

Random Walks, the Diffusion Equation, and Cluster Growth

Tahoe Schrader, Ksenia Sokolova, Xinmeng Tong
PHYS566

Abstract

In this assignment we study the random walks and their application in various systems. First we analyze the 2D random walk. Then observe that behavior of the random walkers can use to illustrate a diffusion process satisfying the diffusion equation, which is simulated using the finite difference form model. Furthermore, cluster growth is simulated by diffusion limited aggregation (DLA) model and the 'mass'-radius relation is given in our computation. Our code is available on GitHub:

https://github.com/tahoeschrader/PHYS566_group4_project-1a_walk-diffusion-cluster/tree/ksenia-presentation

1 Theory

1.1 Random Walks in 2D

Consider a 2D random walk. The displacement in the x direction is equal to the sum of displacement in x direction at every step.

$$X = \sum_{i=1}^n x_i \quad (1)$$

The expectation value of displacement in the x direction can be found using the distributive property of the expectation for identically distributed random variables:

$$\langle x \rangle = \sum_{i=1}^n \langle x_i \rangle = \sum_{i=1}^n (0.25)(-1) + (0.25)(1) = 0 \quad (2)$$

This happens because there is equal probability to step right or left. The probability is 0.25 because overall we are in 2D, and so there is 50% chance we will move in the vertical direction.

For the mean x-squared displacement we get

$$\langle x^2 \rangle = \langle (x_1 + x_2 + x_3 \dots)(x_1 + x_2 + x_3 \dots) \rangle \quad (3)$$

$$\langle x^2 \rangle = \langle x_1^2 \rangle + \langle x_2^2 \rangle + \langle x_3^2 \rangle + \langle x_1 x_2 \rangle + \langle x_1 x_3 \rangle + \langle x_2 x_3 \rangle \dots \quad (4)$$

We see that the result consists of combinations of $\langle x_i^2 \rangle$ and $\langle x_i x_j \rangle$. Note that

$$E(x_i x_j) = 1 * 0.25^2 + 1 * 0.25^2 + (-1) * 0.25^2 + (-1) * 0.25^2 = 0 \quad (5)$$

Therefore, all non-square terms vanish.

For the square terms, on the contrary, we always get $x_i = \pm 1$ and therefore $x_i^2 = 1$ and

$$E(x_i^2) = 1 * 0.25 + 1 * 0.25 = 0.5 \quad (6)$$

This means that

$$\langle x^2 \rangle = \sum_{i=1}^N \langle x_i^2 \rangle = \sum_{i=1}^N 0.5 = 0.5N \quad (7)$$

Now, for the squared distance from the origin, we get

$$\langle d^2 \rangle = \langle x^2 + y^2 \rangle = \langle x^2 \rangle + \langle y^2 \rangle = 0.5N + 0.5N = N = 4Dt \quad (8)$$

if $t=N$, one step at one time interval.

Where we again used the distributivity of the mean and the fact that $E(x^2) = E(y^2)$ in case $P(x) = P(y)$

1.2 Diffusion Equations and the Finite Difference Form

The diffusion equation in 1D is written,

$$\frac{\partial \rho(x, t)}{\partial t} = D \nabla^2 \rho(x, t), \quad (9)$$

where D is the diffusion constant. Equation 9 is turned into an iterable form by noting: $\rho(x, t) = \rho(i\Delta x, n\Delta t) = \rho(i, n)$. This is the finite difference form¹.

After using the formal definition of derivatives and algebraically manipulating Equation 9 in the finite difference form, we get

$$\rho(i, n+1) = \rho(i, n) + \frac{D\Delta t}{\Delta x^2} (\rho(i+1, n) + \rho(i-1, n) - 2\rho(i, n)), \quad (10)$$

where Δt and Δx are the step sizes in an iteration. This solution requires knowledge of initial conditions. We must assume that the x displacement is known at times prior to and including $t_n = n\Delta t$. Two consecutive steps prior to the first unknown step is sufficient to solve such an equation. Finally, to guarantee stability, the following criterion must be met

$$\Delta t \leq \frac{(\Delta x)^2}{2D}. \quad (11)$$

We will use an initial density profile that is sharply peaked around $x = 0$, but extends over a few grid sites to resemble a box. This is sufficient for generating the solution to the diffusion equation. Interestingly, after a couple iterations, the box profile will diffuse into a Gaussian normal distribution. The 1D Gaussian normal distribution has the form,

$$\rho(x, t) = \frac{1}{\sqrt{2\pi\sigma(t)^2}} \exp\left(-\frac{x^2}{2\sigma(t)^2}\right), \quad (12)$$

where $\sigma(t) = \sqrt{2Dt}$.

