Continuum Percolation

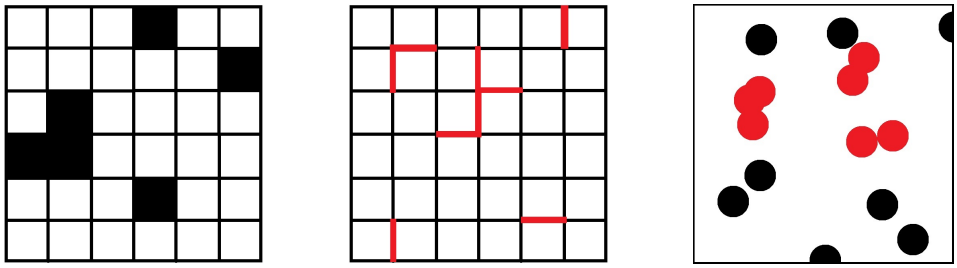
Kamil Dzikowski, Rachel Hendrikse, Luke Manchester,
Andrew Patterson, David Thomas

March 15th 2016

Abstract

Percolation theory describes the connectedness of a random graph. A dimensionless quantity proportional to the area and density of objects in a region known as the microdensity η, is introduced. According to percolation theory, there exists a critical value of microdensity η_c, above which there exists an infinite connected graph. In this model, a phase transition is equivalent to percolation of a 2D system with disk shaped objects (all of which are the same size). This critical value is found to be $\eta = 1.06^{+0.04}_{-0.04}$. This paper also calculates η_c for systems with disks of variable sizes. An analytical approach for continuum percolation threshold calculation is proposed, giving the value of $\eta_c = 1.06$ for circles of constant radius.

Contents



(a) Figure representing site percolation. Black squares are 'filled'.

(b) Figure representing bond percolation. Red bonds are 'open'.

(c) Figure representing continuum percolation. Red circles form 'clusters'.

Figure 2.1: Figures showing different types of percolation.

1 Introduction

The origin of Percolation Theory can be traced back to World War Two when it was used to describe the formation of large macromolecules from smaller branching molecules as chemical bonds are randomly formed between them. This is a process which can lead to a network of bonds spanning the whole system; also known as gelation [1].

The notion that bond formation is random is a controversial one and randomness lies at the heart of Percolation Theory, thus the applicability of it in this scenario is called in to question. The roots of the theory (particularly in mathematical literature) therefore tend to be associated with the work of Broadbent and Hammersley in 1957 [2] in which they considered the hypothetical flow of fluid particles through a random medium. The mechanism for the flow was previously ascribed to the diffusion process in which the fluid decided where to go; their new treatment however, suggested the randomness in such a process arose courtesy of the medium dictating the path of the particles. This new situation was called the Percolation Process due to the resemblance of the fluid spread to coffee flow in a percolator [3]. Since then Percolation Theory has been used to model an incredibly wide variety of situations and processes from galactic formation [3] to global terrorism [4].

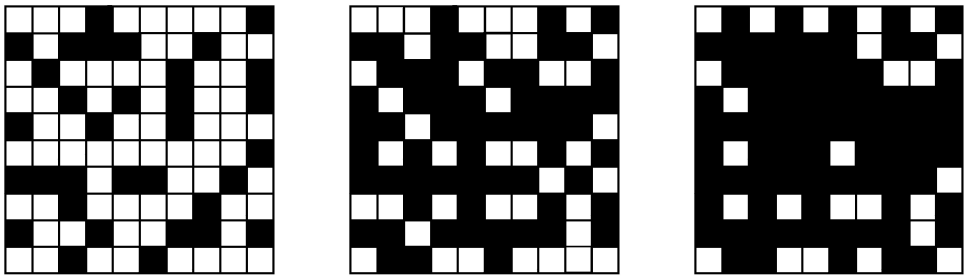
2 Percolation Theory

Percolation theory describes how systems of large numbers of objects with some spacial extent connect and form groups of objects known as *clusters* [5]. There are 3 main types of percolation theory: site, bond and continuum. These different types are illustrated in figure 2.1. Each of these types have slightly different properties due to the nature of the percolation.

2.1 Types of Percolation

Site and bond percolation occur on a regular lattice, whereas continuum percolation can occur with irregular spacings between the objects.

In site percolation the objects in question are the 'spaces' between the lattice, i.e. a square in a 2D square lattice. These sites are then 'filled' with a universal probability p.



(a) $p = 0.33$. **(b)** $p = 0.59$. **(c)** $p = 0.75$.

Figure 2.2: Figure showing site percolation for different values of p.

If two such sites are in contact with each other, i.e. they are adjacent, then they form a cluster. For high enough p, this will result in multiple clusters of varying sizes which can even become large enough to span the space, that is, there exists a path from one side to the other travelling on only filled sites. Figure 2.2 compares different values of p for site percolation on a 2D square lattice. An example of this would be filling a box with plastic and metallic spheres with a probability p that a sphere is metallic; then, for high enough p, there will be a 'spanning cluster' of metallic spheres and a current will be able to flow from one side of the box to the other [5].

Bond percolation follows a similar procedure however, instead of the site being filled, it is the connections between lattice vertices that are either 'open' (or 'closed') with probability p (or $1 - p$). As with site percolation, the higher the probability of a bond being open, the larger the clusters will be. For high enough p, a spanning cluster will again form. Consider a frame in a window mapping out a square lattice; then cut each bond with a probability p. For high enough p a spanning cluster [of broken bonds] will be formed and the window frame will fall apart [5].

The idea of percolation theory is that above some critical probability p_c, a spanning cluster will be formed. As the probability is increased from $p < p_c$ to $p > p_c$, the system undergoes a discrete change of state from one in which no spanning cluster exists to a state containing a spanning cluster [6]. This jump in state is referred to as a *phase transition*. The point at which this phase transition occurs is known as the *percolation threshold* and it is the point at which the system is said to percolate. At the threshold $p = p_c$, there is a finite probability (between 0 and 1) that a spanning cluster exists.

