programming and intensive simulations, and it is a first step in constructing a cryptographic system using *quasi-crystals* (QC).

A LCG is a pseudo-random sequence generator of the form

$$x_n = (ax_{n-1} + b) \pmod{m} \qquad \text{for all integers } n > 0\,, \tag{1}$$

with a given "seed" x_0. When the parameters (a, b, m) are chosen properly, the sequence $(x_n)_{n=0}^{+\infty}$ has the maximal period m, and for any $i \in \mathbb{N}$ and sufficiently suitable N, $(x_n)_{n=i}^{N+i}$ shows good statistical behavior with respect to most reasonable empirical tests [4]. LCGs with small prime moduli are very fast and easily implementable, unfortunately they have small period and are cryptographically insecure [3, 5, 6].

The APRNG consists of several LCGs whose combination, controlled by a QC, forms an infinite aperiodic sequence of pseudo-random numbers. The aperiodicity of this sequence follows from the properties of *cut-and-project sequences* (CPSs) [8, 9, 10, 2]. These sequences are mathematical models of quasi-crystals which are constructed by projection of chosen points from a two dimensional lattice on a straight line. These points are chosen from a strip along the straight line as illustrated in figure 1. The position and width of the strip are free parameters. According to the 3-gap theorem [14], each such projection determines a tiling of the real line where distances between neighboring points have at most three different values. It is possible to choose the slope of the straight line in such a way that this geometric definition can be formulated in a simple algebraic way [11].

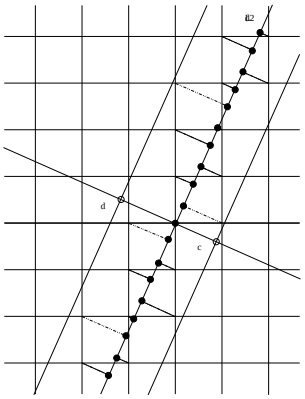

Figure 1: One-dimensional cut-and-project sequence.

As an example, consider the straight line L with slope $\tau = \frac{1}{2}(1 + \sqrt{5})$, the largest solution of $x^2 = x + 1$ (τ is known as the *the golden ratio*). The CPS is defined as

$$\Sigma[c, d) = \left\{ a + b\tau \;\middle|\; a, b \in \mathbb{Z}, a - \frac{b}{\tau} \in [c, d) \right\}\,,$$

where $[c, d)$, $c, d \in \mathbb{R}$, is called the *acceptance interval* or the *acceptance window*. Let us mention several important properties of the sequence $\Sigma[c, d)$.

- $\Sigma[c, d)$ is aperiodic in the strongest sense: it has no periodic subset [11].

- The sequence $\Sigma[c,d)$ determines a tiling of the real line by three different tiles with length of ratio $1:\tau:\tau^2$ [8]. (There exists limit cases with only two different tiles.) We can identify these tiles with letters S (short), M (middle) and L (long). The tiling gives an aperiodic infinite word in the alphabet $\{S,M,L\}$ [2].

- The sequence $\Sigma[c,d)$ can be easily generated from an arbitrary initial point $y_0 \in \Sigma[c,d)$ in both directions.

- Every finite pattern in $\Sigma[c,d)$ is repeated infinitely many times in the sequence. Therefore, $\Sigma[c,d)$ can not be precisely computed only from a finite fragment: the values of parameters c and d must be known, although a reasonably good estimate for the value $d-c$ can be made from a fragment with sufficiently many points [13].

- The sequence $\Sigma[c,d)$ has the *scaling property*: $-\tau\Sigma[c,d) = \Sigma[c/\tau, d/\tau)$ [1]. Thus it suffices to consider acceptance windows with length $1 \le d-c < \tau$. (The value $d-c=1$ corresponds to the two-tile limit case.) Such values of $d-c$ imply that tiles in $\Sigma[c,d)$ have length $S=1$, $M=\tau$ and $L=\tau^2$. (In the limit case $d-c=1$, the tiles have length $M=\tau$ and $L=\tau^2$.)

- If $1 \le d_1 - c_1 < d_2 - c_2 < \tau$, then sequences $\Sigma[c_1,d_1)$ and $\Sigma[c_2,d_2)$ are essentially different: corresponding tilings have different densities of tiles. For example, as $d-c$ decreases to 1 the density of S-tiles decreases to 0, while as $d-c$ increases to τ the density of L-tiles decreases to 0.