The spatial expectation value, $\langle x(t)^2 \rangle$, of Equation 12 is equal to $\sigma(t)^2$. Expectation values are calculated according to the equation

$$\langle x \rangle = \int_{-\infty}^{\infty} f(x) x dx, \quad (13)$$

¹The finite difference form must be used because the diffusion equation is time dependent. Therefore, a relaxation method cannot be used.

where $f(x)$ is the Gaussian normal distribution for our purposes. Because we are looking for $\langle x^2 \rangle$, Equation 13 becomes

$$\begin{aligned}\langle x^2 \rangle &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma(t)^2}} \exp\left(-\frac{x^2}{2\sigma(t)^2}\right) x^2 dx \\ &= 2 \int_0^{\infty} \frac{1}{\sqrt{2\pi\sigma(t)^2}} \exp\left(-\frac{x^2}{2\sigma(t)^2}\right) x^2 dx.\end{aligned}\quad (14)$$

The last step can be done because the x^2 term makes it an even function being symmetrically integrated about zero. Now, we make a change of variable ($x = \sigma\sqrt{2}$) to obtain, after a couple steps of algebra,

$$\langle x^2 \rangle = \frac{4\sigma^2}{\sqrt{\pi}} \int_0^{\infty} x^2 \exp(-x^2) dx. \quad (15)$$

Integrating the above equation by parts yields

$$\langle x^2 \rangle = \sigma^2 \quad (16)$$

Typically, we call this equation the variance.

1.3 Cluster Growth with a DLA Model

The random walk idea could serve as a great tool to model cluster growth. One of the common cluster growth models is the diffusion limited aggregation (DLA) model, which starts with a seed particle at the origin of the system grid, and then release randomly generated starting walkers at a certain distance from the origin. The random walker is assumed to perform random walks until it reaches one of the boundary perimeter sites. If so the sites that has been hit are updated to a new cluster. And the process is repeated until the cluster grows to a desired size. Compared to another model, Eden Model of cluster growth, the DLA cluster is sparser and has a crystal shape with many branches reaching out from the central cluster.

Analogue to a 2-D disk whose mass is $m(r) = \sigma\pi r^2$ with σ being the mass per unit area, and a linear stick whose mass is $m(r) = \lambda r$ with λ the linear mass density, a cluster has its the fractal or effective dimension which is defined to be d_f , where

$$m(r) = C \cdot r^{d_f} \quad (17)$$

thus

$$\log(m) = C + d_f \log(r) \quad (18)$$

For a cluster, assuming each walker is given a unit mass, then the mass m is the number of walkers and r is the radius of the cluster, with a proportionality constant C . If counting the particles within radius r around the origin seed position, we could find the relation:

$$\log(m) \sim d_f \cdot \log(r) \quad (19)$$

,which will be simulated using diffusion limited aggregation method later.

2 Computations

2.1 Random Walks in 2D

To code the basic 2D walk, we assume that the probability to step in either direction left, right, up or down is the same. Then, for every step, we generate a random number. This is

equivalent to drawing from Uniform (0,1). We assign every move to one of the segments of .25. If the random numbers falls within that value, we move in the assigned direction. The evidence for a successfully coded 2D walker program is given in Figure 1 and 2.

