University of Amsterdam

Scientific Computing

Assignment 2

# Applications of the diffusion equation using Finite Difference methods

Author: Kevin de Vries
Student Number: 10579869
E-mail: kevin.devries@student.uva.nl

May 11, 2018

# Contents

# 1 Introduction

Diffusion is a prevalent phenomenon in many areas of mathematics and in real world systems such as stochastic processes like random walks, biochemical processes and reaction-diffusion systems. Due to its prevalence many applications of the time dependent and time independent diffusion equations exist in mathematical models.

In this report we will use Finite Difference (FD) methods in order to analyze some applications of the diffusion equation. We will investigate Diffusion-Limited Aggregation (DLA) using a Partial Differential Equation (PDE) approach and compare it with a Monte Carlo (MC) method. We will also analyze the Gray-Scott reaction-diffusion equation using a numerical FD solver.

The report will be structured as follows. First the DLA will be implemented and analyzed using a PDE approach. Afterwards the PDE approach will be compared with an MC approach for simulating DLA. Next the Gray-Scott reaction-diffusion model will be implemented and analyzed. Lastly a short conclusion for each of the analyses will be given.

# 2 Diffusion-Limited Aggregation

Diffusion-Limited Aggregation (DLA) is a growth process in which particles cluster together to form an aggregate. DLA growth models can be applied to systems in which diffusion is the primary means of transport. DLA can be implemented using various methods. In this report we will be analyzing two of these methods: a PDE approach and an MC approach.

## 2.1 PDE approach

DLA can be implemented using a PDE approach by solving the time-independent diffusion equation and using the solution to determine the growth of the object in the grid to its neighboring grid points. The full algorithm can be described using the following steps:

1. Initialize the concentration for the diffusion equation (using the analytic solution if known) and insert a single point of the object which is grown during the simulation.

2. Solve the time-independent diffusion equation in order to obtain the stable state of the system.

3. Find the set of growth candidates, which is done by finding all grid points adjacent to the object.

4. Use the concentrations found from the solution of the diffusion equation to determine the probabilities $p_g(i, j)$ with which the object grows into the candidate points.

5. Grow the object using the determined probabilities.

6. Go to step 2 or end the algorithm.

We solve the time-independent diffusion equation using Successive Over Relaxation (SOR), which is an iterative method which has been extensively discussed in the report for the first assignment for exercise set 1. Hence the SOR method will only be mentioned for this report. The boundary conditions used for the diffusion equation are periodic in the x-direction and Dirichlet boundary conditions in the y-direction, which means that $c(0, y) = c(1, y)$, $c(x, 1) = 1$ and $c(x, 0) = 0$.

The growth probability at each of the growth candidates is given by

$$p_g(i, j) = \frac{c_{i,j}^{\eta}}{\sum_{(k,l)\in G} c_{k,l}^{\eta}} \tag{1}$$

where $G$ is the set of grid coordinates corresponding to the growth candidates and $\eta$ is the growth parameter which affects the density of the clustering and the growth type of the object. The probabilities are such that the object growths with one grid point per iteration on average. In order to optimize the implementation of the DLA growth model, we use the solution of the SOR method of each previous iteration, since the grid - and thus also the solution of the SOR method - changes very little on average after each growth of the object.

### 2.1.1 Results

For the analysis of DLA using the PDE approach we use a grid size of $512^2$, where the number of grid points in both directions is thus $N = 512$, and with

$x, y \in [0, 1]$. For the SOR method used in the DLA simulation we will use a convergence measure tolerance of $\varepsilon = 10^{-5}$ and a relaxation parameter of $\omega = 1.9$. We use $\omega = 1.9$ because, although the optimal relaxation parameter seems to increase with the grid size, it also seems decrease as the object grows and previous solutions are used as initial conditions. For the analysis we grow the object for 1500 iterations, which should result in an object with a volume of approximately 1500 grid points.

