PROCEEDINGS OF SPIE

SPIEDigitalLibrary.org/conference-proceedings-of-spie

Research on fractal dimension and growth control of diffusion limited aggregation

Minhui Jiang, Zhouqi Zhong

Minhui Jiang, Zhouqi Zhong, "Research on fractal dimension and growth control of diffusion limited aggregation," Proc. SPIE 12163, International Conference on Statistics, Applied Mathematics, and Computing Science (CSAMCS 2021), 1216330 (22 April 2022); doi: 10.1117/12.2628155

Event: International Conference on Statistics, Applied Mathematics, and Computing Science (CSAMCS 2021), 2021, Nanjing, China

Research on fractal dimension and growth control of diffusion limited aggregation

Minhui Jiang[1, a, *, †], Zhouqi Zhong[2, b, *, †]

[1] School of Mathematics, Southeast University, Nanjing, Jiangsu 211189, China;

[2]School of Mathematics, Sun Yat-sen University, Guangzhou, Guangdong 510275, China.

*Corresponding author's e-mail: [a]213180239@seu.edu.cn; [b]zhongzhq25@mail2.sysu.edu.cn

[†]These authors contributed equally.

Abstract.

The fractal theory is a significant research subject in which diffusion limited aggregation (DLA) is a valuable model. DLA is a simple model which can reflect a wide range of natural phenomena. Through simple kinematic and dynamic processes, it can produce a self-similar fractal structure with scale invariance. The growth process is dynamic, far from equilibrium, but the cluster structure has a stable and definite fractal dimension. In this paper, we study the fractal dimension of DLA, try to control the growth of DLA, and simulate the growth process that may occur in some natural environments under some factors. We use simple Java code to build a DLA growth model. We use a fixed number of particles, and the DLA of particle aggregation, to touch the set boundary as the boundary conditions, respectively. We simulate the process of particle random motion aggregation into DLA clusters, and generate images to observe the results. We use the density method to compute the fractal dimension of DLA. To control the growth of DLA, we modify some parameters of the random motion process of particles, including the step size, the probability of moving in different directions, the initial generation region of particles, and so on. By analyzing the fractal dimension, we find that the fractal dimension of DLA can be greatly affected by changing the area of particles. In the study of the growth control of DLA, we mainly study the influence of changing the probability of particles moving in different directions on the growth of DLA (some specific factors can cause this effect in the natural environment). Through these changes, we can make DLA grow in a fixed direction. We can analyze the growth of fractal systems in the real situation through these straightforward simulations which are computer programs based on DLA principles.

Keywords: Diffusion Limited Aggregation   Fractal Theory   JAVA   Fractal Dimension

1. Introduction

Diffusion-limited aggregation (DLA) was first introduced in statistical physics and a brief introduction can be found in [1]. A more formal introduction to both practical intuitions and theoretical foundations of DLA can be found in [2]. In general, DLA is an abstract model for simulating the formation of low-density objects through the aggregation of solid particles.

The fractal dimension is a ratio providing a statistical index of complexity comparing how the detail in a pattern changes with the measuring scale. Usually, the fractal dimension of a geometrical object is greater than its topological dimension (which can only be positive integers or zero if it is disconnected) [3]. There are already some research papers on computing the fractal dimensions of DLA in different ways. For example, in ref. [1], the author used density correlation functions in DLA simulations to compute the fractal dimension. In ref. [4], the author computed the fractal dimension of the trunk of a DLA cluster by the mean-field approximation (the Flory method), which is a theoretical tool in statistical physics. The method in ref.[1] is empirical by launching the computer simulations of DLA, and the method in ref.[4] is purely theoretical.

In our research, we follow the empirical paradigm and compute the fractal dimension of DLA by launching the computer simulation of DLA many times and using the counting method in each simulation. In this way, we count the pixels occupied by the aggregation process and compare it by the size of the whole square. Then this ratio can be used to compute the fractal dimension of DLA. To make our result free from randomness, we compute the ratio many times by launching many simulations and take the average.

