Преобразование Синая-Арнольда и генераторы случайных чисел. Реализации на основе команд SSE.

Бараш Л.Ю. и Щур Л.Н.

Phys. Rev. E **73,** 036701 (2006)
Comp.Phys.Comm., готовится к печати (2009)

Молекулярная динамика и методы Монте-Карло имеют применения в разнообразных областях науки:

- в квантовой физике

K.S.D. Beach, P.A. Lee, P. Monthoux, Phys. Rev. Lett. **92** (2004) 026401

- в статистической физике

D.P. Landau and K.Binder, *A Guide to Monte Carlo Simulations in Statistical Physics*(Cambridge University Press,Cambridge, 2000)

- в ядерной физике

S.C. Pieper and R.B. Wiring, Ann. Rev. Nucl. Part. Sci., **51** (2001) 53

- в квантовой химии

A. Luchow, Ann. Rev. Phys. Chem., **51** (2000) 501

- в науках о материалах

A.R. Bizzarri, J. Phys.: Cond. Mat., **16** (2004) R83

Наиболее используемые RNGs могут быть поделены на **два основных класса:**

1. Линейно-конгруэнтные генераторы (LCG)
2. Генераторы линейного сдвига (GFSR)

Примеры современных модификаций к методам LCG и GFSR:

- Mersenne Twister (M. Matsumoto and T. Nishimura, ACM Trans. on Mod. and Comp. Sim., **8** (1998) 3).
- Комбинированные LCG-генераторы (P. L'Ecuyer, Oper. Res., **47** (1999) 159)
- Комбинированные Tausworthe генераторы (P. L'Ecuyer, Math. of Comp., **68** (1999) 261)

Мы предложим новый метод построения RNG с хорошими свойствами генерируемой последовательности псевдослучайных чисел.

Отличительные особенности метода:

Ансамбль ДС: ансамбль эргодических динамических систем используется вместо единичной системы.

Скрытые переменные: только небольшая часть генерируемой информации остается на выходе генератора. Это помогает избавиться от корреляций и усложняет расшифровку.

Длина периода: период генератора может быть настолько большим, насколько это требуется.

Базовый элемент генератора

Мы используем гиперболический автоморфизм единичного двумерного тора (преобразование Синая-Арнольда, или преобразование кошки). Это преобразование обладает хорошими стохастическими свойствами: эргодичность, перемешивание, чувствительность к начальным условиям, локальная расходимость траекторий.

$$M = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix} \in SL_2(\mathbb{Z}),$$

Базовый элемент генератора

$$M = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix} \in SL_2(\mathbb{Z}),$$

Собственные значения: $\lambda = (k \pm \sqrt{k^2 - 4})/2$, где $k = \mathrm{Tr}(M)$.

