(Bruno Gonçalves)

Numeric Study of Magnetic Systems and Networks

By

Bruno Gonçalves

Doctor of Philosophy, Physics

(Stefan Boettcher)
Adviser

(Fereydoon Family)
Adviser

(Phil Segré)
Committee Member

(Eldad Haber)
Committee Member

(Allon Percus)
Committee Member

Accepted:

(Lisa A. Tedesco, Ph.D.)
Dean of the Graduate School

(Date)

Numeric Study of Magnetic Systems and Networks

By

Bruno Gonçalves

B.S., Universidade do Porto, Portugal, 2001

M.S., Universidade de Aveiro, Portugal, 2004

An abstract of a dissertation

submitted to the Faculty of the

Graduate School of Emory University

in partial fulfillment of the requirements for the degrees of

Doctor of Philosophy and Master of Science

Physics Department

2008

Abstract

Recent years have seen considerable time and effort devoted to both Magnetic Systems and Complex Networks. In both cases, their properties depend strongly on the way the different elements are connected and on disorder. These similarities make their combined study an interesting and useful exercise with insights gained in one area proving valuable in the other.

We started by studying a fully conencted SK-like spin glass model with distance dependent bonds. By varying the distance dependence exponent, we easily interpolate between a fully connected and a nearest neigbor model. The way that the varying conenctivity affects the dynamics of the system is analysed. I then moved on to a purely $1D$ system of Heisenberg spins where I studied the consequences that the Topological Non-connectivity Threshold (TNT) has on systems with long range interactions. We find that this system can display ferromagnetism and hysteretic behavior for experimental times. A system of Potts spins interacting along the edges of k-regular graphs was also studied by comparing the analytical expressions for the entropy and free energy obtained through the Large Deviation Cavity Method, with the numerical results of Metropolis and Creutz simulations.

In the field of Networks, two classes of regular hierarchical graphs that have proven amenable to several nice analytical approaches are introduced. In particular, the Renormalization Group analysis of the RW leads to the use of a boundary layer concept, and results in a fractional diffusion coefficient involving the golden ratio. In the structure of Erdos-Renyi graphs I concluded that an optimal bisection requires not only the pruning of all the trees and dangling sub loops that surround the central core of the GCC. Still in this area, I analyzed how weight-weight correlations in networks impacted the transportation properties of networks. We showed that weight-weight correlations have the effect of facilitating transport. In an interdisciplinary project in the field of human dynamics, I analyzed the access logs to Emory University's website, and showed that one can use these indirect methods to characterize a given population and how it interacts with a website.

Numeric Study of Magnetic Systems and Networks

By

Bruno Gonçalves

B.S., Universidade do Porto, Portugal, 2001

M.S., Universidade de Aveiro, Portugal, 2004

A dissertation submitted to the Faculty of the

Graduate School of Emory University

in partial fulfillment of the requirements for the degrees of

Doctor of Philosophy and Master of Science

Physics Department

2008

Acknowledgments

This dissertation would not have been possible without the unending support of my parents and family, the mentorship of Prof. Stefan Boettcher, Prof. Fereydoon Family, Prof. Fausto Borgonovi and the help and collaboration of all the members of the dissertation committee. I would also like to leave a special mention Dr. José Ramasco, Prof. Allon Percus, Prof. Julien Barré, Dr. Gabriel Istrate and all the friends I've made in Emory over the last 6 years. Finally, I acknowledge the support of Los Alamos National Lab during my visits there, and computer time provided by the Open Science Grid.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

> The most exciting phrase to hear in science, the one that heralds the most discoveries, is not ”Eureka!” (I found it!) but ”That’s funny...”
>
> ˜Isaac Asimov

In 1736, the city of Könisberg on the banks of the river Pregel, in Prussia had 7 bridges connecting it to two river islands. A common topic of debate to its inhabitants was whether or not it was possible to take a walk around town, crossing each bridge exactly once and returning to the starting point. With his seminal paper, swiss mathematician Leonhard Euler answered this question and founded the field of graph theory.

Graph theory remained a mostly theoretical branch of mathematics until 1959 when two Hungarian mathematicians, Paul Erdös and Alfred Renyi introduced their model of random graphs[4, 5] that initiated a flurry of activity that would last for several decades. Their model had the benefit of being highly tractable mathematically, but, unfortunately, it could not explain the “small-world” phenomenon observed in 1969 by the american sociologist Stanley Milgram[6].

The physics community didn’t become involved in a major way in this field until 1998 when Duncan Watts and Steven Strogatz, two mathematicians at Cornell, introduced their model. Theirs was the first quantitative definition of small world, with both a logarithmic diameter and a large degree of clustering. This was closely followed by Albert-Lazlo Baraási’s and Reka Albert’s[7] introduction, in 1999, of the idea of preferential attachment as the ruling mechanism behind the evolution of real world networks. The resulting "scale-free" networks were quickly found to posess some characteristics in common with real world network structures, such as the Internet[8], the World Wide Web[9], the human sexual network[10], and many others.

Since the introduction of the Watts-Strogatz model, and following the mathematical tradition, most of the theoretical and numerical efforts in this area have been invested in studying the way Complex Networks are formed and evolve, what their topological and geometrical properties are and how different assumptions about network evolution affect these quantities. In more recent years, with a firmer understanding of the basic properties of this type of system, interest has started to shift to the ways that different processes are affected by this unusual topology and its intrinsic disorder.

Due to its well established nature, one of first models that was adapted to lie on a network instead of the more common lattices was the Ising model. This model was introduced by Ernst Ising in 1925 as a microscopic model of ferromagnetism for spin $\frac{1}{2}$ particles in $1D$[11]. His solution was not to be easily extended and the $2D$ analog was not successfully solved until Lars Onsager published his 1944, 32 page long, tour-de-force[12]. A complete analytical solution for the $3D$ case has yet to be achieved, proving that both the complexity and richness of this deceptively simple model is strongly dependent on the embedding dimension. As was later shown in 1975 by David Sherrington, Scott Kirkpatrick, Phillip Anderson and S. F. Edwards [13, 14], bond disorder can also play a crucial role in determining the macroscopic observables resulting from the microscopic details.

In this dissertation we study these two classes of problems, starting from an exploration of the effect that different topologies have on magnetic systems and evolving to an analysis of the structure and properties of real world networks. In Chap. 2 we study a class of Spin Glass models that can interpolate between short range and infinite range, and in Chap. 4 we study the effect that simply removing the geometric constraints on a system of Potts spins has on well know thermodynamic properties. Chap. 5 looks at the non-trivial properties of a finite $1D$ Heisenberg spin system. Chap. 6 clearly bridges the two parts of the dissertation. Here, we introduce to new classes of networks and study some of their magnetic, geometric and transport properties. In Chap. 7 and 8 we study the topology of Erdos-Renyi networks and the transport consequences of weight-weight correlations, respectively. Chap. 9 uses the language of networks to study how people act, a theme that we later return to in Chap. 10 where we look in detail in to how viruses in general and sexual diseases in particular spread over networks.

Part I

Magnetic Systems

Chapter 2

Spin Glasses

> Two dangers constantly threaten the world: order and disorder.
>
> - Paul Valery, french poet

2.1 Disordered Systems

The general theory describing the basic behavior of electric and magnetic systems was completed by J. C. Maxwell in his 1873 book[15, 16], ”A Treatise On Electricity And Magnetism”, the culmination of the work he himself had started in 1864 with the publication of his seminal paper, ”A Dynamical Theory Of The Electromagnetic Field”. It now seems, however, that his famous four equations whose importance and magnitude would later be compared by Einstein to the work developed centuries earlier by Sir. Isaac Newton, were only the beginning of an extremely long journey.

Although Maxwell’s equations are capable, in principle, of explaining all the observable electric and magnetic phenomena, their solution proves infeasible in many cases. One such case is the one of disordered magnetic materials with frustration, such as magnetic alloys, where several metals are mixed together in a non uniform way. In these systems, the interactions between any two pairs of magnetic spins present in a material sample is different leading to a wealth of properties that are not easily described in an analytically closed form. In particular, since the strength of each interaction is potentially different, this will lead to the possibility of a group of small, say, positive bonds being able to ”bunch” together to overcome the influence of a particularly strong negative bond thus reaching a lower energy configuration. In this example, the negative bond is said to be frustrated or broken since, locally, a different configuration would be more energetically favorable. Systems with magnetic frustration and disorder in the spin-spin interactions are referred to as spin glasses. In its simpler incarnation, each spin has only two possible states available to it, ”UP”, +, or ”DOWN”, −. This type of spins is usually referred to as Ising spins, after the author who first treated them to study ferro and anti-ferromagnetic systems.

By 1975, over 100 years after the publication of Maxwell’s book, the understanding of the effects of disorder and frustration in magnetic materials was still in its embryonic stages when a pair of papers introduced two new models that would dominate the field of what were by then known as spin glasses for the next decades. The first model, christened ”Edwards-Anderson”[14] in honor of *S. F. Edwards* and *P. W. Anderson*, the two researchers that authored it, considered a system obeying an Ising Hamiltonian of the form:

$$\mathcal{H} = -\sum_{\langle i,j \rangle} J_{ij}\sigma_i\sigma_j - H\sum_i \sigma_i \tag{2.1}$$

The second of these models was introduced by *D. Sherrington* and *S. Kirkpatrick* and obeyed a similar Hamiltonian but described a completely different system. While the *EA* model presented us with a d-dimensional system with interactions only between nearest neighbors, the *Sherrington-Kirkpatrick (SK)* model concerned an infinite dimensional system with interactions between all pairs of Ising spins. The difference between the two models is due to the different choices of the interaction matrix J:

$$J_{ij} = \begin{cases} \neq 0 \quad ij \quad \text{interact} \\ = 0 \qquad \text{otherwise} \end{cases} \tag{2.2}$$

the *EA* model chooses the J_{ij} elements to be non zero only for each pair of nearest neighbors in the d-dimensional hypercube and an *SK* system has its J_{ij} values non zero between any two pairs of nodes. Common choices for J_{ij} are Gaussian distributed random variables and + or −. As we will see, the particular form of the systems disorder is crucial in determining its properties and even an apparently simple choice can enclose a surprising

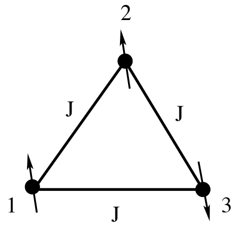

Figure 2.1: Spin configuration.

complexity. In fact, the *SK* model was originally intended to be a ”Solvable Model of a Spin-Glass”[13] due to it’s all-to-all mean-field-like nature, with one possible solution being proposed in the original paper. There were however, other solutions possible that were only uncovered by the introduction of the concept of ”Replica Symmetry Breaking” some time later. Decades would be spent studying all the complexity embedded within this apparently simple Hamiltonian with different approaches and concepts being introduced over the years to better describe and explain the properties observed in experimental spin glass systems and several books[17, 18] and reviews[19, 20, 21, 22, 23] are currently available that provide an introduction to the theoretical and experimental work that has been done in this vast field.

2.2 Frustration

The other important ingredient of spin glasses is frustration, which can be best illustrated with a simple example. Let us consider a system of spins that can interact with their nearest-neighbors and with an external magnetic field. The Hamiltonian of the this class of systems would be of the form:

$$\mathcal{H} = -\sum_{i=1}^{N}\sum_{j>i}^{N} J_{ij}\sigma_i\sigma_j - \sum_{i=1}^{N} h_i\sigma_i \tag{2.3}$$

where J_{ij} is the interaction energy between the two neighboring spins with labels i and j, $h_i$is the external magnetic field felt by spin i, $\sigma_i = \{+,-\}$ represent the orientation that a given spin can have and N is the total number of spins in the system. To simplify our analysis, we treat all spins on an equal footing by setting $J_{ij} \equiv J$ and $h_i \equiv h$. Using these definitions, the Hamiltonian takes on the form:

1	2	3	$\mathcal{H}$
+	+	+	$3(-J+h)$
+	+	-	$J+h$
+	-	+	$J+h$
+	-	-	$J-h$
-	+	+	$J+h$
-	+	-	$J-h$
-	-	+	$J-h$
-	-	-	$-3(J+h)$

Table 2.1: All possible system configurations.

$$\mathcal{H} = -J\sum_{i=1}^{N}\sum_{j=1}^{N}\sigma_i\sigma_j - h\sum_{i=1}^{N}\sigma_i \tag{2.4}$$

From basic statistical physics, we know that we can, in principle use the Hamiltonian we have just defined to calculate the partition function $\mathcal{Z}$ defined as:

$$\mathcal{Z} = \sum_{\{\sigma_1\}}\sum_{\{\sigma_2\}}\cdots\sum_{\{\sigma_N\}} e^{-\beta\mathcal{H}} \tag{2.5}$$

where each summation is taken over all individual spin configurations (±). After we successfully calculate the partition function for any given system, we can use it to determine all the observables we might be interested in[1]. Before we can proceed, we must completely define our system. We choose a system of just three spins organized as in Fig. 2.1.

This simple system has $2^3 = 8$ possible states. The states can be easily enumerated, and the corresponding value of the Hamiltonian determined. We do this in Table: 2.1.