Bond and site percolation are examples of discrete percolation theory where the underlying points all form lattices. This can be generalised to a continuous model known as continuum percolation in which the points are randomly placed in continuous space, often with objects centred on these points. If these objects intersect, they form a cluster. In this way, the percolation threshold is dependent upon the size and shape of the objects concerned. Circles and squares are commonly used in 2D, with spheres and cubes being most common in 3D. The percolation threshold defines the point at which the system will undergo a phase transition and a spanning cluster of infinite size exists beyond this threshold.

For some percolation models the percolation thresholds can be calculated exactly, whereas for others this is impossible. The critical probability for bond percolation on a square lattice is exactly $p_c = 0.5$ [5], but for site percolation on a 2D square lattice the probability is $p_c \approx 0.59$ [5]. This is illustrated on a finite lattice in figure 2.2b. Models in continuum percolation tend to be less likely to have exact values of the percolation

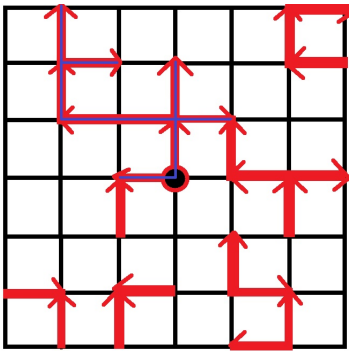


Figure 2.3: Directed percolation on a 2D lattice. Red arrows indicate direction of flow. Blue line indicates flow route originating from centre circle.

threshold and are harder to estimate due to the randomness in the positions of the objects.

2.2 Directed Percolation

Directed percolation is illustrated on a 2D lattice in figure 2.3. The lattice looks like bond percolation, however the open lattice sites have a preferred direction. Flow between lattice sites can happen if the bond is open and in the correct direction. The lattice bond in figure 2.3 has an overall preferred direction of upwards. Common systems that model percolation with a preferred direction include fluid flow through a porous rock (the preferred direction being due to the influence of gravity) and the flow of electricity through networks of resistors. This type of percolation is useful in the modelling of Dutch Elm Disease, with a preferred direction arising from the influence of the wind direction and speed.

2.3 Continuum Percolation Definitions

For continuum percolation, certain parameters can be introduced such as number density ρ, defined as

$$\rho = \frac{N}{A}, \tag{2.1}$$

where A is the area of the system and N is the total number of objects i.e. (in this case) number of trees. The microdensity[1] η, can also be introduced. It is defined as

$$\eta = \frac{aN}{A} = \rho a \tag{2.2}$$

where a is the object area i.e. the area of a circle or square [7]. η_c will be the critical value of microdensity at the percolation threshold, above which an infinite spanning cluster exists. The results can then be characterised by either the value of η_c, or the value of the reduced critical area fraction Φ_c. An equation for Φ in terms of the microdensity η is derived in section 2.4. The exact values are obtained by analysing the behaviour of the microdensity at the threshold in the limit of the object size tending to zero. This is

[1]Known as the dimensionless density (or reduced number density) in other literature, but will be referred to here as the microdensity.

equivalent to taking the size of the forest A and the number of objects N to infinity,

$$\eta_c(a=0) = \lim_{N,A\to\infty} \eta_c(\rho). \tag{2.3}$$

In practise this is realised by extrapolating the dependence of the microdensity needed for percolation on the size of the forest to objects of zero size.

2.4 Calculating Parameters for Percolation of Circles

First, choose a tree. For any other tree, in order to be connected to this one, its centre needs to be closer to it than twice the radius r. The probability of this, in the units of forest area, is given as

$$p = \pi(2r)^2 = 4\frac{\eta}{N}. \tag{2.4}$$

Since there are $N-1$ other trees, the chance of k of them to be connected to it (hence it having a degree k) is

$$P(k) = \binom{N-1}{k} p^k (1-p)^{N-1-k} = \binom{N-1}{k} (4\frac{\eta}{N})^k (1-4\frac{\eta}{N})^{N-1-k}, \tag{2.5}$$

which, as N tends to infinity, becomes

$$P(k) = e^{-4\eta}\frac{(4\eta)^k}{k!}. \tag{2.6}$$

Therefore giving a poissonian degree distribution with mean $\lambda = 4\eta$. Furthermore the critical area fraction can be calculated by noticing that as circles are added, the expected value of the fractional area overlap between a new circle and all circles plotted before is the fraction of the forest they cover. This suggests a recursive formula

$$\Phi_N = \Phi_{N-1} + \pi r^2 - \Phi_{N-1}\pi r^2, \tag{2.7}$$

which can be easily transformed into explicit form

$$\Phi_N = 1 - (1-\pi r^2)^N = 1 - (1-\frac{\eta}{N})^N. \tag{2.8}$$

As N tends to infinity, this formula tends to

$$\Phi = 1 - e^{-\eta}. \tag{2.9}$$

3 Method

3.1 Contact Disease

For the contact disease an adaptation of the Newman-Ziff algorithm was used. The Newman-Ziff algorithm checks whether trees belong to any clusters as it generates new trees, increasing the efficiency of the code in comparison to one which generates all the trees before looking for connections.

In short, the program randomly generates x and y positions for the trees with values between 0 and 1 (for simplicity). Associated with this position is a radius chosen from

a log-normal distribution. This models a real forest due to the many different types and ages of tree present in them. Each tree is also assigned a different *cluster number* which identifies the cluster to which it belongs.

If the circle from a new tree overlaps with an existing one it is assigned to the existing tree's cluster. If it overlaps with multiple trees belonging to different clusters, all these clusters merge into one. The cluster to which a tree belongs is changed by simply changing the tree's cluster number to that of its new cluster. This idea is shown in figure 3.1 as a flowchart. When all the trees have been generated and the clusters found, a graph of this information can be generated using the mathematical plotting module `matplotlib`. Using this, the (x, y) positions of each tree and circles with the radii of each tree at these positions are plotted. The clusters can then be given their own colours, as shown in figure 3.2.

