Universiteit van Amsterdam

Report

Scientific Computing exercise set 2

May 11, 2018

Students:

Sam Ferwerda
10446982

Daan van Ingen
10345078

Teacher:
dr. J.A. Kaandorp

Course:
Scientific Computing

Code:
5284SCCO6Y

1 Introduction

In this report, we will focus our study on diffusion limited aggregation (DLA) and the Gray-Scott model. We will start by implementing the diffusion equation in our earlier obtained "Successive Over Relaxation" iteration method. In this method, we will start with a single cell object on the bottom of the domain, which will be able to spread with a certain chance. The computing time for this method is very large and therefore we are eager to use an other method based on the famous Monte Carlo method, which uses a random walk. Finally we will end this report with the Gray-Scott model and explain the discretization and how we implemented it in our code.

2 Diffusion limited aggregation

In our growth model we will take a closer look at a grid of size 256^2 and 512^2. However, due to the extreme computing time of the larger grid, the influence of variables will be investigated using the smaller grid. Furthermore, to make things a little more easy, we will start the simulation with the analytic concentration values. This means that the concentration is linear distributed from 0 at the bottom to 1 at the top of the domain. The object, which value equals 0, will start at center bottom and will have a chance to spread using the following probability distribution

$$p_g(i,j) = \frac{c_{i,j}^n}{\sum_{candidates} c_{i,j}^n}$$

where the parameter $\eta \in [0,2]$ and the candidates are all the direct neighbours of the object. We will see that if $\eta = 0$, we will get a evenly spread cluster of the object while if $\eta = 2.0$ the object will grow more like a lightning strike. Therefore we want to discuss the following theorem.

Theorem 1. *If $\eta = 0$ every candidate has equal chance of turning, $\eta = 1$ will give all candidates a equal chance based on their concentration and $\eta = 2$ will give large priority to the ones with best concentration.*

In basic calculus we learn that, no matter the base number, something to the power of 0 equals 1. Therefore if $\eta = 0$, all candidates will have a turning chance of

$$p_g(i,j) = \frac{1}{\#\text{candidates}}.$$

This means that we will see a wave around the starting point of the object. Actually, this is exactly what happens as can be seen in the following figure.

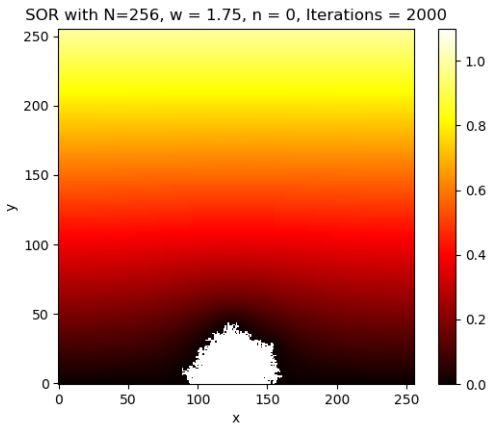

Figure 1: Growth model with equal chances

Pay attention to how nearly perfect the object has spread in every direction. The starting point of the object is clearly the middle and is expanding to all sides. We have plotted this growth with colors representing the concentration levels on the background, however, in this case that is totally unnecessary due to the fact that the probability is independent of the concentration $c_{i,j}$.

But what will happen if we set $\eta = 1.0$? The change with the previous case is that is dependent on the concentration of the candidate. This means that a higher concentration has a higher chance of turning into the object, but it is still equally distributed. This results into a beautiful tree-like object, as shown in figure 2.