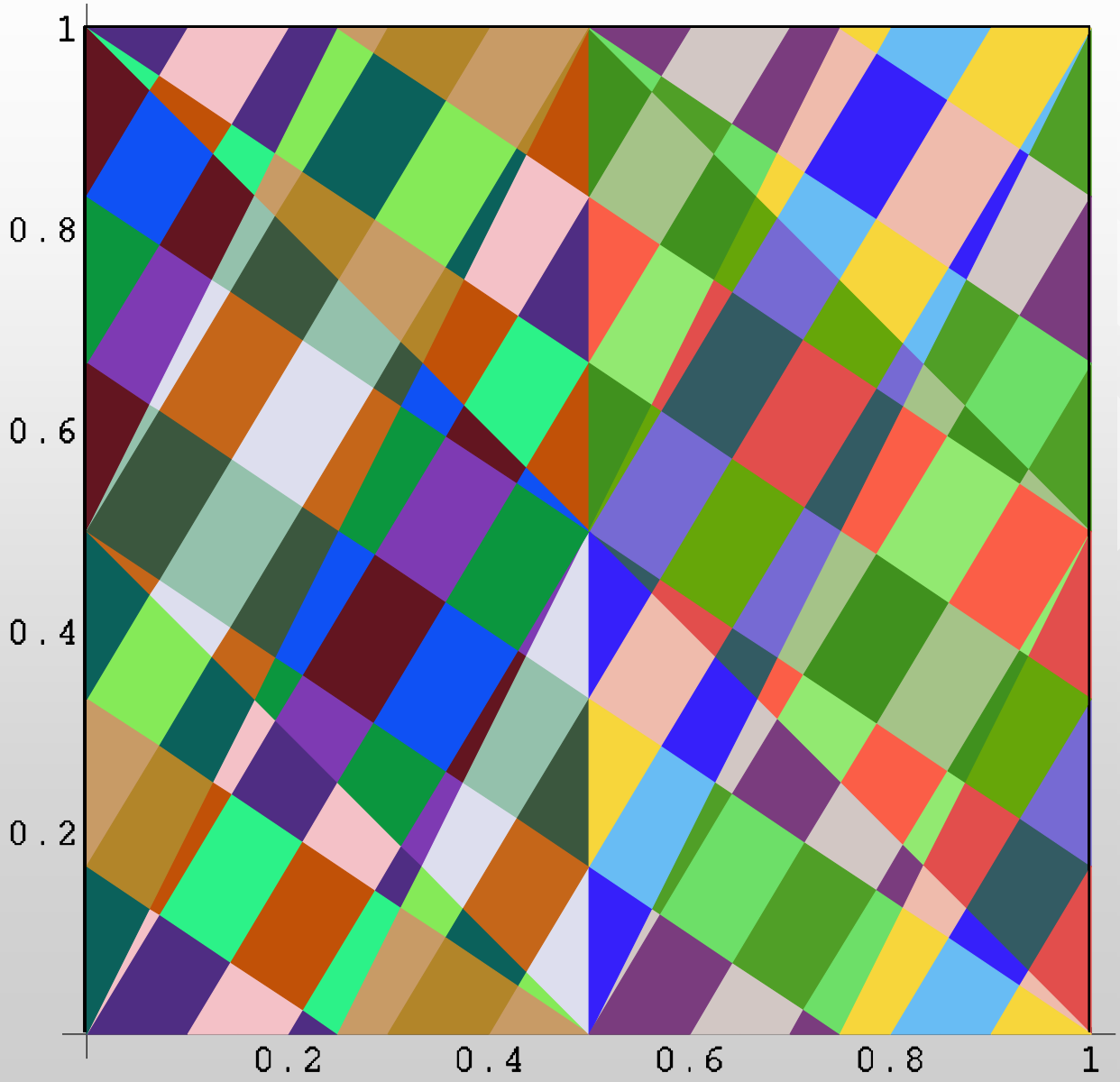
Если след удовлетворяет условию гиперболичности $|k| > 2$

то матрица M определяет диффеоморфизм Аносова двумерного тора.

$$H(p,q) = (k^2-4)^{-1/2} \sinh^{-1}((k^2-4)^{1/2}/2)(m_{12}p^2 - m_{21}q^2 + (m_{11}-m_{22})pq),$$

Здесь $k = \mathrm{Tr}(M) = m_{11} + m_{22} > 2.$

Области на торе и трех-битовые последовательности, генерируемые «преобразованием кошки»



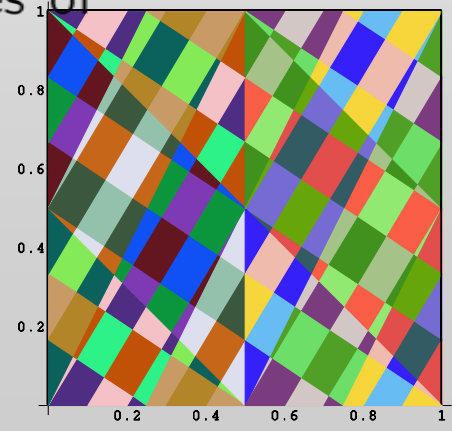
The exact areas $S(Z_{00000}), \ldots, S(Z_{11111})$ were calculated for various toral automorphisms, using CLN library for the exact rational arithmetics.

We **prove** analytically the following geometric propositions:

i) in any case, every subsequence of length 3, 2, or 1 respectively has the same probability 1/8, 1/4, or 1/2;

ii) if $k = \mathrm{Tr}(M)$ is an odd number, then every subsequence of length 4 has the same probability $P_0 = 1/16$;

ii) if k is even, then the probability of the subsequence 0000 depends only on the trace k of matrix M of the cat map, and it equals $P = P_0 \cdot k^2/(k^2 - 1)$, where $P_0 = 1/16$. The probability of 0000 automatically gives the probabilities of all other subsequences of length four.



Random walks test and the probabilities of subsequences of first bits for the single cat map.

It can be conjectured that if k is odd, then the probability of the subsequence 00000 equals $P_0 \cdot (1 + 1/(3k^2 - 6))$, where $P_0 = 1/32$.

