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

$$I_{j+1} = aI_j \pmod{m}$$
$$a, m \in \mathbf{N}$$
$$x_j = I_j / m$$
$$0 \le x_j < 1 \quad x_j \in U(0,1)$$

- Faktorizace *m* (Schragerův algoritmus)

$$m = aq + r, \quad q = \lfloor m/a \rfloor, \; r = m \bmod a$$
$$r < q$$
$$\Downarrow$$
$$0 < I_j < m-1 \Rightarrow \begin{cases} 0 \le a(I_j \bmod q) \le m-1 \\ 0 \le r\lfloor I_j/q \rfloor \le m-1 \end{cases}$$

$$a = 7^5 = 16807$$
$$m = 2^{31} - 1 = 2147483647$$
$$q = 127773 \qquad r = 2836$$

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

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);
}
```

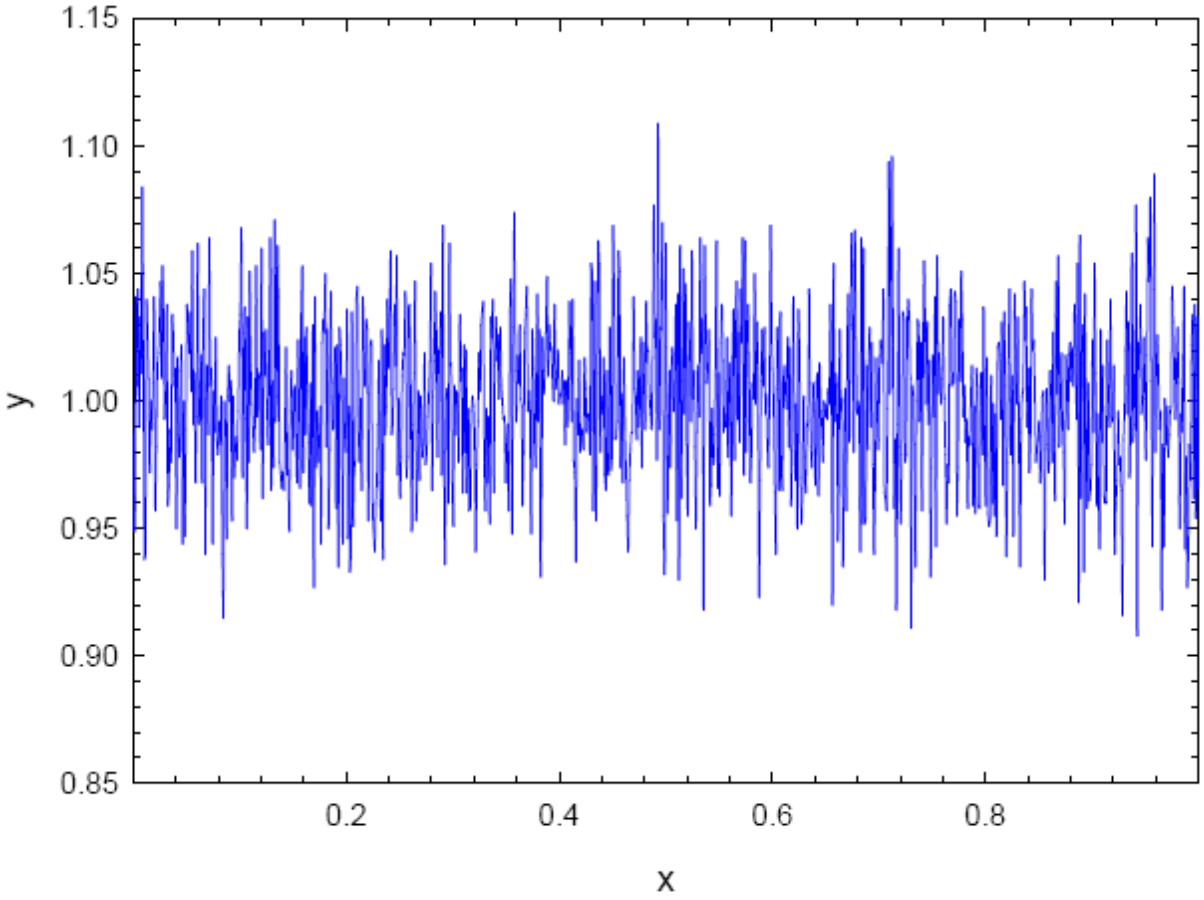
Lineární kongruentní generátor - implementace

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



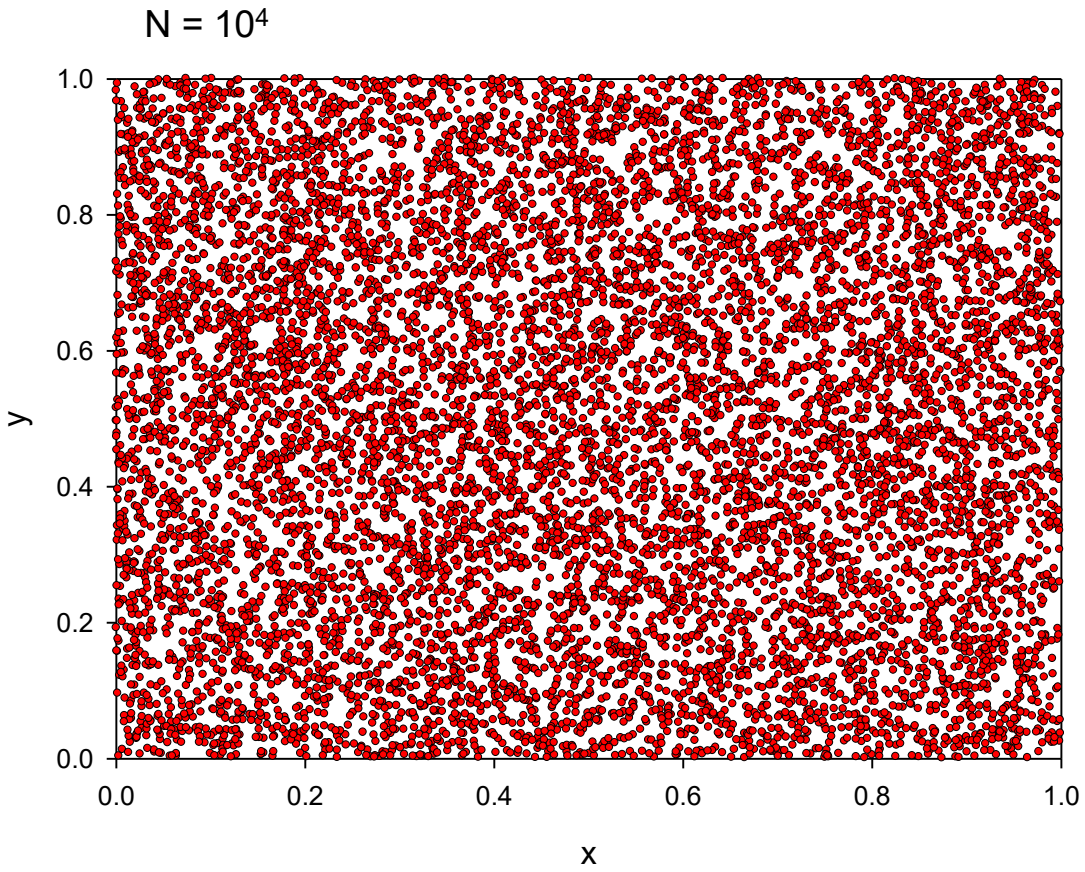
Lineární kongruentní generátor – test

IBM RANDU

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

$$a = 65539$$

$$m = 2^{31}$$



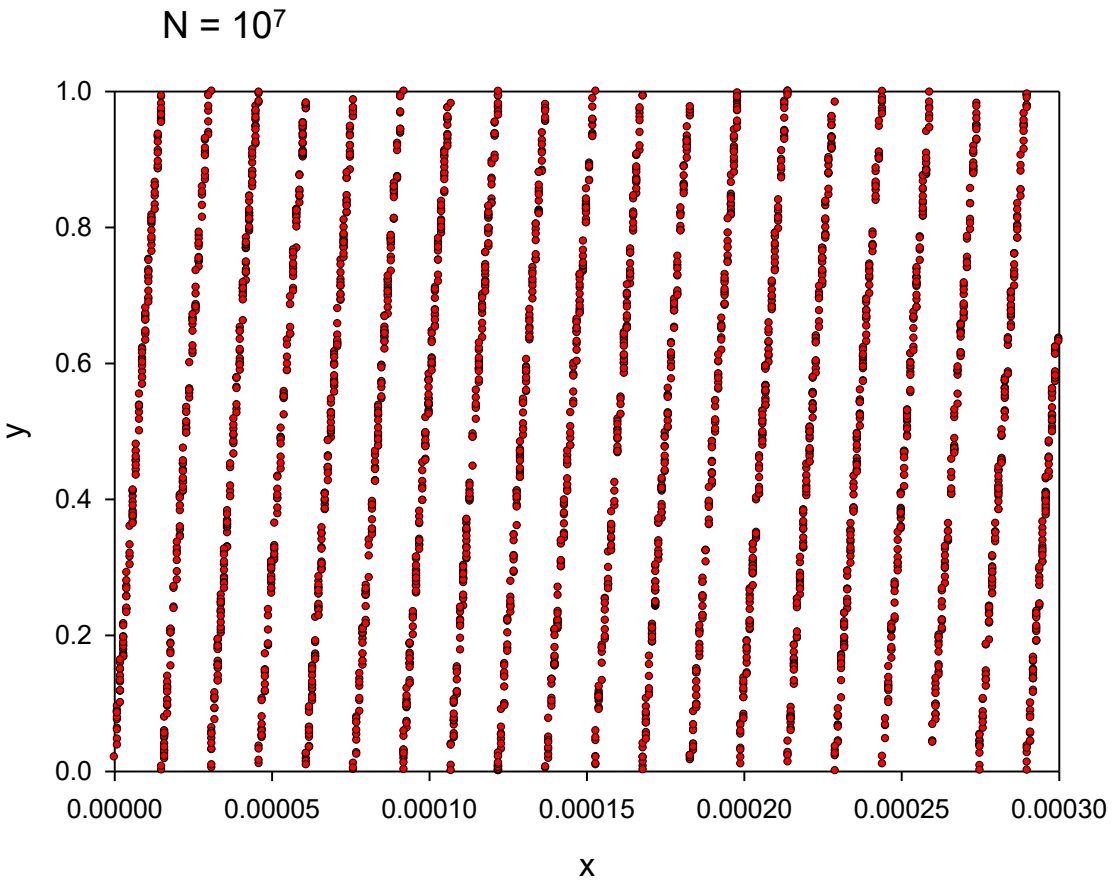
