**Modelling Diffusion Limited Aggregation (DLA)**

Noah Lichtenberg and Ruohong Gao, May 3 2021

1. Analysis of the problem:

In this report, we aim to model Diffusion Limited Aggregation (DLA). DLA is a process that can model various phenomena occurring in physics and chemistry as well as our everyday lives, such as coral growth, crystal growth, and the path taken by lightning[[1]](#footnote-1). The result of our simulation will be an animation of the creation of a fractal-like figure which models DLA.

2. Model Design:

Due to the nature of modelling DLA, our simulation will require nearly no mathematical equations. This is largely due to the fact that DLA is primarily a stochastic process, as underlying processes such as Brownian motion indicates that deterministic equations such as Newton’s Laws do not apply. Rather, we will use repeated random walks of individual particles confined within a lattice.

We will first create a square lattice of variable size. For our simulation, we picked a lattice of 100 units long per side. We create a particle in the center of our lattice, and set that as our first stationary particle. After that, we will release a particle from the boundary of the lattice and let it perform a random walk until it comes into contact with another stationary particle. When this happens, the second particle turns into a stationary particle, and we repeat this process until we obtain a certain number of stationary particles on our lattice. For our simulation, we picked 1000 stationary particles.

We should also note that there are typically two ways of approaching boundary conditions for this DLA simulation, with particles that step outside the boundary either reflected or sent to the other side of the lattice. For our simulation, particles will be reflected if they reach the boundary of the lattice.

3. Model Solution:

In order to simulate DLA, we have implemented a simple random walk algorithm. Essentially, the algorithm simulates a lattice by using a two dimensional array, with the values of the array corresponding to whether or not there exists a stationary particle on each grid.

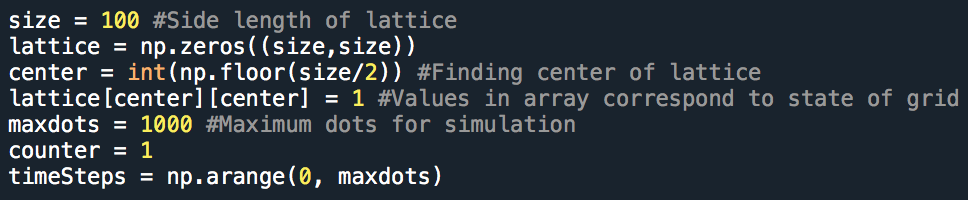


Figure 1: Initialization of lattice

After that, a while loop is implemented to execute while the simulation is incomplete, i.e. the amount of stationary particles on the lattice is less than the desired amount. During each execution of the loop, a random integer between 1 and 4 (inclusive) which determines the direction of the particle’s next step (left, right, up, down) is chosen repeatedly until the particle is in contact another stationary point – at which point the particle becomes stationary and the next particle is released.

