Simulações Computacionais de Sistemas Complexos

Thadeu Penna

Instituto de Física
Universidade Federal Fluminense
tjpp@if.uff.br

Escola da UFMT,2005

Universidade Federal Fluminense

Universidade Federal Fluminense

Universidade Federal Fluminense

Universidade Federal Fluminense

Sistemas Complexos

Sistemas Complexos

- Sistema com um grande número de constituintes ou partes, que interagem entre si.

Sistemas Complexos

Sistemas Complexos

- Sistema com um grande número de constituintes ou partes, que interagem entre si.
- Aparece um comportamento coletivo complexo.

Sistemas Complexos

Sistemas Complexos

- Sistema com um grande número de constituintes ou partes, que interagem entre si.
- Aparece um comportamento coletivo complexo.
- As interações são conflitantes mas não são complicadas.

Sistemas Complexos

Sistemas Complexos

- Sistema com um grande número de constituintes ou partes, que interagem entre si.
- Aparece um comportamento coletivo complexo.
- As interações são conflitantes mas não são complicadas.
- O comportamento do todo é diferente da soma das partes.

Universidade Federal Fluminense

Simulações

Sistemas Booleanos

Propósito: simplificação

Sistemas Booleanos

Propósito: simplificação
Adequado para implementação em computadores digitais

Sistemas Booleanos

Propósito: simplificação
Adequado para implementação em computadores digitais
Fácil paralelização

Universidade Federal Fluminense

Sistemas Booleanos

Propósito: simplificação
Adequado para implementação em computadores digitais
Fácil paralelização

Boole, George (1815-1864)

Sistemas Booleanos

Propósito: simplificação
Adequado para implementação em computadores digitais
Fácil paralelização

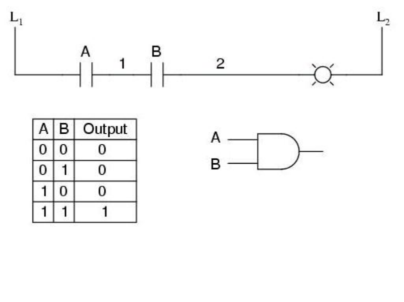
Boole, George (1815-1864)

- Criador da Lógica Matemática
- The Mathematical Analysis of Logic (1847)
- Operações Básicas: AND $\bigwedge$, OR $\bigvee$ e NOT !
- particular: álgebra de ordem 2 (0 e 1)

Universidade Federal Fluminense

Operações Booleanas

Operação AND $\bigwedge$

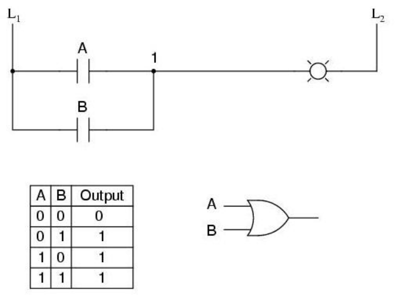


A	B	Output
0	0	0
0	1	0
1	0	0
1	1	1

Universidade Federal Fluminense

Operações Booleanas

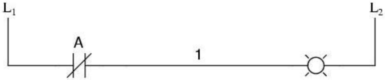
Operação OR $\bigvee$



A	B	Output
0	0	0
0	1	1
1	0	1
1	1	1

Operações Booleanas

Operação NOT !



A	Output
0	1
1	0

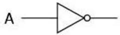


Tabela Verdade

Operações Booleanas

	AND	OR	XOR
00	0	0	0
01	0	1	1
10	0	1	1
11	1	1	0

Universidade Federal Fluminense

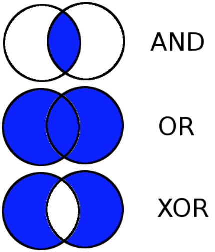
Tabela Verdade

Operações Booleanas

	AND	OR	XOR
00	0	0	0
01	0	1	1
10	0	1	1
11	1	1	0

P: Quantas operações binárias existem ?