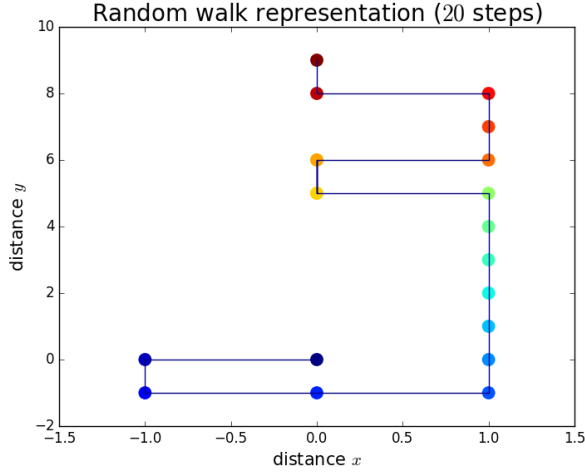


Figure 1: Random walk with 20 steps

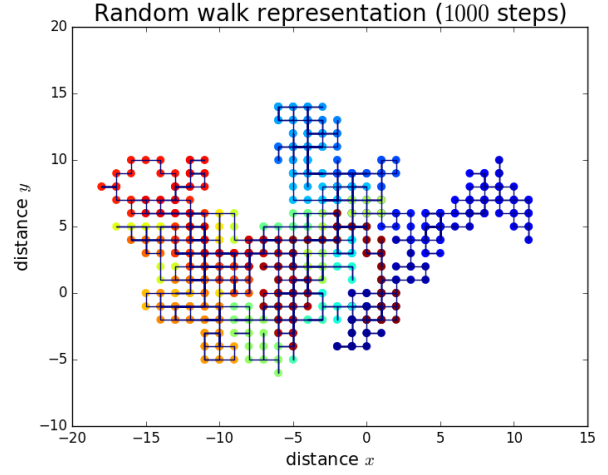


Figure 2: Random walk with 1000 steps

Now, we can simulate 10^4 random walks and compare the theoretical result of theory section with the simulations.

First, we see that indeed $\langle x \rangle \approx 0$, since the curve for $\langle x \rangle$ oscillates around 0, as shown in Figure 3

We have also shown before that

$$\langle x^2 \rangle = \sum_{i=1}^N \langle x_i^2 \rangle = \sum_{i=1}^N 0.5 = 0.5N \quad (20)$$

When we simulate $\langle x^2 \rangle$ we indeed get the straight line with the slope equal to 0.5 (see Figure 4). The mean distance squared ($\langle d^2 \rangle$) graph also looks as expected, as a straight line with slope equal to 1 ((see Figure 5). Therefore, if $\langle d^2 \rangle = 4Dt$, and steps are once at a time, diffusion constant is approximately 0.25.

By observing the data, and noting that the $\log(\langle d \rangle) vs \log(n)$ is linear, we can get a full form for $\langle d \rangle$ as a function of n .