Using this information, we quickly determine the partition function to be of the form:

$$\mathcal{Z} = 2e^{3\beta J}\cosh(3\beta h) + 6e^{-\beta J}\cosh(\beta h) \tag{2.6}$$

The free energy is defined to be:

[1] It should be noted that in most cases of interest, the calculation of the partition function in an impossible task due to the complexity of the system. However, in what follows, we will study a simple case where this calculation is fairly trivial, as a way to obtain a firm physical understanding of what is meant by frustration in spin systems.

$$F = -\frac{1}{\beta} \ln \mathcal{Z} \tag{2.7}$$

which in the $h \equiv 0$ limit becomes:

$$F_0 = -\frac{1}{\beta} \ln \left(2e^{\beta J} + 6e^{-3\beta J} \right) \tag{2.8}$$

where we have added a subscript to indicate the zero field limit. The average energy of the system is, in this case, given by:

$$\langle E \rangle_0 = -\frac{\partial \ln \mathcal{Z}}{\partial \beta} = -\frac{6Je^{-\beta J} + 6Je^{3\beta J}}{6e^{-\beta J} + 2e^{3\beta J}} \tag{2.9}$$

We can use the Free Energy expression gives us the possibility to determine the value of the Energy in the system as a function of the parameters that define the system. In particular, the ground state energy ($T = 0$) is:

$$E_0 = \lim_{\beta \to \infty} F_0 = \lim_{\beta \to \infty} \frac{1}{\beta} \ln \left(6e^{-\beta J} + 2e^{3\beta J} \right) \tag{2.10}$$

where we use the same subscript notation. Before we can proceed with the calculation of the limit, we must distinguish between two different cases, corresponding to the different signs of J. If $J > 0$, the argument of the logarithm is dominated by the second exponential, and we obtain:

$$E_o^+ = -3J \tag{2.11}$$

where the superscript denotes the sign of J. This implies that in this case, the system displays ferromagnetic behavior and that in the low temperature limit, all spins are aligned with each other. On the other hand, if $J < 0$, the first exponential dominates the logarithm and we have:

$$E_0^- = +J \tag{2.12}$$

The Entropy is defined as:

$$S = \beta(\overline{E} - F) \tag{2.13}$$

and for $h \equiv 0$, we have:

$$S_0 = \beta \left(\frac{3Je^{-\beta J} + 3Je^{3\beta J}}{3e^{-\beta J} + e^{3\beta J}} \right) + \ln \left(6e^{-\beta J} + 2e^{3\beta J} \right) \tag{2.14}$$

In the high temperature limit ($\beta \to 0$):

$$S_0^{\infty} = \ln 8 \tag{2.15}$$

On the other hand, if we make the temperature go to zero ($\beta \to \infty$) we must consider two different cases. If $J > 0$, the ferromagnet has two possible ground states, all UP or all DOWN. The entropy is, therefore:

$$S_{\infty}^{+} = \ln 2 \tag{2.16}$$

But if $J < 0$, the system behaves as a frustrated antiferromagnet, which, in this case, implies that at least one of the bonds must be broken. Since we have three possible bonds and an UP-DOWN degeneracy, there are $6 = 3 \times 2$ possible ground states and the entropy is given by:

$$S_{\infty}^{-} = \ln 6 \tag{2.17}$$

This deceptively simple example already displays all the ingredients necessary for the complex behaviors we observe in spin glasses. The complexity can only increase when there is a mix of ferromagnetic and anti-ferromagnetic bonds of different strenghts. We explore this more general case in the rest of this Chapter.

2.3 Sherrington-Kirkpatrick model

Now that we understand the basic ingredients that constitute a spin glass, and as a first approach, we choose to devote some of our attention to the venerable Sherrington-Kirkpatrick

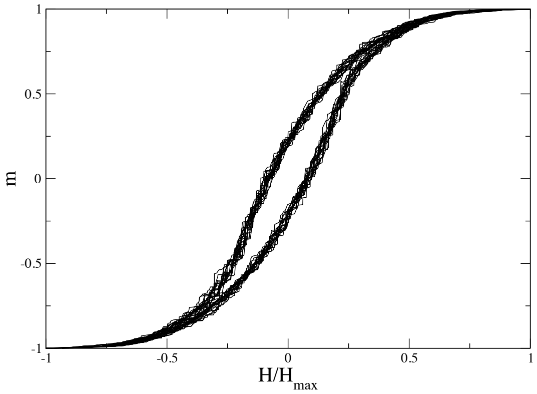

Figure 2.2: Hysteretic cycle. H_{max} is the magnetic field that saturates all the spins in the system leading to complete magnetization.

model before proceeding to higher dimensional models. This will help us hone the methods and physical intuition that will guide us through the study of other systems.

The dynamics of a system can be studied by observing how it reacts to a variation of the magnetic field to which it is subjected. The applied magnetic field will try to make the magnetic spins present in the sample to align with it, a process that consumes energy (supplied by the magnetic field) and requires a certain amount of time. This phenomenon leads to an observable lag between a change in the magnetic field and the system's magnetization, by varying the field in a sinusoidal fashion, and observing the behavior of the magnetization as a function of the field we obtain a figure known as a the hysteretic cycle of the system as shown in Fig. 2.2. Hysteresis is a phenomenon known for several decades and that is commonly used as a tool to study the dynamics of magnetic systems and we shall keep with tradition and start the study of our system by the observation of its hysteretic properties.

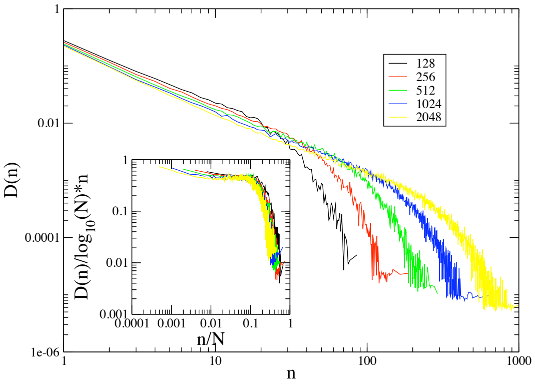

Figure 2.3: Distribution of avalanche sizes for the Sherrington-Kirkpatrick model at different sizes. inset) Scaling collapse.

2.4 $1D$ Ising chain

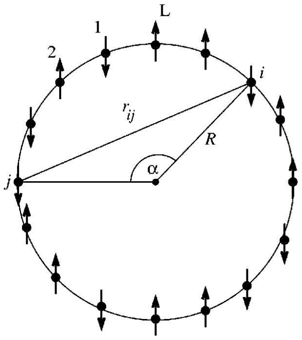

Figure 2.4: Geometry of the $1D$ Ising chain model, from [1].

As interesting as the SK model we studied in the previous section is, we will go one step further and focus our attention in a modification of this model which was proposed in 2003 by *H. G. Katzgraber* and *A. P. Young*[1, 2] that considered an interpolate between the two limits illustrated above by the *SK* and the *EA* models. Their model achieves this interpolation by considering a closed ring of L Ising spins with the interaction constant between any pair of them being a given by a Gaussian distributed random variable as in the *SK* model, but scaled by a factor of $r_{ij}^{-\sigma}$ where r_{ij} , the Cartesian distance between two spins at positions i and j is given by:

$$r_{ij} = \frac{L}{\pi} \sin\left(\frac{\pi\,|i-j|}{L}\right) \tag{2.18}$$

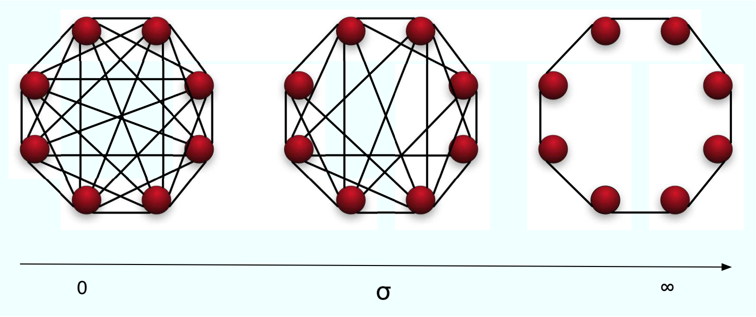

Figure 2.5: Systems described by the KY model as a function of σ. For $\sigma \equiv 0$ we reobtain the SK model and as $\sigma \rightarrow \infty$ all the long range links are destroyed become essentially negligible leaving us only with NN interactions.

The geometry of our system is shown in Fig. 2.4 copied from [1]. Varying σ we can interpolate between the two classes of systems we are interested in. For $\sigma \equiv 0$ we reobtain the classical *SK* model, and as σ increases, the strength of the long range interactions becomes increasingly smaller resulting in a gradual approximation to a Nearest Neighbor *(NN)* model similar to the *EA* one, as shown in Fig. 2.5.

As a first step in our analysis, we measure $\langle S(\sigma)\rangle$, the average avalanche size as a function of σ at different system sizes. This measurement will help us determine what range of σ we need to study, since it should become system size independent in the NN limit. As we show on the left side of Fig. 2.6, this happens near $\sigma \approx 2.0$, thus restricting our interval of interest to $\sigma \in [0, 2]$.

We find that this quantity obeys a scaling law of the form:

$$\langle S(N,\sigma)\rangle \sim \frac{NA(\sigma)\, f(N,\sigma) + B(\sigma)}{\log^2(N)} \tag{2.19}$$

where $A(\sigma)$, $B(\sigma)$ are linear functions of σ, and $f(N,\sigma)$ is plotted on the right side of Fig. 2.6. The overlap of all the curves in the interval $\sigma \in [0, 0.5]$ means that all the N dependence has been captured by the $N/\log^2(N)$ term. The scaling function becomes increasingly worst with $\sigma > 0.5$, signaling an increasingly stronger deviation towards a new N dependence.

This change in behavior at $\sigma = 0.5$ is due to a topological change in the system. For all

σ, the mean field transition temperature, T_c^{MF}, is of the order:

$$\left(T_c^{MF}\right)^2 \propto \sum_{i\neq j} \left[J_{ij}^2\right]_{av} = \sum_{i\neq j} r_{ij}^{-2\sigma} \sim N^{-\sigma+\frac{1}{2}} \tag{2.20}$$

where $[\cdot]_{av}$ denotes an average over disorder, becomes finite in the thermodynamic limit. The system goes from having an Infinite Range behavior to a Long Range character, where the dynamics are dominated by the long range tail of the interaction matrix, J_{ij}. The complete phase diagram as a function of σ is shown in Fig. 2.7.

An avalanches size is determined by the total number of spin flips that occur. If the same spin happens to flip several times, then it will be counted multiple times as well, but we may also be interested in identifying and counting just which spins flip, regardless of how many times they do so. We can easily measure this during our stochastic runs by keeping track of which spins have already been flipped once during the course of a given avalanche, so that we can count them only one. The ratio S/U of the avalanche size over the number of unique spins flipped gives us a measure of how important loops are in the dynamics of the system, a large ratio will indicate that perturbations spread throughout the system and keep returning to the same spin, while a number close to unity would mean that avalanches propagate in just one direction and never double back.

On the left side of Fig. 2.8 we plot $\langle S/U\left(\sigma\right)\rangle$ for different file sizes. We find that the ratio between the size of the avalanche and the number of unique spins flipped is always very close to 1, and obeys a scaling law of the form:

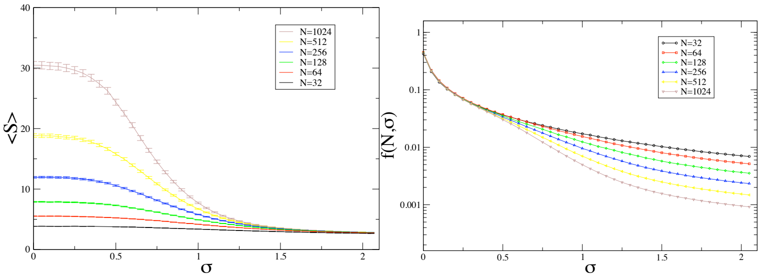

Figure 2.6: left)Average avalanche size as a function of σ for different system sizes. right) Corrections to scaling.

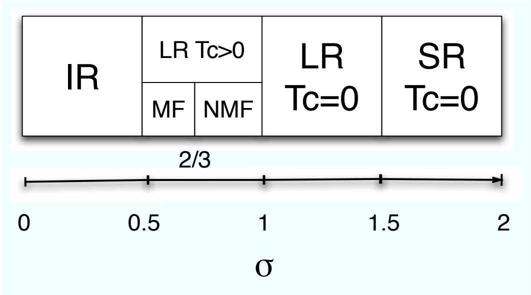

Figure 2.7: Phase diagram of the distance dependent SK model according to [2].

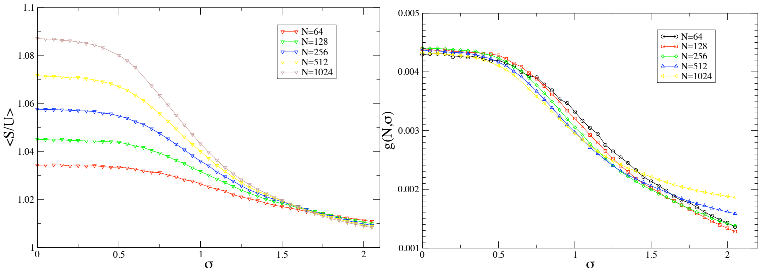

Figure 2.8: left) Ratio between the avalanche size and the number of spins that were flipped. right) Scaling function.

$$\left\langle \frac{S}{U} \left(N, \sigma\right) \right\rangle = 1 + \sqrt{N} \left[g\left(N, \sigma\right) - A \log\left(N\right) \right] \tag{2.21}$$

where A is a constant and $g(N, \sigma)$ is shown on the right side of Fig. 2.8. The scaling collapse is very good up to near $\sigma \approx 1.5$ where the system acquires a purely short range behavior.