There are some benefits to our method of computing the fractal dimensions of DLA. First, the traditional density correlation function method in ref.[1] needs us to choose a center point for computing. Since the DLA is self-similar, theoretically, every point in DLA can be a center. Also, choosing different centers would cause different results, which would make the

International Conference on Statistics, Applied Mathematics, and Computing Science (CSAMCS 2021), edited by Hong-Ming Yin, Ke Chen, Romeo Meštrović, Teresa A. Oliveira, Nan Lin, Proc. of SPIE Vol. 12163, 1216330 · © 2022 SPIE · 0277-786X · doi: 10.1117/12.2628155

result less convincing. In our method, we directly use all the data points in a simulation and thus avoid choosing a center point. Second, our method is easier for other scholars to understand and replicate. In our method, the most difficult step is to write codes to launch the simulation of DLA. Unlike the theoretical method, we do not require a deep understanding of fractal geometry or statistical physics [2].

In our research, we mainly use the Java language to program. The java version is 16.0.1, which can be downloaded from the official website. To show the DLA process visually, we use Picture.java downloaded from Princeton University's website . This is a Java class that can be used to generate pictures on the screen. Every time the DLA process adds a pixel, we use Picture.java to generate a picture. The pictures we generated in order would form an animation of the DLA process.

In our algorithm to simulate the DLA process, we define a big $(2N+1)\times(2N+1)$ Boolean array A to record the DLA process, where N is an integer parameter. We set all elements of A to be false except the very center element $A[N][N]$, and $A[N][N]$ is set to be true. Next, we define a while loop to generate seeds of DLA on the boundary of A, the stop condition for this loop is when some element on the "boundary" of A is set to true. In each loop, one seed is generated on the four boundaries of A randomly, which means that we first choose a boundary of A with a probability of 1/4 each, then we choose a position on this boundary that follows uniform distribution at each place. Then the seed is free to move upward, downward, leftward, rightward until it reaches a point adjacent to a pixel occupied by the DLA process, or it reaches the boundary and becomes assimilated by the boundary. Finally, this point is set to true, which means that the DLA process now occupies it.

It is worth noting that our model of the DLA process is not isotropic in that the direction of growth is not uniformly distributed on a circle. Because we launch seeds on the boundary uniformly instead of following a Gaussian distribution on the boundary centered at the middle point, the DLA process would be more concentrated on the diagonal of the square. The benefits of our model include that the realization is simpler because the array in Java is by nature a square lattice, which corresponds to our method well. Also, we do not need to adapt the Gaussian distribution, which is continuous and extends to infinity, into a discrete and finite approximation on the boundary. For an algorithm to generate isotropic models, the reader can check [5], and this is also a detailed analysis of algorithms to generate the DLA process. Another more realistic modeling algorithm of DLA can be found in ref.[6], where the particles involved in the DLA process are not modelled as pixel points but balls that take up some space.

We repeat our simulation 20 times, and each time N takes increasing values from 200 to 390 with an increment of 10. Then we count the points taken by the DLA process in the array A, and the number of points applies:

$$count \propto N^D \tag{1}$$

Here D is the fractal dimension. Then we use linear regression to calibrate D. The basic knowledge of linear regression can be found in ref.[7]. Our result is $D = 1.6589$.