$$\log(\langle d \rangle) = \log \langle a \rangle + b \log(n) \quad (21)$$

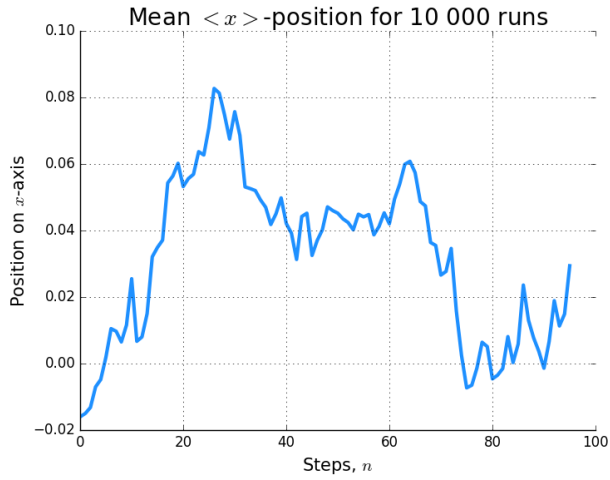
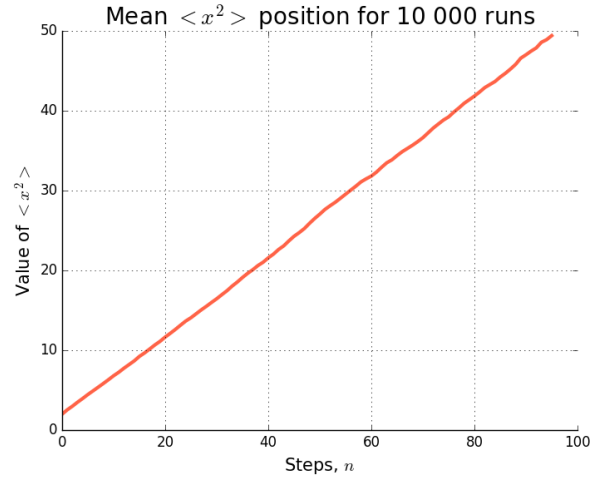
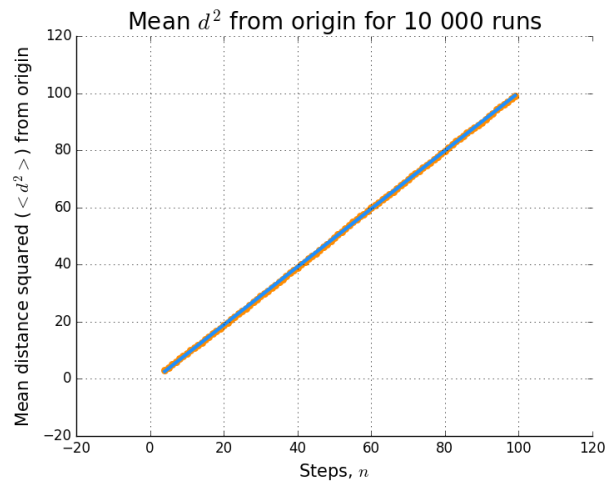
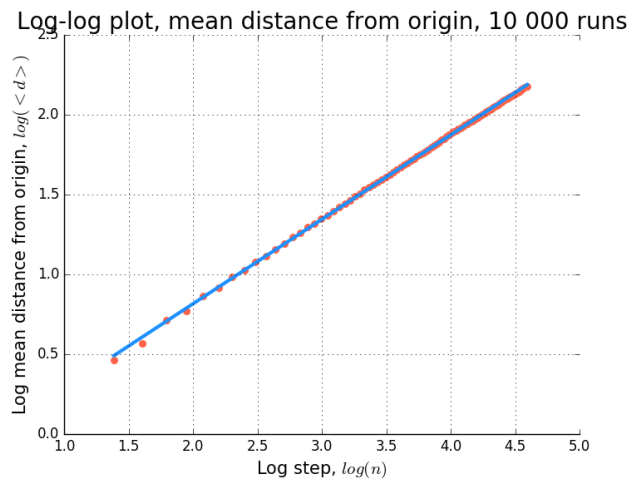
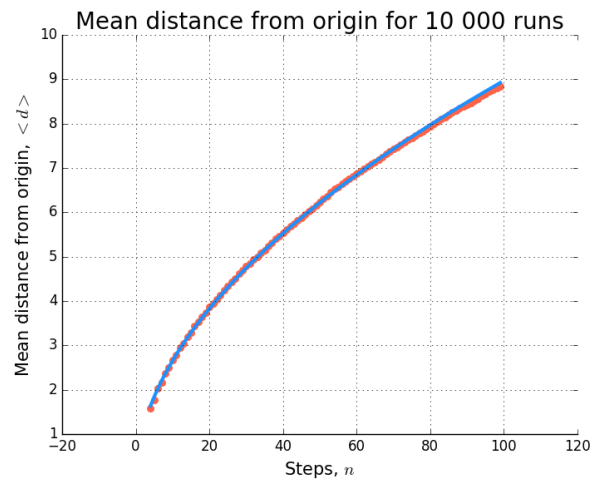
$$\langle d \rangle = an^b \quad (22)$$

First, we plot $\log(\langle d \rangle) vs \log(n)$, find a fit to the log curve and use the parameters to get the desired relation. In this case the curve was:

$$\langle d \rangle = 0.787n^{0.53} \quad (23)$$

The power of n , which represents time in this case, is close to the expected 0.5. The plots with fits can be seen on Figures 6 and 7.