Using this finite region to look for spanning clusters will not be entirely accurate because that would mean approximating the span of infinite space to that of spanning finite space. To fix this, the infinite space can be approximated by a finite space with *periodic boundary conditions* applied.

To do this, the program employs a periodic distance function. This means that if any circles are partly outside the finite region (1x1 grid), the segment outside will be mapped back to the opposite edge, thus giving the impression of circle sticking *into* the finite region, this is shown in figure 3.2. This allows a tree near the edge of the area to be connected to one near the opposite edge. See section 3.2 for details on implementation into the code.

These clusters may form loops which provide a useful way to determine whether an infinite spanning cluster exists. To form a loop, one of two things must have happened:

1. A finite loop must have formed in the middle of the region, which does not span the region.

2. An infinite loop wrapping around the edges of the region, using the periodic boundary conditions, must have formed.

The program can check which of these two cases occurs when it finds a loop because it also keeps track of the vector from the original tree to its current tree via the path of trees it has taken. So if tree c is in the same cluster as tree a but the two aren't connected, and to reach c the program had to go via tree b, then the vector $\overrightarrow{AC}$ is given by

$$\overrightarrow{AC} = \overrightarrow{AB} + \overrightarrow{BC}. \tag{3.1}$$

Then, in case 1, the sum of these vectors when it reaches a tree already in a cluster will simply be equal to zero. In the infinite cluster case, the summed vector will have looped around the edge of the region. Thus, when it reaches a tree already in the cluster the vector will not be zero but rather a multiple (in the x and/or y directions) of that tree's original vector. Graphically, the summed vector will be in same position as the tree, but one region across. This second type of loop is shown in figure 3.3.

When a spanning cluster is found, the code stops running and then approximates the microdensity using equation 2.2, taking the area a to be the area of a circle with a radius equal to the mean root radius and N being the number of trees generated when a spanning cluster was found.

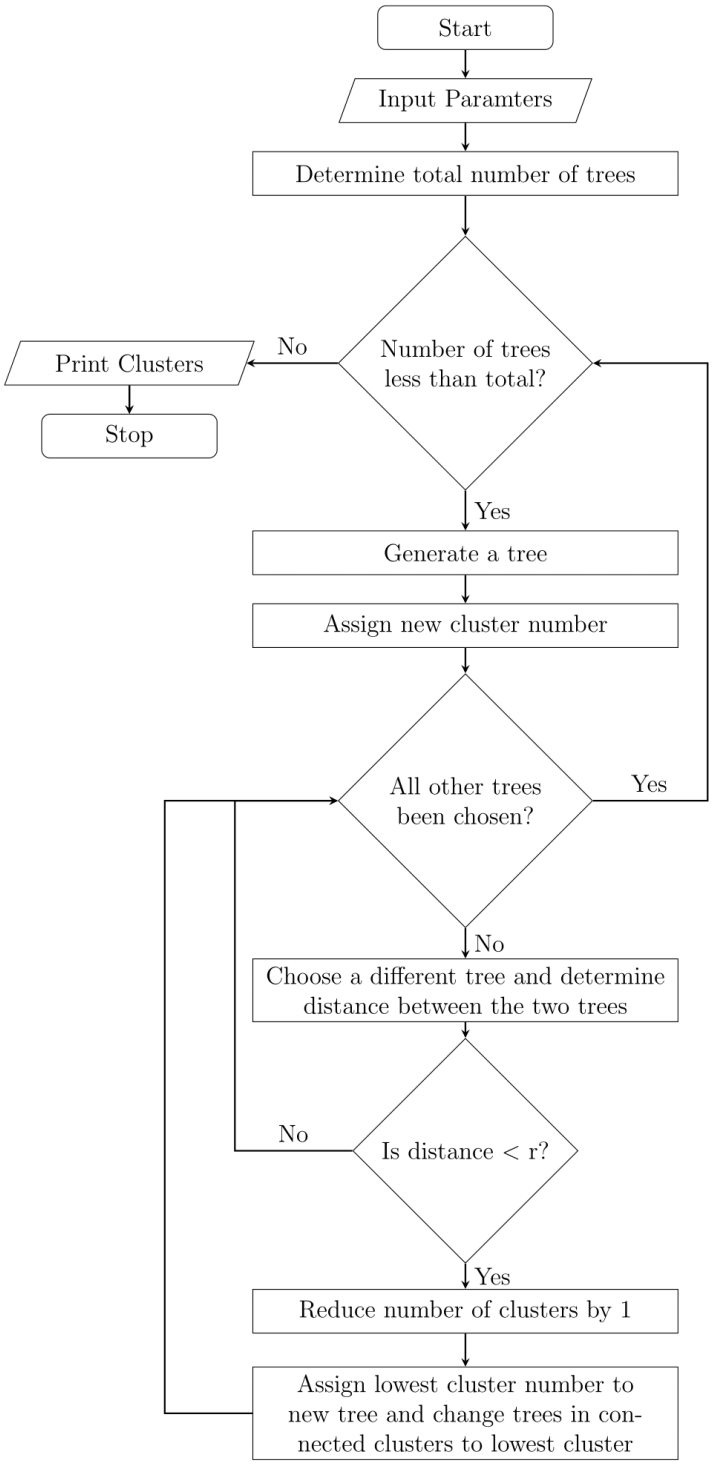


Figure 3.1: Flowchart showing the algorithm for the contact disease.

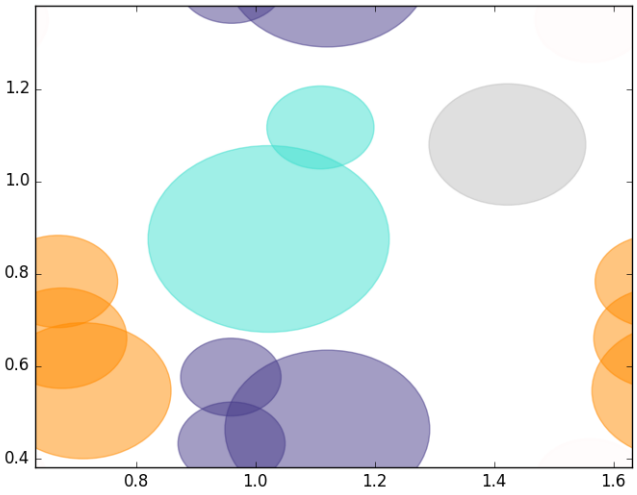


Figure 3.2: Graph showing plots of circles around trees, where circles with the same colour belong to the same sector. The periodic boundary conditions can also be seen.