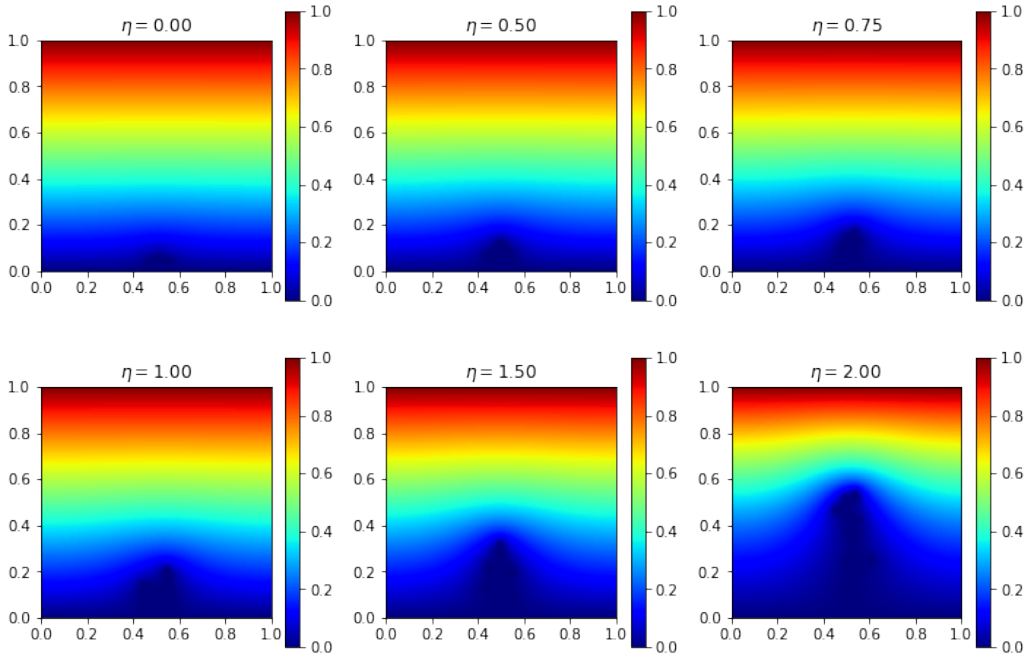


Figure 1: Concentration profiles of grown objects with growth parameters $\eta = 0.0, 0.5, 0.75, 1.0, 1.5, 2.0$

We analyze how the growth of the object depends on the growth parameter $\eta$ by growing an object with an $\eta$ using varying values in the interval $\eta \in [0, 2]$. The concentration profiles of the results can be seen in Figure 1. From Figure 1 we can discern that the specific growth of the object has a large effect on the shape of the concentration profile. The shape of the object however cannot be exactly made out from the Figure, since the grid points around the object also have relatively low concentrations.

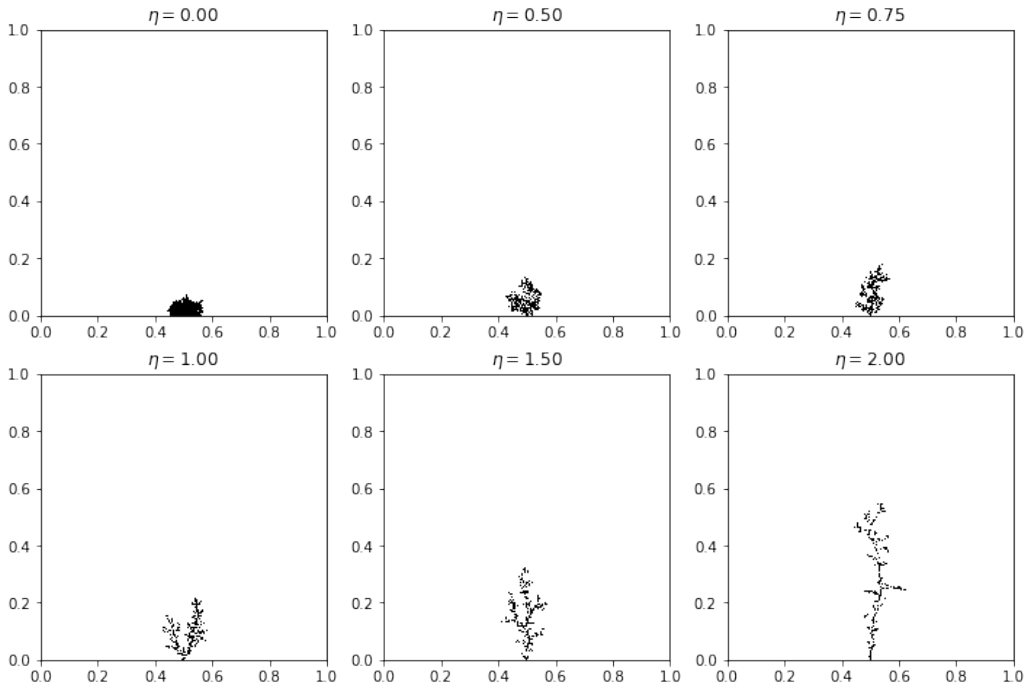


Figure 2: Visualization of grown objects using the PDE approach without the concentration profile and with growth parameters $\eta = 0.0, 0.5, 0.75, 1.0, 1.5, 2.0$

