



Norwegian University of
Science and Technology

Fracturing of Optimal Paths in a Random Lattice

Andre Voigt

Master of Science in Physics and Mathematics

Submission date: June 2011

Supervisor: Alex Hansen, IFY

Abstract

The subject of this thesis is the study of the creation of fault lines in a random lattice, provoked by the successive failure of optimal paths. Using the recently developed Optimal Path Cracked model, we investigate how central characteristics of the successive optimal paths evolve as the lattice breaks down, and how this progression of characteristics depends on the magnitude of disorder imparted on the lattice.

We then see how the OPC model, while originally proposed in the context of the shortest path problem, can be generalized to alternate optimal path problems, namely the minimax problem and the widest path problem. It is shown that for a given lattice, the minimax OPC is equal to the backbone of the shortest OPC. The widest path OPC, although constituting a distinct object on any lattice, is shown to scale with lattice size in the same manner as the minimax OPC and the backbone of the shortest path OPC; with the fundamental process behind it being closely related to the minimax OPC process.

Lastly, we explain the connection between the OPC process and a variety of other phenomena which have previously been shown to exhibit similar scaling behavior. We show how the OPC process for the widest path problem can be reduced to the shortest path problem on the dual lattice using the limit of very high disorder, the so-called ultrametric limit, and how an algorithm based on invasion percolation can be used as a quicker method of finding an OPC.

Preface

The work presented here constitutes my Master's thesis, the final required step for my Master's degree in Physics and Mathematics (MTFYMA) at NTNU. The entirety of the work was carried out during the spring semester of 2011, and corresponds to thirty credit points. The thesis was written at NTNU's Department of Physics, under the supervision of Prof. Alex Hansen.

In addition to my supervisor, whose guidance provided valuable help, I want to thank the many Ph.D. students present at our weekly discussion group, who also provided valuable input, and who showed a genuinely encouraging interest in my work. Further thanks go to Terje Røsten for providing access to the NTNU Department of Physics computing cluster, and to José Soares Andrade Jr of Universidade Federal do Ceará, Brazil, for kindly providing one of his papers. Finally, I want to thank my family for encouragement and support.

Contents

1	Introduction	1
1.1	Motivation and purpose	1
1.2	Scope and limitations	2
1.3	Background	2
1.3.1	Optimization problems in random graphs and ubiquity of scaling exponents	2
1.3.2	Fracturing of optimal paths	3
1.4	Outline	4
1.5	List of terms and abbreviations	4
2	Model	7
2.1	Lattice configuration	7
2.2	The Optimal Path Crack model	10
3	Properties of the OPC fault line for the shortest path prob- lem	15
3.1	Fundamental behavior	15
3.2	Progression of path characteristics	17
3.2.1	Total path energy	18
3.2.2	Progression of energy values for successively removed bonds	18
3.2.3	Distribution of energies on the OPC bonds	22
3.3	Path mass	26
3.4	Significance of higher energy bonds	30
4	Alternate optimal path problems	35
4.1	Minimax problem	35
4.1.1	Similarity to the shortest path problem	36
4.1.2	Ultrametric limit and the exact shortest path	40
4.2	Widest path problem	41

5	Invasion percolation, minimum spanning trees and the ultrametric shortest path	47
5.1	Minimum spanning trees	47
5.2	Strands on invasion percolation	48
6	Common universality of OPCs and the ultrametric shortest path	53
6.1	Computational implications	57
7	Conclusion	59
7.1	Properties of the bonds of the shortest path OPC	59
7.2	Common universality of processes	60
7.3	Possibilities for further research	60

Chapter 1

Introduction

1.1 Motivation and purpose

Random graphs are widely used to represent a variety of systems, ranging from physical entities, such as porous media or human infrastructure (like pipes or roads), to more abstract entities like communication networks or even social circles. In many cases, a primary object of interest is the manner in which a certain entity (such as a fluid, or information) moves from one point (or area) to another; particularly the most efficient way in which it does so, which will often correspond to movement along the optimal path in the specified random graph. This path is generally defined as that maximizing or minimizing certain characteristic; which may, among other possibilities be the path's length, cost, travel time or bandwidth.

As finding optimal paths over a random graph can reveal crucial information about a given entity, understanding how these are affected in a dynamic system is often key in determining its behavior. If a minor part of a larger system prone is to failure, even a local breakdown may have wider consequences due to possible long-range interactions and cascade effects. Predicting how such minor failures may affect the wider system may therefore provide crucial information, making it possible to anticipate and prevent (or at least limit) negative consequences.

The goal of this thesis is the exploration of a newly introduced model describing the appearance of such failures, in order to get a better understanding of the breakdown of paths on random graphs. Additionally, we explore how this model is related to several other problems originally described in contexts entirely distinct from graph breakdown, which, in addition to furthering our

understanding of the related processes, has possible applications in terms of permitting the prediction of the lattice breakdown through the use of more efficient algorithms.

1.2 Scope and limitations

The goal of this thesis is primarily to explore how the critical points of a given graph evolve as this graph breaks down. As most of the attention is geared towards these critical points, the effect of unexpected behavior elsewhere in the system may not necessarily be accounted for in the models used. Most of the processes explored are also entirely deterministic for a given random graph (the only random element is the graph itself, as will be explained in depth in the following chapter) and may therefore be inaccurate for a variety of systems often represented by random graphs. The models used are essentially theoretical constructs, and are not intended to represent any particular real-world problem; at best, they give an indication as to how more specific models may behave.

The entirety of the data presented is derived from numerical computations: no physical experimentation has been performed in order to confirm the validity of any model properties, as no specific material problem has been the center of attention. We should therefore keep in mind any inherent limitations of computers, such as the necessarily discrete nature of any calculations involved and the upper bounds on useable numbers (which does place limitations on the possible range of some parameters).

Lastly, we keep in mind the fact that even highly consistent behavior is not necessarily universal: it is entirely feasible that exceptions to a general rule can occur in principle, but are unlikely to such a degree that they are never observed. Strong conclusions as to the workings of a system should therefore be avoided unless they can be backed up by logical proof.

1.3 Background

1.3.1 Optimization problems in random graphs and ubiquity of scaling exponents

An important part of graph theory is the study of optimal graphs, which may include any graph which minimizes or maximizes a set criterion. One of

the most familiar optimization problems is that of the shortest path: that is, given a source node, a target node, and a set of edges with an assigned cost function, finding the path from the source to the target node where the sum of the cost of the traversed edges is minimal. Besides its obvious applications in real-world pathfinding, it is also a topic of interest in statistical physics, being related to such problems as directed polymers in random media [1] and domain walls in spin glasses and ferromagnets [2]. An interesting aspect of the shortest path is how its topology depends on the degree of disorder: the path crosses over from a self-affine to a self-similar fractal as the disorder of the landscape increases [3] [4].

Another sort of optimization problem is that of minimum spanning trees (MSTs) [5], where we concern ourselves with connecting all nodes in a given graph, while minimizing the sum of the weight of the edges connecting it. Unlike the shortest paths, the MST of a disordered graph is independent of the degree of disorder [6], depending only on the relative rank of the edge weights. In a quadratic lattice, it is found that the number of bonds on the MST connecting two vertices separated by an Euclidean distance L scales as L^D with $D = 1.22$, which, incidentally, is the same as the fractal dimension of the shortest path in highly disordered landscapes [2]. In fact, scaling exponents close to $D = 1.22$ are also found for other phenomena such as strands on invasion percolation [7] and watershed lines [8]. This has led to the hypothesis that these phenomena may be related, and form a distinct universality class [9].

1.3.2 Fracturing of optimal paths

A recent paper by Andrade *et al* [10] introduced the Optimal Path Crack (OPC) model, which explores how system-wide failures spread in a disordered landscape, through the successive failure of shortest paths. The model finds the optimal path through the landscape, and disables this path by blocking its least optimal site. It then finds the new optimal path, and disables this in the same way. This is then repeated until the blocked sites form a spanning fault line across the landscape, and no paths remain. It was found that the topology of the crack largely depends on the magnitude of disorder: for highly disordered landscapes, the resultant crack is a singly connected fractal, with the same fractal dimension as the shortest path in the ultrametric limit. In the case of lower disorder, however, far more sites are blocked, creating several disconnected fractures. Interestingly, the backbone (the main fault

line) of the crack remains the same, regardless of disorder; which indicates that the OPC process also belongs to the same proposed universality class mentioned previously [9]. A more in-depth description of the OPC model will be presented in the next chapter.

1.4 Outline

We will begin by defining the fundamental parameters of the graphs studied throughout the thesis, followed by a description of the OPC algorithm, which will be the central process studied. The next chapter contains various results quantifying certain aspects of the behavior of the basic OPC process for a variety of parameters.

We then proceed to investigate the effect of slight modifications to the original lattice has on the formation of fault lines in the OPC model, followed by a comparison of the original OPC model and variants with some basic alterations. These results are then compared with those of processes which, while entirely different from the OPC model in principle, are used in related problems and exhibit a degree of similarity in their behavior.

1.5 List of terms and abbreviations

For reference, we keep a compact list of some of the central quantities, definitions and parameters used throughout the thesis, along with the abbreviations, symbols and terms commonly used to refer to each. In addition to the overview here, they will also be formally introduced in the following chapters, as they become necessary.

- (OPC) Backbone: the subset of the OPC, such that no bond on it can be opened without causing the existence of a spanning path.
- Bond (or site) energy, ϵ : randomly distributed value assigned to each bond (or site). Energies of specific bonds may be denoted by subscripts or bond coordinates $\epsilon(i_1, j_1, i_2, j_2)$.
- Disorder parameter, γ : defines the distribution of ϵ according to the relation $\epsilon = \rho^{\frac{1}{\gamma}}$.
- Edge: either of the four sets of lattice sites with coordinates $i = 0$, $i = L - 1$ (the vertical edges), $j = 0$ or $j = L - 1$ (the horizontal edges).

- Energy of optimal known path, $V(i, j)$. For the node (i, j) , the central value which is either to be minimized or maximized by a given optimal path problem.
- Lattice size, L : square root of the number of sites on the lattice.
- Mass, M : number of bonds forming a given structure.
- Minimax path: spanning path for which the energy of the costliest bond ϵ_{max} is minimal.
- Optimal path, S : common term for the minimax, widest and shortest paths.
- Optimal path crack (OPC) process/method/algorithm: the process previously described in this chapter.
- Optimal path crack (OPC): the set of bonds removed by the OPC process.
- (Total) Path energy, E : sum of the energy of all bonds (or sites) over a given path. May be denoted by subscripts.
- ρ : random variable, drawn from the uniform distribution over $[0:1]$. Related to the bond (or site) energy by $\epsilon = \rho^{\frac{-1}{\gamma}}$.
- Scaling exponent, or fractal dimension, D : for a structures of mass M , the value solving $M \propto L^D$.
- Shortest path: spanning path for which the total energy E is minimal.
- Spanning path: path connecting any source site to a target site. Unless otherwise specified, this will correspond to a path connecting the upper and lower edges of the lattice.
- Widest path: spanning path for which the energy of the cheapest bond ϵ_{min} is maximal.

Chapter 2

Model

2.1 Lattice configuration

Throughout this thesis, the object of study is a two-dimensional regular quadratic lattice, consisting of a set of L^2 regularly spaced sites (which are held in an ordinary two-dimensional C++ array), each of which can be described by ordinary integer cartesian coordinates (i, j) , (which must be of an integer value between 0 and $L - 1$). For order, whenever specific coordinates are used, we shall assume the point $(0, 0)$ to be the leftmost site on the lower edge of the lattice, and $(1, 0)$ the one immediately to its right.

Each site is connected by a single bond to any and all sites directly adjacent (immediately above, below, right or left; but not diagonally adjacent). Bonds may not connect non-adjacent sites. The position of each bond is described by the pair of cartesian coordinates (i_1, j_1, i_2, j_2) , where (i_1, j_1) and (i_2, j_2) are the coordinates of the sites on either end of the bond. The lattice corresponds to an undirected graph, hence, the coordinates (i_1, j_1, i_2, j_2) and (i_2, j_2, i_1, j_1) represent the same bond.

Unless otherwise specified, the lattice is subjected to periodic boundary conditions in the horizontal direction, but not vertically: a site on the rightmost edge of the lattice is connected by a single bond to the leftmost site with the same y -coordinate, while sites with a y -coordinate of 0 or $L - 1$ may not be connected to each other in a similar manner. For the sake of clarity, the term "edge" will be used nearly exclusively (with a minor exception when discussing minimal spanning trees, in accordance with graph theory terminology) to refer to the four extremities of the lattice ($i = 0$, $i = L - 1$, $j = 0$ and $j = L - 1$), while the connections between sites will be consistently

referred to as "bonds".