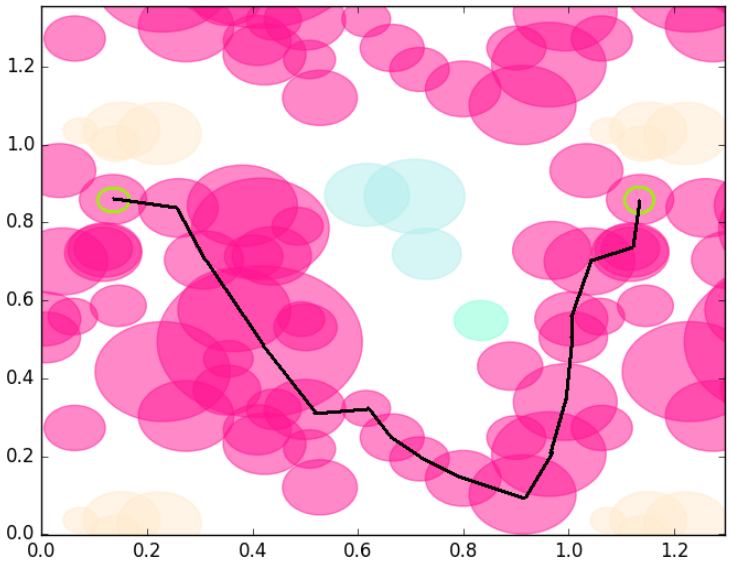


Figure 3.3: Graph illustrating the vector from the cluster origin back to the same tree in the cluster having used the periodic boundary conditions. The vector is the same as the tree's original vector but the x co-ordinate is bigger. Origin tree is circled in green.

Average radius	Time on Single Core (s)	Time on Multiple Cores (s)
0.03	2.13	1.20
0.02	7.48	2.76
0.01	114.16	27.80

Table 3.1: Tables showing the time advantage of running the code on multiple cores. Times shown for six runs.

3.2 Periodic Boundary Conditions

An early issue with non-contact transmission was found with the edges of the forest. With a wind bias in the positive x direction, the majority of the sectors are aligned to the right and, because the trees are positioned randomly in the forest, the region close to $x = 0$ was sparsely populated. This had the additional effect of creating a disallowed region for spanning the forest, one which was solved by use of a periodic distance function:

```
def Pdist(a,b):  ##Periodic Distance Function
    dx=b[0]-a[0]  #x pos of first tree - x pos of second tree
    dy=b[1]-a[1]  #y pos of first tree - y pos of second tree
    return ((dx-round(dx))**2 + (dy-round(dy))**2)**.5)
    # dx - round(dx) is equivalent to addition modulo 1 in the set of real
    numbers, returning just the amount the original number was above an
    integer. This will return a distance through an edge, if it is shorter
    than the distance across the region.
```

With this, the cluster calculations allowed trees close to $x = 1$ in the forest to connect to trees close to $x = 0$, or equivalently $y = 1$ and $y = 0$. When graphed however, the problem was still evident. To solve this, it was decided that nine copies of the forest would be plotted in a grid which not only filled in the gap close to $x = 0$, but also allowed convenient viewing of the progression of the disease throughout the forest.

3.3 Multiprocessing

One of the main issues with the analysis of both models came from computing power. This is because `Python` natively only supports the use of a single processor core and modern consumer computers rely on several cores to run reasonably quickly. Further, one paper [20] remarked that in a similar set-up of continuum percolation, their results were only valid for test regions with over 4000 trees. As such, it was decided that a variety of values over and above this level would be tested. Fortunately, when the simulations were performed, it was found that only 1500 trees were needed to get a consistent reading, however, this was not known until all the simulations had been run.

This phenomenal quantity of trees would have taken more weeks than were available for the project to run, given a single computer processor, so all the processing power available to needed to be used. This meant getting `Python` to run on multiple cores.

Use of the `Python` module `multiprocessing` allowed the code to be run five times quicker. The module opens five copies of `Python` within the main process and distributes them evenly to the cores. A comparison of the times taken to run the program on single and multiple cores is shown in table 3.1. However, even with this improvement, a third

of the data needed took approximately 36 hours run. A further issue arose with runs this long. The algorithm generated processes normally, but would not close them after they had finished running. This meant that after approximately 20 hours there were nearly two thousand open `Python` processes running, causing a significant detriment to performance. Even with these issues, the dataset was eventually generated and associated values calculated.

4 Theoretical Calculation of the Threshold

4.1 Approach

It is possible to calculate exactly, or at least approximately, the percolation thresholds of many lattices by exploiting their symmetry and analysing the influence on the spanning cluster from a single unit cell. In continuum percolation however, no such symmetry exists and a different approach is necessary; in particular one which sees the result of continuum percolation as a random geometric graph. A method which can serve as a starting point for such an approach is introduced here which involves the adaptation of methods used in analysis of random graphs and random networks [21]. By ignoring cycles, it has been found that random graphs resulting from the Erdős-Rényi model have poissonian degree distribution and contain an infinite cluster if its expected value is greater than 2 [21]. Geometric random graphs resulting from continuum percolation also have poissonian degree distribution, see section 2.4, but the essential difference is that cycles cannot be ignored. However if the objects are roughly spherical, simple geometry, details in appendix A, suggest that chordless cycles of length bigger than 3 are very uncommon and can therefore be neglected without producing much error. To quantify the contribution of order 3 cycles (triangles) and cliques that result from their clustering, a set of dimensionless parameters q_i expressing the probability that adding a neighbour to a member of $(i-1)$-vertex clique results in a formation of i-vertex clique, are introduced. This allows investigation into the properties of clusters in geometric random graphs.

