

Student name: **Bugs Bunny**
Email to: Edwin Kite (kite@berkeley.edu)

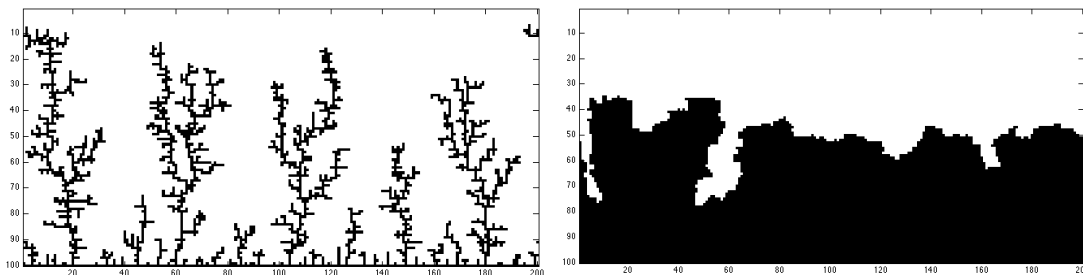
EPS 109, Fall 2008
Instructor: Burkhard Militzer

Computer Lab / Homework Assignment 3

The whole set is due: Monday Sept 22, 2008, midnight

Diffusion Limited Aggregation

In this lab, you will edit a code that simulated the 2D diffusion of particles and their aggregation on a surface. Each particle performs a random walk until it gets stuck on the substrate, or on one of the forming dendrites. You should understand why different growth mechanisms lead to two so different pictures shown here:



(1) Please download the Matlab file `diffusion_limited_aggregation2D_exercise.m` from bSpace and familiarize yourself with every section of it. Please pay attention to the 2D array `A(x,y)` and understand what the ranges 'x' and 'y' mean.

(2) Please insert the diffusion step at lines marked with `%%%`. Based on the value of 'rval' please go left, right, up, or down with 25% probability each. Add 7 lines of code.

```
rval = rand;  
%%% Determine the direction of motion using the random number 'rval'  
%%% Update 'x' and 'y' with the following probabilities  
%%% 25% go LEFT  
%%% 25% go RIGHT  
%%% 25% go UP  
%%% 25% go DOWN  
%%% --> need to enter about 10 lines of code
```

(3) Right below in the file apply periodic boundary conditions in x direction. If the particle has left the simulation cell on the left, let it reappear on the right, and vice versa.

(4) Now run the code by type the following in the command window:

```
diffusion_limited_aggregation2D(1000,200,50)
```

1000, 200, 50 are input parameters that are passed into the subroutine in the `diffusion_limit_aggregation2D.m` file and set to the variables `nParticles`, `maxX`, `maxY`.

Now increase the 3 input parameters until you have reproduced the left picture above using 10000 particles. Include your answer to the following questions here: Is this a fractal? Why does it form such long dendrites? Why does it not solidify like an ice crystal?

To understand how probabilities work, run the following code *more than once*:

```
p1 = 0.49;
p2 = 0.48;
nm = 0;
nn = 0;
no = 0;

for i=1:100
    r = rand;
    if (r<p1)
        display('Obama');
        no = no+1;
    elseif (r<p1+p2)
        display('McCain')
        nm = nm+1;
    else
        display('Nader')
        nn = nn+1;
    end
end
display(no);
display(nm);
display(nn);
```

Part II: Now we want to see if we can adapt the program to make more compact structures. One main problem is that new particles get stuck too early before they can diffuse further down to fill any gaps. This will now be changed. (The other problem is that attractive forces play an important role in forming ice crystals. This will not be addressed in this exercise.)

Main change: A particle will not necessarily get stuck the first time it gets close to others. We introduce a sticking probability 'p'. If the diffusing particle hits other then its sticks only with probability 'p' and keeps moving on otherwise. In part I, the particle would always stick, which is equivalent to $p=1$.

(1) Add a line that prevents the particle diffusion into the lower wall. If y goes below 1 then set it back to 1.

(2) Add 'p' as additional input parameter near the top of the *.m file

```
function diffusion_limited_aggregation2D(npath, maxX, maxY, p)
```

This mean you now have to start the program with for example
`diffusion_limited_aggregation2D(1000,200,50,0.02)`

(3) Now introduce another if statement before

```
if (A(xp,y) > 0 | A(xm,y) > 0 | A(x,y+1) > 0 | A(x,y-1) > 0)
```

so that [if the site is empty ($A(x,y)=0$)] the particle sticks only with probability 'p'. Use another random variable using the 'rand' function.

(4) Run the code with 10000 particles and see what pictures you get for different sticking probabilities ranging from 0.01 to 1.0. Enter your pictures for four different 'r' values, $0 < r < 1$.

Attach the final version of your code when you're your homework by email.

Optional: *Part III: You should have realized that lowering the sticking probability was not quite enough to fill all the wholes. One needs specifically increase the chance that the particle sticks where it has not only one, but two or more neighbors.*

Update the code so that the particle sticks

- with probability p if there is 1 neighbor,
- with probability 10*p if there are 2 neighbors,
- with probability 20*p if there are 3 neighbors,
- with probability 30*p if there are 4 neighbors.

Run the code with different 'p' until you reproduce the right graph above. Again enter four picture for the same four 'r' values that you have used in part II. Briefly discuss the differences to plot in part II. **Attach the final version of your code when you're your homework by email.**