Computational Physics
Spring 2021

Lalit Chaudhary

February 24, 2021

Pseudo random Generator and Random Walk

Contents

List of Figures

1 Random Number Generator

1.1 Implementation

A linear congruent generator was implement in python as below:

```
#linear congruent generator
def LCG( m, a, c, lim, I0):
    r = [I0/m]
    for i in range (lim-1):
        I = (a*I0 + c)%m
        I0 = I                  #Update seed for next iteration
        ri = I0/m                #Random number between 0 and 1
        r.append(ri)
    return r
```

Figure 1: Code snippet for implementation of LCG in python

1.2 Uniformity Test and Comparison

The LCG was implemented with the parameters

$$a = 57$$
$$c = 1$$
$$m = 256$$
$$r_1 = 1$$

Then the following tests were performed for both the above implemented LCG as well as the in-built "random" module (Mersenne Twister generator) in python.

The following plots were obtained for N = 1000 randomly generated numbers:

1.2.1 Observations

From figures (2) and (3), it is seen that the randomly generated numbers in both the cases fill up almost all of the square. This implies that the numbers are distributed through the range (0,1).

The figures (4) and (5) show contrasting natures. The distinct pattern in figure (4) implies that the numbers are not random in contrast to the evenly distributed pattern filling up the square seen in fig (5). Hence, it can be concluded that the LCG implementation with given parameters result in random numbers wherein consecutive numbers are co-related.

In figure (6) and (7), the blue lines denote the deviation of the approximate 2nd moment from the exact value of moment. It can be seen that in both the cases, the deviation reaches 0 for higher N. It is also noticeable the the deviation is asymptotically bounded by $\frac{1}{\sqrt{N}}$.

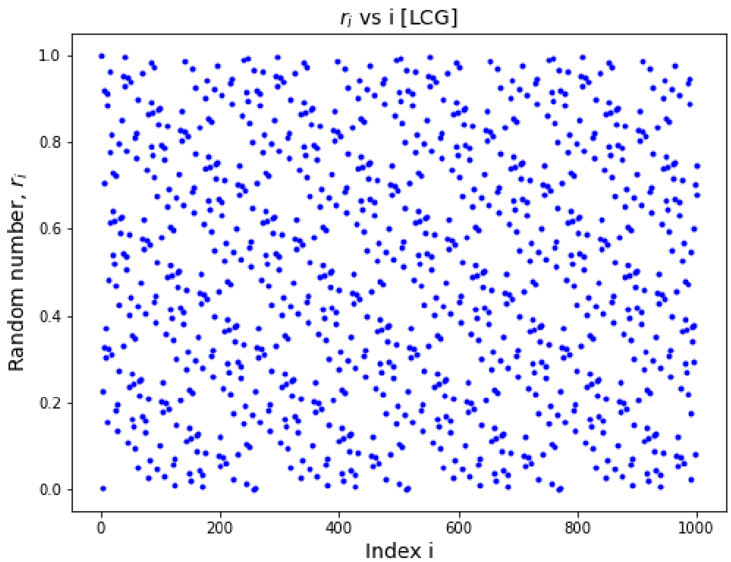

Figure 2: Random numbers vs index [LCG]

1.3 Random Walk

With 1000 iteration of the random walk, the following results were obtained:

The linear dependence of variance and the distinct lattice steps visited can be seen from plots in figure (9, 10)

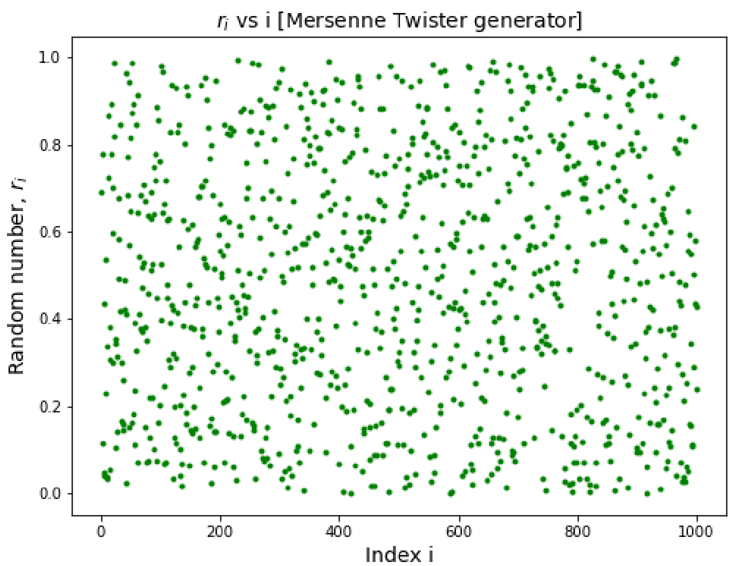

Figure 3: Random numbers vs index [MTG]

