NTNU – Trondheim
Norwegian University of
Science and Technology

Optimal Paths in Random Conductor Networks

Morten Stornes

Master of Science in Physics and Mathematics
Submission date: June 2013
Supervisor: Alex Hansen, IFY

Norwegian University of Science and Technology
Department of Physics

NORGES TEKNISK-NATURVITENSKAPELIGE UNIVERSITET

Abstract

Faculty of Natural Sciences and Technology
Department of Physics

Optimal Paths in Random Conductor Networks

by Morten Stornes

In this master thesis, the spanning path through a random conductor network with a quenched threshold distribuion has been studied. An argument is made using a hierarchial model, that for bonds having a piecewise linear character with slopes α before the threshold and β after, with α, $\beta \in [0, \infty)$, it is the ratio $r = \alpha/\beta$ which is important for where the spanning path forms. It is shown that in the limit of $r \to \infty$, the spanning path follows the optimal path through the network, as in the case of a perfect plastic. Numerical results which support the argument are also given. The fracture path in brittle fracture has also been studied in a hierarchy of optimal paths, without leading to any conclusions.

Sammendrag

I denne masteroppgaven har den gjennombrytende stien i et nettverk av ledere med en fastsatt terskel blitt studert. Ved hjelp av en hierarkisk modell, vises det at det for ledere med en karakteristisk strøm med stigningstall α før terskelen og β etter terskelen, er det forholdet $r = \alpha/\beta$ som er avgjørende for hvor den gjennombrytende stien går. Det viser seg at i grensen $r \to \infty$ følger stien den optimale veien gjennom nettverket. Dette er også tilfellet for ledere med en perfekt plastisk oppførsel. Numeriske resultater som støtter argumentene er også gitt. Bruddveien i sprø brudd har også blitt studert i et hierarki av optimale veier, uten å lede til noen konklusjoner.

Preface

This master's thesis is submitted as the final requirement for the grade of MSc. in Applied Physics at NTNU, and was done under the supervision of Professor Alex Hansen. The thesis is a continuation of the project work conducted the in the fall semester of 2012. During this work, it was found, contrary to the hypothesis, that the spanning path in a random conductor network does not follow the optimal path in the plastic-elastic regime. The goal of this thesis has been to further understand spanning paths in the plastic-elastic regime, and relate the results to that of Bingham flow in porous media. Some time was also spent on studying brittle fracture in a hierarchy of optimal paths.

As this is the continuation of the project work last semester, some of the parts in this thesis are very similar to that of the project report. This is mainly a few sections in the theory chapter which are identical, as well as similar descriptions of the model and implementation. Some parts have been updated, such as the algorithm used for finding the optimal path, which changed. The hierarchial model presented in the theory part, was put there as it felt more natural. The numerical results are then used to confirm the arguments made for the model. In reality, the hierarchial model was studied after some interesting initial results were acquired from the simulations, in order to explain these. This was then done in parallel with running more simulations to further validate the findings.

The code used for this thesis, builds upon the code written for the project work. As this has gotten quite long, it has been decided not to include it as an appendix. It will be uploaded along with the thesis when it is delivered on DAIM. Should anyone be interested in the code, or have questions about the work done in this thesis, feel free to e-mail the author at: m.stornes@gmail.com

Morten Stornes

Trondheim, June 2013

Acknowledgements

I would like to thank my supervisor, Professor Alex Hansen, for providing me with an interesting project and helping me out with problems when needed. His insight into the subject has been of great value. I've also really enjoyed the weekly meetings in the porous media flow group, and have benefited from the discussions there. These meetings, in addition to the journal club, have helped me gain a better understanding of the general process of performing research, and I'd like to thank the members for creating an enjoyable and welcome atmosphere.

At last, I'd like to thank my family for their support and my friends for making my time so far in Trondheim, a good one.

Contents

List of Figures

Chapter 1

Introduction

Random conductor networks are an often used system in statistical physics. They can be used as a simplified system to study features in brittle fracture in materials, directed growth of polymers in random media, the flow of a Bingham fluid in porous media [1] or universal effects of disorder in systems. The random conductor network is very useful for this case, as it is fairly simple to implement on a computer, and one just have to solve Kirchoff's laws for the entire network.

When the conductor network is used to study brittle fracture, the computational needs are reduced, as a spring network is replaced by a fuse network, which reduce tensor and vector fields to a scalar field [2]. In the directed polymer growth analog, the network itself can be viewed as the medium the polymer grows through [3], and similar for the Bingham fluid, which flows through channels in the network.

Whereas fracture can be a quite complex process [2–4], that of directed polymer growth is generally simpler, at least in the limit of zero temperature. In this case, the polymer simply follows the optimal path [3, 5]. As per the argumentation in [1], this should also be the case of Bingham flow as the polymer growth can be seen as the case of a Bingham flow with the intensity scale stretched.

In the so-called plastic-elastic regime, the current through an electric bond continue to increase with increasing voltage drop after the bond reach a threshold, but at a slower rate than before. Using the idea of a dual network, this can also be seen as a crude approximation to a shear-thinning fluid, of which the Bingham fluid is an idealized version with a yield stress. The random conductor network then works as an analog to Bingham flow in a porous medium. As the spanning path develops differently depending on the local characteristic, the main focus of this thesis is to study how the spanning

path develops for different parameters in this regime, and how it compares to the optimal paths.

Chapter 2

Theory

2.1 The random conductor network

One possible random conductor network is that of a square lattice of conducting bonds, rotated 45° and situated between two electrodes such as the one in Figure 2.1. Each bond has a conductance g_{ij} and connects two nodes, i and j, with voltages v_i and v_j. Kirchoff's current law states that the sum of the current going through a node is zero, and thus the current going through node i is given by

$$\sum_j g_{ij}(v_j - v_i) = 0 \tag{2.1}$$

where the sum is over all connected neighbours j. Each node is connected to four other nodes, with the exception of the nodes which are considered to lie on the electrodes themselves. As such, Eq. (2.1) is only valid in the case that i is an internal node situated between the electrodes, while j can run over the nodes lying on the electrodes as well. By rearranging the above equation a bit, one gets

$$\sum_j D_{ij} v_j = I_i \tag{2.2}$$

which runs over all internal nodes j. Here, all the voltages v_j referring to nodes belonging to the electrodes have been moved over to the right side of the equation, thus giving the current I_i going to node i from the electrodes for all internal nodes directly connected to one of the electrodes. j is then only run over internal nodes, excluding the need to look at the nodes situated on the electrodes. For all nodes not directly connected to the electrodes, I_i is zero. D_{ij} is here equal to $g_{ij}(\frac{v_i}{v_j} - 1)$. Using Eq. (2.2), the voltage for all internal nodes can be found by running over all internal nodes i, giving a set of coupled

linear equations

$$\mathrm{D}\vec{v} = \vec{I}. \tag{2.3}$$

Here, D is a sparse matrix and can be solved using a matrix solving method, such as the conjugate gradient method.

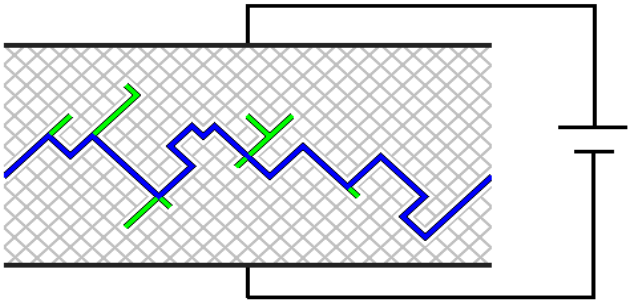


FIGURE 2.1: A conductor network. A square lattice of conducting bonds is rotated by 45° with respect to the electrodes at top and bottom. The network is put under an increasing voltage drop until a spanning path (blue) is developed. Green arms are part of the spanning cluster.

2.1.1 Behaviour of the system

Each voltage node has four bonds connecting it to other voltage nodes, except for the nodes on the conductors. Each bond connects two voltage nodes and is assigned a random threshold t_{ij}. The voltage difference, $v_i - v_j = \Delta v$, over the bond determines the behaviour of it. All bonds are considered to have a conductance α as long as $\Delta v < t_{ij}$. Once Δv reaches above the threshold, the current going through the bond is changed according to the scenario studied, as can be seen in Figure 2.2(a). Dropping the subscripts i and j and using i to denote the electric current going through a bond, the current after the threshold is given by

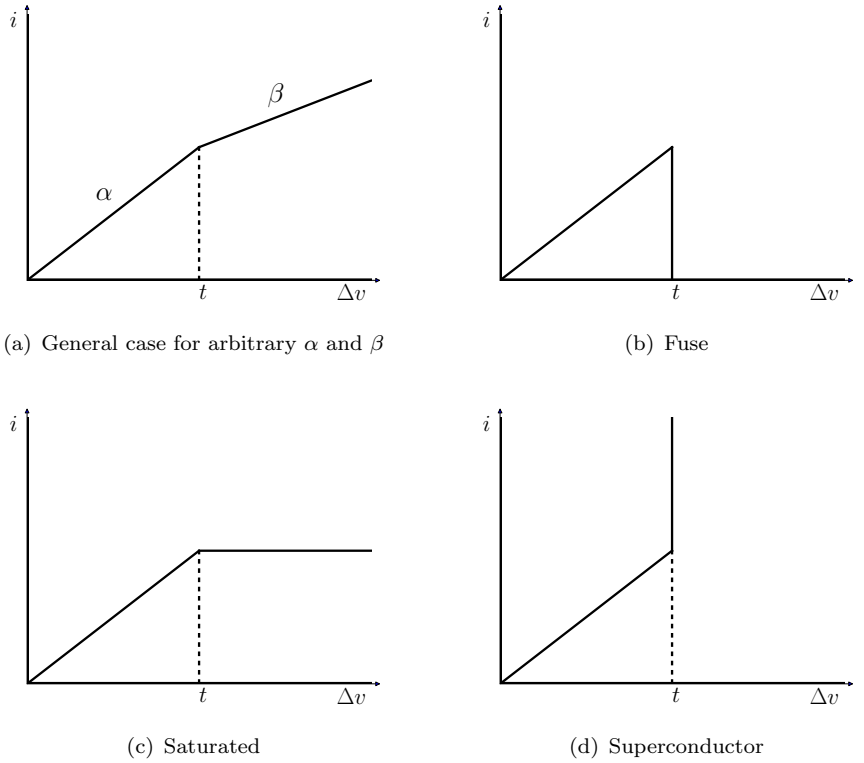
$$i(\Delta v) = \alpha t + \beta(\Delta v - t) = g\Delta v \tag{2.4}$$

with $\alpha \in [0, \infty)$, $\beta \in (-\infty, \infty)$. This gives the conductance

$$g(\Delta v) = (\alpha - \beta)\frac{t}{\Delta v} + \beta. \tag{2.5}$$

It should be noted that the conductance will always be positive or zero, as it is the voltage difference which decides which direction the current is flowing in. In the case of the right side of Eq. (2.5) being negative, g is set to be zero. Thus, for the fuse case, the conductance is set to 0 at the moment the threshold is reached. Such a bond is broken, and can not conduct any current. In the rest of this thesis, any bond which reach its

threshold is considered to be "broken", irrespective of the characteristic of the bond.



(a) General case for arbitrary α and β

(b) Fuse

(c) Saturated

(d) Superconductor

FIGURE 2.2: The current i flowing through a bond under a voltage drop Δv for different cases. (a) shows the general characteristic where the increase in current changes according to the slope β after the threshold t has been reached. (b)-(d) shows the special cases for $\alpha = 1$ and $\beta \to -\infty$ (bonds acting as a fuse), $\beta = 0$ (saturated bonds) and $\beta \to \infty$ (superconducting bonds).

2.1.2 Edge cases

The random conductor model has a few different edge cases which are interesting to study. These can be seen in Figure 2.2(b)-2.2(d). In the case of $\beta \to -\infty$, the bonds act as a fuse. A random conductor network consisting of fuses is often called a random fuse network in the literature. This network model can be used as a simplified model to study fracture phenomena in brittle materials. For the case of $\beta = 0$, the conducting bonds are saturated at the threshold. This behaviour is equivalent to a material with a perfect plastic behaviour. The case $\beta \to \infty$ gives a superconducting bond.

In the case of the random fuse network, bonds will break irreversibly when they reach their thresholds, i.e. $g = 0$ for $\Delta v > t$. As the sum of the current going through the nodes is still zero, all the current going through the broken bond at the time of breaking, will be redirected and go through other bonds. This can in turn cause other bonds to

reach their threshold and break. In the case of brittle fracture, one would often like to study the fracture that breaks the material in two separate pieces. In the fuse model, this is equivalent to a path of broken bonds that spans across the network parallel to the electrodes. Thus, when the path spans across the entire network, there will be no current going through the network, and it is considered broken or disconnected. Due to the bonds redistributing the current after they break, the breakdown of the network can be quite complex due to avalanches and nucleation[4]. To find the spanning path one has to solve Eq. (2.3) for increasing voltage drops on the network and break one bond at a time. After a bond is broken, Eq. (2.3) must be solved again to find the current running through the remaining bonds and see if other bonds break.

Looking at the case of $\beta = 0$, the same spanning path will cause the network to be saturated. That is, when there is a spanning path of bonds having reached the threshold, no more current can go through between the electrodes, irrespective of their voltage difference. The total current going between the electrodes is then given by the sum of the thresholds of all the bonds in the spanning path. In this case, when a bond reach the threshold, there is no extra current being redistributed from the bond when it breaks. For increasing Δv, the surplus current coming in is the only one being redistributed to other bonds. Due to this, the spanning path will be the path that minimizes the current going through the network, i.e. the optimal path.

The case of $\beta = \alpha$ is not that interesting in itself, as a network of conductors. As there is no redirection of currents, the bonds "break" in order of increasing threshold. This is equivalent to standard bond percolation which is described in section 2.3. When bonds break, they do not change the behaviour of the system in any way, and when a spanning path occurs and disconnects the network, there is no change in the current going between the electrodes. However, the case separates the behaviour of the conductance. For $\beta < \alpha$, the conductance through a bond will decrease with increasing Δv. For $\beta > \alpha$, the conductance increase, and will go towards infinity for $\beta \to \infty$. Transposing the result to mechanics with strain (ε) replacing the voltage and stress (σ) the current, as done in [1], the bond can be viewed as a material. Assuming an elastic bond, for $\beta = \alpha$, the bond can be viewed as linear elastic. However, elastic materials do not always show a linear dependence in the stress-strain relationship [6]. If $\sigma \propto \varepsilon^k$ for $k \neq 1$, the material is hyperelastic. For $k < 1$, the material softens for large strain, while it stiffens for $k > 1$. Thus, the case of $\beta \gtrless \alpha$ can be viewed as a crude approximation for hyperelastic stiffening/softening in materials, and $\beta = \alpha$ separates these two behaviours.

The conductor network can also be used as a model for fluid flow through a porous medium. Here, voltage nodes can be considered as pores (with infinite volume) and bonds are links connecting the pores. The electrical current can be replaced by the

volume flow or flux of the fluid, and the voltage drop with a pressure difference. For $\beta < 0$, the fluid will clog as the links fail, and eventually the flow will stop. The path which stops the fluid will then be reminescent of the fracture path in the case of a brittle material. If $\beta = 0$, the links will simply saturate.

