# Funkce vektorové náhodné proměnné

náhodná proměnná $\boldsymbol{x} = (x_1, x_2, \ldots x_m)$ popsaná celkovou hustotou pravděpodobnosti $f(\boldsymbol{x})$

nová náhodná proměnná $\boldsymbol{a} = (a_1, a_2, \ldots a_m)$ vytvořená transformací $\boldsymbol{a} = \boldsymbol{h}(\boldsymbol{x})$

$$\begin{aligned} a_1 &= h_1(x_1, x_2, \ldots, x_m) \\ &\ldots \\ a_m &= h_m(x_1, x_2, \ldots, x_m) \end{aligned}$$

zpětná transformace $\boldsymbol{x} = \boldsymbol{h}^{-1}(\boldsymbol{a})$

$$\begin{aligned} x_1 &= h_1^{-1}(a_1, a_2, \ldots, a_m) \\ &\ldots \\ x_m &= h_m^{-1}(a_1, a_2, \ldots, a_m) \end{aligned}$$

celková hustota pravděpodobnosti nové náhodné proměnné ***a*** je $g(\boldsymbol{a}) = f\left(\boldsymbol{h}^{-1}(\boldsymbol{a})\right) |\mathbf{J}|,$

kde **J** je Jacobiho matice

$$\mathbf{J} = \begin{pmatrix} \frac{\partial h_1^{-1}}{\partial a_1} & \frac{\partial h_1^{-1}}{\partial a_2} & \ldots & \frac{\partial h_1^{-1}}{\partial a_m} \\ \ldots & & & \\ \frac{\partial h_m^{-1}}{\partial a_1} & \frac{\partial h_m^{-1}}{\partial a_2} & \ldots & \frac{\partial h_m^{-1}}{\partial a_m} \end{pmatrix}$$

# Funkce náhodné proměnné – konvoluce

náhodné proměnné $(x, y)$ s celkovou hustotou pravděpodobnosti $f(x,y)$

nová náhodná proměnná $u = x + y$ pomocná proměnná $v = y$

$u = x + y$ zpětná transformace $x = u - v$

$v = y$ $y = v$

celková hustota pravděpodobnosti nových náhodných proměnných $g(u, v) = f(u - v, v)|\mathbf{J}|$

$$\mathbf{J} = \begin{pmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix} \qquad |\mathbf{J}| = 1 \longrightarrow g(u, v) = f(u - v, v)$$

marginální hustota pravděpodobnosti $g_u(u) = \int_{-\infty}^{\infty} g(u, v)\, dv = \int_{-\infty}^{\infty} f(u - v, v)\, dv$

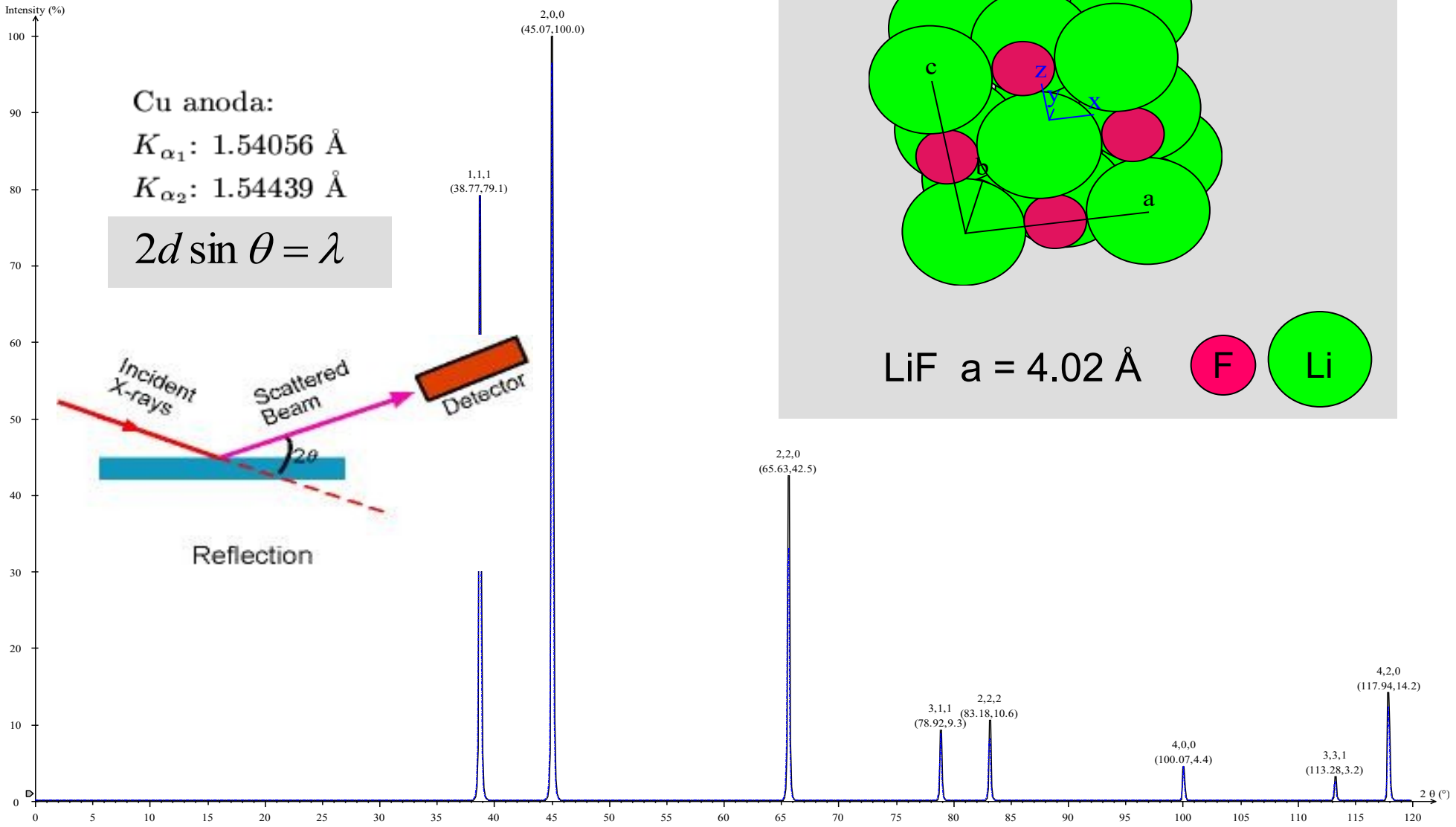
pokud jsou náhodné proměnné $x$, $y$ nezávislé $f(x, y) = f_x(x) f_y(y)$

$$g_u(u) = \int_{-\infty}^{\infty} f_x(u - v) f_y(v)\, dv = f_x \star f_y$$

konvoluce

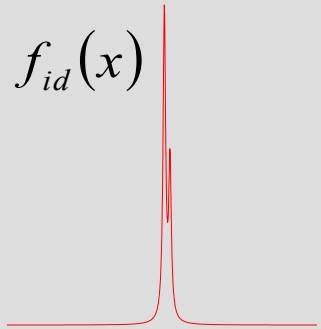
# Funkce náhodné proměnné – konvoluce

difrakce rtg. záření

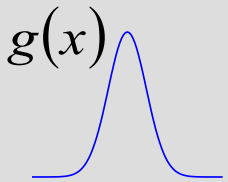


# Funkce náhodné proměnné – konvoluce

„ideální“ spektrum




spektrometr

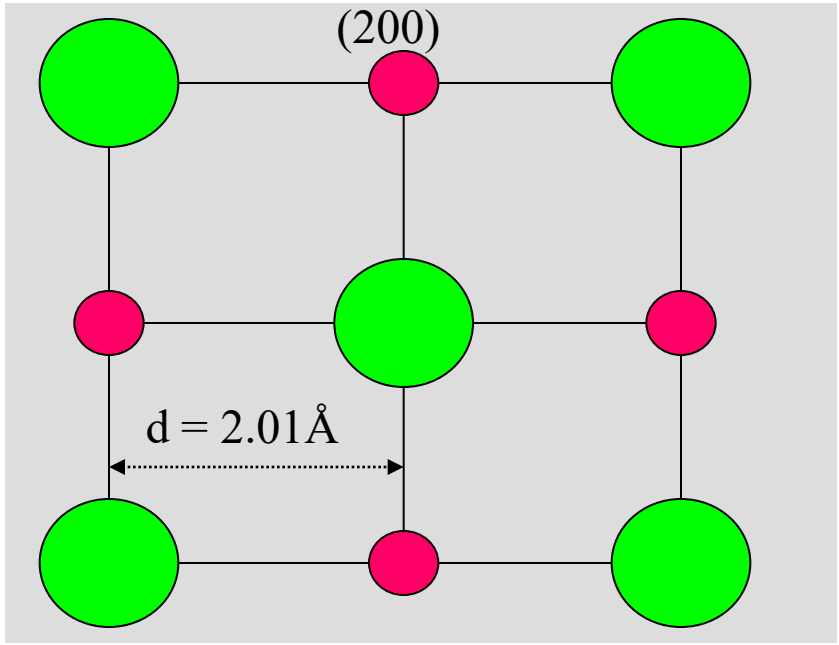



„reálné“ spektrum


$$f(x) = \int_{-\infty}^{\infty} f_{id}(x-y)g(y)dy$$

# Funkce náhodné proměnné – konvoluce

difrakce rtg. záření




Cu anoda:

