Random Processes

Monte Carlo Simulation

1

Random or Stochastic processes

You cannot predict from the observation of one event, how the next will come out

Examples:

Coin: the only prediction about outcome – 50% the coin will land on its tail

Dice: In large number of throws – probability 1/6

2

Question: What is the most probable number for the sum of two dice?

36 possibilities

6 times – for **7**

	1	2	3	4	5	6
1|	2	3	4	5	6	7
2|	3	4	5	6	7	8
3|	4	5	6	7	8	9
4|	5	6	7	8	9	10
5|	6	7	8	9	10	11
6|	7	8	9	10	11	12

3

Applications for MC simulation

- Stochastic processes
- Complex systems (science)
- Numerical integration
- Risk management
- Financial planning
- …

4

How do we do that?

- You let the computer to throw “the coin” and record the outcome
- You need a program that generates randomly a variable
  … with relevant probability distribution

5

Random Number Generators (RNG)

- There are no true random number generators but pseudo RNG!
- Reason: computers have only a limited number of bits to represent a number
- It means: the sequence of random numbers will repeat itself (period of the generator)

6

Good Random Number Generators

- equal probability for any number inside interval [a,b]
- yet independent of the previous number
- long period
- produce the same sequence if started with same initial conditions
- fast

7

Linear Congruent Method for RNG

Generates a random sequence of numbers $\{x_1, x_2, \ldots x_k\}$ of length M over the interval $[0,M\text{-}1]$

$$\boxed{x_i = \mathrm{mod}(ax_{i-1} + c, M)}$$

- starting value x_0 is called "seed"
- coefficients *a* and *c* should be chosen very carefully

note:

$$\mathrm{mod}(b, M) = b - \mathrm{int}(b/M) * M$$

8

Example:

$$x_i = \mathrm{mod}(ax_{i-1} + c, M)$$
$$\mathrm{mod}(b, M) = b - \mathrm{int}(b/M) * M$$

a=4, c=1, M=9, x_1=3
x_2 = 4
x_3 = 8
x_4 = 6
$x_{5\text{-}10}$ = 7, 2, 0, 1, 5, 3

interval: 0-8, i.e. [0,M-1]
period: 9 i.e. M numbers (then repeat)

9

Random Numbers on interval [A,B]

- Scale results from x_i on [0,M-1] to y_i on [0,1]

$$y_i = x_i / (M - 1)$$

- Scale results from x_i on [0,1] to y_i on [A,B]

$$y_i = A + (B - A)x_i$$

10

Magic numbers for Linear Congruent Method

- M (length of the sequence) is quite large
- However there is no overflow
  (for 32 bit machines $M=2^{31} \approx 2*10^9$)
- Good "magic" number for linear congruent method:

$$\boxed{x_i = \mathrm{mod}(ax_{i-1} + c, M)}$$

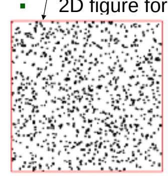
a = 16,807, c = 0, M = 2,147,483,647

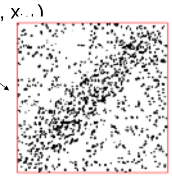
11

How can we be check the RNG?

Plots:

- 2D figure, where x_i and y_i are from two random sequences (parking lot test)
- 3D figure (x_i, y_i, z_i)
- 2D figure for correlation $(x_i, x_{\ldots})$





12

How can we check the RNG?

Example of other assessments

Uniformity. A random number sequence should contain numbers distributed in the unit interval with equal probability. Use bins.

k-th momentum $\left\langle x^k \right\rangle = \frac{1}{N}\sum_{i=1}^{N} x_i^k \approx \frac{1}{k+1}$

near-neighbor correlation $\frac{1}{N}\sum_{i=1}^{N} x_i x_{i+k} \approx \frac{1}{4}$

13

"Industrial" methods

- rand
- random
- drand48
- rn
- drand
- srand
- …

1. call SEED
   Changes the starting point of the pseudorandom number generator.
2. call RANDOM
   Returns a pseudorandom number greater than or equal to zero and less than one from the uniform distribution.

For real applications use "industrial" random number generators

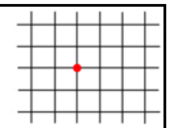
14

Practice 1 (homework)

1. Write a program to generate random numbers using the linear congruent method
2. Plot 2D distribution for two random sequences x_i and y_i
3. Plot 2D distribution for correlation (x_i, x_{i+4})
4. Evaluate 5-th moment of the random number distribution
5. Use some built-in RNG for problems 2-4.

15

Random Walk



A random walk is a sequence of unit steps where each step is taken in the direction of one of the coordinate axis, and each possible direction has equal probability of being chosen.

Random walk on a lattice:

- In two dimensions, a single step starting at the point with integer coordinates (x,y) would be equally likely to move to any of one of the four neighbors (x+1,y), (x-1,y), (x,y+1) and (x,y-1).
- In one dimension walk there are two possible neighbors
- In three dimensions there are six possible neighbors.