Diagrama de Conjuntos



Propriedades

Propriedades

Idempotente:

$$a \vee a = a \wedge a = a$$

Propriedades

Idempotente:

$$a \vee a = a \wedge a = a$$

Comutativa:

$$\begin{aligned} a \vee b &= b \vee a \\ a \wedge b &= b \wedge a \end{aligned}$$

Propriedades

Idempotente:

$$a \vee a = a \wedge a = a$$

Comutativa:

$$\begin{aligned} a \vee b &= b \vee a \\ a \wedge b &= b \wedge a \end{aligned}$$

Associativa:

$$\begin{aligned} a \vee (b \vee c) &= (a \vee b) \vee c \\ a \wedge (b \wedge c) &= (a \wedge b) \wedge c \end{aligned}$$

Universidade Federal Fluminense

Problema 1

Escreva

- AND em termos de OR e NOT
- OR em termos de NOT e AND
- AND em termos de NAND (NOT AND)
- XOR em termos de AND, OR e NOT
- XOR em termos de NAND e NOT
- XNOR (NOT XOR) em termos de AND, OR e NOT

Bits & Bytes

Os computadores lidam com números inteiros e de ponto flutuante.

Bits & Bytes

Os computadores lidam com números inteiros e de ponto flutuante.
Operações com inteiros são mais eficientes: não tem expoentes.

Universidade Federal Fluminense

Bits & Bytes

Os computadores lidam com números inteiros e de ponto flutuante.
Operações com inteiros são mais eficientes: não tem expoentes.
Números inteiros são restritos a uma faixa ($[-2^{31}, 2^{31}-1]$ para 32 bits).

Universidade Federal Fluminense

Bits & Bytes

Os computadores lidam com números inteiros e de ponto flutuante.

Operações com inteiros são mais eficientes: não tem expoentes.

Números inteiros são restritos a uma faixa ($[-2^{31}, 2^{31} - 1]$ para 32 bits).

Operações de aritmética devem ser escritas em termos das funções anteriores.

Bits & Bytes

- Os computadores lidam com números inteiros e de ponto flutuante.
- Operações com inteiros são mais eficientes: não tem expoentes.
- Números inteiros são restritos a uma faixa ($[-2^{31}, 2^{31}-1]$ para 32 bits).
- Operações de aritmética devem ser escritas em termos das funções anteriores.
- C possui `unsigned long int`.

Representação dos inteiros

Números de 3 bits

$$
\begin{aligned}
0 &= 000_2 \equiv 0 \\
1 &= 001_2 \equiv 1 \\
2 &= 010_2 \equiv 2 \\
3 &= 011_2 \equiv 3 \\
4 &= 100_2 \equiv -4 \\
5 &= 101_2 \equiv -3 \\
6 &= 110_2 \equiv -2 \\
7 &= 111_2 \equiv -1
\end{aligned}
$$

Representação dos inteiros

Números de 3 bits

$$
\begin{aligned}
0 &= 000_2 \equiv 0 \\
1 &= 001_2 \equiv 1 \\
2 &= 010_2 \equiv 2 \\
3 &= 011_2 \equiv 3 \\
4 &= 100_2 \equiv -4 \\
5 &= 101_2 \equiv -3 \\
6 &= 110_2 \equiv -2 \\
7 &= 111_2 \equiv -1
\end{aligned}
$$

Complemento de 2 ($-Y = 2^B - Y$)

Operações Básicas

Soma

$$\begin{array}{r} + \\ \hline \end{array} \equiv \begin{array}{r} + \\ \hline \end{array}$$

Operações Básicas

Soma

$$\begin{array}{r} 6 \\ + \quad \\ \hline \end{array} \equiv \begin{array}{r} 110_2 \\ + \quad \\ \hline \end{array}$$

Universidade Federal Fluminense

Operações Básicas

Soma

$$\begin{array}{rr} & 6 \\ + & 3 \\ \hline \end{array} \equiv \begin{array}{rr} & 110_2 \\ + & 011_2 \\ \hline \end{array}$$

Universidade Federal Fluminense

Operações Básicas

Soma

$$\begin{array}{rr} & 6 \\ + & 3 \\ \hline & \end{array} \equiv \begin{array}{rr} & 110_2 \\ + & 011_2 \\ \hline & 001_2 \end{array}$$

Universidade Federal Fluminense

Operações Básicas

Soma

$$\begin{array}{rr} & 6 \\ + & 3 \\ \hline & 1 \end{array} \equiv \begin{array}{rr} & 110_2 \\ + & 011_2 \\ \hline & 001_2 \end{array}$$

Operações Básicas

Soma

$$\begin{array}{rr} & 6 \\ + & 3 \\ \hline & 1 \end{array} \equiv \begin{array}{rr} & 110_2 \\ + & 011_2 \\ \hline & 001_2 \end{array}$$

Overflow!!! Carry !!!