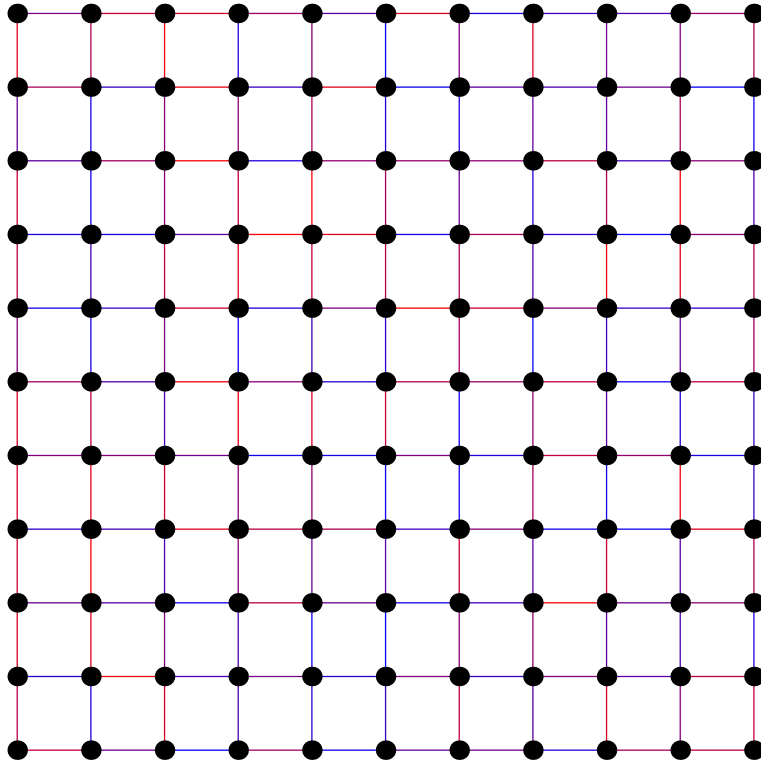


Figure 2.1: *A simple representation of the main lattice model studied throughout this thesis. The bonds are colored to represent the randomly drawn energy value: red shades represent higher energy values, blue shades represent lower. The black dots represent sites, which are not assigned a random energy value. Contrast with figure 2.2.*

For each bond, we associate a positive (not necessarily integer) dimensionless energy value ϵ , which will generally be considered to constitute the "cost" of moving from either of the ends of the bond to the other (although we will also consider a case where it can be viewed as the capacity of the bond). When necessary, we will use $\epsilon(i_1, j_1, i_2, j_2)$ to denote the energy of the bond (i_1, j_1, i_2, j_2) . For convenience, all bonds on both the upper and lower edge of the lattice will be assigned an energy value of 0. The other bonds will have their energies drawn at random from a power law distribution when the

lattice is first created. The power law distribution is obtained by drawing:

$$\epsilon = \rho^{-\frac{1}{\gamma}} \quad (2.1)$$

where ρ is drawn from the uniform distribution over $[0, 1]$. γ is a parameter defining the system disorder: lower values correspond to high disorder. In practice, values of γ below 0.25 have a general tendency to create energy values exceeding the range of the data type used to store energy values (C++ *double long*, which allows values up to $3.4 \cdot 10^{308}$), which imposes an upper limit on the possible level of disorder. All reasoning presented will base itself on the assumption that no two energy values in a lattice are exactly equal; due to the limited range of the uniform number generator used (up to 2^{31} different values are possible), this is not necessarily the case for the larger simulated lattices ($L = 200$ or above).

In addition to its position in the lattice and its energy value, each bond is also defined by a Boolean value determining whether or not it is "open", that is, whether any valid path may include it. For a given bond, setting this value to 0 is equivalent to assigning it an infinite energy value, or removing it from the lattice altogether. The bonds on the horizontal edges of the lattice are always considered to be open. An example of the lattice configuration is shown in figure 2.1.

A closely related but still distinct lattice configuration assigns a random energy value to each site in the lattice, instead of the bonds. Blockages are also considered to occur at the sites, and not the bonds; blocking one site corresponds to simultaneously blocking up to four bonds. The bonds and sites remain connected in the same way. Although the modification is relatively minor, the two systems are considered different problems in percolation theory: the first configuration is used in what we call *bond* invasion percolation, while the latter is used for *site* invasion percolation. In order to avoid confusion between the two models, we will refer to the first one (where energy values are assigned to bonds) as the "bond-oriented" approach, while the latter constitutes the "site-oriented" approach. Figure 2.2 shows the site-oriented configuration, contrasting with the bond-oriented lattice shown in figure 2.1. Although the site-oriented approach was the one used in the original definition of the OPC model [10](which will be explained in the next section), this thesis concerns itself primarily with the bond-oriented configuration. Unless the opposite is explicitly specified, the results and arguments presented apply to this (bond-oriented) configuration.

In both approaches a higher energy bond or site will be said to be "costlier" or "greater", while a lower energy one is said to be "cheaper" or "lesser".

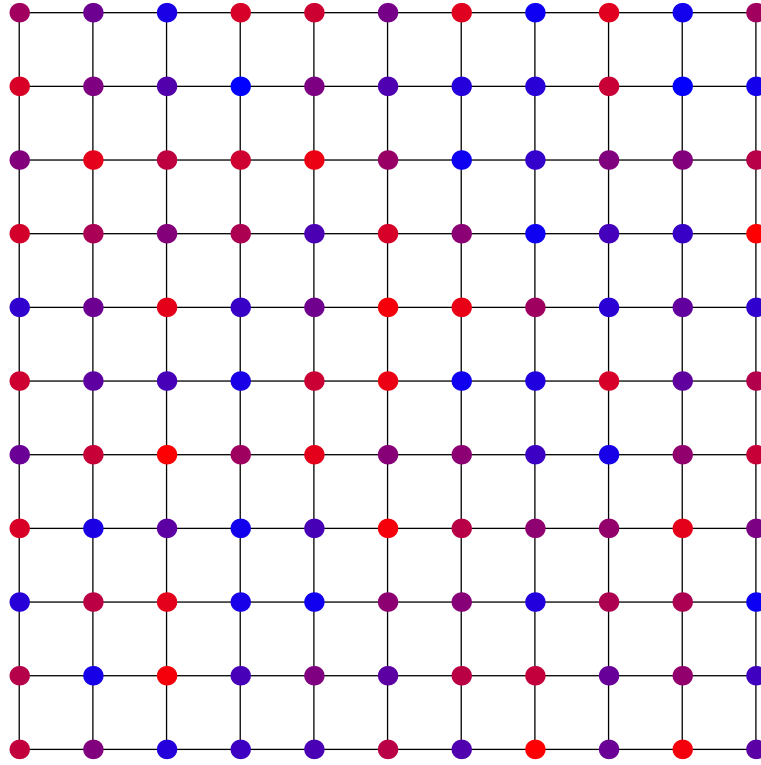


Figure 2.2: A representation of an alternate lattice model used in the original definition of the OPC model, including the original formulation of the OPC problem. Here, it is the sites that are assigned an energy value, and are colored accordingly (using the same color scheme as in figure 2.1). The bonds, in turn, are not assigned an energy value, and are therefore left black.

2.2 The Optimal Path Crack model

The first object of study of this thesis is the formation of lattice-spanning fault lines through the so-called Optimal Path Crack (OPC) algorithm, introduced in 2009 by Andrade *et al* [10]. It bases itself on the argument that in a transportation system containing multiple independent acting elements sharing the same environment, each of these actors acting selfishly is likely to put the highest strain on the solution optimal to the needs of the greatest number. An optimal path is therefore liable to attract a considerable amount

of traffic. If the load remains relatively uniform along that given path, the least optimal segment of that path is the most likely to experience a load greater to its capacity, and consequently break down. This breakdown will in turn affect the optimality of the entire path: consequently, the load is likely to adapt itself to another optimal solution, which in turn has a most vulnerable segment which may be liable to breakdowns. The OPC problem concerns itself with modelling such breakdowns and understanding how they behave.

The aforementioned process is reproduced by the following steps:

1. Find the optimal path connecting any site on the lower edge of the lattice to any site on the upper edge of the lattice. If no such path is found, the algorithm terminates. The optimal path is defined as the path connecting the horizontal edges of the lattice whose total energy cost $E = \sum_i \epsilon_i$ (which is, the sum of the energies ϵ_i of each bond or site it traverses) is minimal. For a given lattice, we denote by E_{min} the total energy cost of the optimal path. If no path connects the lattice's horizontal edges, terminate.
2. Find the bond along this path with the highest energy.
3. Block this bond (or set its energy value to infinity).
4. Return to step 1. As the optimal path previously found is no longer valid, we will find a new "next best" path.

We denote "next best" with quotation marks as the next path found is not necessarily the second most optimal path through the lattice in its original form: it is entirely possible that the actual second lowest energy path included the highest energy bond of the shortest path, and was therefore disabled by the removal of that bond.

For the sake of clarity, we notice that setting $\epsilon = 0$ for the horizontal edges of the lattice does not affect which of the non-zero bonds constitute the optimal path (technically, while there is no *unique* optimal path, they all share the same non-zero bonds). This will be of great importance in Chapter 6.

Once the algorithm has finished running, we have what was originally a quadratic lattice, with a number of bonds removed in such a way that no vertically spanning path exists. This set of bonds constitute what is called the Optimal Path Crack (from which the process draws its name), or OPC for

short. We will pay special attention to a subset of this crack, which consists of those bonds on the OPC which cannot be re-opened without this causing the existence of a vertically spanning path. We call this subset of bonds the "backbone" of the OPC. A central aspect of this thesis is how the size of the OPC backbone depends on the lattice size L : this size is determined by the bonds included in it, which we will call the backbone mass, M_b . Although we will not pay much attention to the size of the entire OPC, the mass of the OPC can be similarly defined, and will be denoted as M_{OPC} where needed.

In the original formulation of the OPC method, Dijkstra's algorithm [11] is used to compute the optimal path between the top and bottom edges of the lattice, as the optimal path problem as defined here is the same as what is commonly called the *shortest path problem* (which Dijkstra's algorithm was originally devised in order to solve). As we will also consider other problems whose solutions constitute what can be also be thought of as "optimal" paths, we will therefore refer to this specific optimal path problem as the "shortest path problem" for the remainder of this thesis. The solution path to this problem will also be referred to as the "shortest path". However, in order to avoid confusion, we will refrain from using the term "length" in describing characteristics of a path, using the terms "total path energy", E , for the sum of the energy values of the constituting bonds, and "mass", M_S , for the number of bonds forming the path (as we have with the OPC and its backbone).

Throughout this thesis, however, Dijkstra's algorithm has been substituted for another algorithm based on the repeated relaxation of path costs, originally proposed by Hansen and Kertész [1]. We first introduce the value $V(i, j)$, which is the total energy cost of the shortest known path connecting the site (i, j) to any of the source sites. The algorithm works as follows:

1. We designate all sites on the lower edge of the lattice (that is, with $j = 0$) as sources. They are marked as "linked" (to the lower edge), with $V(i, j) = 0$. All other sites are marked as "unlinked", with $V(i, j) = \infty$
2. For each linked site in the lattice with coordinates (i_1, j_1) , execute the following steps for all non-infinite bonds (i_1, j_1, i_2, j_2) connected to this site:
 - (a) Compute a tentative total path cost $V_t(i_2, j_2)$ for the site on the other end of the edge, $V_t(i_2, j_2) = V(i_1, j_1) + \epsilon(i_1, j_1, i_2, j_2)$.

- (b) If the site (i_2, j_2) is not linked, set $V(i_2, j_2) = V_t(i_2, j_2)$, and change its status to "linked". If it is linked, set $V(i_2, j_2) = \min(V(i_2, j_2), V_t(i_2, j_2))$.
- 3. If $V(i, j)$ has been changed by step 2 for of any site in the lattice, return to step 2.
- 4. Otherwise, the system has reached steady state, and each value of $V(i, j)$ is the cost of the shortest possible path linking (i, j) to the lower edge of the lattice.

Regardless of the algorithm used, once the cost of the shortest path from the lower edge of the lattice to each given site is known, we can search the upper edge of the lattice for the the site with the lowest value of $V(i, L - 1)$. When this is found, we can find the last bond of this path, which is the bond whose energy value equals the difference between the shortest path costs of the sites it connects. We can then repeat this until we find the entire shortest path, while at each step checking if the last bond found is the costliest bond on the shortest path (as has been found so far).

Chapter 3

Properties of the OPC fault line for the shortest path problem

3.1 Fundamental behavior

Figure 3.1 shows different resultant OPC patterns on 4 lattices, generated from the same seed with various degrees of disorder. We clearly see that, as disorder is reduced, more bonds are removed by the process; the backbone (that is, the looping fracture severing the lattice), however, remains unchanged, and is very nearly the only thing remaining in the highest disorder lattice. This is entirely consistent with the findings of Andrade et al [10] in the original (site-oriented) formulation of the OPC method.

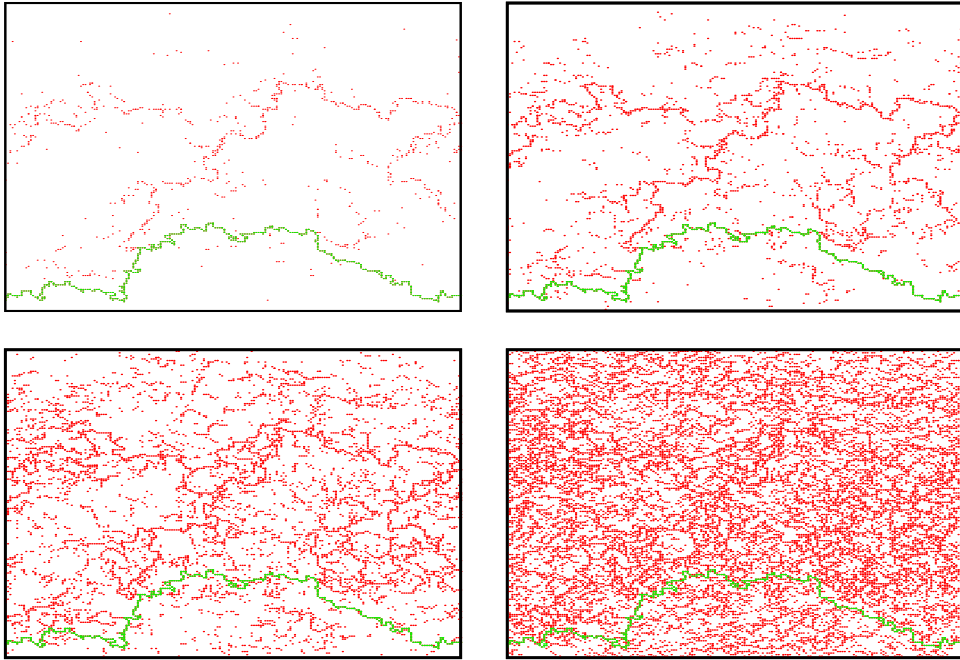


Figure 3.1: *OPCs patterns on four 500 by 500 lattices generated from the same seed, with different disorder parameters. Green points correspond to bonds on the backbone. Top left: $\gamma = 0.25$, top right: $\gamma = 0.5$, bottom left: $\gamma = 1.0$, bottom right: $\gamma = 5$.*

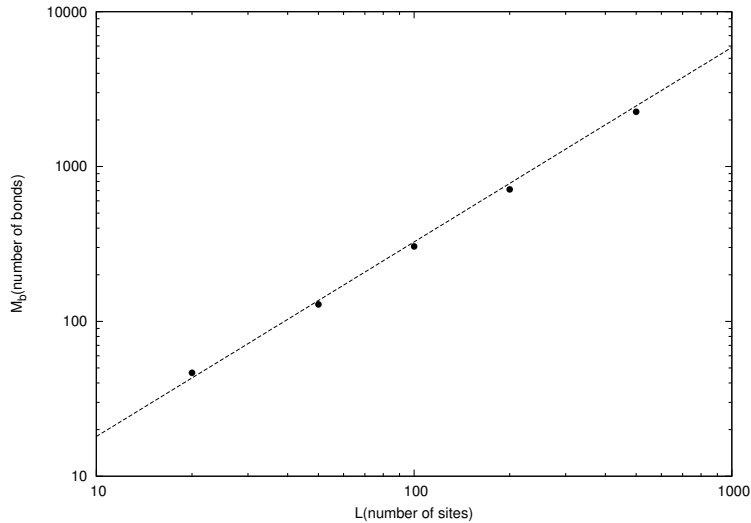


Figure 3.2: Average size M_b of the OPC backbone (in number of bonds) as a function of the lattice size L . The slope of the dotted line is $D = 1.227$. Each data point is obtained from a sample of 50 OPC backbones. The error bars are approximately the same size as the data points.

