Complex Physics Project 2

Jakob Schauser, pwn274

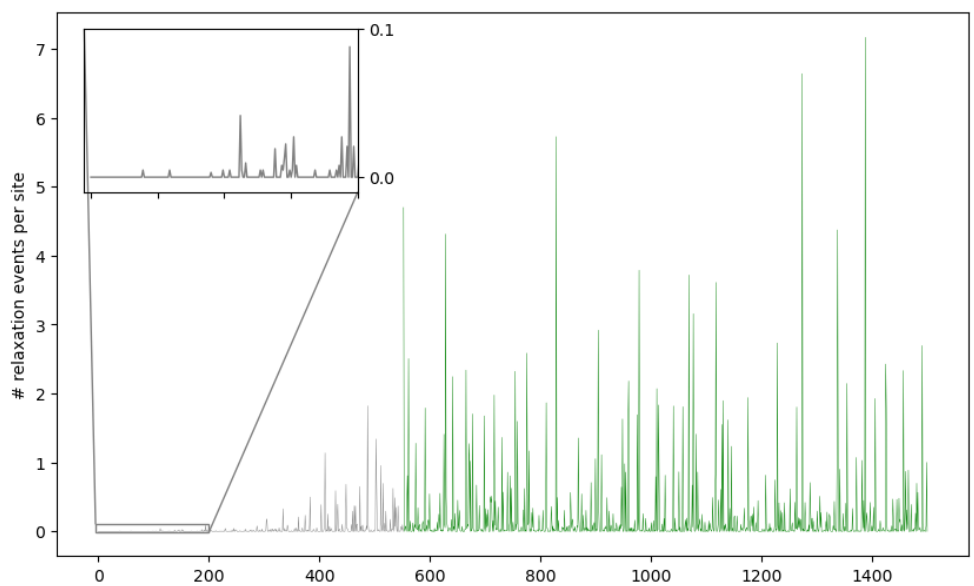
October 2023

1

Implement the above model for a L $\times$ L lattice with L = 25 and plot the sequence of avalanches as a function of time. The avalanche size is defined as the number of single-site relaxations until the new relaxed state is reached. Convince yourself when steady state dynamics is reached (this critical attractor is characterized by avalanches that do not anymore increase in size)

I have implemented the model. I chose (as it was not otherwise described in the assignment) to allow for the same neighbor to be chosen twice.

Below is a run of a 25x25 model, trying to emulate fig 4.9 from the notes but changing the y-axis to be 'per site':



I defined (for lack of better) steady state as the first time an event of 4 relaxations per site has happened. This seemed to be a good heuristic point on scales i checked [10 : 500]. Unless otherwise noted, my data from this point on is taken after this point.

2

Consider system sizes L = 25, L = 50, L = 100 and L = 200 and simulate each system until a steady state is reached. In steady state, you should extract the size distribution of the avalanches, and estimate the exponent for the size distribution and for the scaling of the avalanche cut off with system size L (dimension of avalanches)

To do this, I have run the simulations for a large number of times, making a histogram akin to fig. 4.10 in the notes and fitted the following piecewise function:

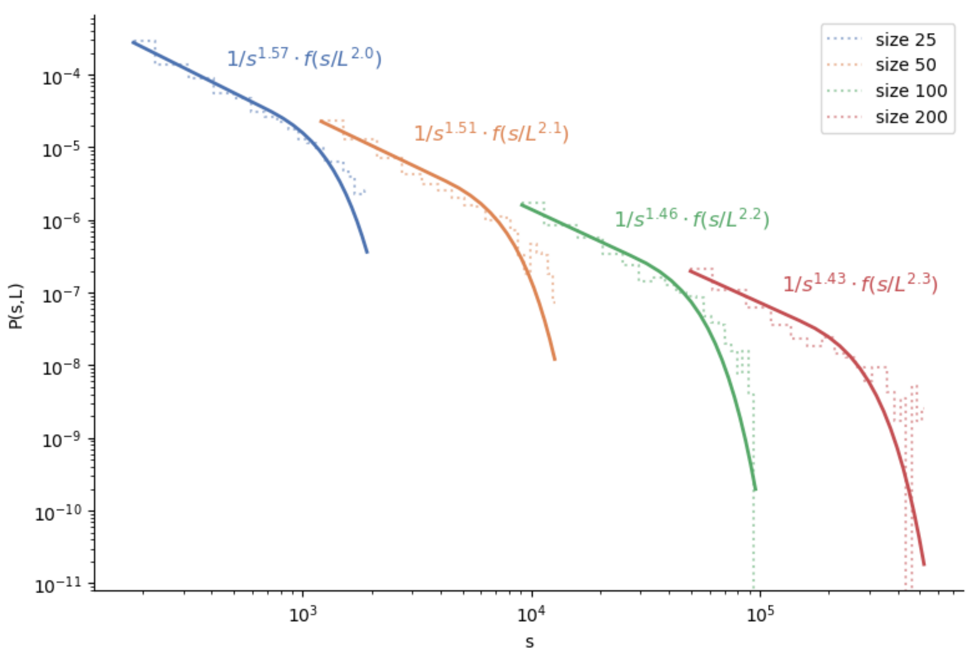
```
def fit_fun_over(x, tau, cutoff):
    # A standard power law
    y =  x ** (-tau)

    # a quick exponential decay after the cutoff, multiplied and centered to keep it continous
    y[x > cutoff] *= np.exp(-((x[x>cutoff] - cutoff)/cutoff)**2)

    return y

# Choosing the cutoff to be L^d, effectively fitting for d
fit_fun = lambda x, tau, d: fit_fun_over(x, tau, L**d)
```

This effectively allows me to fit for the exponent and the dimensionality at the same time. The motivation behind this function was mainly Equation 4.18 in the notes, and the small paragraph right after.