Universidade Federal Fluminense

Operações Básicas

Multiplicação por 2^n

$$\begin{array}{c} \quad + \quad \\ \hline \end{array} \equiv \begin{array}{c} \quad + \quad \\ \hline \end{array}$$

Universidade Federal Fluminense

Operações Básicas

Multiplicação por 2^n

$$\begin{array}{r} 3 \\ + \\ \hline \end{array} \equiv \begin{array}{r} 011_2 \\ + \\ \hline \end{array}$$

Operações Básicas

Multiplicação por 2^n

$$\begin{array}{rr} & 3 \\ + & 3 \\ \hline \end{array} \equiv \begin{array}{rr} & 011_2 \\ + & 011_2 \\ \hline \end{array}$$

Universidade Federal Fluminense

Operações Básicas

Multiplicação por 2^n

$$\begin{array}{rr} & 3 \\ + & 3 \\ \hline \end{array} \equiv \begin{array}{rr} & 011_2 \\ + & 011_2 \\ \hline & 110_2 \end{array}$$

Universidade Federal Fluminense

Operações Básicas

Multiplicação por 2^n

$$\begin{array}{rr} & 3 \\ + & 3 \\ \hline & 6 \end{array} \equiv \begin{array}{rr} & 011_2 \\ + & 011_2 \\ \hline & 110_2 \end{array}$$

Universidade Federal Fluminense

Operações Básicas

Multiplicação por 2^n

$$\begin{array}{rr} & 3 \\ + & 3 \\ \hline & 6 \end{array} \equiv \begin{array}{rr} & 011_2 \\ + & 011_2 \\ \hline & 110_2 \end{array}$$

$\times e \div$ por potências de 2 correspondem a deslocamentos (“shifts”)

Universidade Federal Fluminense

Operações Básicas

Multiplicação por 2^n

$$\begin{array}{rr} & 3 \\ + & 3 \\ \hline & 6 \end{array} \equiv \begin{array}{rr} & 011_2 \\ + & 011_2 \\ \hline & 110_2 \end{array}$$

$\times$ e $\div$ por potências de 2 correspondem a deslocamentos ("shifts")

Também podem levar a overflow

Operações Básicas

Multiplicação por 2^n

$$\begin{array}{rr} & 3 \\ + & 3 \\ \hline & 6 \end{array} \equiv \begin{array}{rr} & 011_2 \\ + & 011_2 \\ \hline & 110_2 \end{array}$$

$\times e \div$ por potências de 2 correspondem a deslocamentos ("shifts")

Também podem levar a overflow

Note que esta representação funciona para a multiplicação

Operações Básicas

Multiplicação por 2^n

$$\begin{array}{rr} & 3 \\ + & 3 \\ \hline & 6 \end{array} \equiv \begin{array}{rr} & 011_2 \\ + & 011_2 \\ \hline & 110_2 \end{array}$$

$\times e \div$ por potências de 2 correspondem a deslocamentos ("shifts")

Também podem levar a overflow

Note que esta representação funciona para a multiplicação

É imediato a adaptação para mais bits ;)

Universidade Federal Fluminense

Problema 2

Escreva um programa para

a soma de dois números inteiros, só utilizando as funções binárias

FORTRAN

```
integer*4 a,b
write(*,*) iand(a,b)
write(*,*) ior(a,b)
write(*,*) ieor(a,b)
write(*,*) ishft(a,1)
write(*,*) ishft(a,-1)
write(*,*) not(a)
```

C

```
unsigned int a,b;
printf(``%d\n'',a&b);
printf(``%d\n'',a|b);
printf(``%d\n'',a^b);
printf(``%d\n'',a<<1);
printf(``%d\n'',a>>1);
printf(``%d\n'',~a);
```

Universidade Federal Fluminense

Potências de 2

Clique aqui

Potências de 2

Clique aqui

Tente descobrir 2^n-1

Potências de 2

Clique aqui

Tente descobrir 2^n-1

Tente descobrir o i^o bit

Potências de 2

Clique aqui

Tente descobrir 2^n-1

Tente descobrir o i^o bit

Como setar um bit ?