Another part of our research is to control the growth of the DLA process. First, we control the total number of seeds launched, and the DLA stops on a smaller scale. Second, we change the probabilities of moving into 4 different directions of the seed we launch. The graphs of our simulations show that if seeds are more likely to move upward than downward, then the DLA would accumulate downward. Similarly, the DLA would accumulate leftward if the seed is more likely to move rightward than leftward. This is because if seeds tend to move upward, then the seeds that form DLA are less likely to be those released from the upper boundary, since most of those seeds released from the upper boundary would move upward and become assimilated to this boundary. When most accumulating particles are seeds released from lower boundary, and these seeds are coming upward, the DLA tree is growing downward visually. Previous research has been done on the relation between the probability of directions of moving particles and DLA growth by L. A.Turkevich and H. Scher (1986) [8]. Also, research has been done on changing patterns of diffusion-limited aggregation by variation of parameters by T. Vicsek (1984).[9]

Some research has been done on controlling the growth of DLA. Barra et al. (2001) [10] constructed a one-parameter family of fractal growth patterns that will lead to a continuously varying fractal dimension. Wu, H. et. al (2013) considered the relationship between the free particles' radii and the fractal dimension for more realistic modeling [11].

2. Mechanism

In this section, we will introduce some methods we used in the research. According to the specific types of work, we divide these parts into three parts: DLA generation simulation, DLA fractal dimension calculation and DLA growth direction control, and introduce the methods we use, respectively. In the DLA generation simulation, we introduce some principles and code methods of DLA generation in Java. In the part of fractal dimension calculation of DLA, we introduce the principle and process of fractal dimension calculation using the density method. In the part of growth control of DLA, we introduce the principle and implementation means of changing the growth direction of DLA.

2.1 DLA generation simulation

We use Java to realize DLA growth simulation. The generation code of DLA includes the initial particle delivery, the particle's random motion, and whether the particle collides with the generated DLA and whether the particle delivery has reached our initial boundary conditions. In our code, particles are delivered in two ways. The first is that the particles are randomly generated from the boundary of the set plane, and the second is that the particles are randomly generated from the delimited plane. In the random motion of a particle, we make the particle move up, down, left, and right with equal probability each time, one unit length. When DLA particles grow on the lattice in the eight directions of top, bottom, left, right, top left, top right, bottom left and bottom right, the particle stops moving and becomes a part of DLA. We use Boolean variables to check this and establish a matrix with the same size as the plane composed of the set lattice, using 0 and 1 to indicate whether particles are at the position. When the particle moves above the boundary we set, we decide that the particle is far away from the DLA we generated and jumps out of the cycle of the particle. We set two conditions to stop the growth of DLA. The first is to stop the growth of DLA when it touches our set boundary. The second is to fix the number of particles. In fact, only a part of the particles grows into the DLA, so the specific number of particles in the second condition needs to be determined by many attempts.

2.2 Calculation of fractal dimension

We use the density method to calculate the fractal dimension of DLA using Java. We do 20 simulations, using different plane sizes, and calculate the number of particles included in DLA when DLA touches the boundary and stops launching particles (using the generated Boolean variable to count the number of particles). Each simulation writes the results into the data set to get the data of 20 simulations. Then, the data are regressed, and the fractal dimension is calculated. The function gives the relationship between the fractal dimension and the number of points

$$N(R) \propto R^D \tag{2}$$

By generating the histogram of r and m, we can observe the approximate regression relationship between the data. We take the logarithm of m (the number of particles) and regenerate it into the histogram of r and log (m), and we can observe the approximate linear regression relationship. At this time, we remove some outliers and fit them with the least square method to get the fractal dimension.

2.3 Changing the growth direction of DLA

Firstly, we try to change the growth direction of DLA by changing part of the code of random motion of particles. We increase the probability of particles moving in a certain direction and reduce the probability of particles moving in the opposite direction, expecting to get a DLA growing in the opposite direction. In addition, we try to change the step size of particles moving in different directions, which can achieve the same effect as changing the probability of particles moving in different directions. By doing this, we can make DLA grow in the roughly horizontal direction or vertical direction.

3. Result

In this section, we will lead to the results of our research. Through the appealed research method, we will first generate two models of DLA. Wes will use the result from the DLA of particle aggregation to touch the set boundary as the boundary conditions and will calculate their fractal dimensions through nonlinear regression. Second, we will further control the direction of the growth and the size of the DLA. And observe the result. Also, to control the growth of DLA, we modify some parameters of the random motion process of particles, including the step size, the probability of moving in different directions.

