SHARIF UNIVERSITY OF TECHNOLOGY
PHYSICS DEPARTMENT

Problem Set 3

Student name: *Ali Fayyaz - 98100967*

Course: *Computational Physcis - (Spring 2023)*
Due date: *March 14, 2023*

Exercise 3.2

Simulate a Ballistic Deposition with Relaxation model for a line of width 200 units. Set periodic boundary conditions.

(*a*) Show the dynamics of the model on the screen.

(*b*) Calculate the mean height and the roughness (standard deviation) of the surface for a number of time intervals. How many particles are needed to reach saturation?

(*c*) Sketch the surface roughness vs. time.

(*d*) Calculate β, α and z; explain what you understand from their values.

Answer. In this model, a position on the lattice is chosen randomly and the particle checks the height of the two neighboring bins to the left and right, and falls on the one with minimum height. Two functions were defined as follows, for graphical representation and numerical analysis of the model, respectively:

- **graphic_bd_rlx():**

  Optimized for graphical representation of the model, as the name suggests, this function takes the number of *bins* or positions where the particles can drop, the number of all the particles that will be dropped, and aesthetic parameters for changing the colors of the particles. The function returns a matrix that corresponds to the pixels on an image. Using this matrix, the model can be demonstrated graphically with the efficient *matplotlib.pyplot.imshow()* method.

- **numeric_bd_rlx():**

  This function generates numberical data for the analysis of the model, particularly for the calculation of the saturation parameters. It takes, similar to its graphic counterpart explained above, the number of *bins* and the number of particles, and also the number of *snapshots* that will be used to study the model. As the particles drop and the model grows in height, one can virtually pause and take *snapshots* and study different factors of the model at that certain moment of time, e.g. the mean height and the variance. These snapshots can be taken, if the user wills, at linearly increasing moments or at moments that increase with the powers of 2, i.e. after the deposition of 2, 4, 8, 16, 32, ... particles.

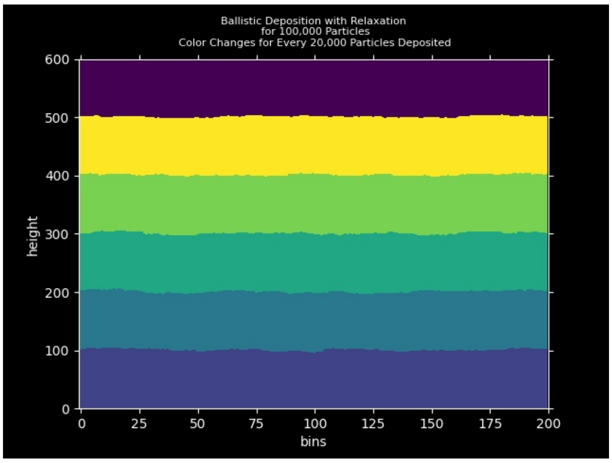
The second function was called for different number of bins (denoted "L" in the plots) and snapshots were taken untill the model reaches saturation. The saturation points were determined approximately by observing the points on the scatter plot. In general, this should be possible by fitting two lines on the points, and finding their intersection to determine the saturation point; however, due to the cumbersome calculations and the tight deadline, resort to approximate fitting of the lines was inevitable. Nevertheless, the results were satisfactory and corresponded with theory.