An interesting addition to the random walk average in x direction would look at the graph (see Figure 8):

Figure 3: $\langle x \rangle$ for random walk, 100 stepsFigure 4: $\langle x^2 \rangle$ for random walk, 100 stepsFigure 5: $\langle d^2 \rangle$ for random walk, 100 stepsFigure 6: $\log(\langle d \rangle)$ vs $\log(n)$ Figure 7: Graph for $\langle d \rangle$ with a fit

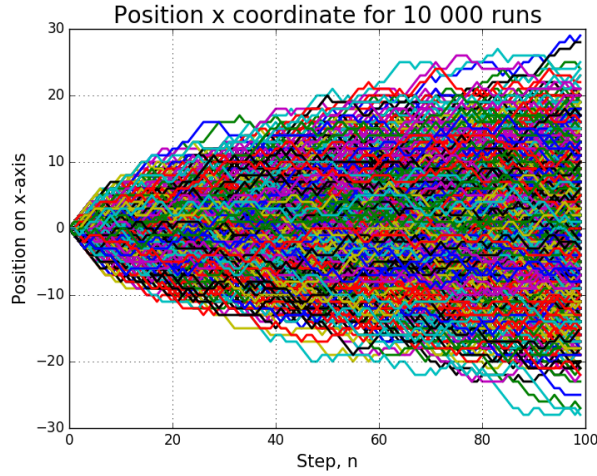


Figure 8: x position for many runs of random walk

2.2 Diffusion of a Box Density Distribution

Using the arguments in Section 1.2, we solve the 1D Diffusion Equation over a period of time. Six different snapshots in time were then fit against Equation 12 to show a box shaped density will eventually diffuse into a Gaussian normal distribution.

It should be noted that it was important to place this box density far from the edges of the grid. Otherwise, the grid started to act as a makeshift boundary condition that, over time, made the distribution look less Gaussian. The initial distribution of our 1D diffusion solver is shown in Figure 9.

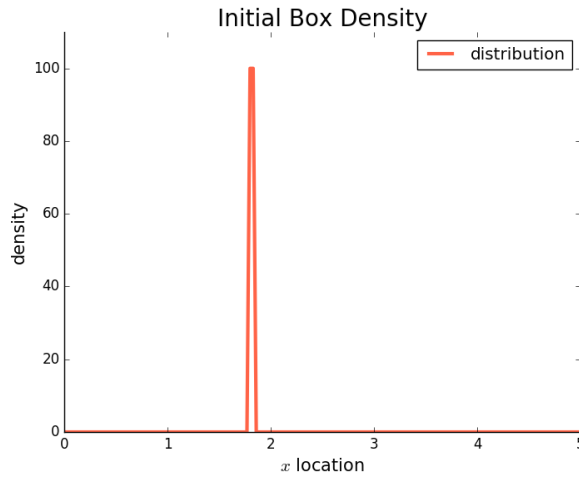


Figure 9: initial box-shaped density

In Figures 10-15, we took snapshots of the 1D density equation and fit a Gaussian to the solution. A value of σ was then extracted and compared to the analytical solution, $\sigma = \sqrt{2Dt}$. As can be seen, the fit worked exceptionally well for all time snapshots. The edge effects start to ruin our σ fit in Figure 15.