4.2 Analysis of Clusters

The starting point is to look for the dependence of the expected cluster size n as a function of mean degree λ. First of all, number of edges (2-vertex cliques) in a graph of n vertices and degrees d is simply

$$C_2 = \frac{1}{2}\sum_i (d_i) = \frac{n}{2}\langle d\rangle \tag{4.1}$$

where $\langle d\rangle$ is the expected value of d. Furthermore, each vertex has d_i edges and q_3 of its remaining $d_i - 1$ edges lead to a shared neighbour of this connection, forming triangles. At the same time each edge has two vertices and each triangle contributes 3 shared neighbours. Hence the total number of triangles (3-vertex cliques) in a graph is given by

$$C_3 = \frac{1}{3}\sum_i (\frac{q_3}{2} d_i(d_i-1)) = q_3\frac{n}{3!}\langle d(d-1)\rangle. \tag{4.2}$$

Similarly, each member of an $(s-1)$-vertex clique has $d-s+2$ other edges, each having q_s chance of increasing it to a s-vertex clique (which contributes s cliques of $(s-1)$ vertices).

Repeating that argument s times, we get the total number of s-vertex cliques in a graph

$$C_s = \frac{1}{2}\sum_i (d_i \prod_{k=3}^{s} \frac{q_k}{k}(d_i - k + 2)) = \frac{n}{s!}\langle d \prod_{k=3}^{s} q_k(d - k + 2)\rangle. \tag{4.3}$$

Using the condition for connectedness of a graph with no chordless cycles is

$$\sum_{i=1} (-1)^{i-1} C_i = 1, \tag{4.4}$$

where $C_1 = n$; substituting in equation 4.3 and dividing by n yields

$$\frac{1}{n} = 1 - \frac{1}{2!}\langle d\rangle + \frac{q_3}{3!}\langle d(d-1)\rangle - \frac{q_3 q_4}{4!}\langle d(d-1)(d-2)\rangle + \dots \tag{4.5}$$

Given that d is poisson distributed (section 2.4),

$$\langle \prod_{k=0}^{t} (d-k)\rangle = \lambda^{t+1}, \tag{4.6}$$

this can be substituted in to simplify the result to

$$\frac{1}{n} = 1 - \frac{1}{2!}\lambda + q_3\frac{\lambda^2}{3!} - q_3 q_4 \frac{\lambda^3}{4!} + q_3 q_4 q_5 \frac{\lambda^4}{5!} - \dots \tag{4.7}$$

To find the threshold value λ_c at which n becomes infinite the smallest positive root of this infinite polynomial needs to be found. Using the values of q coefficients calculated in appendix A this can approximated as $\lambda_c = 4.24$ (to 3 s.f.), which yields the critical microdensity $\eta_c = 1.06$. This value differs only by about 6% from the one obtained in literature [19], which is good given that the values of most of the q coefficients are just estimates and could, at least theoretically, be found with much better precision.

4.3 The Influence of the Distribution of Radii

The key fact that allowed for the analysis in the previous section is the poissonian degree distribution of random geometric graphs. However, if percolating circles have variable radii the situation is more complicated and requires redoing of the calculation from section 2.4.

First of all, assume that the mean and variance of the distribution of radii are M and V respectively. Microdensity can now be expressed as

$$\eta = \sum_i \frac{\pi}{A} r_i^2 \approx \pi \frac{N}{A}\langle r^2\rangle = \pi\rho(V + M^2), \tag{4.8}$$

for circles of radius r. Now, to calculate the distribution of degrees, choose a tree of radius r_0. For any other tree, in order to be connected to this one, its centre needs to be closer to it than the sum of their radii. The chance of that is given by

$$p_i = \frac{\pi}{A}(r_0 + r_i)^2 = \frac{\rho}{N}\pi(r_0^2 + 2r_0 r_i + r_i^2), \tag{4.9}$$

which means that,

$$\langle p_i\rangle = \frac{\rho}{N}\pi(r_0^2 + 2r_0 M + M^2 + V) = \frac{\eta}{N}\frac{(r_0 + M)^2 + V}{V + M^2} = \frac{\lambda(r_0)}{N}. \tag{4.10}$$

Since there are $N-1$ other trees, the probability of k of them to be connected to it (hence it having a degree k) is

$$P(k) = \sum_{i_1 \neq i_2 \neq \ldots \neq i_k}^{N-1} (p_{i_1} p_{i_2} \ldots p_{i_k} \prod_{j \neq i_1, i_2, \ldots i_k}^{N-1} (1 - p_j)) = \binom{N-1}{k} \langle \prod_{i=1}^{k} p_i \prod_{i=k+1}^{N-1} (1 - p_i) \rangle. \quad (4.11)$$

Owing to the fact that all radii are independent and the expected value of a product of independent random variables is the product of their expected values, equation 4.11 becomes

$$P(k) = \binom{N-1}{k} \langle p \rangle^k (1 - \langle p \rangle)^{N-k} = \binom{N-1}{k} \frac{\lambda^k(r_0)}{N^k} (1 - \frac{\lambda(r_0)}{N})^{N-k}. \quad (4.12)$$

When N tends to infinity, the above expression tends to

$$P(k) = e^{-\lambda(r_0)} \frac{\lambda(r_0)^k}{k!}, \quad (4.13)$$

hence we get a poissonian degree distribution with the mean equal to

$$\lambda(r_0) = \eta \frac{(r_0 + M)^2 + V}{V + M^2} = \eta \frac{(\frac{r_0}{M} + 1)^2 + c_v^2}{c_v^2 + 1}, \quad (4.14)$$

where $c_v^2 = \frac{V}{M^2}$ and c_v is the coefficient of variation of the distribution of radii. An important point is that even though the probabilities for a given vertex to have a certain degree is poissonian, the distribution of degrees throughout the graph is not. This is because shapes of these individual poisson distributions are different depending on radii of individual circles. Therefore, the actual form of the distribution of degrees (fraction of nodes with degree k) is

$$F(k) = \int_0^\infty e^{-\lambda(r)} \frac{(\lambda(r))^k}{k!} P(r)\, dr. \quad (4.15)$$