Analogous to the hyperelastisc stiffening and softening, the model could also be used as an approximation of a shear-thinning or shear-thickening fluid flow in porous media. In that case, $\beta > \alpha$ approximates a shear-thickening fluid, while $\beta < \alpha$ gives a shear-thinning fluid. In the electrical network, the voltage can be considered an equivalent for the shear stress, and the electrical current is equivalent to the strain rate. The simplest non-newtonian fluid is the Bingham fluid. It is an idealized case of a shear-thinning fluid, and is a solid until it hits a yield threshold, as in Figure 2.3(a), and afterwards it acts as a newtonian fluid with a viscosity μ. Thus, in the electrical Bingham equivalent, $\alpha = 0$ and $\beta > 0$.

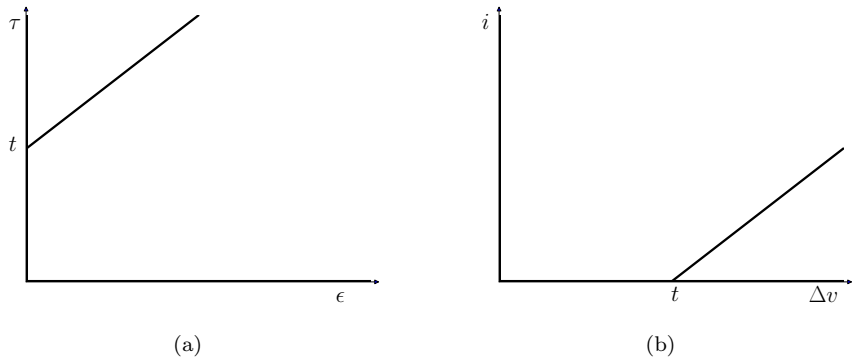


FIGURE 2.3: (a) Characteristic of a Bingham fluid with shear stress τ, strain rate ϵ and yield threshold t. (b) Bingham equivalent behaviour in conducting elements in the electrical network.

2.2 Dual network

While the superconducting case is not possible to simulate directly with the approach used here, one can instead look at a dual network with the reciprocal conductivity of the original network. The derivation here closely follows that of Straley[7].

In an ordinary square network with node potentials v_{ij}, the quantities Δ_i and Δ_j are defined such that $\Delta_i v_{ij} = v_{i+1,j} - v_{ij}$ and $\Delta_j v_{ij} = v_{i,j+1} - v_{ij}$. Similarly, the conductance of a bond protruding from a node (i, j) is given as $\sigma_i(i, j)$, where the subscript denotes if it is the bond going out in the i or j direction. The coordinates are not explicitly written when it denotes a current through a bond, so $\sigma_i \Delta_i v_{ij}$ is the current in the bond

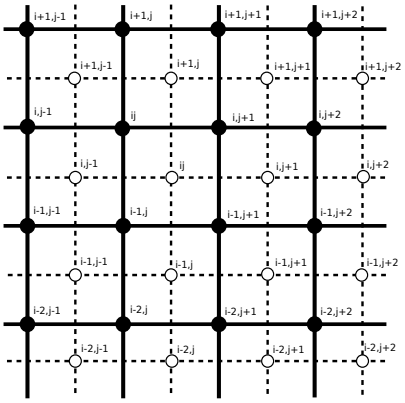


FIGURE 2.4: A square network with its dual network drawn with dashed lines. The dual network is shifted by $(-\frac{1}{2},\frac{1}{2})$ with respect to the original network.

between nodes (i,j) and $(i+1,j)$ with σ_i denoting the conductance of said bond. Thus, Kirchoff's laws take the form

$$\Delta_i v_{i-1,j} + \Delta_j v_{ij} - \Delta_i v_{i-1,j+1} - \Delta_j v_{i-1,j} = 0 \tag{2.6}$$

$$\sigma_i \Delta_i v_{ij} + \sigma_j \Delta_j v_{ij} - \sigma_i \Delta_i v_{i-1,j} - \sigma_j \Delta_j v_{i,j-1} = 0 \tag{2.7}$$

where Eq. (2.6) says that the sum of the voltage differences in a closed loop is zero, and Eq. (2.7) is the conservation of current through a node. With the dual network defined as in Figure 2.4, the voltage drop between two nodes on the dual network is defined as

$$\begin{aligned} \Delta_i v'_{ij} &= \sigma_j \Delta_j v_{ij} \\ \Delta_j v'_{ij} &= -\sigma_i \Delta_i v_{i-1,j+1}. \end{aligned} \tag{2.8}$$

For this definition to be valid, it has to fulfill Eq. (2.6). Inserting into Eq. (2.6) with v' instead of v produces Eq. (2.7). Thus, Kirchoff's voltage law on the dual network corresponds to the current law on the original network. Similarly, using

$$\begin{aligned} \Delta_i v_{ij} &= -\tfrac{1}{\sigma_i(i,j)} \Delta_j v'_{i+1,j-1} &&= -\sigma'_j \Delta_j v'_{i+1,j-1} \\ \Delta_j v_{ij} &= \tfrac{1}{\sigma_j(i,j)} \Delta_i v'_{ij} &&= \sigma'_i \Delta_i v'_{ij} \end{aligned} \tag{2.9}$$

in Eq. (2.6), produces Eq. (2.7) for the dual network. Thus, the conductance of a dual bond is the reciprocal conductance of the original bond it crosses.

The negative relation between the voltage in the i-direction of the original network and the j-direction of the dual network allows for an interesting property. This causes the net current direction to change in the dual network. Assume for instance that all the current

flows in the direction of increasing i and j on the original network, i.e. $v_{ij} > v_{i+1,j}$ and $v_{ij} > v_{i,j+1}$, causing the current to flow upwards and right in Figure 2.4. Then, due to the negative relation between the i-direction on the original network and the j-direction on the dual network, the dual current will flow upwards and left, i.e. $v'_{ij} > v'_{i+1,j}$, $v'_{ij} < v'_{i,j+1}$. The net current direction of the dual network is thus perpendicular to that of the original network. By rotating the network 45° counterclockwise, the net current in the original network will flow upwards, while in the dual network it will flow towards the left, as seen in Figure 2.5.

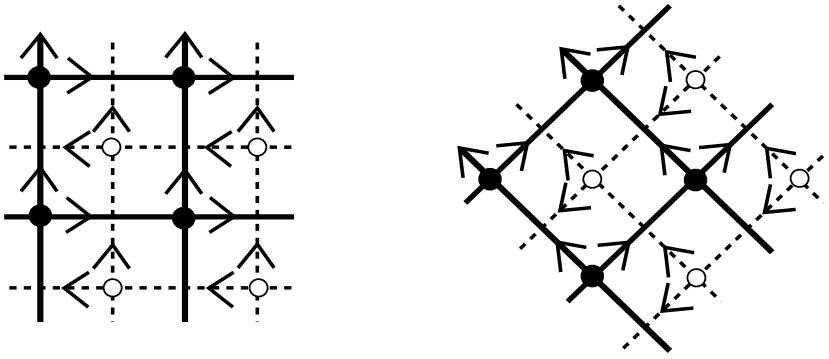


FIGURE 2.5: The networks with the flow direction of the current indicated by arrows. The net currents of the original and dual network are perpendicular. Rotating by 45°, causes the current to flow upwards in the original network, and towards the left in the dual network.

This relation allows one to study the path through a network in one regime by using the dual network and parameters from the reciprocal regime. For instance, if one is to set $\beta = 0$ in the random conductor network described in section 2.1, the spanning path from right to left, which is saturated in the original network, is a spanning path of bonds in the dual network which have reached their threshold and has a conductivity going towards $1/\beta \to \infty$, and are turning superconducting.

2.3 Percolation

Percolation is a much studied problem in statistical physics, and can be used to study different things such as forest fires, flow through porous media, diffusion in random media and conductance in electrical networks [8]. In bond percolation, one has a network of sites which can be connected by bonds with a probability $p \in [0, 1]$. For small values of p, there will be few bonds connecting different sites, and only small, separate clusters of connected sites exist in the system. As p increase, more and more sites will be connected, and clusters will grow larger and join together. At some point, a cluster

connecting two sides of the system will occur. This cluster is known as the *spanning cluster*, as it spans the network. On average, the spanning cluster will occur at the critical probability p_c, also known as the percolation threshold, for finite system sizes. p_c is defined to be the probability needed for an infinite cluster to exist in an infite system. The percolation threshold is dependent on the structure of the network, and for a square lattice, $p_c = 0.500$ for bond percolation [8].

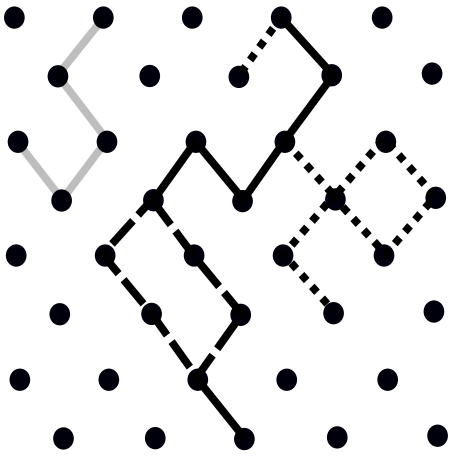


FIGURE 2.6: A percolating system on a rotated square lattice. Black circles are sites which are connected by bonds. The spanning cluster consists of the black bonds, and spans from the top row to the bottom. The whole black lines are singly connected bonds, while the small dashed bonds are dangling arms. The backbone consists of the singly connected bonds and the larger dashed lines. Grey bonds form a separate, non-spanning cluster.

While the spanning cluster is a typical structure to look at in percolation problems, a subset of the spanning cluster, the backbone, is sometimes of more interest. If the bonds are viewed as electrical conducting bonds and the spanning cluster connects two electrodes, the backbone would be conducting all the current between the electrodes. No current would flow through bonds in the dangling arms. The backbone can be defined as consisting of bonds which have at least two independent paths leading to the edges of the system. That is, if one were to put a walker on a bond in the backbone, the walker could reach one side of the system by going in one direction and the other side by going in the other direction, without walking on any same bond twice. If the system is periodic in the direction of the spanning cluster, the backbone connects with itself. In this case, a walker could start anywhere on the backbone and have two independent paths to walk around the system and come back to the starting position. With this definition of the backbone, any path spanning from one side of the system to the other will lie on, and be a subset of, the backbone. Should the backbone only consist of singly linked bonds, the spanning path and the backbone would be identical.

2.4 Conjugate gradient method

The conjugate gradient method is an iterative method which can be used to solve systems of linear equations, and is especially useful when the system is sparse. Only a brief overview of it is given here, as it is explained several places in the literature, see e.g. [9–11].

Given a symmetric and positive-definite matrix A, a system of linear equations is given by

$$\mathrm{A}\mathbf{x} = \mathbf{b}. \tag{2.10}$$

This gives a quadratic function

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}\mathrm{A}\mathbf{x} - \mathbf{b}\mathbf{x}, \tag{2.11}$$

a surface of which the minimum is the solution to Eq. (2.10). The conjugate gradient method tries to minimize Eq. (2.11), which is done when the gradient of the quadratic function is zero, that is, $\nabla f = \mathrm{A}\mathbf{x} - \mathbf{b} = 0$ in the case of A being symmetric and positive-definite. By choosing a starting point, the method calculates successive search directions $\mathbf{d}_i$ which are conjugate to each other. This is done by conjugating the residuals $\mathbf{r}_i$, which are always orthogonal to the previous search directions. Then, a minimizer $\mathbf{x}_{i+1}$ is found by moving from the previous minimizer $\mathbf{x}_i$ in the search direction. A step length α_i is calculated and multiplied with the search direction to ensure that the quadratic function $f(\mathbf{x}_{i+1})$ is minimized.

For an initial guess of $\mathbf{x} = \mathbf{x}_0$, the initial search direction $\mathbf{d}_0$ and residual $\mathbf{r}_0$ is given by

$$\mathbf{d}_0 = \mathbf{r}_0 = \mathbf{b} - \mathrm{A}\mathbf{x}_0. \tag{2.12}$$

Then, using the algorithm as given by [9]

$$\alpha_i = \frac{\mathbf{r}_i^T \mathbf{r}_i}{\mathbf{d}_i^T \mathrm{A}\mathbf{d}_i} \tag{2.13}$$

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha_i \mathbf{d}_i \tag{2.14}$$

$$\mathbf{r}_{i+1} = \mathbf{r}_i - \alpha_i \mathrm{A}\mathbf{d}_i \tag{2.15}$$

$$\beta_{i+1} = \frac{\mathbf{r}_{i+1}^T \mathbf{r}_{i+1}}{\mathbf{r}_i^T \mathbf{r}_i} \tag{2.16}$$

$$\mathbf{d}_{i+1} = \mathbf{r}_{i+1} + \beta_{i+1} \mathbf{d}_i \tag{2.17}$$

the estimate for $\mathbf{x}$ is continously improved. Here, α_i is the length $\mathbf{x}_i$ moves in the search direction $\mathbf{d}_i$ to minimize $\mathbf{x}_{i+1}$, while β_i is a Gram-Schmidt constant which ensures that

the next search direction is conjugate to all previous directions. By using Eq. (2.3) in place of Eq. (2.10), the voltage of the nodes in the random conductor network described in section 2.1 is calculated.

The conjugate gradient method converges when $\mathbf{r}^2 = 0$. In theory, this will happen in a maximum of N steps for an $N \times N$ matrix. In practice however, round-off errors are present and can cause the method to run for longer than theoretically necessary. The method still converges in $\mathcal{O}(N)$ iterations as long as a limit ϵ is set for $\mathbf{r}^2$. The conjugate gradient method is then stopped when $\mathbf{r}^2 \leq \epsilon$, and the precision of the algorithm can be adjusted by adjusting ϵ.

While the method described here only works for a symmetric, positive-definite matrix A, there are more general methods which do not require these properties, such as the biconjugate gradient method. In the case of non-linear systems, there exist methods such as the ones introduced by Fletcher and Reeves or Polak and Ribiere. Descriptions of these can be found in [9] and [10]. Furthermore, the speed of the conjugate gradient method can be improved by preconditioning. The basic idea of the preconditioned conjugate gradient method, is to find a matrix M which approximates A well, but is easier to invert. Then, one gets

$$\mathrm{M}^{-1}\mathrm{A}\mathbf{x} = \mathrm{M}^{-1}\mathbf{b}.$$

If the eigenvalues of $\mathrm{M}^{-1}\mathrm{A}$ lies closer than those of A, then the preconditioned system can be solved faster than the original one. One method of preconditioning, which is well suited for the problems studied here, is the case of Fourier acceleration, as described in [11] and [12].