Figure 3.2 shows how the size M_b (in number of edges blocked) of the OPC backbone scales with the lattice size L . As in the site-oriented formulation, we find that the mass of the OPC backbone scales as $M_b \approx L^D$, with the found value of $D = 1.227 \pm 0.015$. This value of D for the bond-oriented OPC backbone is equal (within statistical error bounds) to the one found for the site-oriented method, which suggests that both approaches are equivalent.

3.2 Progression of path characteristics

While the single shortest path through a disordered lattice is a relatively well-studied phenomenon, the formation of the OPC is also determined by the characteristics of the less-optimal paths which are progressively selected as the lattice geometry changes; information about these could serve to explain the general properties of the OPC lattice. We will look at three basic characteristics: a path's total energy E , the energy ϵ_{max} of the costliest bond along that path, and the mass M of the path (in number of bonds crossed); and how they depend on each other, lattice size, and the amount of disorder.

3.2.1 Total path energy

Before investigating in any detail, we can already determine a basic (and rather trivial) detail about the energy of the chosen paths: as the OPC method only removes bonds (or sites, depending on formulation), any shortest path found at a given iteration has also been a *possible* path at each previous iteration. Therefore, all shortest paths found for earlier lattice configurations are shorter than this one. If we then plot the total energy of each chosen path as a function of the order in which they were chosen, we must necessarily get a strictly increasing curve. Again, this point might be trivial, but it should be kept in mind. The non-optimality of a given path can be the result of two things: either, it is because the path needs to span a large amount of bonds, or it is because it needs to pass through one or several costly bonds. We will explore the impact of both factors:

3.2.2 Progression of energy values for successively removed bonds

As we see in figures 3.3-3.5, the costliest edge's share of the total path energy depends on both the path length and the disorder parameter. In the high disorder regime, the results are rather predictable: the costliest edge accounts for a relatively large amount of the total path cost, especially for the less optimal paths, where the bond/path energy ratio very nearly reaches unity. The few non-backbone bonds are the ones of lower energy, where the other bonds on the path constitute a relatively important share of the total energy. The medium disorder regime, in turn, exhibits nearly no correlation between the cost of the costliest bond and the total path cost. There is also no obvious pattern as to the distribution of the backbone bonds which could be considered distinct from the others. The low disorder case is perhaps the most surprising: the less optimal paths actually tend to correspond to lower values of ϵ_{max} .

We also notice that while disorder seemingly has no bearing on which bonds constitute the backbone, it strongly affects the order in which they are added: higher disorder sees the backbone bonds generally added in order of increasing energy, while the order appears far more random for lower values.

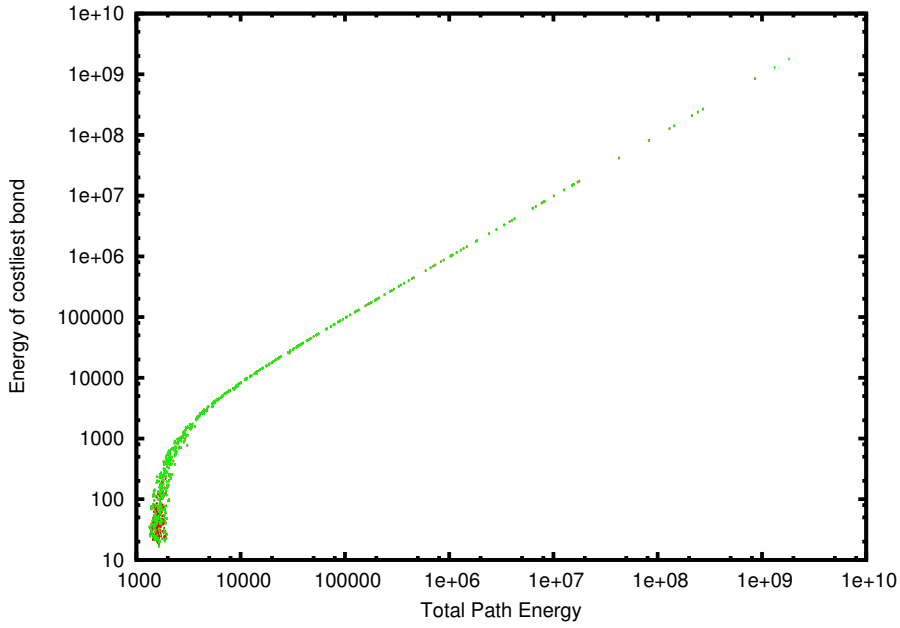


Figure 3.3: Energy ϵ_{max} of each bond removed by the OPC process on a single lattice ($L = 200$), as a function of the total energy E of the corresponding path for $\gamma = 0.25$ (using the the same random seed as in figures 3.4 and 3.5. Green dots (slightly magnified for visibility) denote bonds on the backbone, while red dots denote other bonds. Note that both axes follow a logarithmic scale, in contrast with figures 3.4 and 3.5. For greater visibility, five data points have been truncated, corresponding to $E \approx \epsilon_{max} \approx 1.32 \cdot 10^9$, $E \approx \epsilon_{max} \approx 1.81 \cdot 10^9$, $E \approx \epsilon_{max} \approx 8.42 \cdot 10^{12}$, $E \approx \epsilon_{max} \approx 4.61 \cdot 10^{13}$, $E \approx \epsilon_{max} \approx 2.57 \cdot 10^{17}$.

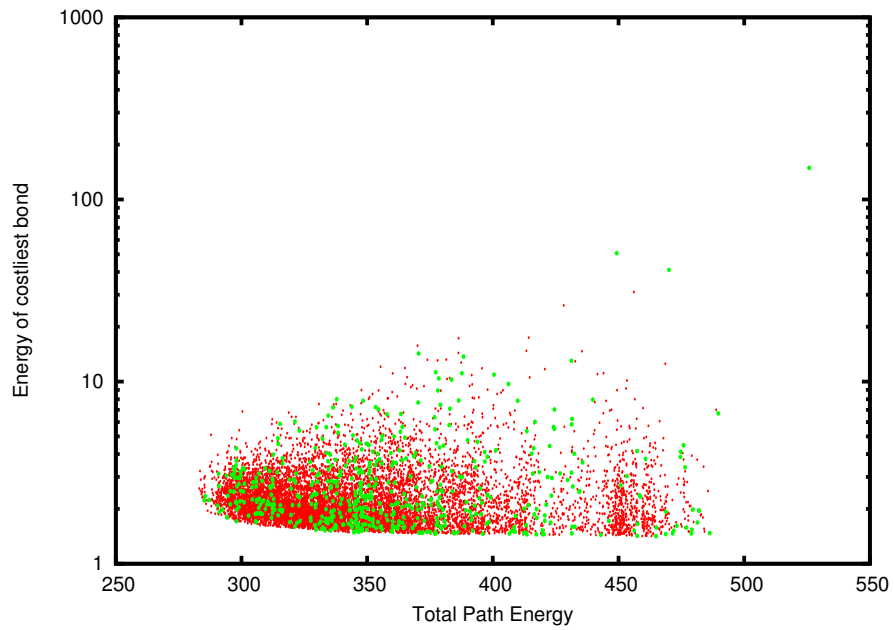


Figure 3.4: Energy ϵ_{max} of each bond removed by the OPC process on a single lattice ($L = 200$) as a function of the total energy of E the corresponding path for $\gamma = 2$. Green dots (enlargened for visibility) denote bonds on the backbone, while red dots denote other bonds. Note that only the y -axis follows a logarithmic scale, in contrast with figures 3.3 and 3.5.

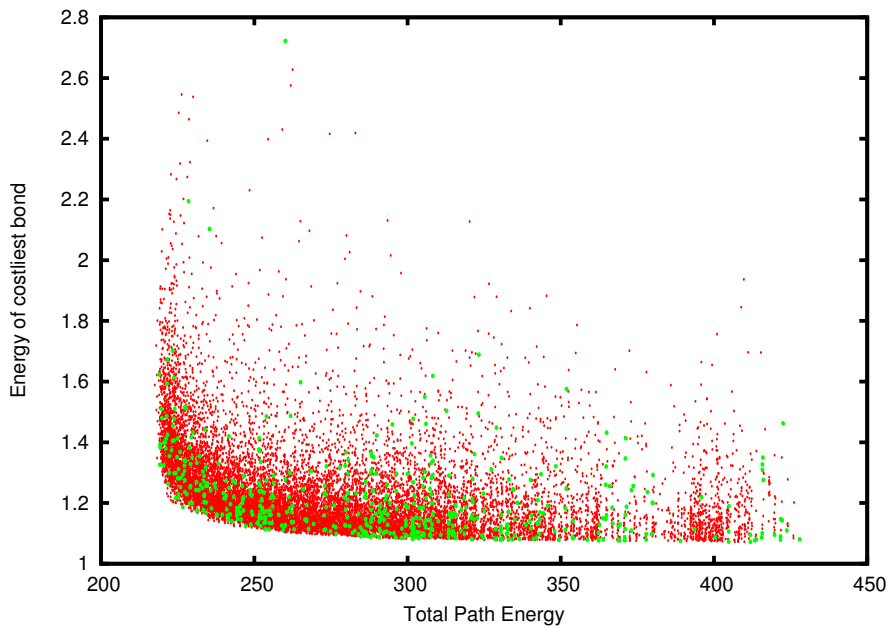


Figure 3.5: Energy ϵ_{max} of each bond removed by the OPC process on a single lattice ($L = 200$) as a function of the total energy E_{tot} of the corresponding path for $\gamma = 10$. Green dots (enlarged for visibility) denote bonds on the backbone, while red dots denote other bonds. Note that both axes follow a linear scale, in contrast with figures 3.3 and 3.4.

3.2.3 Distribution of energies on the OPC bonds

While the scatterplots in figures 3.3-3.5 did not show any obvious differences between the backbone and non-backbone paths in terms of ϵ_{max} , figures 3.6 and 3.7 demonstrate different likelihoods of bonds with different energies belonging to the OPC. We visualize the energy as $1 - \rho$, as ρ is uniformly distributed over $[0:1]$ for any value of γ (which means $1 - \rho$ is as well), which permits an easier comparison of different disorder regimes. We recall that ϵ decreases for increasing ρ ; the choice of $1 - \rho$ for the x-axis corresponds to having higher energy bonds on the right side of the plot, which follows the scheme used in previous figures.

Several observations can be made: firstly, only bonds above a certain energy threshold participate in the OPC. Second, the value of ρ for which this happens appears to be independent of disorder (but dependent on lattice size), at about $\rho = 0.6$ for $L = 40$ and $\rho = 0.56$ for $L = 100$. Since ρ is uniformly distributed over $[0 : 1]$, this can be interpreted to mean that the cheapest 40% (for $L = 40$) or 44% (for $L = 100$) of bonds never break down.

Third, the probability of a bond with a given energy being a non-backbone OPC bond is highly dependent on disorder, with lower disorders drawing heavily from high-energy bonds; the plausible explanation being that a bond being part of the OPC is determined by a balance of two factors: on one hand, lower energy bonds are more likely to be part of an optimal path, on the other, they are less likely to be the maximal energy bond along said path. Lower disorder reduces the impact of the first, naturally increasing the chance of high-energy bonds being part of the OPC. Additionally, for a given value of ρ , the probability of belonging to the OPC will not increase indefinitely as we reduce disorder:

Lastly, the probability $P_b(\rho)$ of a bond with energy $\epsilon = \rho^{-\frac{1}{\gamma}}$ belonging to the OPC backbone appears to follow

$$P_b(\rho) = A - B \exp[-\lambda(1 - \rho)] \quad (3.1)$$

for values of ρ between 0 and approximately 0.5. Fitting P_b over the range $0.5 < \rho < 1$ as to minimize the residual sum of squares yields $A = 0.0287$, $B = 2.314$ and $\lambda = 9.622$ for $L = 40$, and $A = 0.0177$, $B = 2.742$ and $\lambda = 10.683$ for $L = 100$. Figure 3.8 shows that this appears to be a reasonable fit, especially for $L = 100$.

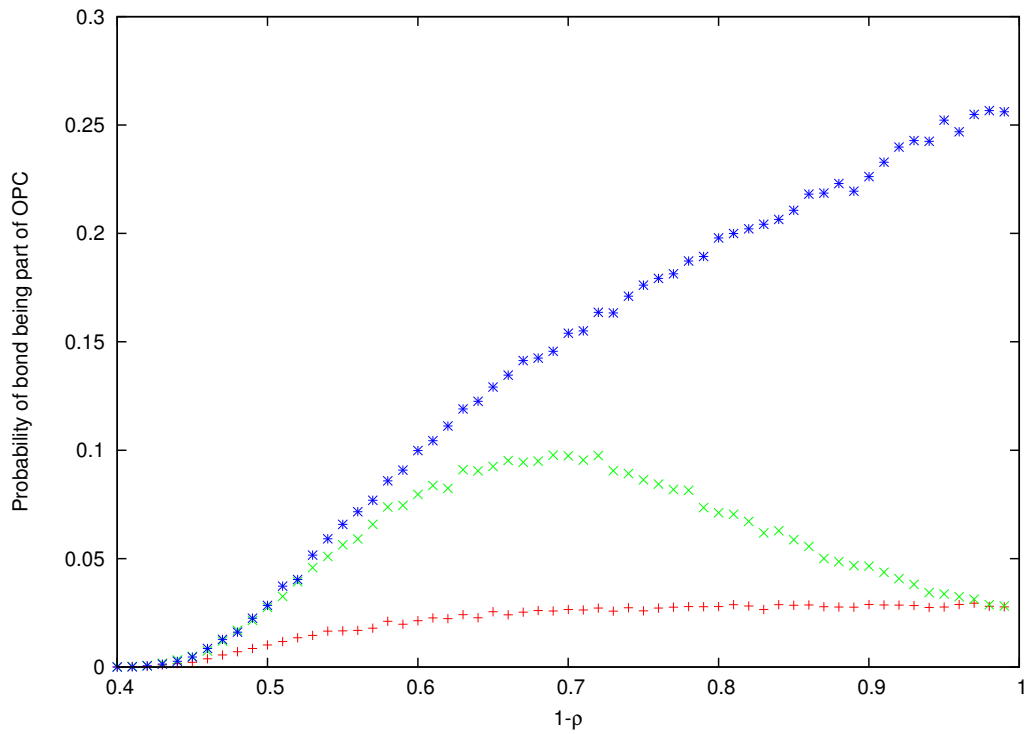


Figure 3.6: Probability of a bond belonging to the OPC backbone (red), and to the entire OPC (including backbone) for $\gamma = 1$ (green) and $\gamma = 10$ (blue), as a function of $1 - \rho = 1 - \epsilon^{-\gamma}$. Each curve is averaged from the the OPCs of 500 random lattices ($L = 40$).