$K_{\alpha_1}$: 1.54056 Å

$K_{\alpha_2}$: 1.54439 Å

$$2d \sin\theta = \lambda$$

# Funkce náhodné proměnné – konvoluce

Pearson VII

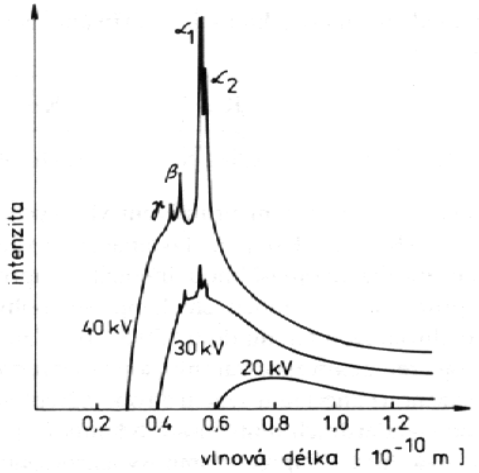
$$I(x) = \frac{I_0}{\left[1 + C(x - x_0)^2\right]^m}$$

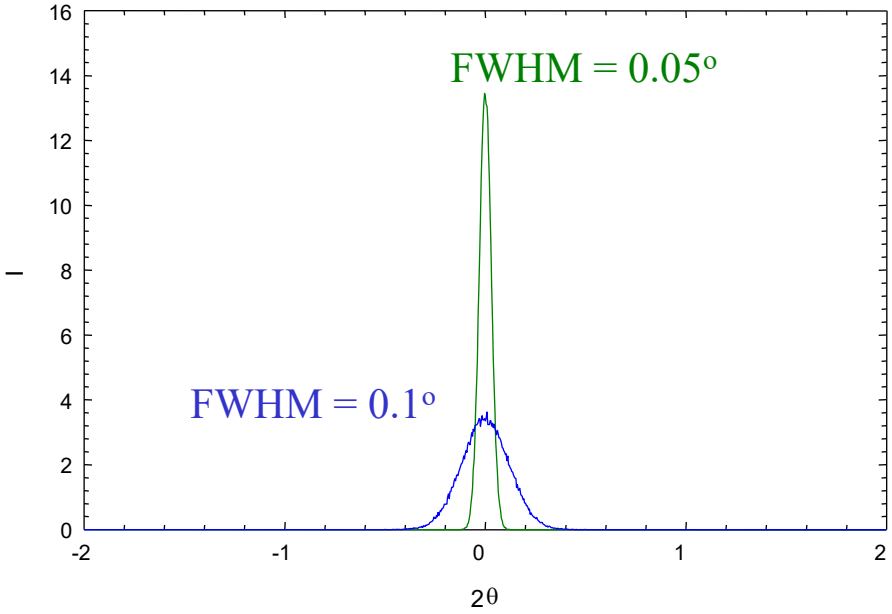
$$C = \frac{\sqrt[m]{2} - 1}{w^2} \qquad \text{FWHM} = 2w$$

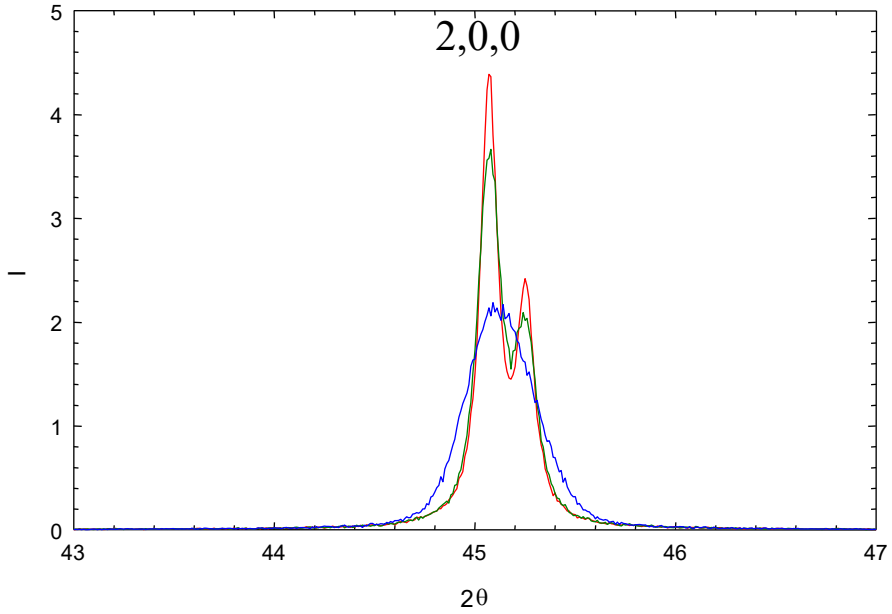
Cu anoda:

$K_{\alpha_1}$: 1.54056 Å

$K_{\alpha_2}$: 1.54439 Å



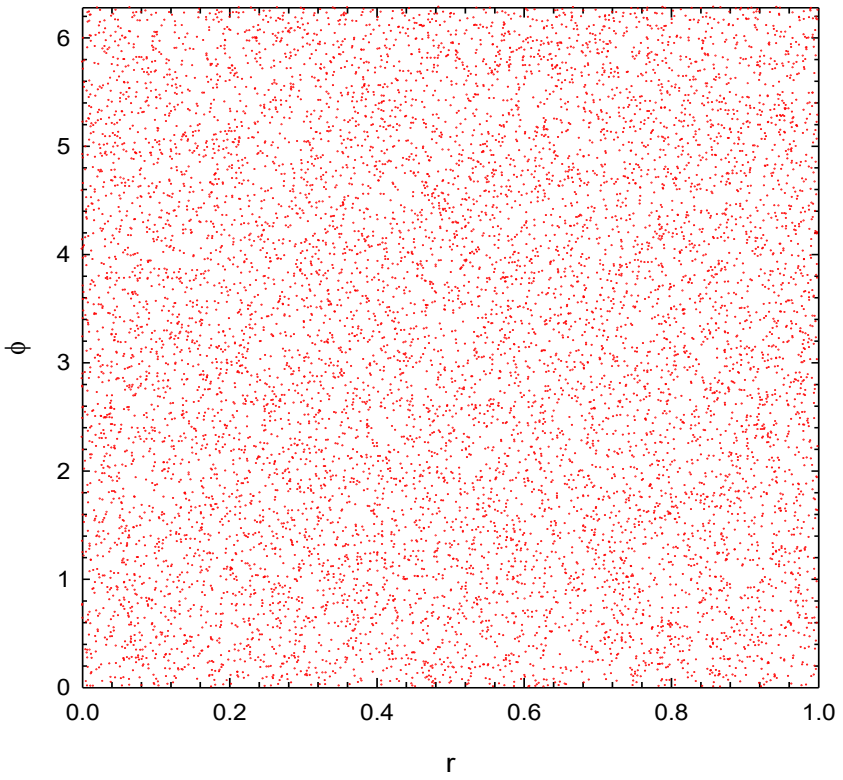



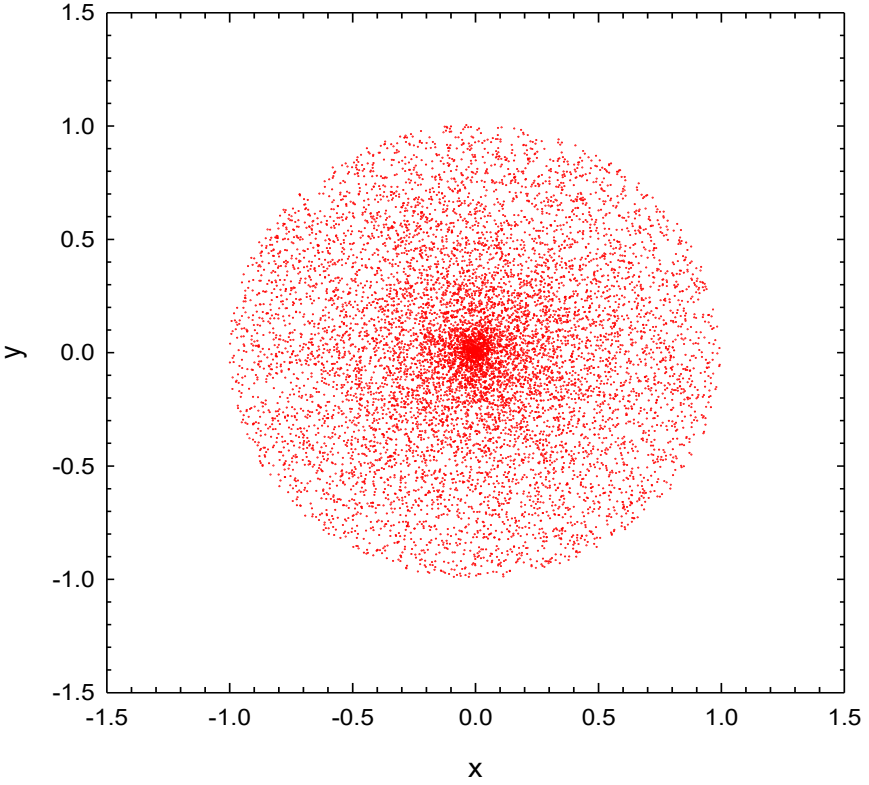
# Funkce náhodné proměnné – transformace souřadnic