2.5 Hysteretic Optimization

In 2002, Zarand *et al[24, 25, 26]* proposed using a process similar to a.c. demagnetization to search for the ground state of spin glasses. The classical example of this process is when a CRT screen is magnetized by proximity to a magnet. In the magnetized region, the colors are distorted. To repair it, another magnet must be put in close proximity to the affected region and then gradually moved away in a circular pattern. In this way, the screen will be brought to a demagnetized state.

Algorithm 1 Hysteretic Optimization

```
1. Set $H = H_1$ large enough such that $S_i = \xi_i \ \forall\ i$. Set $E_{\{\min\}} = \mathcal{H}\,(= \mathcal{H}_{HO}|_{H=0})$.
2. Decrease $H$ until one spin becomes unstable and allow the system to relax. If $\mathcal{H} < E_{\{\min\}}$, set $E_{\{\min\}} = \mathcal{H}$.
3. Optional: When $H$ passes zero, randomize $\xi_i$, leaving the current configuration stable.
4. At each turning point $H = H_n = -\gamma_{n-1} H_{n-1}$, for $0 < \gamma_n < 1$, reverse the direction of $H$.
5. Terminate when amplitude $|H_n| < H_{\{\min\}}$,.
6. Restart at 1 for $N_{run}$ times with a new, random set of $\xi_i$'s .
7. Return the best $E_{\{\min\}}$ over all runs.
```

They proposed replacing the Hamiltonian:

$$\mathcal{H} = -\sum_{ij} J_{ij}\sigma_i\sigma_j \tag{2.22}$$

by:

$$\mathcal{H} = -\sum_{ij} J_{ij}\sigma_i\sigma_j - H\sum_i \xi_i\sigma_i \tag{2.23}$$

where $\xi_i = \pm 1$ are random local fields, and H is an oscillating external field. In effect, the process described above corresponds to a damped hysteretic cycle, where the field during half-cycle n is given by: $H_n = -\gamma H_{n-1}$ with $\gamma < 1$. This algorithm is described in Alg.1, and was shown to work very well for the Sherrignton-Kirkpatrick model and not at all in the case of the Edwards-Anderson model.

The $1D$ Ising chain described in the previous sections, with its interpolation between these two limits, is ideally placed to explore this dichotomy. Why doesn't HO work for the EAmodel? What are the ingredients that HO requires to work? We start by comparing the results obtained by HO with the results obtained using a long run of Extremal Optimization[27, 28]. The results are shown in Fig.2.9, where we plot the percentage difference between the HO results and the EO results for exactly the same instances. As we can see, HO becomes increasingly worse as σ increases, but there is a clear transition at $\sigma = 0.5$. As we saw in Fig. 2.7, this is when our model loses it's Infinite Range character

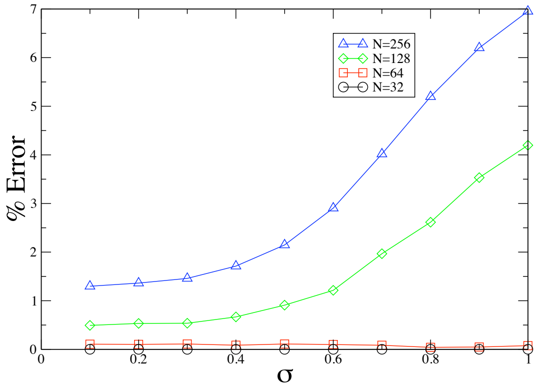

Figure 2.9: Percentage difference between the ground state found by Hysteretic Optimization and the one found by Extremal Optimization.

to become merely Long Ranged.

The explanation lies in Fig. 2.6, where we see that this is also where the characteristic size of the avalanches changes so it no longer scales with the system size, N. This is a strong indication that the efficacy of HO is strongly dependent on the presence of large avalanches that can take the system from one point in configuration space to a very different one, thus gradually exploring distant regions of this space in search for the lower energy configuration.

To understand how the dynamics changes, we study how the algorithms approaches the final configuration, at $H \approx 0$ and $m = 0$, by looking at the auto-correlation function given by:

$$\langle S_i^0 S_i^\tau \rangle - \langle S_i^0 \rangle \langle S_i^\tau \rangle \tag{2.24}$$

where the indices denote summation over all i. In Eq. (2.24), we measure the overlap between the final configuration and those obtained a number of τ *complete cycles* backwards in the past at their $H = 0$-crossing. Intuitively, we expect that the configurations seen at the beginning of the procedure (large values of τ) will be completely unrelated to the final configuration ($\tau = 0$), resulting in a value near zero for this quantity. However, as the algorithm takes its course and approaches its conclusion, so too must the configurations start approaching the final one, corresponding to a value close to 1. The way in which it varies from values near 0 to values close to 1 gives us information about the way exploration of configuration space occurs. The longer the period during which the correlations are close to 0, the larger the volume explored, and the faster it gets close to zero, the earlier the

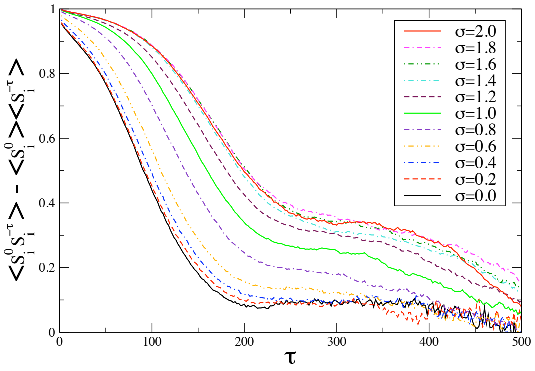

Figure 2.10: Correlation with the final solution as a function of the time in the past.

system restricts itself to a given region, thus limiting the quality of the solution it is able to find.

In Fig. 2.10 we plot this quantity for the case of $N = 256$, averaged over 1000 different instances for each value of σ and with 10 different runs per instance. For small values of σ, the plateau at low correlations is extended (lower solid black curve), followed by an increase towards the value of 1 near the final stages $\tau \rightarrow 0$. As σ increases, the autocorrelations increase within the plateau which itself shortens, and the tendency towards 1 becomes noticeable right from the onset (upper solid black curve for $\sigma = 2$). This is a clear demonstration of the ideas expressed earlier, that the volume of configuration space explored becomes smaller with the decrease in avalanche size corresponding to increasing σ.

2.6 Magnetic recording

One exceedingly important technological application of magnetizeable materials is in the recording of information, be it in the form of hard disk drives, magnetic tapes or even your old vinyl music records. By defining one value of the magnetization (say $m_1 > 0$) to represent the bit 1, and its opposite ($m_1 < 0$) to represent the bit 0 and dividing a long stripe of magnetic material (or a long circular track in the case of disks) in small independent sections of predetermined length, one can easily record and preserve the large amounts of digital information which are essential to today's information driven age.

Realistic recording materials must possess several important properties; they must be

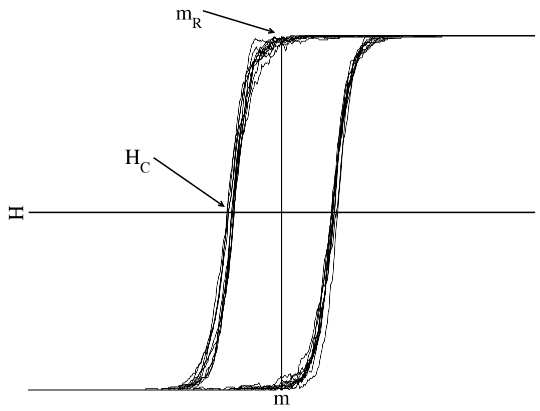

Figure 2.11: Coercivity and Remanence on the hysteresis cycle.

easily recorded (and erased) by the application of precise external fields, yet be able to maintain their magnetization (preserve the recorded information) over as long a period of time as possible. while resisting spurious or stray magnetic fields[2]. We also want to be able to quickly record and retrieve the information contained in these devices. In summary, we want to be able to produce, in a period of time, $\tau_W \to 0$, a change in magnetization that will last for a period $\tau_S \to \infty$, while still being able to retrieve it in $\tau_R \sim \tau_W$.

These magnetic properties that determine whether or not a material can obey these stringent requirements are encoded in its hysteretic cycle. The ability to preserve magnetization for an extended period of time, is determined by the systems coercivity, H_c, the field required to reduce the magnetization to zero, and the point where the hysteretic line intercepts the H axis coming from positive magnetizations. The capacity to clearly record information is given by the remanence, m_R, the magnetization that the material can retain after the driving external field has been reduced to zero. In the hysteresis loop this occurs when the line first crosses the m axis. These two quantities are represented in Fig. 2.11.

2.7 Publications

[29] B. Gonçalves and S. Boettcher, Hysteretic Optimization For Spin Glasses, *J. Stat. Mech*, P01003, (2008)

[2] Such as the ones produced by CRT screens which are still widely available.

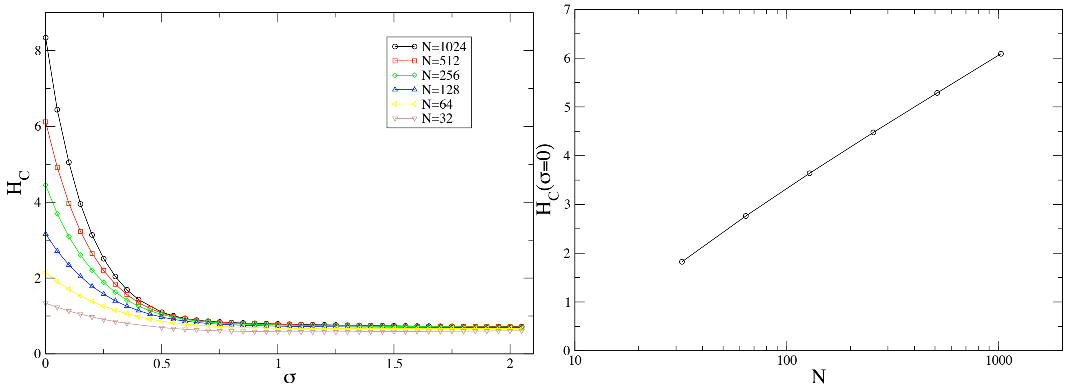

Figure 2.12: Coercivity as a function of σ.

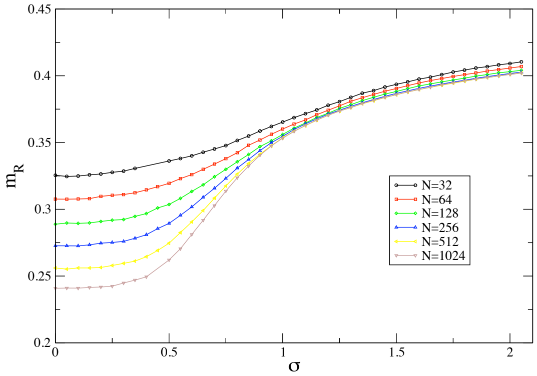

Figure 2.13: Remanence as a function of σ.

Chapter 3

Monte-Carlo Methods

> ”If you must play, decide upon three things at the start: the rules of the game, the stakes, and the quitting time.”
>
> - Chinese proverb

Named after a casino in Monaco, Monte-Carlo methods work in a way that closely mimics the gambling that occurs there. Dice are rolled, and fortune decides who is the winner for the day. Players hope they can beat the odds and make their dreams come true., while casino owners rest assured in the knowledge that the large fluctuations that attract gamblers will average out and leave them with a healthy profit.

Monte-Carlo methods[30, 31, 32, 33, 34] are, in a way, a formalization of this process in a way that allows them to be used to model stochastic problems. Despite its variety, they always involve generating random configurations of the system (rolling the dice), measuring some property of this configuration (who is the winner), and, finally, averaging over many such trials (the overall profit).

Physical processes are often stochastic and just as often we don’t posses detailed information about the system. What will be the result of a coin toss? Where exactly are the impurities in a semi conductor? However, we must somehow account for this while studying these problems, and this is where Monte-Carlo methods have proven most useful.

3.1 Master Equation

Stochastic models where the transitions from a given state s to another state s' occur spontaneously at a rate $\omega_{s\rightarrow s'} \geq 0$, have a probability distribution $P\left(s,t\right)$, of being in state s at time t that evolves deterministically in time according to a master equation of the form:

$$\frac{\partial}{\partial t}P\left(s,t\right)=\sum_{s'}\omega_{s'\rightarrow s}P\left(s',t\right)-\sum_{s}\omega_{s\rightarrow s'}P\left(s,t\right) \tag{3.1}$$

where the first summation accounts for transitions from all states s' in to state s and the second term accounts from transitions away from state s on to some other state s'. This equation describes the probability flow that is responsible from creating or destroying any given configuration. Both terms are correlated in such a way that the usual normalization condition $\sum_{s}P\left(s',t\right)\equiv 1$ is valid at all possible values of t. If we know the complete $P\left(s,t\right)$ at any point in time, we can use this equation to determine the form of $P\left(s,t\right)$ at any other instant. One important consequence of this fact is that processes of this type don't posses any type of memory about how they reached state s, thus clearly marking them as Markov Chain[35, 36, 37, 34, 32] processes. It is also important to note that the $\omega_{s\rightarrow s'}$ are rates and not probabilities, being able to take on values larger than unity and defining an intrinsic time scale which can be modified by rescaling all the ωs by a constant.