The probabilities can thus be approximated as $P/P_0 = 1 + Bk^{-2}$ for large k, where $P_0 = 2^{-n}$ for subsequences of length $n = 4, 5$. Here, $B = 1$ when k is even and $n = 4$, $B = 1/3$ when k is odd and $n = 5$.

We conclude that the deviations found by the random walks test will vanish as the trace k increases.

In order to **determine the particular Random Number Generator**, one should define:

1. a finite set of states R,
2. an initial state $r_0 \in R$,
3. a finite set of output symbols U,
4. the transition function $T : R \to R$,
5. the output function $G : R \to U$.

The generator changes its state at each step n, calculating $r_n = T(r_{n-1})$ with the transition function T.

The sequence of pseudo random numbers is $(a_0, a_1, a_2, \ldots, a_n, \ldots)$. Each of these numbers is generated by the output function G: $a_n = G(r_n)$, and is also called the observation a_n.

Описание метода: построение RNG

Рассмотрим генератор с $R = L^s$, где $L = \{0, 1, 2, \ldots, g-1\} \times \{0, 1, 2, \ldots, g-1\}$ – решетка на торе, а s – положительное целое число. На практике, мы используем $g = 2^m$ и $g = 2^m - 1$, где $2^m - 1$ – простое число.

Состояние генератора случайных чисел преобразуется на каждом шаге при помощи преобразования кошки:

$$\begin{pmatrix} x_i^{(n)} \\ y_i^{(n)} \end{pmatrix} = M \begin{pmatrix} x_i^{(n-1)} \\ y_i^{(n-1)} \end{pmatrix} \pmod{g}, \qquad (1)$$

где s точек ($i = 0, 1, \ldots, (s-1)$) решетки L используются в расчете каждого шага.

Здесь M – матрица $M = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix} \in SL_2(\mathbb{Z})$, которая действует на решетке $g \times g$ тора.

Описание метода: построение RNG (продолжение)

Пусть $\alpha_i^{(n)}$ обозначает первый бит $x_i^{(n)}$: $\alpha_i^{(n)} = \lfloor 2x_i^{(n)}/g \rfloor$.

Выходная функция генератора $G : L^s \to \{0, 1, \ldots, 2^s - 1\}$ определяется как $a_n = \sum_{i=0}^{s-1} \alpha_i^{(n)} \cdot 2^i$.

Другими словами, a_n – s-битовое целое число, которое состоит из битов $\alpha_0^{(n)}$, $\alpha_1^{(n)}$, …, $\alpha_{s-1}^{(n)}$. В случае если $g = 2^m$, $a^{(n)}$ содержит в точности первые биты целых чисел $x_0^{(n)}$, $x_1^{(n)}$, …, $x_{s-1}^{(n)}$.

Мы видим, что построенный RNG содержит много скрытой информации. Именно, все биты точек $\binom{x_i^{(n)}}{y_i^{(n)}}$, которые не участвуют в построении значения $a^{(n)}$ являются скрытыми переменными.

Связи с другими последовательностями

$$\begin{pmatrix} x_i^{(n)} \\ y_i^{(n)} \end{pmatrix} = M \begin{pmatrix} x_i^{(n-1)} \\ y_i^{(n-1)} \end{pmatrix} (\mathrm{mod}\ g), \quad i = 0, 1, \ldots, (s-1).$$

Для каждого i последовательность $\{x_i^{(n)}\}$,
как и последовательность $\{y_i^{(n)}\}$,
удовлетворяет линейно-реккурентному соотношению
по модулю g:

$$x^{(n)} = kx^{(n-1)} - qx^{(n-2)} (\mathrm{mod}\ g)$$
$$y^{(n)} = ky^{(n-1)} - qy^{(n-2)} (\mathrm{mod}\ g),$$

Здесь $k = \mathrm{Tr}\ (M),\ q = \det M.$

Характеристический полином рекуррентного соотношения:
$f(x) = x^2 - kx + q.$

Методы для вычисления периода RNG

1. Для q=1: метод основан на расширенном и обобщенном методе из работы I.C.Percival и F.Vivaldi, Physica D, 25 (1987) 105.

2. Для простых q: метод основан на теории конечных полей.

To calculate period of our RNG we first need to obtain **the periods of trajectories of the single cat map** on the $2^m \times 2^m$ lattice on the torus.

It was pointed out by Percival and Vivaldi that for any given trace $k > 2$, there exists a unique map $M \in SL_2(\mathbb{Z})$ such that the direct connection between the properties of periodic orbits of the automorphism and the arithmetic of quadratic integers can be found.