In Figure 2 the visualizations of the objects with different growth parameters can be seen. We find that the objects become less densely clustered as $\eta$ increases. With $\eta = 0$ we find a very dense cluster which is called an Eden cluster. For $\eta \in [0, 1]$ specifically we find that the clusters become more open and less dense as $\eta$ increases, while for $\eta \in [1, 2]$ we find that the clusters do not just become more dense, but also start to develop longer branches as $\eta$ increases. Lastly, with $\eta = 2$ we see that the object starts to resemble a lightning bolt, since it consists of a single long branch connected with many much smaller branches.

## 2.2 Monte-Carlo approach

The DLA growth model can also be implemented using a Monte Carlo (MC) approach. For the MC approach random walkers are released on a randomly chosen point at the top of the grid one at a time, which walk through the

grid in steps of 1 grid point at a time in any of the cardinal directions and stick to the object with a chance of $p_s$ when they enter a grid point adjacent to a grid point which is occupied by the object. In order to simulate the boundary conditions used by the PDE approach, the random walkers are removed and replaced by a new random walker when they leave the grid in the y-direction and move to the other side of the grid when they leave the grid in the x-direction. Lastly, the random walkers are not allowed to move into a grid point which is occupied by the object, which is otherwise possible if a random walker does not stick to the object when it is adjacent to the object.

### 2.2.1 Results

For the analysis of DLA using the MC approach we use a grid size of $200^2$ (thus $N = 200$) and with $x, y \in [0, 1]$. The grid used for the MC approach is smaller than the grid used for the PDE approach, because the MC implementation becomes much less computationally efficient as the grid size increases due to the fact that the random walkers have a smaller chance of meeting the object if the grid size is larger. The algorithm does however become more efficient as the volume of the object increases. In order to analyze the object in reasonable detail, the growth size of the object is set to 1000, which means that the growth of the object is terminated when 1000 random walkers are stuck onto the object. We simulate the DLA growth for objects with sticking probabilities of $p_s = 0.1, 0.25, 0.5, 0.75, 1.0$ in order to analyze the effect of the sticking probability on the growth of the object.

The visualizations of the objects grown using different sticking probabilities $p_s$ are visualized in Figure 3. In Figure 3 We find that the growth becomes more open and less dense as $p_s$ increases, which means that the object will likely start to resemble an Eden cluster when $p_s$ goes to zero just like with the PDE approach with $\eta = 0$. We also find some differences with the objects resulting from the PDE approach. For example, we find that the MC generated objects generally produce objects with longer branches, which causes the objects to grow more horizontally that the objects generated with the PDE approach.