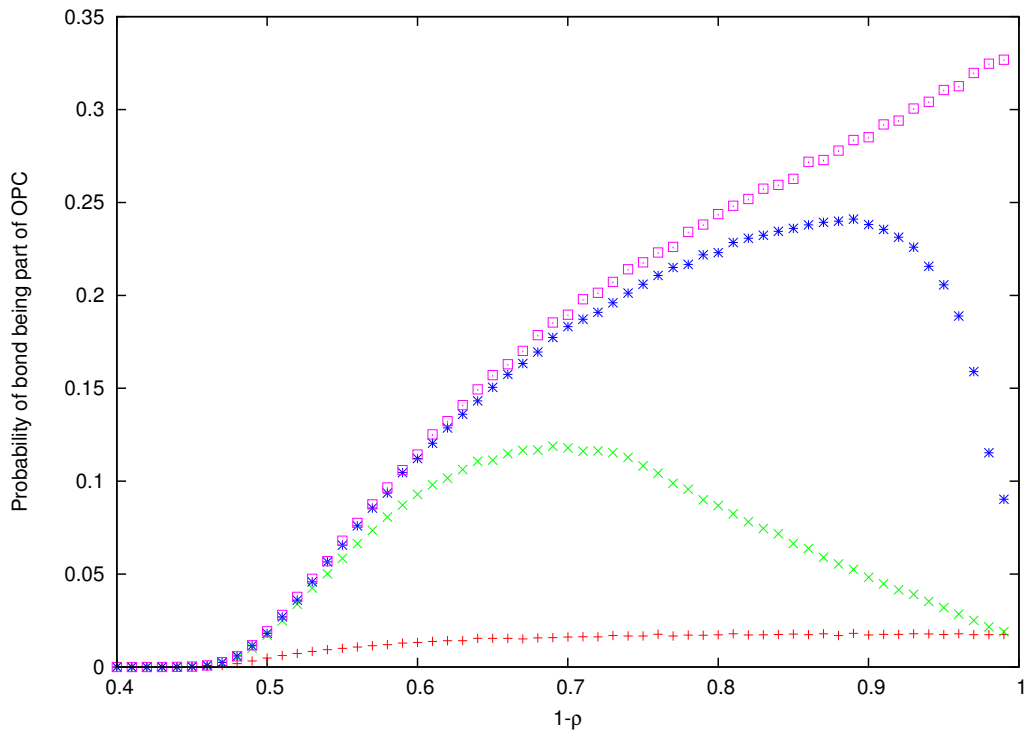


Figure 3.7: Probability P_b of a bond belonging to the OPC backbone (red), and to the entire OPC (including backbone) for $\gamma = 1$ (green), $\gamma = 3$ (blue) and $\gamma = 10$ (purple), as a function of $1 - \rho = 1 - \epsilon^{-\gamma}$. Each curve is averaged from the results of the OPCs of 500 random lattices ($L = 100$).

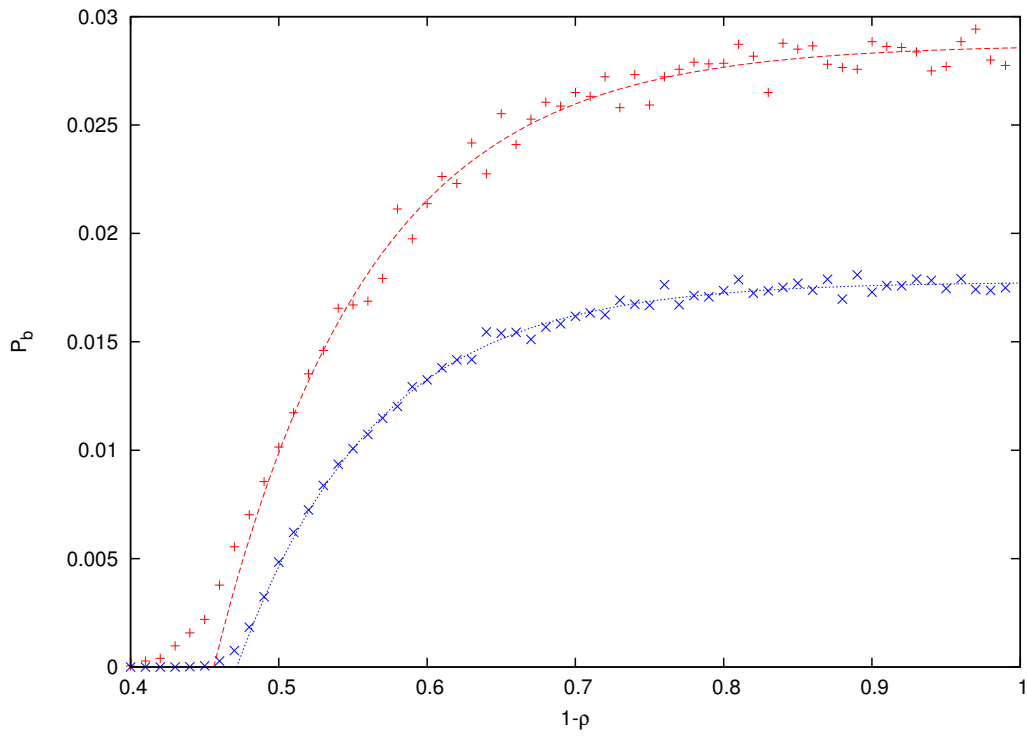


Figure 3.8: Probability of a bond belonging to the OPC backbone, as a function of $1 - \rho$, for $L = 40$ (red) and $L = 100$ (blue). Points correspond to the same sample data shown in figures 3.6 and 3.7, while the lines correspond to a least residual sum of squares fit of equation 3.1 to that data (for $1 - \rho > 0.5$).

3.3 Path mass

Figures 3.9 through 3.11 show how the mass of the optimal path through the lattice evolves as the lattice breaks down. For low disorder, we see a consistent gradual increase in the path mass: naturally, given the tendency of the ϵ_{max} to decrease with increasing E_p , the increase in E must come from increasingly long paths. Additionally, the greater number of removed bonds does make for a more erratic lattice, which naturally eliminates many less massive paths. In the high disorder case, there is no strong dependency between E_p and the mass of the path, especially for less optimal paths. This further emphasizes the importance of the path's highest energy bond: the cost of passing through an increasing number of bonds being relatively inconsequential compared to the benefit of reducing ϵ_{max} : the path will readily pass as many low-energy bonds as necessary, in order to minimize the energy of its maximal bond. For more specific results, we compute the Pearson correlation coefficient between path energies and path mass:

$$r = \frac{\text{cov}(E, M_S)}{\sigma_E \sigma_{M_S}} \quad (3.2)$$

Where $\text{cov}(E, M_S)$ is the sample covariance of E and M_S , while σ_E and σ_{M_S} are the sample standard deviations of E and M_S . The results are shown in table 3.1:

As was conjectured based on figures 3.9-3.11, there is no significant linear

γ	Correlation factor (entire OPC)	Correlation factor (backbone)
0.25	-0.016	0.016
0.5	0.032	0.068
0.75	0.618	0.468
1	0.974	0.552
1.25	0.985	0.942
1.5	0.993	0.984
2	0.996	0.982
10	0.999	0.999

Table 3.1: Pearson correlation coefficients for E and M_S of successive shortest paths found for lattices with different disorder regimes ($L = 200$).

correlation between E and M_S for strong disorder, while the weak disorder cases exhibits very strong correlation (as is to be expected). The most surprising result, however, is that found for $\gamma = 0.75$ $\gamma = 1$: the correlation

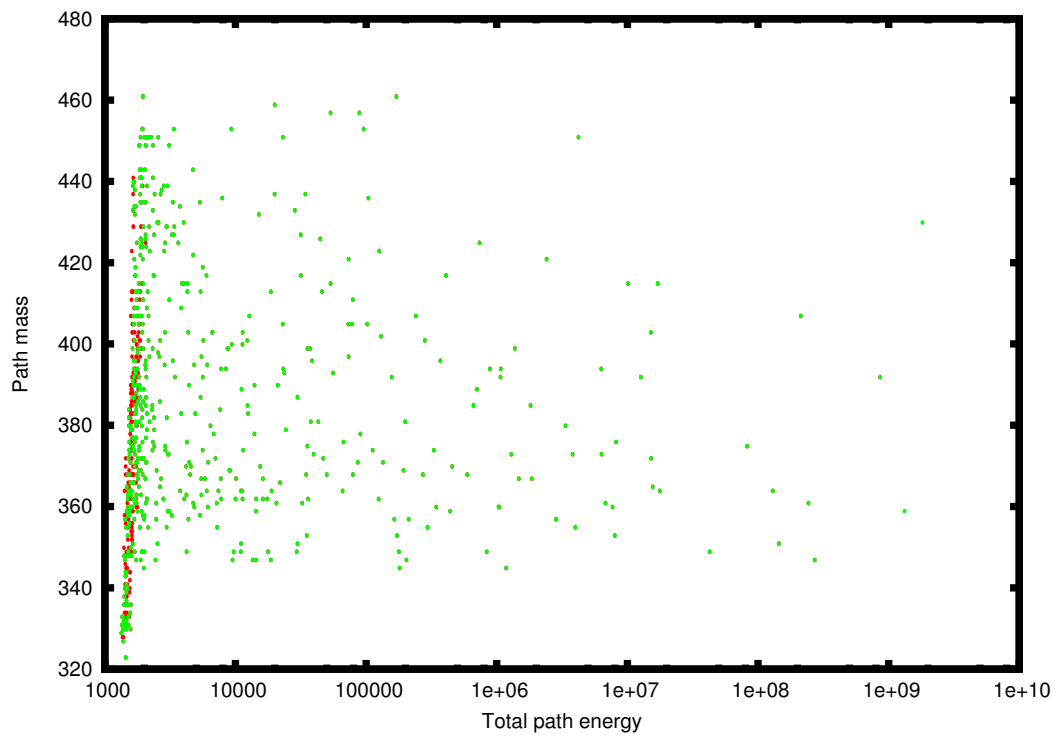


Figure 3.9: *Mass of the successive optimal paths found by the OPC process, plotted against their total cost, on a random $L = 200$ lattice with $\gamma = 0.25$. Green points denote paths contributing to the OPC backbone.*

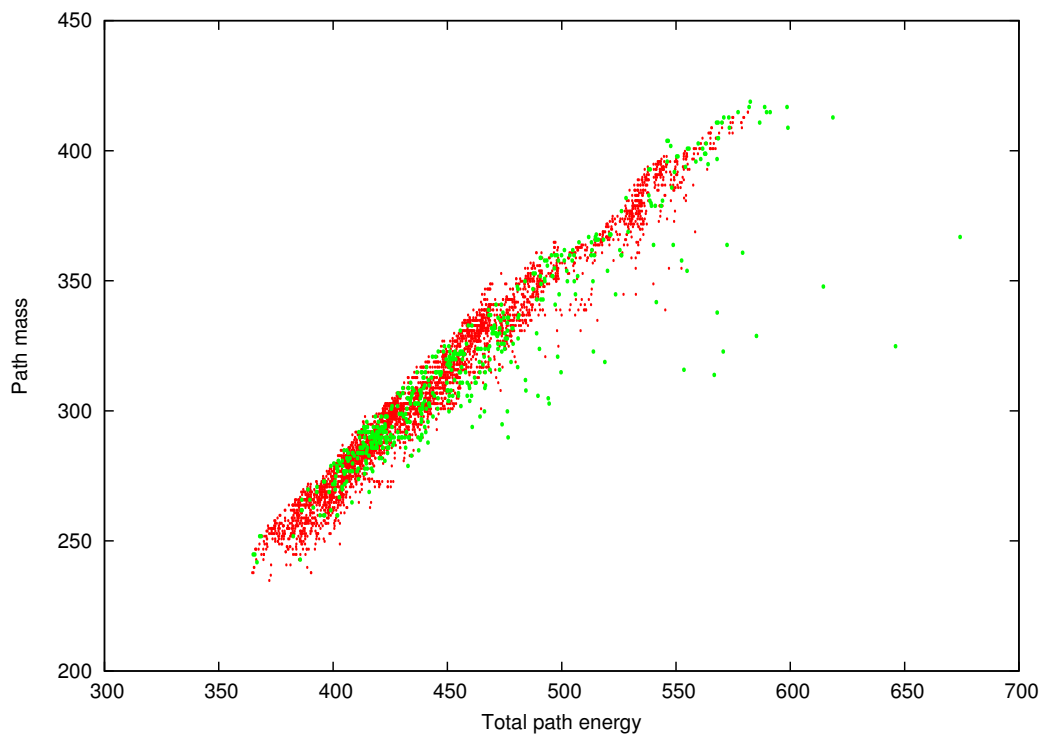


Figure 3.10: *Mass of the successive optimal paths found by the OPC process, plotted against their total cost, on a random $L = 200$ lattice with $\gamma = 1$. Green points (slightly enlarged for visibility) denote paths contributing to the OPC backbone.*