Indeed, if we consider a matrix M such that

$$\begin{cases} \lambda &= m_{11} + \tau m_{21} \\ \lambda\tau &= m_{12} + \tau m_{22} \end{cases},$$

where τ is the base element of the ring of quadratic integers $R_D = \{a + b\tau : a, b \in \mathbb{Z}\}$ that contains λ, then $x' + y'\tau = \lambda(x + y\tau)$ is equivalent to $\binom{x'}{y'} = M\binom{x}{y}$ for any x, y, x', y'.

The period of an orbit containing the point $\binom{x}{y}$ of the integer lattice L equals the smallest integer T such that $\lambda^T z \equiv z \pmod{\langle g \rangle}$. Here $z = x + y\tau$, and $\langle g \rangle = \{ag + bg\tau : a, b \in \mathbb{Z}\}$ is the principal quadratic ideal generated by g.

To determine the **structure of periodic orbits** on the $2^m \times 2^m$ lattice, we prove the following propositions, that generalize results of Percival and Vivaldi.

1. For all m, either $T_{m+1} = 2T_m$ or $T_{m+1} = T_m$.

2. For all m, either $T'_m = T_m$ or $T'_m = T_{m-1}$.

3. For all $m \geq 3$, $T_m \neq T_{m-1} \Rightarrow T_{m+1} \neq T_m$.

4. If $m \geq 4$, $T_m \neq T_{m-1}$, and $T'_m = T_m/a$, where $a \in \{1, 2\}$, then $T'_{m+1} = T_{m+1}/a$.

T_m is the period of free orbits for $g = 2^m$, T'_m is the period of those ideal orbits for $g = 2^m$ that do not belong to the sublattice $\frac{g}{2} \times \frac{g}{2}$.

We find out the number of free orbits in the inert case.

There are $2^{2m}-1$ points on a lattice. The ideal orbits contain $2^{2m-2}-1$ points. Consequently, there are $(2^{2m}-2^{2m-2})/T_m = 3\cdot 2^{2m-2}/T_m$ free orbits.

In the typical inert case $T_m = 3\cdot 2^{m-2}$, and the phase space is divided in such a manner that:

- 3/4 of the phase space is swept by 2^m trajectories of period T_m,
- 3/16 of the phase space is swept by 2^{m-1} trajectories of period $T_{m-1} = T_m/2$,
- 3/64 of the phase space is swept by 2^{m-2} trajectories of period $T_{m-2} = T_m/4$,
- etc.

Number of the orbits having the huge period $\sim 2^{m-2}$ is also huge $\sim 2^{m+1}$.

The RNG Period

These results we proved for the single cat map help us to understand the periodic properties of the ensemble of the cat maps, and thus the period of the proposed RNG.

Indeed, at least one of the s initial points $\binom{x_i^{(0)}}{y_i^{(0)}}$ belongs to a free orbit, with the probability $(1 - 4^{-s})$ in the inert case.

It follows that the period T of the sequence $\{a_n\}$ **equals the period** T_m **of free orbits** of the cat map for the overwhelming majority of RNG initial conditions.

Probability that **two arbitrary points** of the $2^m \times 2^m$ lattice on the torus **belong to the same orbit** of the cat map equals $9/(7 \cdot 2^{m+2})$ in the inert case.

The probability that s **arbitrary points** of the lattice **do not belong to** s **different orbits** of the cat map (i.e., that at least two of the points belong to the same orbit) is $9s(s-1)/(7 \cdot 2^{m+3})$.

For sufficiently large m these probabilities are quite small. This **guarantees the absence of the correlations** between different bits $\alpha_i^{(n)}$ of the output a_n for the typical initial conditions.

Proper Initialization for $q = 1$:

1. Norms of all points should be different modulo 256. In particular, this guarantees that the initial points $\binom{x_i^{(0)}}{y_i^{(0)}}$, $i = 0, 1, \ldots, (s-1)$ belong to different orbits of the cat map, and that none of the symmetries may convert one orbit to another

2. At least one point should belong to a free orbit, i.e. at least one of the coordinates x or y should be an odd number. This guarantees that the period length is not smaller than T_m

Norms of the orbits for $q = 1$.

$\alpha = a + b\sqrt{D}$; $N(\alpha) = \alpha\alpha^* = a^2 - b^2 D$.