Given that Eq. 3.1 describes a time dependent process, we can follow the usual physical definition of equilibrium, namely, that an equilibrium process is such that any explicit time dependence is eliminated, or, mathematically, the point at which:

$$\frac{\partial}{\partial t}P\left(s,t\right)\equiv 0 \tag{3.2}$$

from where we can easily see that at equilibrium, the system must obey:

$$\sum_{s'}\omega_{s'\rightarrow s}P\left(s'\right)=\sum_{s}\omega_{s\rightarrow s'}P\left(s\right) \tag{3.3}$$

As a particular case, we have:

$$\omega_{s'\to s} P\left(s'\right) = \omega_{s\to s'} P\left(s\right) \tag{3.4}$$

This condition is usually referred to as the ”detailed balance condition”. One should note, that detailed balance is a ”sufficient” condition for equilibrium, but not a required condition. This means that some system can disobey this principle and still be at equilibrium, but it also means that any process that obeys this prescription is necessarily in equilibrium.

3.2 Importance Sampling

Now that we have a better understanding as to when a system reaches equilibrium and of how it evolves from equilibrium state to equilibrium state we can start answering the important question of how we can measure average equilibrium properties of the system in a way that ensures the correctness of the result. In conventional Statistical Physics, the average of an observable is given by:

$$A \equiv \left\langle A\left(s\right)\right\rangle_S = \frac{1}{Z}\int_S \mathrm{d}s A\left(s\right) e^{-\beta H(s)} \tag{3.5}$$

where:

$$Z = \int_S \mathrm{d}s e^{-\beta H(s)} \tag{3.6}$$

is the system's partition function, $\beta = \left(k_B T\right)^{-1}$ is the inverse temperature, $H\left(s\right)$ is the Hamiltonian and the integration is carried out over all phase space S. However, for practical reasons, it is usually impossible to perform these integrals and we are forced to estimate their value using a finite number, n, of phase space points, $\{s_1, s_{2,} \cdots, s_n\}$.

A back of the envelope calculation allows us to discretize integrals, and obtain:

$$\left\langle A\left(s\right)\right\rangle_S = \frac{\sum_{i=1}^{n} A\left(s_i\right) e^{-\beta H(s_i)}}{\sum_{i=1}^{n} e^{-\beta H(s_i)}} \tag{3.7}$$

which is exact in the $n \to \infty$ limit. Ideally, these points would be chosen and weighted using some distribution, $P\left(s\right)$, such that we can obtain the best possible result with the fewest number of points possible. With this approach, the last expression takes on the form:

$$\langle A(s)\rangle_S = \frac{\sum_{i=1}^{n} A(s_i)\, e^{-\beta H(s_i)} P(s_i)^{-1}}{\sum_{i=1}^{n} e^{-\beta H(s_i)} P(s_i)^{-1}} \tag{3.8}$$

which clearly identifies, the choice of $P(s_i)$:

$$P(s_i) = e^{-\beta H(s_i)} \tag{3.9}$$

that simplifies Eq. 3.7 in to:

$$\langle A(s)\rangle_S = \frac{1}{n}\sum_{i=1}^{n} A(s_i) \tag{3.10}$$

this is usually referred to in the literature as ”importance sampling”. By using this form of $P(s_i)$ in Eq. 3.4, we easily obtain:

$$\frac{\omega_{s\to s'}}{\omega_{s'\to s}} = e^{-\beta\delta H} \tag{3.11}$$

where δH is the change in energy between the final state s' and the initial state s. This practical prescription on how to implement a Markov Chain process that would allow us to easily calculate the equilibrium value of physical observables using a computer was introduced in 1953 by Nicholas Metropolis *et al*[38] and is probably one of the most influential algorithms of the last 100 years. It has since been used in a multitude of different applications, with several good reviews and introductory texts [34, 39, 30, 35, 40] available.

3.3 Spin Dynamics

Eq. 3.11 doesn't give us a unique definition of the transition rates $\omega_{s\to s'}$, it only specifies ratios, giving us some freedom to choose the set of transition probabilities that best suits our needs. Since we are mostly concerned with Magnetic Spin Systems we will focus on the different possibilities that are suited to this type of system. One of the better know choices

Algorithm 2 Metropolis Algorithm

```
1. Start with any valid configuration.
2. Choose any other valid configuration and measure the energy difference, $\delta H$, between the two states.
3. Generate a random number, $p$, uniformly distributed between 0 and 1.
4. Move the system to the new state if $p \leq P\left(s \rightarrow s'\right)$ , where $P\left(s \rightarrow s'\right)$ is given by Eq. 3.12.
5. Repeat steps 2–3 until the system reaches equilibrium.
6. Continue to evolve the system in the same way as long as necessary to measure the observables you are interested in.
```

for the probability of accepting a transition from state s to state s', $P\left(s \rightarrow s'\right)$, is:

$$P\left(s \rightarrow s'\right) = \begin{cases} e^{-\beta\delta H} & \delta H > 0 \\ 1 & \delta H < 0 \end{cases} \tag{3.12}$$

Physically, this means that transitions to states with lower energies are always accepted, while transitions to higher energy states are accepted with a probability that decays exponentially on the energy difference separating the two states. The energy jump that the system is allowed to take is measured with respect to the temperature, thus correctly mimicking thermal fluctuations.

The basic Metropolis Algorithm, shown in Alg. 2, still poses some questions that must be answered before it can be successfully used, in particular, what's the best way to choose new possible configurations in order to minimize the rejection, and when should we stop. The answer to the second question is provided by the definition of standard error:

$$S_E = \sqrt{\frac{\sum_{i=1}^{N}\left(A_i - \langle A\rangle\right)^2}{N}} \tag{3.13}$$

where N is the number of independent measurements, A_i, and $\langle A\rangle$ is their average value. By monitoring this quantity during step 6, while taking care to make sure that two consequent measurements are uncorrelated, we can determine when the error in the measured value of our observables is acceptably small.

The answer to the first question is somewhat more involved. Every time we attempt a transition, we must generate a new configuration and measure its energy, which can require an expensive computation. It would be best to only generate configurations that have a reasonable chance of being accepted so to minimize wasteful rejections. The best way to do this is notoriously application dependent. We want to explore phase space as much as possible, while minimizing the energy difference at each attempt. In spin systems, this can be done by changing the state of just one spin at a time. For Ising spins, this means just flipping it, while for Heisenberg spins, due to their vector nature, it implies changing their orientation in a multidimensional space. This method is usually referred to as Metropolis Dynamics. Another possible choice, known as Kawasaki Dynamics, is to choose two different spins, and exchange their configurations, thus keeping the total magnetization constant, which can be useful in some cases. On the other extreme, we can try to ”flip” large connected clusters of spins, where the energy difference between the two configurations scales with the length of the boundary and can allow for large jumps in phase space which would require large amounts of time if one were to use Metropolis Dynamics. Cluster Algorithms to determine the best way of selecting the clusters to be flipped is still an active field of research.

3.4 Density Of States

As recently observed by Hove[41] in 2004, if during our Metropolis runs we maintain a histogram, $\mathcal{N}(E)$, of the energy values, E, our system visits, at the end we would have:

$$\mathcal{N}(E) \propto g(E)\, e^{-\beta E} \tag{3.14}$$

where $g(E)$ is the density of states of the system. In principle, we can invert this equation to estimate $g(E)$for the entire energy range:

$$A_\beta g(E) = \mathcal{N}(E)\, e^{\beta E} \tag{3.15}$$

where A_β is a unknown temperature dependent proportionality constant. In practice, however, due to limited runtime and statistical limitations, our histogram only includes a small range of energies in the vicinity of $\langle E(\beta)\rangle$, so we must combine the result of different runs at different temperatures in order to obtain a complete estimate of $g(E)$:

$$g(E) = g_0 \sum_{i=1}^{N} A_{\beta_i} \mathcal{N}_i(E)\, w_i(E)\, e^{\beta_i E} \tag{3.16}$$

with:

$$w_i(E) = \frac{\mathcal{N}_i(E)}{\sum_{i=1}^{N} \mathcal{N}_i(E)} \tag{3.17}$$

being the weight corresponding to each of the N different histograms, and g_0 being an overall proportionality constant. On must also adjust the different proportionality constants A_{β_i} so that the different estimates match in the overlapping regions. Hove proposes to do this by minimizing:

$$\chi^2 = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \sum_{E} \mathcal{N}_i(E)\mathcal{N}_j(E) \times \left(\ln A_i + \beta_i E + \ln \mathcal{N}_i(E) - \ln A_j - \beta_j E - \ln \mathcal{N}_j(E)\right)^2 \tag{3.18}$$

where the term in parenthesis is simply given by $\ln g_{\beta_i}(E) - \ln g_{\beta_j}(E)$. The natural logs are used in order to avoid numerical overflows due to the large magnitude of the quantities involved.

3.5 Microcanonical Monte-Carlo

In 1983, Creutz[39] introduced a variation of the metropolis algorithm we described above, that allows for the determination of constant energy properties of the system. Creutz envisioned a demon that travels around the system, transferring energy from one part of the system to another, while keeping the overall energy, $E_{system} + E_{demon}$, constant, as required by the microcanonical ensemble.

When the demon attempts to move the system from configuration s to a configuration $s'$with lower energy, it keeps the excess energy. On the other hand, when the configuration s' has a higher energy, the move is only allowed for, if the demon can "pay" for the extra energy from it's own energy reserves.

Standard statistical mechanics arguments can be used to show that the demon's energy will be distributed as:

$$P\left(E_{demon}\right) \propto \exp\left(-\beta E_{demon}\right) \tag{3.19}$$

with

$$\beta = \frac{1}{4}\ln\left(1 + \frac{4}{\langle E_{demon}\rangle}\right) \tag{3.20}$$

The attentive reader will note that, since the demon's energy is not strictly constant, the system's energy must also vary. Fortunately, for large systems, the demon's energy will be just a small fraction of the total energy and the system's energy will be, effectively, constant. We should also note that if, at each step, we replace the demon's energy with a random variable selected with Boltzmann weight $\exp\left(-\beta E_{demon}\right)$, then this algorithm reduces to the usual canonical Metropolis.

Chapter 4

Ensemble Inequivalence

> ”Inequality is the only bearable thing, the monotony of equality can only lead us to boredom.”
>
> – Francis Picabia, French painter and poet

Textbook statistical physics teaches us that there is a clear and precise equivalence between the different statistical ensembles, microcanonical, canonical and grand canonical that allows us to easily move from one to another without affecting the basic properties of the system. In particular, this has lead to the possibility of affecting the knowledge about one of the ensembles to guide us in the study of the others, with spectacular results in various systems.

The entire construction of thermodynamics and statistical physics rests on one underlying fundamental principle, that the systems under consideration are large enough and that the interactions involved decrease fast enough that the total energy of the system is extensive, or in other words, energy increases linearly with the system size[1]. This implies, among other things, that all forces between particles have an effective range beyond which the system can be divided in to independent macroscopic pieces, and boundary effects are negligible beyond a small distance from the surface.

However, if the interactions are sufficiently strong, boundary effects will be significant through out the ”bulk” of the material and a whole new host of physical phenomena becomes possible, such as the possibility of a region of negative specific heat in the microcanonical

[1] There are several possible formulations of the thermodynamical limit which are all equivalent. We chose this one for reasons that will become clear later.

ensemble that is determinantly forbidden in the canonical ensemble, leading to different physics, and phase transitions, in each of the ensembles, a property known as ensemble inequivalence. In this chapter we study this type of inequivalence in a model system both analytically and numerically.

4.1 Model

Let us consider a system of Potts spins with three possible states each, a, b and c. In the "ferromagnetic" case, only spins in the same state interact with each other decreasing the total energy by J, the interaction energy. The Hamiltonian for this system can be written, in the absence of an external field:

$$\mathcal{H} = -J \sum_{\langle i,j \rangle} \delta_{q_i q_j} \tag{4.1}$$

where $\langle i, j \rangle$ denotes all the bonds in the system, $q_{1,2}$ is the state of spin i and j, respectively, and $\delta_{q_i q_j}$ is a kronecker delta. In this form the energy simply counts the number of satisfied bonds between spins. The ground state energy is simply the total number of bonds in the system (and the total number of terms in the summation), E. By adding EJ to the energy, we obtain:

$$\mathcal{H} = J \sum_{\langle i,j \rangle} (1 - \delta_{q_1 q_2}) \tag{4.2}$$

which in effect redefines the ground state to have energy 0 and the Monte-Carlo the system to count the number of broken bonds. This form, while preserving the physics of the system, simplifies our analysis since it's easy to count broken bonds.

4.2 Large Deviation Theory

The Large Deviation Cavity Method is a modification of the usual cavity method introduced by *Mezzard* and *Parisi* [42]. Like its predecessor, it analytically evolves the system by considering three different situations, the addition of a site, the addition of a link and iteration, the addition of a cavity spin in a self-consistent way. Using these three processes we can then calculate the microcanonical entropy L as a Legendre transform of the canonical free energy $\mathcal{F}$. We shall apply the method described in detail by *Rivoire* in [43], and we divide the calculation in several parts for easier understanding. We refer to the results in Rivoire's paper and simply apply them to our system.