Lineární kongruentní generátor – test

IBM RANDU

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

$$a = 65539$$

$$m = 2^{31}$$



Lineární kongruentní generátor – sériová korelace

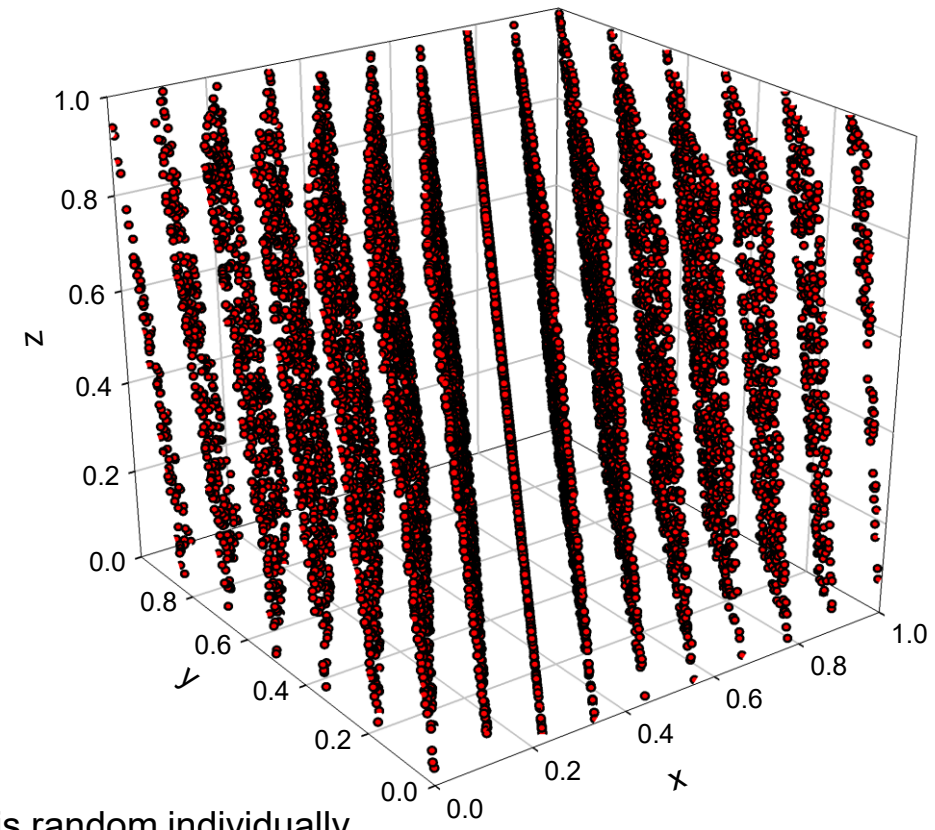
IBM RANDU

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

$$a = 65539$$

$$m = 2^{31}$$

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

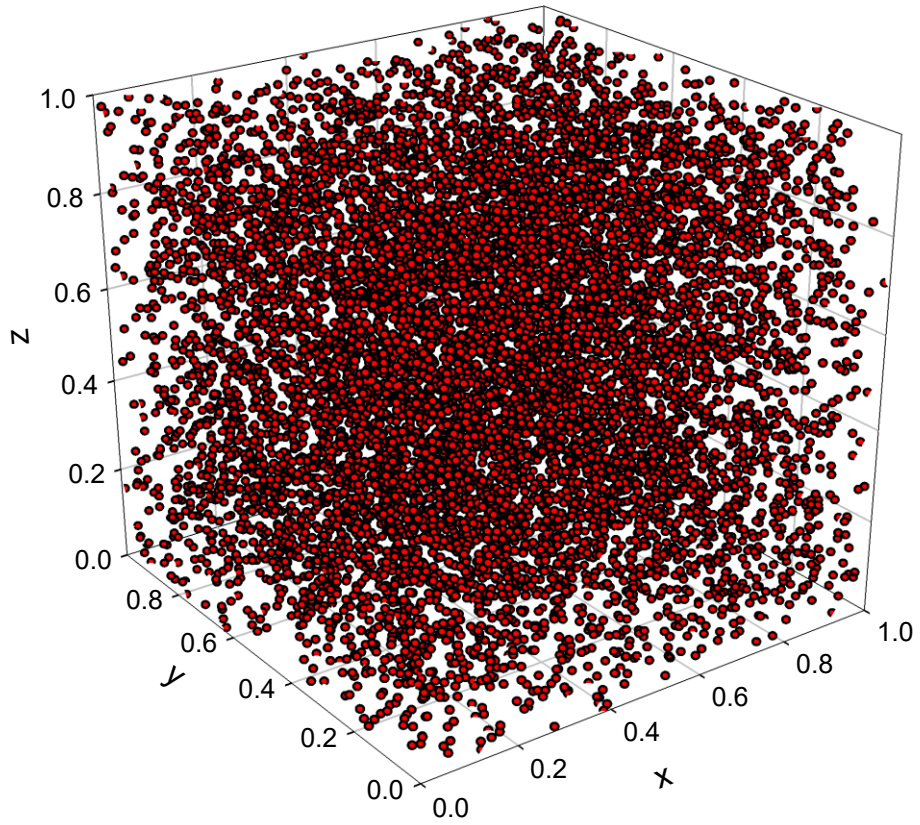
RAN0

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