3.1. Programming to generate DLA

Figure 1. DLA of Model 1.

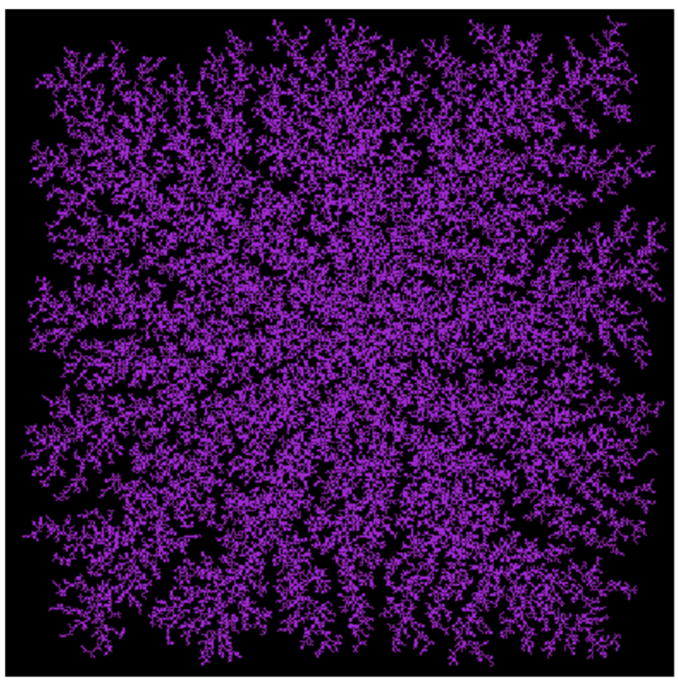

Figure 2. DLA of Model 2.

We use Java to generate the DLA of Model 1 and 2. The results of DLAs are different. As Figure 1 shows, the branches of the first model mostly go left, right, up or down. There are many sponsors in the first model because the particles generate from a boundary. So, it is likely to reach the outer boundary of the tree. For the second model, according to Figure 2, particles can come out of anywhere, so they are more likely to fill the space.

3.2 Computing the Fractal Dimension of DLA

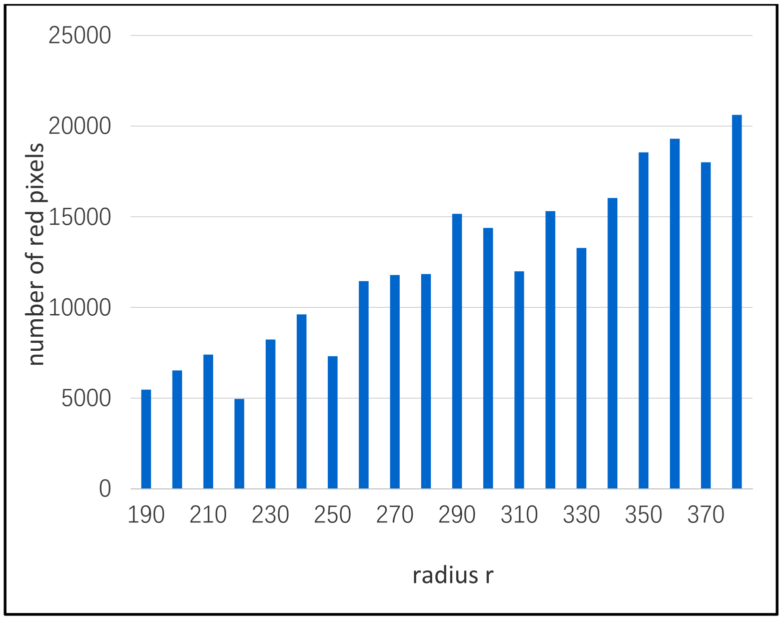

Figure 3. 20 Simulations generated from Model 1.