This may be a very complicated expression in general, but it still allows the use of the properties of the poisson distribution because the expected value of any function of d is now

$$\langle f(d) \rangle = \sum_{d=0}^{\infty} f(d) F(d) = \sum_{d=0}^{\infty} \int_0^\infty f(d) e^{-\lambda(r)} \frac{(\lambda(r))^d}{d!} P(r)\, dr, \quad (4.16)$$

and assuming uniform convergence of this sequence of integrals

$$\langle f(d) \rangle = \int_0^\infty \langle f(d) \rangle_k P(r)\, dr = \langle \langle f(d) \rangle_k \rangle_r. \quad (4.17)$$

In particular, a form of equation 4.6 still applies but modified as

$$\langle \prod_{k=0}^{t} (d - k) \rangle = \langle (\lambda(R))^{t+1} \rangle_r. \quad (4.18)$$

This allows us to use the same reasoning as in section 4.3 and arrive at a similar result

$$\frac{1}{n} = 1 - \frac{1}{2!} \langle \lambda \rangle_r + q_3 \frac{\langle \lambda^2 \rangle_r}{3!} - q_3 q_4 \frac{\langle \lambda^3 \rangle_r}{4!} + q_3 q_4 q_5 \frac{\langle \lambda^4 \rangle_r}{5!} - \ldots \quad (4.19)$$

However in general the values of q coefficients may now be dependent on the probability distribution of R.

4.4 Log-Normal distribution

In the model we use the log-normal distribution of radii for which raw (crude) moments can be expressed as

$$\langle x^s \rangle = M^s (\sqrt{1+c_v^2})^{s^2-s}, \tag{4.20}$$

which means that

$$\langle (\frac{r}{M}+1)^s \rangle = \sum_{i=0}^{s} \binom{s}{i} \frac{\langle R^i \rangle}{M^i} = \sum_{i=0}^{s} \binom{s}{i} (\sqrt{1+c_v^2})^{i^2-i} = \sum_{i=0}^{s} \sum_{j=0}^{\frac{1}{2}(i^2-i)} \binom{s}{i} \binom{\frac{i^2-i}{2}}{j} c_v^j. \tag{4.21}$$

This leads to

$$\langle \lambda^k \rangle_r = \eta^k \frac{\langle ((\frac{r}{M}+1)^2 + c_v^2)^k \rangle}{(c_v^2+1)^k} = \frac{\eta^k}{(c_v^2+1)^k} \sum_{s=0}^{k} \binom{k}{s} \langle (\frac{r}{M}+1)^{2s} \rangle c_v^{2k-2s}, \tag{4.22}$$

showing that $\langle \lambda^k \rangle$ is just equal to $(4\eta)^k$ (as before) multiplied by a function of c_v,

$$\langle \lambda^k \rangle_r = \frac{\eta^k}{(c_v^2+1)^k} \sum_{s=0}^{k} \sum_{i=0}^{2s} \sum_{j=0}^{\frac{1}{2}(i^2-i)} \binom{k}{s} \binom{2s}{i} \binom{\frac{i^2-i}{2}}{j} c_v^{j+2k-2s}, \tag{4.23}$$

which can be substituted into equation 4.19 along with all of the q coefficients. To get a sense of the dependence this expression can be approximated for small values of c_v,

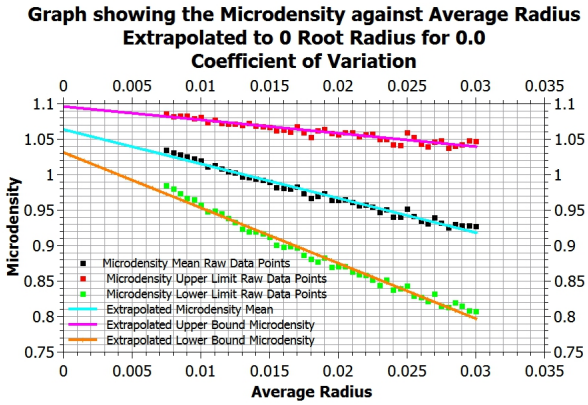
$$\langle \lambda^k \rangle_r = (4\eta)^k (1 + \frac{k}{4}(2k-1)c_v + \frac{k}{32}((k-1)(2k-1)(2k-5)-24)c_v^2 + O(c_v^3)). \tag{4.24}$$

Given that the value of λ is no longer constant and the expected value now needs to be calculated, this adds to an already lengthy calculation, which is not feasible to calculate by hand and so is omitted. The important point is that the threshold value for distributed radii is expected to only depend on c_v.