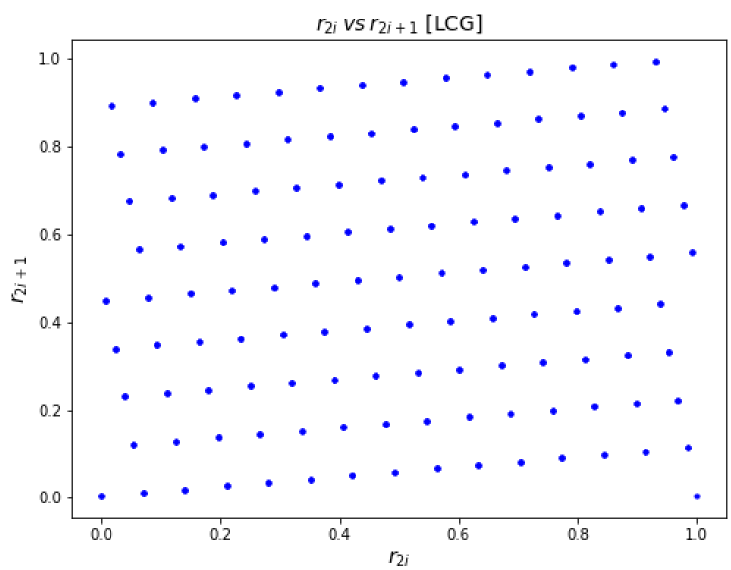

Figure 4: Consecutive numbers as a function of each other [LCG]

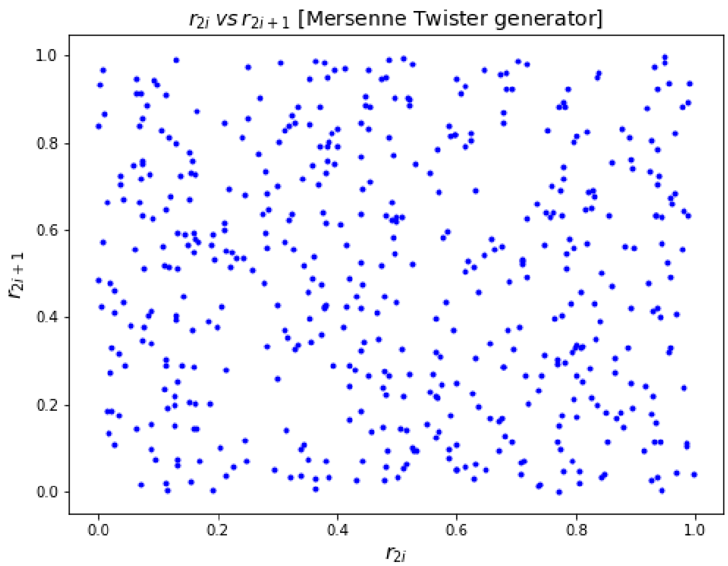

Figure 5: Consecutive numbers as a function of each other [MTG]

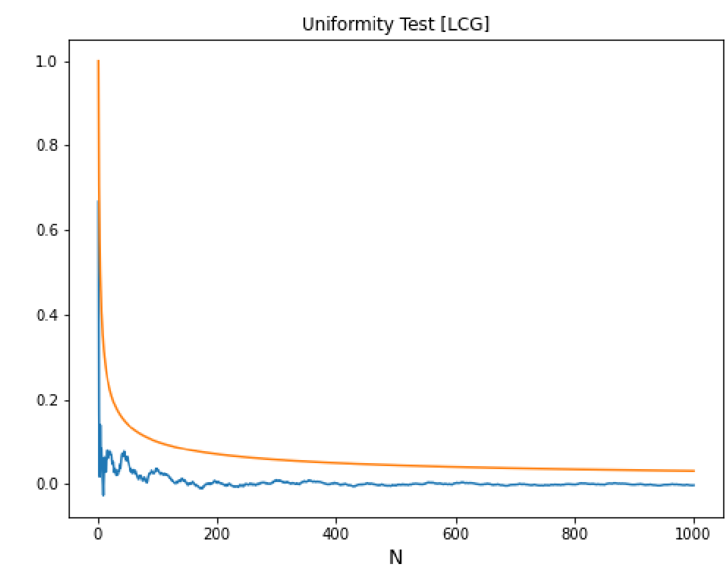

Figure 6: Uniformity Test [LCG]

