Percolation of a 2D Lattice

Tahoe Schrader, Ksenia Sokolova, Xinmeng Tong
PHYS566

Abstract

In this assignment we computationally explore various $2D$ lattices and generate their spanning clusters, p_c and β. These variables refer to the probability a spanning cluster arises at and the critical exponent of a spanning cluster, respectively. Theoretically, these values for an infinite $2D$ lattice are $p_c \approx 0.593$ and $\beta = \frac{5}{36}$.
Our code is available on GitHub:

`www.github.com/tahoeschrader/PHYS566_group4_projects`

1 Theory

Percolation theory describes the behavior of connected clusters. This theory has many important relations to physics and chemistry. For example, it can be used to describe the movement and filtering of fluids through porous materials. When a system *percolates*, a 2^{nd} order phase transition can also result.

The act of percolation refers to the existence of a spanning cluster. A spanning cluster is a cluster that reaches all edges of a physical boundary. For example, a spanning cluster must touch all four sides of a $2D$ box as seen in Figure **??**. Percolation, i.e. a system exhibiting a spanning cluster, is interesting because universality is *also* exhibited. Universality is when the properties of the system are independent of the systems dynamical details.

One feature of the percolating system that we would like to document is the clustering probability, p_c, where generally

$$p = \frac{\#\text{ of occupied lattice sites}}{\text{total lattice sites}}. \tag{1}$$

Probability ranges and the theoretical results of such values are displayed in Table 1.

small p	isolated clusters
$p \approx 0.4$	many small clusters
$p \approx 0.6$	large, barely connected clusters
$p \approx 0.8$	most sites belong to same cluster

Table 1: The physical representation of various occupation probabilities on a lattice.

Another important value in percolation theory is the fraction of all sites in the spanning cluster with respect to all occupied sites. This value is defined as F and takes the form

$$F = F_0 \left(p - p_c\right)^{\beta}, \tag{2}$$

where β is the critical exponent.

1.1 2D Lattice Systems

The steps for generating a $2D$ percolating lattice are quite simple:

1. Generate a 2D lattice
2. Populate a single lattice site at random and define it to be a cluster
3. Populate another single lattice site at random
   (a) If the new site touches an old site, add it to the cluster
   (b) If the new site doesn't touch an old site, define it to be a new cluster
   (c) If the new site touches multiple different clusters, choose the smaller numbered cluster to "win" and overtake the other clusters
4. Repeat until a spanning cluster arises

It is evident that larger values of p will more likely result in the existence of a spanning cluster than smaller values of p. Interestingly, the transmission from many clusters to a spanning cluster is sharp, qualifying it as a phase transition. For an infinite $2D$ lattice, the theoretical probability where this occurs is

$$p_c \approx 0.593 \tag{3}$$

whereas the critical exponent of the spanning cluster fraction is

$$\beta = \frac{5}{36}. \tag{4}$$

In Section 2, we computationally explore such a $2D$ lattice and generate spanning clusters, p_c and β for various lattice sizes.

2 Results

First we determine the critical probability by running the code for $N = 5, 10, 15, 20, 30, 50, 80$ for 50 times, every time finding the p_c.

To do this computationally, we used the cluster labeling method outlined above. Before running the averaging step, we tested the performance of our code by checking the dependencies between the time required to run one iteration and the size of the matrix. Please see results on Figure 1. We observe that our code is complexity $O[N^2]$, this is expected from the algorithm. This means that the larger the matrix, the worse the code performs, with squared scaling. And the change for the bad side happens relatively quickly. While this is durable for the cases we are observing, for larger matrices we would need to switch to the different algorithm.

Before checking finding the p_c we can also visually observe formation of clusters. (Note: on github project you can also find a .gif of the process). On the Figures 2 to 4 we see the development of the cluster for $N = 30$. In the beginning, as expected, we see many small clusters. As p increases we observe formation of larger clusters, but still no spanning cluster. And at the end we see one large spanning cluster, and just a couple of smaller clusters.

To find critical probability, we ran the code for $N = 5, 10, 15, 20, 30, 50, 80$ for 50 times, averaging for every N. Please see the resulting plot attached (Figure 5). Extrapolating, we found the critical probability to be 0.598. This is very close to the theoretical value of 0.593.