$$g(x,y)=\frac{1}{r}f(r,\varphi)$$

$r \in U(0,1) \quad \varphi \in U(0,2\pi)$

$x = r\cos\varphi \qquad y = r\sin\varphi$



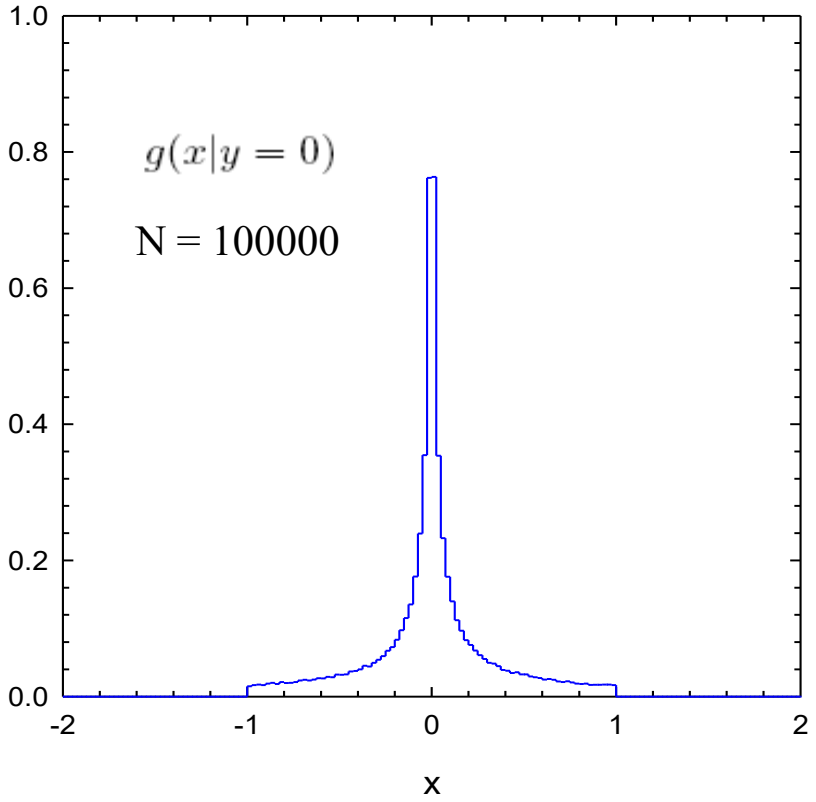



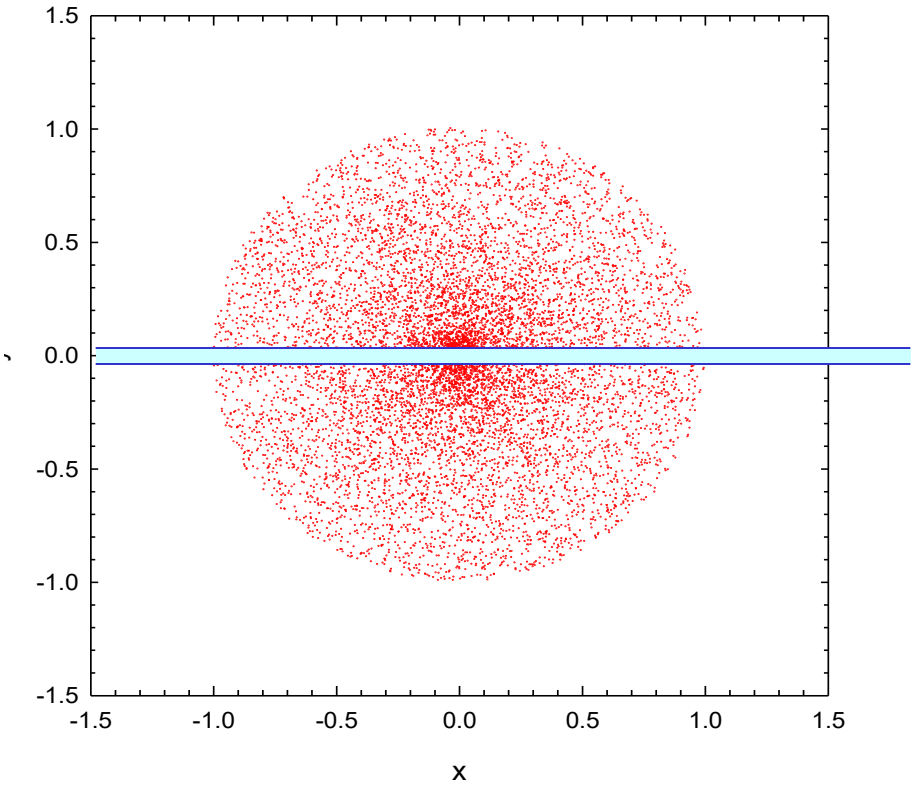
# Funkce náhodné proměnné – transformace souřadnic

$$g(x,y)=\frac{1}{r}f(r,\varphi)$$

$r \in U(0,1) \quad \varphi \in U(0,2\pi)$

$x = r\cos\varphi \qquad y = r\sin\varphi$



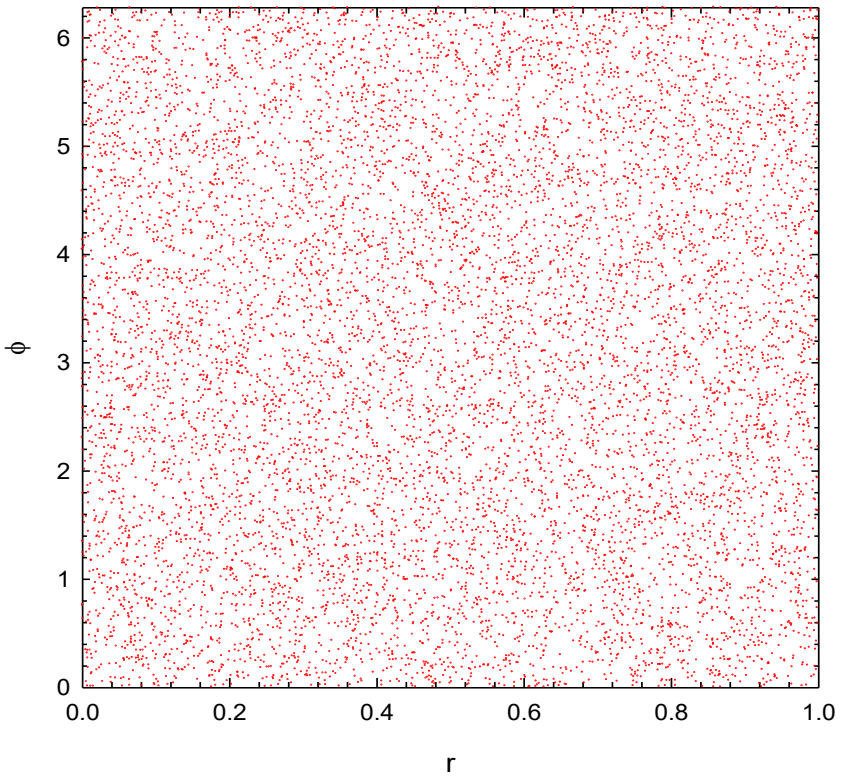



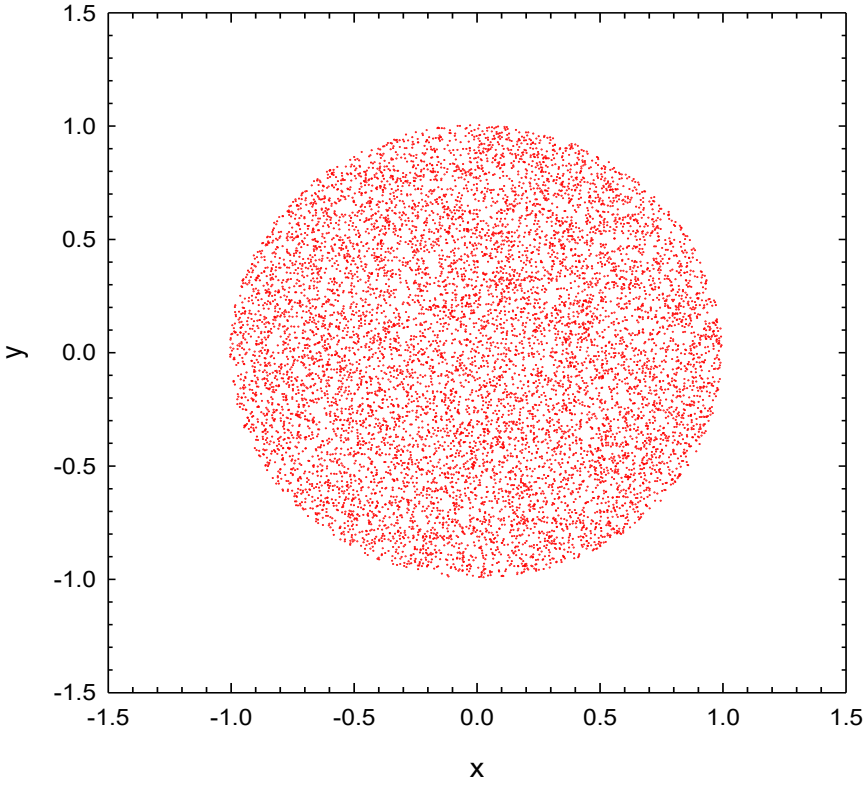
# Funkce náhodné proměnné – transformace souřadnic