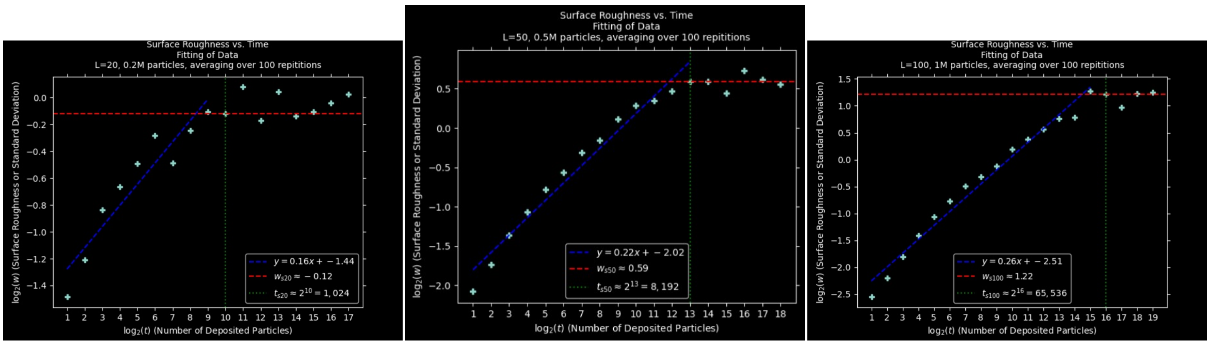
β, α and z were calculated to be as follows:

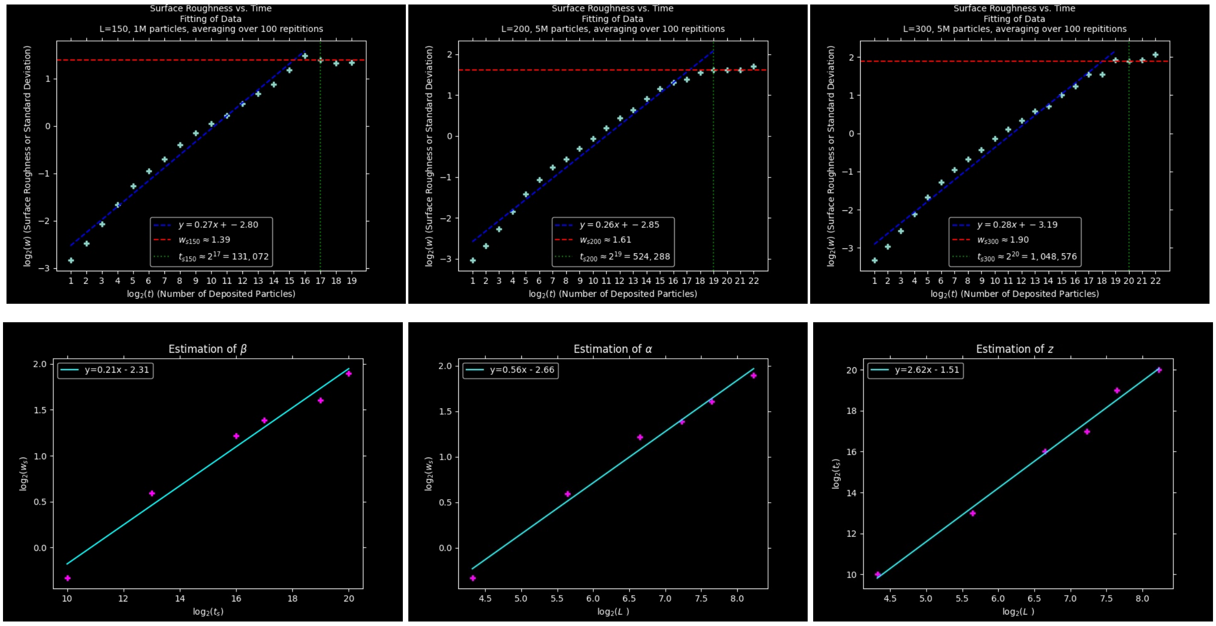
$$\beta = 0.21 \qquad \alpha = 0.56 \qquad z = 2.6$$

This means the roughness of the surface reaches saturation with approximately the power of 0.21 of the time of saturation. It is related to the length L of the surface (the number of bins) as follows:

$$w_s \sim t_s^{\beta} \sim L^{z\beta} \sim L^{\alpha}$$







Exercise 3.3

Simulate a Ballistic Deposition model for a line of width 200 units. Set periodic boundary conditions.

(*a*) Show the dynamics of the model on the screen.

(*b*) Calculate the mean height and the roughness (standard deviation) of the surface for a number of time intervals. How many particles are needed to reach saturation?