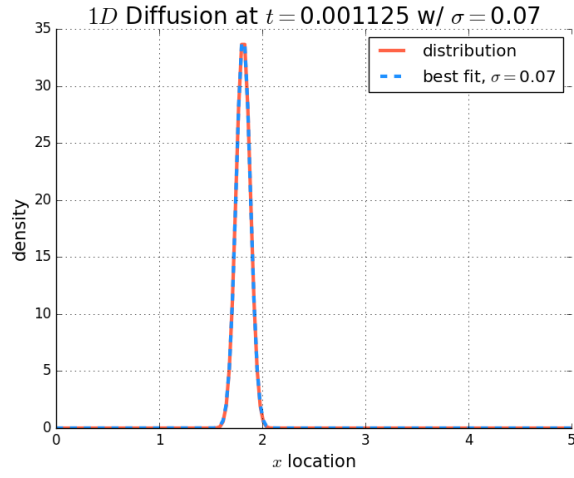


Figure 10

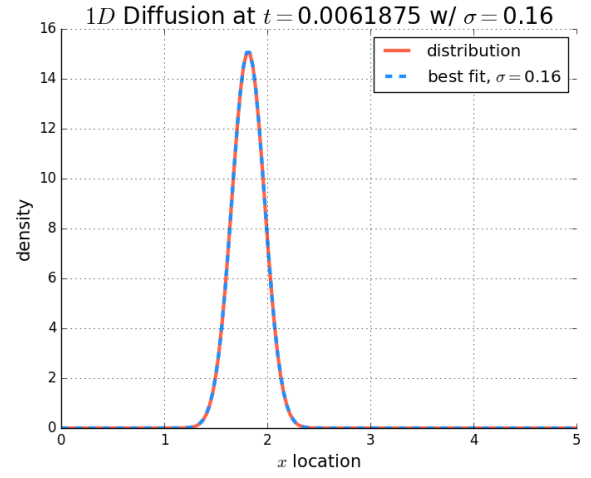


Figure 11

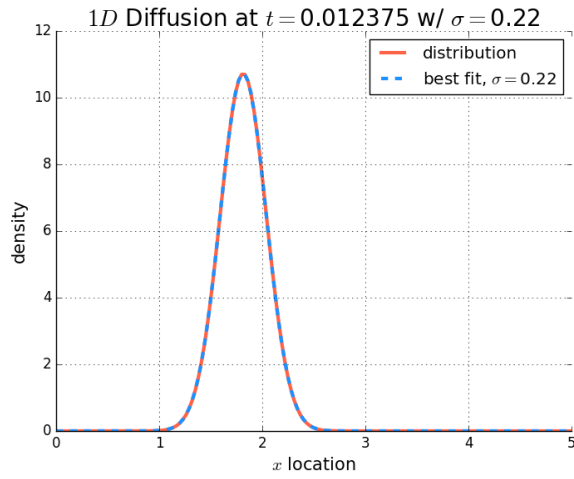


Figure 12

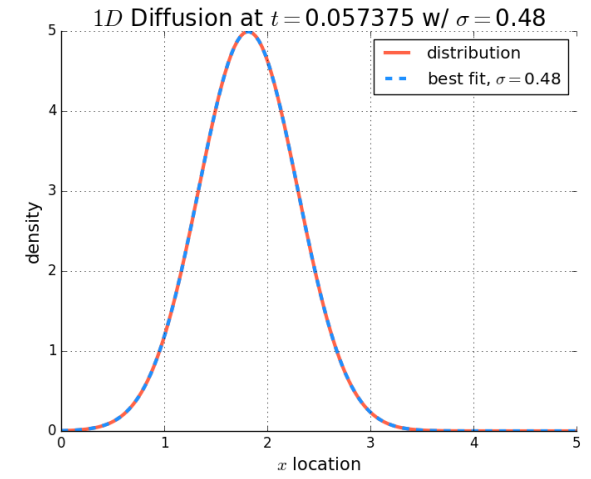


Figure 13

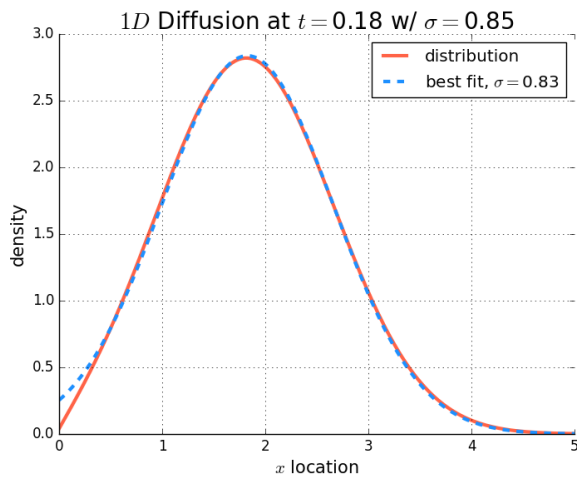


Figure 14

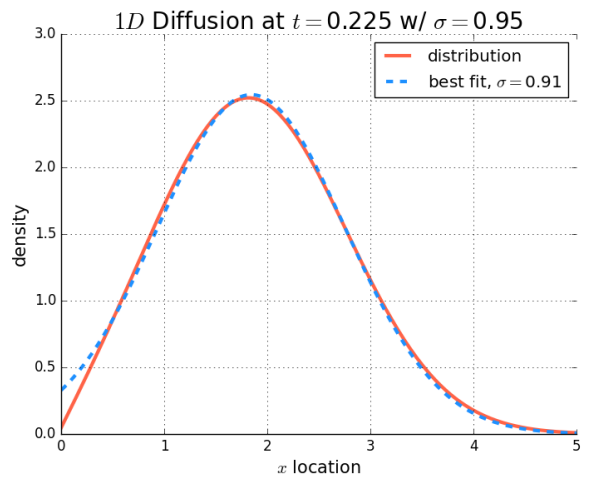


Figure 15

2.3 Growing Clusters and Extracting Fractal Dimensions

First, we simulate the DLA cluster by placing a seed in the middle. Then, the random walker is generated at the distance equal to the radius from the seed (center). This is done using a random number generator: we draw from (0,1) using `random.random()` and then multiply by 2π to get the angle in radians. Then we extract the coordinates in the matrix.

To enhance the speed of the code we use a matrix in the form (X, Y, label). Label corresponds to either out of bounds of radius (=2), occupied by walker (=1) or vacant (=0).

Assumptions of our model: walker eventually finds either his 'friend', or in other words the other walker, or it wanders off to the edge of the square where it gets 'lost'.

Please reference Figures 16 to 21 for the DLA structures for varying radius (12, 32, 52, 62, 72, 102 respectively).

By plotting the figure of $\log(m)$ versus $\log(r)$ simulated by the DLA cluster, the linear relation between these two parameters is obvious, with the slope of the straight line giving the fractal dimension of the generated cluster. And the proportionality constant is implied by the intercept of line with y axis. We repeat the generation of cluster for 10 times to get the average fractal dimension with the given parameter so that the accuracy of d_f is added.

First, note that the trend line is very similar to the one in the textbook, and provides a good fit (Figure 22. The parameter of the slope shown is 2.06

After 10 iterations, the resulting fractal dimensions were:

2.067, 1.653, 1.877, 1.833, 1.607, 1.832, 1.782, 1.712, 1.843, 1.862

The mean value is then 1.807. This is between 1 (line) and 2 (solid disc) and therefore close to the expected value for the dimensionality.