$$g(x,y) = 2f(r,\varphi)$$

$r \in U(0,1) \quad \varphi \in U(0,2\pi)$

$x = \sqrt{r}\cos\varphi \qquad y = \sqrt{r}\sin\varphi$



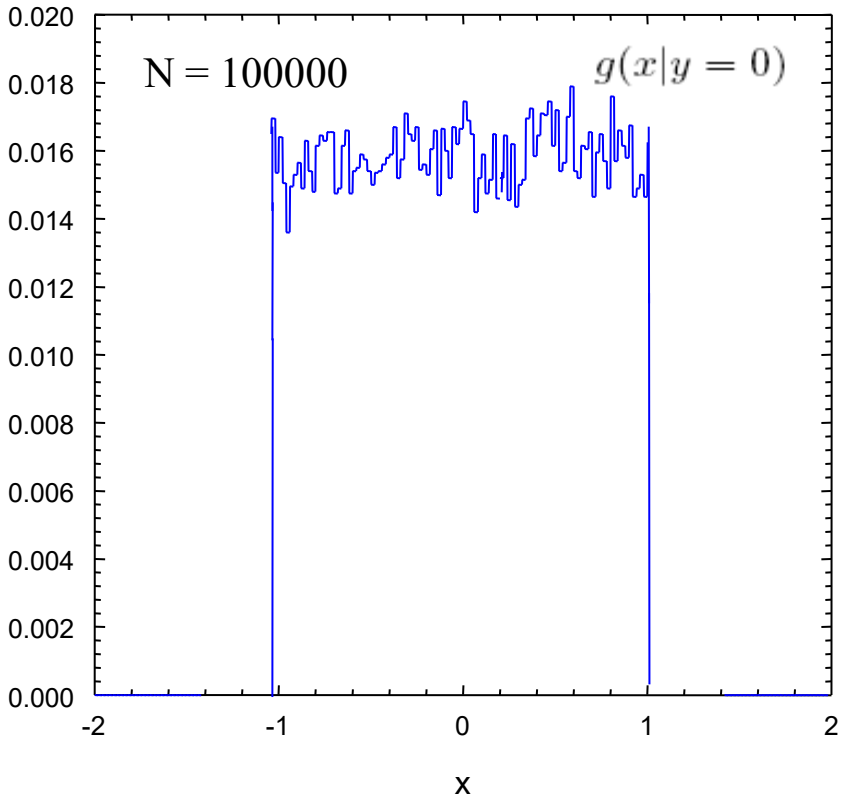



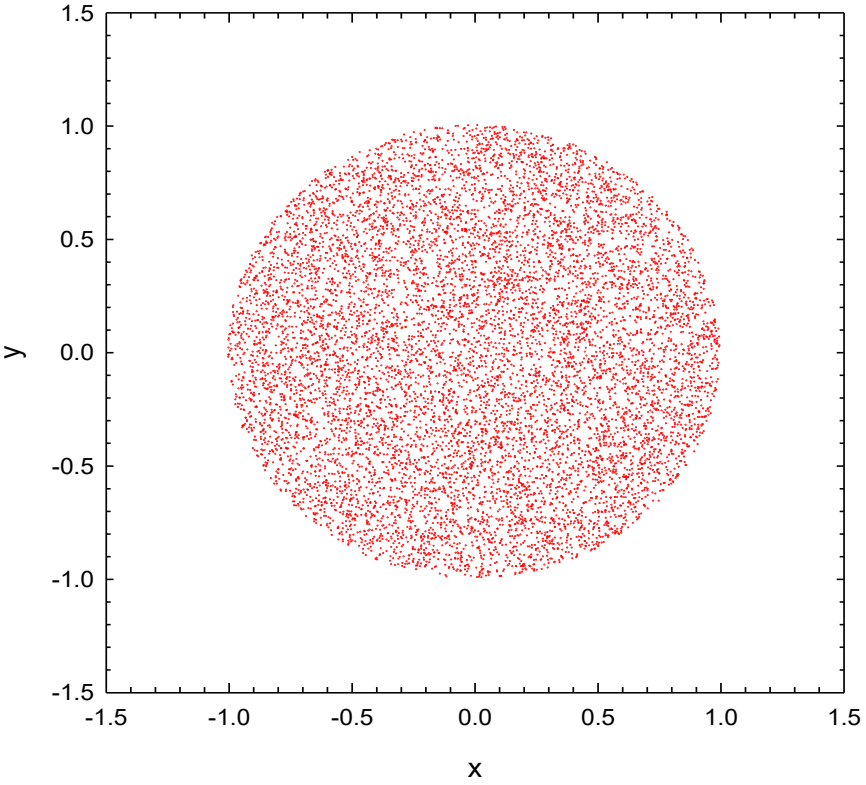
# Funkce náhodné proměnné – transformace souřadnic

$$g(x,y) = 2f(r,\varphi)$$

$r \in U(0,1) \quad \varphi \in U(0,2\pi)$

$x = \sqrt{r}\cos\varphi \qquad y = \sqrt{r}\sin\varphi$



# Lineární kongruentní generátor

## smíšený generátor

$$I_{j+1} = aI_j + c \pmod{m}$$

$$a, c, m \in \mathbf{N}$$

$$x_j = I_j / m$$

$$0 \le x_j < 1 \quad x_j \in U(0,1)$$

$m \approx 2^{32}$

maximální perioda $m$

$I_0$ : semínko

korelace

## čistě multiplikativní generátor

$$I_{j+1} = aI_j \pmod{m}$$

$$a, m \in \mathbf{N}$$

$$x_j = I_j / m$$

$$0 \le x_j < 1 \quad x_j \in U(0,1)$$

$a = 7^5 = 16807$

$m = 2^{31} - 1 = 2147483647$

perioda $2^{31} - 2 \approx 2.1 \times 10^9$

# Lineární kongruentní generátor - implementace

## čistě multiplikativní generátor

$$I_{j+1} = aI_j \pmod{m}$$

$$a, m \in \mathbf{N}$$

$$x_j = I_j / m$$

$$0 \le x_j < 1 \quad x_j \in U(0,1)$$

$$a = 7^5 = 16807$$

$$m = 2^{31} - 1 = 2147483647$$

$$q = 127773 \qquad r = 2836$$

## Faktorizace $m$ (Schragerův algoritmus)

$$m = aq + r, \quad q = [m/a], \; r = m \bmod a$$

$$r < q$$

$$\Downarrow$$

$$0 < I_j < m-1 \Rightarrow \begin{cases} 0 \le a(I_j \bmod q) \le m-1 \\ 0 \le r[I_j/q] \le m-1 \end{cases}$$

$$aI_j \bmod m = \begin{cases} a(I_j \bmod q) - r[I_j/q], & \text{je - li tento výraz} \ge 0 \\ a(I_j \bmod q) - r[I_j/q] + m, & \text{jinak} \end{cases}$$

# Lineární kongruentní generátor - implementace

```
float ran0(int *p_i0)
{
          #define a (16807)
          #define m (2147483647)
          #define q (127773)
          #define r (2836)

          int k,i0;
          float x;

          i0=*p_i0;
          k=i0/q;                          //[I0/q]
          i0=a*(i0-k*q)-r*k; //a(I0 mod q)-r[I0/q]
          if(i0<0) i0=i0+m;
          x=(float)i0/m; //converze na realne cislo z intervalu (0,1)
          *p_i0=i0;
          return(x);
}
```