$$a = 16807$$

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

N = 10^4



Lineární kongruentní generátor – posuvný registr

RAN2 – L'Ecuyer

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

$$a_1 = 40014$$

$$m_1 = 2147483563$$

$$a_2 = 40692$$

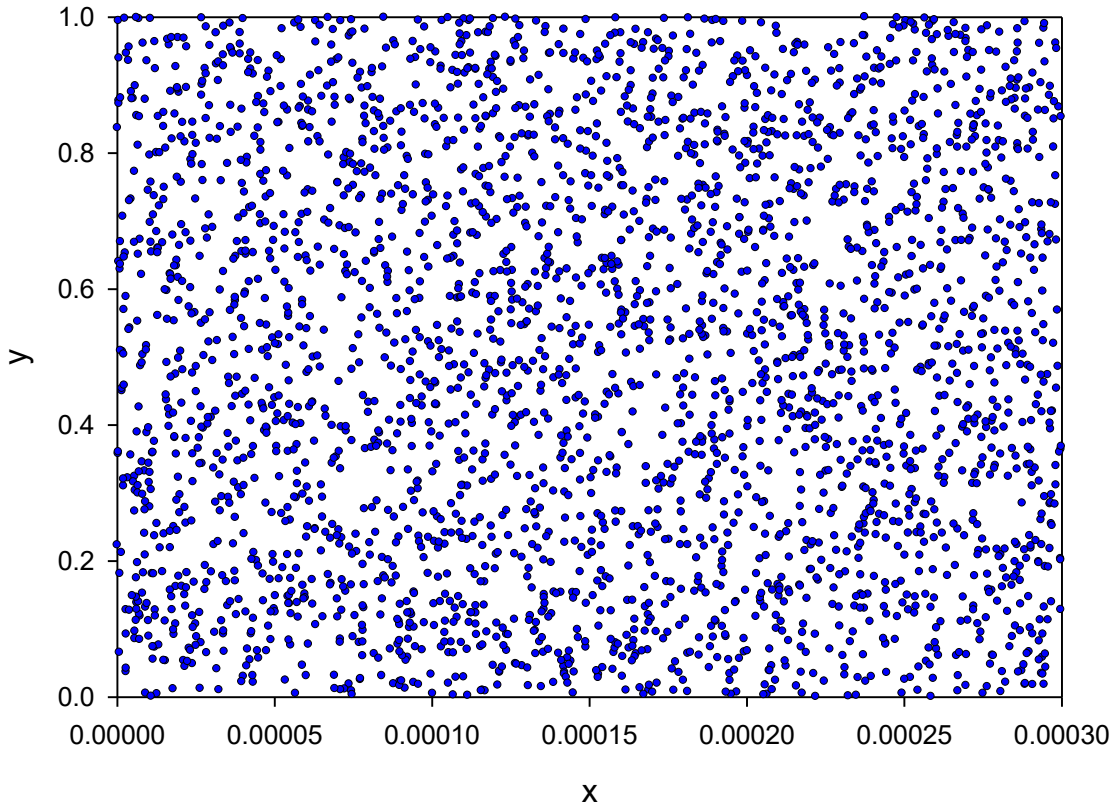
$$m_2 = 2147483399$$

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

+ posuv registru

perioda ≈ 2.3 × 10^{18}

N = 10^7



Lineární kongruentní generátor – barevný test