2.5 The Hoshen-Kopelman algorithm

The Hoshen-Kopelman algorithm is a fairly simple algorithm to find clusters spanning a system. Starting out at one corner of the system, one runs through each site in the system, checking if they are connected to neighbours. In the beginning, all sites are considered to have the cluster number 0. As one runs through the system, e.g. by starting at (0,0) and running through row by row, sites are given new cluster numbers. If the site is not connected to any other sites, it is not given a new number. If one site is connected only to sites which have not yet been checked, the site is assigned a new cluster number. Should the site be connected to other sites, it is given the same cluster number as these sites. The algorithm can be described as follows:

- Set the cluster number k to 2[1]. Initialize an array C which stores the cluster numbers in their respective position. That is, set the second element of the array to 2, the rest to 0.
- Start at one end of the lattice, run through row by row.
- If a site is not connected to previously checked sites, but is connected to other sites, assign it the cluster number k. Increment k, set the k'th element of C to k, $C[k] = k$.
- If a site is connected to one site previously checked, assign the current site the cluster number of the previously checked site.
- If a site is connected to two sites previously checked, there are two possibilities: that the cluster numbers of the previously checked sites, l and m, are the same or different. If $l = m$, the cluster number of the current site is set to l. If $l \neq m$, the cluster number of the current site is set to the lowest cluster number. The element in C belonging to the largest cluster number, is set to the negative value of the lowest cluster number. Thus, if $l < m$, the current site is assigned the cluster number l, while the m'th element of C is set to $-l$, $C[m] = -l$. This indicates that the clusters l and m are the same cluster, and any sites having cluster numbers m are a subset of the cluster l.

When the entire system has been run through, one can check for a spanning cluster simply by checking if sites at each end of the system belong to the same cluster. This is done by checking the cluster number assigned to the sites. If a site has cluster number n, $C[n]$ is checked to find the root cluster the site belongs to. If $C[n] = -q$, $C[q]$ is checked. This goes on until a positive element is found, indicating the root cluster the site belongs to. If sites on both ends of the system belongs to the same root cluster, that cluster is the spanning cluster.

2.6 Optimal paths and pathscape

2.6.1 Optimal paths

Finding the optimal path through a system is a much studied problem in computer science and mathematics. Given a weighted graph and two points a and b on it, a path

[1]Starting with $k = 2$ is just a habit from site percolation, where one might work with sites with the number 0 (unoccupied) or 1 (occupied). Thus, one can assign cluster numbers to occupied sites by changing from 1 to k and saying that all sites with numbers > 0 are occupied. In the system described here, i.e. bond percolation, using $k = 1$ works just as well.

between the two points has a weight equal to the sum of all weights it passes. The optimal path between the two points is then the path which has the minimum possible weight. There are a multitude of different algorithms to find the optimal path, but the focus here is on the one developed by Hansen and Kertész[13] and its use in the pathscape method [14].

In the method by Hansen and Kertész, the premise is that the optimal path between two points a and b, ab, can be decomposed to the sum of two optimal paths, ac and cb, where c is an intermediate point on the path ab. Thus, by finding the optimal path between a and c and the one between c and b, the optimal path $ab = ac + cb$ can be found. This is also the basic premise of the well-known Bellman-Ford algorithm which is also used in the Floyd-Warshall algorithm [15]. In the Hansen-Kertész algorithm, all nodes in a network is assigned a large value, e.g. ∞. The nodes are connected by links with a positive weight. One node, a, is locked to zero. One then iterates over all nodes, setting the value of each node i to

$$v_i = \min(v_j + t_{ij}) \tag{2.18}$$

where j runs over all the nodes adjacent to i, and t_{ij} being the weight of the link between i and j. This is done for all nodes in the system. The iterations through the system continues until no nodes have their values updated any longer. Each v_i will then be equal to the cost of the optimal path between node i and a. This is an example of a single-source shortest path algorithm.

2.6.2 Pathscape

The above method can also be used to make a landscape of optimal paths, or a "pathscape", as done in [14]. By finding the optimal path between node a and all other nodes, and doing the same for node b, one can combine these to make a pathscape for the optimal path between a and b. Let d_i^a be the cost of the optimal path between nodes a and i, and similar for d_i^b. $d_i = d_i^a + d_i^b$ is then the minimal cost of a path between a and b which goes through node i. Furthermore, $d_a = d_b = d_{min}$, where d_{min} is the minimum value in the system. Each node i with the value $d_i = d_{min}$ is part of the optimal path between a and b, while nodes j which are parts of less optimal paths have values $d_j > d_{min}$. This would then be a simple pathscape, where the optimal path could be seen as the deepest "valley", with "mountains" going upwards on either side of it. The valley then has a low cost associated with traveling along it, while the mountains have a higher cost.

Now, assuming one wants to find the optimal path between two edges of a system, A and B, both containing several nodes, the above procedure is run for each node $a \in \{A\}$ and similar for $b \in \{B\}$. Each possible combination of a and b is used to find a simple pathscape. The pathscape is then found by setting $d_i = \min d_i^{sp}$, where sp runs over all the simple pathscapes. When this is done, the final pathscape will have one optimal path where each node has the value d_{min}. All other nodes have larger values. The optimal path can also be seen as a global minimum. There can still be several local minima, which have values higher than d_{min}. If two nodes on both edges have values $d_a = d_b > d_{min}$, they have a continous path of nodes between them with the same value $d_i = d_a$. These nodes then form a local minima. In other words, any node in an optimal spanning path has a constant threshold cost associated with it, T, which is the same for each node in the path.

In the case of the spanning path wrapping around the periodic boundary, the same procedure as above apply, except that the only possible combination of nodes a and b is if $b = a$, such that the spanning path traverses the pathscape from node a in one direction, and connects back on itself, as in Figure 2.7.

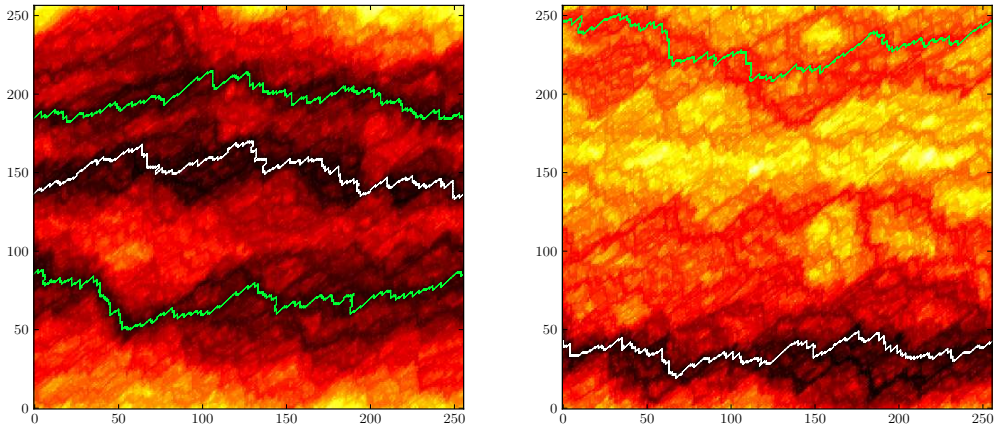
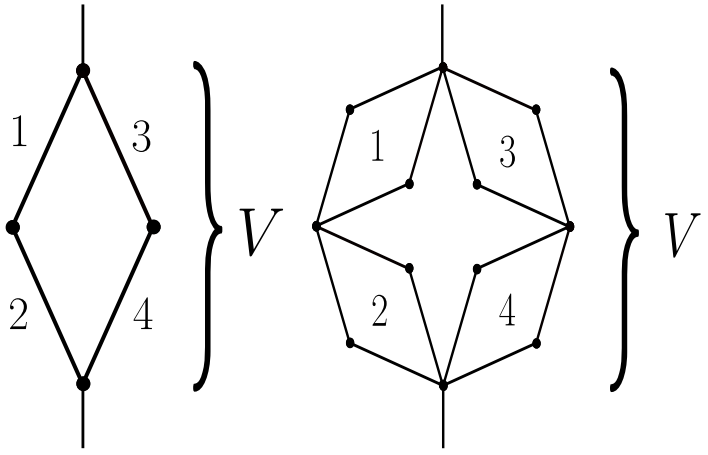


FIGURE 2.7: The pathscapes for two different realizations of a network as described in section 2.1 with $L = 256$ and periodic boundary conditions in the lateral direction. The current flows in the vertical direction, and the dual current flows perpendicular to it. Darker areas are nodes with small costs, while brighter areas are costlier to move through. Optimal spanning paths are colored. The white paths are the global minima, and the global optimal paths. The green paths are local minima.

2.7 Hierarchial model

As the square network is difficult to look at from an analytical perspective, it can be instructive to look at a simplified hierarchial model. This is initially constructed as

four bonds between two nodes which have a constant voltage drop V. Each bond is then replaced by a similar four-bond network, giving a sixteen-bond network. This can then be continued, replacing each bond in an n-bond network with a four-bond network, creating a network of size $4n$.



(a) Four-bond network (b) Sixteen-bond network

FIGURE 2.8: The two first steps in the hierarchial model. (a) shows the basic subnetwork consisting of 4 bonds, while (b) shows the network when each bond in (a) has been replaced by a four-bond network as in (a). The network is kept under a fixed voltage drop V.

2.7.1 General behaviour

Assuming a basic four-bond network between two electrodes as in Figure 2.8(a). The voltage drop over the network is V, with the voltage drop over each bond being v. Each bond also has a threshold t. Rewriting the conductance for $v \geq t$ as given in Eq. (2.5) with $r = \alpha/\beta$ gives

$$g = \alpha \frac{1 + (r-1)\frac{t}{v}}{r}. \tag{2.19}$$

Thus, the conductance can be written as $g = \alpha f$ with

$$f = \begin{cases} 1 & \text{if } v \leq t \\ \frac{1+(r-1)\frac{t}{v}}{r} & \text{if } v \geq t \end{cases} \tag{2.20}$$

Consider now two bonds connected in series, with voltage drops v_k and v_l, e.g. v_1 and v_2 in Figure 2.8(a). Bond k is broken, while bond l is not. This means that $t_k < t_l$, as $v_k = v_l$ before any of the bonds break since they both have conductance α. The current flowing through the bonds is conserved, i.e. $i_k = g_k v_k = g_l v_l = i_l$. This gives

$f_k v_k = f_l v_l$. Combining this with $V = v_k + v_l$ gives

$$\begin{array}{rcccl} v_k & = & \frac{V}{1+1/r} - t_k \frac{1-1/r}{1+1/r} & = & \frac{rV}{1+r} + t_k \frac{1-r}{1+r} \\ v_l & = & \frac{V}{1+r} - t_k \frac{1-r}{1+r} & & \end{array} \tag{2.21}$$

In the case of a perfect plastic behaviour, i.e. $\beta = 0$, $r \to \infty$. It is clear then, from Eq. (2.21), that $v_k = V - t_k$ and $v_l = t_k$. As v_l is constant, and $t_l > t_k$, bond l will never break, and no loops can form. This should then also be the case if $\alpha \to \infty$, for finite β. As using infinite α with a voltage threshold creates some problems, as discussed later, the case of $\alpha \to \infty$ should be seen as an approximation of very large values of r. Opposite to the case of $r \to \infty$, is the case of $r \to 0$. This is the case if bonds turn superconducting after reaching the threshold, i.e. $\beta \to \infty$. From Eq. (2.21), it is seen that $v_k = t_k$ and $v_l = V - t_k$, i.e. the opposite of the perfect plastic case. This will increase the chance of bond l breaking, thus having a path of broken bonds form between the two electrodes. It is worth noting that bond k will still have a voltage drop over it. This is due to the continous nature of the transition. The voltage drop over bond k can be written as a function of f_k and f_l by inserting for v_l into $V = v_k + v_l$, where v_l is found from current conservation:

$$v_k = \frac{V}{1 + f_k/f_l} = \frac{f_l V}{f_k + f_l}. \tag{2.22}$$

Setting $f_l = 1$ and $v_k = t_k$, gives $f_k = V/t_k - 1$. Thus, as the voltage drop over the network increase, so does the conductance of the bond. This gives the bond a current $i_k = \alpha(V - t_k)$, which increase with V. If $\alpha = 0$ then, as bond l is not broken, $i_l = 0 = i_k$.

The conductance of the entire four-bond network is given by

$$\begin{array}{rcl} G & = & (\frac{1}{g_1} + \frac{1}{g_2})^{-1} + (\frac{1}{g_3} + \frac{1}{g_4})^{-1} \\ & = & (\frac{1}{\alpha f_1} + \frac{1}{\alpha f_2})^{-1} + (\frac{1}{\alpha f_3} + \frac{1}{\alpha f_4})^{-1} \\ & = & \alpha(\frac{f_1 f_2}{f_1 + f_2} + \frac{f_3 f_4}{f_3 + f_4}) \\ & = & \alpha F \end{array} \tag{2.23}$$

As can be seen from Eq. (2.23), the conductance of the network is dependent on how many bonds are broken. From now on, only the case of $r \to \infty$ is considered. In the basic four-bond network, maximum two bonds can break. As both bonds in a series, e.g. the two bonds on the left half of the network, will have the same voltage drop until one is broken, it is naturally the bond with the lowest threshold which break first. Thus, when two bonds have broken and there is a path through the network (from left to right), it is the two bonds with the smallest threshold on both sides which have broken.

These bonds will henceforth be labeled i and j, with $t_i < t_j$. In the case of $r \to \infty$, $f_i = t_i/v_i = t_i/(V - t_i)$. This gives

$$\begin{aligned} F &= \frac{f_i}{1+f_i} + \frac{f_j}{1+f_j} \\ &= \frac{t_i}{v_i+t_i} + \frac{t_j}{v_j+t_j} \\ &= \frac{t_i}{V} + \frac{t_j}{V} \\ &= \frac{T}{V} \end{aligned} \tag{2.24}$$

where $T = t_i + t_j$.

Now, considering a larger network, where each bond in the four-bond network is replaced by a similar four-bond network, as in Figure 2.8(b), this network will have the conductance $\widetilde{G} = \alpha\widetilde{F}$, with

$$\widetilde{F} = \frac{F_1 F_2}{F_1 + F_2} + \frac{F_3 F_4}{F_3 + F_4}. \tag{2.25}$$

Looking at half the network, i.e. two sub-networks k and l in series, $\widetilde{V} = V_k + V_l$ and current conservation gives

$$V_k = \widetilde{V} - \frac{V_k F_k}{F_l}. \tag{2.26}$$

If k is broken, $F_k = T_k/V_k$ from Eq. (2.24) and $V_k = \widetilde{V} - T_k/F_l$. This gives

$$\begin{aligned} \frac{F_k F_l}{F_k+F_l} &= \frac{T_k F_k/V_k}{T_k/V_k+F_l} \\ &= \frac{T_k F_l}{T_k+F_l(\widetilde{V}-T_k/F_l)} \\ &= \frac{T_k}{\widetilde{V}} \end{aligned} \tag{2.27}$$

Thus, if two structures i and j in the network are broken, one in each half, Eq. (2.25) and Eq. (2.27) combine to give

$$\begin{aligned} \widetilde{F} &= \frac{T_i}{\widetilde{V}} + \frac{T_j}{\widetilde{V}} \\ &= \frac{\widetilde{T}}{\widetilde{V}} \end{aligned} \tag{2.28}$$

which is on the same form as Eq. (2.24).