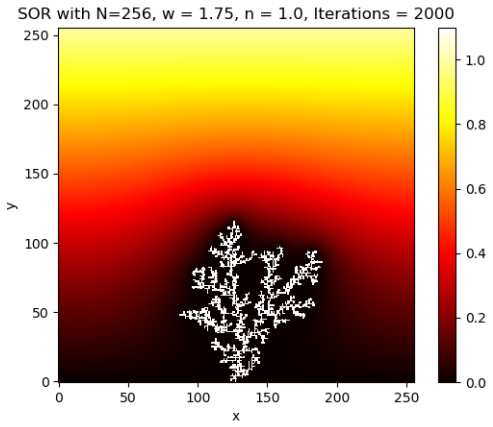

Figure 2: Growth model with chances based on concentration value $c_{i,j}$

From now on, if we increase η towards 2, we will see that only cells with a high concentration have a chance to turn. How to visualize this is with a simple example. Lets say $c_{i,j} = 0.1$ and $c_{k,l} = 0.8$, then $c_{i,j}^2 = 0.01$ and $c_{k,l}^2 = 0.64$. Note that the chance of $c_{i,j}$ turning, has just decreased with 90%, while for $c_{k,l}$ it only decreased with 20%. How this decrease is determined exactly depends therefore on the value of η. Lets look at the same simulation as figure 1 and 2, but using $\eta = 2.0$ now.

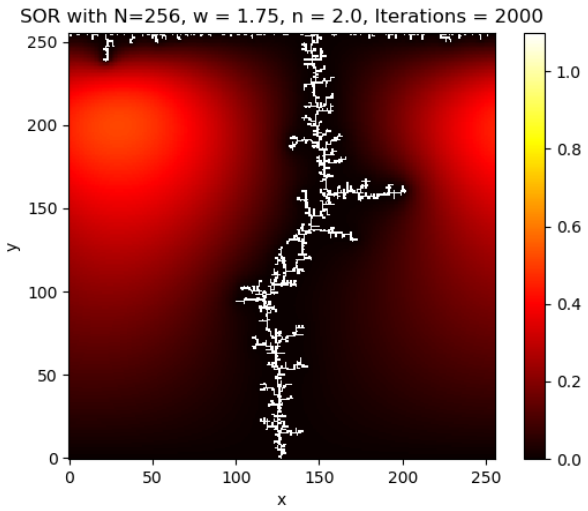

Figure 3: Chance prior to concentration levels

Due to the fact that our concentration is linearly increasing towards 1 at the top of the domain, our object will grown towards it. Always looking for the highest neighbour. If we would iterate this for even longer, the red areas at the center left and the center right will probably also be taken by the object since those places are currently the highest in concentration. Until now we have only studied the influence of η in our growth model. The grid size and the iterations are not very interesting to study since grid size will only change the visuals of the object and the iterations the size of the object. However, what does w do? We have seen that the optimal w was obtained dependent on the grid size N and using the facts from previous exercise set, we could say $w = 2.0$ is optimal for a grid of this size. To study this, we have plotted the number of cells that were part of the object after a number of iterations versus the value of w. This gave us the following result.

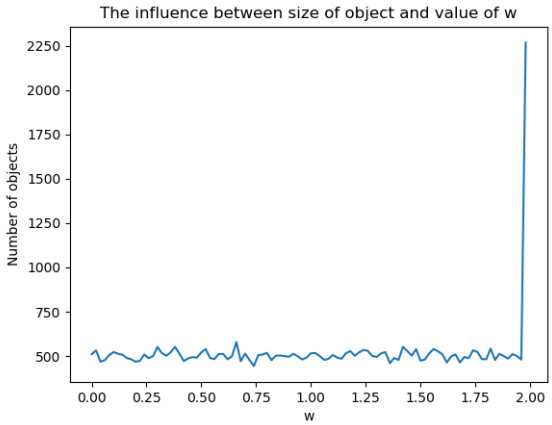

Figure 4: Influence of w on the object size

In global, the value of w does not really influence the growth of our object. However, when nearing a value of 2.0, the graph does something remarkable and jumps towards a very large number of cells. We did not know if this was meant to be, or coincidence so we did 2 simulations, one with $w = 1.7$ and the other with $w = 1.98$. The outcomes of these simulations are given below.