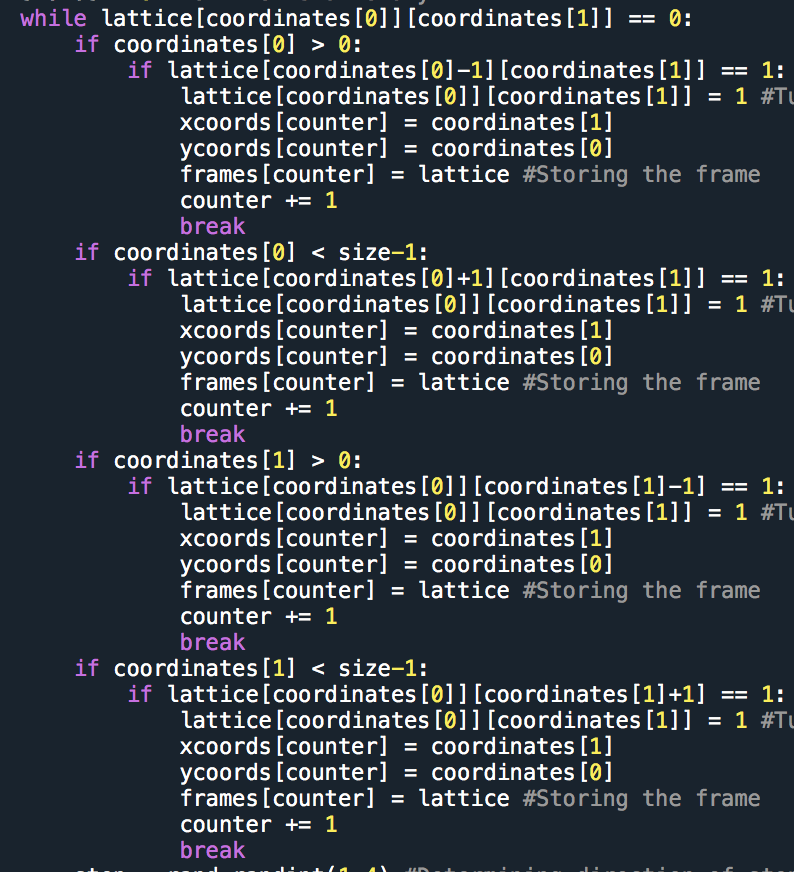


Figure : Checking to see if particle comes into contact with another particle

To simulate the boundary conditions, we set it such that whenever the particle would exit the boundaries of the lattice on the next step, its coordinates are changed right before the step such that the particle doesn’t cross the boundary. For example, if a particle’s y coordinate is at 0 and the next step is downwards, the algorithm would set the y coordinate to 1 right before executing the step, such that it would be as if the coordinate of the particle never changed. This effectively creates reflecting boundary conditions, as any particle that is about to step out of bounds would return to its original position.

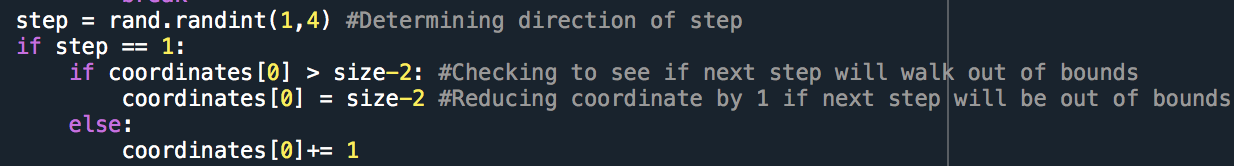


Figure : Reflecting boundary conditions

After each particle completes its random walk and becomes a stationary particle, a for loop was implemented to check for the x coordinates of the leftmost and rightmost particle. By finding the difference, we can effectively find the maximum horizontal width of the DLA cluster at any point during the simulation.

4. Results, Verification, and Conclusion

The final product of our simulation would be an animation showing the growth of the DLA cluster as well as a plot of the maximum width of the cluster as it varies with time (in steps/amount of stationary particles). For our animation, an extra stationary particle is added per frame of the animation.



Figure 4: Frame 50 of DLA cluster

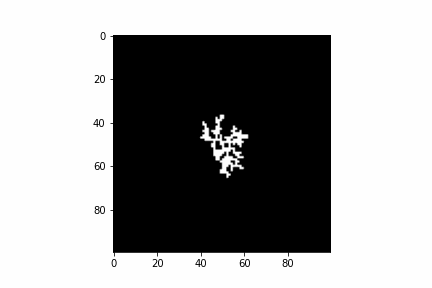


Figure 5: Frame 200 of DLA cluster



Figure 6: Frame 500 of DLA cluster



Figure 7: Frame 1000 of DLA cluster

As seen in various steps along the growth of the DLA cluster, the fractal like pattern that appears in DLA clusters become more apparent as the cluster grows in size. This can be attributed to the general fact that there will be less variance in the data as the data set grows in size, meaning that the data set will converge towards the expected result.

It is also valuable to observe the growth of the width of the DLA cluster as time progresses:

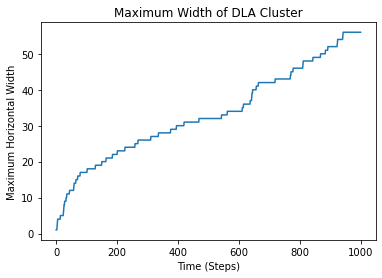


Figure 8: Maximum width of DLA cluster as more particles are added

From the graph, one may guess that the width of the DLA cluster grows with time logarithmically or as a square root function from just observation. However, we believe that no confident conclusion can be made from this data alone. Due to long computational times to generate larger lattices, it’s relatively difficult to collect meaningful data for much larger DLA clusters.

However, if we make some simplifying assumptions regarding the growth of the DLA cluster, we suspect that the width may vary as a square root function of time. Assuming that the cluster grows in all directions at an equal rate, the cluster very roughly forms a circle. Considering that the diameter of the circle would be the width of the cluster and the amount of particles would be proportional to the area, we see that the amount of particles would be proportional to the square of the diameter, or the width of the DLA.

1. http://paulbourke.net/fractals/dla/ [↑](#footnote-ref-1)