(*c*) Sketch the surface roughness vs. time.

(*d*) Calculate β, α and z; explain what you understand from their values.

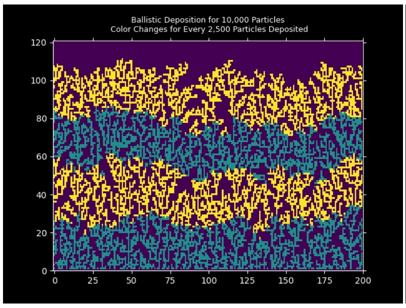
Answer. The approach taken at examining this model was rather identical to the previous exercise. Two functions **graphic_bd()** and **numeric_bd()** were defined similar to the ones explained above. The only difference was the way the particles were deposited, as this time they would check the height of the neighbors and stick to the taller pillar, as described in the textbook. As a result, one can expect to see *pores* in this model.

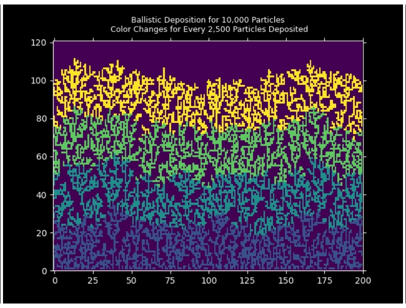
Again an identical approach was taken at finding β, α and z, which were calculated to be as follows:

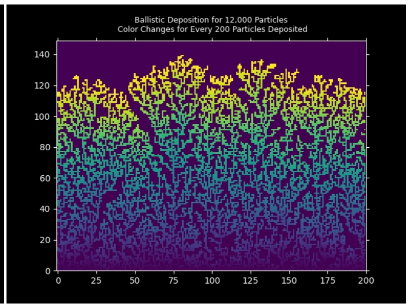
$$\beta = 0.23 \qquad \alpha = 0.31 \qquad z = 1.3$$

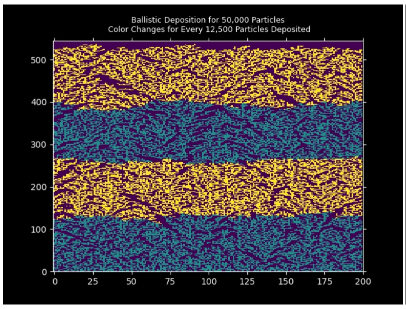
In this model also, the roughness of the surface is related to the length L of the surface (the number of bins) as follows:

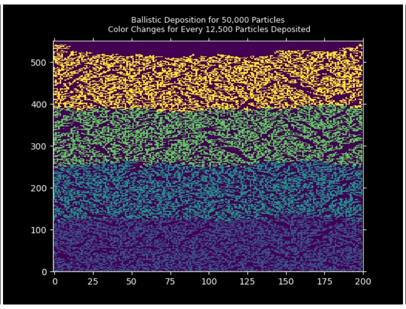
$$w_s \sim t_s^{\beta} \sim L^{z\beta} \sim L^{\alpha}$$

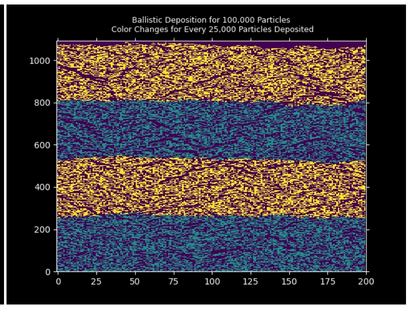

Ballistic Deposition for 10,000 Particles
Color Changes for Every 2,500 Particles Deposited

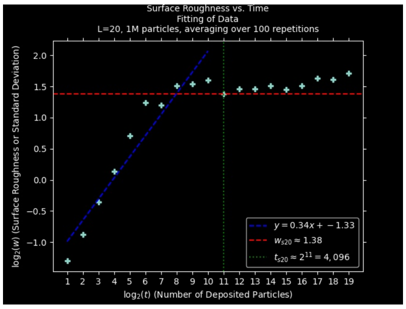

Ballistic Deposition for 10,000 Particles
Color Changes for Every 2,500 Particles Deposited

