# Fys4460 – 2013 – Project 3

## Percolation

In this project, we will develop the tools and knowhow necessary to study scaling in numerical, experimental and real-world data. You will gain experience with image analysis, discrete models for phase transitions, finite size scaling models, the geometry of percolation clusters including subset geometry, and dynamic processes on fractals, with particular emphasis on the dynamics of a random walker on a self-similar fractal: the percolation cluster.

### Generating percolation cluster

First, we use Matlab to generate and visualize percolation cluster. We generate an $L \times L$ matrix of uniformly distributed random numbers, and introduce the tools necessary to visualize and analyze the clusters.

We generate the percolation matrix consisting of occupied (1) and unoccupied (0) sites, using

```
L = 100;
r = rand(L,L);
p = 0.6;
z = r<p; % This generates the binary array
[lw,num] = bwlabel(z,4);
```

We have then produced the array `lw` that contains labels for each of the connected clusters, and the variable `num` that contains the number of clusters.

We can examine the array directly by mapping the labels onto a color-map, using `label2rgb`.

```
img = label2rgb(lw,'jet','k','shuffle');
image(img);
```

We can extract information about the labeled image using `regionprops`, for example, we can extract an array of the areas of the clusters using

```
s = regionprops(lw,'Area');
area = cat(1,s.Area);
```

You can also extract information about the `BoundingBox` and other properties of clusters using similar commands

```
s = regionprops(lw,'BoundingBox');
bbox = cat(1,s.BoundingBox);
```

**(a)** Using these features, you should make a program to calculate $P(p, L)$ for various $p$. Hint: you can use either `BoundingBox` or `intersect` and `union` to find the spanning cluster. How robust is your algorithm to changes in boundary conditions? Could you do a rectangular grid where $L_x \gg L_y$? Could you do a more complicated set of boundaries? Can you think of a simple method to ensure that you can calculate $P$ for any boundary geometry?

**(b)** We know that when $p > p_c$, the probability $P(p, L)$ for a given site to belong to the percolation cluster, has the form

$$P(p, L) \sim (p - p_c)^{\beta} .$$

Use your program to find an expression for $\beta$. For this you may need that $p_c = 0.59275$.

### Determining the exponent of power-law distributions

First, we need to develop tools to analyse power-law type probability densities.

Generate the following set of data-points in Matlab:

```
z = rand(1e6,1).^(-3+1);
```

Your task is to determine the distribution function $f_Z(z)$ for this distribution. Hint: the distribution is on the form $f(u) \propto u^{\alpha}$.

**(c)** Find the cumulative distribution, that is, $P(Z > z)$. You can then find the actual distribution from

$$f_Z(z) = \frac{dP(Z > z)}{dz}$$

**(d)** Generate a method to do logarithmic binning in Matlab. That is, you estimate the density by doing a histogram with bin-sizes that increase exponentially in size. Hint: Remember to divide by the correct bin-size.

### Cluster number density $n(s,p)$

We will generate the cluster number density $n(s,p)$ from the two-dimensional data-set.

**(e)** Estimate $n(s,p)$ for a sequence of $p$ values approaching $p_c = 0.59275$ from above and below.

Hint 1: The cluster sizes are extracted using `.Area` as described in a previous exercise.

Hint 2: Remember to remove the percolating cluster.

Hint 3: Use logarithmic binning.

**(f)** Estimate $n(s, p_c; L)$ for $L = 2^k$ for $k = 4, \ldots, 9$. Use this plot to estimate $\tau$.

**(g)** Can you estimate the scaling of $s_\xi \sim |p - p_c|^{-1/\sigma}$ using this data-set?

Hint 1: Use $n(s,p)/n(s,p_c) = F(s/s_\xi) = 0.5$ as the definition of $s_\xi$.

### Mass scaling of percolating cluster

**(h)** Find the mass $M(L)$ of the percolating cluster at $p = p_c$ as a function of $L$, for $L = 2^k$, $k = 4, \ldots, 11$. Plot $\log(M)$ as a function of $\log(L)$ and determine the exponent $D$.

### Finite Size Scaling

In this exercise we will use a finite size scaling ansatz to provide estimates of $\nu$, $p_c$, and the average percolation probability $\langle p \rangle$ in a system of size $L$.

We define $p_{\Pi=x}$ so that

$$\Pi(p_{\Pi=x}) = x \ ,$$

notice that $p_{\Pi=x}$ is a function of system size $L$ used for the simulation.

**(i)** Find $p_{\Pi=x}$ for $x = 0.8$ and $x = 0.3$ for $L = 25, 50, 100, 200, 400, 800$. Plot $p_{\Pi=x}$ as a function of $L$.