4.2.1 Iteration

From [43], Eq. 39 we know, that in general:

$$P\left(h_0\right) = \frac{1}{Z'} \int \prod_{j=1}^{r-1} \mathrm{d}h_j P\left(h_j\right) \delta\left(h_0 - \widehat{h}^{(r-1)}\left(\{h_j\}\right) e^{-y\Delta E_n^{(r-1)}}\left(\{h_j\}\right)\right) \tag{4.3}$$

The cavity fields h tell us what is the state of the spin. If we call the three possible spin states a, b and c, then $h \in \{a, b, c\}$ and we can associate the probabilities p_a, p_b and p_c to these three different possibilities. We must now compute $\Delta E_n^{(r-1)}$, the energy shift when a new node is added and connected to $k-1$ other nodes. If $k = 3$, the new node can be in states a, b and c, with *a priori* probability of 1/3 and $h_0 = a, b, c$ accordingly. The different possible values of the fields h_1 and h_2 are described in Table 4.1.

h_0	(h_1, h_2)	$\Delta E_n^{(r-1)}$	*prob*
a	(a,a)	0	$\frac{1}{3}p_a^2$
	(a,b)	1	$\frac{1}{3}2p_ap_b$
	(a,c)	1	$\frac{1}{3}2p_ap_c$
	(b,b)	2	$\frac{1}{3}p_b^2$
	(b,c)	2	$\frac{1}{3}2p_bp_c$
	(c,c)	2	$\frac{1}{3}p_c^2$
b	(b,b)	0	$\frac{1}{3}p_b^2$
	(b,a)	1	$\frac{1}{3}2p_bp_a$
	(b,c)	1	$\frac{1}{3}2p_bp_c$
	(a,a)	2	$\frac{1}{3}p_a^2$
	(a,c)	2	$\frac{1}{3}2p_ap_c$
	(c,c)	2	$\frac{1}{3}p_c^2$
c	(c,c)	0	$\frac{1}{3}p_c^2$
	(c,a)	1	$\frac{1}{3}2p_cp_a$
	(c,b)	1	$\frac{1}{3}2p_cp_b$
	(a,a)	2	$\frac{1}{3}p_a^2$
	(a,b)	2	$\frac{1}{3}2p_ap_b$
	(b,b)	2	$\frac{1}{3}p_b^2$

Table 4.1: Energy shifts and probabilities for the $k = 3$ case.

Using these values in Eq. 4.3, we obtain:

$$\begin{cases} p_a = \frac{1}{Z}\frac{1}{3}\left\{p_a^2 + 2p_a\left(p_b+p_c\right)e^{-y} + \left(p_b+p_c\right)^2 e^{-2y}\right\} \\ p_b = \frac{1}{Z}\frac{1}{3}\left\{p_b^2 + 2p_b\left(p_a+p_c\right)e^{-y} + \left(p_a+p_c\right)^2 e^{-2y}\right\} \\ p_c = \frac{1}{Z}\frac{1}{3}\left\{p_c^2 + 2p_c\left(p_a+p_b\right)e^{-y} + \left(p_a+p_b\right)^2 e^{-2y}\right\} \\ Z = \frac{1}{3}\left\{\left[p_a + \left(p_b+p_c\right)e^{-y}\right]^2 + \left[p_b + \left(p_a+p_c\right)e^{-y}\right]^2 + \left[p_c + \left(p_a+p_b\right)e^{-y}\right]^2\right\} \end{cases} \tag{4.4}$$

from where we can easily calculate numerically $p_{a,b,c}$. For larger k the generalization is straightforward, we have:

$$p_a = \frac{1}{3Z}\left[p_a + \left(p_b+p_c\right)e^{-y}\right]^{k-1} \tag{4.5}$$

Before we can use Rivoire's [43] Eq. 40:

$$\mathcal{F}\left(y\right) = -\ln\left[\int \mathrm{d}\Delta E P_n^{(r)}\left(\Delta E\right)e^{-y\Delta E}\right] + \frac{r}{2}\ln\left[\int \mathrm{d}\Delta E P_l\left(\Delta E\right)e^{-y\Delta E}\right] \tag{4.6}$$

to calculate $\mathcal{F}\left(y\right)$ we must first determine $P_l\left(\Delta E\right)$, that describes link addition, and $P_n^{(r)}\left(\Delta E\right)$, that describes site addition.

4.2.2 Link addition

When adding a new link between two pre-existing cavity sites, we must take in to account six possible cases, corresponding to the different configurations of the two sites. Three of these configurations increase the energy by one unit and the remaining three conserve the energy. All six possibilities are described in Table: 4.2.

(h_1, h_2)	ΔE	$prob$	$P_l\left(\Delta E\right)$
(a,a)	0	p_a^2	$p_a^2 + p_b^2 + p_c^2$
(b,b)	0	p_b^2	
(c,c)	0	p_c^2	
(a,b)	1	$2p_ap_b$	$2\left(p_ap_b + p_ap_c + p_bp_c\right)$
(a,c)	1	$2p_ap_c$	
(b,c)	1	$2p_bp_c$	

Table 4.2: Configurations (h_1, h_2), energy shifts ΔE and total probabilities $P_l\left(\Delta E\right)$ for the case of a link addition. The numeric factors stem from combinatoric arguments.

Using these values we can calculate the first integral in the expression for $\mathcal{F}(y)$:

$$\int \mathrm{d}\Delta E P_l\left(\Delta E\right) e^{-y\Delta E} = \left(p_a^2+p_b^2+p_c^2\right) + 2\left(p_a p_b + p_a p_c + p_b p_c\right) e^{-y} \tag{4.7}$$

4.2.3 Adding a site

When adding a new site, we have an *a priori* probability of $\frac{1}{3}$ of the new site being in any of the three allowed states, a, b and c. In the case $k=3$, this new site has to be connected to three previously existing cavity sites. The different states in which these three sites can be create a large number of different situations that need to be accounted for. All the possibilities are enumerated Table 4.3 and grouped by energy shift ΔE and state of the new site h_0 for brevity.

h_0	(h_1,h_2,h_3)	ΔE	$P_n^{(r)}(\Delta E)$
a	(a,a,a)	0	$\frac{1}{3}p_a^3$
	$(a,a,b),(a,a,c)$	1	$\frac{1}{3}\left(3p_a^2p_b+3p_a^2p_c\right)$
	$(a,b,b),(a,b,c),(a,c,c)$	2	$\frac{1}{3}\left(3p_ap_b^2+3p_ap_c^2+6p_ap_bp_c\right)$
	$(b,b,b),(b,b,c),(b,c,c),(c,c,c)$	3	$\frac{1}{3}\left(p_b^3+p_c^3+3p_bp_c^2+3p_cp_b^2\right)$
b	(b,b,b)	0	$\frac{1}{3}p_b^3$
	$(b,b,a),(b,b,c)$	1	$\frac{1}{3}\left(3p_b^2p_a+3p_b^2p_c\right)$
	$(b,a,a),(b,a,c),(b,c,c)$	2	$\frac{1}{3}\left(3p_bp_a^2+3p_bp_c^2+6p_bp_ap_c\right)$
	$(a,a,a),(a,a,c),(a,c,c),(c,c,c)$	3	$\frac{1}{3}\left(p_a^3+p_c^3+3p_ap_c^2+3p_cp_a^2\right)$
c	(c,c,c)	0	$\frac{1}{3}p_c^3$
	$(c,c,b),(c,c,a)$	1	$\frac{1}{3}\left(3p_c^2p_b+3p_c^2p_a\right)$
	$(c,b,b),(c,b,a),(c,a,a)$	2	$\frac{1}{3}\left(3p_cp_b^2+3p_cp_a^2+6p_cp_bp_a\right)$
	$(b,b,b),(b,b,a),(b,a,a),(a,a,a)$	3	$\frac{1}{3}\left(p_b^3+p_a^3+3p_bp_a^2+3p_ap_b^2\right)$

Table 4.3: Possible configurations (h_1,h_2,h_3), energy shifts ΔE and probabilities for the different states n which the new site, $h_0$can be. The overall factor of $\frac{1}{3}$ corresponds to the *a priori* probability that the new site is in state a and the remaining numeric multipliers stem from combinatorics.

Plugging all terms in to the second integral in the expression for $\mathcal{F}(y)$, and simplifying, we obtain:

$$\begin{aligned}
\int \mathrm{d}\Delta E P_n^{(r)}(\Delta E)\, e^{-y\Delta E} &= \tfrac{1}{3}\left\{\left(p_a^2+p_b^2+p_c^2\right)+3\left[p_a^2\left(p_b+p_c\right)+p_b^2\left(p_a+p_c\right)+p_c^2\left(p_a+p_b\right)\right]e^{-y}\right.\\
&+ 3\left[p_a\left(p_b+p_c\right)^2+p_b\left(p_a+p_c\right)^2+p_c\left(p_a+p_b\right)^2\right]e^{-2y}\\
&+ \left.\left[\left(p_b+p_c\right)^3+\left(p_a+p_c\right)^3+\left(p_a+p_b\right)^3\right]e^{-3y}\right\}\\
&= \tfrac{1}{3}\left[p_a+\left(p_b+p_c\right)e^{-y}\right]^3+\tfrac{1}{3}\left[p_b+\left(p_a+p_c\right)e^{-y}\right]^3\\
&\quad +\tfrac{1}{3}\left[p_c+\left(p_a+p_b\right)e^{-y}\right]^3
\end{aligned} \tag{4.8}$$

And for general k, we have:

$$\begin{aligned}
\int \mathrm{d}\Delta E P_n^{(r)}(\Delta E)\, e^{-y\Delta E} &= \tfrac{1}{3}\left[p_a+\left(p_b+p_c\right)e^{-y}\right]^k+\tfrac{1}{3}\left[p_b+\left(p_a+p_c\right)e^{-y}\right]^k\\
&+ \tfrac{1}{3}\left[p_c+\left(p_a+p_b\right)e^{-y}\right]^k
\end{aligned} \tag{4.9}$$

4.2.4 Entropy

Combining all the previous results, we have that the free Monte-Carlo the system is, according to Eq. 4.6 and for general k:

$$\begin{aligned}
\mathcal{F}(y) &= -\ln\left[\left(p_a^2+p_b^2+p_c^2\right)+2\left(p_ap_b+p_ap_c+p_bp_c\right)e^{-y}\right]\\
&+ \tfrac{r}{2}\ln\left[\tfrac{1}{3}\left\{\left[p_a+\left(p_b+p_c\right)e^{-y}\right]^k+\left[p_b+\left(p_a+p_c\right)e^{-y}\right]^k+\left[p_c+\left(p_a+p_b\right)e^{-y}\right]^k\right\}\right]
\end{aligned} \tag{4.10}$$

where the three densities p_a, p_b and p_c are given by:

$$\begin{aligned}
p_a &= \tfrac{1}{Z}\tfrac{1}{3}\left\{p_a^2+2p_a\left(p_b+p_c\right)e^{-y}+\left(p_b+p_c\right)^2e^{-2y}\right\}\\
p_b &= \tfrac{1}{Z}\tfrac{1}{3}\left\{p_b^2+2p_b\left(p_a+p_c\right)e^{-y}+\left(p_a+p_c\right)^2e^{-2y}\right\}\\
p_c &= \tfrac{1}{Z}\tfrac{1}{3}\left\{p_c^2+2p_c\left(p_a+p_b\right)e^{-y}+\left(p_a+p_b\right)^2e^{-2y}\right\}\\
Z &= \tfrac{1}{3}\left\{\left[p_a+\left(p_b+p_c\right)e^{-y}\right]^2+\left[p_b+\left(p_a+p_c\right)e^{-y}\right]^2+\left[p_c+\left(p_a+p_b\right)e^{-y}\right]^2\right\}
\end{aligned} \tag{4.11}$$

Finally, the systems entropy as a function of y, $L(y)$ can be calculated using the slope transform[44] of $\mathcal{F}$:

$$L(y) = ey-\mathcal{F}(y) \tag{4.12}$$

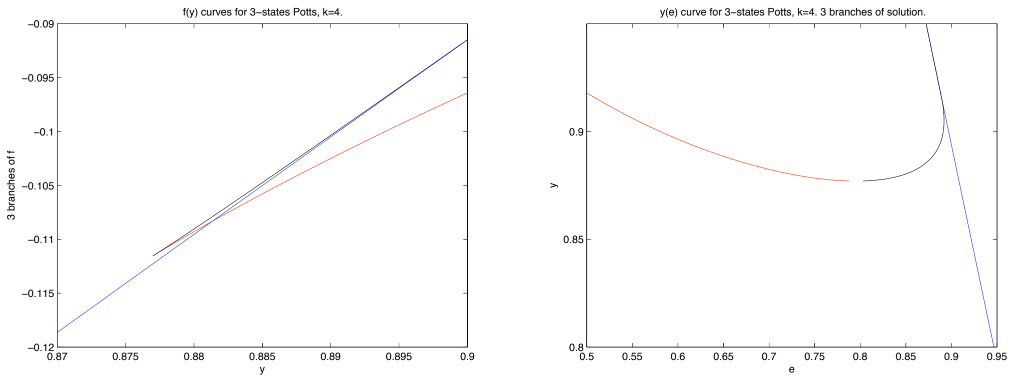

Figure 4.1: left) Three solution branches of the entropy. right) The three corresponding branches for $y\left(e\right)$.

where:

$$e \equiv \frac{\partial \mathcal{F}}{\partial y} \tag{4.13}$$

can easily be calculated numerically using finite differences. This is the final, implicit, solution for the physical observable, $L\left(e\right)$, the entropy as a function of the energy, e. It also helps us identify

$$y \equiv \frac{\partial L}{\partial e} \equiv \beta\left(e\right) \tag{4.14}$$

the inverse temperature. A simply procedure will allow us to plot $L\left(e\right)$. Starting with values of y, calculate $F\left(y\right)$. Differentiate it numerically, to obtain $e\left(y\right)$, and use it to calculate $L\left(y\right)$, which can now be plotted as a function of e. In principle, one should invert $y\left(e\right)$ and apply it to $L\left(y\right)$ to obtain $L\left(e\right) \equiv L\left(y\left(e\right)\right)$, but, in this case, a multi valued function of the energy, so we must divide the solution space, numerically, in several branches and analyze them separately. In fig. 4.1, we plot the 3 different solution branches for the entropy $L\left(e\right)$, and the inverse temperature $y\left(e\right)$.