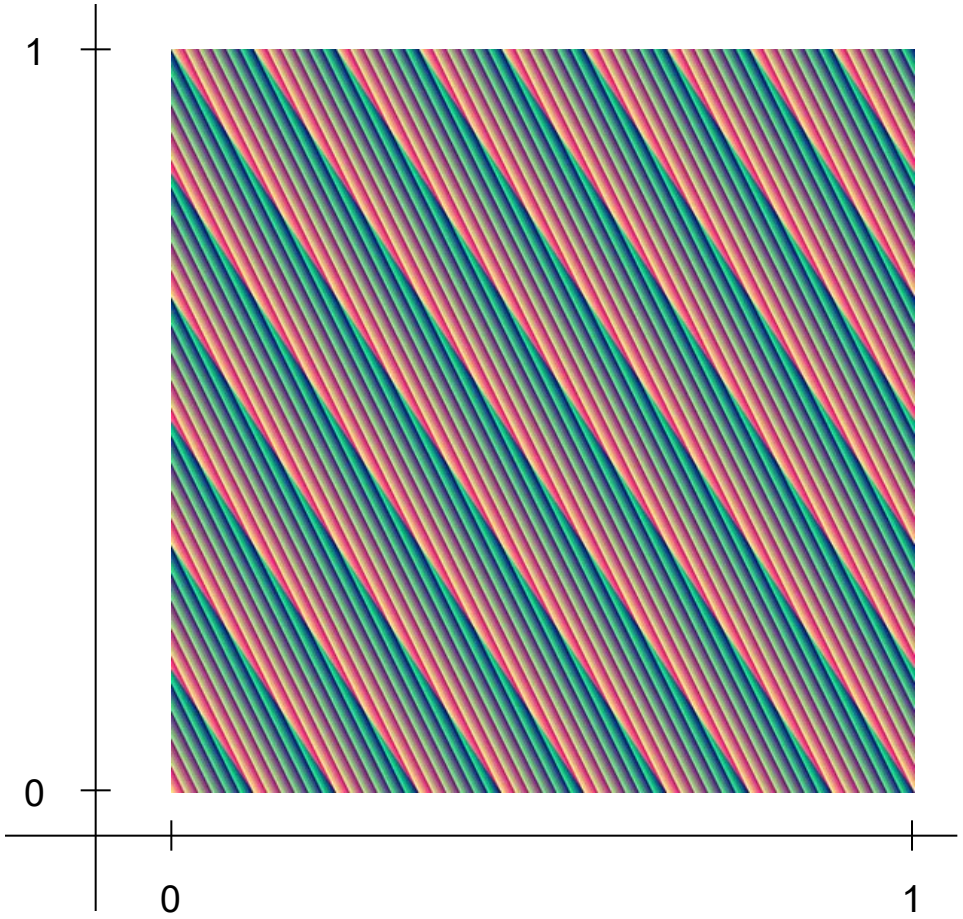
IBM RANDU

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

$$a = 65539$$

$$m = 2^{31}$$

- 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

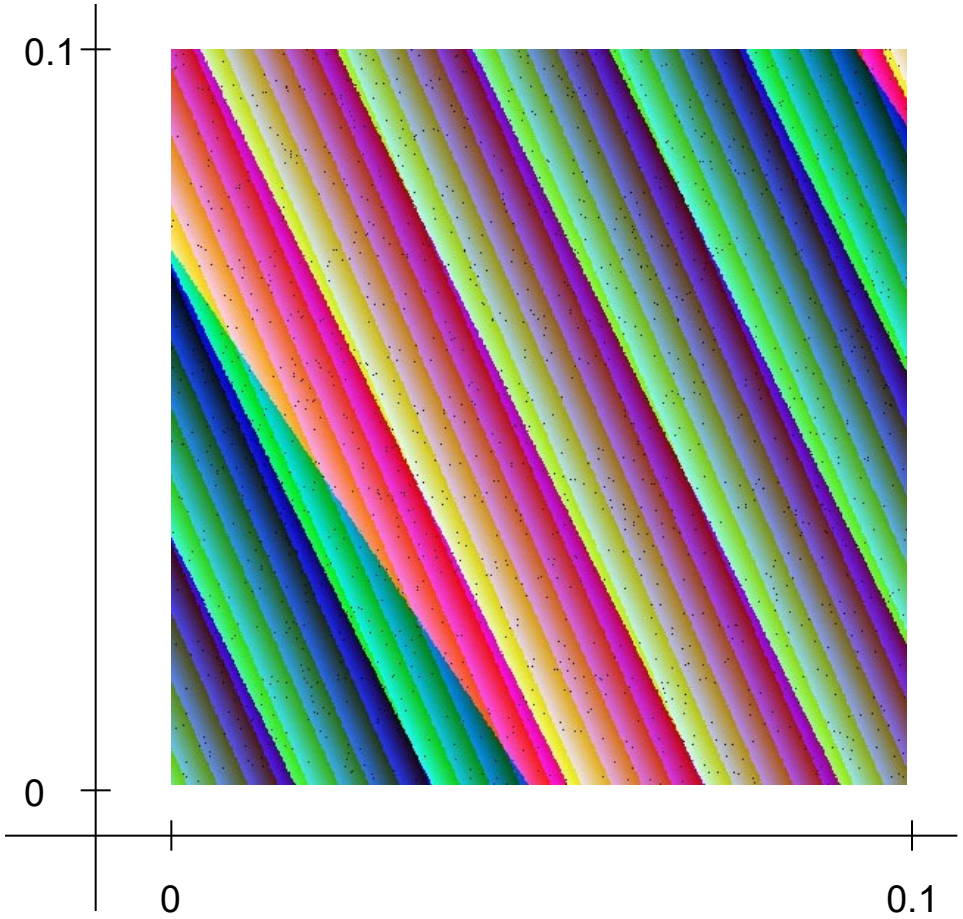
IBM RANDU

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

$$a = 65539$$

$$m = 2^{31}$$

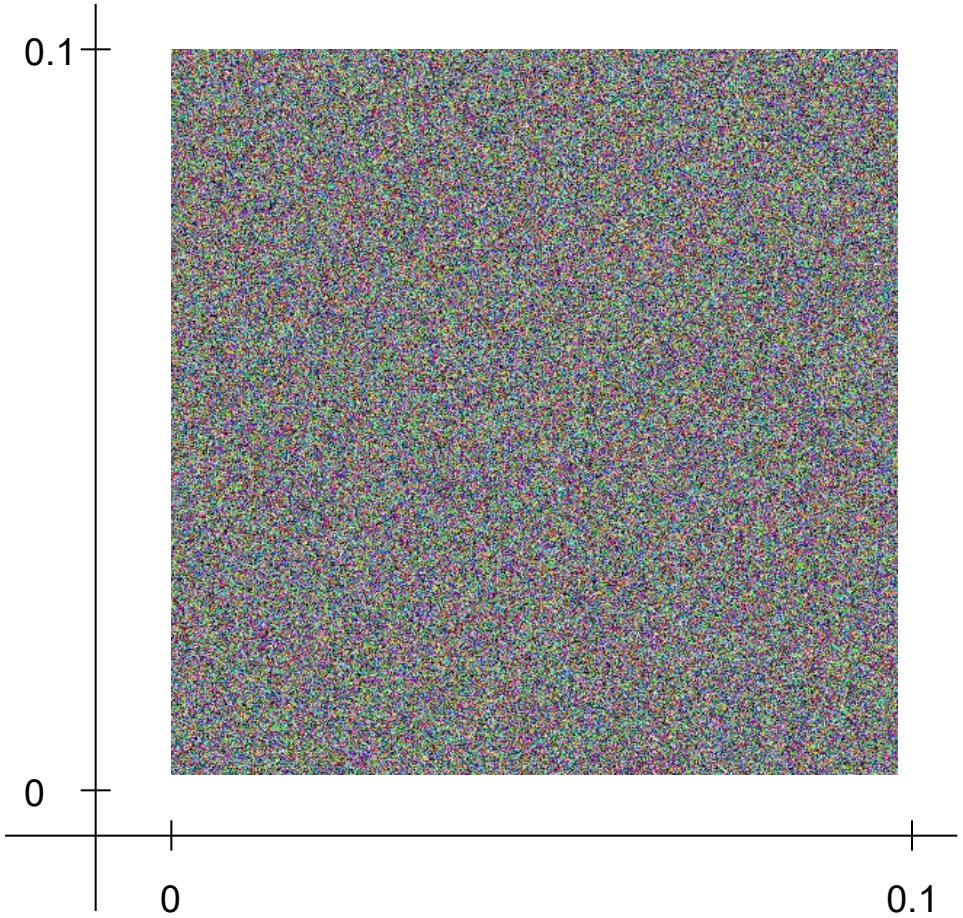
- ze dvou čísel generován bod ve čtverci [0,1] x [0,1]
- barva náhodně ze stejného generátoru, například pomocí RGB složek v rozsahu 0-255