Potências de 2

Clique aqui

Tente descobrir 2^n-1

Tente descobrir o i^o bit

Como setar um bit ?

Como resetar um bit ?

Potências de 2

Clique aqui

Tente descobrir 2^n-1

Tente descobrir o i^o bit

Como setar um bit ?

Como resetar um bit ?

Como inverter um bit ?

Universidade Federal Fluminense

Potências de 2

Clique aqui

Tente descobrir 2^n-1

Tente descobrir o i^o bit

Como setar um bit ?

Como resetar um bit ?

Como inverter um bit ?

Como fazer um shift circular ?

Mini Vocabulário

- Elementos, partes, componentes,

Mini Vocabulário

- Elementos, partes, componentes,
- Interações, conexões, relacionamentos, redes,

Mini Vocabulário

- Elementos, partes, componentes,
- Interações, conexões, relacionamentos, redes,
- Emergência - não linearidade, escalas,

Universidade Federal Fluminense

Mini Vocabulário

- Elementos, partes, componentes,
- Interações, conexões, relacionamentos, redes,
- Emergência - não linearidade, escalas,
- Padrões, descrição, informação,

Mini Vocabulário

- Elementos, partes, componentes,
- Interações, conexões, relacionamentos, redes,
- Emergência - não linearidade, escalas,
- Padrões, descrição, informação,
- Dinâmica, resposta, feedback, homeostase, controle

Mini Vocabulário

- Elementos, partes, componentes,
- Interações, conexões, relacionamentos, redes,
- Emergência - não linearidade, escalas,
- Padrões, descrição, informação,
- Dinâmica, resposta, feedback, homeostase, controle
- Adaptação, evolução, organização, auto-organização,

Universidade Federal Fluminense

Mini Vocabulário

- Elementos, partes, componentes,
- Interações, conexões, relacionamentos, redes,
- Emergência - não linearidade, escalas,
- Padrões, descrição, informação,
- Dinâmica, resposta, feedback, homeostase, controle
- Adaptação, evolução, organização, auto-organização,
- Conflito, satisfação, frustração, otimização

Universidade Federal Fluminense

Mini Vocabulário

- Elementos, partes, componentes,
- Interações, conexões, relacionamentos, redes,
- Emergência - não linearidade, escalas,
- Padrões, descrição, informação,
- Dinâmica, resposta, feedback, homeostase, controle
- Adaptação, evolução, organização, auto-organização,
- Conflito, satisfação, frustração, otimização
- História, diversidade, sensibilidade às condições iniciais

Universidade Federal Fluminense

Mini Vocabulário

- Elementos, partes, componentes,
- Interações, conexões, relacionamentos, redes,
- Emergência - não linearidade, escalas,
- Padrões, descrição, informação,
- Dinâmica, resposta, feedback, homeostase, controle
- Adaptação, evolução, organização, auto-organização,
- Conflito, satisfação, frustração, otimização
- História, diversidade, sensibilidade às condições iniciais
- Complexidade, descrição do sistema, entropia

Como Estudar Sistemas Complexos ?

- Macroscópico $\longleftrightarrow$ Microscópico

Universidade Federal Fluminense

Como Estudar Sistemas Complexos ?

- Macroscópico $\longleftrightarrow$ Microscópico
- Termodinâmica $\longleftrightarrow$ Mecânica Estatística

Universidade Federal Fluminense

Como Estudar Sistemas Complexos ?