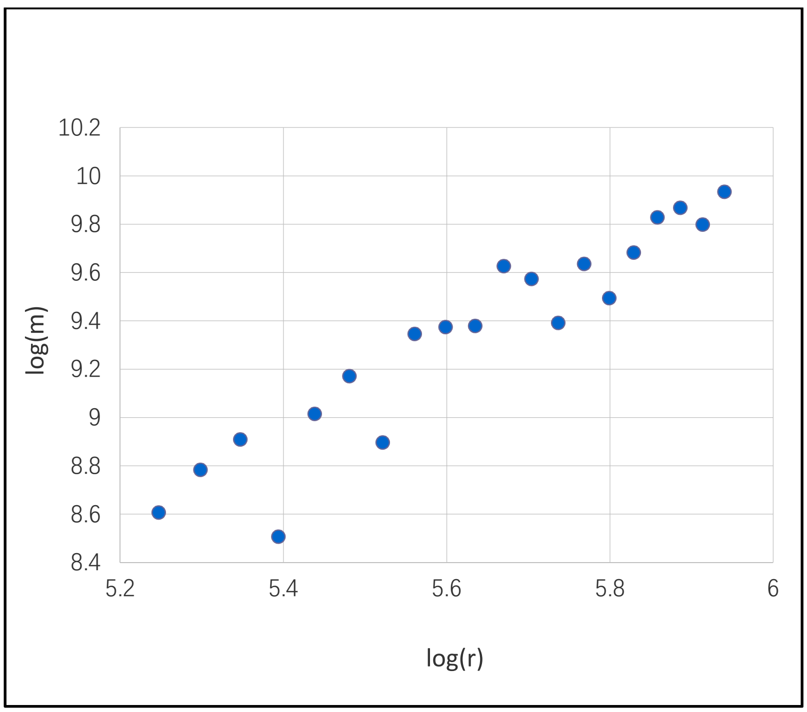

Figure 4. Nonlinear regression from 20 simulations of Model 1

We generate the histogram of simulations from both models. As shown in Figure 3, there are 20 simulations from Model 1. When the particle number is fairly small, the aggregates are fairly small. The aggregate gradually grows with rising particle numbers. The fractal dimension increases rapidly as the particle number increases from 250 to 350 and changes little as the particle number rises further. Then we use the nonlinear regression to compute the relationship between log(m) and log(r), as shown in Figure 4. The fractal dimension is about 1.65878. As shown in Figure 5, there are 15 simulations from Model 2. We can see that the aggregate gradually grows with rising particle numbers. Then, as shown in Figure 6, we use nonlinear regression to compute the relationship between log(m) and log(r). The fractal dimension is about 1.99088.