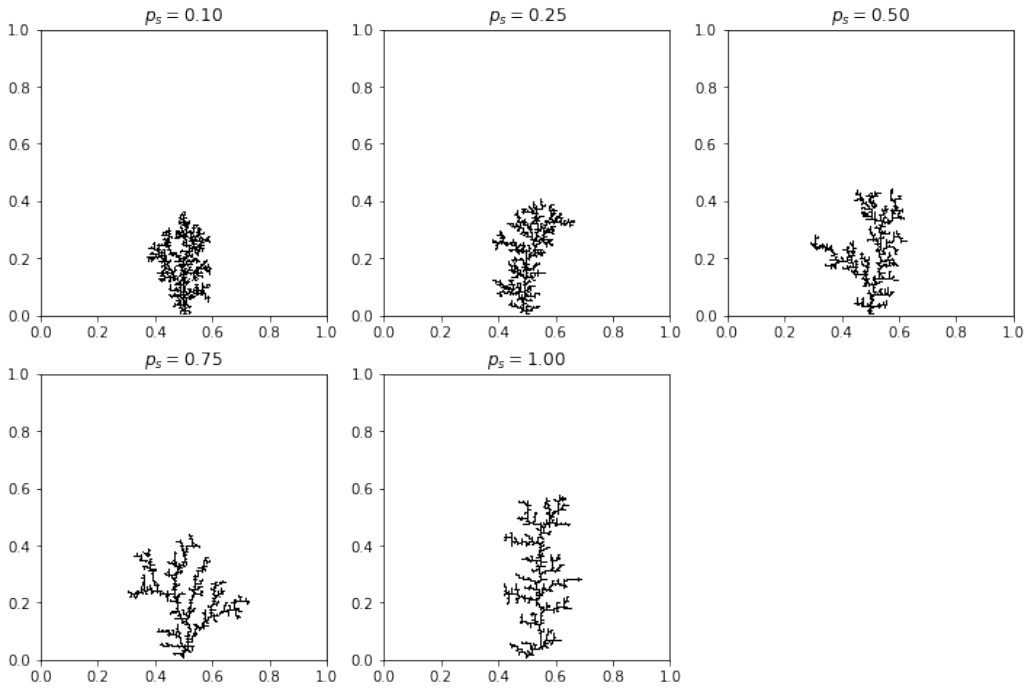


Figure 3: Visualization of grown objects using the MC approach with sticking probabilities $p_s = 0.1, 0.25, 0.5, 0.75, 1.0$

# 3 Gray-Scott Reaction Diffusion Equations

Lastly we will analyze the Gray-Scott reaction-diffusion model using numerical solutions found using an explicit FD scheme.

The Gray-Scott reaction-diffusion equations model the following biochemical reactions

$$U + 2V \to 3V \tag{2}$$

$$V \to P \tag{3}$$

where $U$ is constantly supplied to the system. $U$ reacts with $V$ to produce more of $V$, which makes the reaction autocatalytic. $V$ itself decays into $P$, which does not react with $U$ or $V$. The reaction-diffusion process is given by

the equations

$$\frac{\partial u}{\partial t} = D_u \nabla^2 u - uv^2 + f(1-u) \tag{4}$$

$$\frac{\partial v}{\partial t} = D_v \nabla^2 v + uv^2 - (f+k)v \tag{5}$$

where $f$ controls the supply rate of $u$ and (f+k) controls the decay rate of $v$.

We will implement the Gray-Scott equations in 2D using an explicit FD scheme. For the boundary conditions we take periodic boundary conditions in both the $x$ and $y$-directions. Using a forward approximation in time and central approximations in space, the solutions of the reaction-diffusion equation can be approximated using the following FD scheme:

$$\begin{aligned} u_{i,j}^{k+1} =& u_{i,j}^k + D_u \frac{\Delta t}{\Delta x^2}(u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k - 4u_{i,j}^k) \\ & - u_{i,j}^k (v_{i,j}^k)^2 + f(1 - u_{i,j}^k) \end{aligned} \tag{6}$$

$$\begin{aligned} v_{i,j}^{k+1} =& v_{i,j}^k + D_v \frac{\Delta t}{\Delta x^2}(v_{i+1,j}^k + v_{i-1,j}^k + v_{i,j+1}^k + v_{i,j-1}^k - 4v_{i,j}^k) \\ & + u_{i,j}^k (v_{i,j}^k)^2 - (f+k)v_{i,j}^k \end{aligned} \tag{7}$$

Since periodic boundary conditions will be used, the FD scheme can be updated at the boundaries by using an appropriate substitution given $u_{0,j}^k = u_{N,j}^k$ and $u_{i,0}^k = u_{i,N}^k$ where $N$ is the maximum grid point number and thus the size of the grid. We will use the same substitutions at the boundaries for $v_{i,j}^k$.

## 3.1 Results

For the analysis of the Gray-Scott Equations we use a grid of $100^2$ (thus $N = 100$) for 5000 time steps $M = 5000$ with a step size of $\Delta x = 1$ in space and $\Delta t = 1$ in time. We use the initial conditions $u_{i,j}^0 = 0.5, \forall i, j$ and $v_{i,j}^0$ equal to zero for all $i, j$ except in the middle of the grid where we set a small square of 5 by 5 grid points to 0.25. We then perturb the initial conditions with a small amount of uniform random noise while keeping the values of $u_{i,j}^0$ and $v_{i,j}^0$ between zero and one.