- Macroscópico $\longleftrightarrow$ Microscópico
- Termodinâmica $\longleftrightarrow$ Mecânica Estatística
- Escalas, frustração $\longleftrightarrow$ Transições de Fase

Universidade Federal Fluminense

Como Estudar Sistemas Complexos ?

- Macroscópico $\longleftrightarrow$ Microscópico
- Termodinâmica $\longleftrightarrow$ Mecânica Estatística
- Escalas, frustração $\longleftrightarrow$ Transições de Fase
- Interações assimétricas, metaestabilidade, flutuações

Universidade Federal Fluminense

Como Estudar Sistemas Complexos ?

- Macroscópico $\longleftrightarrow$ Microscópico
- Termodinâmica $\longleftrightarrow$ Mecânica Estatística
- Escalas, frustração $\longleftrightarrow$ Transições de Fase
- Interações assimétricas, metaestabilidade, flutuações
- Simulações Computacionais eficientes

Como Estudar Sistemas Complexos ?

- Macroscópico $\longleftrightarrow$ Microscópico
- Termodinâmica $\longleftrightarrow$ Mecânica Estatística
- Escalas, frustração $\longleftrightarrow$ Transições de Fase
- Interações assimétricas, metaestabilidade, flutuações
- Simulações Computacionais eficientes
- Monte Carlo, automata celulares, simulated annealing, amostragem entrópica

Como Estudar Sistemas Complexos ?

- Macroscópico $\longleftrightarrow$ Microscópico
- Termodinâmica $\longleftrightarrow$ Mecânica Estatística
- Escalas, frustração $\longleftrightarrow$ Transições de Fase
- Interações assimétricas, metaestabilidade, flutuações
- Simulações Computacionais eficientes
- Monte Carlo, automata celulares, simulated annealing, amostragem entrópica
- Tamanho finito, multigrid

Universidade Federal Fluminense

Características

- "Importante demais para deixar ao acaso"

Características

- “Importante demais para deixar ao acaso”
- ”Anyone who considers arithmetical methods of producing random digits is , of course, in a state of sin.” J. von Neumann

Características

- “Importante demais para deixar ao acaso”
- ”Anyone who considers arithmetical methods of producing random digits is , of course, in a state of sin.” J. von Neumann
- Random number generators: good ones are hard to find. [Commun. ACM, 31, 1192–1201]

Características

- "Importante demais para deixar ao acaso"
- "Anyone who considers arithmetical methods of producing random digits is , of course, in a state of sin." J. von Neumann
- Random number generators: good ones are hard to find. [Commun. ACM, 31, 1192–1201]
- Números aleatórios gerados a partir de operações matemáticas, portanto determinísticos.

Características

- “Importante demais para deixar ao acaso”
- "Anyone who considers arithmetical methods of producing random digits is , of course, in a state of sin." J. von Neumann
- Random number generators: good ones are hard to find. [Commun. ACM, 31, 1192–1201]
- Números aleatórios gerados a partir de operações matemáticas, portanto determinísticos.
- Grande período, baixa correlação e velocidade.

Características

- "Importante demais para deixar ao acaso"
- "Anyone who considers arithmetical methods of producing random digits is , of course, in a state of sin." J. von Neumann
- Random number generators: good ones are hard to find. [Commun. ACM, 31, 1192–1201]
- Números aleatórios gerados a partir de operações matemáticas, portanto determinísticos.
- Grande período, baixa correlação e velocidade.
- Geradores diferentes falham em testes diferentes.

Características

- "Importante demais para deixar ao acaso"
- "Anyone who considers arithmetical methods of producing random digits is , of course, in a state of sin." J. von Neumann
- Random number generators: good ones are hard to find. [Commun. ACM, 31, 1192–1201]
- Números aleatórios gerados a partir de operações matemáticas, portanto determinísticos.
- Grande período, baixa correlação e velocidade.
- Geradores diferentes falham em testes diferentes.
- O melhor gerador depende do problema em questão

LCG

Universidade Federal Fluminense

LCG

Da forma

$$x(n) = (a * x(n-1) + b) \mod M$$

LCG

Da forma

$$x(n) = (a * x(n-1) + b) \mod M$$

com valores especiais para a e M

LCG

Da forma

$$x(n) = (a * x(n-1) + b) \mod M$$

com valores especiais para a e M

Exemplos: $a = 16807$ (Park e Muller),65539 (IBM RANDU), $69621, 1103515245$ e $M = 2^{31} - 1$