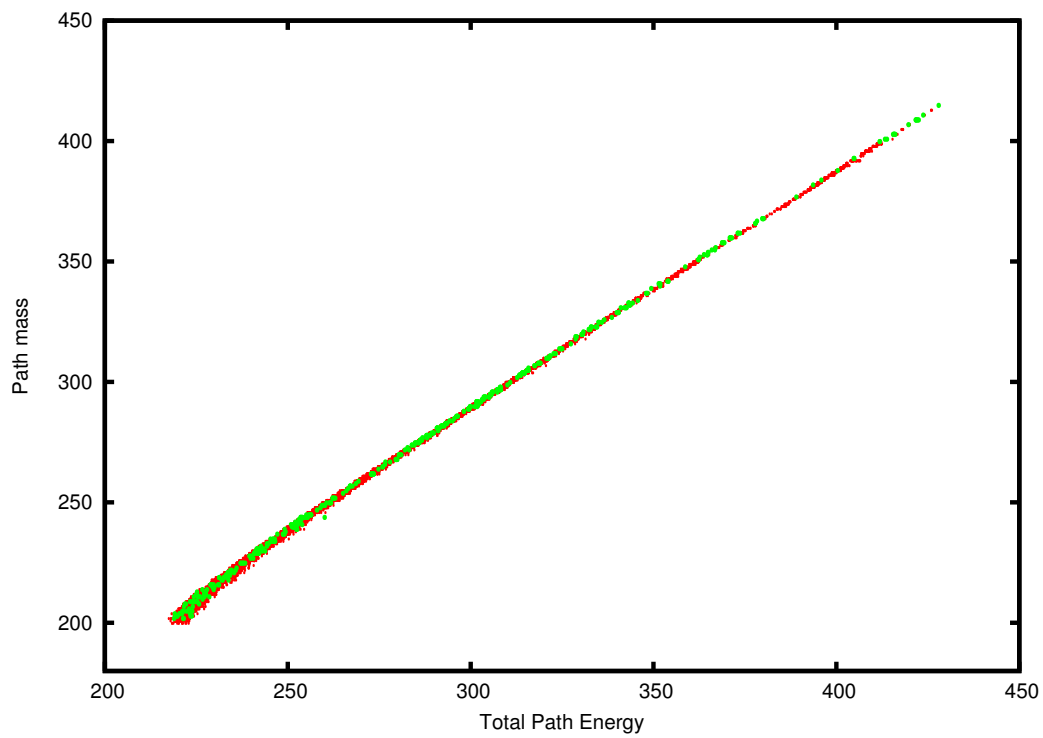


Figure 3.11: *Mass of the successive optimal paths found by the OPC process, plotted against their total cost, on a random $L = 200$ lattice with $\gamma = 10$. Green points (slightly enlarged for visibility) denote paths contributing to the OPC backbone.*

between E and M_S is noticeably weaker for the backbone bonds compared to the OPC as a whole.

3.4 Significance of higher energy bonds

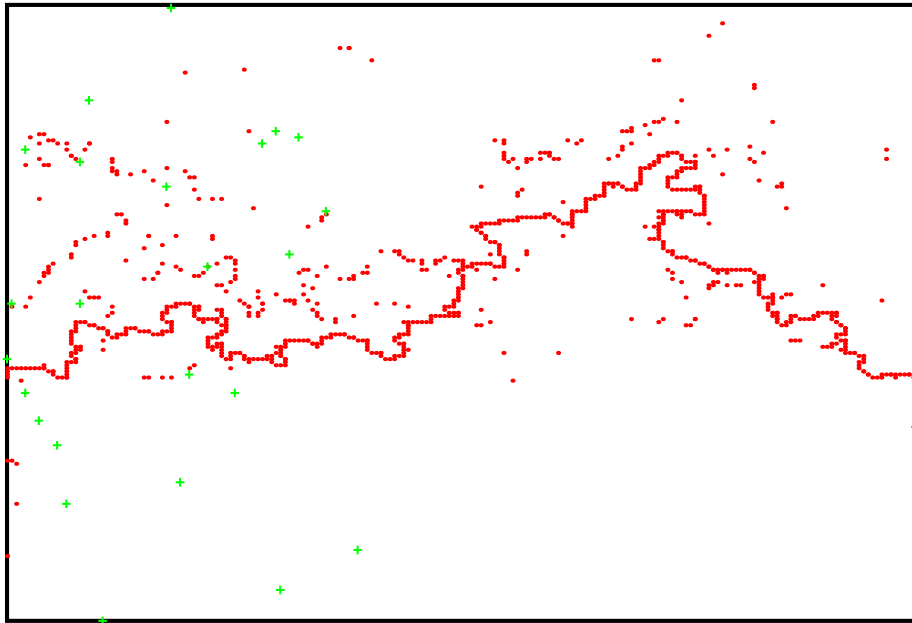


Figure 3.12: *An OPC fracture (red) on a 200 by 200 lattice, with $\gamma = 0.25$. The green crosses denote bonds with higher energies than the costliest bond on the OPC.*

Let us now consider the lattice just as the last bond has been blocked, and the OPC backbone spans from left to right. As shown in figure 3.12, a certain number of bonds exist whose energies are superior to that of the largest blocked bond. Naturally, these cannot have been part of any of the paths found throughout the OPC process. Therefore, if we were to block all of these bonds, open those on the OPC, and repeat the OPC algorithm, we will necessarily still obtain the same crack. If we consider the high disorder case, the energy of each bond removed is superior to that of all bonds previously removed. The bond that is to be removed at a given step is determined by the paths previously found, and these have invariably been composed of lower energy bonds. If we then were to block all bonds with an energy above

a given threshold prior to performing the OPC algorithm on a high disorder lattice, the bonds blocked would necessarily be the same as those blocked by the original OPC (except, naturally, all those bonds on the original OPC above the set threshold).

The low disorder case is not as clear-cut, initially. As shown in figures 3.4 and 3.5, the bonds are not blocked in order of increasing energy value (in fact, there is a general trend towards the opposite for very low disorders). It might therefore be conceivable that blocking certain sites in advance could change the order in which sites are blocked by the OPC process, and could therefore have an effect on the final shape of the OPC. Figures 3.13 and 3.14, however, indicate that this is not the case: all sites on the original lattice's OPC which are not blocked due to exceeding the threshold ultimately end up blocked by the OPC process anyway.

Actually, this is explained by the fact that the OPC process never creates new potential paths: that is, regardless of when a bond is selected to be added to the OPC, the shortest path of which it constitutes the longest bond has always *existed*. This path always existed: naturally, the removal of any bonds in the lattice costlier than the path's costliest could not affect the path. This enables us to conclude that whether any given bond is on the backbone is entirely independent of the characteristics (and order of removal) of any higher energy bonds.

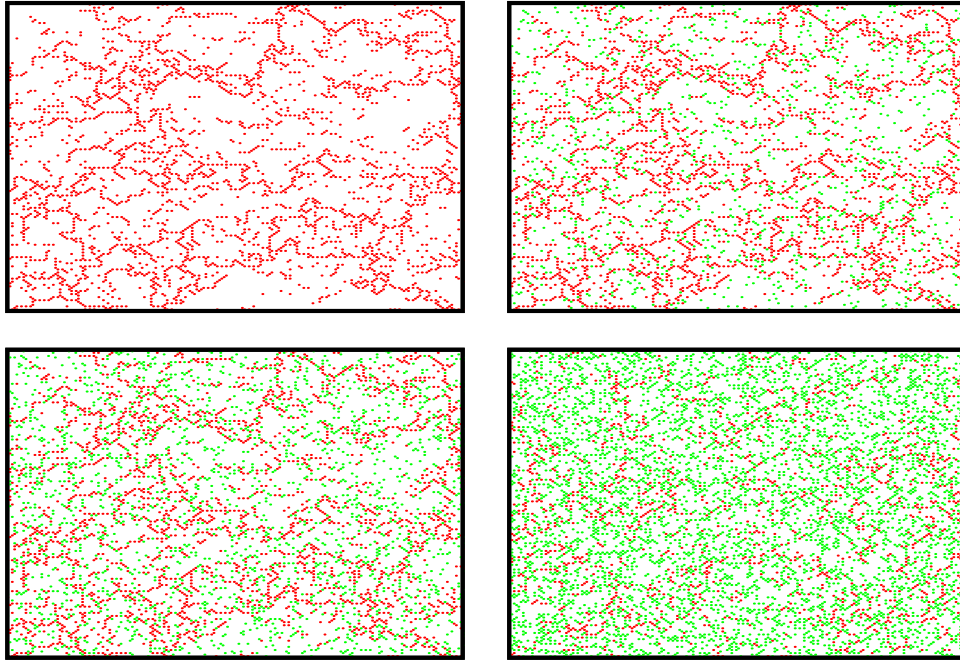


Figure 3.13: *OPC fractures on lattices with higher energy bonds pre-removed on the same 100 by 100 lattice with $\gamma = 2$. The red points correspond to the bonds blocked by the OPC process, green points correspond to bonds above the removal threshold τ . Top left: $\tau = \infty$, top right: $\tau = 4$, bottom left: $\tau = 3$, bottom right: $\tau = 2$.*

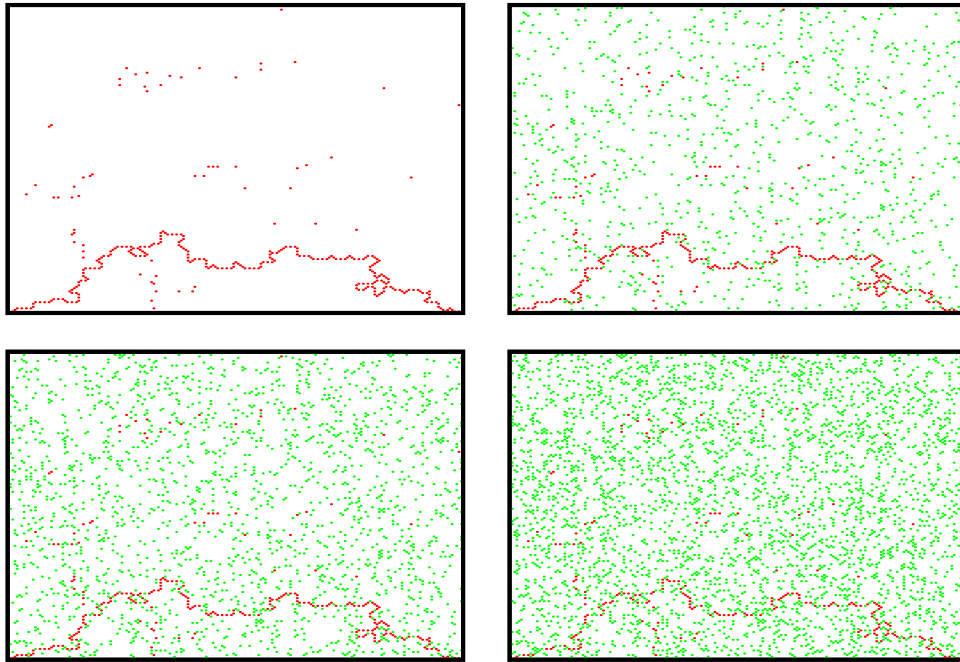


Figure 3.14: *OPC fractures on lattices with higher energy bonds pre-removed on a 100 by 100 lattice with $\gamma = 0.25$. The red points correspond to the bonds blocked by the OPC process, green points correspond to bonds above the removal threshold τ . Top left: $\tau = \infty$, top right: $\tau = 10^5$, bottom left: $\tau = 10^4$, bottom right: $\tau = 10^3$. The random seed of the lattice is the same as the one used in figure 3.13*

Chapter 4

Alternate optimal path problems

While the OPC algorithm was originally formulated with respect to the shortest path problem, it could easily be applied to a variety of other optimal path problems. In principle, any path problem whose solution consists of a singly connected line, and where a sensible criterion to remove a single edge along this solution line exists, can be used to substitute the repeated steps in the OPC algorithm. Two such alternative problems will now be presented.

4.1 Minimax problem

The first problem we are going to be investigating is the minimax problem: the solution path is the path, which, among all possible paths connecting the upper and lower edges of the lattice, has the lowest possible energy value on its costliest edge. A central application for this problem is found in percolation theory [12]. If we then remove this costliest edge on the solution path, we can find a new minimax path, and repeat until no path connects the lattice from top to bottom. We should note that for a given lattice configuration, there is not necessarily a unique solution path: it is perfectly conceivable that from a given solution, a detour could be made close to either edge of the lattice, and that it is possible to reach this end without moving along edges which are longer than the costliest edge of the existing solution path. However, as long as no two edges in the lattice have the same cost, the multiple solutions will all share a common costliest edge, preventing ambiguity as to which edge should be removed from the lattice.

Implementing this process could be done in mainly two ways. We can ei-

ther simply find a path solving the problem on a given lattice, remove its costliest edge, and repeat, as has been done with the shortest path problem. Such a solution path (or rather, the maximum edge along it, which is what we are interested in) can be found by a modified version of algorithms used to find shortest paths. For instance, using the iterative relaxation algorithm for the shortest path problem as previously described for the shortest path problem in Chapter 2, we can find the minimax path by replacing the tentative value $V_t(i_2, j_2) = V(i_1, j_1) + \epsilon(i_1, j_1, i_2, j_2)$ in step 2.(a) with $V_t(i_2, j_2) = \max(V(i_1, j_1), \epsilon(i_1, j_1, i_2, j_2))$, while keeping step 2.(b) unmodified. At steady state, $V(i, j)$ then corresponds to the energy of the costliest bond on minimax path. We will call this the top-down method, as it starts with a full lattice, and gradually removes edges until no path spans the lattice from top to bottom.

The alternate method consists in starting with all edges already blocked, but still assigned a finite energy value and coordinates in the lattice (that is, two sites it connects to). We then open the lowest energy edge, followed by the edge with the second lowest energy, and so on, until a path connecting the upper and lower edges of the lattice is entirely open. Since the edges have been opened by increasing order, the last edge opened must be the costliest edge of all paths enabled by opening it. Further, since opening all edges with a lower energy values than this last one was not enough to create a spanning path, we know that no spanning path going exclusively through lower energy edges is possible. The last edge opened must therefore solve the minimax problem. We then re-block this last edge opened (again disconnecting the lattice), which will remain blocked forever, and proceed to opening the remaining bonds, still in the order of increasing energy value. Each new occurrence of a spanning path corresponds to a new point on the OPC crack formation for the minimax problem. We call this the bottom-up method, as in opposition to the top-down method, it begins with an "empty" lattice (that is, all sites are blocked), and gradually "adds" edges until the lattice is filled (minus the OPC).