If $\langle 2 \rangle$ is inert, then $\binom{x}{y} \Leftrightarrow x + y\tau$, where

$$\tau = \frac{1 + \sqrt{D}}{2}$$

$$N\binom{x}{y} = x^2 + xy - \frac{D-1}{4}y^2$$

$$x' + y'\tau = \lambda(x + y\tau) \pmod{\langle 2^m \rangle}$$

$$N(\lambda) = 1$$

$$N\binom{x'}{y'} \equiv N\binom{x}{y} \pmod{2^m}.$$

Therefore, the norm modulo g is a characteristic of the whole orbit.

Example of **proper initialization** for $q \neq 1$: GM19 and GM31.

$$x_i^{(0)} = x_{i \cdot A}$$
$$x_i^{(1)} = x_{i \cdot A+1},$$

where $, i = 0, 1, \ldots, 31$, and A is a value of the order of $(p^2-1)/32$.

Batteries of stringent statistical tests: test results

Numbers of statistical tests with p-values outside the interval $[10^{-2}, 1 - 10^{-2}]$, $[10^{-5}, 1 - 10^{-5}]$, $[10^{-10}, 1 - 10^{-10}]$.

Generator	k	q	SmallCrush	Diehard	Crush	Bigcrush
GS	3	1	0, 0, 0	44, 29, 29	20, 16, 14	22, 20, 19
GR	3	1	0, 0, 0	5, 0, 0	5, 1, 0	15, 10, 7
GSI	11	1	0, 0, 0	1, 0, 0	10, 1, 0	13, 7, 6
GRI	11	1	1, 0, 0	6, 0, 0	5, 0, 0	13, 6, 5
GM19	15	28	0, 0, 0	2, 0, 0	2, 0, 0	3, 0, 0
GM31	7	11	0, 0, 0	2, 0, 0	3, 0, 0	1, 0, 0
RAND	–	–	13, 13, 12	88, 84, 82	102, 100, 100	85, 83, 79
RAND48	–	–	5, 5, 3	27, 23, 22	22, 20, 20	27, 23, 22
RANDOM	–	–	3, 2, 2	17, 15, 15	13, 11, 10	21, 15, 14
MRG32k3a	–	–	1, 0, 0	3, 0, 0	4, 0, 0	2, 0, 0
LFSR113	–	–	0, 0, 0	3, 0, 0	8, 6, 6	8, 3, 3
MT19937	–	–	0, 0, 0	2, 0, 0	1, 0, 0	4, 0, 0

Speed and parameters for the RNGs

Generator	g	s	k	q	Rotation	SSE2	Period	CPU-time
GS	2^{32}	32	3	1	$-$	$-$	$3.2 \cdot 10^{9}$	55.4
GS-SSE	2^{32}	32	3	1	$-$	$+$	$3.2 \cdot 10^{9}$	2.49
GR-SSE	2^{32}	32	3	1	$+$	$+$	$3.2 \cdot 10^{9}$	2.79
GSI-SSE	2^{32}	32	11	1	$-$	$+$	$3.2 \cdot 10^{9}$	3.66
GRI	2^{32}	32	11	1	$+$	$-$	$3.2 \cdot 10^{9}$	78.2
GRI-SSE	2^{32}	32	11	1	$+$	$+$	$3.2 \cdot 10^{9}$	4.03
GM19	$2^{19}-1$	32	6	3	$+$	$-$	$2.7 \cdot 10^{11}$	120.5
GM19-SSE	$2^{19}-1$	32	6	3	$+$	$+$	$2.7 \cdot 10^{11}$	6.11
GM31-SSE	$2^{31}-1$	32	7	11	$+$	$+$	$4.6 \cdot 10^{18}$	8.86
RAND	$-$	$-$	$-$	$-$	$-$	$-$	$2.1 \cdot 10^{9}$	2.48
RAND48	$-$	$-$	$-$	$-$	$-$	$-$	$2.8 \cdot 10^{14}$	4.64
RANDOM	$-$	$-$	$-$	$-$	$-$	$-$	$3.4 \cdot 10^{10}$	1.88
MT19937	$-$	$-$	$-$	$-$	$-$	$-$	$4.3 \cdot 10^{6001}$	2.45
MRG32k3a	$-$	$-$	$-$	$-$	$-$	$-$	$3.1 \cdot 10^{57}$	11.14
LFSR113	$-$	$-$	$-$	$-$	$-$	$-$	$1.0 \cdot 10^{34}$	2.98