LCG

Da forma

$$x(n) = (a * x(n-1) + b) \mod M$$

com valores especiais para a e M

Exemplos: $a = 16807$ (Park e Muller),65539 (IBM RANDU), $69621, 1103515245$ e $M = 2^{31} - 1$

Para 64 bits $a = 13^{13}$,44485709377909

LCG

Da forma

$$x(n) = (a * x(n-1) + b) \mod M$$

com valores especiais para a e M

Exemplos: $a = 16807$ (Park e Muller),65539 (IBM RANDU), $69621, 1103515245$ e $M = 2^{31} - 1$

Para 64 bits $a = 13^{13}$,44485709377909

Rápidos e gastam pouca memória

Universidade Federal Fluminense

Rodando os LCG

Clique aqui

Rodando os LCG

Período pequeno. Estime...
Os bits são aleatórios ?

Rodando os LCG

Período pequeno. Estime...

Os bits são aleatórios ? Considere dois números aleatórios em sequência: x_n e x_{n+1}. Vamos graficar $x_{n+1} \times x_n$ e verificar se o espaço é preenchido uniformemente. As cores mudam a cada $65536 = 2^{16}$ passos.

Universidade Federal Fluminense

Rodando os LCG

Período pequeno. Estime...

Os bits são aleatórios ? Considere dois números aleatórios em sequência: x_n e x_{n+1}. Vamos graficar $x_{n+1} \times x_n$ e verificar se o espaço é preenchido uniformemente. As cores mudam a cada $65536 = 2^{16}$ passos. Clique para a distribuição uniforme

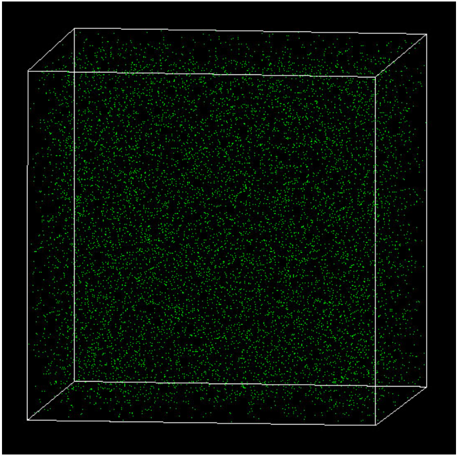
Universidade Federal Fluminense

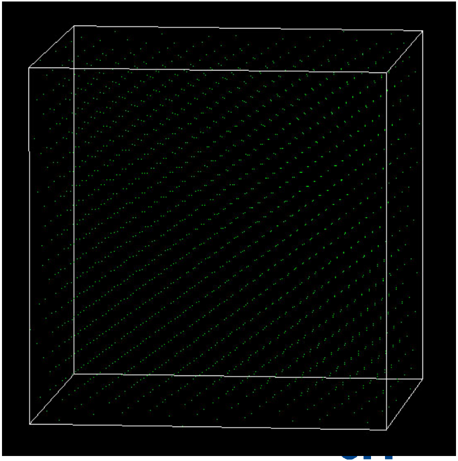
Rodando os LCG

Período pequeno. Estime...

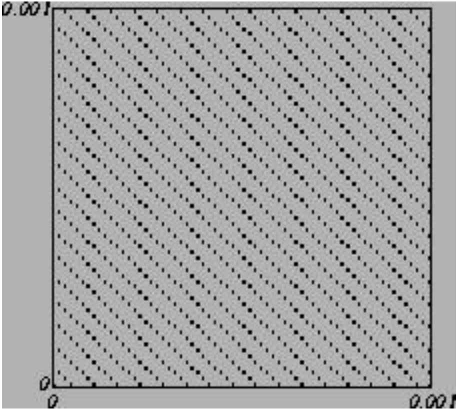
Os bits são aleatórios ? Considere dois números aleatórios em sequência: x_n e x_{n+1}. Vamos graficar $x_{n+1} \times x_n$ e verificar se o espaço é preenchido uniformemente. As cores mudam a cada $65536 = 2^{16}$ passos. Clique aqui para os mapas de retorno