4.1.1 Similarity to the shortest path problem

In the shortest path problem with energy values drawn from a power law distribution, the costliest bond of the shortest path gradually accounts for a greater and greater share of the total path cost as the disorder increases. In the infinite disorder limit, the energy of the costliest bond is then nearly

equal to the total path cost: it should follow that the shortest path problem should coincide with the minimax problem for that specific lattice. We already know that in the high disorder regime, the OPC of the shortest path problem is reduced to a singly-connected line corresponding to the backbone of the OPC in the low disorder regime, with certain specific properties.

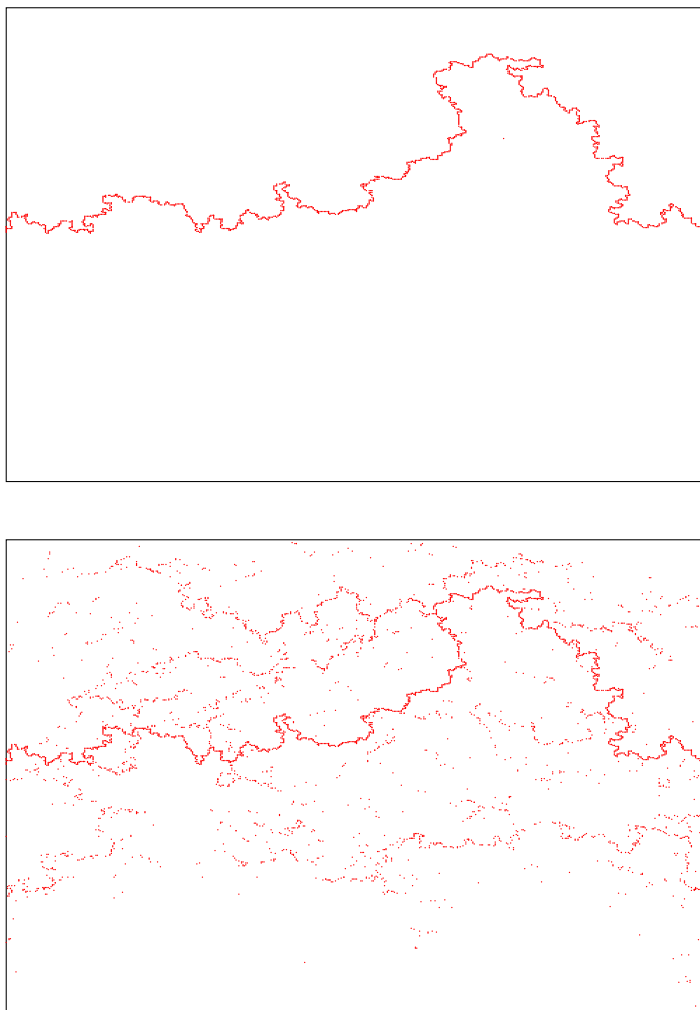


Figure 4.1: *OPC of the minimax problem (top) compared to the shortest path OPC (bottom). Both processes are run on the same lattice ($L = 500$). In order to permit a readily discernible backbone, the disorder parameter has been set to $\gamma = 0.25$, so as to place us in the low disorder regime.*

As is shown in figure 4.1, the OPC of the minimax problem does coincide with the backbone of a the OPC formation in a medium/low disorder lattice. It should be noted that the minimax problem has been solved on *exactly* the same lattice as the shortest path problem; that is, with the same disorder. In fact, as shown in figure 4.2, the minimax OPC does not appear to exhibit any dependence whatsoever on the disorder parameter: in all cases it corresponds

to the simply connected backbone obtained by the shortest path OPC in the high disorder regime.

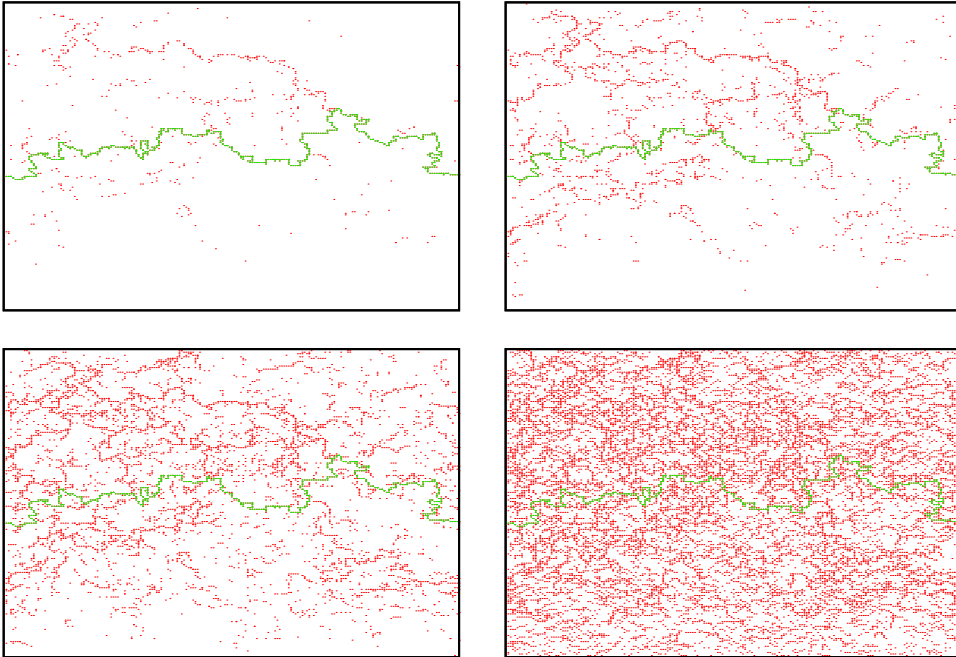


Figure 4.2: *OPC of the minimax problem (green) for four lattices, generated from the same seed with various degrees of disorder. Top left: $\gamma = 0.25$, top right: $\gamma = 0.5$, bottom left: $\gamma = 1$, bottom right $\gamma = 5$. The bonds of the shortest OPC not part of the minimax OPC have been added for reference. The result is strikingly similar to figure 3.1.*

The explanation for the disorder-independence of the minimax OPC is fairly straightforward. If we look at the second method (bottom-up) of generating a minimax OPC, we notice that as we gradually open edges with increasing energy values, the exact value is irrelevant to the process; the formation of the OPC is defined purely by the order in which the bonds are opened. For any two bonds compared, the one with the higher energy value will still be of a higher value if the disorder is increased. Then, as the disorder has no effect on the sequence in which bonds are opened, neither does it effect the ultimate result of the bottom-up algorithm, which is the minimax OPC.

Further, we note that for lattice whose bond energies are derived from the

same seed, changing the disorder parameter γ from a value γ_{old} to a new value γ_{new} corresponds to performing the operation $\epsilon_{new} = \epsilon_{old}^{\frac{\gamma_{old}}{\gamma_{new}}}$ on each bond. For any bond of energy $\epsilon = s^{1/\gamma}$ and (finite) path of energy $E = \sum_i \epsilon_i$, where $\epsilon_i = s_i^{1/\gamma}$ is the energy of the i -th bond constituting that path, if $\epsilon > \epsilon_i$ (which also implies $s > s_i$) for all i , we have, since s and s_i are independent of γ :

$$\lim_{\gamma \rightarrow 0} \frac{\epsilon - E}{\epsilon} = \lim_{\gamma \rightarrow 0} \left(1 - \sum_i \frac{\epsilon_i}{\epsilon}\right) = \lim_{\gamma \rightarrow 0} \left[1 - \sum_i \left(\frac{s_i}{s}\right)^{\frac{1}{\gamma}}\right] = 1 \quad (4.1)$$

Since $\lim_{\gamma \rightarrow 0} \frac{\epsilon - E}{\epsilon}$ is positive, we know that for any bond, there is a *finite* value of γ for which the total energy of any given path passing exclusively through lower-energy bonds is lower than the energy of that bond (and, as all bond energies are positive, all paths passing through it). Therefore, for any finite lattice, a finite value of γ exists where the shortest path and minimax OPC processes are entirely equivalent. As the previous reasoning also holds true if ϵ and ϵ_i denote site energies instead of bonds, the site-oriented shortest path OPC backbone (which is independent of disorder) and minimax OPC must also coincide entirely with each other for any finite lattice.

4.1.2 Ultrametric limit and the exact shortest path

Even though the minimax process described coincides with the shortest path OPC for high disorder, the term "optimal path" is somewhat problematic in the minimax case. As previously mentioned, the processes described here only find a bond to be blocked, no *path*. It is entirely possible for two different paths to share a common costliest bond, and this possibility of a non-unique solution of the minimax problem means that it would be wrong to equate the minimax and shortest path problems themselves in the high disorder case; what we can equate is the results of the OPC method applied to them. However, a useful construct to "merge" these problems is what we call the ultrametric limit. In [2], it was defined as a very highly disordered lattice where the optimality of any given path was determined by its costliest bond.

We will use a slightly stricter and more precise definition: a lattice is considered to be in the ultrametric limit if and only if the energy value of each bond is greater than the sum of the energies of all lesser bonds. The shortest path in this case is found by first finding all paths solving the minimax problem: the shortest path is the one whose second costliest bond has the lowest energy value. If any of the paths share the second costliest bond, we rank

these by their third costliest bond, and so on, until a unique path remains. Interestingly, this path is also a self-similar fractal [2], with the same fractal dimension of $D = 1.22$ as the OPC backbone. This definition of the ultrametric shortest path is, in fact, of great relevance to many of the processes discussed in this thesis, and we will revisit it in further detail in the following chapters.

4.2 Widest path problem

Another common optimal path problem is that of the "widest path". In this problem, the energy value assigned to each bond is not viewed as a cost, but rather as the maximal bandwidth available between the two sites. The problem then consists in finding the path with the highest total bandwidth; this is defined as the bandwidth of its bottleneck, which is the bond along that path with the lowest bandwidth. It can in fact be thought of as the opposite of the minimax problem, and can be solved analogously, by modifying the relaxation algorithm from Chapter 2 with $V_t(i_2, j_2) = \min[V(i_1, j_1), \epsilon(i_1, j_1, i_2, j_2)]$ in step 2.(a), and setting $V(i_2, j_2) = \max[V(i_2, j_2), V_t(i_2, j_2)]$ in step 2.(b). The steady-state value of $V(i, j)$ is then the energy of the lowest energy bond on all possible paths connecting the site (i, j) to the source sites.

Alternately, the bottom-up method of solving the minimax problem can also be easily adapted to the widest path by opening bonds in the reverse order; that is, beginning with the highest energy bond, followed by the second-highest energy bond, and so on. Just as with the minimax bottom-up method, whenever the horizontal edges of the lattice are connected, the last bond opened must correspond to the cheapest bond on its path, while no path passing exclusively through costlier bonds exists (barring, of course, those already permanently blocked by the OPC process). This bond must then be the bottleneck of the widest path, and is removed from the lattice (which in this case corresponds to setting $\epsilon = 0$, not $\epsilon = \infty$).

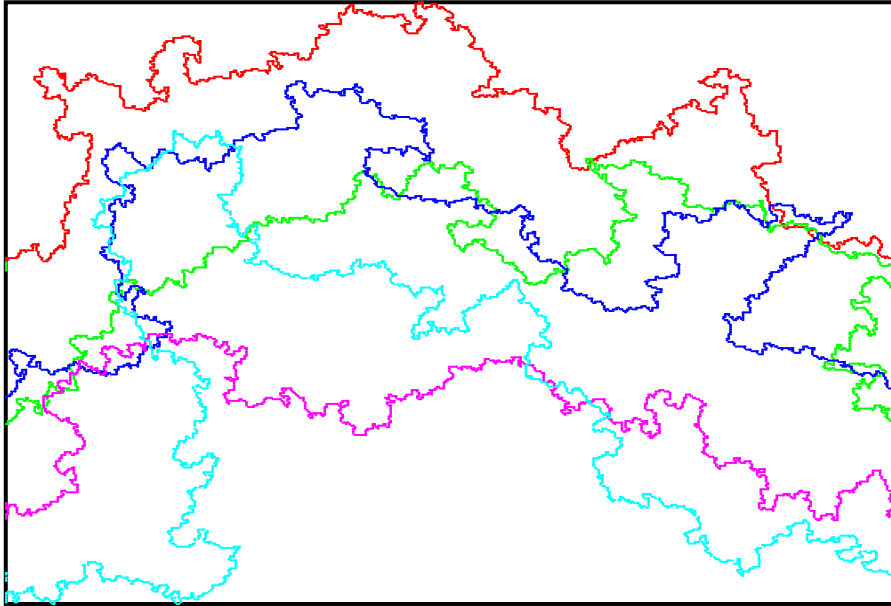


Figure 4.3: *A sample of five widest path OPCs. Each OPC has been generated from a different lattice with $L = 500$. Changing the disorder parameter has no impact on the widest path OPC.*

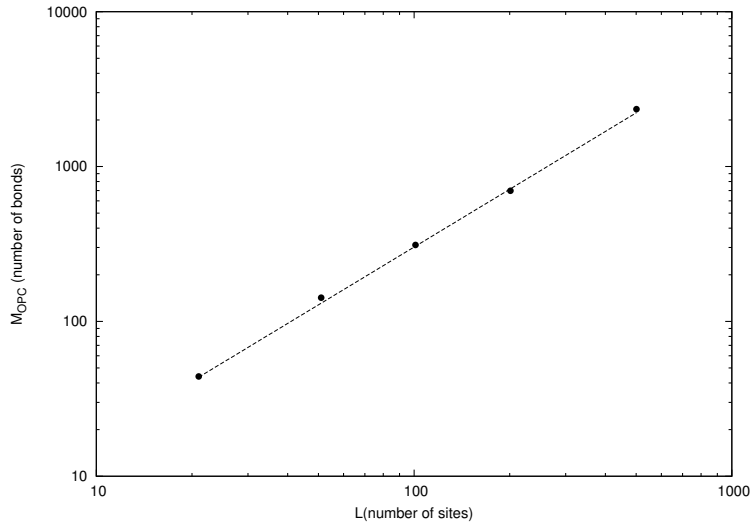


Figure 4.4: Mass M of the widest path OPC (in number of blocked bonds) as a function of lattice size L . Each data point corresponds to the average of 50 samples, and the error bars are approximately the same size as the data points. The slope of the dashed line is $D \approx 1.24$.