If the network is increased in size, again by replacing each bond with a four-bond network, Eqs. (2.25)-(2.28) provides the same result as for the subnetworks. Thus, any broken network in the case of $r \to \infty$, will have $F = T/V$, where T is the sum of the thresholds of the broken subnetworks and V is the fixed voltage drop over the network.

2.7.2 Optimal path

In the basic four-bond network, the path which breaks the network is simply the one which minimize $t_i + t_j$, i.e. the optimal path. For a larger network, it is not necessarily that simple. Again, only the case of $r \to \infty$ is considered here.
The basic network to look at above the four-bond, is the sixteen-bond network. Looking at half the network, i.e. two four-bond networks in series, the subnetworks are labeled k and l. The bond i_k is the bond with the smallest threshold, t_i^k, in k. Bond j_k is the second bond that breaks, thus breaking k. Similar for l. Furthermore, bond i_k is considered to have the smallest threshold of all bonds, thus breaking first. With $F_l = 1$, this gives

$$\begin{aligned} V_k F_k &= V_l F_l \\ V_k(\tfrac{1}{2} + \tfrac{f_i^k}{1+f_i^k}) &= V_l \\ V_k(\tfrac{1}{2} + \tfrac{t_i^k}{V_k}) &= V_l \\ \tfrac{V_k}{2} + t_i^k &= V_l \end{aligned} \tag{2.29}$$

which leads to

$$\begin{aligned} V &= V_k(\tfrac{3}{2} + \tfrac{t_i^k}{V_k}) \\ &= \tfrac{3}{2}V_k + t_i^k. \end{aligned} \tag{2.30}$$

Inserting and solving for V_l instead gives

$$V = 3V_l - 2t_i^k. \tag{2.31}$$

At a given $V = V_b$, a second bond will break. The first subnetwork to break is the one that minimize V_b. When a bond in the four-bond network breaks, the voltage drop over the network is twice the threshold of the bond, i.e. $V_k = 2t_j^k$ when j_k breaks, and similar for $V_l = 2t_i^l$. Inserting this, the first bond of j_k and i_l to break is the one which minimize V_b, i.e. $\min(3t_j^k + t_i^k\ ,\ 6t_i^l - 2t_i^k)$, which is equivalent to $\min(t_i^k + t_j^k\ ,\ 2t_i^l)$. Thus, if j_k breaks first, the threshold of k is less than that of l, i.e. $T_k < T_l$, as $T_l = t_i^l + t_j^l \geq 2t_i^l$. Should i_l break first, however, we have

$$\begin{aligned} V_k F_k &= V_l F_l \\ \tfrac{V_k}{2} + t_i^k &= \tfrac{V_l}{2} + t_i^l \end{aligned} \tag{2.32}$$

leading to

$$\begin{aligned} V &= 2V_k + 2(t_i^k - t_i^l) \\ &= 2V_l + 2(t_i^l - t_i^k). \end{aligned} \tag{2.33}$$

The third bond to break would be $\min(2t_j^k + t_i^k - t_i^l\ ,\ 2t_j^l + t_i^l - t_i^k)$, which is equivalent to $\min(t_i^k + t_j^k\ ,\ t_i^l + t_j^l)$. Thus, the four-bond network which breaks first, will be the one which minimize the cost of the path $\mathcal{P}$ through, i.e. $\min \sum_{n\in\mathcal{P}} t_n$, where n runs over all

the nodes in the path.

Doing this in a more general fashion, we have

$$\begin{aligned} V &= V_k(1+\tfrac{F_k}{F_l}) \\ &= \tfrac{V_k}{F_l}(F_k+F_l) \\ &= V_l(1+\tfrac{F_l}{F_k}) \\ &= \tfrac{V_l}{F_k}(F_k+F_l). \end{aligned} \tag{2.34}$$

Minimizing V to find which subnetwork breaks first, is equivalent to finding $\min(\frac{V_k}{F_l}\ ,\ \frac{V_l}{F_k})$, where k breaks first if the left side is the minimum, and l otherwise. Multiplying by F_kF_l, shows that this is equivalent to $\min(V_kF_k\ ,\ V_lF_l)$, i.e. minimizing the current. Now, assuming k breaks first, then $F_k = T_k/V_k$. This is the minimum possible value of F_k, which increase with increasing V. For the subnetwork l, $F_l \in (T_l/V_l, 1]$. The lower limit is exclusive, as l will never break if k breaks first, in which case F_l will never reach its minimum value of T_l/V_l. It is clear from this, that if k breaks first, then $\min(V_kF_k\ ,\ V_lF_l) = \min(T_k\ ,\ V_lF_l)$, and $T_k < T_l$ if the minimum possible value of F_l is checked against. As $V_kF_k = V_lF_l$ and F_k decrease as bonds break in k, $V_k > V_l$ if bonds only break in k, but not in l. To ensure that there is no bias in the breaking, i.e. that bonds with larger thresholds in k break before smaller thresholds in l, with the path through k actually being less optimal than the one through l, the largest value of F_l is checked against. If k breaks first and $F_l = 1$, i.e. no bonds are broken in l, then $T_k < V_l$. As no bonds in l are broken, they will have have the same voltage drop over them. Any subnetwork in l will have a threshold of less than $V_l/2$. Going down to the basic four-bond network, the voltage drop over this is smaller than $2t_i$. Thus, as no bonds are broken, $V_l < 2T_i^l < T_l$, and the path through k is the optimal.

An important point to note, is that typically one use a quenched current threshold rather than a voltage threshold when looking at a random conductor network. As one often sets $\alpha = 1$, this doesn't matter much, as the voltage drop over a bond equals the current before the bond breaks. For $\alpha \neq 1$, $t_i = \alpha t_v$, where the subscripts denote the current and voltage thresholds. As the current threshold is just a multiple of the voltage threshold, the distribution of thresholds is the same. Thus, whether one break bonds at a certain current or voltage drop in the simulation, doesn't matter. For α infinite however, this is not true. If one consider current being sent through the network, a perfectly conducting network would not have any voltage drop. Thus, for a quenched voltage threshold, no bonds would break. In addition, the current thresholds would be infinite, and never break. One would then need a quenched, finite, current threshold. The program used in this thesis, use the conjugate gradient method to find the voltage of the nodes. For

large conductances, it will take longer to converge, and if the conductance is too large, it risks never converging. As such, there is a limit to how large values of α can be studied. The limit of $r \to \infty$ can be studied by setting $\beta = 0$, but not by setting α infinite. The above analysis doesn't discriminate between these two cases of infinite r, and a valid question is whether the spanning path does follow the optimal path for infinite α and $\beta > 0$.

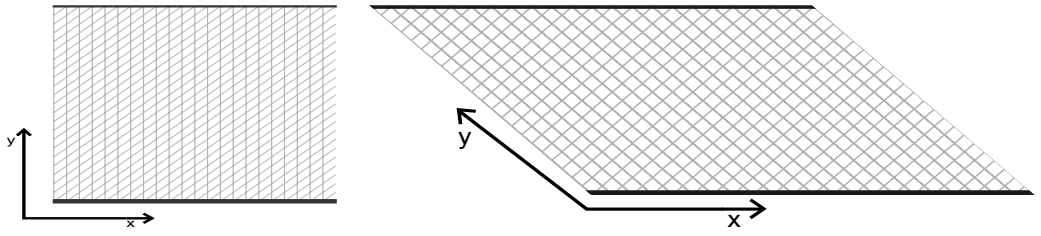
The following argument tries to explain what should happen for infinite α with a current threshold. Looking at the case of $r = 0$ again, and assuming $\alpha = 0$, the voltage drop over the unbroken bond l in series with k, is $v_l = V - t_k$. Assuming again the basic four-bond network, the first side of the two parallel series to break is the one which minimize $V = v_l + t_k$ for $v_l = t_l$. Thus, the spanning path through the four-bond network follows the optimal path. Note that this spanning path goes between the electrodes, and not parallel to them, as was the case for $r \to \infty$.
For any broken bonds in a network, the voltage drop over them is equal to their threshold until there is a spanning path between the electrodes. Before there is a spanning path between the electrodes, no current goes through the network due to current conservation. The voltage drop over unbroken bonds in series with broken bonds increase with increasing V. Any series of broken bonds can be seen as one large bond with a threshold $T = \sum_n t_n$, and the voltage drop over unbroken subnetworks will increase when other subnetworks in series break. Similar to the arguments made above for the spanning path following the optimal path for $r \to \infty$, this should also be the case for $r = 0$. If then the spanning path follows the optimal path for $r = \alpha = 0$ with a voltage threshold and a fixed voltage drop, this must also be the case for $\alpha \to \infty$ with a current threshold and a fixed current throughput. This is due to the properties of the dual network, as the dual network switch the behaviour of current and voltage compared to the original network. Thus, as the spanning path follows the optimal path for $\alpha = 0$ and a voltage threshold in the original network, it does so for $\alpha \to \infty$ and a current threshold in the dual network.

Chapter 3

Implementation

3.1 The network

The network implemented is that of a square lattice rotated 45° with respect to two electrodes such as the one described in section 2.1. To simplify the implementation and subsequent programming, the voltage nodes are considered to form a square lattice. Each voltage node has two bonds protruding upwards; one to the neighbouring node straight above, and one to the top right neighbour. Thus, each node is connected to two nodes above, and two below, with the exception of nodes considered to lie on the electrodes. This can be seen in Figure 3.1(a). The boundary in the x-direction is periodic.



(a) (b)

FIGURE 3.1: (a) shows the network as it is modelled in the code. (b) is what one gets by transforming the y-axis. This is equivalent to the network in Figure 2.1 due to the periodic boundary.

The thresholds for the bonds are drawn from a uniform probability distribution with $t \in [0, 1)$. The random numbers are produced using a Mersenne Twister generator from the GNU Scientific Library.

In the code used, the bonds are considered reversible until there is no more current going through them, i.e. $i = 0$ for $\Delta v \geq t$, in which case they break irreversibly. This

is illustrated in Figure 3.2. In the fuse case, the bonds break irreversibly when they hit the threshold.

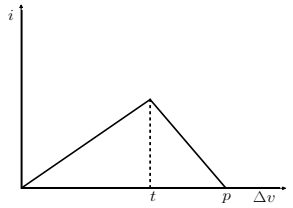


FIGURE 3.2: The current going through a bond for increasing Δv. The bond is considered reversible until it reaches the point p, where it breaks permanently.

The conjugate gradient method implemented use only the voltage drop over the bonds and their conductance. The conductance of bonds is updated after the conjugate gradient is run, which calculates the voltage of all the nodes. Having bonds turn irreversible after reaching the thresholds could then turn problematic due to round-off errors, e.g. if the voltage drop was subsequently reduced to below the threshold of the bond. This is particularly the case for $\beta < 0$. For $1 \leq r < \infty$, with $r = \alpha/\beta$, this is no problem, as the voltage drop over a bond will never decrease.

For $0 < r < 1$ on the other hand, the reversibility causes problems. As the conductance of the bonds increase after the threshold is reached, the voltage drop over them decrease. Thus, for small enough values of r, the bonds will start fluctuating between being above and below the threshold, causing the program to run in an infinite loop. For $r \to 0$, the voltage drop over the bond should rather stay equal to the threshold, as per Eq. (2.21) in section 2.7, but the conjugate gradient method as implemented here, does not allow for this. This particular behaviour does not allow one to study the system for $0 \leq r < 1$ with the current code. However, using $r > 1$, the equivalent can be studied in the dual network, as shown in section 2.2. For any value of $r > 0$, the dual network will have the ratio $r' = 1/r$.

It should also be noted that for large conductances, the conjugate gradient method will take longer to converge. Thus, for very large values of α, the systems can run for quite some time before breaking.

3.2 Disconnecting the network

The voltage drop over the entire system is intially set to $\Delta V = 10^{-3}$. The conjugate gradient method is run to find the voltage on all the nodes in the system. Then, each bond is checked to see if it passes the threshold. If any bond passes the threshold, it is subsequently marked, and the Hoshen-Kopelman algorithm is run to check if there

is a spanning cluster in the network. Conjugate gradient is then run again, and bonds are checked to see if they cross the threshold. This is done until no new bonds cross the threshold or there is a spanning cluster. If there is no spanning cluster and no bonds crossing the threshold, the voltage drop over the system is increased by ΔV. This repeats until there is a spanning cluster.

The Hoshen-Kopelman algorithm is used to extract the spanning cluster from the system. The backbone is subsequently extracted by removing dangling arms from the spanning cluster. When finding the spanning path, one looks at a path through the dual network, i.e. a path which crosses the conducting bonds. The empty space between four bonds (or two bonds and an electrode) is a site in the dual network, and the sites can be considered to connect when the bonds reach their thresholds. In the beginning, when no bonds have reached the threshold, the system consists only of disconnected sites. When enough bonds reach their thresholds and connect sites, a spanning cluster is formed. This separates, or disconnects, the original network, as seen in Figure 2.1. Again, as with broken bonds, the fuse nomenclature is used, and a disconnected network can still have an increasing current flowing through for increasing voltage drops for $\beta > 0$.

While one could use the approach described above to decide when there is a spanning path, it becomes problematic for finite values r which are also above 0, as loops can form. This, coupled with the periodic boundary, makes it difficult to decide when there is a spanning path which disconnects the network. For instance, one could have a loop which spans the network and wraps around the boundary, but still does not disconnect the network, such as the simple example in Figure 3.3. Therefore, two implementations of the Hoshen-Kopelman algorithm are used in this implementation: one which extracts the spanning cluster as described above, and one which checks when a spanning cluster exists and the network is disconnected. The latter function view the voltage nodes of the original network as sites, instead of the dual network. The voltage nodes are connected together by bonds, which are considered to disconnect sites when they reach the threshold. Thus, in the beginning, there is a spanning cluster of voltage nodes connecting the electrodes. When there is no spanning cluster of voltage nodes connecting the two electrodes together, the network is disconnected and considered broken.

It should be noted that such an implementation requires Hoshen-Kopelman to be run every time a new bond crosses the threshold. This can be quite ineffective, and if speed is of the essence, an implementation such as the one described by Newman and Ziff in [16] would be a better choice to check if the network is disconnected.

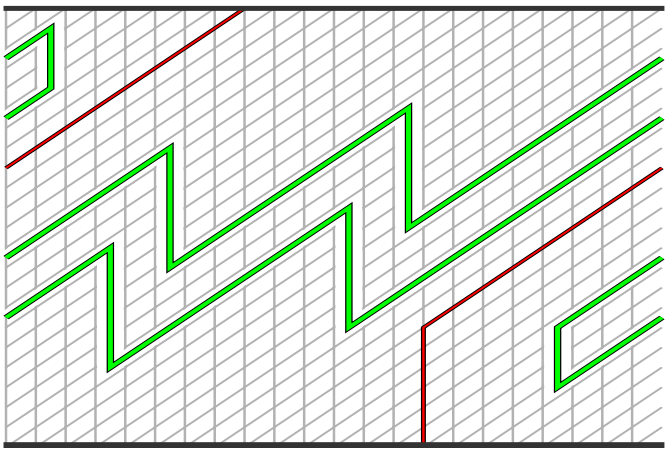


FIGURE 3.3: A closed loop which crosses all columns in the network and wraps around the periodic boundary. This is a possibility for $0 < r < \infty$. If all bonds in the closed loop (green path) were to get zero conductance, there would still be current flowing between the electrodes, e.g. along the bonds in the red line. Thus, the network would not be disconnected.