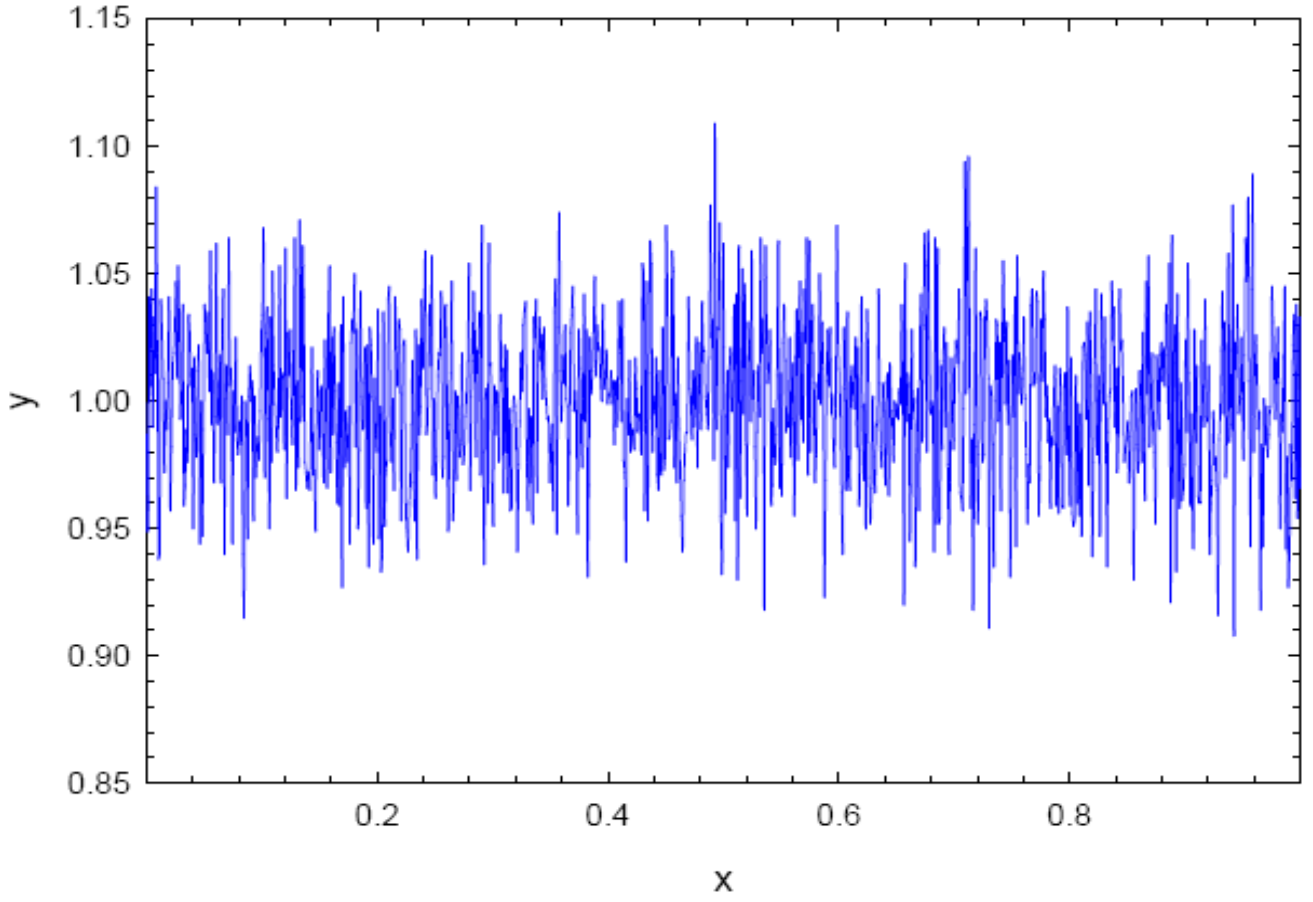
# Lineární kongruentní generátor - implementace

```
void main()
{
        FILE *f,*g;
        int iseed;
        int i;
        float x;


        if((f=fopen("iseed.dat","r"))==NULL) iseed=123456789;
                else
                {
                        f=fopen("iseed.dat","r");
                        fscanf(f,"%d",&iseed);
                        fclose(f);
                }

        g=fopen("ran0.dat","w");
        for(i=0;i<1000000;i++)
        {
                x=ran0(&iseed);
                fprintf(g,"%f\n",x);
        }

        f=fopen("iseed.dat","w");
        fprintf(f,"%d",iseed);
        fclose(f);
        fclose(g);
}
```