Lineární kongruentní generátor – barevný test

RAND( )
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

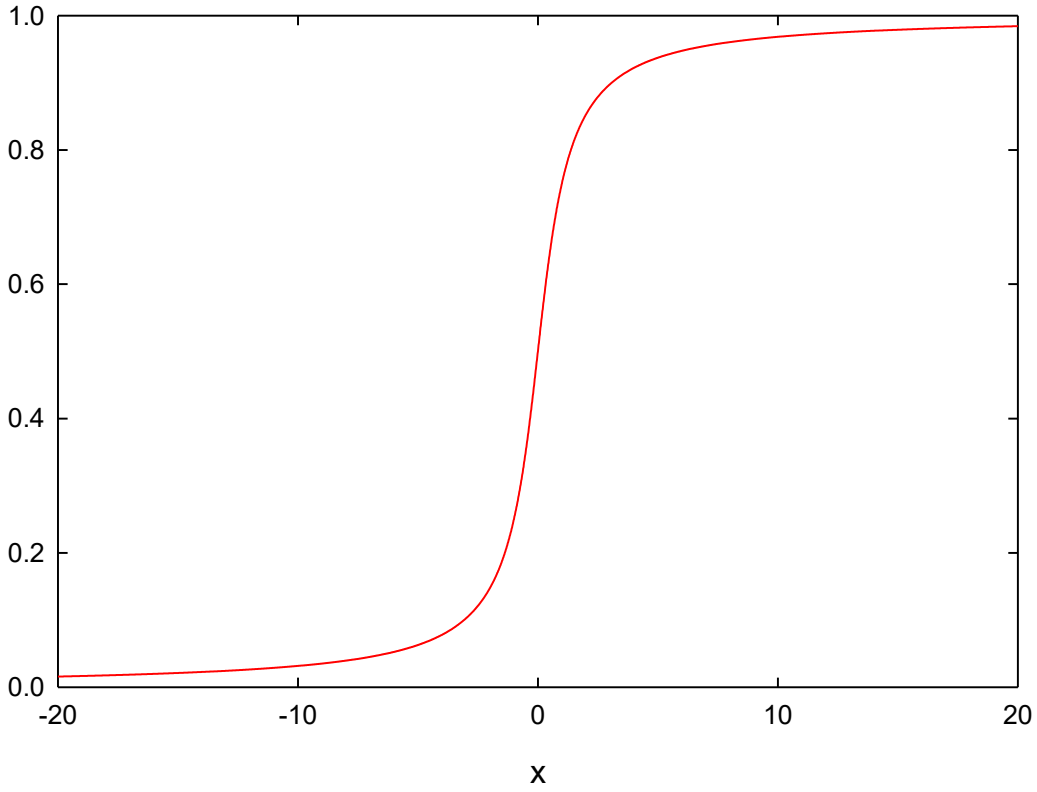


Cauchyho rozdělení – Metoda inverzní funkce

Distribuční funkce

$$F(x) = \frac{1}{\pi}\left(\operatorname{arctg} x + \frac{\pi}{2}\right)$$

$$x = \operatorname{tg}\left[\pi\left(t - \frac{1}{2}\right)\right] \qquad t \in U(0,1)$$

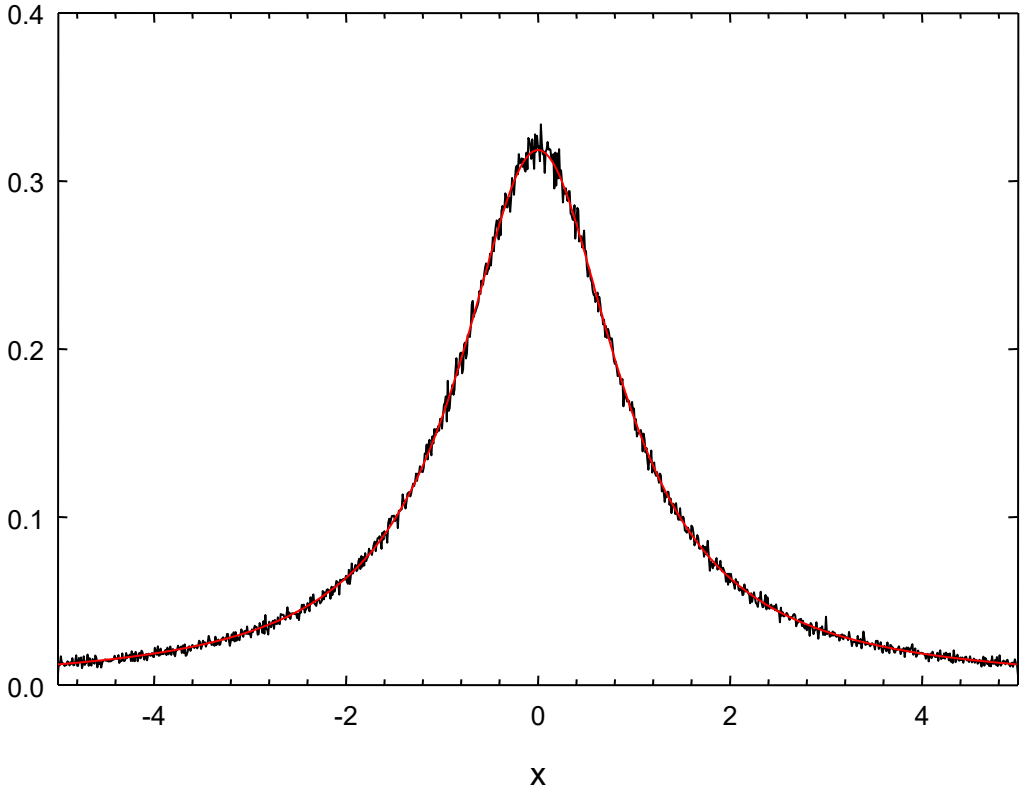