3.3 Pathscape

The pathscape method, as described in section 2.6 is used to find the optimal path. This is done by considering the voltage nodes in the dual network as sites, which are in a bond percolation system. The network is initially disconnected, and sites are connected together when bonds reach their threshold. The threshold of the bond is the weight of the link, and each node is then given the value $d_i = \min\left(d_{j_1}+t_{ij_1}, d_{j_2}+t_{ij_2}, d_{j_3}+t_{ij_3}, d_{j_4}+t_{ij_4}\right)$, where j_n denotes the sites adjacent to i.

As the network is periodic in the direction of the spanning path, one essentially have to find the optimal path going from one node and crossing the network before coming back on itself. This is done by appending a copy of the system on to the sides of the original, thus having a system which is twice as wide. In the implementation used here, this was done by appending the right half onto the left half, and vice versa, altough one might as well just append the full system on one side. This gives a system where one can find the optimal path between sites in a column n from the original system, and its twin column $\tilde{n}$ in the copied system. The entire system is copied to ensure that any overhangs in the optimal paths are included. In a network of size $L \times L$, i.e. L bonds between the electrodes, the system as described here will have $(L+1) \times 2L$ sites which paths can run through. The $L+1$ rows come from the fact that there are two "half-rows" next to the electrodes, i.e. all the sites there only have two neighbours.

A site k in a row m and column n is picked, and is kept locked at 0 to find the optimal path from that site to all other sites, as explained in section 2.6. The same is then done for the twin site $\tilde{k}$ in row m and column $\tilde{n}$. The two pathscapes are then added

together, i.e. $d_i = d_i^n + d_i^{\tilde{n}}$. Thus each site in the expanded system denotes the minimum cost for a path between k and $\tilde{k}$ crossing that site. The system is then reduced to its original size of $(L+1) \times L$, i.e. each site i is given the value $\min(d_i, d_{\tilde{i}})$. This gives the pathscape for the optimal path for one row. The same procedure is then run for all rows, before the optimal pathscape is constructed by setting each site i to $d_i = \min(d_i^m)$, where $m \in [1, L+1]$ runs over all the pathscapes found, i.e. one for each of the $L+1$ rows.

The pathscape method can be quite computationally expensive as one has to iterate through the system several times before it is relaxed to its final state. One thing to keep in mind is that the pathscape from one site to all other sites will be similar to the pathscapes for adjacent sites. In the pathscape from a site k to all other sites, the site k will be the minimum and could be seen as the bottom of a well, with the walls growing the further away from k one gets. The adjacent sites will then sit very close to the bottom of this well, while having similar pathscapes. Thus, when iterating through the rows to find the pathscapes, the pathscape from site k in row m is kept as the basis for finding the pathscape in row $m+1$. This saves a lot of time, and the main time spent finding the optimal pathscape, is spent in finding the pathscapes for sites k and $\tilde{k}$ in the first row.

As this algorithm involves iterating through several sites over and over again, it can benefit a lot from parallelization. A simple program was written using OpenCL to run it on a GPU. Without much time spent on optimizing the code for the specific architecture, the GPU code greatly outperforms the CPU code. Speedups of a factor of ≈ 30 were achieved for networks with $L = 64$, and a factor ≈ 60 for $L = 128$. This is largely attirbuted to the number of threads available on the GPU (71680 on an AMD Radeon HD 7950).[1]

[1]The CPU code used here could probably be improved a lot as well. While sizes of $L = 256$ could use up to a day to run with my implementation, Alex Hansen reports having a CPU version which runs "like a bat out of hell" on similar problems with system sizes of 1024.

Chapter 4

Results

4.1 Plastic-elastic regime

As per the model used in section 2.7, the parameter $r = \alpha/\beta$ should be the only thing affecting the path of the backbone for a system with quenched disorder. Thus, running through the system with different values of α and β, but keeping r constant, should give the same backbone and spanning cluster. This was checked by running through the same system three times for each value of r. r was set to 5, 50, 500 and 5000, while β was set to 0.1, 0.5 and 0.9, with $\alpha = r\beta$. 20 different systems of size $L = 64$ were run through for $r = 5$, 50 and 500, and 12 systems were run through for $r = 5000$. Each system was then compared with the two others, and the number of bonds in the backbone and spanning cluster which were differing were counted. For instance, if one bond was in the backbone in the system with $\beta = 0.1$, but not in the systems with $\beta = 0.5$ and 0.9, this would result in a count of 2. The results are shown in Figure 4.1.

As can be seen from this, the backbone and spanning clusters are identical for the same values of r, with the exception of a single system. The reason there are more bonds in the backbone differing than the spanning cluster for $r = 5$, is that the single bond differing in the spanning cluster, opens up a loop in the backbone. This causes two more bonds to not be registered as part of the backbone, while they are still part of the spanning cluster.

Figure 4.2 show the backbone of the spanning path created for two different systems with increasing r. As can be seen, as r grows larger, the number of loops and the mass of the backbone decrease, while the path follows more closely the minima.

To see if the backbone follows the optimal path for large r, systems were run through with $\beta = 0.1$ and $\alpha = r\beta$ for increasing values of r. This was done for $L = 16$, 32

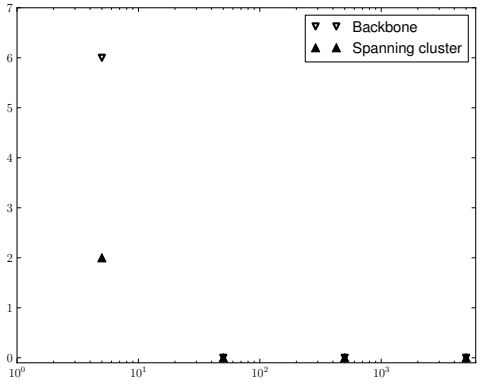


FIGURE 4.1: Number of bonds differing in the backbone and spanning cluster for same values of r of a given system, $L = 64$. This is checked over 20 runs for $r = 5$, 50 and 500, and 12 runs for $r = 5000$, with three different values of β for each system.

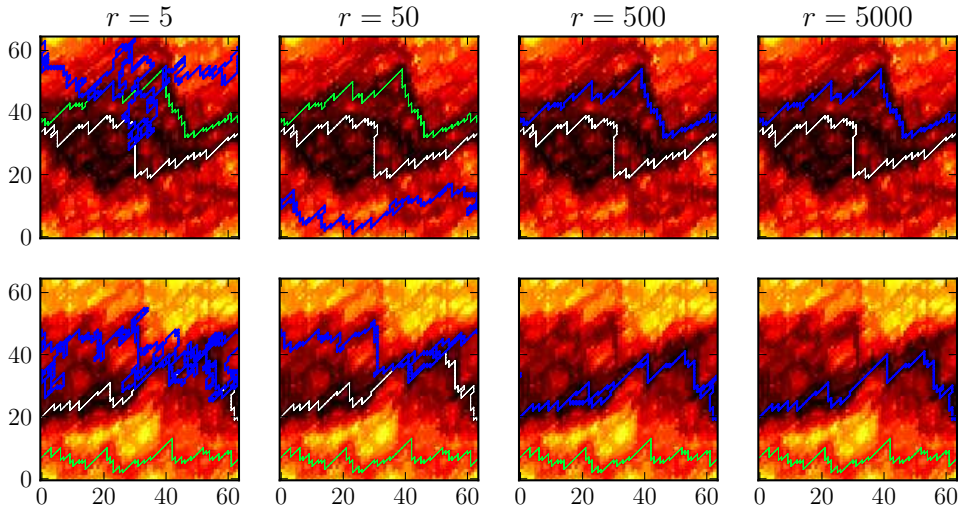
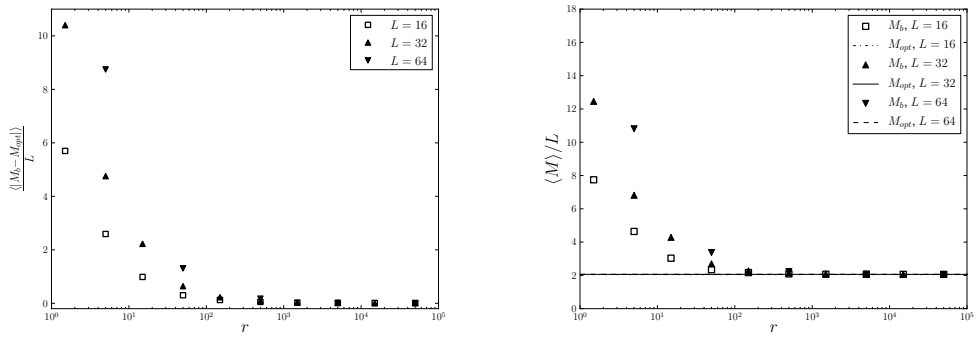


FIGURE 4.2: Realizations of two different systems of size $L = 64$ when a spanning path is created. Each row shows a system with the same quenched disorder but different r, with r increasing from left to right. White paths are global minima, green paths are local minima and the blue paths are the backbones of the spanning paths.

and 64, and the results were averaged over 200, 60 and 40 systems respectively. The same systems were used for each value of r for a given system size. For $L = 16$ and 32, $r \in [1.5, 5 \cdot 10^4]$, with 10 different values for r. Due to the computational cost, only 5 different values of r were used for $L = 64$, with $r \in [5, 5 \cdot 10^4]$. Figure 4.3(a) shows the average difference between the mass of the backbone, M_b and the mass of the optimal path, M_{opt} divided by L. As can be seen, this decrease with increasing r, and goes towards 0 for large values of r. Figure 4.3(b) shows the average backbone mass and optimal path mass. For large r, the average backbone mass goes toward the average mass of the (globally) optimal path.



(a) (b)

FIGURE 4.3: Average over 200, 60 and 40 systems for $L = 16$, 32 and 64. (a) The mass difference of the backbone and global optimal path with increasing r. (b) Points show the average backbone mass, while line shows the average mass of the global optimal path.

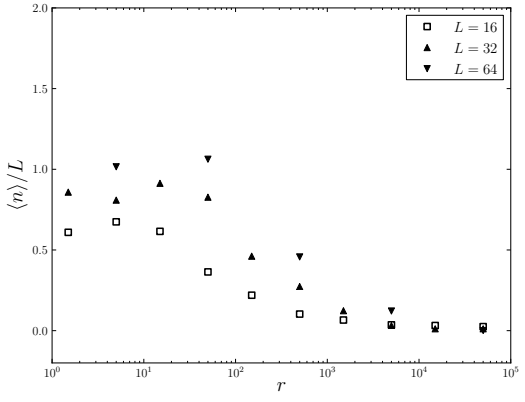


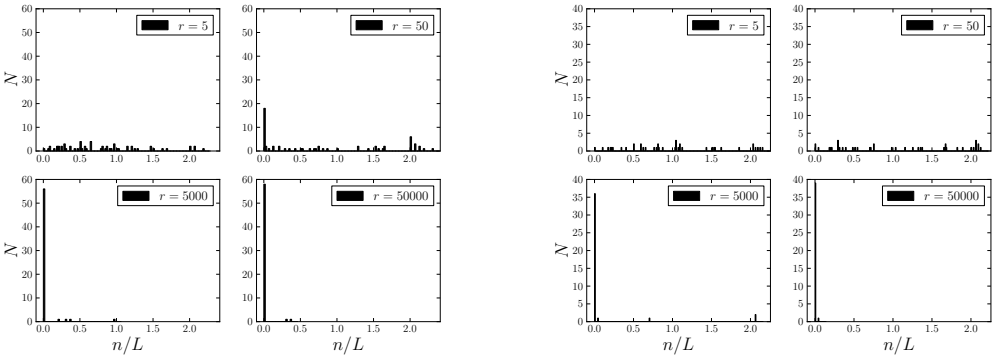
FIGURE 4.4: Overlap of backbone and the global optimal path. n denotes the number of bonds which are in the optimal path, but not in the backbone.

To ensure that the backbone actually follows the optimal path, and not only decrease in size, the overlap between the global optimal path and the backbone is checked. The number of bonds in the global optimal path which are not in the backbone is counted, i.e.

$$n = |O \setminus B|$$

where O and B denotes the sets of bonds in the optimal path and the backbone, respectively, and $\setminus$ is the set difference. As the backbone overlap more of the optimal path, n decrease. The results are seen in Figure 4.4.

As can be seen from this, n goes toward 0 as r increase, and thus the backbone overlaps more and more of the optimal path. As M_b also decrease and go towards M_{opt}, for increasing r, this indicates that the backbone follows the optimal path more closely for



(a) $L = 32$ (b) $L = 64$

FIGURE 4.5: Histograms over overlap between the backbone and optimal path. N is the number of systems with a given bin value.

increasing r and the bonds act more and more as a perfect plastic ($\beta = 0$).

Figure 4.5 shows that for increasing r, the prevalent cause of non-zero n, i.e. not complete overlap, is a few systems in which the backbone does not overlap with the optimal path to a large degree. This is in opposition to having several systems with few non-overlapping bonds.

From the above results, it is fairly clear that the only parameter relevant for the path of the backbone in random conductor networks as studied here, is $r = \alpha/\beta$. Furthermore, as r increase, the backbone follows the global optimal path to a larger degree. It is well-known that for $\beta = 0$, the backbone does follow the optimal path, and the above results indicate that this is the case in general for $r \to \infty$ with non-zero, positive β. This is also in accordance with the results derived in section 2.7.

Looking at the spanning cluster instead of the backbone, the data shows the same tendency. For increasing r, the mass of the spanning cluster approach that of the perfect plastic (PP) spanning cluster ($\alpha = 1$, $\beta = 0$). The spanning clusters also overlap more. This means that as $\alpha \to \infty$ for a constant β, the behaviour of the system goes towards that of the perfect plastic case. As argued in sections 2.1.2 and 2.2, the random conductor network can be used to simulate the flow of Bingham fluids in porous media by using a large value of α. This can also be accomplished by setting β small, as long as r is very large.