Figures 4.3 and 4.4 show a clear similarity to the results of the minimax problem: we obtain singly connected fault lines regardless of the degree of disorder, whose mass scales as $M_{OPC} = L^D$, with $D = 1.24 \pm 0.02$. It should be kept in mind, however, that while the minimax OPC and the high disorder shortest path OPC exhibit similar characteristics due to the resultant crack coinciding, the widest path OPC for a given lattice is (naturally) entirely different from its minimax OPC, as we can see from figure 4.5.

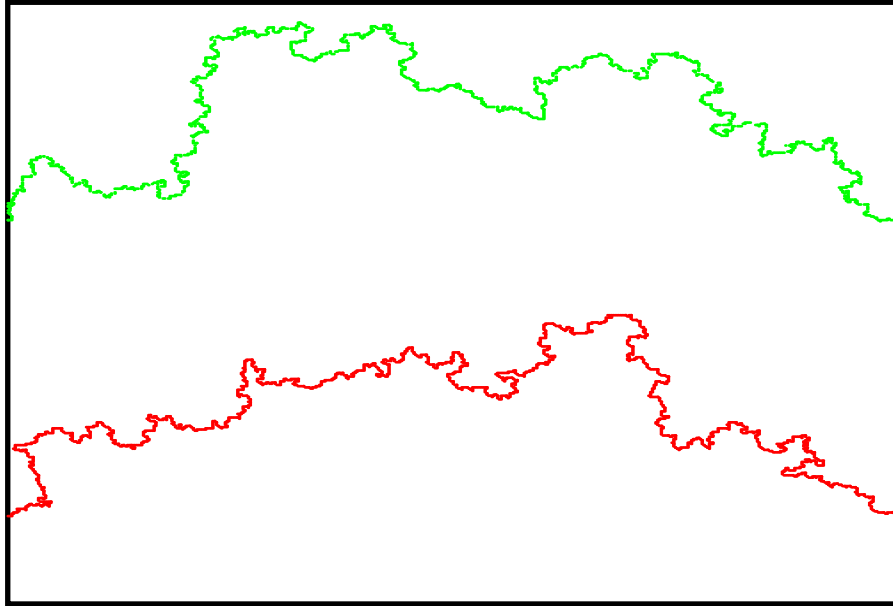


Figure 4.5: *Superposition of the minimax (red) and widest path (green) OPCs for a random lattice ($L = 500$).*

The justification for this similarity is based on the fact that for both of these problems, the random generation of the edge weights for the lattice only really serves to determine the order in which edges are opened by the "bottom-up" algorithms (since the magnitude of the difference between edge weights is of no importance, as discussed previously). For any given lattice A , we can easily imagine a "mirrored" lattice B whose edge weights are reversed: that is, the highest-energy edge of lattice A has the same coordinates as the lowest (non-zero) energy edge of B and vice-versa, the second highest-energy edge of A has the same coordinates and orientation as the second lowest-energy edge of B , and so on. Performing the bottom-up widest path OPC method on such a mirrored lattice removes bonds in the exact same order as the minimax OPC on the original one (and the connectivity check does not take the bond energies into account, only placement), and would therefore yield the same OPC. Since the energy of each bond is drawn at random from the same distribution, and independently of the energy of other bonds, we have, for any two bonds (i_a, j_a) and (i_b, j_b) with respective energies ϵ_a and ϵ_b :

$$P[(\epsilon_a = x_1) \cup (\epsilon_b = x_2)] = P(\epsilon_a = x_1) \cdot P(\epsilon_b = x_2) \quad (4.2)$$

$$= P(\epsilon_b = x_1) \cdot P(\epsilon_a = x_2) \quad (4.3)$$

$$= P[(\epsilon_a = x_2) \cup (\epsilon_b = x_1)] \quad (4.4)$$

This can be generalized to any arbitrarily large combination of energy values, from which it follows the probability of generating a given lattice must be equal to the probability of creating the "reversed" one. Therefore, for a lattice with unknown bond energies, the probability of obtaining a given OPC crack should be the same whether the minimax or widest path process is used. As a consequence, the average characteristics of the OPCs over a large sample of lattices must naturally coincide.

Chapter 5

Invasion percolation, minimum spanning trees and the ultrametric shortest path

As mentioned in the introduction, the fractal dimension $D \approx 1.22$ has been shown to appear in a variety of phenomena in addition to those already presented [2] [6] [7] [9]. In the previous chapters, a central characteristic of the obtained fractals have been that they span a quadratic lattice horizontally, dividing the system in two; the fractals essentially block a hypothetical vertical flow. We now consider the other kind of process, where the fractal appears as part of a "movement" or growth process through a lattice. We will look at two such processes: minimum spanning trees and invasion percolation. We will then show that both these problems are fundamentally connected to that of the ultrametric shortest path.

5.1 Minimum spanning trees

In graph theory, a minimum spanning tree (MST) can be defined as the set of edges (our bonds) and nodes (sites) such that the edges connect all the nodes, while the sum of the weight (energy cost) of all edges is minimal (for consistency, we will retain the terms site and bond for what is commonly referred to respectively as nodes/vertices and edges in graph theory). One central aspect of such trees is that for each pair of sites, a single unique path exists in each tree, connecting the pair of sites. It is found that these paths scale with lattice size with the same exponent $D \approx 1.22$ as the OPC backbone. As with the minimax and widest path problems, it can be readily shown that the formation of the minimum spanning tree is defined only by

the relative energies of the bonds [6] (as should be apparent from the solving algorithm described shortly), and the tree is therefore unaffected by changes in disorder; this enables us to work with the assumption that each MST is found on a lattice which is in the ultrametric limit.

A common method of finding an MST is known as Prim's algorithm[5], which consists of the following steps:

1. Begin with an empty tree.
2. Select an arbitrary site, and assign it to the tree.
3. Find the cheapest bond which connects a site in the tree to one not in it. Add the bond and the latter site to the tree.
4. If any sites in the lattice are not part of the tree, return to step 3.
5. Once all sites are part of the tree, this tree constitutes the MST for the lattice.

It should be noted that MSTs are a construct whose definition is based on an undirected graph, with weight assigned to the edges. As a result, we cannot directly apply the notion to a site-oriented lattice: in graph theory, such a lattice would need to be represented by a *directed* graph, with two opposing directed edges connecting each pair of adjacent vertices; the cost of each edge would be equal to the energy value of the site it points towards.

5.2 Strands on invasion percolation

Another process which exhibits similarity to the OPC fractal is invasion percolation (IP) [7]. In invasion percolation, we begin by imagining fluid at a source site, which, at each new step, spreads to the site which is the easiest to reach: this can be determined either by the cost of crossing a bond (bond invasion percolation), or reaching a given site (site invasion percolation); these problems work in lattices which correspond to our bond-oriented and site-oriented OPC problems, respectively. Cieplak *et al*[7] found that the length of strands in such a bond IP process (where a strand is the path which leads us from a given site to the source) scale as $L^{1.22}$, where L is the straight-line distance between sites, just as was found for minimum spanning trees. Actually, we note that in order to ensure the strand from a given site was unique (as ordinary non-trapping invasion percolation does not in any way prevent loops), the fluid was forbidden from percolating between two invaded sites.

With this minor change to bond invasion percolation, the process does in fact become exactly identical to Prim's algorithm as described previously, with the source site being the arbitrarily chosen site and the tree corresponding to the set of invaded sites and the bonds constituting the strands. The only difference between the processes is the ending condition: while IP stops once a target site is invaded, Prim's algorithm persists until all sites are part of the MST. Hence, all IP strands are also paths on the minimum spanning tree.

Staying with bond invasion percolation, let us now consider the case where we *do* allow the fluid to invade bonds which connect two invaded sites, while still identifying unique paths between each pair of sites, based on a "preferred" invasion route. This sort of IP can then be simulated with the following steps:

1. Mark a source site (or several) as "invaded". Mark all bonds as "closed". Mark one or several sites as "targets".
2. Find the lowest energy closed bond, and mark it as "open".
3. If the bond opened in 2 is connected to an invaded site, mark the bond as invaded as well, and execute the following steps:
 - (a) If one of the sites connected to this bond is invaded and the other is not, invade the latter. Mark the bond as a "predecessor" (which can be considered as either a special case of "open" or an entirely different switch). Similarly, the previously invaded site is marked as the newly invaded site's predecessor.
 - (b) If a new site is invaded by the previous step, check if any of its neighbouring bonds is open and non-invaded.
 - (c) If a bond neighbouring any site invaded after the last iteration of step 2 is open, invade it, and apply steps a) through c) to that bond (this step can lead to a nesting loop, which ends once either all open sites are invaded, or the invaded region no longer borders any open non-invaded bonds). If several such bonds are open, apply the steps to the cheapest one.
4. If no target sites are invaded, return to step 2. Else, terminate.

If the sites on the lower edge of the lattice are designated as sources and the ones on the upper edge designated as targets, the mass of opened bonds at the completion of the process equals the set of bonds opened by the bottom-down approach to the minimax OPC problem just as we find the first edge

of the said OPC, as the IP algorithm stops as the first path connecting the horizontal edges is found.

We now recall the case of the shortest path problem in very high disorder lattices, the ultrametric limit. Like percolation strands and spanning trees, the resultant figure connects sites (while the OPCs prevent such connections), and all these appear to scale in the same way. We will now show that the ultrametric shortest path between two points is in fact the same figure as the percolation strand and MST path linking these two points.

First, we should make clear the role of the "predecessor" bonds as marked by the aforementioned bond IP process. As the next site invaded at any given time is the non-invaded one connected to the set of invaded sites by the cheapest bond, the invasion of non-predecessor bonds (which does not lead to the invasion of a site) has no bearing on the order in which new sites are invaded or that of the bonds through which they are. As a result, the sites are invaded in the same order as they are added to the tree by Prim's algorithm; the predecessor bonds are also invaded in the same order as Prim's algorithm adds them to the MST. As a result, when the IP algorithm has reached a target site, the set of predecessor bonds corresponds to the subset of the minimum spanning tree which connects all the invaded sites. Therefore, there exists a unique path along predecessor bonds from each invaded target site to each source site connected to the spanning invaded cluster, which we will denote as the "predecessor path". Since several predecessor paths may be adjacent to a given site, assigning each node a predecessor node (which was invaded previously, and is therefore closer to the source) allows us to quickly find the path connecting a given site to the source sites.

We now look at the last bond opened (which is not necessarily the last bond *invaded*) by the IP algorithm. Since opening all lesser bonds was not enough to allow invasion of the target sites, it must necessarily be part of any path connecting the sources to the invaded targets. Since it is also the costliest of all bonds opened, it must also be the costliest bond on all such paths. The last bond opened must therefore be the costliest bond on all paths solving the minimax problem, and consequently, also belong to the ultrametric shortest path. In fact, since all bonds connected to source sites through open bonds are invaded, the set of invaded bonds must contain any and all paths solving the minimax problem. Therefore, the ultrametric shortest path can only contain invaded bonds.

If only one invaded path connects the source and target sites, then this

is naturally the shortest path; and since a unique predecessor path exists between any two invaded sites, the shortest path and predecessor path must coincide. If several distinct paths connect source and target sites, this is the result of the invaded bonds forming cycles (remember that the bonds on the horizontal edges have $\epsilon = 0$, and therefore automatically belong to the MST). We then find the highest energy bond part of such a cycle. As the sites on either end of that bond are connected through a set of bonds, all of which were opened prior to the cycle's highest energy bond, the first of those sites invaded would have been able to invade the other one through that set of bonds prior to invading through the highest-energy bond: as a result, the highest-energy bond of a cycle cannot be a predecessor bond. Additionally, as we are in the ultrametric limit, the highest-energy bond of a cycle cannot be part of a shortest path: even for connecting the sites on the ends of the bond, the path going through the other bonds of said cycle would still be cheaper.

As a result, the highest-energy bond on that cycle can be safely blocked, without affecting either the shortest path or the predecessor path (as it belongs to neither). We then proceed to check for further cycles, remove the costliest bond belonging to any of those cycles, and repeat until no such cycles remain. Ultimately, only one path from a target site to a source site remains. Since both the shortest path and the predecessor path must have remained unchanged by the breaking of the cycles, and must also be able to connect the sources and targets (a condition we know only one remaining path fulfills), these two paths must also necessarily be one and the same. Since the predecessor path belongs to the MST (on which there exists only one path connecting any two sites), we can conclude that for a given pair of sites on a given lattice, the ultrametric shortest path between these two, the MST path between them and the percolation strand from either with the other as a source are all the same object, which naturally explains why the scaling exponent was found to be identical for all three processes.

Chapter 6

Common universality of OPCs and the ultrametric shortest path

Let us now return to the concept of the ultrametric shortest path. We already know that in the ultrametric limit, the shortest path is one of the (possibly) multiple solution paths to the minimax problem. Although it can ostensibly be found by the same means as any other shortest path, the required disorder (and resulting range of energy values) often makes it more feasible to solve it by the repeated tiebreaking of minimax paths [2]. As mentioned previously, the costliest edge of all minimax paths can be found by starting with a fully blocked lattice, and opening bonds by increasing energy value until a connection exists between a target site and a source site, or equivalently starting with a fully open lattice and blocking bonds until the target and source sites are no longer connected, the former method constituting a step of what we termed the "bottom-up" method for finding the minimax OPC. In both cases, the last bond removed is the lowest energy bond through which a path exists which only traverses equal or cheaper bonds, i.e. the maximal energy bond on the ultrametric shortest/minimax path. If we use the latter method, we can then find the second highest energy bond by keeping the maximal energy bond open and continue blocking bonds in order of decreasing energy until the target and source nodes are again disconnected. Again, we re-open the bond whose removal breaks the lattice, and repeat. The entire shortest path is found when the points found in this manner constitute a path connecting a source site to a target.