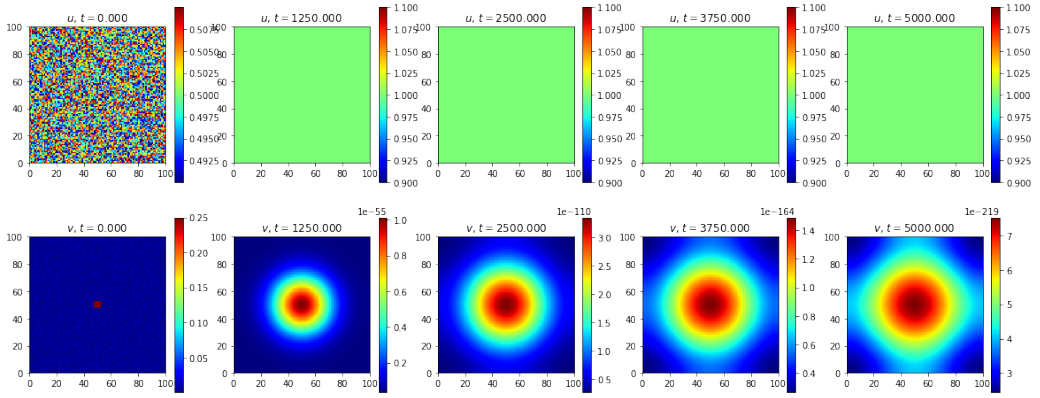


Figure 4

We first try the parameters $f = 0.035$, $k = 0.060$, $D_u = 0.16$ and $D_v = 0.08$ with some uniformly distributed noise added, which is in the interval $[-0.01, 0.01]$ for $u$ and $[0, 0.01]$ for $v$. The progression of the numerical solution over time is shown in Figure 4. We find that $u$ reaches a stable state with a value of 1 over the whole grid, whereas $v$ converges to 0 with a small peak in the middle of the grid. We also find that the solution is relatively robust with respect to small perturbations.

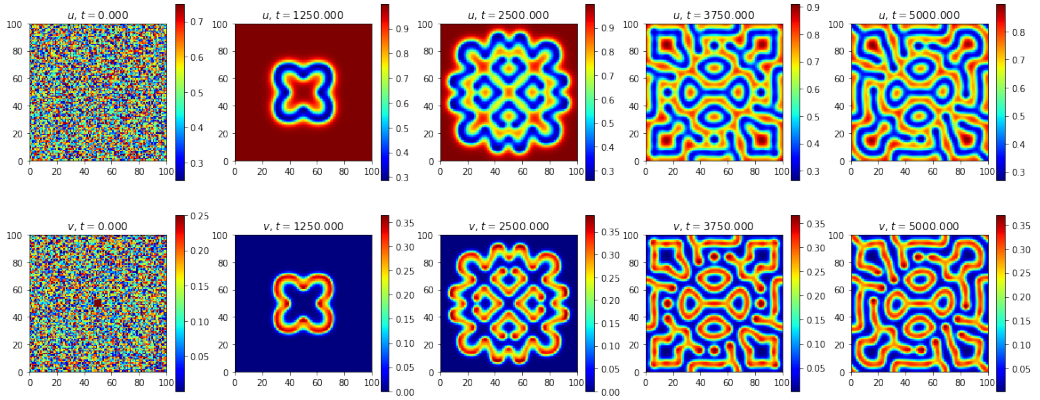


Figure 5

When we add more noise to the initial conditions, the patterns become a bit more interesting. When we add uniformly distributed noise in the interval $[-0.25, 0.25]$ for $u$ and $[0, 0.25]$ for $v$, we start to reliably find different results from the result in Figure 4. The resulting pattern is shown in Figure 5. We

find that strips of larger $u$ and $v$ concentrations start to form. We also find that low $u$ concentration is associated with high $v$ concentration and vice versa, which is expected from the reactions which the equations model.