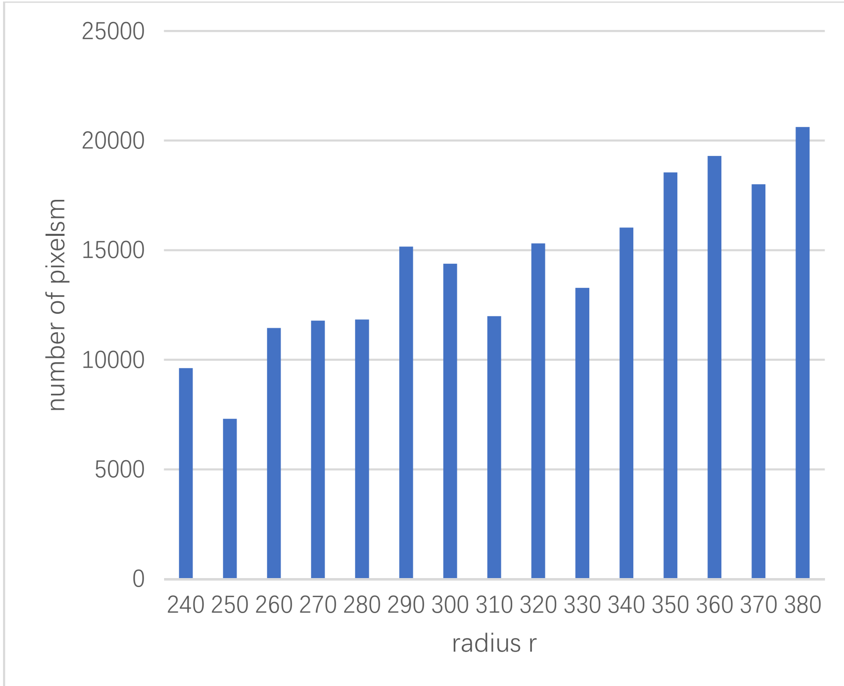

Figure 5. 15 Simulations generated from Model 2.

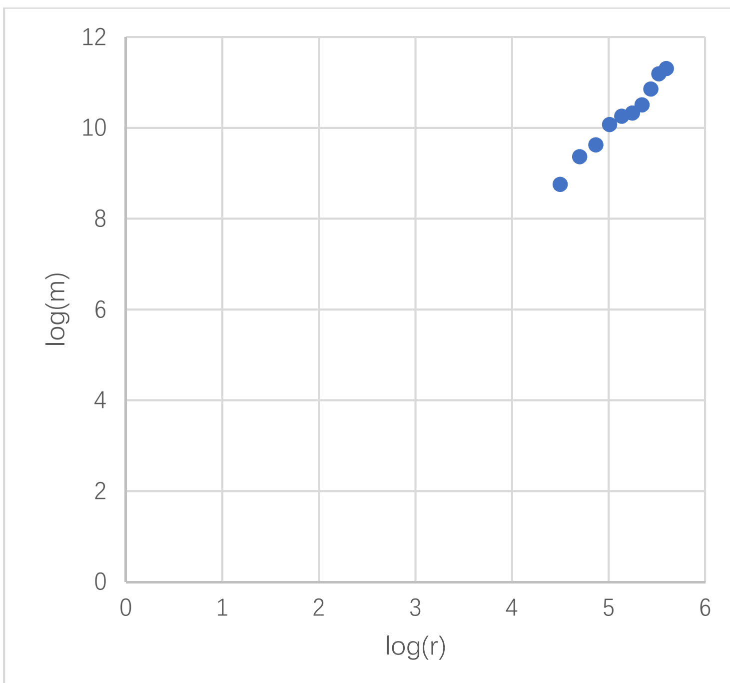

Figure 6. Nonlinear regression from 15 simulations of Model 2

The result of Model 1 in Figure 3 is a clear indicator since it shows a clear positive relationship between number of particles and radius, which is an important signal that our assumption might be right. However, the Figure 3 also shows that the relationship is not a simple linear relation, because the units vertical axis is exponentially increasing. Thus it implies that the equation (2) is true since the dimension D is greater than 1. To compute the dimension, we take natural logarithms on both sides of equation (2) , and then its become a linear relation, so we can compute the dimension D in this way (D =1.65878).

3.3 Improve the program to control the growth of DLA

Now we are trying to control the growth direction and the size of a DLA.

3.3.1 Controlling the growth direction of DLA

Figure 7. The growth direction of DLA

As shown in Figure 7, it has a bigger probability to move from the positive direction of the y-axis to the negative direction of the y-axis, and it has a bigger probability to move from the positive direction of the x-axis to the negative direction x-axis. We also change the step size of seeds in a different direction. If the seed step sides are in the larger horizontal direction than the vertical direction, DLA is more likely to grow horizontally.