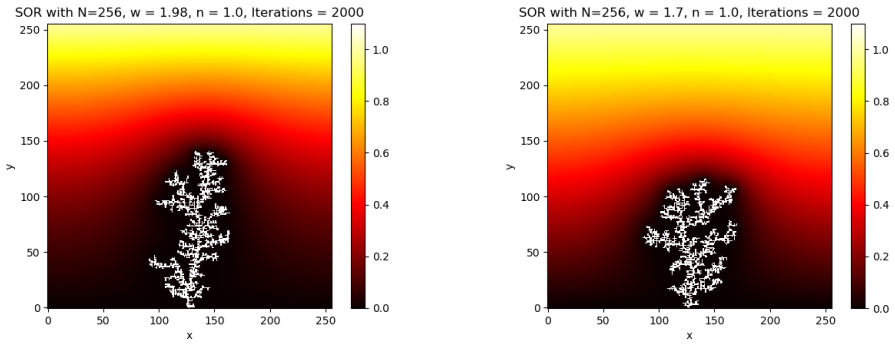

Figure 5: Two simulations with different w

The difference between these two trees, if there is any, is hardly visible and we can only assume that it has something to do with the height of the object. As we can see, the left object is getting higher, towards the higher concentrations while the right object is spreading more towards the left and right of the domain. We could not determine what the exact influence of w was, because of the high computational demand. However, it would be interesting to take a closer look at this next time.

2.1 On the larger grid

To iterate over a matrix is very computational expensive, therefore increasing your domain will rapidly increase the computing time of the growth model. We did most of our research on a $N = 256$ grid, since that is pretty big already. However, we did make some plots with a domain of $N = 512$ to check if the outcome was similar or totally different. In figure 6, we have simulated two growth models with different values for η. You can see how similar the objects are with the objects for $N = 256$. Note that, if we would like the object to reach the top, the number of iterations should be massive.

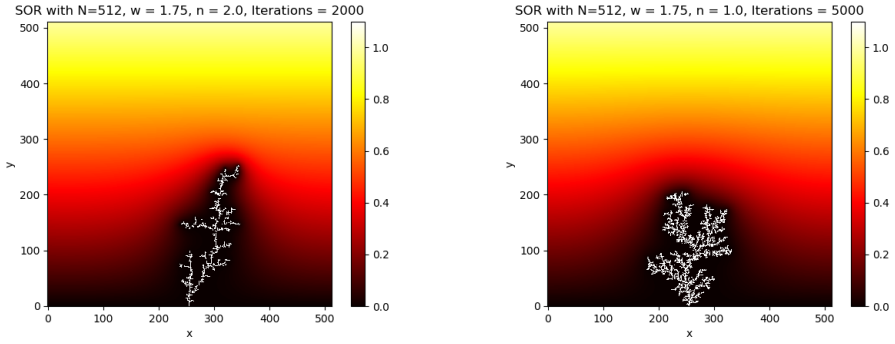

Figure 6: Growth model on a large grid

3 Monte Carlo simulation of DLA

In this section we will look at Monte Carlo simulation for the DLA by releasing random walkers in the grid, and letting them walk until they hit the cluster. The walkers are initialized at a random place at the top of the grid and walk left, right, upwards or downwards every iteration. If a walker

walks out of the system it is removed and a new walker is initialized. We initialize 256 walkers into the system.

The system has periodic boundaries in the horizontal direction. To be able to compare the results of MC with the ordinary DLA we choose the same grid size (256x256) for the MC simulation. To save computation time we initialize the grid in a steady state and only recalculate the diffusion every 10 time steps. The result of the simulation can be seen in figure 7.

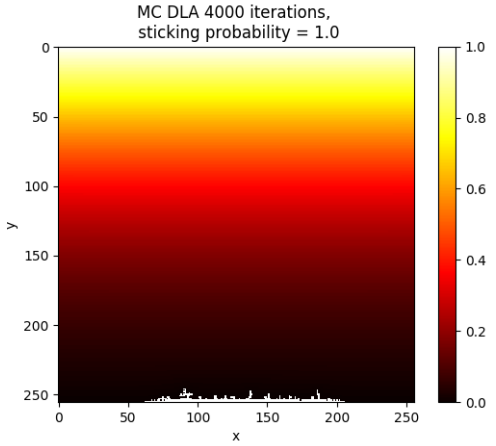

Figure 7: MC simulation of the DLA with a probabilty to stick to the object of 1

As we can see this looks nothing like the growth described in the first part of the assignment. The difference comes from the fact that the MC method doesn't look at the concentrations around the growing object. As there is a higher chance for the object to grow horizontally (two neighbours instead of one vertical), it is likely that the object spreads horizontally.