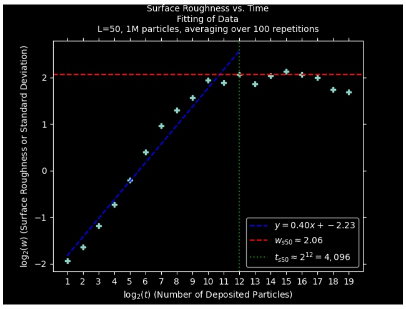

Ballistic Deposition for 12,000 Particles
Color Changes for Every 200 Particles Deposited

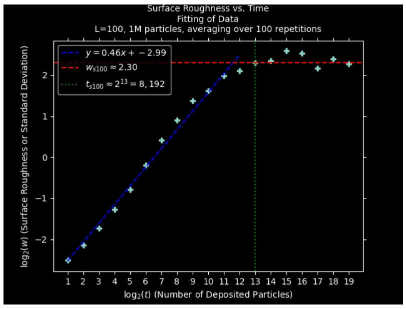

Ballistic Deposition for 50,000 Particles
Color Changes for Every 12,500 Particles Deposited

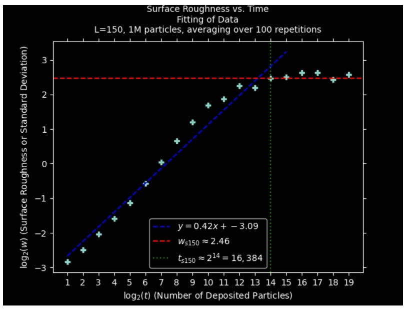

Ballistic Deposition for 50,000 Particles
Color Changes for Every 12,500 Particles Deposited

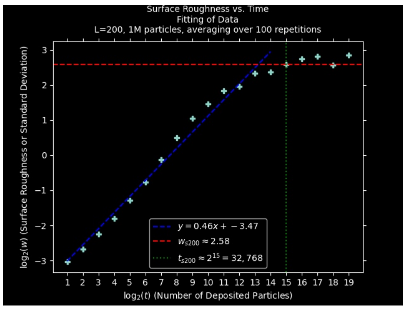

Ballistic Deposition for 100,000 Particles
Color Changes for Every 25,000 Particles Deposited

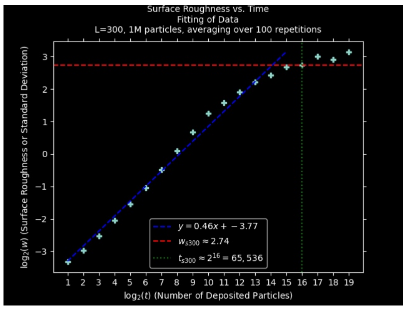

Surface Roughness vs. Time
Fitting of Data
L=20, 1M particles, averaging over 100 repetitions
$y = 0.34x + -1.33$
$w_{s20} \approx 1.38$
$t_{s20} \approx 2^{11} = 4,096$
$\log_2(w)$ (Surface Roughness or Standard Deviation)
$\log_2(t)$ (Number of Deposited Particles)

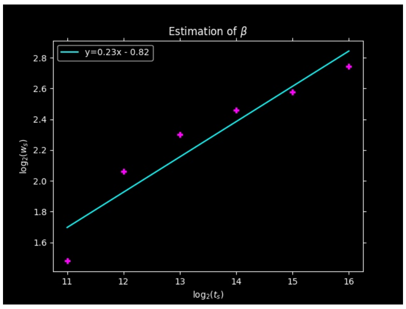

Surface Roughness vs. Time
Fitting of Data
L=50, 1M particles, averaging over 100 repetitions
$y = 0.40x + -2.23$
$w_{s50} \approx 2.06$
$t_{s50} \approx 2^{12} = 4,096$
$\log_2(w)$ (Surface Roughness or Standard Deviation)
$\log_2(t)$ (Number of Deposited Particles)