3.3.2 Controlling the size of DLA

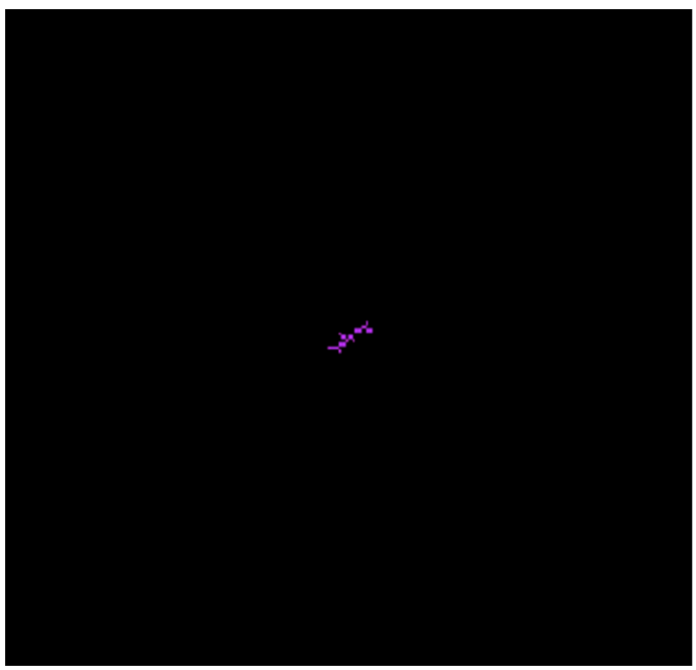

Figure 8. 10000 seeds

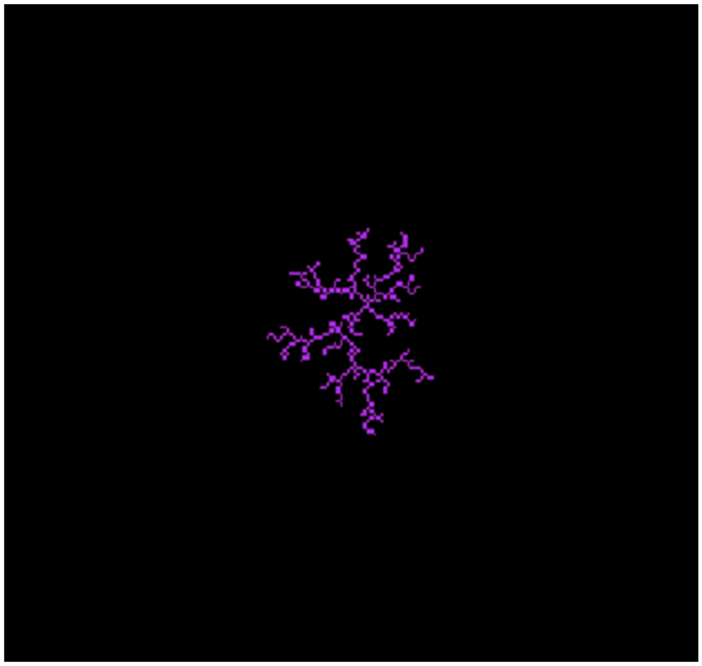

Figure 9. 1x10^5 seeds

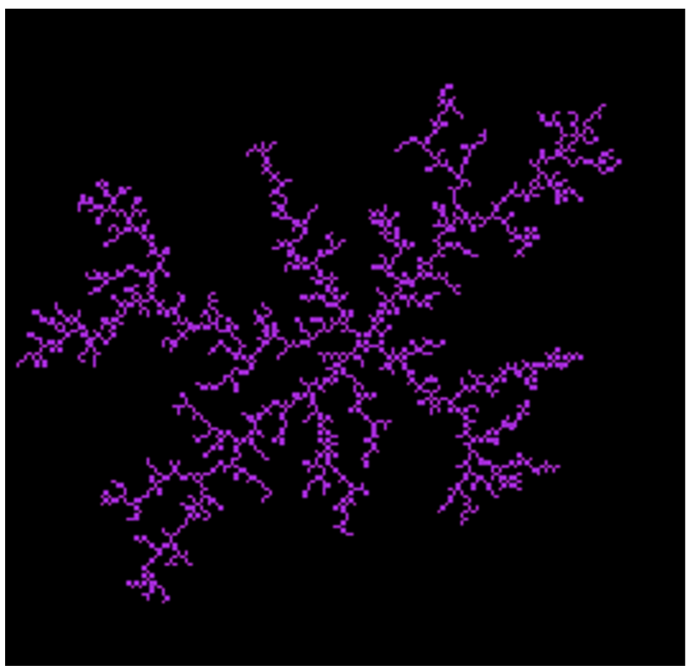

Figure 10. 1x10^6 seeds

We show DLA generated with different amounts of seeds. We get different sizes of DLA in the same area. In Figure 8, DLA launches from a small range and then gradually expands to the larger range, as shown in Figures 9 and 10.

4. Conclusion

Using simple Java, we computed the fractal dimension of DLA, controlled the growth of DLA, and simulated the growth process. First, we used a fixed number of particles and DLA aggregation to generate set boundaries as boundary conditions for two models. We got two different results of DLA. Model 1 had fewer branches and particles, while the DLA of Model 2 had more branches and particles. Second, to compute fractal dimensions of DLAs by density method, we used two histograms to list the simulations of both models. By getting the total number of points of DLA, we got different DLAs of different sizes. Through the fractal dimension analysis, we found that the fractal dimension of DLA could be greatly affected by changing the area of particles. For example, the fractal dimension of particles randomly placed on the boundary of a square is about 1.7, and the fractal dimension of particles randomly placed on a set plane is close to 2. Third, we chose to change the parameters of random motion processes to control the growth of DLA. We changed the probability of moving in different directions, the initial generation region of particles, and the size of DLA. Through these simple changes of simulation, we could make DLA grow in a fixed direction and size.