16

Random Walk simulates:

- Brownian motion
  (answer the question - how many collisions, on average, a particle must take to travel a distance R).
- Electron transport in metals, …
- …

17

Practice 2 (random walk)

1. Write a program that simulate a random 2D walk with the same step size . Four directions are possible (N, E, S, W).
   Your program will involve two large integers, M = the number of random walks to be taken and N = the maximum number of steps in a single walk.
2. Find the average distance to be from the origin point after N steps
3. Is there any finite bound on the expected number of steps before the first return to the origin?

18

Monte Carlo Integration

- There are very many methods for numerical integration
- Can MC approach compete with sophisticated methods?
- Can we gain anything from integration by "gambling"?

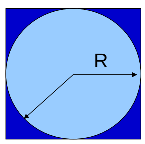
Problem: High-Dimensional Integration

Example: Integration for a system with 12 electrons.

- 3*12=36 dimensional integral
- If 64 points for each integration then $=64^{36}$ points to evaluate
- For 1 Tera Flop computer = 10^{53} seconds
- That is … 3 times more then the age of the universe!

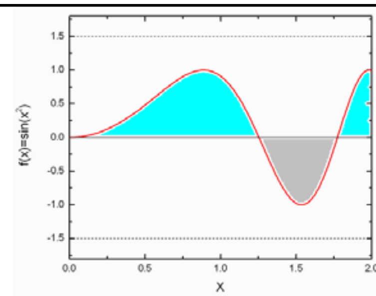
Integration by rejection hit and miss method

Example: area of a circle
Radius: R
Area of the square: $4R^2$



1. loop over N
2. generate a pair of random numbers x and y on [-1,1]
3. if (x*x+y*y) < 1 then m=m+1
4. since A_{circle}/A_{square} = m/N
5. A_{circle} = m/N*A_{square} = (m/N)*$4R^2$

One more example



Compute N pairs of random numbers x_i and y_i with $0.0 \le x \le 2.0$ and $-1.5 \le y \le 1.5$.

$$F_n = A\left(\frac{n_+ - n_-}{N}\right)$$

Integration by mean value

$$I = \int_a^b f(x)dx = (b-a)\langle f \rangle \quad \text{and} \quad \langle f \rangle = \frac{1}{N}\sum_{i=1}^{N} f(x_i)$$

$$\boxed{I = \int_a^b f(x)dx = (b-a)\frac{1}{N}\sum_{i=1}^{N} f(x_i)}$$

Traditional methods (Simpson, …) – the N points are chosen with equal spacing

Monte Carlo method – random spacing

Multidimensional Monte Carlo

$$\boxed{\int_a^b dx \int_c^d dy f(x,y) \cong (b-a)(d-c)\frac{1}{N}\sum_{i=1}^{N} f(x_i, y_i)}$$

Error in Monte Carlo integration

- error in Monte Carlo 1D integration $\frac{1}{\sqrt{N}}$
- error in "common" 1D integration $\frac{1}{N}$
- error in "common nD integration $\frac{n}{N/n}$
- error in Monte Carlo nD integration $\frac{1}{\sqrt{N}}$

at n=4 the error in Monte Carlo integration is similar to that of conventional scheme

25

Practice: Integration

- Use Monte Carlo integration (both rejection and mean value methods) to evaluate

$$\int_0^3 \exp(-x)dx \quad \text{and} \quad \int_0^5 \sin(2x^2)dx$$

- Evaluate 7-D integral

$$\int_0^1 dx_1 \int_0^1 dx_2 \int_0^1 dx_3 \int_0^1 dx_4 \int_0^1 dx_5 \int_0^1 dx_6 \int_0^1 (x_1 + x_2 + \ldots + x_7)^2 dx_7$$

26

Non-uniform distributions

Most situation in physics – random numbers with non-uniform distribution

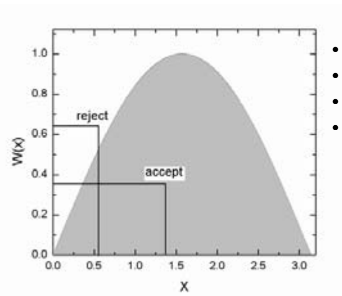
- radioactive decay
- experiments with different types of distributions
- …

Principal idea: Generating non-uniform random number distributions with a uniform random number generators

27

Method 1: von Neumann rejection

Generating non-uniform distribution with a probability distribution w(x)



- generate (x_i, y_i)
- if $y_i < w(x_i)$, accept
- if $y_i > w(x_i)$, reject
- The x_i so accepted will have the weighting w(x)

28

Method 2: Inversion method

- Works if the function you are trying to use for a distribution has an inverse

$$y = F(x)$$
$$x = F^{-1}(y)$$

- Example: exponential distribution

$$w(x) = \exp(-x)$$
$$x = -\ln(1-y)$$

very many program libraries have most common non-uniform distributions

29

Practice: non-uniform distribution

Use the von Neumann rejection technique to generate a normal distribution of standard deviation 1.0

30