The sequence $\Sigma[c,d)$ is aperiodic but can not be used directly as a PRNG. Indeed, the letters forming this sequence have different densities and lead to bad statistics. The APRNG uses the listed properties of CPSs to break the periodicity of LCGs while LCGs are used to eliminate the non-uniformness of CPSs.

The design of the APRNG is the following. Let three LCGs given by parameters a_i, b_i, m_i, $i=1,2,3$, and three seeds $x_0^{(i)}$. Select an interval $[c,d)$ and a point $y_0 \in \Sigma[c,d)$ with $1 < d-c < \tau$. (We illustrate the APRNG in the generic three-tile case.)

We simultaneously use four sequences: three linear congruential sequences $\left(x_n^{(i)}\right)_{n=0}^{+\infty}$, for $i=1,2,3$, and one aperiodic sequence $(y_n)_{n=0}^{+\infty}$, where $y_n \in \Sigma[c,d)$ is defined recursively by the following prescription: y_{n+1} is the closest right neighbor of y_n in $\Sigma[c,d)$. From these sequences we create a pseudo-random sequence $(z_k)_{k=0}^{+\infty}$ in the following way.

Assume that in the step k we have generated the point y_k from $\Sigma[c,d)$ and the elements $x_{s_k}^{(1)}$, $x_{m_k}^{(2)}$ and $x_{l_k}^{(3)}$, where $s_k + m_k + l_k = k$.

Step $k+1$:

1. Generate point y_{k+1} from $\Sigma[c,d)$.

2. Generate element z_{k+1} (and thus elements $x_{s_{k+1}}^{(1)}$, $x_{m_{k+1}}^{(2)}$ and $x_{l_{k+1}}^{(3)}$ where $s_{k+1}+m_{k+1}+l_{k+1} = k+1$):

   If $y_{k+1} - y_k = S$, then set

   $$s_{k+1} := s_k + 1,\ m_{k+1} := m_k,\ l_{k+1} := l_k,\ \text{and}\ z_{k+1} := x_{s_{k+1}}^{(1)}.$$

   If $y_{k+1} - y_k = M$, then set

   $$s_{k+1} := s_k,\ m_{k+1} := m_k + 1,\ l_{k+1} := l_k,\ \text{and}\ z_{k+1} := x_{m_{k+1}}^{(2)}.$$

If $y_{k+1} - y_k = L$, then set

$$s_{k+1} := s_k,\ m_{k+1} := m_k,\ l_{k+1} := l_k + 1,\ \text{and} z_{k+1} := x^{(3)}_{l_{k+1}}.$$

The APRNG is now in the process of being implemented and tested (both statistical analysis and cryptanalysis). Implement issues mainly concern CPS generation since efficient LCGs implementations are known. There are two theoretical methods of generating CPSs: numeric and symbolic. Both methods can be efficiently implemented [12].

Numeric implementation of CPSs (and therefore of the pseudo-random sequence $(z_k)_{k=0}^{+\infty}$ generation) can be made very fast. Indeed, the most trivial way of generating $y_{n+1} \in \Sigma[c, d)$ from a point y_n is to verify whether $y_n + S$, $y_n + M$, or $y_n + L$ belongs to $\Sigma[c, d)$. This simply requires comparison of certain numbers with the boundaries value of the acceptance window. Unfortunately, due to computer's finite arithmetics, such simple numeric generations on an ω-bit CPU are bound to generate points of the form $y_{n+P} = y_n$ for some $P \leq \omega$, thus destroying the aperiodicity of the sequence $(y_n)_{n=0}^{+\infty}$.

If we choose the boundaries of the acceptance window $c, d \in \mathbb{Q}[\tau]$, (which in fact means no limitation since $\mathbb{Q}[\tau]$ is dense in $\mathbb{R}$), then the CPSs can be generated symbolically using *substitution rules* [10]. This method of generation is exact on any type of CPU. However, it requires an initialization procedure which builds the substitution rule generating the CPS [12]. This procedure can be made very fast.

The disadvantage of the present implementation of the symbolic method stems in increasing memory requirements which increases as the logarithm of the number of generated points. This affects the competitiveness of the method but implies no limitation on it. The memory requirement increase may be reduced using the combination of both numeric and symbolic ways of generation [7].