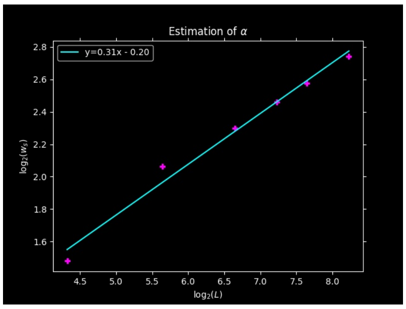

Surface Roughness vs. Time
Fitting of Data
L=100, 1M particles, averaging over 100 repetitions
$y = 0.46x + -2.99$
$w_{s100} \approx 2.30$
$t_{s100} \approx 2^{13} = 8,192$
$\log_2(w)$ (Surface Roughness or Standard Deviation)
$\log_2(t)$ (Number of Deposited Particles)

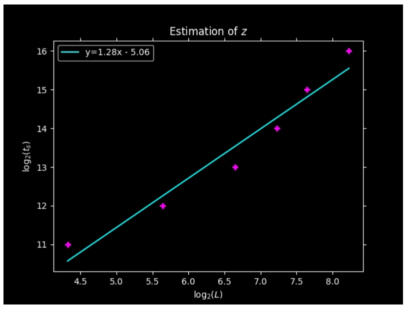

Surface Roughness vs. Time
Fitting of Data
L=150, 1M particles, averaging over 100 repetitions
$y = 0.42x + -3.09$
$w_{s150} \approx 2.46$
$t_{s150} \approx 2^{14} = 16,384$
$\log_2(w)$ (Surface Roughness or Standard Deviation)
$\log_2(t)$ (Number of Deposited Particles)


Surface Roughness vs. Time
Fitting of Data
L=200, 1M particles, averaging over 100 repetitions
$y = 0.46x + -3.47$
$w_{s200} \approx 2.58$
$t_{s200} \approx 2^{15} = 32,768$
$\log_2(w)$ (Surface Roughness or Standard Deviation)
$\log_2(t)$ (Number of Deposited Particles)


Surface Roughness vs. Time
Fitting of Data
L=300, 1M particles, averaging over 100 repetitions
$y = 0.46x + -3.77$
$w_{s300} \approx 2.74$
$t_{s300} \approx 2^{16} = 65,536$
$\log_2(w)$ (Surface Roughness or Standard Deviation)
$\log_2(t)$ (Number of Deposited Particles)