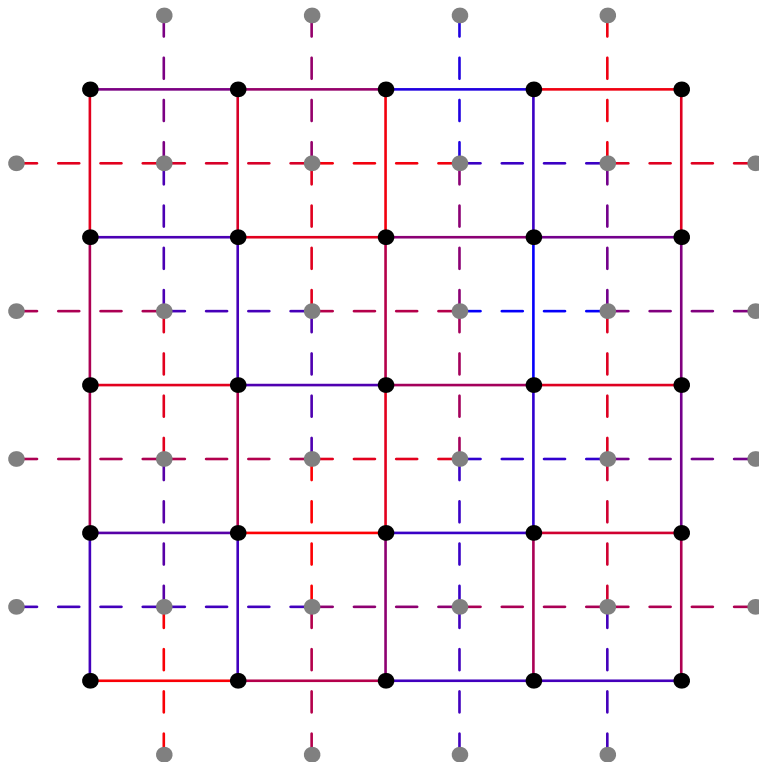


Figure 6.1: *Representation of an original lattice (sites are black dots, bonds are solid lines) and its associated dual lattice (sites are grey dots and bonds dashed lines), with colors representing energy values (red is higher, blue is lower). For each bond on the original lattice, the associated bond is represented by the stapled line intersecting it through its center; note how intersecting lines have the same energy value.*

We now introduce the "dual" lattice, which we will define in relation to an "original" lattice (which is a regular quadratic bond-oriented lattice as we have been studying throughout this thesis). The dual lattice contains the same number of bonds as the original lattice; in fact, each bond in the the original lattice will be associated to a single bond in the dual lattice, which will be assigned the same energy value. These bonds are then connected on both ends to sites (which do not have equivalents on the original lattice), which are connected to either 0 or 3 other bonds. These connections are made such that if the coordinates of a bond on the primal lattice are $(i, j, i + 1, j)$, its associated edge shares one site with bonds associated $(i, j, i, j + 1)$, $(i + 1, j, i + 1, j + 1)$ and $(i, j + 1, i + 1, j + 1)$, and the other site with bonds associated $(i, j, i, j - 1)$, $(i, j, i + 1, -1)$, $(i, j - 1, i + 1, j - 1)$,

lattice to its top, using the method explained previously in this chapter. The bonds in the dual lattice are set to be closed whenever their associated bond is open (which they all are at the beginning of the process). As we gradually open the original lattice bonds in order of decreasing energy, their associated bonds are closed accordingly. As the existence of a spanning path in the dual lattice is a necessary and sufficient condition for the associated bonds to disconnect the original lattice, a new path is found connecting the left and right edges of the dual lattice for each bond of the shortest path found in the original lattice. Therefore, keeping open the bonds whose removal disconnect the original lattice also means we keep closed those bonds who permit new horizontal paths in the dual. The bonds of the original lattice are removed by descending order of energy: the dual lattice bonds are also added in the same order. As a result, we are effectively subjecting the dual lattice to the bottom-up solution algorithm for the widest (horizontal) path OPC.

Therefore, finding the shortest path on a regular quadratic lattice (with fixed boundary conditions in all directions) in the vertical direction corresponds to finding the OPC of the horizontal widest paths. Since a spanning path in the original lattice also disconnects the dual lattice, the inverse must also hold true: finding the OPC of horizontal widest paths in a regular quadratic lattice corresponds to finding the ultrametric shortest vertical path of its dual.

On a lattice with periodic boundary conditions on the vertical edges, the situation becomes a little more complex. In order to split the lattice, the OPC must form a loop. Therefore, it is not just the ultrametric shortest left-to-right path on the dual lattice that forms the OPC backbone: it must be the shortest one which starts and ends at the same y-coordinate. Additionally, the periodic dual lattice would permit a path "taking a shortcut" to the rightmost sites by simply crossing over the left edge; such paths obviously do not disconnect the lattice and must therefore be discarded.

Also, it should be noted that while each OPC backbone constitutes an ultrametric shortest path, the average size of the OPC backbone may deviate from that of shortest paths, for the simple reason that the OPC backbone is the shortest of all the shortest paths between all pairs of source and target sites. However, as we saw in figures 3.9 and table 3.1 the correlation between total path energy and path mass is weak in highly disordered lattices: hence, selecting lower energy paths does not cause us to select paths with lower *mass*.

6.1 Computational implications

If we have fixed boundary conditions on all edges, we can use the fact the equivalence of the widest path OPC and the ultrametric shortest path in order to quickly find the backbone of the shortest path OPC. The intuitive approach is to sort the bonds by energy value and create the "mirrored" lattice, whose widest path OPC corresponds to the minimax OPC we are trying to find. We then increase the disorder enough to place us in the ultrametric limit. A single run of a quick shortest path algorithm such as Dijkstra's or the one proposed by Hansen and Kertész could then find the widest path OPC on the reversed lattice, and consequently, the backbone of the shortest path OPC on the initial one. However, as we mentioned earlier, the necessary amount of disorder in order to reach the ultrametric limit will generally lead to problematically large energy values.

Instead, we can use the invasion percolation algorithm described in Chapter 5. We have already shown that the predecessor path leading to the first invaded target site also corresponds to the ultrametric shortest spanning path, and will now show that this can be used to find the ultrametric shortest path in $O(L^2 \log L)$ time:

- First, we sort the lattice bonds by energy, and, if we want to find the minimax OPC or shortest path OPC backbone, mirror the lattice. With L^2 bonds, this can be done in $O(L^2 \log L^2) = O(L^2 \log L)$. [5]
- Second, we need to invade at most L^2 bonds before reaching the target sites.
 - Finding the cheapest closed bond and invading it is done in constant time.
 - Checking the site on the other end of the last invaded bond, invading it if necessary, and checking it for open neighbouring bonds is done in constant time.
 - If store the list of open bonds adjacent to invaded sites as a binomial heap (which will contain no more than L^2 elements), we can find the cheapest of these and invade it in $O(\log L^2) = O(\log L)$ [5].
- Hence, each bond is invaded in *at most* $O(\log L)$. As we need to invade at most $4L^2$ bonds, the target sites are invaded in $O(L^2 \log L)$ (as a worst case scenario).

- From any target site, following the predecessor sites to a source site is done in no more than $O(L^2)$.
- The worst case total run time is then $O(L^2 \log L + L^2 \log L + L^2) = O(L^2 \log L)$

Note that the dual lattice does not originally have bonds on its vertical edges: we therefore add those bonds (with $\epsilon = 0$), which causes the subset of the MST found by the IP algorithm to connect any pair of source/target sites using only zero-energy bonds and the single shortest of all source-to-target paths. This enables us to find the ultrametric shortest path by backtracking through predecessor sites from an arbitrary target site. This allows us to avoid comparing distinct shortest paths between each source/target pair (for which we have no immediate means, as the IP algorithm takes no account of path lengths).

In any case, this is a considerable improvement compared to the run time of Dijkstra's algorithm with Fibonacci heaps at $O(L^2 \log L)$ [14] or the (unrealistic) best case run time of the Hansen-Kertész algorithm at $O(L^2)$, as they must each be run *at least* L times in order to find a spanning fault line.

Chapter 7

Conclusion

7.1 Properties of the bonds of the shortest path OPC

Our initial attempts to find path characteristics which would distinguish the backbone bonds from the others did not yield any clear-cut characteristic which would enable us to readily separate the backbone (and thus explain their apparent invariance to disorder). In each disorder regime, the paths whose costliest bond is a part of the backbone do not universally distinguish themselves from the others by the energy of their costliest bond or the path mass.

However, in general terms, the energy of the backbone bonds does follow a different distribution than the non-backbone energies. For those paths not contributing to the backbone, the interdependence between the distribution of costliest bond energies and system disorder appears to be fairly complex: as the disorder increases, the increase in mass of the OPC draws mostly from moderate-energy bonds; however, even further increases in disorder lead to the OPC again drawing from higher-energy bonds.

The most practically applicable result from our study of the OPC bond energies is probably the fact that bonds with energy values below a certain threshold seem to never fail, regardless of the system disorder (although this threshold does vary with lattice size); this could naturally be taken into consideration in terms of creating contingency plans or predicting for systems exposed to the risk of spanning breakdowns. As the proportion of bonds with energies below the failure threshold seems to be independent of disorder, it can actually be equated to the percolation threshold of the corresponding

lattice, since the costliest bond of the shortest path is the same in the high disorder limit as the costliest bond of any solutions to the minimax problem. As we have already mentioned, on an unbroken lattice (that is, with no bonds removed), finding the latter with ϵ drawn from the uniform distribution between 0 and 1 corresponds to finding the percolation threshold[12], while in the high disorder limit, the bond found is also the least costly bond on the OPC backbone.

7.2 Common universality of processes

The ultrametric limit is interesting in that it allows both the shortest path problem and its OPC backbone to be equated to the same process; namely, the repeated addition of bonds at random places in an initially empty lattice, followed by checking the lattice for connectivity. Because of this equivalence, the backbone of the shortest path OPC is further related to paths in invasion percolation and minimum spanning trees, as these have been shown to correspond to the ultrametric shortest paths between any pair of sites in the kind of random lattices described. This can be used to gain a considerable advantage in predicting system breakdown: by nature, the original OPC algorithm can be rather slow for large lattices, as it doesn't only take more time to find the shortest path; the number of shortest paths that one needs to find increases as well. If we want to find the minimax or widest path OPCs, or are only interested in finding the backbone of a shortest path OPC, and the lattice is not subjected to periodic boundary conditions, the equivalency of the widest path OPC and the ultrametric shortest path on the dual lattice allows us to substitute the longer OPC process with the considerably faster invasion percolation algorithm presented in Chapter 5.

7.3 Possibilities for further research

Firstly, while we have been able to explain the link between a variety of those problems previously suspected to belong to the same universality class as the OPC backbone [9], several problems remain whose exact connection to this universality class is still not as clear. Examples of such problems are multiple invasion percolation, where an invasion percolation process is repeated several times with varying source sites; or the fractal dimension of watershed lines delimiting simulated drainage basins. Additionally, we have

mostly kept to the bond-oriented configuration. While some of the reasoning used for such lattices is easily applied to site-oriented lattices, several crucial arguments are not. Further work might help determine whether the observed similarity between site-oriented lattice phenomena can be explained by other underlying processes, or whether it is simply a result of each site-oriented process bearing a superficial resemblance to their bond-oriented equivalents, leading to similar but ultimately distinct results.

Secondly, this thesis has focused mostly on the high disorder ultrametric limit of the shortest path OPC. Further research could investigate the other disorder regimes: for instance, we could also present an extreme low-disorder limit, where the cost of a path is primarily decided by the number of bonds it contains. A better understanding of the OPC process would also depend largely on a precise explanation as to why the backbone remains unchanged even as the disorder parameter spans through a large range of values. Particularly, it could help shed light on the apparent invariance of the backbone of the shortest path OPC throughout a variety of values of the disorder parameter remains unexplained; while it consistently corresponds to the minimax OPC, we have no clear explanation as to why that is the case.

Finally, the OPC algorithm could be generalized to other problems than those presented in this thesis, such as the travelling salesman problem. The shortest path OPC model has also been applied in three dimensions; of course, exploring the behavior of other OPCs in three (or more) dimensions could be the object of further research.

Bibliography

- [1] A. Hansen, J. Kertész. Phase Diagram of Optimal Paths. *Phys. Rev. Letters.*, 93, 040601 (2001).
- [2] M. Cieplak, A. Maritan, J. R. Banavar. Optimal Paths and Domain Walls in the Strong Disorder Limit. *Phys. Rev. Letters*, 72, 2320 (1994).
- [3] M. Porto, N. Schwartz, S. Havlin, A. Bunde. Optimal paths in disordered media: Scaling of the crossover from self-similar to self-affine behavior. *Phys. Rev. Letters*, 60, R2448 (1999).
- [4] M. Cieplak, A. Maritan, M. R. Swift, A. Bhattacharya, A. L. Stella, J. R. Banavar. Optimal paths and universality. *J. Phys. A: Math. Gen.* 28, 5693 (1995).
- [5] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein. Introduction to Algorithms, Second Edition, MIT Press, 2001.
- [6] R. Dobrin, P. M. Duxbury. Minimum Spanning Trees on Random Networks. *Phys. Rev. Letters*, 86, 5076 (2001).
- [7] M. Cieplak, A. Maritan, J. R. Banavar. Invasion Percolation and Eden Growth: Geometry and Universality. *Phys. Rev. Letters*, 76, 3754 (1996).
- [8] E. Fehr, J. S. Andrade Jr, S. D. da Cunha, L. R. da Silve, H. J. Herrmann, D. Kadau, C. F. Moukarzei, E. A. Oliveira. New efficient methods for calculating watersheds. *J. Statistical Mechanics*, Vol. 2009, Sept. 2009, P09007.
- [9] J. S. Andrade Jr., S. D. S. Reis, E. A. Oliveira, E. Fehr, and H. J. Herrmann, Ubiquitous Fractal Dimension of Optimal Paths, *Comput. Sci. Eng.* 13, 74 (2011).
- [10] J. S. Andrade Jr., E. A. Oliveira, A.A. Moreira, H. J. Hermann. Fracturing the Optimal Paths. *Phys. Rev. Letters*, 103, 225503 (2009).

- [11] E. W. Dijkstra. A Note on Two Problems in Connexion with Graphs. *Numerische Mathematik*, 1, 269-271 (1959).
- [12] A. Hansen, S. Roux. Application of 'logical transport' to determine the directed and isotropic percolation thresholds. *J. Phys. A: Math. Gen.* 20 L873.
- [13] E. A. Oliveira, K. J. Schrenk, N. A. M. Araújo, H. J. Herrmann, J. S. Andrade Jr. Optimal-path cracks in correlated and uncorrelated lattices. *Phys. Rev. E*, 83, 046113.
- [14] M.L. Fredman, R.E. Tarjan, "Fibonacci Heaps And Their Uses In Improved Network Optimization Algorithms," *focs*, pp.338-346, 25th Annual Symposium on Foundations of Computer Science (FOCS 1984), 1984