Got the average dimension to be 2.16[1] and critical power-law exponent to be 1.49 both of which seem

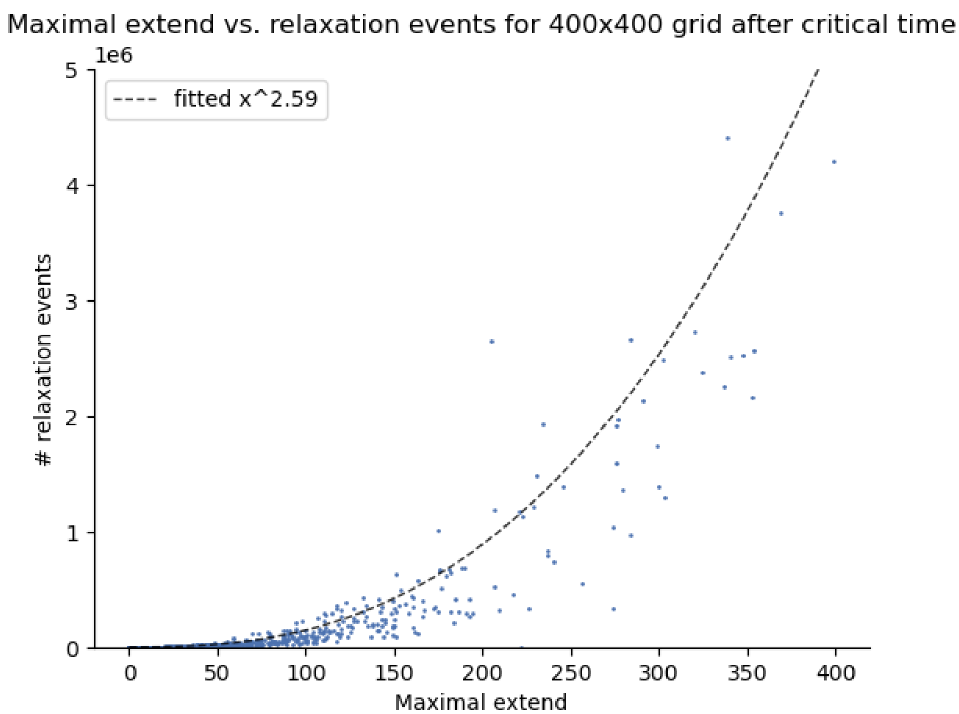
[1] I am sorry i did not have more decimals in the exponent in the graphic. The fitted dimensions were: [2.037, 2.134, 2.238, 2.256], the rest can be seen in my code

reasonable. The dimensionality seems to go up with the system size, which makes sense as we near the thermodynamic equilibrium.

3

For a big lattice, then plot the size of avalanches (in number of relaxation events) as a function of the largest linear dimension of each avalanche (longest extension of avalanche along either the x-axis or y-axis). Deduce the dimension of the avalanche

Not much interesting happened in the implementation of this one. Here is the scatter plot and the fit:

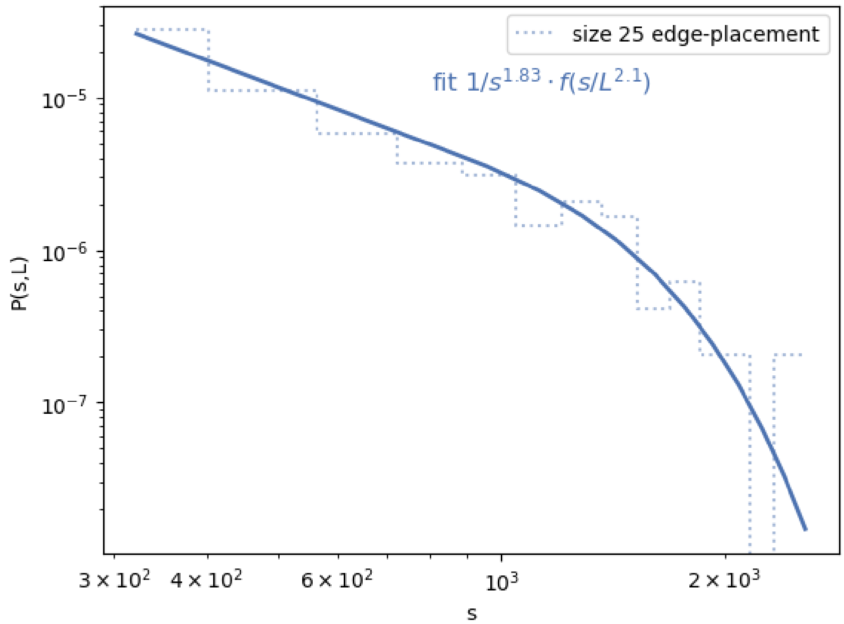


By fitting the described in the notes I find a dimension of 2.59, I did not make a fit for such a large system before, but this dimensionality dies not seem completely off from the previous estimates.

4

4.1 Simulate the system where grains are only added on the edge of the system and record the avalanche size distribution with this constraint. Deduce the corresponding exponent for these avalanches

Here I am basically doing the same analysis as I did in **2**, but changing the way the simulation runs. The implementation was once again not too hard, so I will jump straight to my results:



This time I get a slightly larger tau/exponent. This is corresponding to a slightly sharper distribution ie. with fewer very large events.