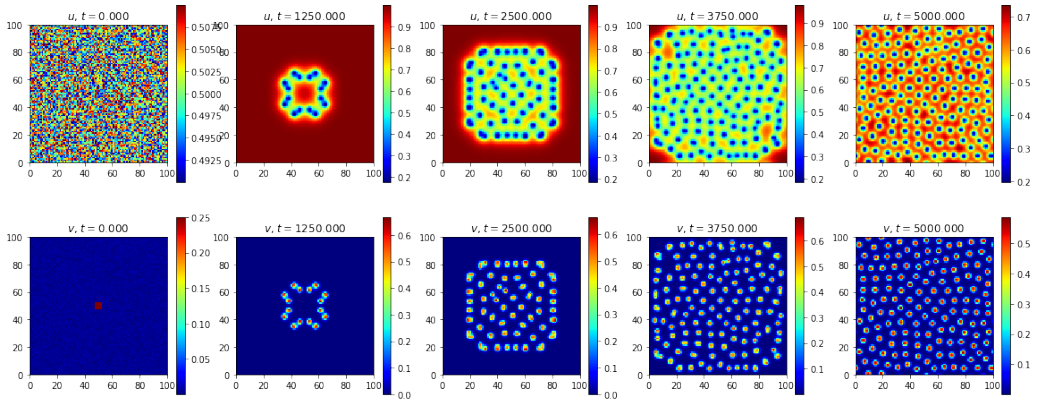


Figure 6

We also try the parameters $f = 0.02$, $k = 0.077$, $D_u = 0.2$ and $D_v = 0.035$ with uniformly distributed noise added, which is in the interval $[-0.01, 0.01]$ for $u$ and $[0, 0.01]$ for $v$. Using these parameters, we find that the system converges to a stable state, which is given by a dotted pattern. The resulting pattern is shown in Figure 6.

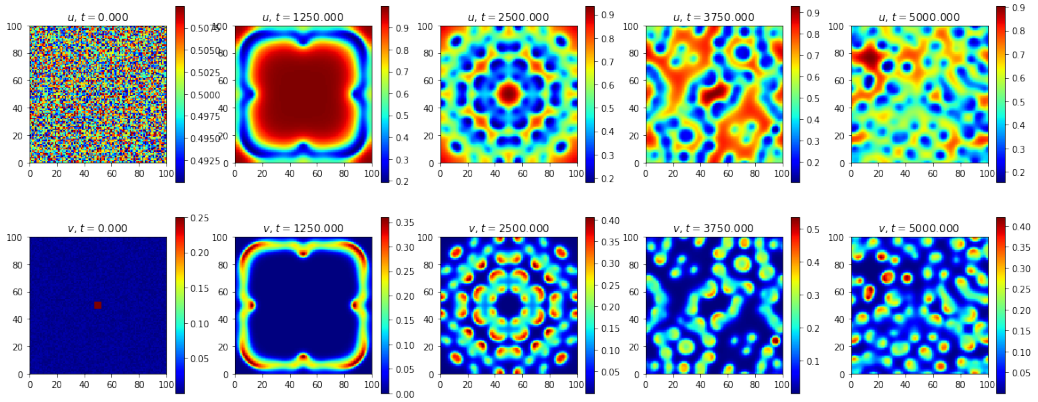


Figure 7

Lastly, we try the parameters $f = 0.017$, $k = 0.05$, $D_u = 0.2$ and $D_v = 0.08$ with uniformly distributed noise added, which is in the interval $[-0.01, 0.01]$

for $u$ and $[0, 0.01]$ for $v$. The resulting patterns using these parameters are shown in Figure 7. From Figure 7 we can discern that the system does not converge to a stable state, but instead keeps evolving seemingly indefinitely. We find that for these parameters self propagating wave like 'clusters' seem to be prevalent. These clusters seem to annihilate each other when two of them get too close to together. This specific behavior seems to be what causes the indefinitely evolving patterns.

# 4 Conclusions

From the DLA analysis using the PDE approach we can conclude that increasing the growth parameter increases the openness and density of the clusters. We can also conclude that it increases the length of the object when the growth parameter becomes larger than 1. From the DLA analysis using the MC approach we can conclude that increasing the sticking probability, similar to the growth parameter used in the PDE approach, increases the openness and density of the object. The main difference between the PDE and MC approaches is concluded to be the length of the minor branches, which is generally higher when using the MC approach and which causes the object to grow more horizontally. Lastly, from the Gray-Scott reaction-diffusion equations analysis we can conclude that many possible patterns can form when using different parameter combinations and that the numerical solutions are relatively robust to small perturbations in the initial conditions. In order to gain more accurate results however, the equations would likely be more reliably and accurately solved using implicit finite difference methods of using finite element methods.