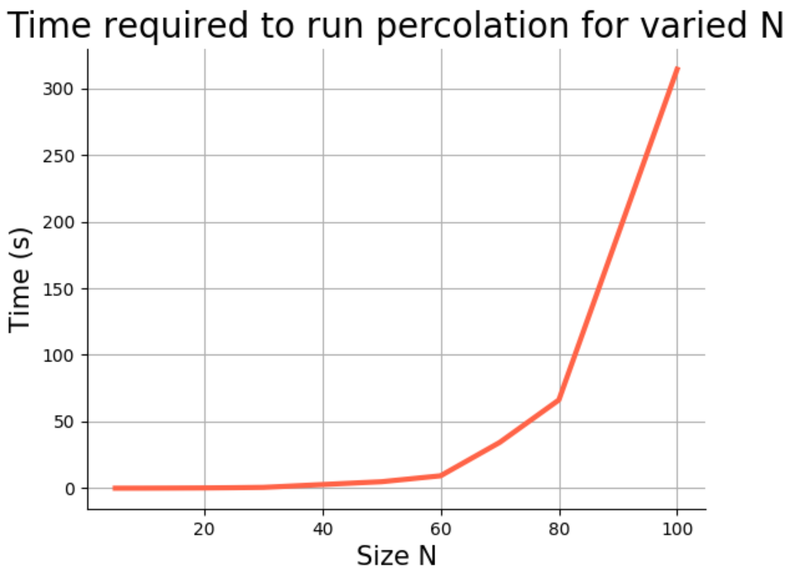

Figure 1: Time elapsed and size of the matrix

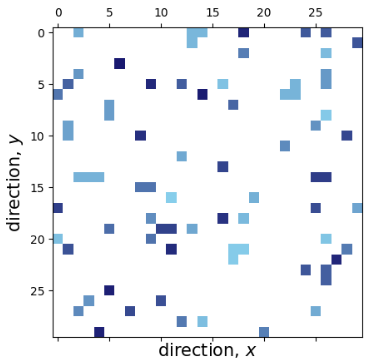

Figure 2: 2D percolation start

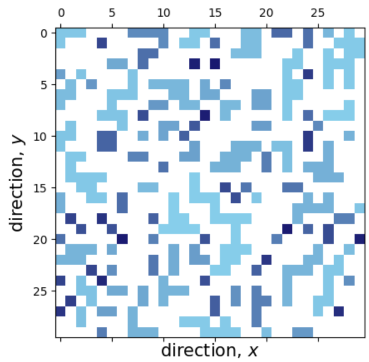

Figure 3: 2D percolation intermediate

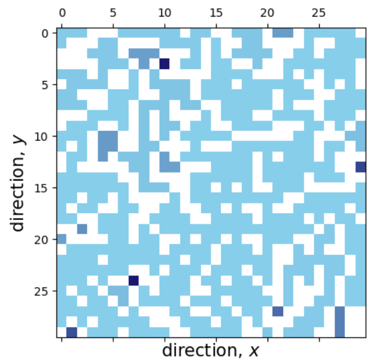

Figure 4: 2D percolation spanning cluster

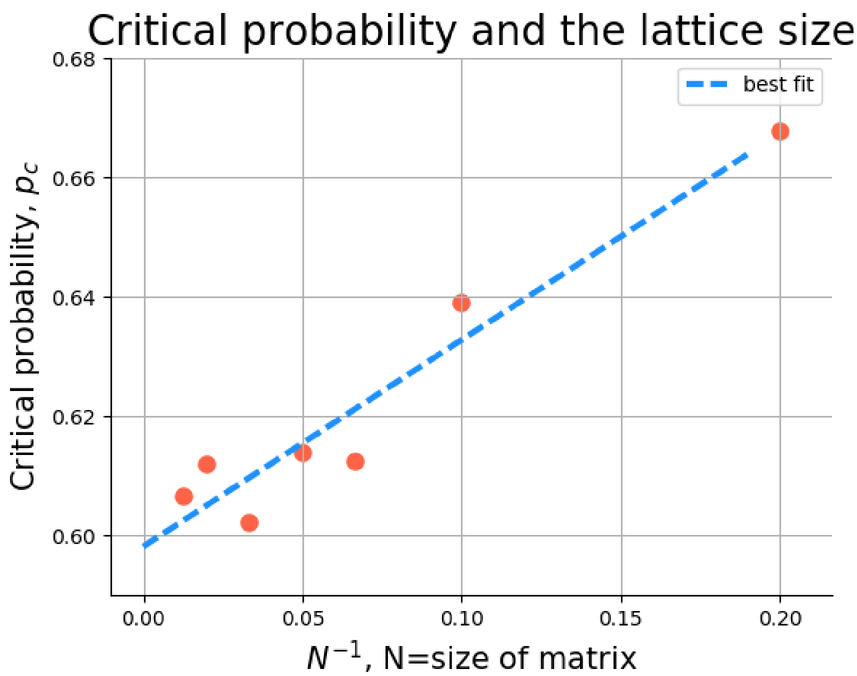

Figure 5: Critical probability plot for2D percolation