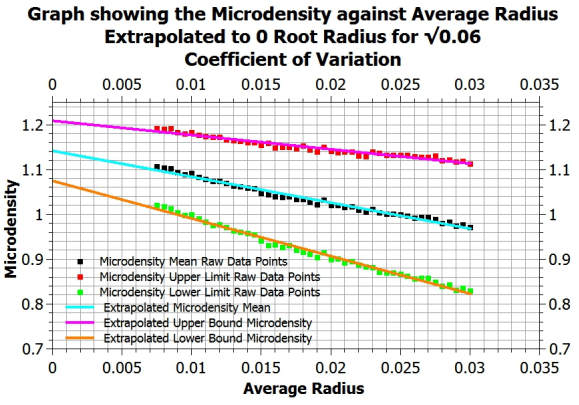
5 Results and Analysis

5.1 Simulation

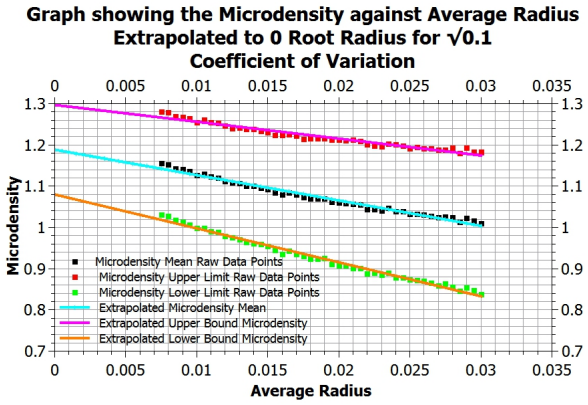
To calculate the critical microdensity for this simulation it was decided to keep the coefficient of variation c_v constant and run through a range of decreasing mean root radii to find the behaviour of η as the average root radius tends to zero. This pattern was investigated for 3 different values of c_v: 0.0, $\sqrt{0.06}$ and $\sqrt{0.1}$. Use of $c_v = 0$ is equivalent to circles of constant radius, allowing for comparison with known values. For each of these 3 coefficients of variation the simulation was run with a the range of mean root radii $0.03 - 0.0075$, each time decreasing in steps of 0.0005. The simulation was carried out 1000 times for each combination of root mean and coefficient of variation. This number of runs was deemed suitable as the time taken to run the program this many times was manageable and the higher the number of runs, the better. The runs for larger mean root size didn't take as long because it took fewer trees to obtain a spanning cluster (≈ 300 trees), whereas for lower mean root size the number of trees needed increased vastly ($\approx$ 5000 trees).



(a) $c_v = 0.0$



(b) $c_v = \sqrt{0.06}$



(c) $c_v = \sqrt{0.1}$

(d) Comparison.

Figure 5.1: Graphs showing data plots of microdensity against average radius for different coefficients of variation. The lines of best fit are also plotted and extrapolated to zero average root radius, hence the lack of points near zero root radius. Also shown is the comparison of mean microdensity for different coefficients of variation.

Each run of the simulation outputs the microdensity; this was then averaged over all of the 1000 runs for each set of input parameters (root mean and c_v). The standard deviation in the microdensity was also calculated allowing for upper and lower bounds on the microdensity to be calculated. The data was plotted using the `QtiPlot` program and is shown in figure 5.1.

From the graphs it can be seen that the data follows an approximately linear trend, hence lines of the form $y = mx+c$ were fitted to the data and these lines were extrapolated to find the value of microdensity at zero average radius. This extrapolation of the data has to be done and cannot be calculated because a circle of 0 radius (or very close to zero in cases of $c_v \neq 0$), corresponds to a point-like object, hence the idea of two circles overlapping no longer makes sense. The range of radii the simulation uses goes as low as possible, given the computational power available. The data points for the mean, upper, and lower limits of the microdensity were fit to lines, the plots of which are shown in figure 5.1. Using these lines it was possible to find the values of interest, the microdensity at zero average radius, these values are shown in table 5.1.

The value for circles of constant radius ($c_v = 0.0$) can be compared with established estimates of this value. From the simulation the value $\eta_c = 1.06$ (3 s.f.) was obtained, while the widely accepted best estimate is $\eta_c = 1.13$ (3 s.f.) [19], thus giving an error

Coefficient of Variation, c_v	Microdensity, η_c, at zero average root radius
0.0	1.06 ± 0.04
$\sqrt{0.06}$	1.14 ± 0.07
$\sqrt{0.1}$	1.19 ± 0.11

Table 5.1: Critical microdensity needed for there to exist an infinite spanning cluster in forests with different root variances.

of 6.2% and so is outside our error range. However, this value compares nicely with the result obtained from the theoretical approach to finding the threshold, shown in section 4, which was found to be $\eta_c = 1.06$. As the coefficient of variation increases it can be seen that the critical microdensity increases to such a point that for $c_v^2 = 0.06,\ 0.1$ the known value of the critical microdensity, $\eta_c = 1.13$, is within the error range and is close to the value for $c_v^2 = 0.06$. The mean microdensities for each the coefficients of variation can be plotted on the same graph to allow comparison and are shown in figure 5.1d. It can be seen for higher c_v the microdensities are higher and increase faster with decreasing average radius (i.e. the lines have a steeper gradient for higher c_v).

References

[1] *Introduction to Percolation Theory*, D. Stauffer, Taylor and Francis Ltd, 1985.

[2] *Percolation Processes*, S. R. Broadbent, J. M. Hammersley, Mathematical Proceedings of the Cambridge Philosophical Society, 1957, Vol.53(3), pp.629-641.

[3] *Applications of Percolation Theory*, M. Sahimi, Taylor and Francis Ltd, 1994.

[4] *Global physics: from percolation to terrorism, guerilla warfare and clandestine activities*, Galam. S, Physica A, **330**, 139-149, 2003.

[5] *Percolation Theory for Flow in Porous Media*, A. Hunt, Lect. Notes Phys. 674, Springer, Berlin Heidelberg, 2005.

[6] *PHYS379: Group Project - Percolation Modelling Project*, H. Schomerus, Lancaster University, 2016.

[7] *Continuum percolation thresholds for mixtures of spheres of different sizes*, R. Consiglio, D.R. Baker, G. Paul, H. E. Stanley, Physica A **319**, 49-55, 2003.

[8] *Guidelines on Brown Root Rot Disease*, Tree Management Office Greening, Landscape and Tree Management Section Development Bureau, The Government of the Hong Kong Special Administrative Region, December 2012.

[9] *Phellinus noxius Brown Root Rot of Fruit and Ornamental Trees in Taiwan*, P. Ann, T. Chang, W. Ko, The American Phytopathological Society, Plant Disease, Vol. 86 No. 8, August 2002.

[10] *The Genetic Structure of Phellinus noxius and Dissemination Pattern of Brown Root Rot Disease in Taiwan*, C. Chung, S. Huang, Y. Huang, et al, Han Y, ed. PLoS ONE, 2015.

[11] *Learning from history, predicting the future: the UK Dutch elm disease outbreak in relation to contemporary tree disease threats*, C. Potter, T. Harwood, J. Knight, I. Tomlinson, Philosophical Transactions of the Royal Society B: Biological Sciences. 2011;366(1573):1966-1974.

[12] *Fungal Wilt Diseases of Plants*, M. E. Mace, A. A. Bell, C. H. Beckman, Academic Press Inc (London) LTD, 1981.