# Lineární kongruentní generátor - implementace

N = 1000000

multiplikativní generátor, $a$ = 16807, $m = 2^{31}-1$



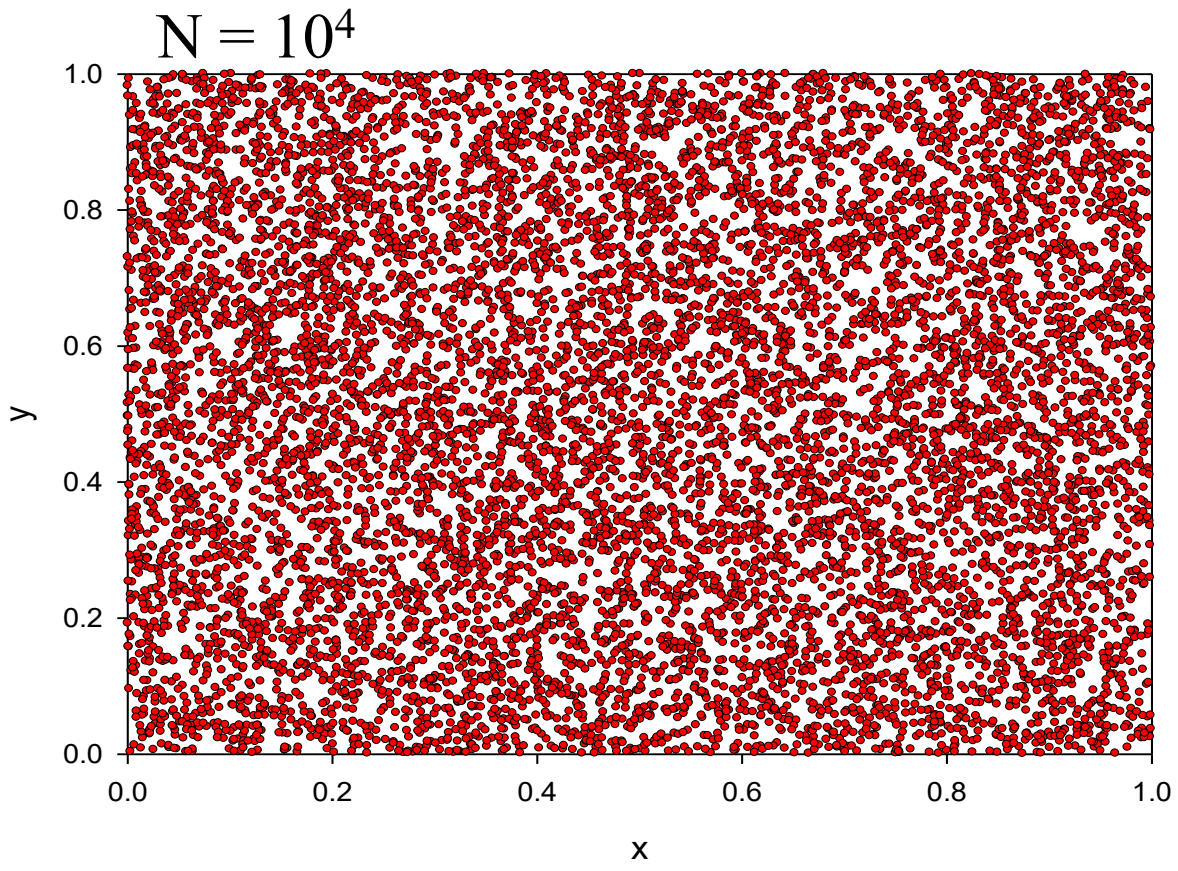
# Lineární kongruentní generátor – test

IBM RANDU

$$I_{j+1} = a\, I_j \pmod{m}$$

$$a = 65539$$

$$m = 2^{31}$$



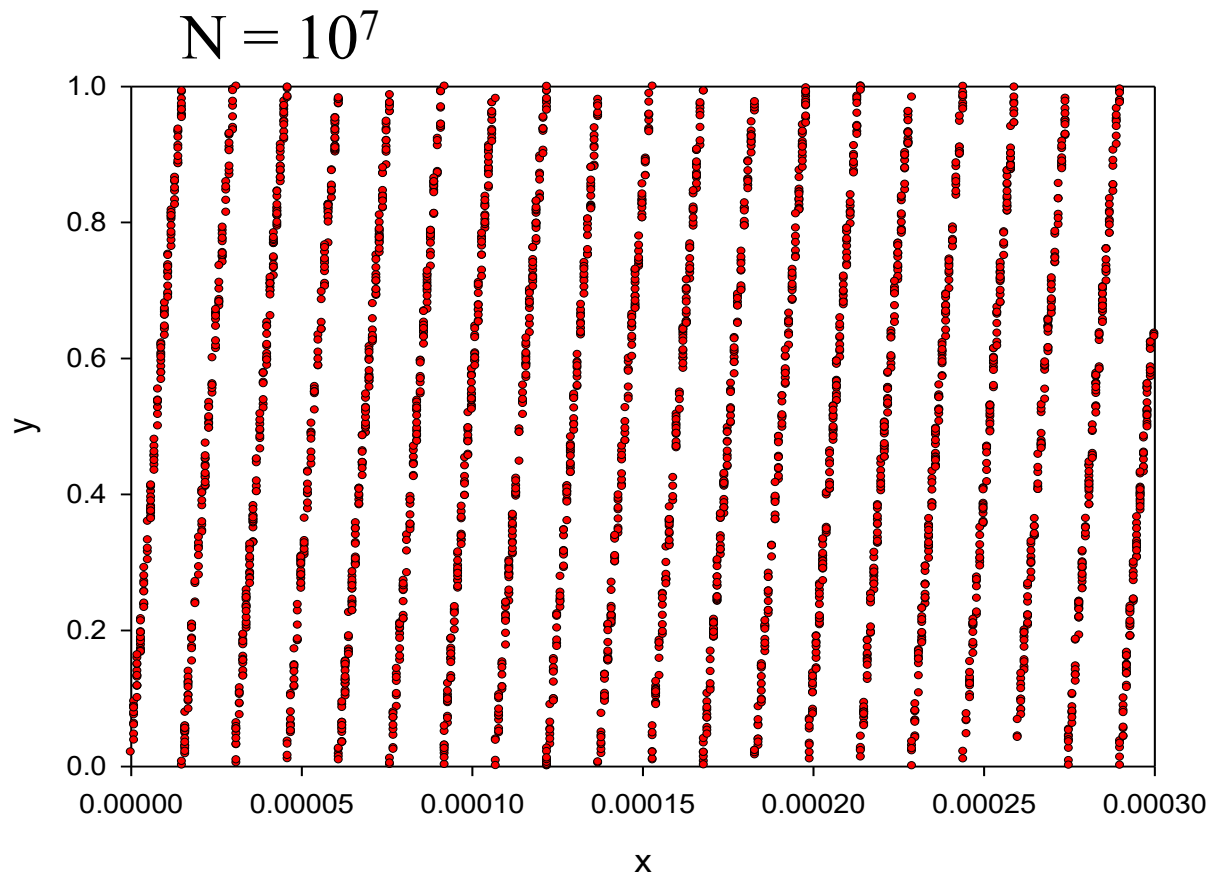
# Lineární kongruentní generátor – test

IBM RANDU

$$I_{j+1} = a\, I_j \pmod{m}$$

$$a = 65539$$

$$m = 2^{31}$$



# Lineární kongruentní generátor – sériová korelace

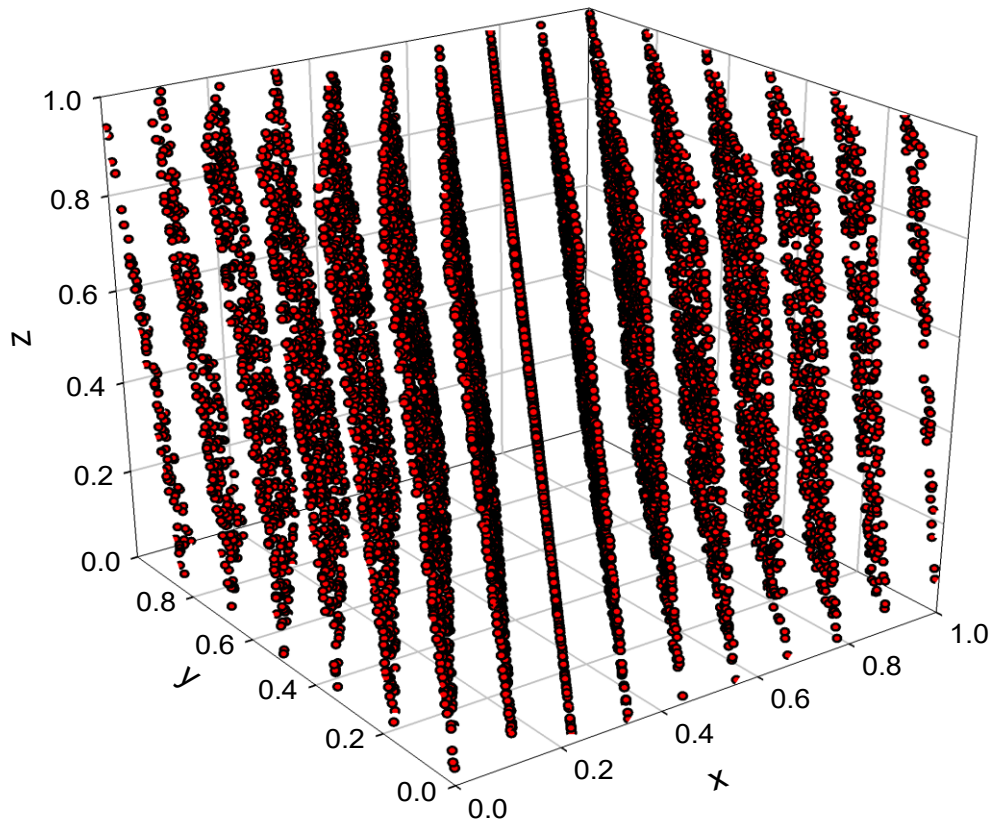
IBM RANDU

$$I_{j+1} = a\, I_j \pmod{m}$$

$$a = 65539$$

$$m = 2^{31}$$

$$N = 10^4$$




“ We guarantee that each number is random individually,
but we don’t guarantee that more than one of them is random.”

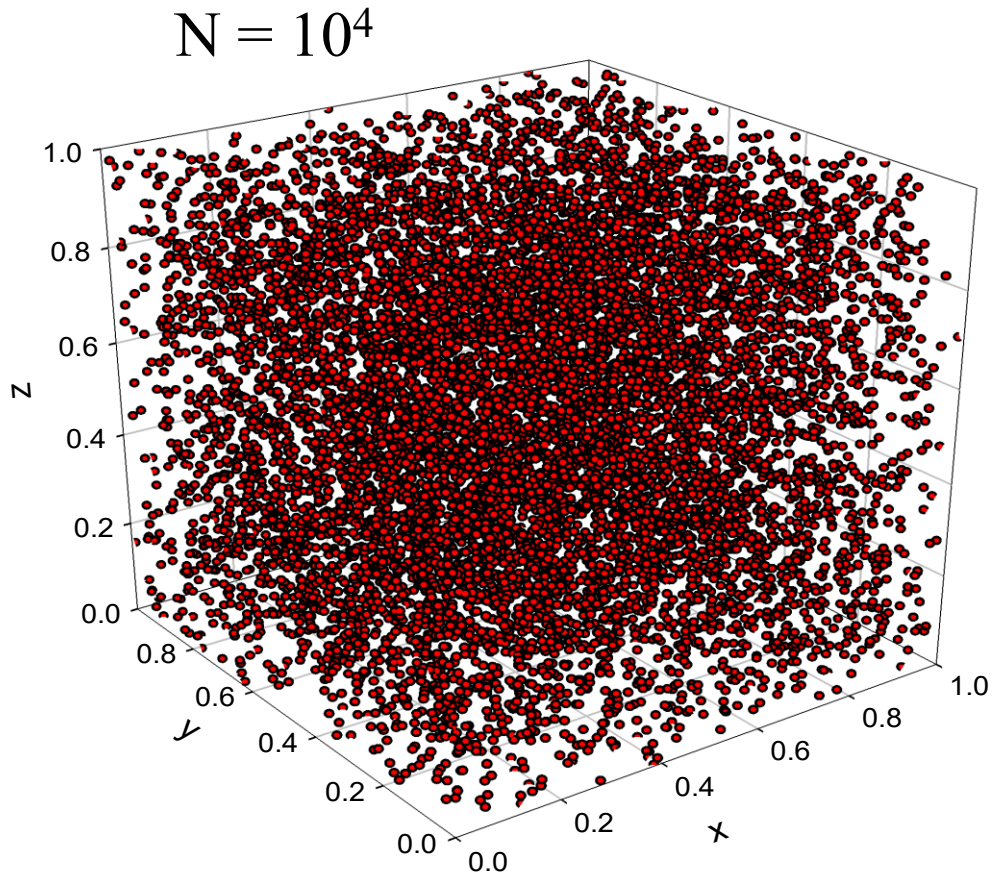
# Lineární kongruentní generátor – sériová korelace