The next part of the assignment asked us to implement a sticking probability which is the chance that a random walker will stick to the cluster if it hits the cluster. A sticking probability of 1 indicates that the walker has a 100% chance of sticking to the cluster if it hits. If the walker does not stick to the cluster it continues his path (without moving into the cluster of course). We ran four experiments with probabilities 0.8, 0.6, 0.4 and 0.2. The results are displayed in figure 8.

It is perhaps unsurprising that the plots look very similar to figure 7 but with smaller clusters, as even if a walker hits the cluster (it has to randomly move down 255 times more than going up) there is a chance it wont attach to the cluster.

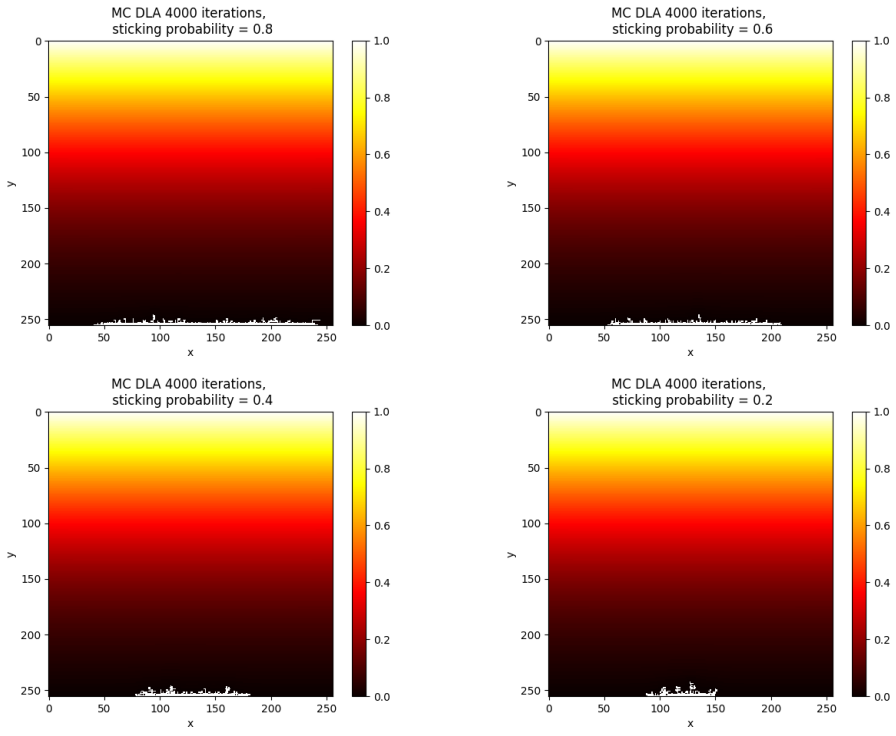

Figure 8: MC simulation of the DLA with different sticking probabilities

4 The Gray-Scott model - A reaction-diffusion system

The Gray-Scott model describes a system where two chemicals, U and V, react with each other. The reaction rate are determined by the local concentrations of the chemicals. The reactions are:

$$U + 2V \rightarrow 3V$$
$$V \rightarrow P$$

U is continuously fed into the system and reacts with V, V then decays into P. The local change for U and V can be described with the following equations:

$$\frac{\delta u}{\delta t} = D_u \nabla^2 u - uv^2 + f(1-u)$$
$$\frac{\delta v}{\delta t} = D_v \nabla^2 v + uv^2 - (f+k)v$$

Here, u and v denote the concentrations of U and V, f controls the rate at which U is supplied into the system and $f + k$ controls the rate at which V decays.