One may ask why, in the symbolic method, not use general substitution rules providing aperiodic sequences rather than specifically use CPSs. The advantage of substitutions generating CPSs is that they have simple description by two parameters c and d determining the acceptance window, even though they are generally long prescriptions with large alphabet. Since the rules may be derived from the two parameters c and d using a simple and fast procedure [12], the 2-parameter description ease the usage of CPSs without adding real overhead. Indeed, the length of the substitution rule has minor influence on the speed of generation of the sequence $(y_n)_{n=0}^{+\infty}$. Moreover, in crypto-systems, an N-parameter description with large N would dramatically enlarge the key size.

In general, CPSs may be obtained for any *quadratic unitary Pisot number* (larger solution of $x^2 = mx \pm 1$, $m \in \mathbb{N}$) [9]. Moreover, one may use irrationalities of higher degree. In such cases, the definition of a CPS requires more than one acceptance window.

The structure of the APRNG is open to various modifications. For example, we could combine more than three LCGs. The substitution rule generating the CPS $(y_n)_{n=0}^{+\infty}$ divides naturally the tiles S, M, and L of the sequence into more groups, $S_1, \cdots, S_{k_1}$, $M_1, \cdots, M_{k_2}$, and $L_1, \cdots, L_{k_3}$, according to the alphabet of the substitution [10]. The groups would guide the combination of $k_1 + k_2 + k_3$ LCGs.

This method is readily amenable to parallel processing since QC can be generated from several (many) seed points simultaneously. Moreover, N-dimensional QCs generate N-dimensional CPSs which could be of interest in crypto-systems for N-dimensional digital documents.

In conclusion, the proof of the aperiodicity of the APRNG has been shown in the two-tile limit case [2]. The authors are presently working on the proof of the generic case.

Acknowledgments: The authors would like to thank Claude Crépeau for his stimulating discus-

sions and suggestions. We would also like to thank the Department of Mathematics of the Faculty of Nuclear Science and Physical Engineering at the Czech Technical University in Prague where part of this work was done. This work was partially supported by NSERC of Canada and FCAR of Québec.

References

[1] S. Berman and R. V. Moody, *The algebraic theory of quasicrystals with five-fold symmetry*, J. Phys. A: Math. Gen. (1994), no. 27, 115–130.

[2] L.-S. Guimond and J. Patera, *Proving the deterministic period breaking of linear congruential generators using two-tile quasi-crystals*, Preprint (1999).

[3] D. E. Knuth, *Deciphering a linear congruential encryption*, IEEE Trans. Inform. Theory **31** (1985), no. 1, 49–52.

[4] ———, *The art of computer programming: Seminumerical methods*, third ed., vol. 2, Addison-Wesley, 1998.

[5] H. Krawczyk, *How to predict congruential generators*, Advances in Cryptology–Crypto '89 (G. Brassard, ed.), Lecture Notes in Computer Science, no. 435, Spinger-Verlag, 1989.

[6] ———, *How to predict congruential generators*, J. Algorithms (1992), no. 13, 527–545.

[7] Z. Masáková, J. Patera, and E. Pelantová, *Patent pending.*

[8] ———, *Minimal distances in quasicrystals*, J. Phys. A: Math. Gen. **31** (1998), 1539–1552.

[9] ———, *Quadratic irrationalities and geometric properties of one-dimensional quasicrystals*, Preprint, Université de Montréal, CRM-2565, 1998.

[10] ———, *Substitution rules for 1-dimensional cut and project tilings*, Preprint, Université de Montréal, CRM-XXXX, 1999.

[11] R. Moody and J. Patera, *Quasicrystals and icosians*, J. Phys. A: Math. Gen. **26** (1994), 2829–2853.

[12] Jan Patera, *Methods of computer-based generation of quasicrystals*, Master's thesis, Czech Technical University, 1999, `email: patera@km1.fjfi.cvut.cz`.

[13] Jiří Patera, *Acceptance windows compatible with a quasicrystal fragment*, Order, Chance and Risk (F. Axel, ed.), Proc. Winter School (Les Houches, March 98, to appear).

[14] N. B. Slater, *Gaps and steps for the sequence* $n\theta \mod 1$, Proc. Camb. Phil. Soc. **73** (1967), 1115–1122.