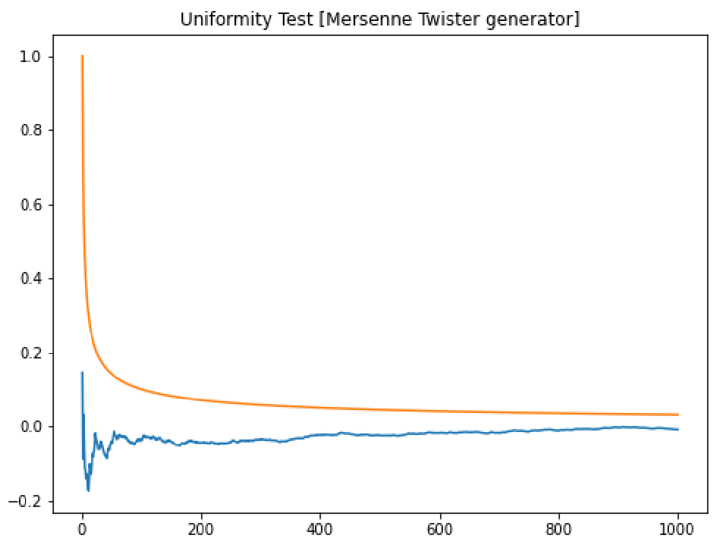

Figure 7: Uniformity Test [MTG]

```
For N =  4
Expected values of x, <x> =  2.422
Variance of x, <del (x^2)> =  2.457915999999999
For N =  8
Expected values of x, <x> =  4.82
Variance of x, <del (x^2)> =  4.599599999999999
For N =  16
Expected values of x, <x> =  9.53
Variance of x, <del (x^2)> =  10.007100000000008
For N =  32
Expected values of x, <x> =  19.148
Variance of x, <del (x^2)> =  19.218096000000003
For N =  64
Expected values of x, <x> =  38.364
Variance of x, <del (x^2)> =  40.611504000000195
For N =  128
Expected values of x, <x> =  76.614
Variance of x, <del (x^2)> =  83.18700399999943
For N =  256
Expected values of x, <x> =  153.986
Variance of x, <del (x^2)> =  164.85180400000536
```

Figure 8: Results of random walk with different number of steps

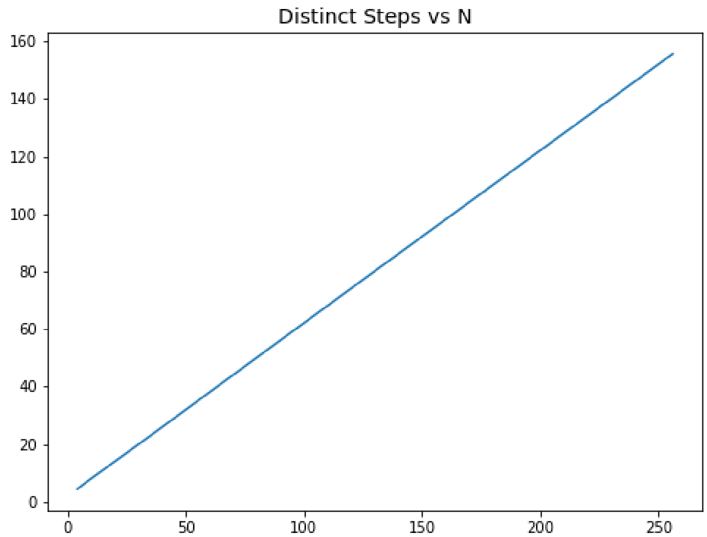

Figure 9: Distinct Steps variation with N

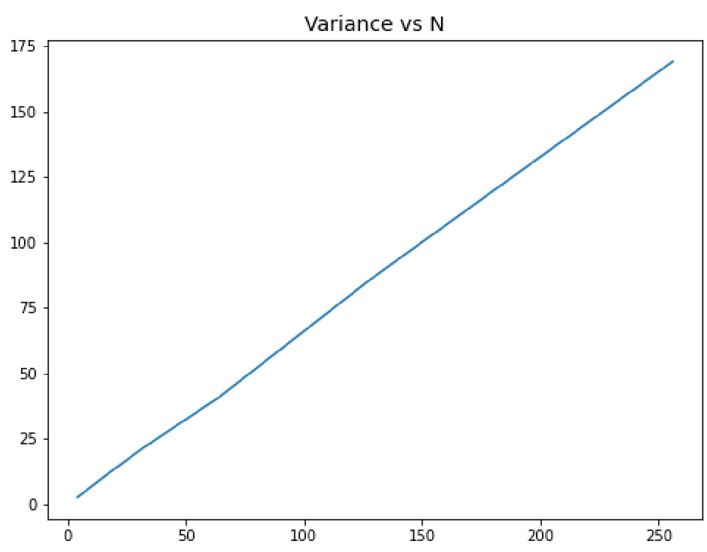

Figure 10: Variance with N