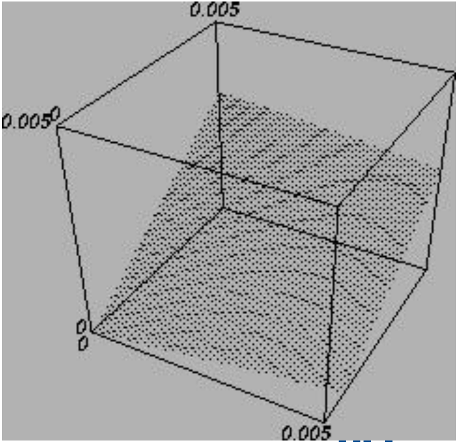
Falha dos LCG





Falha do RANDU





- Recursivos Múltiplos $x_n = (a_1 x_{n-1} + a_2 x_{n-2}) \mod m$ com $a_1 = 271828183$, $a_2 = 314159269$ e $m = 2^{31} - 1$.

Universidade Federal Fluminense

- Recursivos Múltiplos $x_n = (a_1 x_{n-1} + a_2 x_{n-2}) \mod m$ com $a_1 = 271828183$, $a_2 = 314159269$ e $m = 2^{31} - 1$.
- Kirkpatrick and Stoll $x_n = x_{n-103} \otimes x_{n-250}$

Universidade Federal Fluminense

- Recursivos Múltiplos $x_n = (a_1 x_{n-1} + a_2 x_{n-2}) \mod m$ com $a_1 = 271828183$, $a_2 = 314159269$ e $m = 2^{31} - 1$.
- Kirkpatrick and Stoll $x_n = x_{n-103} \otimes x_{n-250}$
- RANLUX (período de 10^{171})

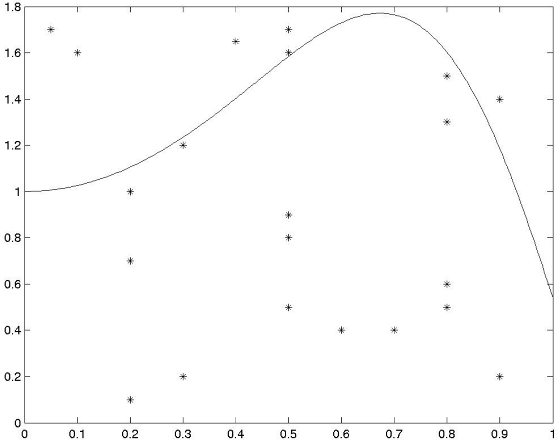
- Recursivos Múltiplos $x_n = (a_1 x_{n-1} + a_2 x_{n-2}) \mod m$ com $a_1 = 271828183$, $a_2 = 314159269$ e $m = 2^{31} - 1$.
- Kirkpatrick and Stoll $x_n = x_{n-103} \otimes x_{n-250}$
- RANLUX (período de 10^{171})
- Tausworthe $x_n = (s1_n \otimes s2_n \otimes s3_n)$, com três números embaralhados com $\otimes$.

Universidade Federal Fluminense

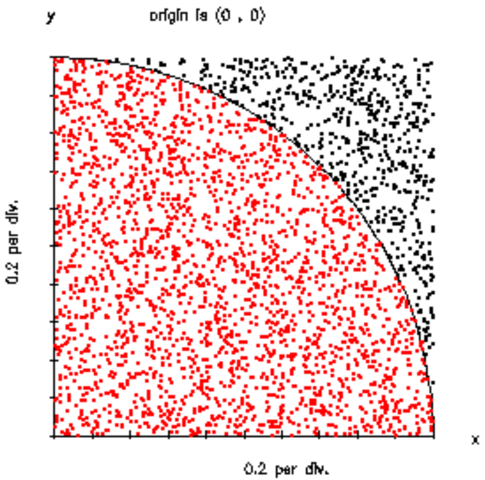
- Recursivos Múltiplos $x_n = (a_1 x_{n-1} + a_2 x_{n-2}) \mod m$ com $a_1 = 271828183$, $a_2 = 314159269$ e $m = 2^{31} - 1$.
- Kirkpatrick and Stoll $x_n = x_{n-103} \otimes x_{n-250}$
- RANLUX (período de 10^{171})
- Tausworthe $x_n = (s1_n \otimes s2_n \otimes s3_n)$, com três números embaralhados com $\otimes$.
- Lagged Fibonacci $r_n = r_{n-A} \otimes r_{n-B} \otimes r_{n-C} \otimes r_{n-D}$ com $A = 471, B = 1586, C = 6988, D = 9689$.

