Experiment 2.1: Generating Random Numbers

Introduction

Many processes in nature are practically random. A famous example is the thermal motion of molecules which is an important basis for the understanding of diffusion. In order to be able to model such processes with an appropriate computer algorithm one needs random numbers. Obviously, a computer is a deterministic machine and as such can only perform deterministic calculations. Nevertheless, a computer *can* indeed calculate a (deterministic) chain of numbers which fulfills certain criteria such that they can be regarded as so-called pseudo random numbers. These criteria are:

- the numbers are uncorrelated
- they have a well-defined range of values
- they have a well defined probability distribution
- the chain is sufficiently long so that practically no periodicity can be observed (note that a random number generator is always periodic since the computer has only a finite set of states and performing the same calculation from an identical initial state leads to the same sequence of numbers)

An algorithm which calculates a chain of numbers fulfilling the criteria above is called a (pseudo) random number generator. Basic random number generators usually create uniformly distributed random numbers. These build the basis for the calculation of random numbers with other specific distributions.

Objectives

- to program a linear-congruential random number generator for uniformly distributed random numbers
- to program a random number generator for normally distributed random numbers using the Box-Müller algorithm
- to test these random number generators against the criteria above

Random numbers with uniform probability distribution

There are different algorithms available for the calculation of uniformly distributed random numbers. The most simple ones are the linear-congruential generators. Here, integer valued random numbers x_i are calculated by performing the calculation

$$x_{i+1} = \text{mod}\Big(Ax_i + B, M\Big) \tag{1}$$

successively, starting from some initial value x_0.
The function $\text{mod}(y, z)$ returns the rest of the integer division y/z. Hence, this generator calculates integer random numbers in the range $[0, M-1]$ which can be normalized later on by a division by M.
The two constants A and B are so-called magical numbers. The quality of the random number generator depends crucially — and in an almost magical fashion — on how skilful these numbers are chosen. E. g., imagine you would choose A and B even: then, the generator would only return either even or odd random numbers (depending on M), certainly a correlation which is not desired.

Random numbers with normal probability distribution

Random numbers y_{2i} with a Gauß distribution with mean $\overline{y}$ and variance σ^2 can be calculated from two uniformly distributed random numbers x_{2i} and x_{2i-1} by calculating

$$y_{2i} = \sigma\sqrt{-2\ln(1 - x_{2i})}\ \cos(2\pi x_{2i-1}) + \overline{y}. \tag{2}$$

The random numbers x_i have to be in the range $[0; 1[$. This method is called the Box-Müller algorithm.

Experiment

Write a code which calculates uniformly distributed random numbers from a linear-congruential generator as described above. Re-normalize the random numbers such that they are in the interval $[0, 1[$. On the basis of these uniformly distributed random numbers develop also a code for the calculation of normally distributed random numbers using the Box-Müller method.
Test the generators for different sets of magical numbers, especially:

- $A = 100$, $B = 104001$, $M = 714025$
- $A = 1001$, $B = 100000$, $M = 714025$
- $A = 137$, $B = 150887$, $M = 714025$
- $A = 1103515245$, $B = 12345$, $M = 2^{32} - 1$

Test and discuss the following points:

1. **Correlations:** calculate $N = 10000$ uniformly distributed random numbers building pairs (x_i, x_{i+1}). Represent these pairs as 5000 points in a coordinate system. When your random numbers are correlated, you will see a regular pattern instead of a random crowd of points.

2. **Periodicity** How long is the period of your random number generator? Is it always equal to M? Estimate the time that your computer would need to calculate one period of random numbers. Which of the suggested magical numbers are the best ones?

3. **Uniform distribution:** Split the interval $[0; 1[$ in $I = 100$ equal interval parts. Investigate the variance $\langle \chi^2 \rangle$ of the number of random numbers which are in each of the interval parts. If n_i is the number of random numbers in the ith interval part and $\overline{n} = N/I$ the mean number of random numbers per interval part, then it is
$$\langle \chi^2 \rangle = \frac{1}{I} \sum_{i=1}^{I} (n_i - \overline{n})^2 \,. \tag{3}$$
How does $\langle \chi^2 \rangle$ and the relative error $\langle \chi^2 \rangle^{1/2}/N$ depend on the overall number N of drawn random numbers? Vary N from 10^2 to 10^7.

4. **Gauß distribution:** Calculate normally distributed random numbers with zero mean and variance 1. Once again, split an appropriate interval in $I = 100$ equal interval parts. Calculate the probability distribution directly from $N = 1000000$ random numbers. Compare the numerical result graphically with the analytical form
$$p(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}. \tag{4}$$