Cauchyho rozdělení – Metoda inverzní funkce

Cauchyho rozdělení

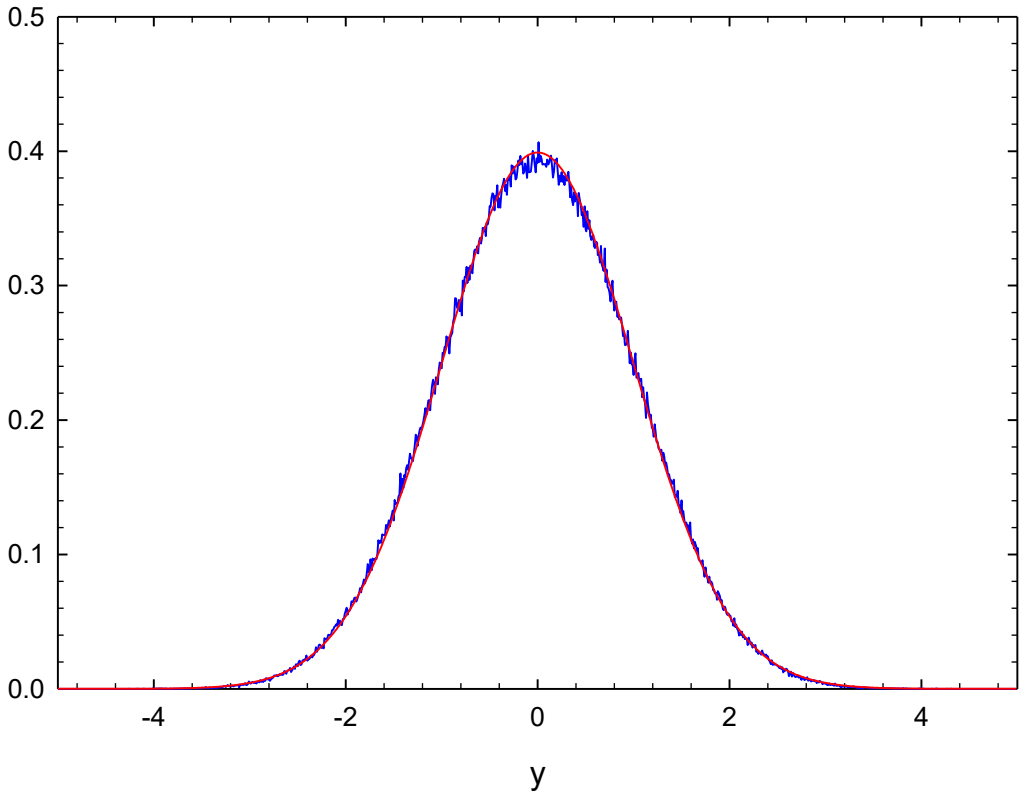
$$f(x) = \frac{1}{\pi\left(1 + x^2\right)}$$

$$x = \mathrm{tg}\left[\pi\left(\mathrm{t} - \frac{1}{2}\right)\right] \qquad t \in U(0,1)$$



Normální rozdělení – využití CLT

$$y = \sum_{i=1}^{12} x_i - 6 \qquad x_i \in U(0,1) \qquad \text{CLT} \Rightarrow \quad y \approx N(0,1)$$



Normální rozdělení – ad hoc generátor

$$y_1 = \sqrt{-2\ln x_1}\cos 2\pi x_2$$

$$y_2 = \sqrt{-2\ln x_1}\sin 2\pi x_2$$

