In-Class Work: DLA — Fractal Dimension

We have been implementing the Diffusion Limited Aggregation (DLA) model by Witten and Sander [T.A. Witten, L.M. Sander, Phys. Rev. Lett **47**, 1400 (1981)]

8. Finished DLA program

Last class you all worked on the last pieces of the implementation of the DLA model. Awesome! Today we will work on how the resulting DLA-cluster can be analyzed, namely you will measure the so called fractal dimension. To ensure that everybody will work on this analysis (instead of finishing their own version of the program), copy the following program into your working directory
`~kvollmay/share.dir/inclass2025.dir/classfractal8.py`
I will guide you through this program.

9. Fractal Dimension of DLA Cluster

9a. I will give you an intro to a defintion for the fractal dimension. (You find a description of the fractal dimension in the textbook of Gould, Tobochnik, Christian on the first three pages of Chapter 13. We use the technique describted in problem 13.1b.)

9b. Now lets get ready to analyze the pattern of the DLA model. You will determine the fractal dimension of one pattern using the method of checking squares of length `b`, as just described in class.

To avoid having to run the DLA program again and again, let us first prepare one pattern, which you then will analyze in 9c. Run the program
`~kvollmay/share.dir/inclass2025.dir/classfractal8.py`
This program makes the file `bigDLAcluster.dat` (and a nice pdf-file `frame8.pdf` just for fun). (Or if you have your own finished DLA program, have a look at the last few lines of `classfractal8.py` to see how to write the file `bigDLAcluster.dat`.) Ensure that you run the program for `LATSIZE=500` and for `NPARTMAX=3000`. This will take a while, but we have to do this only once, because for the analysis we use `bigDLAcluster.dat`.

9c. Now you need a program which reads in the 224x224 matrix from your file `bigDLAcluster.dat`. You may use for this task
`~kvollmay/share.dir/inclass2025.dir/classfractal9start.py`
To get the fractal dimension d_f we use the following relation.

$$\ln(N) = \ln(c) + d_\mathrm{f} * \ln(b) \tag{1}$$

where N is the number of occupied sites, c is some constant and b is the length of your square for which you count the number of occupied sites. You see that Eq.(1) defines d_f and it tells us that if we plot $\ln(N)$ as a function of $\ln(b)$ then we should get a line with slope d_f. So our task is to get N and b. Add to your program that you count the number of occupied sites N for a lattice of lenght b, where you center your lattice of lenght b around the midpoint of your 224 x 224 lattice. Loop over the length of your lattice and print out $\mathrm{l}n(N)$ as a function of $\ln(b)$. Let's say you do

`classfractal9c.py` > `lnNoflnb.dat`
Hint: ln(N) is in python `np.log(N)`

9d. Next we fit a line to our data from 9c stored in file `lnNoflnb.dat`. For this we use gnuplot. So type in the command line "`gnuplot`". Then type "`plot "lnNoflnb.dat"`". Define a function $f(x)$ by typing "`f(x) = a*x + b`". Now fit your data within the xrange [2.0,4.2] to a line by typing "`fit [2.0:4.2] f(x) "lnNoflnb.dat" via a,b`". The resulting a is the fractal dimension d_f. You can look at the data and fit with "`plot "lnNoflnb.dat",f(x)`"
Or if you like all commands within gnuplot together:

```
a=2;b=1;f(x)=a*x+b; fit [2.0:4.2] f(x) "lnNoflnb.dat" via a,b; plot "lnNoflnb.dat",f(x)
```

Play some with the fit range. Compare your fractal dimension with the expected value of 1.71

To make figure:

To save your gnuplot session in the file `DLAdf.gnu`, you type within gnuplot `save "DLAdf.gnu"`
When you start a new gnuplot session, you can load your gnuplot-session within gnuplot with `load "DLAdf.gnu"`
You get for example the xlabel with `set xlabel "ln(b)"` and to save your figure into a postscript file use within gnuplot for example `set term postscript landscape; set out "DLAdf.ps";replot`
To convert the DLAdf.ps into a pdf-file, you can use outside of gnuplot in the terminal window `ps2pdf DLAdf.ps`