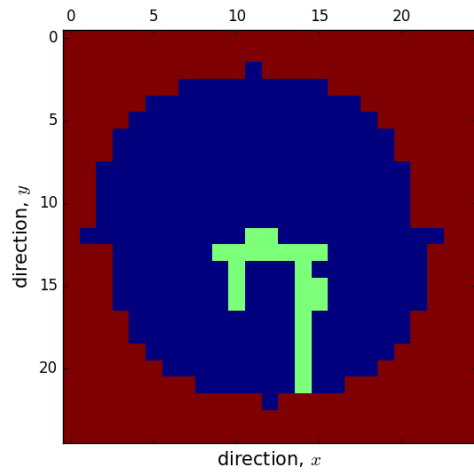


Figure 16

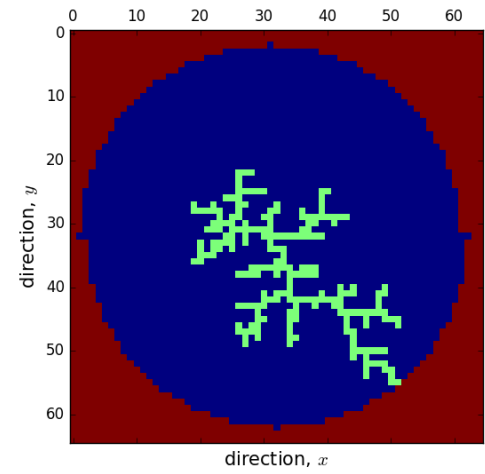


Figure 17

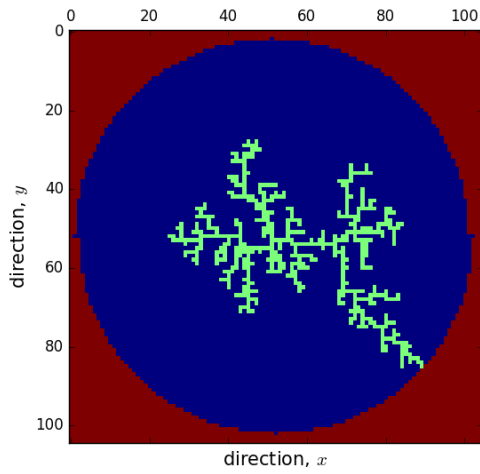


Figure 18

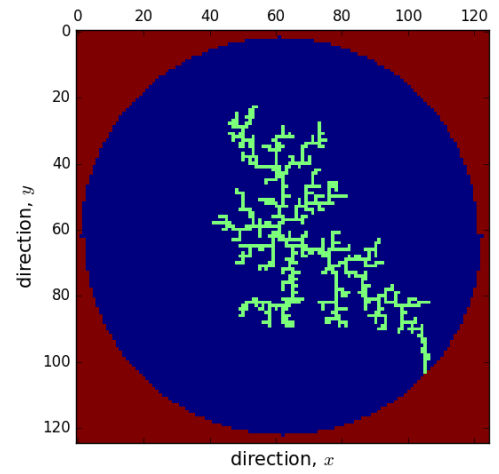


Figure 19

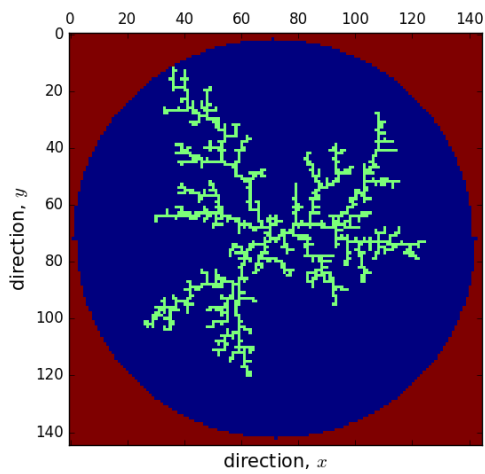


Figure 20

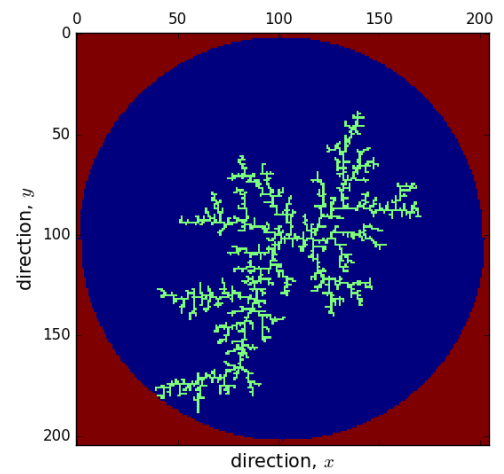


Figure 21

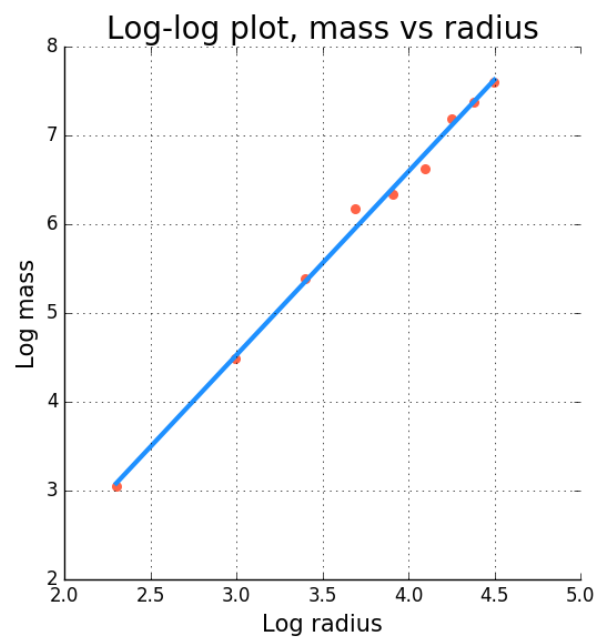


Figure 22: Log radius vs log mass for the DLA cluster