While a clear value of how large r has to be for the backbone to follow the optimal path has not been found, it seems to be of the order of $10^4 - 10^5$ for the system sizes studied here. As seen from Figure 4.4, the overlap seem to improve at a sharper rate for the larger system sizes for r in the order of $10^1 - 10^2$. For the larger values of r, the overlap is better for $L = 32$ and 64, than for $L = 16$. Whether this is a general tendency or just

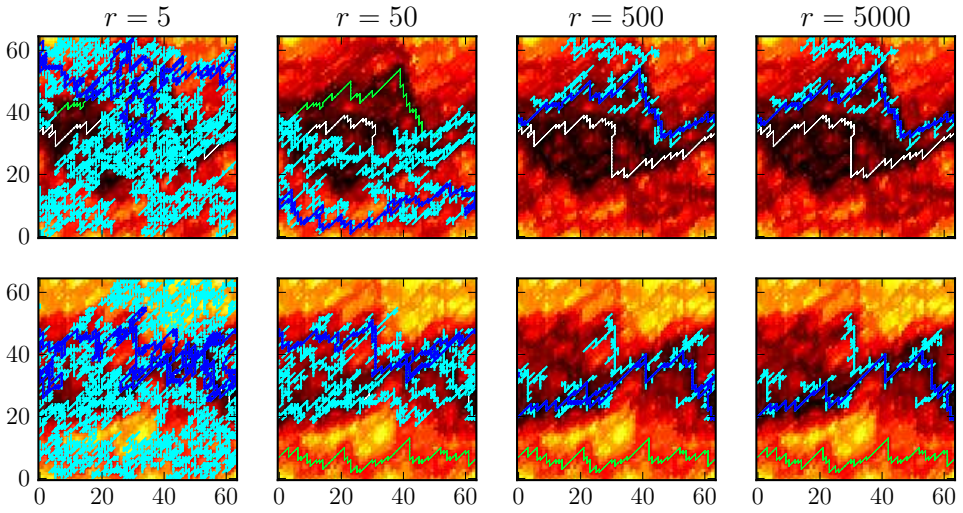
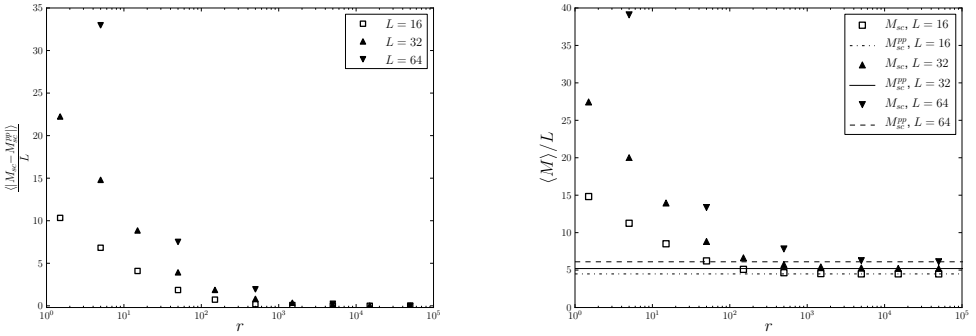


FIGURE 4.6: Same as Figure 4.2, but with the spanning cluster included in cyan.



(a) (b)

FIGURE 4.7: Average over 200, 60 and 40 systems for $L = 16$, 32 and 64. (a) The mass difference of the spanning cluster and the PP spanning cluster, with increasing r. (b) Points show the average mass of the spanning cluster, while lines shows the average mass of the PP spanning cluster.

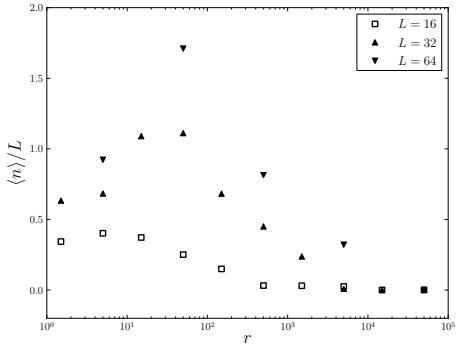


FIGURE 4.8: Overlap of the spanning clusters for the PP and general case. n denotes the number of bonds which are in the PP spanning cluster, but not in the general spanning cluster.

due to a low number of systems averaged over, is uncertain. Errorbars have been left out for readability here, and plots with errorbars are found in Appendix A. As can be seen from them, the variance in the overlap is larger for larger systems, except for the largest values of r, where it tends to be lower. However, the current statistics are not good enough to conclude whether this is true in general, or if it is just due to the low number of systems checked here. It would be of interest to improve the statistics and answer this.
Furthermore, it would be interesting to study the transition from a generally chaotic backbone path at low r to the perfect plastic at large r. It is not clear if there is a given value of r where the transition to a perfect plastic occur for all system sizes, or if it dependent on the system size. One would need to study larger systems to answer this, as well as run over several more values of r and average over more systems than have been done here.

4.2 Fuse model

Some work was done studying brittle fracture, i.e. the fuse model. The motivation for this was to look at how the fracture acted in the pathscape and see if it could be similar in the general flow case. This approach was abandoned after results suggested that it was the ratio α/β which was important for the flow path. The results found for the fracture problem are included here for completeness, and in case there might be something of interest here for others.

Fracture is a fairly complex problem, and depending on the disorder and size of the system, cracks can form and propagate via nucleation, avalanches or as a result of percolation[4]. In systems studied here, avalanches are numerous. This results in crack paths which can form seemingly at random in the system, as seen in Figure 4.9.

Initial results indicated a possibility of the fracture lying close to one of the local optimal paths. These results were on small system sizes however, and the effect decreased with increasing L. Still, the RMS distance between the backbone and the optimal paths was found for different system sizes, to see if it could scale non-linearly with the size.

The RMS distance is defined as

$$d = \sqrt{\langle y_i^2 \rangle - \langle y_i \rangle^2} \tag{4.1}$$

To find the RMS distance between the backbone and the optimal paths, the distance between two nodes is set to $y_i = y_b - y_o$, where y_b is the y-coordinate of a given site

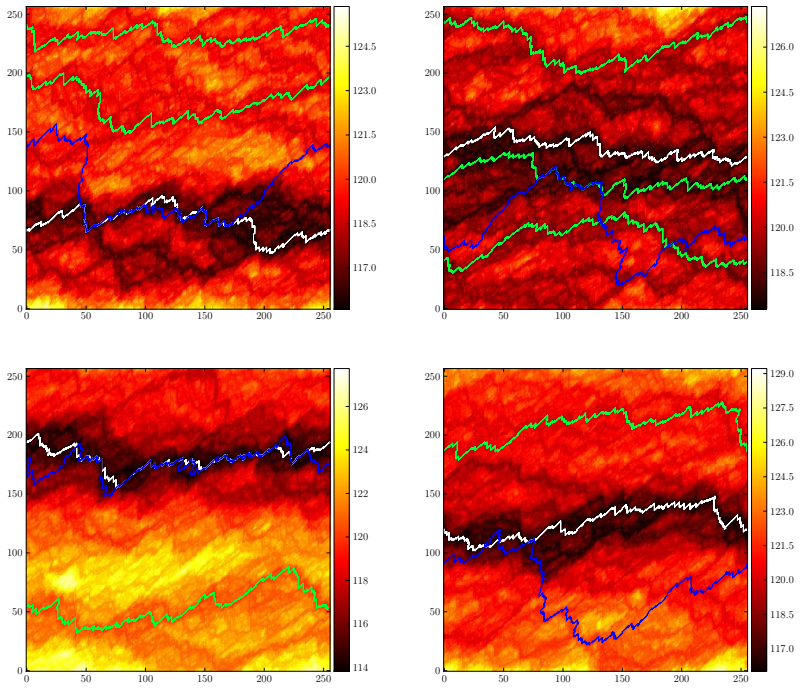


FIGURE 4.9: A few different results of fracture, $L = 256$. The spanning paths are colored green and the optimal path is white. The blue path is the backbone of the fracture. The colours have been normalized for each pathscape, and show the minimum total cost for a spanning path going through a given node.

on the backbone, while y_o is the coordinate of a given site in the optimal path. y_o is selected as the site lying closest to y_b, i.e. it fulfills

$$y_i = \min_{o \in j}(|y_b - y_o|).$$

where j is the set of bonds in the optimal path with that particular x-coordinate. This could lead to some sites in the optimal path not being included in the case of overhangs. As the optimal paths generally have few overhangs[13], this is assumed not to have a large impact on the result. The results are plotted in Figure 4.10, where systems of size 16, 24, 32, 48, 64, 128 and 256 have been studied. The results are averaged over 3000, 1000, 500, 250, 250, 50 and 24 runs, respectively. The RMS distance was found both between the backbone and only the global optimal path, and between the backbone and all optimal spanning paths (global and local). In the latter case, d was calculated for all optimal spanning paths, and the minimum value was chosen. As can be seen from Figure 4.10, the values are very close at small system sizes, where there are fewer spanning paths, and for the most part only one for the smallest L. As the system size increase, so does the number of spanning paths, and $\langle d \rangle$ decrease for the case of all optimal paths, compared to the case of only the global minimum.

These results indicate that $d \sim L^{\nu}$, with $\nu \approx 0.89$ for the global optimal path, and

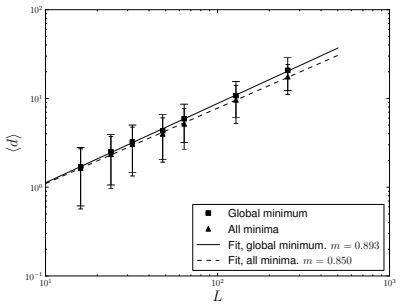


FIGURE 4.10: RMS distance between fracture and optimal minima. Error bars show one standard deviation. Longer bar on edge of the errorbars belong to global minima, shorter to all minima.

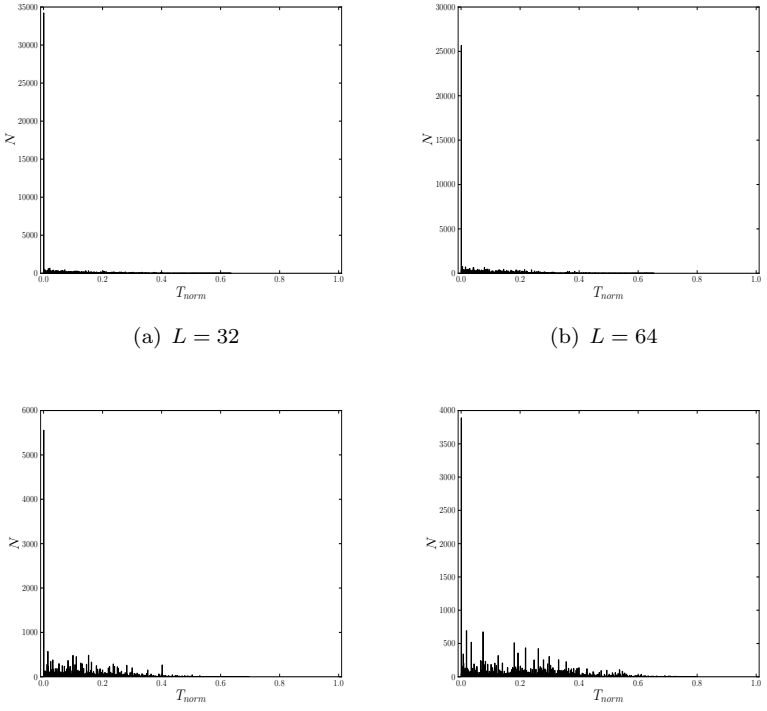
$\nu \approx 0.85$ for all optimal paths. However, the error bars are large, and ν seem to increase when larger system sizes are included. Thus, it is hard to conclude that $\nu < 1$. An additional source of potential error, is the fact that the same seed has been used for systems of different sizes. This could create additional correlations, as smaller regions of the larger systems have the same thresholds as a full smaller system.

To get a clearer view of where in the pathscape the fracture lies, histograms of the number of nodes with a certain cost are made. The histograms are created by normalizing each pathscape. The node values then go from 0 to 1, i.e. each node has a normalized cost

$$T_{norm} = \frac{T - T_{min}}{T_{max} - T_{min}}$$

where T is the real value of the node and T_{min} and T_{max} are the minimum and maximum costs in the entire pathscape. This is done for each system, and the results are added together in the histogram.

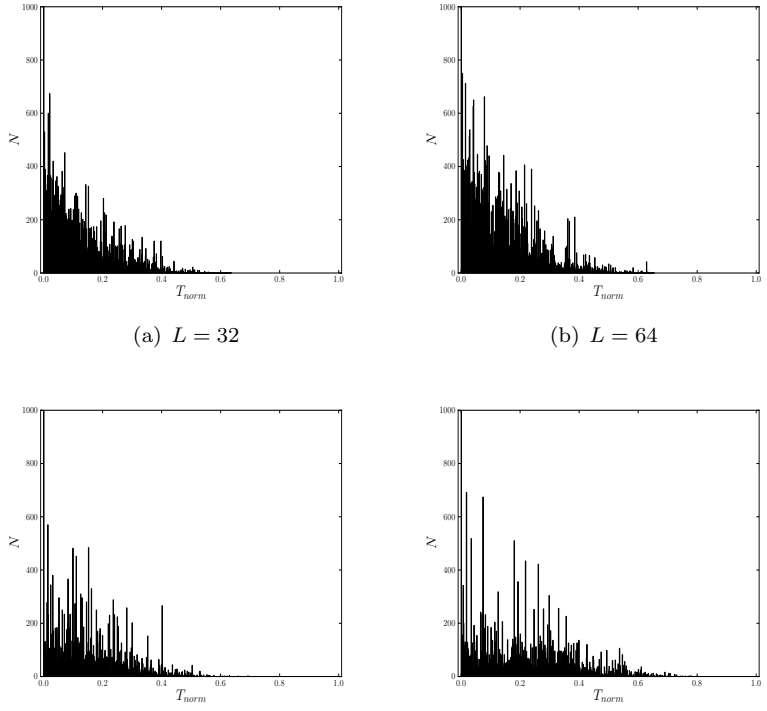
As can be seen from Figures 4.11-4.12, nodes in the backbone lie mostly in the less costly parts of the pathscape, and barely in the more expensive half. Several nodes tend to lie along the cheapest parts of the pathscape, such as nodes with a cost close to that of the global optimal path. Part of this can be attributed to the fact that there are always at least $2L$ nodes with $T = T_{min}$. This gives a large amount of nodes with $T_{norm} = 0$, as seen in Figure 4.13. The histogram of the fracture seem to be flatter for larger L, while the opposite is the case with the pathscape histogram. Assuming an average optimal path size of $2L$, a bit below what is seen in Figure 4.3(b), the ratio of nodes belonging to the optimal path is $2L/(L(L+1)) = 2/(L+1)$. Thus, the amount of nodes in the optimal path decrease compared to the total amount of nodes in a system. This gives a larger probability of having the fracture cross high-cost nodes, causing a flatter distribution of the fracture cost for larger system sizes.



(a) $L = 32$ (b) $L = 64$

(c) $L = 128$ (d) $L = 256$

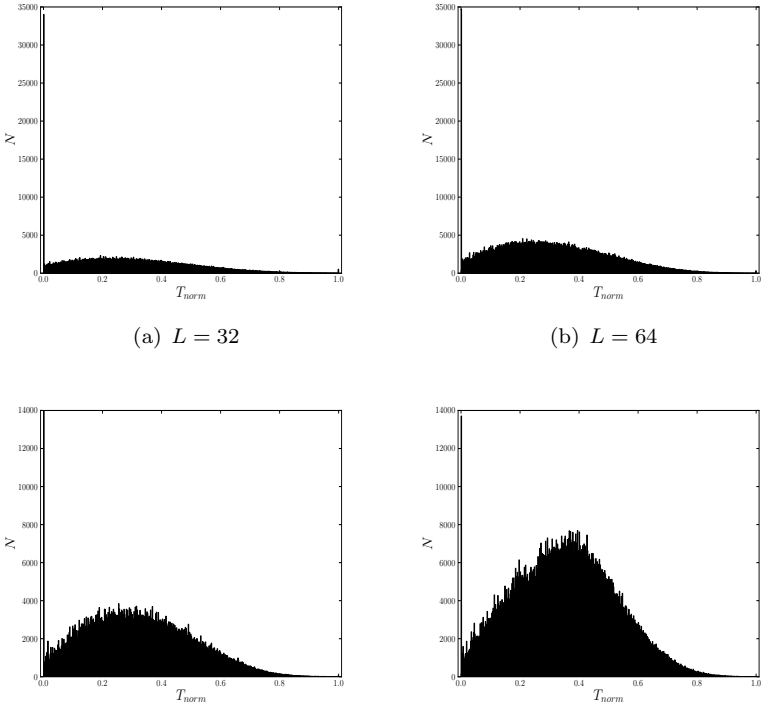
FIGURE 4.11: Histograms of the backbone, counted over 500, 250, 50 and 24 systems for $L = 32$-256. N is the number of nodes at a given value T_{norm}.