SSE2 алгоритм для генератора GRI

```
unsigned long x[4],y[4];

[.......]

asm("movaps (%0),%%xmm0\n" \
    "movaps (%1),%%xmm1\n" \
    "paddd  %%xmm1,%%xmm0\n" \
    "paddd  %%xmm1,%%xmm0\n" \
    "movaps %%xmm0,%%xmm2\n" \
    "pslld  $2,%%xmm0\n" \
    "paddd  %%xmm1,%%xmm0\n" \
    "movaps %%xmm0,(%0)\n" \
    "psubd  %%xmm2,%%xmm0\n" \
    "movaps %%xmm0,(%1)\n" \
    ""::"r"(x),"r"(y));
```

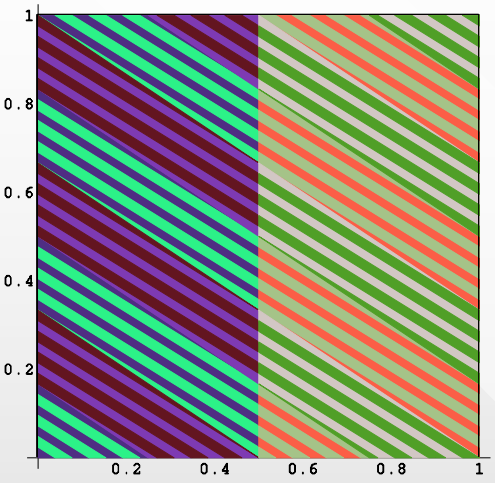
```
unsigned long i,newx[4],x[4],y[4];

[.......]

for(i=0;i<4;i++){
  newx[i]=4*x[i]+9*y[i];
  y[i]=3*x[i]+7*y[i];
  x[i]=newx[i];
}
```

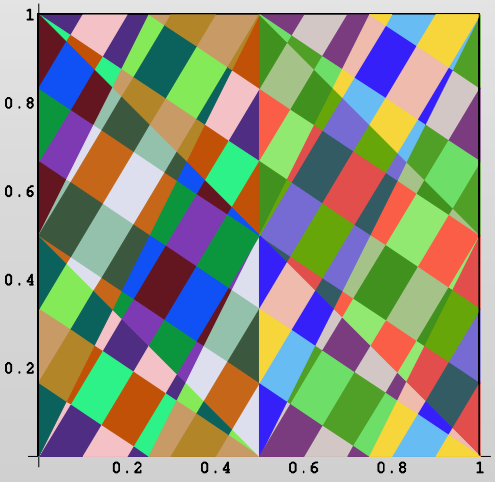
SSE2 алгоритм для упаковки

```
unsigned long x[16],output;

  [.......]

  asm("movaps (%1),%%xmm0\n" \
      "movaps 16(%1),%%xmm1\n" \
      "movaps 32(%1),%%xmm2\n" \
      "movaps 48(%1),%%xmm3\n" \
      "psrld  $31,%%xmm0\n" \
      "psrld  $31,%%xmm1\n" \
      "psrld  $31,%%xmm2\n" \
      "psrld  $31,%%xmm3\n" \
   "packssdw %%xmm1,%%xmm0\n" \
   "packssdw %%xmm3,%%xmm2\n" \
   "packsswb %%xmm2,%%xmm0\n" \
      "psllw  $7,%%xmm0\n" \
      "pmovmskb %%xmm0,%0\n" \
      "":"=r"(output):"r"(x));
```

```
const unsigned long halfg=2147483648;
unsigned long x[16],i,output=0,bit=1;

  [.......]

for(i=0;i<16;i++){
  output+=((x[i]<halfg)?0:bit;
  bit*=2;
}
```

Итоги





- Построен генератор случайных чисел на основе параллельной эволюции ансамбля преобразований кошки (преобразований кошки Синая-Арнольда).
- Каждое преобразование кошки дает один бит, поэтому чтобы построить s-битовое случайное число, нужно вычислять s преобразований кошки.
- Вычислен период генератора при помощи наших обобщений теории Percival-Vivaldi, а также при помощи использования теории конечных полей.
- Найдены корреляции в единичном преобразовании кошки, и предложен способ уменьшения этих корреляций.
- Представлены эффективные практические реализации для RNGs. Кроме того, генераторы протестированы современными и мощными батареями статистических тестов.
- Числа на выходе генератора сложно расшифровать, поскольку большая часть информации о состоянии ансамбля преобразований кошки является скрытой информацией.