RAN0

$$I_{j+1} = a\, I_j \pmod{m}$$

$$a = 16807$$

$$m = 2^{31} - 1$$



# Lineární kongruentní generátor – posuvný registr

RAN2 – L'Ecuyer

$$I_{j+1} = a\, I_j \pmod{m}$$

$$a_1 = 40014$$

$$m_1 = 2147483563$$

$$a_2 = 40692$$

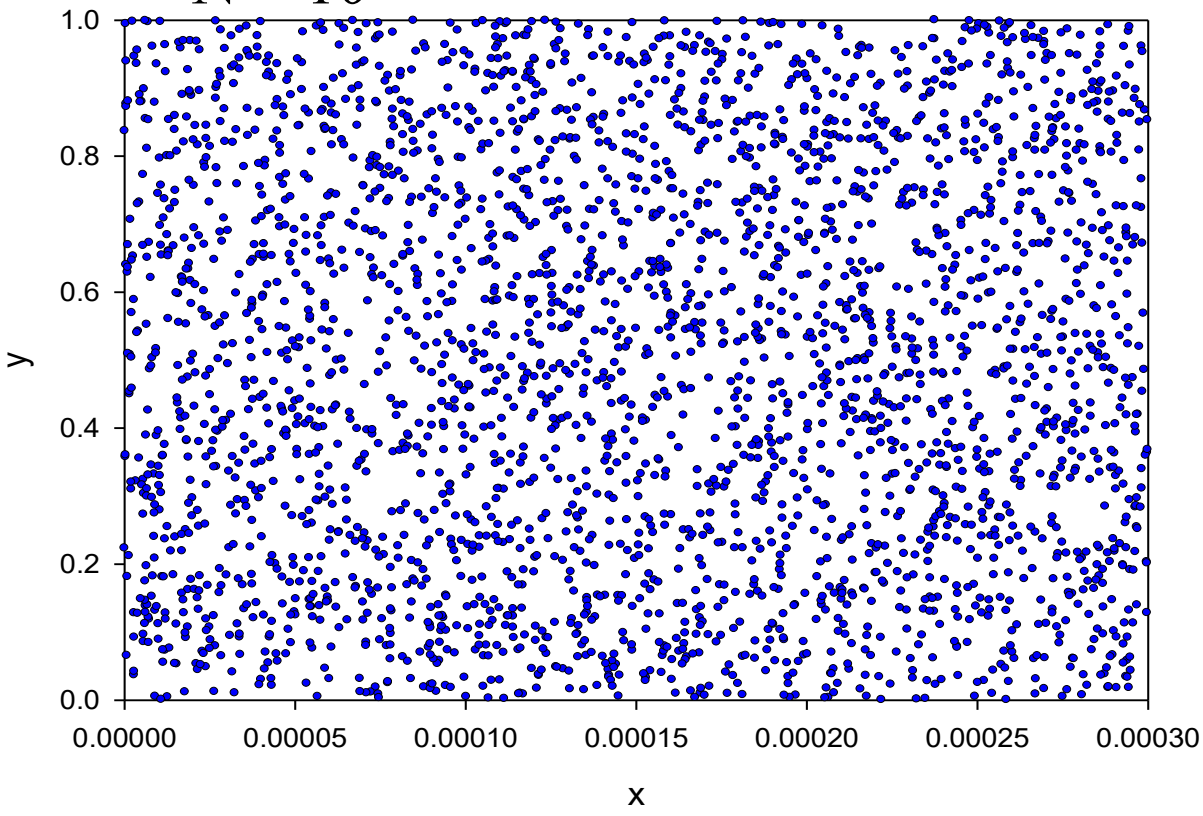
$$m_2 = 2147483399$$

$$I_j = I_{j,1} + I_{j,2} \pmod{m_1 + m_2}$$

+ posuv registru

perioda ≈ 2.3 × $10^{18}$

N = $10^7$



# Lineární kongruentní generátor – barevný test

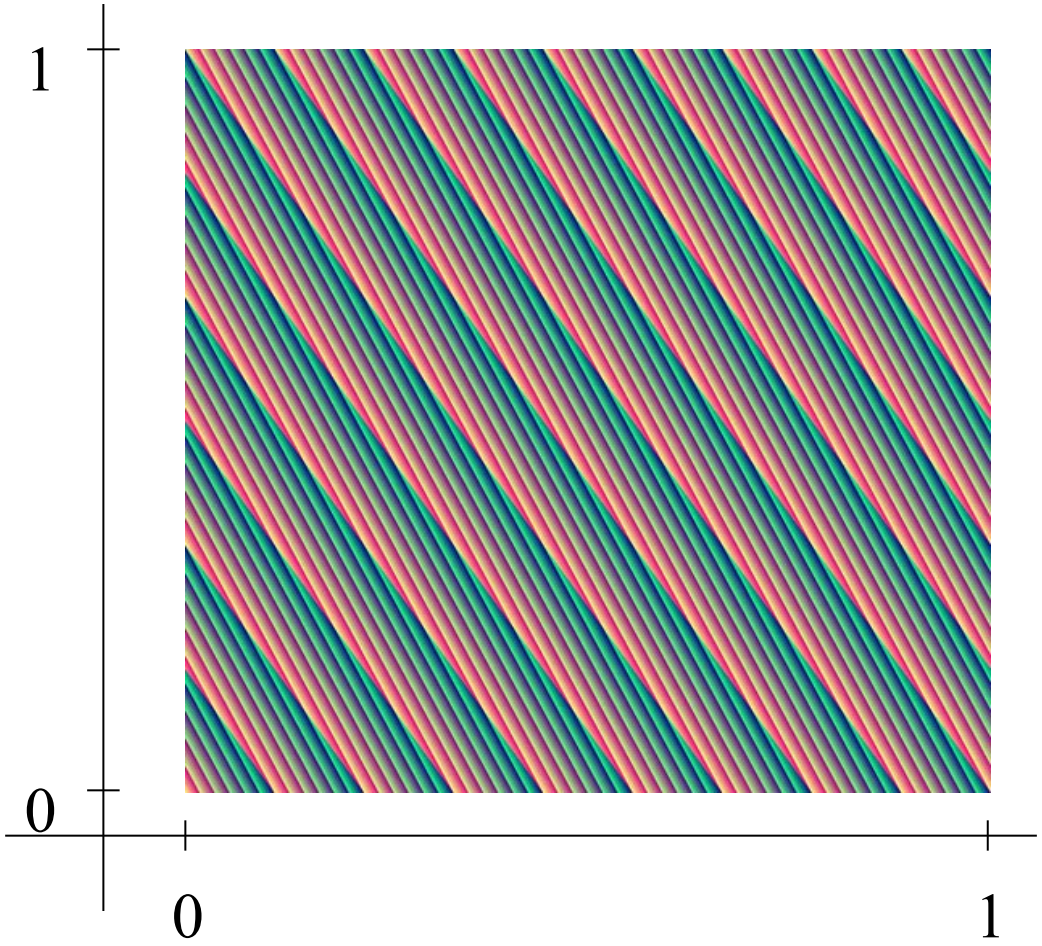
IBM RANDU

$$I_{j+1} = a\, I_j \pmod{m}$$

$$a = 65539$$

$$m = 2^{31}$$

ze dvou čísel generován bod ve čtverci [0,1] x [0,1]
barva náhodně ze stejného generátoru, například
pomocí RGB složek v rozsahu 0-255