(a) $L = 32$ (b) $L = 64$

(c) $L = 128$ (d) $L = 256$

FIGURE 4.12: Figure 4.11 with the y-axis cut at $N = 1000$.



(a) $L = 32$ (b) $L = 64$

(c) $L = 128$ (d) $L = 256$

FIGURE 4.13: Histograms of the pathscape, counted over 500, 250, 50 and 24 systems for $L = 32$-256. N is the number of nodes at a given value T_{norm}.

An idea with the pathscape was to look at the possibility of using it to determine to some extent where the fracture path would occur, without having to resort to actually breaking the system. As seen in Figure 4.9, the fracture path sometimes lie close to the optimal path. The pathscapes can have a quite varying topology. Some might be very cheap along the global optimal path and fairly expensive everywhere else, like a valley with hills coming up on either sides, while others might be "flatter", in that they have several optimal spanning paths with costs very close to each other. Thus, the topology of the pathscape could possibly be used to find systems where it is probable that the fracture path follows close to the optimal path, e.g. by using wavelet-methods to find the general topology. Another possibility could be to exclude regions where it is highly unprobable that the fracture follows.

With the above results, it is hard to determine if it is possible to use the pathscape to finding probable/unprobable regions where the fracture occurs. While the fracture seem highly unlikely to lie on the costliest nodes, the number of nodes one can exclude that way is very few. In addition, it seem to be more difficult as the system size increase, with the fracture lying on even costlier nodes relative to the maximum cost. What has not been taken into account here, are the threshold costs of the local minima, and whether

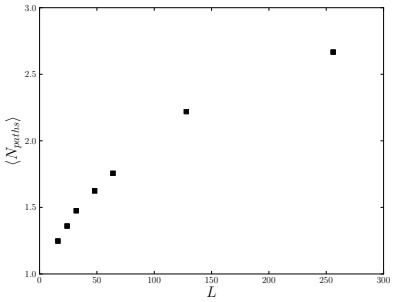


FIGURE 4.14: Average number of spanning paths for systems of size L

or not there is some correlation between them and the fracture nodes. As the system size increase, so does the number of local minima on average, as seen in Figure 4.14.

Chapter 5

Conclusion

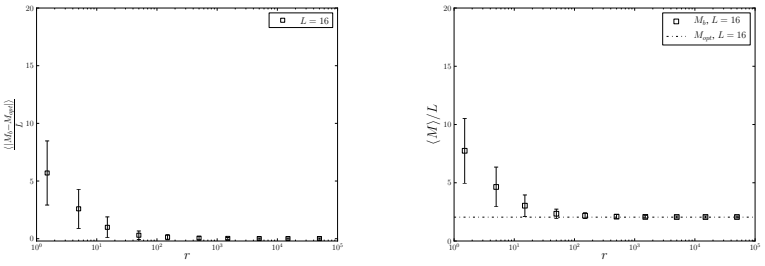
It has been found that in a random conductor network with a piecewise linear characteristic which has the slopes α and β, it is the ratio of the slopes, $r = \alpha/\beta$, which determine where the spanning path forms. An argument for this has been given analytically using an hierarchial model, and simulations conducted on a rotated square lattice support the conclusion of the argument. For the case of a perfect plastic, $\beta = 0$, it was already known that the path follows the optimal path. The results strongly indicate that this is the case in general for $r \to \infty$, even with $\beta > 0$, and that the conducting bonds behave more like a perfect plastic as r increase.
It would be of interest to further study this system at larger system sizes, and see if the value of r where the transition from a plastic-elastic to perfect plastic happens is dependent on the system size.

Some time was also spent on studying the fuse model, namely the fracture path in the optimal pathscape. While initial results indicated that the fracture might lie close to the optimal path, this seem not to be the case after all. Due to the complex process of the fracture, it is not possible to predict where it will occur. It could be of potential interest to further study the fracture path in the optimal pathscape, and see if it, to a certain degree, correlates with locally optimal spanning paths or the topology of the network. This could possibly be done e.g. by using wavelet methods to study the topology.

Appendix A

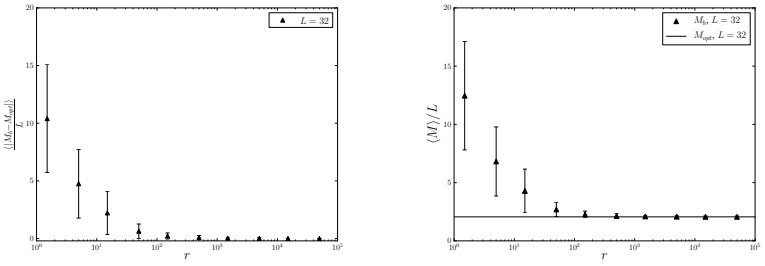
Error plots

This appendix contain the plots from section 4.1 with errorbars. The errorbars were not included in the plots in the results section to improve readability. All data points are averaged over 200, 60 and 40 systems for $L = 16$, 32 and 64, respectively. Errorbars denote one standard deviation.



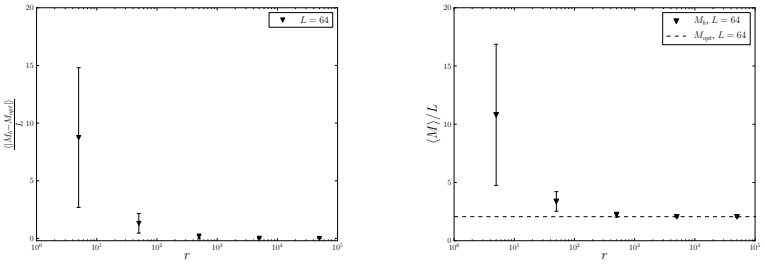
(a) (b)

FIGURE A.1: $L = 16$. (a) The mass difference of the backbone and global optimal path with increasing r. (b) Points show the average backbone mass, while line shows the average mass of the global optimal path.



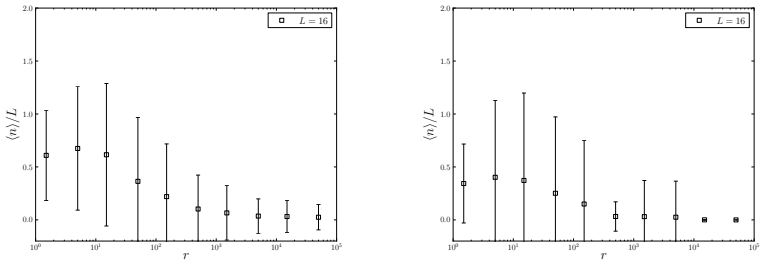
(a) (b)

FIGURE A.2: $L = 32$. (a) The mass difference of the backbone and global optimal path with increasing r. (b) Points show the average backbone mass, while line shows the average mass of the global optimal path.



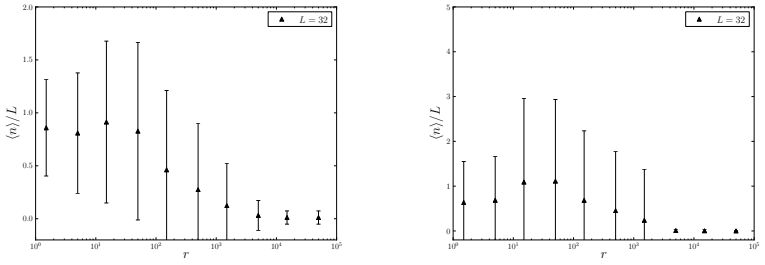
(a) (b)

FIGURE A.3: $L = 64$. (a) The mass difference of the backbone and global optimal path with increasing r. (b) Points show the average backbone mass, while line shows the average mass of the global optimal path.



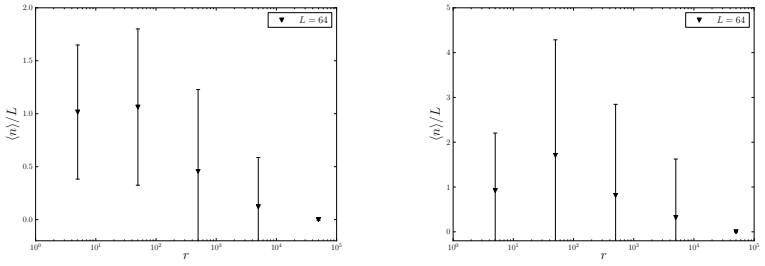
(a) (b)

FIGURE A.4: $L = 16$. (a) Overlap of backbone and the global optimal path. n denotes the number of bonds which are in the optimal path, but not in the backbone. (b) Overlap of the spanning clusters for the PP and general case. n denotes the number of bonds which are in the PP spanning cluster, but not in the general spanning cluster.



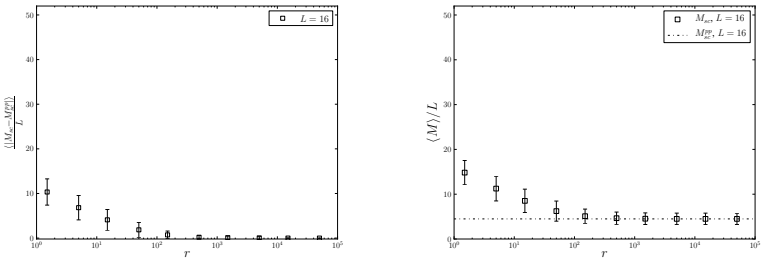
(a) (b)

FIGURE A.5: $L = 32$. (a) Overlap of backbone and the global optimal path. n denotes the number of bonds which are in the optimal path, but not in the backbone. (b) Overlap of the spanning clusters for the PP and general case. n denotes the number of bonds which are in the PP spanning cluster, but not in the general spanning cluster.



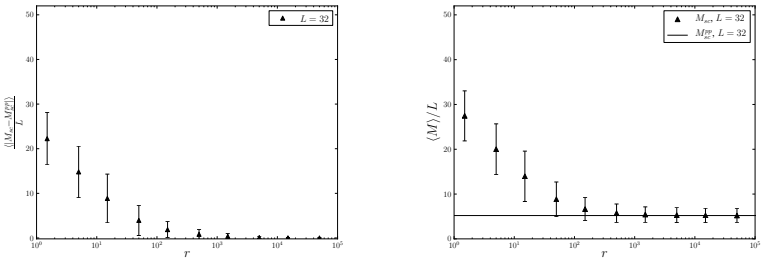
(a) (b)

FIGURE A.6: $L = 64$. (a) Overlap of backbone and the global optimal path. n denotes the number of bonds which are in the optimal path, but not in the backbone. (b) Overlap of the spanning clusters for the PP and general case. n denotes the number of bonds which are in the PP spanning cluster, but not in the general spanning cluster.



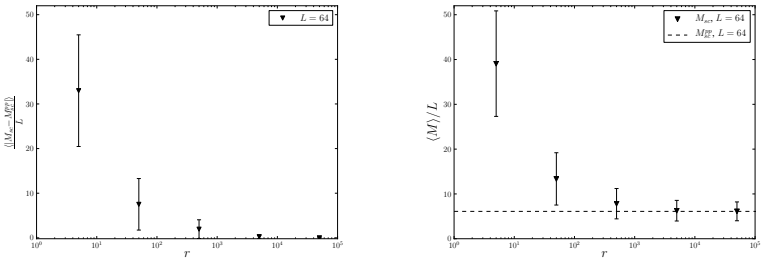
(a) (b)

FIGURE A.7: $L = 16$. (a) The mass difference of the spanning cluster and the PP spanning cluster, with increasing r. (b) Points show the average mass of the spanning cluster, while lines shows the average mass of the PP spanning cluster.



(a) (b)

FIGURE A.8: $L = 32$. (a) The mass difference of the spanning cluster and the PP spanning cluster, with increasing r. (b) Points show the average mass of the spanning cluster, while lines shows the average mass of the PP spanning cluster.



(a) (b)

FIGURE A.9: $L = 64$. (a) The mass difference of the spanning cluster and the PP spanning cluster, with increasing r. (b) Points show the average mass of the spanning cluster, while lines shows the average mass of the PP spanning cluster.

Appendix B

Paper draft

This appendix contain a draft for a paper based on the results in the plastic-elastic regime. The plan is to (possibly) publish these results at the IUPAP Conference on Computational Physics in Moscow, August 2013.

Optimal paths in random resistor networks

Morten Stornes* and Alex Hansen†

Department of Physics, Norwegian University of Science and Technology, N-7491 Trondheim, Norway

June 5, 2013

Abstract

In a random resistor network with a quenched random threshold distribution and a piecewise linear characteristic, the slopes of the characteristic, α and β, are important for how the spanning path develops. Using a hierarchial model, we argue that it is the ratio of the slopes, α/β which control the main behaviour of the spanning path in the network and the problem reduces to that of the perfect plastic for $\alpha/\beta \to \infty$ as the spanning path follows the optimal path. Results from simulations on a rotated square network are used to support our arguments.

INTRODUCTION

Random resistor networks are commonly used to study problems related to statistical physics. They are fairly easy to implement, and can be used as a simplified network to study e.g. brittle and ductile fracture in disordered media [1–4], directed polymer growth [1] and flow of Bingham fluids [5] in porous media.

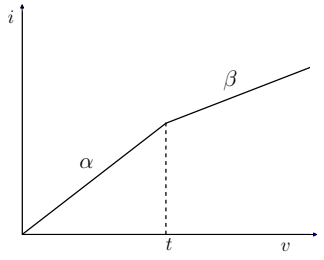


Figure 1: The characteristic of the conductor response. At the threshold t, the current through the bond changes.

Using the resistor network as a replacement for a material, the electric current, voltage drop and conductance map to stress, strain and elastic modulus in the material. For flow problems, the voltage drop map to the pressure/shear stress, the electric current to the flux/strain rate and the conductance map to the reciprocal viscosity. We will here focus on the plastic-elastic regime, i.e. α, $\beta \in [0, \infty)$, $\alpha > \beta$ as in Figure 1. For $\beta = 0$, the bonds act as perfect plastics,

*m.stornes@gmail.com
†Alex.Hansen@ntnu.no

and the spanning path through the network follows the optimal path.
The plastic-elastic regime can also be seen as the shear-thinning regime, as the current through the bonds will act as a piecewise shear-thinning fluid in the flow case, as per the dual lattice [6]. In this case, $\alpha \to \infty$ gives a Bingham behaviour.