For different values of f and k different behaviour can be observed, there are stable patterns and time-dependent reactions. Because the reaction of U and V is auto-catalytic, it is likely that stable patterns will be hard to find as there is probably a fine balance between f and k for which that pattern emerges.

To implement the Gray-Scott model we created a 100x100 grid where each cell only has interactions with its direct neighbours. We thought it would be interesting to create a petri dish like environment where v starts as a small square in the center and u is everywhere. At the start u is

everywhere in the system and the concentration is 0.5 for each cell and v is only in a small square in the middle with concentration = 0.25 and 0 everywhere else. We added some noise to the system by changing the size of the initial square in the middle.

The initial parameter settings are: $dt = 1$, $dx = 1$, $D_u = 0.16$, $D_v = 0.08$, $f = 0.035$ and $k = 0.060$. We played around by changing f and k and found that for most settings the system is unstable and either V decays too fast (when $f + k$ is too large) or the concentration of V grows and system gets out of control due to the auto-catalytic reaction. The remaining concentration of v after 2000 iterations for different parameter settings are shown in figure 9

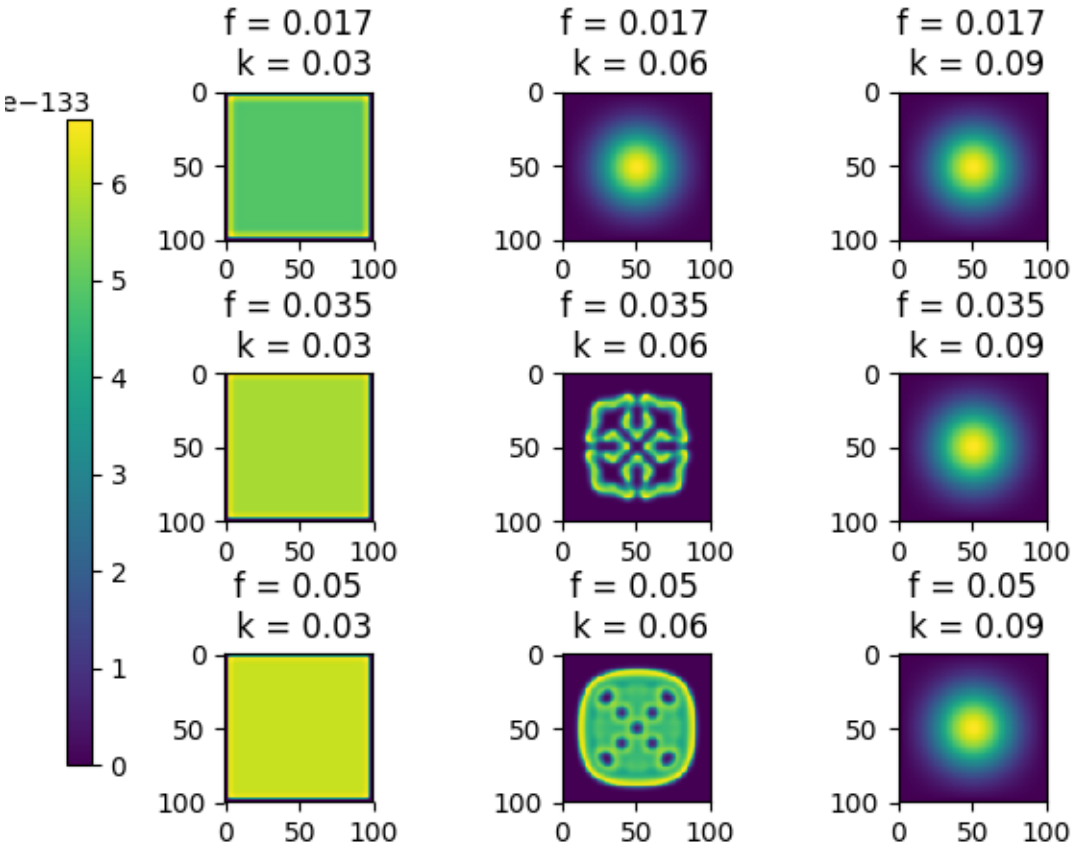

Figure 9: 9 plots of the remaining concentration of v for different parameter settings