# Lineární kongruentní generátor – barevný test

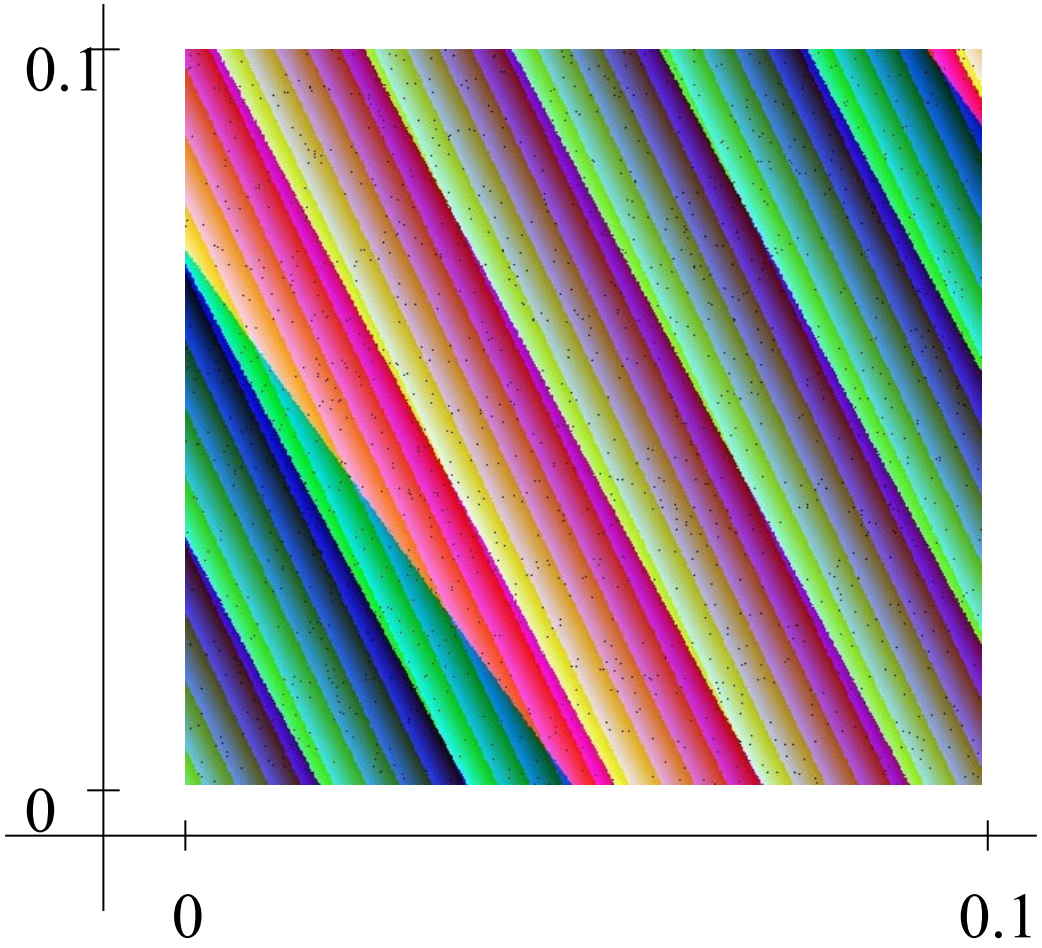
IBM RANDU

$$I_{j+1} = a\, I_j \pmod{m}$$

$$a = 65539$$

$$m = 2^{31}$$

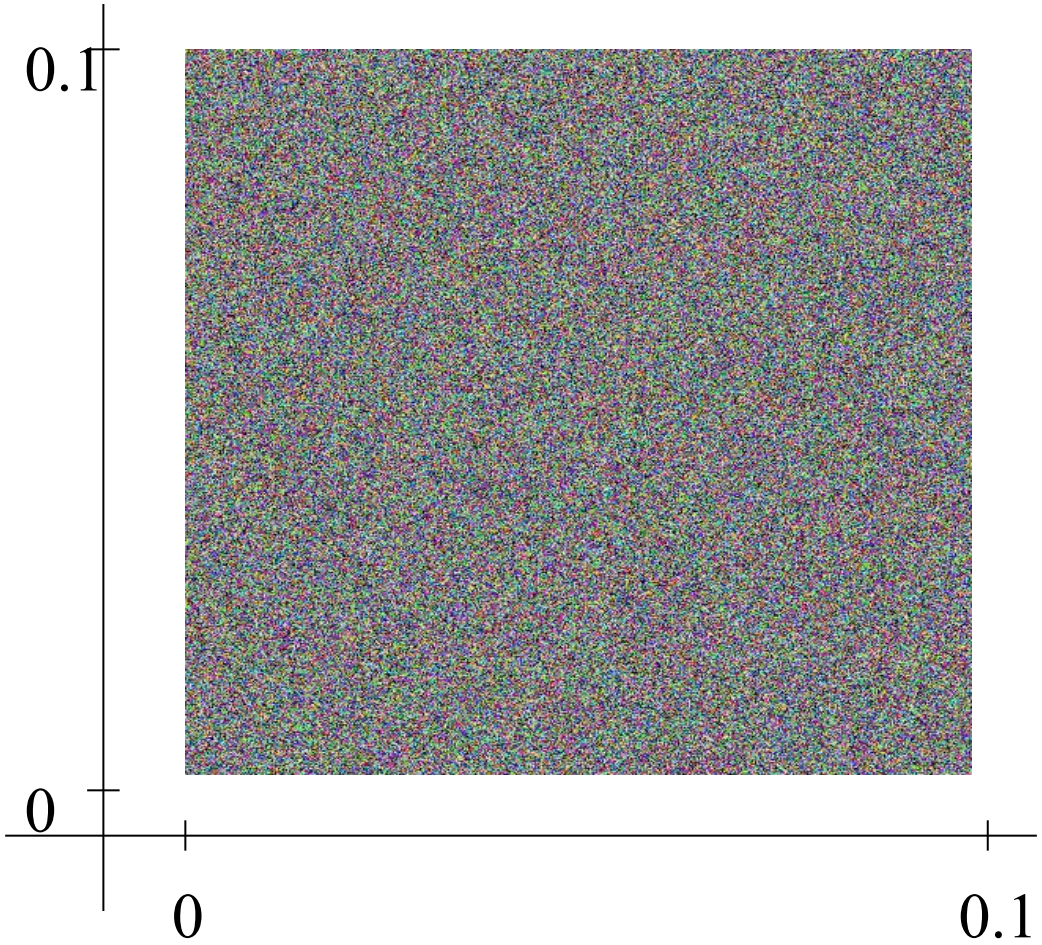
ze dvou čísel generován bod ve čtverci [0,1] x [0,1]
barva náhodně ze stejného generátoru, například
pomocí RGB složek v rozsahu 0-255



# Lineární kongruentní generátor – barevný test

RAND( )
Mersenne twister
perioda $2^{19937}$-1
MS Visual C++ 6.0

ze dvou čísel generován bod ve čtverci [0,1] x [0,1]
barva náhodně ze stejného generátoru, například
pomocí RGB složek v rozsahu 0-255



# Generátory pseudonáhodných čísel

| | Statistical Quality | Prediction Difficulty | Reproducible Results | Multiple Streams | Period | Useful Features | Time Perfomance | Space Usage | Code Size & Complexity | *k*-Dimensional Equidistribution |
|---|---|---|---|---|---|---|---|---|---|---|
| **PCG Family** | Excellent | Challenging | Yes | Yes (e.g. $2^{63}$) | Arbitrary | Jump ahead, Distance | Very fast | Very compact | Very small | Arbitrary* |
| **Mersenne Twister** | Some Failures | Easy | Yes | No | Huge $2^{19937}$-1 | Jump ahead | Acceptable | Huge (2 KB) | Complex | 623 |
| **Arc4Random** | Some Issues | Secure | Not Always | No | Huge $2^{1699}$ | No | Slow | Large (0.5 KB) | Complex | No |
| **ChaCha20**† | Good | Secure | Yes | Yes ($2^{128}$) | $2^{128}$ | Jump ahead, Distance | Fairly Slow | Plump (0.1 KB) | Complex | No |
| **Minstd (LCG)** | Many Issues | Trivial | Yes | No | Tiny $< 2^{32}$ | Jump ahead, Distance | Acceptable | Very compact | Very small | No |
| **LCG 64/32** | Many Issues | Published Algorithms | Yes | Yes $2^{63}$ | Okay $2^{64}$ | Jump ahead, Distance | Very fast | Very compact | Very small | No |
| **XorShift 32** | Many Issues | Trivial | Yes | No | Small $2^{32}$ | Jump ahead | Fast | Very compact | Very small | No |
| **XorShift 64** | Many Issues | Trivial | Yes | No | Okay $2^{64}$ | Jump ahead | Fast | Very compact | Very small | No |
| **RanQ** | Some Issues | Trivial | Yes | No | Okay $2^{64}$ | Jump ahead | Fast | Very compact | Very small | No |
| **XorShift* 64/32** | Excellent | Unknown? | Yes | No | Okay $2^{64}$ | Jump ahead | Fast | Very compact | Very small | No |

# Lineární kongruentní generátor – barevný test

PCG-64 (Permutation congruential generator)
Perioda $2^{138}$
Python

| | Statistical Quality | Prediction Difficulty | Reproducible Results | Multiple Streams | Period | Useful Features | Time Perfomance | Space Usage | Code Size & Complexity | *k*-Dimensional Equidistribution |
|---|---|---|---|---|---|---|---|---|---|---|
| **PCG Family** | Excellent | Challenging | Yes | Yes (e.g. $2^{63}$) | Arbitrary | Jump ahead, Distance | Very fast | Very compact | Very small | Arbitrary* |
| **Mersenne Twister** | Some Failures | Easy | Yes | No | Huge $2^{19937}$ | Jump ahead | Acceptable | Huge (2 KB) | Complex | 623 |
| **Arc4Random** | Some Issues | Secure | Not Always | No | Huge $2^{1699}$-1 | No | Slow | Large (0.5 KB) | Complex | No |
| **ChaCha20**† | Good | Secure | Yes | Yes ($2^{128}$) | $2^{128}$ | Jump ahead, Distance | Fairly Slow | Plump (0.1 KB) | Complex | No |
| **Minstd (LCG)** | Many Issues | Trivial | Yes | No | Tiny $< 2^{32}$ | Jump ahead, Distance | Acceptable | Very compact | Very small | No |
| **LCG 64/32** | Many Issues | Published Algorithms | Yes | Yes $2^{63}$ | Okay $2^{64}$ | Jump ahead, Distance | Very fast | Very compact | Very small | No |
| **XorShift 32** | Many Issues | Trivial | Yes | No | Small $2^{32}$ | Jump ahead | Fast | Very compact | Very small | No |
| **XorShift 64** | Many Issues | Trivial | Yes | No | Okay $2^{64}$ | Jump ahead | Fast | Very compact | Very small | No |
| **RanQ** | Some Issues | Trivial | Yes | No | Okay $2^{64}$ | Jump ahead | Fast | Very compact | Very small | No |
| **XorShift* 64/32** | Excellent | Unknown? | Yes | No | Okay $2^{64}$ | Jump ahead | Fast | Very compact | Very small | No |