[13] *Tree Disease Concepts*, P. D. Manion, Prentice Hall, Inc, 1971.

[14] *Brown root rot of trees caused by Phellinus noxius in the Ryukyu Islands, subtropicalareas of Japan*, N. Sahashi, M. Akiba1, M. Ishihara, et al, Forest Pathology Volume 42, Issue 5, pp 353–361, October 2012.

[15] *Floristic composition, woody species diversity, and spatial distribution of trees based on architectural stratification in a subtropical evergreen broadleaf forest on Ishigaki Island in the Ryukyu Archipelago, Japan*, S. M. Feroz, M. Wu, S. Sharma, Tropics Vol. 18, November 1, 2009.

[16] *At the Root of it.* URL: `http://www.isa-arbor.com/education/resources/educ_Portal_RootGrowth_AN.pdf`

[17] *Average Weather for Isle of Man, United Kingdom.* URL: `https://weatherspark.com/averages/28738/Isle-of-Man-United-Kingdom`

[18] *An encounter rate model of bark beetle populations searching at random for susceptible host trees*, J. A. Byers, Ecological Modelling, **91**, 57-66, 1996.

[19] *Continuum percolation threshold for interpenetrating squares and cubes*, D. R. Baker Gerald Paul, S Sreenivasan, H. E. Stanley, Phys. Rev. E **66**, 046136, 2002.

[20] *Continuum percolation thresholds in two dimensions*, C. Mertens, c. Moore, Phys. Rev. E **86**, 061109, 2012.

[21] *Complex Networks Structure, Stability and Function*, R. Cohen, S.Havlin, Cambridge University Press, 2007.

[22] *Lens.* URL: `http://mathworld.wolfram.com/Lens.html`

[23] *Area of Common Overlap of Three Circles*, M.P. Fewell, Maritime Operations Division, Defence Science and Technology Organisation, 2006.

[24] *Circle-Circle Intersection.* URL: `http://mathworld.wolfram.com/Circle-CircleIntersection.html`

A Calculation of q Coefficients

The coefficients q_i express the probability that an $(i-1)$-vertex clique in a geometric random graph becomes an i-vertex clique by adding a neighbour to one of its constituent nodes. Here follows an attempt to calculate, or estimate, their values for circles of constant radii r. To do that we notice that for a new circle to be connected to $(i-1)$ existing ones it centre must fall within the distance of $R = 2r$ to every single one of them, or in other words onto the intersection of circles with radii R. Since it is equally likely to be at any point inside one of the constituent circles (of radius R), the value of q_i is given by the expected value of the ratio of the intersection of $i-1$ circles and the area of a single circle.

A.1 Calculation of q_3 and q_4

Union of two circles of radii R and separation d is a lens: $L(R, R, d)$. Using the formula from [22], the fraction of the circles area covered by a lens can be expressed as

$$F_2(x = \frac{d}{R}) = \frac{1}{\pi R^2} A_L(R, R, d) = 1 - \frac{2}{\pi} \arcsin(\frac{x}{2}) - \frac{1}{\pi} x \sqrt{1 - \frac{x^2}{4}}. \tag{A.1}$$

Probability density of separation is linear in two dimensions, so normalization requires $P(x)$ to be equal to $2x$. Then we can get the result as

$$q_3 = \langle F_2(x) \rangle = \int_0^1 F_2(x) 2x dx = 1 - \frac{3}{4} \frac{\sqrt{3}}{\pi}. \tag{A.2}$$

Interestingly,

$$F_2(\langle x \rangle) = F_2(\frac{2}{3}) = 1 - \frac{2}{\pi} \arcsin(\frac{1}{3}) - \frac{4}{9} \frac{\pi}{\sqrt{2}}, \tag{A.3}$$

is just 0.5% different, giving

$$\langle F_2(x) \rangle \approx F_2(\langle x \rangle). \tag{A.4}$$

This fact can be used to estimate the value of q_4. Calculating the area of the union of more than 2 circles is a very complicated geometrical problem, so instead of doing it directly, a reasonably correct value can by obtained by using an approximation analogous to the one above,

$$\langle F_3(x, y, z) \rangle \approx F_3(\langle x \rangle, \langle x \rangle, \langle x \rangle) = F_3(\frac{2}{3}, \frac{2}{3}, \frac{2}{3}). \tag{A.5}$$

Using the formula from [23],

$$q_4 \approx F_3(\frac{2}{3}, \frac{2}{3}, \frac{2}{3}) \approx 0.437. \tag{A.6}$$

A.2 Estimation of Higher Coefficients

Estimating the expected value of the area of the intersection of four circles is an even more complicated problem, so a different approximation is needed. Therefore, the overlap of three circles is approximated as a smaller circle, giving the expected value of its overlap with another big circle as

$$q_5 \approx \frac{1}{\pi R^2} \langle A_L(R, R\sqrt{q_4}, d) \rangle_d. \tag{A.7}$$

If the distance d is smaller than the difference in radii, the area of intersection is simply the area of the smaller circle. Otherwise, it is given by the formula from [24]. The probability density of separation is once again linear,

$$q_5 \approx \frac{1}{\pi R^2} \int_{R(1-\sqrt{q_4})}^{R\sqrt{q_4}} A_L(R, R\sqrt{q_4}, d) P(d) dd + q_4 \int_0^{R(1-\sqrt{q_4})} P(d) dd \approx 0.32. \tag{A.8}$$

Using the same procedure again we can arrive at: $q_6 \approx 0.29$ and $q_7 \approx 0.27$. Finally, to get an estimate for the q_i coefficients for higher values of i, we can look for the limiting value, as i tends to infinity. It is straightforward to see that given infinitely many circles all within a distance not greater than the radius, R, from each other, their centres will form a circle with a radius $\frac{R}{2}$. This means that the fraction of the area covered by the intersection and hence the value of q_∞ is exactly 0.25. Given that $q_7 \approx 0.27$, all q_i for $i \geq 7$ will be approximated as $q_i = 0.25$.