We also note that multiple values of the energy compatible with the same microcanonical temperature imply that the entropy curve must be non-concave in that region, resulting in negative specific heat, the clear fingerprint of a system with ensemble inequivalence.

4.3 Comparison with numerical simulations

In this section we compare the analytical solution with the results obtained through numerical simulations. Microcanonical simulations were performed using Creutz [39] dynamics. During which, a fictitious "demon" is introduced, carrying an energy e_{demon}. At each step, a spin flip in the system is attempted, and the corresponding energy change δE is computed. If $\delta E < 0$, the move is accepted; if $\delta E > 0$, the move is accepted only if $e_{demon} \geq \delta E$. In both cases e_{demon} is then updated so that the total energy $E + e_{demon}$ is kept constant; the Monte-Carlo the system E is then constant up to a $O(1/N)$. For long run times, the demon's energy reaches an exponential distribution $P(e_{demon} = e) \propto \exp(-e/T_\mu)$, from where one can compute the corresponding microcanonical temperature $T_\mu = 1/\beta_{mu}$ of our system:

$$\beta_\mu = \log \left[1 + \frac{1}{\langle e_{demon} \rangle} \right] \ . \tag{4.15}$$

Results of the Creutz dynamics are plotted on Fig. 4.2 and compared with the analytical solution of the previous section. The agreement between the two is very good, with the β vs energy curve clearly showing a region of negative specific heat.

Finally, we performed canonical Metropolis[38] simulations and calculated the average energy in the temperature range where our results predict ensemble inequivalence. As expected, the canonical caloric curve obeys Maxwell's construction and clearly "jumps over" the region where the specific heat is negative.

4.4 Conclusion and perspectives

We have presented a complete canonical and microcanonical solution of the 3-states Potts model on k-regular random graphs, and shown that this toy model displays ensemble inequivalence.

There is little doubt that this result should generically apply to models on different types of random graphs, such as Erdös-Rényi ones, among others. We also expect to observe ensemble inequivalence on small world networks, since in these systems, the presence of random long-range links should prevent the system from separating in two different phases.

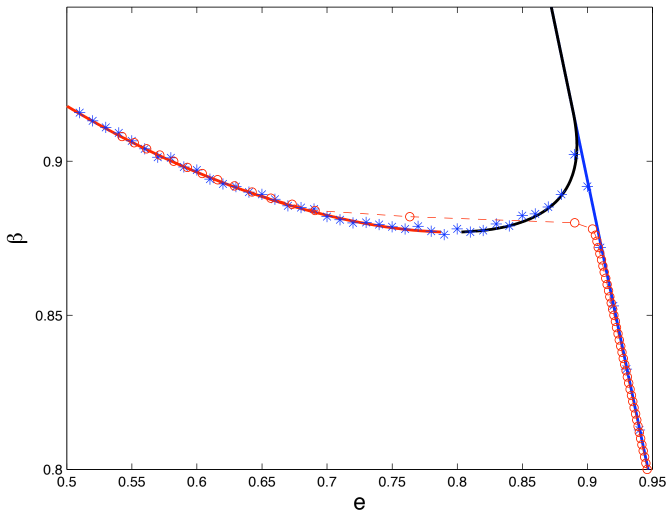

Figure 4.2: Comparison for the caloric curve $\beta(e)$ between the analytical solution (solid lines), the Creutz dynamics results (stars), and the Metropolis simulations (circles) for $k = 4$. The Creutz simulations were performed on networks with $N = 40000$ sites, for 10^8 "Creutz steps", and the results were averaged over 20 network realizations. The Metropolis results were obtained using 50 different networks with $N = 10000$ nodes, by performing 10^{10} Monte-Carlo steps. In both cases, the size of the error bars is comparable to the size of the symbols.

Beyond the inequivalence between microcanonical and canonical statistical ensemble, non concave large deviation functions should be expected for some properties on random graphs. Fig. 4 of [45] gives an example of this. The present work provides an example where the Large Deviation Cavity method allows to deal with such a situation, and to compute the non concave part of the large deviation function.

4.5 Publications

[46] J. Barré and B. Gonçalves, Ensemble inequivalence in random graphs, *Physica A*, 386:212, (2007)

Chapter 5

Topological Non-Connectivity Threshold

> ”He who controls magnetism controls the world”
>
> – Dick Tracy

The last few of decades have witnessed a prodigious development in our ability to easily manufacture a wide array of structures on the level of individual atoms (as shown in Fig. 5.1), with which we can probe and reevaluate the validity of some well known classical results obtained on the macroscopic level. These ultra thin objects, due to their small size and consequent anisotropy, display a whole new spectrum of phenomena that can have important consequences in the development of the emerging field of nanotechnology which

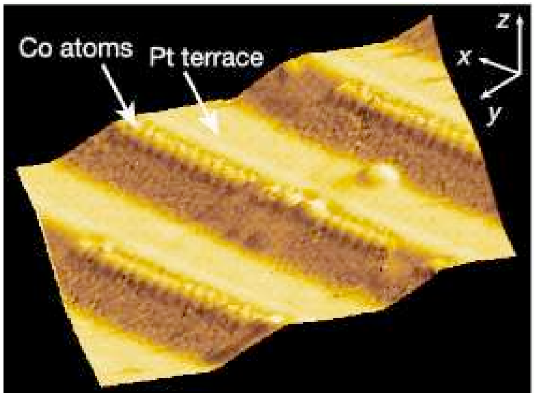

Figure 5.1: Scanning Tunneling Microscopy image of *Co* mono atomic chains along the *Pt* (997) step edges. From [3].

holds the key of future improvements in the computation, data storage and medical fields.

In particular, magnetic properties of linear chains that are just a few atoms long have received a great deal of interest. Specially, since it has been found experimentally by Gambardella *et al* [3] in 2003 that this system can exhibit Ferromagnetic behavior, even though the standard theory states that there should be no Ferromagnetic behavior in $1D$ systems with only short range interactions. It has also been found that small systems with long-range interactions can display negative specific heat[47] and non-ergodic behavior [48].

One of the possible ways in which ergodicity can be broken is by dividing the phase space of the system in to two or more regions and to restrict the dynamics in such a way as to always keep the system within a given region. Borgonovi *et al*[48, 49, 50] recently considered a model of Heisenberg spins with an adjustable interaction range, which presents ergodicity breaking for any N, is described by the following Hamiltonian:

$$H = -\frac{J}{2}\sum_{i\neq j}^{N}\left(\frac{S_i^z S_j^z}{r_{ij}^{\alpha}} + \eta_y\frac{S_i^y S_j^y}{r_{ij}^{\alpha}} + \eta_x\frac{S_i^x S_j^x}{r_{ij}^{\alpha}}\right), \tag{5.1}$$

where S_i^x, S_i^y, S_i^z are the spin components, assumed to vary continuously; $i, j = 1, ...N$ label the spin positions on a suitable lattice of spatial dimension d, and r_{ij} is the inter-spin spatial separation. Each spin satisfies $|\vec{S}| = 1$ and $0 \leq \alpha < \infty$ parameterizes the range of interactions: decreasing range for increasing α, so that $\alpha = 0$ corresponds to an all-to-all interacting model (close to phenomenological anisotropic models), while $\alpha = \infty$ refers to a nearest neighbor interacting spin model, similarly to the model described in Chap. 2. $|\eta_{x,y}| < 1$ parameterizes the degree of anisotropy and $J > 0$ indicates that the ground state is ferromagnetic, with an preferential direction of the magnetization in the ground state along the Z axis (easy-axis). Of course, this is not the simplest Hamiltonian with an easy axis: putting $\eta_x = \eta_y = 0$ will lead to the same conclusion. Nevertheless, in such a case the model becomes integrable and some results become model-dependent. Moreover, as shown in [48], the presence of both anisotropies (along X and Y axis) does not suppress the TNT: for this reason, without losing generality, we may set $\eta_y = 0$. As for the anisotropic parameter η_x, it could represent as well, a ferromagnetic $\eta_x > 0$ or an antiferromagnetic $\eta_x < 0$ coupling. Defining, $\eta = -\eta_x$ we arrive at the following Hamiltonian:

$$H = -\frac{J}{2}\left(\sum_{i\neq j}^{N} \frac{S_i^z S_j^z}{r_{ij}^{\alpha}} - \eta \frac{S_i^x S_j^x}{r_{ij}^{\alpha}}\right), \tag{5.2}$$

which has been already considered in previous papers [48, 50], and where the constant $J > 0$ has been added in order to fix the scale of time and to describe the model as ferromagnetic. Let us also notice that, for $\eta = -1$ this Hamiltonian is invariant for rotations about the Z axis so that there is no a preferential direction of the magnetization in ground state (absence of easy axis).

Here we review previous results about the TNT obtained for any dimension d of the system, even if in the following sections we will focus on the $d = 1$ case.

The minimum energy configuration, with energy E_{min}, is attained when all spins are aligned along the $Z-$axis[49] which defines implicitly the easy axis of magnetization. While it is quite simple to derive conditions for the non existence of an easy axis of magnetization (for instance it is sufficient that the system is invariant under any rotation), it is not so obvious to write down conditions under which the existence of an easy axis is guaranteed. For instance the presence of a rotational symmetry axis about which the Hamiltonian is invariant is not enough to give an easy axis, as the Hamiltonian $H = -\sum_{i,j} S_i^x S_j^x + S_i^y S_j^y$ shows.

The phase space for $E = E_{min}$ contains only two spin configurations, parallel or antiparallel to the $Z-$axis. Therefore, the phase space at the minimal energy is disconnected, due to the uniaxial anisotropy and it consists of two points only. We may ask now when and whether at a higher energy the constant energy surface is connected. To this purpose, let us define the TNT energy E_{tnt} as the minimum energy compatible with the constraint of zero magnetization along the easy axis of magnetization (hereafter we call m the magnetization along the easy axis)

$$E_{tnt} = \text{Min}\left[H \;\middle|\; m \equiv \sum_i S_i^z / N = 0 \right]. \tag{5.3}$$

If $E_{tnt} > E_{min}$ the phase space will be disconnected for all energies $E < E_{tnt}$. We call this situation Topological Non-connection, and, as will become clear in a moment, its physical (dynamical as well as statistical) consequences are rather interesting. Indeed, since below the TNT the phase space is disconnected, no energy conserving dynamics can bring the system from a configuration with $m > 0$ to a configuration with $m < 0$, thus indicating an ergodicity breaking (impossibility to visit the whole energy surface).

A useful quantity measuring how large the disconnected energy region is, compared to the total energy range, can be introduced [49]:

$$r = \frac{E_{tnt} - E_{min}}{|E_{min}|}. \tag{5.4}$$

In [49] it has also been shown that the disconnection ratio r, for $N \to \infty$,

$$r \to \begin{cases} 0 & \text{for} \quad \alpha \geq d \\ const \neq 0 & \text{for} \quad \alpha < d, \end{cases} \tag{5.5}$$

where d is the dimension of the embedding lattice. Since this point has not been remarked in Ref. [49], let us stress here that the existence of a phase transition for $\alpha < d$ can be inferred from the finiteness of r in the thermodynamic limit. Indeed, for long range systems, in order to define the thermodynamic limit it is convenient to make the energy extensive. This can be achieved by multiplying the Hamiltonian by $N/|E_{min}|$. If we define the energy per particle, $e = E/N$, we can write:

$$e_{tnt} - e_{min} = \frac{N}{|E_{min}|}\left(\frac{E_{tnt}}{N} - \frac{E_{min}}{N}\right) \equiv r. \tag{5.6}$$

Since below e_{tnt} the most probable magnetization is for sure different from zero, then the specific energy at which the most probable magnetization is zero will be greater than the minimum energy in the thermodynamic limit, thus implying a phase transition. On the other hand, let us also notice that when $r \to 0$ neither the existence nor the absence of a phase transition can be deduced.

An estimate for the TNT was also given for $\alpha < d$ (long range) and large N. More precisely, it can be shown that [49]:

$$E_{tnt} \approx \begin{cases} 4E'_{min} - E_{min} & \text{for} \qquad \eta \geq \eta_{cr} \\ \\ -\eta E_{min} & \text{for} \qquad \eta < \eta_{cr}, \end{cases} \tag{5.7}$$

where

$$\eta_{cr} \simeq 1 - 2^{\alpha/d} < 0, \tag{5.8}$$

and E'_{min} is the minimal energy for a system of $N/2$ spins. For $\eta > \eta_{cr}$ the TNT is given by the minimum of the second term in Eq. (5.2) under the constraint $m = 0$, while for $\eta < \eta_{cr}$ it is given by the minimum, under the same constraint, of the first term in Eq. (5.2). For $\eta < 0$ and finite N, there is a competition between the two different TNTs, therefore, in what follows, we will fix the anisotropy parameter $\eta = 1$, for which we have ferromagnetic coupling along the Z direction and antiferromagnetic coupling along the X direction, however, this choice does not affect the generality of our results.

5.1 Density of States

The density of states (DOS) for a Mean-Field approximated Hamiltonian can be computed analytically, using large deviations techniques [51]. In particular we will show that $\rho(E) \simeq (E - E_{min})^N$ for E close to E_{min}. We will also give numerical evidence that this law still constitutes an excellent approximation of the full Hamiltonian (5.2) and for generic power law interaction $\alpha \neq 0$.