$$x_{1,2} \in U(0,1)$$

$$x_1 = \exp\left(-\frac{y_1^2 + y_2^2}{2}\right)$$

$$x_2 = \frac{1}{2\pi}\operatorname{arctg}\left(\frac{y_2}{y_1}\right)$$

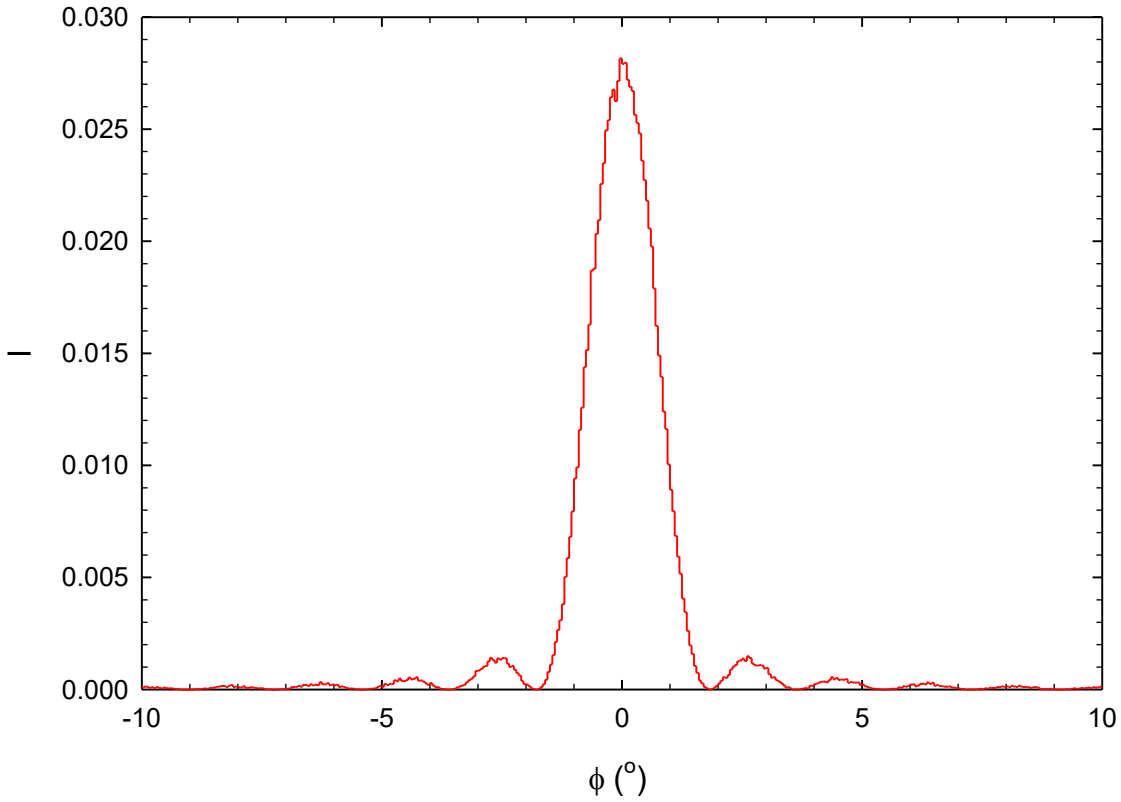
$$g(y_1, y_2) = \frac{1}{\sqrt{2\pi}} e^{-\frac{y_1^2}{2}} \frac{1}{\sqrt{2\pi}} e^{-\frac{y_2^2}{2}}$$

$$y_{1,2} \in N(0,1)$$

Von Neumannova zamítací metoda

$$f(\phi) = I_0 \left(\frac{\sin u}{u} \right)^2$$

$$u = \frac{\pi}{\lambda} a \phi$$



Monte Carlo integrace

π = 3.141592654