According to the scaling theory we have

$$p_{x_1} - p_{x_2} = (C_{x_1} - C_{x_2})L^{-1/\nu} \ .$$

**(j)** $\log(p_{\Pi=0.8} - p_{\Pi=0.3})$ as a function of $\log(L)$ to estimate the exponent $\nu$. How does it compare to the exact results.

In the following, please use the exact value of $\nu$.

The scaling theory also predicted that

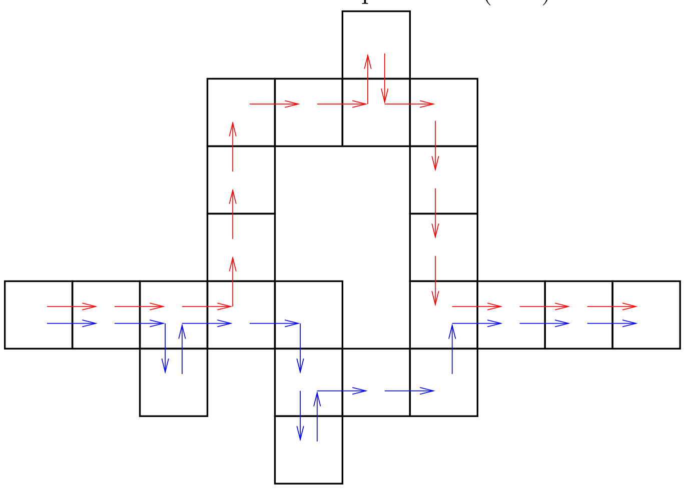
$$p_{\Pi=x} = p_c + C_x L^{-1/\nu} \ .$$

**(k)** Plot $p_{\Pi=x}$ as a function of $L^{-1/\nu}$ to estimate $p_c$. Generate a data-collapse plot for $\Pi(p, L)$ to find the function $\Phi(u)$ from the lecture notes.

### Singly connected bonds

We have provided a subroutine and an example program that implements the left/right-turning walker algorithm. The algorithm works on a given clusters. From one end of the cluster, two walkers are started. The walkers can only walk according to the connectivity rules on the lattice. That is, for a nearest-neighbor lattice, they can only walk to their nearest neighbors. The left-turning walker always tries to turn left from its previous direction. If this site is empty, it tries the next-best site, which is to continue straight ahead. If that is empty, it tries to move right, and if that is empty, it moves back along the direction it came. The right-turning walker follows a similar rule, but prefers to turn right in each step. The first walker to reach the other end of the cluster stops, and the other walker stops when it reaches this site.

The path of the two walkers is illustrated in the figure below. The sites that are visited by both walkers consitute the singly connected bonds. The union of the two walks consitutes what is called the external perimeter (Hull) of the cluster.



**(l)** Run the program `exwalk.m` to visualize the singly connected bonds. Can you understand how this algorithm finds the singly connected bonds? Why are some of the bonds of a different color?

**(m)** Find the mass, $M_{SC}$, of the singly connected bonds as a function of system size $L$ for $p = p_c$ and use this to estimate the exponent $D_{SC}$: $M_{SC} \propto L^{D_{SC}}$. Can you find the behavior of $P_{SC} = M_{SC}/L^d$ as a function of $p - p_c$?

### Alternative 1: Flow on fractals

In this exercise we will use and modify the program `exflow.m` to study flow on a spanning percolation cluster. This program takes as an input an array of (site) conductivities for each individual site and calculates the local current in each bond connecting two sites in the lattice. Most of the program is used to set up the solution of the linear problem for the local currents, given as the solution to Kirchoffs equations. (Program provided by Marin Søreng). We want to use this program to find the behavior of the conductivity near $p_c$.

**(n)** Run the example program to visualize the currents on the spanning cluster.

**(o)** Modify the program to find the singly connected bonds, the backbone and the dangling ends of the spanning cluster and find their dimensionality.

**(p)** Find the conductivity as a function of $p - p_c$. Determine the exponent $\tilde{\zeta}_R$ by direct measurement. Find the conductivity at $p = p_c$ as a function of system size $L$.

### Alternative 2: Random walks on the spanning cluster

In this exercise we will use and modify the program `testpercwalk.m` to study random walks in percolation systems, and on the spanning cluster in particular. We want to find the dimension $d_w$ of a two-dimensional random walk on the spanning cluster.

**(q)** Find the distance $\langle R^2 \rangle$ as a function of the number of steps $N$ for random walks on the spanning cluster for $p > p_c$. Plot $\log\langle R^2 \rangle$ as a function of $N$ for various values of $p$. Can you produce a data-collapse plot for $\langle R^2 \rangle$ as a function of $N$?

**(r)** Can you find the behavior of the correlation length $\xi$ from this plot? Discuss the behavior of the characteristic cross-over time $t_0$ based on the plot. Find the dimension, $d_w$ of the walk, from the relation $\langle R^2 \rangle \propto N^{2/d_w}$.

---

End of Project 3