Estimation of β
y=0.23x - 0.82
$\log_2(w_s)$
$\log_2(t_s)$


Estimation of α
y=0.31x - 0.20
$\log_2(w_s)$
$\log_2(L)$

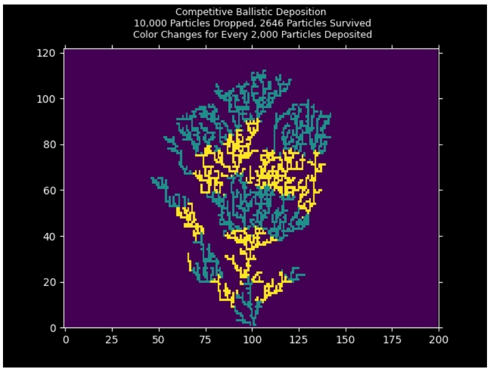

Estimation of z
y=1.28x - 5.06
$\log_2(t_s)$
$\log_2(L)$

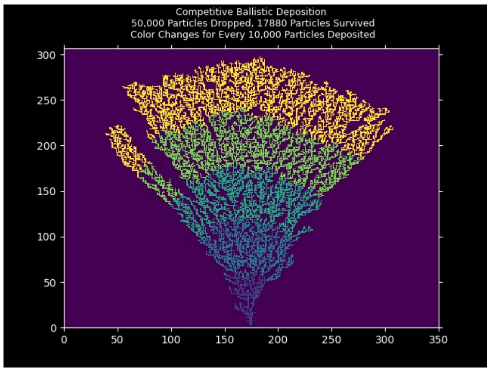
Exercise 3.4

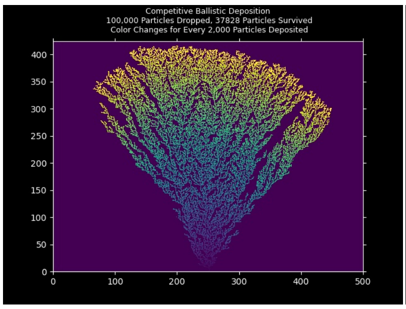
Correlation Length in Ballistic Deposition: In the model described in the previous exercise, instead of using a surface to deposit particles onto, start with a single point and study the long range effects of this single point on the entire surface.

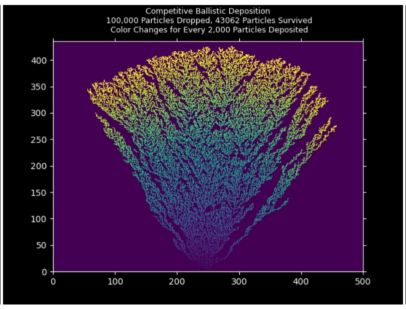
Answer. In this model, a number of particles are dropped, many of which fall off the plot and some of them stick to the starting particle, AKA *the seed*, and the model seems to *grow* or *stem* from this initial point. Only one function was defined for this exercise: **graphic_cbd()**, which has similar input parameters as the functions described earlier, and it outputs a matrix for the pixels of an image. This matrix is initialized to zero, except for the one point in the middle and bottom of the image: the seed. Particles randomly choose different bins to fall into, and if close enough to the seed or the branches, stick to them, otherwise are disregarded. Thus the number of particles the user inputs, is not, in the end, the number of particles that have *survived*, which is clearly smaller.

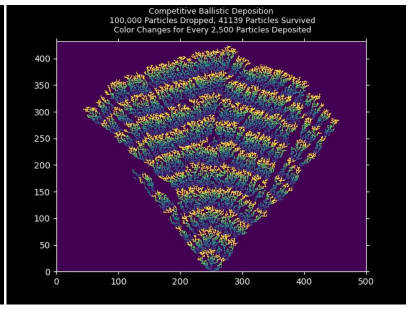
In order to examine the numerical data, 100 samples were *grown*, the width of each sample was measured 20 times as it grew, and the average was taken. To measure the width of the sample, all that needs done is to count the non-zero elements of the image matrix accross the rows. It was found that the width growth rate was linear and thus the correlation length grows in a linear fashion with time.

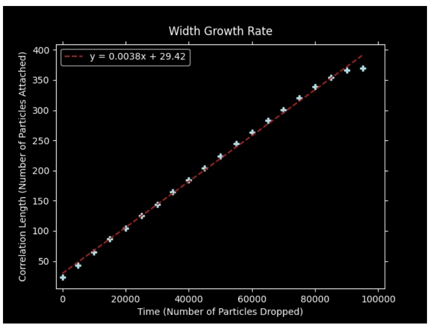












Exercise 3.5

> Simulate a Pin-Like Competitive Ballistic Deposition model, as described in the text book.

Answer. Basically, one needs the data for a histogram to plot this model. So a function was defined to generate such data, and *matplotlib.pyplot.bar()* was used to graph the data.

- **pin_bd():**

  This function takes as input the number of bins and the number of particles as parameters of the ballistic deposition. It also takes the number of adjacent bins that a particle sees before dropping. The particle will be dropped at a random bin. However, it will not settle down immediately; rather, it will examine 10 (num_adjacent_bins) bins to the right and choose one randomly, with a probability that is proportional with the height of the bin, and settle on the chosen bin. All bins are initialized to heights of 1. The function will return a 1D array that shows the heights of every bin.