We can see that only the default parameter setting (middle figure) and $f = 0.05$ and $k = 0.06$ look interesting. The other reactions seem either too quick (left hand side) or too slow (right hand side). To investigate these parameter settings further we let those run again but then for 10.000 iterations. The results are displayed in figures 10 and 11. We can see that figure 10 has a complex looking structure where the concentration of v seems to be moving in waves that bounds backwards from the edges. For figure 11 it is still unclear if it will result in a stable pattern. There are also some open areas inside the concentration that are not symmetrically distributed in the y direction.

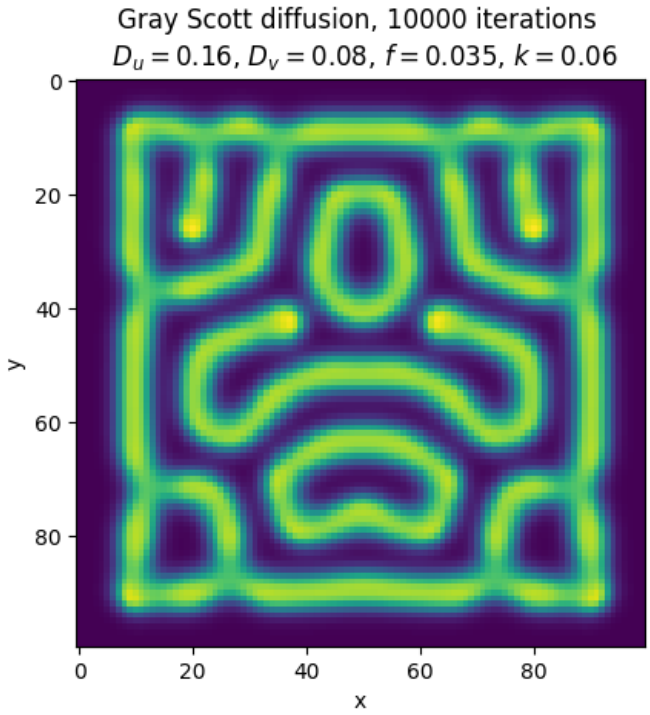

Figure 10: Resulting concentrations for the default settings ($f = 0.035, k = 0.06$)

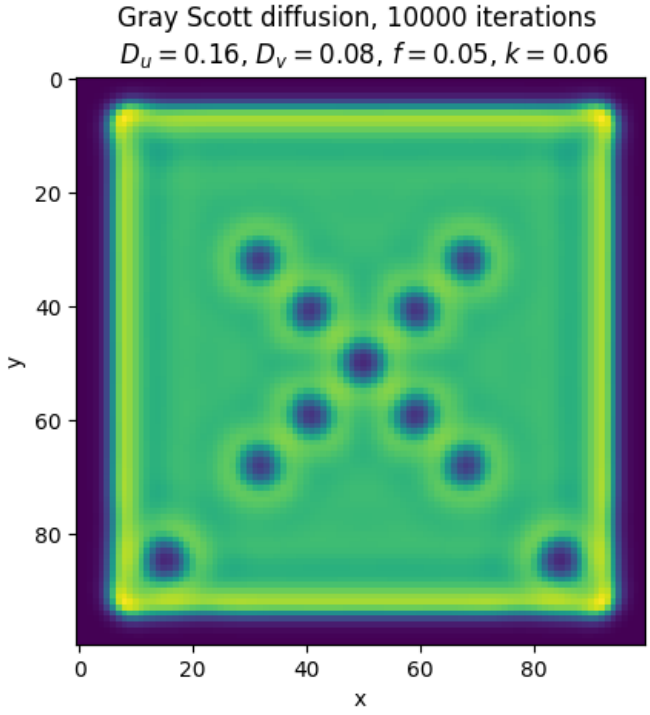

Figure 11: Resulting concentrations for $f = 0.05, k = 0.06$