Let us consider the following Mean-Field Hamiltonian:

$$H_{mf} = -\frac{J}{N}\left(\sum_k S_k^z\right)^2, \tag{5.9}$$

which can be considered a Mean-Field approximation of the Hamiltonian (5.2), for low energy and $\alpha = 0$. Defining

$$m_z = \frac{1}{N}\sum_k S_k^z \tag{5.10}$$

and $e = H_{mf}/N$, Eq. (5.9) can be rewritten as

$$e = -Jm_z^2. \tag{5.11}$$

Let us also assume that S_k^z are random variables uniformly distributed in $[-1, 1]$.

We can compute the entropy per particle as a function of m_z using Cramer's theorem [51], so that we have:

$$s(m_z) = -sup_\lambda \left[\lambda m_z - \ln\psi(\lambda)\right], \tag{5.12}$$

where

$$\psi(\lambda) = \langle e^{\lambda S^z}\rangle = \frac{1}{2}\int_{-1}^{1} e^{\lambda S^z} dS^z = \frac{e^\lambda - e^{-\lambda}}{2\lambda}. \tag{5.13}$$

Taking the *sup* in Eq. (5.12) we get:

$$\frac{\psi'(\lambda)}{\psi(\lambda)} = m_z, \tag{5.14}$$

which defines λ as a function of m_z. It is easy to see that for $m_z \sim 1$ then $\lambda \to \infty$ (we could as well consider $m_z \sim -1$ of course, and the result would be the same) so we restrict our considerations to $|m_z| \simeq 1$. Simplifying the expression of ψ we have

$$\psi(\lambda) = \frac{e^\lambda}{2\lambda}, \tag{5.15}$$

and inverting Eq. (5.14):

$$\lambda = \frac{1}{1-m_z} \equiv \frac{1}{\delta}. \tag{5.16}$$

From Eq. (5.12) we obtain:

$$s(\delta) = \ln\delta + const. = \ln\left(\frac{e - e_{min}}{-e_{min}}\right) + const., \tag{5.17}$$

since

$$e = -Jm_z^2 \simeq -J(1-2\delta) = e_{min} + 2\delta J, \tag{5.18}$$

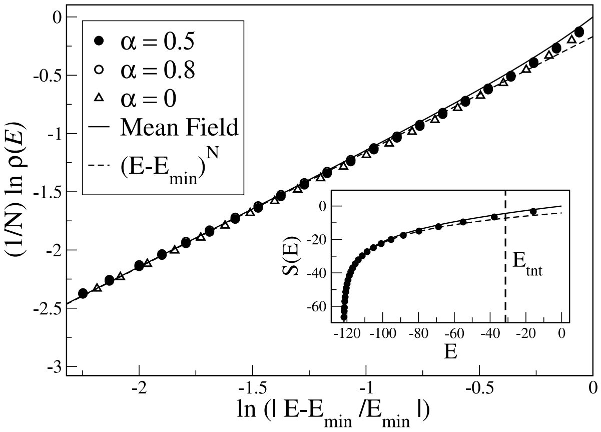

Figure 5.2: The specific entropy *vs* energy, obtained numerically for $N = 24$ and different α values (symbols indicated in the legend) is compared with that of the Mean Field Hamiltonian (full curve) and with the power law (dashed line), see Eq. (5.19). In the inset the entropy is shown *vs* the energy for the case $N = 24$, $\alpha = 0.5$.

and $e_{min} = -J$. From that we immediately have that at low energy,

$$\rho(E) \simeq (E - E_{min})^N. \tag{5.19}$$

The next leading order in δ can be calculated from

$$e = -Jm_z^2 = -J(1-\delta)^2, \tag{5.20}$$

so that

$$\delta = 1 - \sqrt{-\frac{e}{J}}, \tag{5.21}$$

and

$$s(e) = \ln\left(1 - \sqrt{\frac{e}{e_{min}}}\right) + const. \tag{5.22}$$

It is immediate to see that Eq. (5.22) gives Eq. (5.17) for $e \simeq e_{min}$. We compared this analytical result for the Mean-Field model with the DOS computed numerically for the full Hamiltonian (5.2) and different α values in Fig. 5.2. The DOS has been calculated using a modified Metropolis algorithm introduced in [41]. The idea behind is performing a random walk in phase space within an energy range defined by the system temperature. The probability $P(E,T)$ of visiting a configuration with energy E and temperature T, obtained keeping a histogram of the energy values found during a Metropolis run, is related to the DOS through the Boltzmann factor $\exp(-E/T)$ (here is $k_B = 1$). That provides us a conceptually simple way of determining the DOS. However, due to finite run time, $P(E,T)$ will only contain information near $\langle E \rangle$ and we must combine the results for runs at different temperatures to obtain the complete DOS over the entire energy range.

As one can see in Fig. 5.2 the entropy per particle, in the long range case, is almost independent of the range of the interaction, also confirming a result obtained in [52]. Moreover, the theoretical approach gives a very good approximation of the entropy per particle at low energy. When the energy is increased, the first term in the full Hamiltonian (5.2) becomes important and some deviations appear (see for instance the upper right corner in Fig. 5.2). Needless to say the excellent agreement confirms the power law behavior for the DOS, Eq. (5.19), even for energy values sufficiently high. As an example in the inset of Fig. 5.2, we can see that the power law expression (5.19) holds up to E_{tnt}.

5.2 Probability of Zero Magnetization

Due to the disconnection, below E_{tnt} the dynamical time of magnetic reversal is infinite while above and close to the energy threshold (for chaotic systems), it was found to diverge as a power law [50]:

$$\tau \sim \frac{1}{(E - E_{tnt})^{\gamma}} \sim e^{\Delta S} = \frac{P_{max}}{P_0}, \tag{5.23}$$

where ΔS is the entropic barrier between the most likely magnetic states. Here, $P_E(m)$ is the probability distribution of the magnetization m at fixed energy E, so that

$$P_{max} \; = \; \mathrm{Max}_m \; P_E(m), \tag{5.24}$$

and $P_0 = P_E(m = 0)$.

The divergence found in [50] also shows that the phase space becomes connected at E_{tnt}, a non trivial result, which cannot be deduced from the true existence of E_{tnt}.

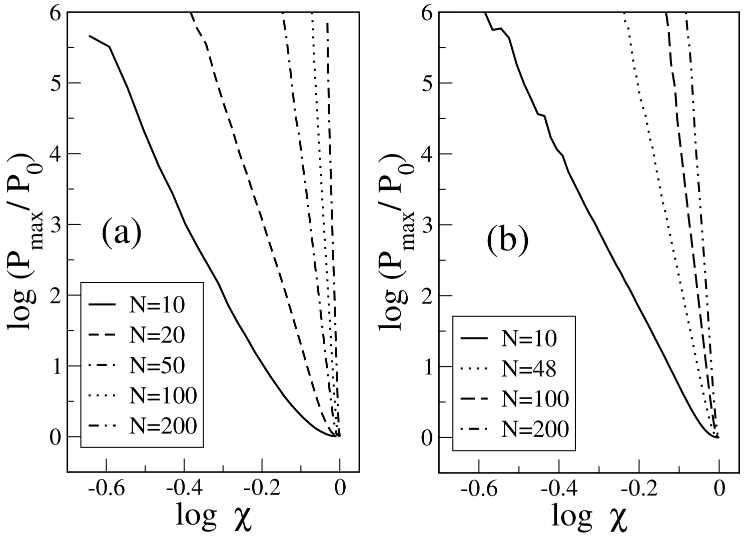

Figure 5.3: $\log P_{max}/P_0$ *vs* $\log\chi$ for different N values as indicated in the legend and (a) $\alpha = 0.5$; (b) $\alpha = 1$.

Also, for all-to-all interacting spins ($\alpha = 0$), the exponent $\gamma \propto N$. This is related with the extensivity of the entropy $S(E, m) = \ln P_E(m)$ (here we set $k_B = 1$) and explain the huge metastability of such states even for small systems (say $N \sim 100$) and not necessarily close to the threshold E_{tnt}.

We numerically checked that, even for other power-law decreasing potentials in the long-range case $\alpha < d$,

- a power law divergence at E_{tnt} still occurs as given by Eq. (5.23);
- reversal time is still proportional to P_{max}/P_0 (same equation);
- the exponent $\gamma = N$.

In order to do that we computed $P_E(m = 0)$ and P_{max} for different systems using the Wang-Landau algorithm [53].

In Fig. 5.3 we show the power law divergence of P_{max}/P_0 for different α and N values. In order to improve the presentation we choose as a variable on the $X-$axis

$$\chi = \frac{E - E_{tnt}}{E_{stat} - E_{tnt}}, \tag{5.25}$$

where E_{stat} has been defined as the energy at which $P_{max} = P_0$ (that is when the probability

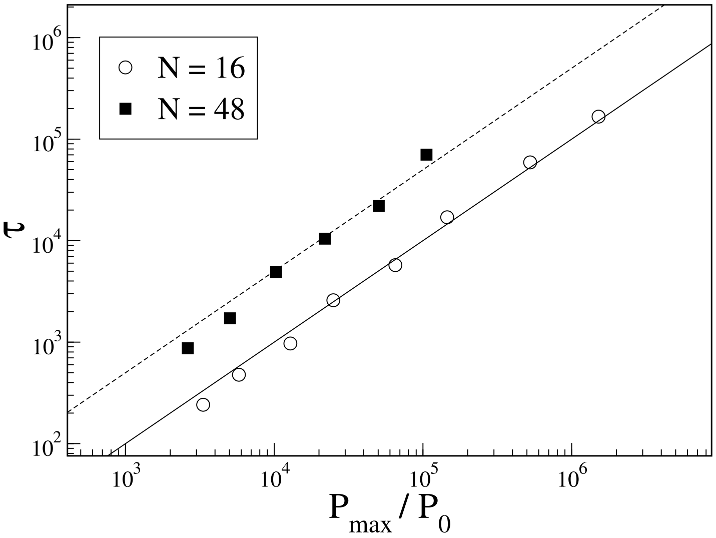

Figure 5.4: Dynamical reversal time τ *vs* the statistical one P_{max}/P_0 for $\alpha = 1$ and different N values as indicated in the legend. Straight lines are $\tau = (1/k)P_{max}/P_0$ with $k = 10$ for $N = 16$ and $k = 2$ for $N = 48$. They have been drawn with the only purpose to guide the eye showing the proportionality between the two quantities over 3 orders of magnitude.

distribution of the magnetization has a single peak). That way all curves have a common origin.

When the isolated system has a chaotic dynamics we computed the magnetic reversal time from the direct integration of the equations of motion and we compare it with the "statistical" time P_{max}/P_0 as given by Eq. (5.23). We show this comparison in Fig. 5.4 where each point on the graph has a X coordinate P_{max}/P_0 and a Y coordinate τ. The straight lines indicated proportionality over 3 orders of magnitude.

The linear dependence $\gamma = N$ found in [50] for the case $\alpha = 0$ holds for generic $\alpha < 1$ too.

In Fig. 5.5 we show the results of our numerical simulation for $\alpha = 0.5, 0.9, 1$. Each point γ, at fixed N has been obtained computing the statistical reversal time for different energies, as plot in Fig. 5.3, using the power law (5.23). Assuming a power law dependence $\gamma \propto N^\sigma$ we have found $\sigma \approx 1$ (within numerical errors) for all cases $\alpha \ll 1$ (we show for simplicity only the case $\alpha = 0.5$ in Fig. 5.5).

On the other hand, for $\alpha \sim 1$ we have numerical evidence of a slower dependence on N: $\gamma \sim N^\sigma$ with $\sigma < 1$. In the same Fig. 5.5 we show for sake of comparison the critical case $\alpha = d = 1$ where $\sigma = 0.78(2)$, and the close-to-critical case $\alpha = 0.9$ where we have found

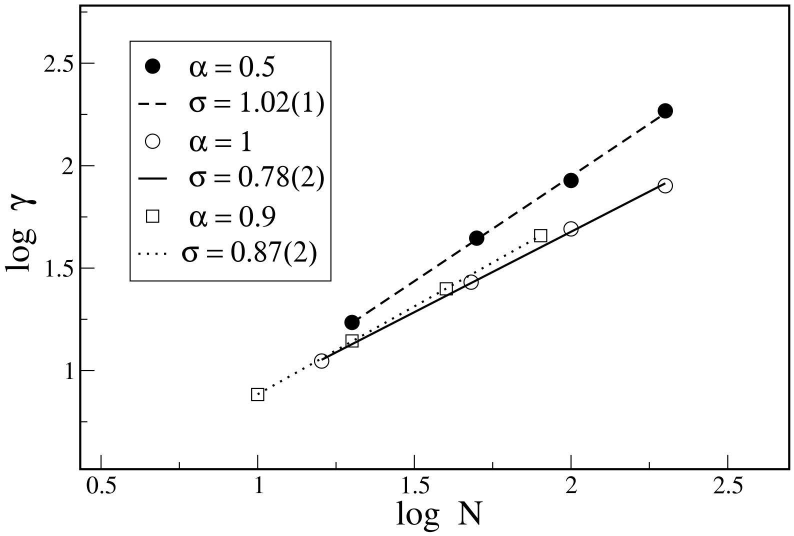

Figure 5.5: $\log\gamma$ as a function of $\log N$ for different α values. Full circles stand for $\alpha = 0.5$ and dashed line is the linear fitting with slope 1.02(1). Open circles is the critical case $\alpha = d = 1$ and full line is the linear fit with slope 0.78(2). Open squares are for $\alpha = 0.9$. Standard fit procedure gives $\sigma = 0.87$ thus signaling the presence of a smooth transition at the point $\alpha = 1$ for finite N.