Reference

[1] T. A. Witten, L.M. Sander, Diffusion-Limited Aggregation, 1981, a Kinetic Critical Phenomenon. Physical Review Letters, 47(19), pp. 1400–1403.

[2] T. C. Halsey, 2000, Diffusion-Limited Aggregation: a Model for Pattern Formation, Physics Today 53, 11, 36.

[3] K. Falconer, 2014, Fractal Geometry: Mathematical Foundations and Applications, third ed., Wiley, West Sussex, pp. xxviii

[4] P. Pikhitsa, 1993, Fractal dimension of the trunk of a diffusion limited aggregation cluster. Physica A: Statistical Mechanics and Its Applications, 196(3), pp. 317–319.

[5] A. Offringa, 2006, Diffusion Limited Aggregation, can be found on: http://andreoffringa.org/

[6] F.L. Braga, M.S. Ribeiro, 2011, Diffusion Limited Aggregation: Algorithm optimization revisited. Computer Physics Communications, 182(8),1602–1605.

[7] G. James, D. Witten, T. Hastie and R. Tibshirani, 2013, an Introduction to Statistical Learning with Applications in R, Springer, New York, pp. 61-63.

[8] L. A. Turkevich, H. Scher, 1986, Sticking Probability Scaling in Diffusion-Limited Aggregation, Fractals in Physics, Elsevier, North Holland, pp. 223-229.

[9] T. Vicsek, 1984, Pattern Formation in Diffusion-Limited Aggregation, Physical Review Letters, 53(24).

[10] F. Barra, B. Davidovitch, A. Levermann, I. Procaccia, 2001, Laplacian Growth and Diffusion Limited Aggregation: Different Universality Classes. Physical Review Letters, 87(13).

[11] H. Wu, M. Lattuada , M. Morbidelli, 2013, Dependence of fractal dimension of DLCA clusters on size of primary particles. Advances in Colloid and Interface Science, Elsevier,195-196, 41–49.