Universidade Federal Fluminense

Método de Monte Carlo



Cálculo de Pi

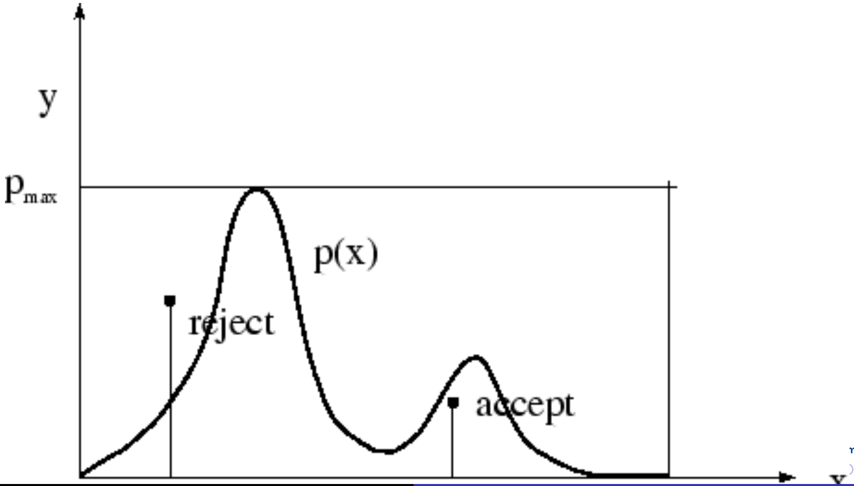


Rejeição por von Neumann

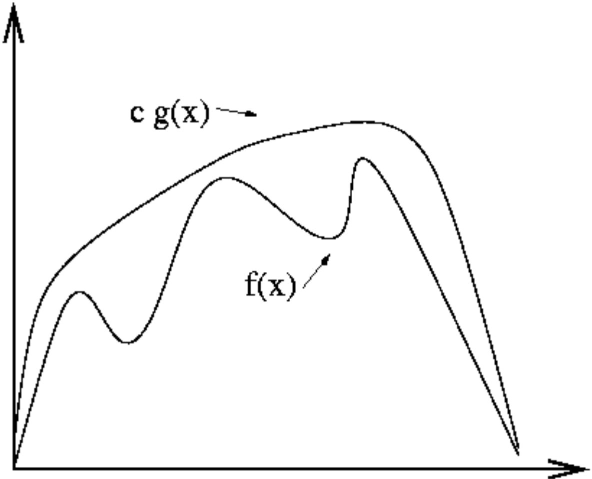
Universidade Federal Fluminense

Rejeição por von Neumann

Método da Rejeição



Método da Rejeição Otimizado



Definição

Seja $p(x)$ a distribuição desejada, com $y = P(x) = \int p(x')dx'$.
$P^{-1}(y)$ é conhecida.
Se y é aleatório (uniforme), então
$x = P^{-1}(y)$ é distribuida segundo $p(x)$.

Exemplos: $p(x) = e^{-x}$, $y = e^{-x}$, $x = -\ln(y)$

Prob: Adapte o método da transformação para a Lorentziana.

Universidade Federal Fluminense

Box-Muller

$$
\begin{aligned}
z_1 &= \sqrt{-2\ln x_1}\cos(2\pi x_2) \\
z_2 &= \sqrt{-2\ln x_1}\sin(2\pi x_2)
\end{aligned}
$$

Universidade Federal Fluminense