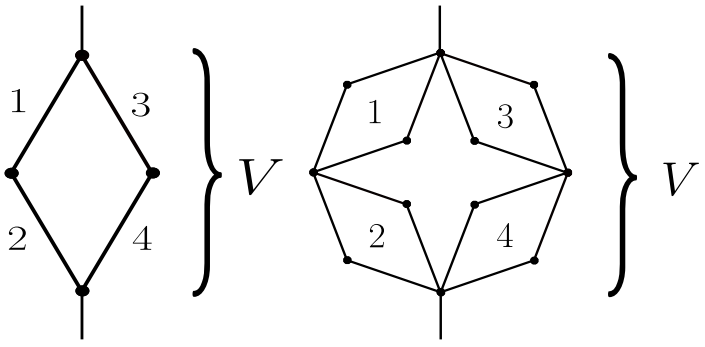
HIERARCHIAL MODEL

The basic hierarchial model consist of four bonds connected between two nodes with fixed voltages. The number of bonds and size of the network is increased by replacing each bond with a similar four-bond network, as in Figure 2. Doing this n times gives a network with 4^{n+1} bonds. Each network of size 4^{n+1} will then consist of four subnetworks of size 4^n, of which two are in series with each other, and there are two series in parallel.

The conductance of a bond with a character as in Figure 1 is given by $g = \alpha f$, with

$$f = \begin{cases} 1 & \text{if } v \leq t \\ \frac{1+(r-1)\frac{t}{v}}{r} & \text{if } v \geq t \end{cases} \tag{1}$$

where $r = \alpha/\beta$ has been used. Note that the conductance is continous, unlike what is usually the case for the fuse network used in fracture models, where the bond breaks irreversibly when $v = t$ and g is set to 0.



(a) Four-bond network (b) Sixteen-bond network

Figure 2: The two smallest networks in the hierarchial model. (a) shows the basic network consisting of 4 bonds, while (b) shows the network when each bond in (a) has been replaced by a four-bond network as in (a). The network is kept under a fixed voltage V.

The voltage drop over the entire network is fixed to V. Each bond has a randomly distributed voltage threshold t where the conductance changes. For two bonds in series in the four-bond network, k and l, e.g. bonds 1 and 2 in Figure 2(a), $i_k = i_l$ and $V = v_k + v_l$. The voltage drops v_k and v_l after bond k breaks are then

$$\begin{aligned} v_k &= \frac{V}{1+1/r} - t_k \frac{1-1/r}{1+1/r} = \frac{rV}{1+r} + t_k \frac{1-r}{1+r} \\ v_l &= \frac{V}{1+r} - t_k \frac{1-r}{1+r} \end{aligned} \tag{2}$$

For the case of $r \to \infty$, as is the case of a perfect plastic, $v_k = V - t_k$ and $v_l = t_k$. As $t_l > t_k$, bond l will never break, and no loops will form. In the four-bond network, the only two bonds to break will then be the ones with the lowest threshold in the left and right series.

Let's from now on only consider the case of $r \to \infty$. Looking at a more general network of size 4^{n+1}, which consist of four subnetworks of size 4^n, it will have the conductance

$$\begin{aligned} G &= (\tfrac{1}{g_1} + \tfrac{1}{g_2})^{-1} + (\tfrac{1}{g_3} + \tfrac{1}{g_4})^{-1} \\ &= \alpha(\tfrac{f_1 f_2}{f_1+f_2} + \tfrac{f_3 f_4}{f_3+f_4}) \\ &= \alpha F \end{aligned} \tag{3}$$

where g_m is the conductance of subnetwork m. If the network is broken, i.e. there is a spanning path through it, then, for $n = 0$,

$$\begin{aligned} F &= \tfrac{f_i}{1+f_i} + \tfrac{f_j}{1+f_j} \\ &= \tfrac{t_i}{v_i+t_i} + \tfrac{t_j}{v_j+t_j} \\ &= \tfrac{T}{V} \end{aligned} \tag{4}$$

Here, bonds i and j are the bonds which have broken, and they are not connected in series. Thus, $f_i = t_i/v_i = t_i/(V - t_i)$ as per Eqs. (1) and (2), and similar for j. Furthermore, we have used $T = t_i + t_j$, which denotes the threshold of the network.
Let's now look at half of the network for a general n. Looking at two subnetworks in series, k and l, the current conservation $i_k = i_l$ gives $\alpha f_k v_k = \alpha f_l v_l$. Combining this with $V = v_k + v_l$ gives

$$v_k = V - \frac{v_k f_k}{f_l}. \tag{5}$$

Assuming that subnetwork k is broken, and that $f_k = t_k/v_k$, as is the case for the four-bond network, $v_k = V - t_k/f_l$. This lead to

$$\begin{aligned} \tfrac{f_k f_l}{f_k+f_l} &= \tfrac{t_k f_k/V_k}{t_k/V_k+f_l} \\ &= \tfrac{t_k f_l}{t_k+f_l(V-t_k/f_l)} \\ &= \tfrac{t_k}{V} \end{aligned} \tag{6}$$

Thus, if two subnetworks i and j have broken and there is a spanning path in the network,

$$\begin{aligned} F &= \tfrac{t_i}{V} + \tfrac{t_j}{V} \\ &= \tfrac{T}{V} \end{aligned} \tag{7}$$

which is the same as Eq. (4). Thus, the assumption $f_k = t_k/v_k$ holds for any broken network k of size 4^n, and Eq. (7) is valid for any general broken network of size 4^{n+1}.

We now look at the cost of the spanning path in a general network. Again, looking at two subnetworks k and l in series and with current conservation giving $v_k f_k = v_l f_l$ and a fixed $V = v_k + v_l$, we have

$$\begin{aligned} V &= \tfrac{v_k}{f_l}(f_k + f_l) \\ &= \tfrac{v_l}{f_k}(f_k + f_l). \end{aligned} \tag{8}$$

As V is fixed (but increasing), the first subnetwork to break is the one which break at the lowest V. Thus, whichever subnetwork minimize V, i.e. $\min(V) = \min((f_k+f_l)v_k/f_l,\, (f_k+f_l)v_l/f_k) = \min(v_k f_k,\, v_l f_l)$, breaks first. Assuming that k break first, i.e. $v_k f_k = t_k$ as per Eq. (7), we have that $t_k < v_l f_l$, with $f_l \in (t_l/v_l, 1]$ as it is not broken. If l is of size 4^n and no bonds are broken, then the voltage drop over each bond is $v_i = v_l/2^n$. As $t_l/2^n \geq t_i$, where t_i is the minimum

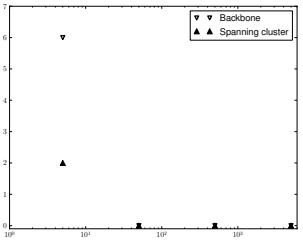


Figure 3: Number of bonds differing in the backbone and spanning cluster for the same value of r of a given system, $L = 64$.

threshold value in subnetwork l, $v_i < t_i \Rightarrow v_l < t_l$. Thus, if subnetwork k breaks first, then $t_k < t_l$ for both the lower and upper limit for f_l, and $t_k < t_l$ irrespective of how many bonds have broken in subnetwork l.
As can be seen from this, the first, and the only, of two subnetworks in series to break, will be the one with the minimum threshold. This subnetwork will then be part of the spanning path. Thus, for $r \to \infty$, the spanning path will follow the path $\mathcal{P}$ which minimize $\sum_{m\in\mathcal{P}} t_m$, i.e. the optimal path.

NUMERICAL RESULTS

The numerical results are produced using a random resistor network of size $L \times L$ with a quenched threshold distribution. This is a square network of conducting bonds rotated 45° and put between two electrodes at the top and bottom with periodic boundaries in the lateral direction. Each bond is assigned a random threshold drawn from a uniform probability distribution. The network is then put under a voltage drop V. A bond i is considered broken when $v_i \geq t_i$, and its conductance is adjusted to fulfill the given current-voltage relation as given by α and β. The voltage of all the nodes is then calculated using the conjugate gradient method. This is done under increasing V until a spanning path forms. This model is often used for studying brittle fracture [1–4], but then with a current threshold rather than a voltage threshold.

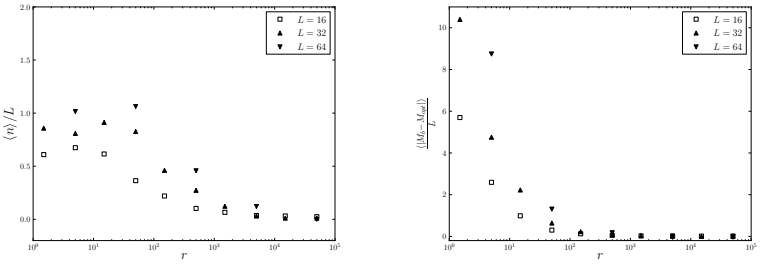
As seen in the hierarchial model, the only relevant parameters for where the spanning path forms, are the thresholds and $r = \alpha/\beta$. To check if this also holds for the rotated square network, several systems of size $L = 64$ were run through with r kept constant, but with different values of α and β. This was done for $r = 5$, 50, 500 and 5000, with $\beta = 0.1$, 0.5 and 0.9 and $\alpha = r\beta$. 20 different systems were checked for the three lowest values of r, and 12 systems for $r = 5000$. The number of bonds in the backbone and spanning cluster which were different in the three systems with identical threshold distribution and r were then counted. Each system was checked against the two others, and all disrepancies were counted. Thus, if one bond belonged to the backbone in the system with $\beta = 0.1$, but not in the two other systems, this was counted twice. The results are plotted in Figure 3.

As can be seen, the number of disrepancies is larger for the backbone than the spanning cluster. The reason for this is that the one bond which was not part of the spanning cluster in one of the three systems, opened up a loop in the backbone, causing two more bonds to be excluded from it, bringing the total backbone count up to 6. The disrepancies are only from a single system, and is likely due to round-off errors in the program, as all other systems have identical backbones and spanning clusters for the same r.

To see if the backbone follows the optimal path for large values of r, a pathscape[7] is created

for each system. This creates a hierarchy of optimal spanning paths, and gives each node a cost T which is the cost of the cheapest possible spanning path crossing that node. Nodes belonging in the optimal path will then have $T = T_{min}$, and all nodes in the optimal path will have the same value. If there is a spanning path of nodes with the same value $T_{path} > T_{min}$, these form a local minimum.

The number of bonds in the optimal path which are not covered by the backbone, n, is then counted. This is done for systems of size $L = 16$, $L = 32$ and $L = 64$, and the results are averaged over 200, 60 and 40 runs respectively. β is set to 0.1, and $\alpha = r\beta$, with different values of r used. The results are shown in Figure 4(a) and show that on average the backbone follows the optimal path more closely. Figure 4(b) shows the mass difference between the backbone and the optimal path, which also goes toward 0 for large r. Thus, as r increase, the mass of the backbone goes toward that of the optimal path, and the overlap increase, which is in agreement with the argument from the hierarchial model.



(a) (b)

Figure 4: (a) Average number of bonds in optimal path which are not covered by the backbone. (b) The mass difference of the backbone and global optimal path with increasing r.

CONCLUSION

We have shown here that the developement of the spanning path in a random resistor network, is dependent on the ratio r of the slopes of the local characteristic of the bonds. For different slopes in the characteristic, but same r, the spanning path is identical in systems with the same threshold distribution. Furthermore, we see that as $r \to \infty$, the bonds start behaving like a perfect plastic and the spanning path follows the optimal path.

References

[1] A. Hansen, E.L. Hinrichsen and S. Roux, *Phys. Rev. Lett.*, **66**, 2476 (1991)

[2] A.A. Moreira, C.L.N. Oliveira, A. Hansen, N.A.M. Araújo, H.J. Herrmann and J.S. Andrade Jr, *Phys. Rev. Lett.*, **109**, 255701 (2012)

[3] A. Shekhawat, S. Zapperi and J.P. Sethna, *Phys. Rev. Lett.* **110**, 185505 (2013)

[4] C.B. Picallo, J.M. López, S. Zapperi and M.J. Alava, *Phys. Rev. Lett.*, **105**, 155502 (2010)

[5] S. Roux, H.J. Herrmann, A. Hansen and E. Guyon, *C. R. Acad. Sci. Paris*, **305**, 943 (1987)

[6] J.P. Straley, *Phys. Rev. B*, **15**, 5733 (1977)

[7] L. Talon, H. Auradou, M. Pessel and A. Hansen, *Europhys. Lett.* (submitted for review)

Bibliography

[1] S. Roux, H.J. Herrmann, A. Hansen, and E. Guyon. Relation between different types of non-linear behavior in disordered lattices. *C. R. Acad. Sci. Paris*, 305:943, 1987.

[2] A.A. Moreira, C.L.N. Oliveira, A. Hansen, N.A.M. Araújo, H.J. Herrmann, and J.S. Andrade Jr. Fracturing highly disordered materials. *Phys. Rev. Lett.*, 109: 255701, 2012.

[3] A. Hansen, E.L. Hinrichsen, and S. Roux. Roughness of crack interfaces. *Phys. Rev. Lett.*, 66:2476, 1991.

[4] A. Shekhawat, S. Zapperi, and J.P. Sethna. From damage percolation to crack nucleation through finite size criticality. *Phys. Rev. Lett.*, 110:185505, 2013.

[5] S. V. Buldyrev, S. Havlin, and H.E. Stanley. Optimal paths in strong and weak disorder: A unified approach. *Phys. Rev. E*, 73:036128, 2006.

[6] M. J. Buehler. *Atomistic Modeling of Materials Failure*. Springer, New York, 2008.

[7] J.P. Straley. Critical exponents for the conductivity of random resistor lattices. *Phys. Rev. B*, 15:5733, 1977.

[8] D. Stauffer and A. Aharony. *Introduction to Percolation Theory*. Taylor & Francis, London, Washington, DC, 2nd edition, 1992.

[9] J. R. Shewchuk. An introduction to the conjugate gradient method without the agonizing pain. 1994. URL `http://www.cs.cmu.edu/~quake-papers/painless-conjugate-gradient.pdf`.

[10] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes*. Cambridge University Press, New York, 3rd edition, 2007.

[11] G.G. Batrouni and A. Hansen. Fourier acceleration of iterative processes in disordered systems. *J. Stat. Phys.*, 52:747, 1987.

[12] G.G. Batrouni, A. Hansen, and M. Nelkin. Fourier acceleration of relaxation processes in disordered systems. *Phys. Rev. Lett*, 57:1336, 1986.

[13] A. Hansen and J. Kertész. Phase diagram of optimal paths. *Phys. Rev. Lett.*, 93: 040601, 2004.

[14] L. Talon, H. Auradou, M. Pessel, and A. Hansen. Geometry of optimal path hierarchies. *Europhys. Lett.* (submitted for review).

[15] T. H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, Cambridge, Massachusetts, 3rd edition, 2009.

[16] M. E. J. Newman and R. M. Ziff. Fast monte carlo algorithm for site or bond percolation. *Phys. Rev. E*, 64:016706, 2001.