$\sigma = 0.87(2)$. Even if these results indicate that the simple linear relation $\gamma \propto N$ is valid for long-range interacting systems only, care should be used to extend the results of the case $\gamma = d = 1$ to large N since finite N effects are huge in this case. Numerical evidence for $\sigma < 1$ has also been found in the short range case ($\alpha \gg 1$) but it will be discussed elsewhere.

The next obvious step in the analysis of this model is to put it in contact with a heat bath. In this case, we expect the thermal fluctuations to play a major role, since through them the system will be able to reach states with an energy range, which might be large enough to allow it to reverse the magnetization, thus making the TNT irrelevant.

Let us define $P_0(E)$ to be the probability of having a state with 0 magnetization and energy E, and, by definition, we have that $P_0(E < E_{TNT}) \equiv 0$. As we know from standard statistical physics, if we know the microcanonical $P_0(E)$, we can easily calculate its canonical counterpart, $P_0(T)$, the probability of the system having zero magnetization when in thermal equilibrium with a heat bath at temperature T, using:

$$P_0(T) = \frac{\int_{E_{TNT}} P_0(E)\, g(E)\, e^{-\beta E} \mathrm{d}E}{\int_{E_{TNT}} g(E)\, e^{-\beta E} \mathrm{d}E} \tag{5.26}$$

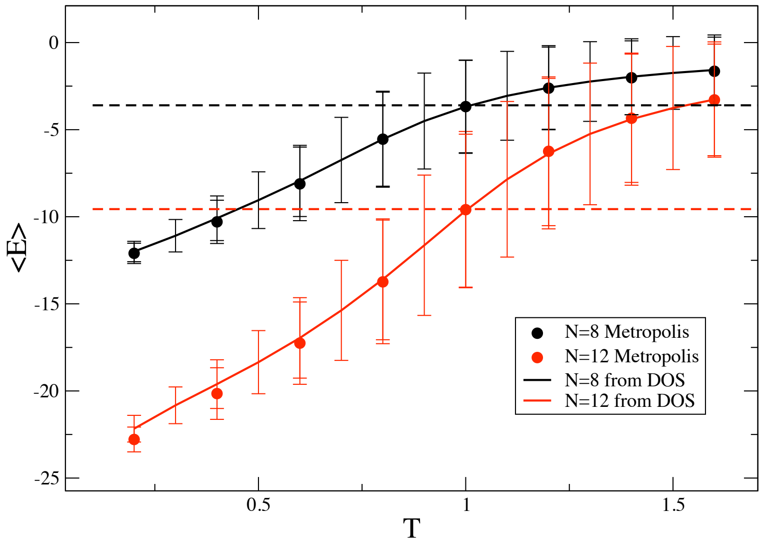

Figure 5.6: Average energy as a function of temperature. The horizontal lines mark E_{TNT} and the ”error bars” represent the standard deviation of the energy distribution $P(E)$.

where $g(E)$ is the density of states. Our task is now two fold, since on one hand we must calculate $g(E)$, and on the other, we must also calculate $P_0(E)$. Due to the numerical difficulties of determining the density of states, we compared the average energy as a function of temperate calculated using:

$$\langle E(T)\rangle = \frac{\int_{E_{TNT}} Eg(E)\, e^{-\beta E} \mathrm{d}E}{\int_{E_{TNT}} g(E)\, e^{-\beta E} \mathrm{d}E} \tag{5.27}$$

and the direct result of measuring it in our Metropolis runs. As shown in Fig.5.6 the two approaches agree very well with one another, which gives us a high degree of confidence that our approach so far is correct. This plot also gives us the possibility of defining T_{TNT} as the temperature at which the average energy is equal to E_{TNT}.

5.3 Magnetization Reversal Times

Since the TNT has been introduced for isolated systems, question arises if and how it can be defined when the system is in contact with a thermal bath. From the theoretical point of view we might expect that due to thermal noise the magnetization will be able, soon or later, to change its sign at any temperature T, thus suppressing the ergodicity breaking found in isolated systems. Therefore, strictly speaking, a critical temperature below which the phase space is topologically disconnected for open finite systems does not exist.

Nevertheless we are here interested in more practical questions, for instance: Will the energy threshold E_{tnt} still determine the magnetic reversal time in presence of temperature as it does in isolated chaotic systems? Can we predict the dependence of reversal time from temperature or any other system parameters, like the number of particles?

Since the system is in contact with a thermal bath we may consider it as a member of a canonical ensemble. We may properly define the probability density to have a certain energy value E at the temperature T: $P(E, T)$. Considering all members of the ensemble as independent objects we may guess that when the average energy $\langle E \rangle$ is much less than E_{tnt} and the probability density $P(E, T)$ sufficiently peaked around its average value, the majority of the members of the ensemble will not cross the barrier, or at least, the probability of crossing it will be small. On the other hand for mean energy $\langle E \rangle$ on the order of E_{tnt} each member will be allowed to jump, with a time essentially given by the microcanonical expression Eq. (5.23).

Let us further assume, following the standard fluctuation theory [54, 55], that the magnetic reversal times between states with opposite magnetization are determined by the free energy barrier ΔA between states at the most probable magnetization and states with zero magnetization:

$$\tau_T \propto \exp\left(\frac{\Delta A}{T}\right) = \frac{\text{Max}_m[P_T(m)]}{P_T(m=0)}, \tag{5.28}$$

where

$$P_T(m) = \exp[-A(T, m)/T], \tag{5.29}$$

is the probability density to have magnetization m at the temperature T. Since $\text{Max}_m[P_T(m)]$ is usually a slow varying function of the temperature, we can write

$$\tau_T \propto \frac{1}{P_T(m=0)}. \tag{5.30}$$

The crucial point now, is to obtain such value using the microcanonical results obtained in the previous Section, namely:

Algorithm 3 Magnetization Reversal Times

1. Thermalize the system according to steps 1–5 of Alg. 2.
2. Evolve the system until the magnetization changes sign, while counting the number of spin flips.
3. Repeat steps 1–2 a large number of times
4. Compute the average number of spin flips required to reverse the sign of the magnetization.

$$P_T(m=0) = \frac{1}{Z_T} \int P_E(m=0)\ e^{-E/T}\ \rho(E)\ dE, \tag{5.31}$$

where $\rho(E)$ is the density of states and

$$Z_T = \int\ e^{-E/T}\ \rho(E)\ dE, \tag{5.32}$$

is the partition function. Since $P_E(m=0) = 0$ for $E < E_{tnt}$, the ergodicity breaking acts as a cut-off energy of the integral (5.31).

In order to verify that Eq. (5.31) actually gives the magnetic reversal time, we simulated the dynamics of a spin system in contact with a thermal bath in two different ways, the Metropolis algorithm [38], and using the stochastic differential equations of the Langevin type as suggested in [56].

The Metropolis approach uses a small modification of Alg. 2, where we thermalize the system and then count the number of spin flips until the magnetization changes sign. We repeat this procedure multiple times, with a thermalization period between each measurement to ensure that consecutive measurements are decorrelated. The complete process is described in Alg. 3.

It should be noted that while these different ways of calculating the magnetization reversal time are correct only up to an arbitrary constant, they will necessarily give us the correct functional dependence on the temperature. Our results are shown on Fig. 5.7. Both approaches agree except for higher temperatures where the approximation used in the derivation of Eq. 5.51 starts to break down.

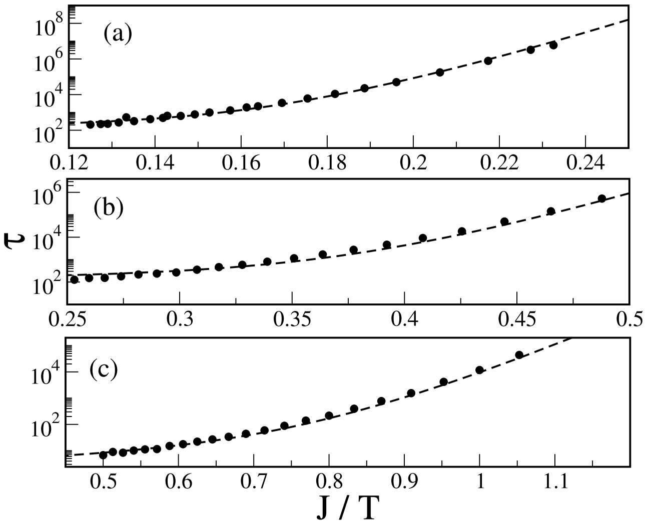

Figure 5.7: Average reversal time τ as a function of the rescaled inverse temperature J/T for different interaction range. a) all-to-all $\alpha = 0$, $N = 24$; b) long-range case $\alpha = 0.5$, $N = 24$; c) critical-case, $\alpha = 1$, $N = 20$. Circles are numerical data, dashed line is the integral calculated in Eq. (5.31).

In the Metropolis dynamics the change in the spin direction has been taken at each step completely random on the unit sphere while in the Langevin approach a small dissipation has been added. We checked that the results are independent of such dissipation and that the two approaches give the same results.

Obviously, both approaches can give directly the distribution function: $P_T(m = 0)$, but we prefer here the direct calculation of the density of states and thus the possibility to get $P_T(m = 0)$ for any temperature[41], with less numerical effort and greater reliability.

Results are shown in Fig. 5.7 where the average reversal time (obtained with the Metropolis dynamics) *vs* the rescaled inverse temperature J/T has been considered for different α and N values as indicated in the caption. As one can see the agreement between the integral (5.31) (dashed line in Fig. 5.7) and the numerical results (full circles) is excellent over many orders of magnitude. It is worth of mention that no parameter fitting, other than a multiplicative constant has been used.

It is also remarkable that a small variation in the temperature scale generates a huge variation of the average time. This signals a strong dependence on the inverse temperature, but, generally speaking, not exponential (one should have in Fig. 5.7 straight lines).

On the other side, the exponential $1/T$ dependence is found in the limit of very low temperature by studying the asymptotic behavior of the integral (5.31).

Using saddle point approximation in Eq. (5.31) we get :

$$P_T(m=0) \simeq P_{E^*}(m=0)\; e^{(\langle E\rangle - E^*)/T}\; e^{S(E^*)-S(\langle E\rangle)}, \tag{5.33}$$

where $\langle E\rangle$ and E^* are given by:

$$\frac{1}{T} = \frac{dS}{dE}(\langle E\rangle) = \frac{ds}{de}(e), \tag{5.34}$$

where $S = \ln\rho$, and

$$\frac{1}{T} = \frac{dS}{dE}(E^*) + \frac{N}{E^* - E_{tnt}} \tag{5.35}$$

where we used $P_{E^*}(m=0) \propto (E^* - E_{tnt})^N$, as above. An approximate expression for (5.34) and (5.35) can be obtained for small temperature. Indeed, using for the entropy the expression Eq. (5.22) obtained in Sect. III, and inverting Eq. (5.34), one obtains:

$$\frac{e}{e_{min}} = \frac{1}{2}\left(1 + \frac{T}{e_{min}} + \sqrt{1 + \frac{2T}{e_{min}}}\right), \tag{5.36}$$

so that, for $T \ll |e_{min}|/2$, we get:

$$< E >= E_{min}\left[1 + \frac{NT}{E_{min}} + O\left(\frac{NT}{E_{min}}\right)^2\right]. \tag{5.37}$$

In the same way Eq. (5.35) can be written as:

$$E^* = E_{tnt} + NT\left[1 + \frac{NT}{\Delta} + O\left(\frac{NT}{\Delta}\right)^2\right], \tag{5.38}$$

where $\Delta = E_{tnt} - E_{min}$ and, for temperature sufficiently low,

$$T \ll T_{cr} = \frac{E_{tnt} - E_{min}}{2N} < \frac{e_{min}}{2}, \tag{5.39}$$

we have

$$E^* = E_{tnt} + NT. \tag{5.40}$$

Eq. (5.33) can be further simplified, using the approximated expressions for $\langle E \rangle$ and E^* obtained above and Eq. (5.19) for the DOS:

$$\begin{aligned} S(E^*) &\simeq N \ \ln(\Delta + NT), \\ S(\langle E \rangle) &\simeq N \ \ln NT, \\ P_{E^*}(m=0) &\simeq (E^* - E_{tnt})^N \simeq (NT)^N. \end{aligned} \tag{5.41}$$

Finally, neglecting terms of order (NT/Δ):

$$\tau_T \sim \frac{1}{P_T(m=0)} \sim \exp\left(\frac{E_{tnt} - E_{min}}{T}\right). \tag{5.42}$$

Even if Eq. (5.42) has been obtained for low temperature ($T \ll T_{cr}$), it should be kept in mind that this is a classical model so that for $T \to 0$, when quantum effects become important, it looses its validity).

The law (5.42) has been checked numerically in Fig. 5.8 where the integral (5.31) has been calculated for very low temperature and compared with the true exponential law. As one can see asymptotically they are very close.

This allows to compute directly the reversal time in presence of a thermal bath at low temperature T without any complicated statistical calculations but the knowledge of the Hamiltonian itself. Moreover, the calculation of both the ground state energy and the Topological Non-connectivity Threshold constitutes a mechanical problem and they can be easily estimated even for complicated models.

Furthermore, it also has some suggestive interpretation. If we consider the path followed by the magnetization, as a random path of a Brownian particle between two potential wells separated by a potential barrier ΔE, according to Kramer's theory [57, 58] the average transition time between the two wells follows the Arrhenius law, $\tau \sim \exp(\Delta E/T)$. Therefore it is clear that the disconnected energy region $\Delta = E_{tnt} - E_{min}$ can be thought of as the

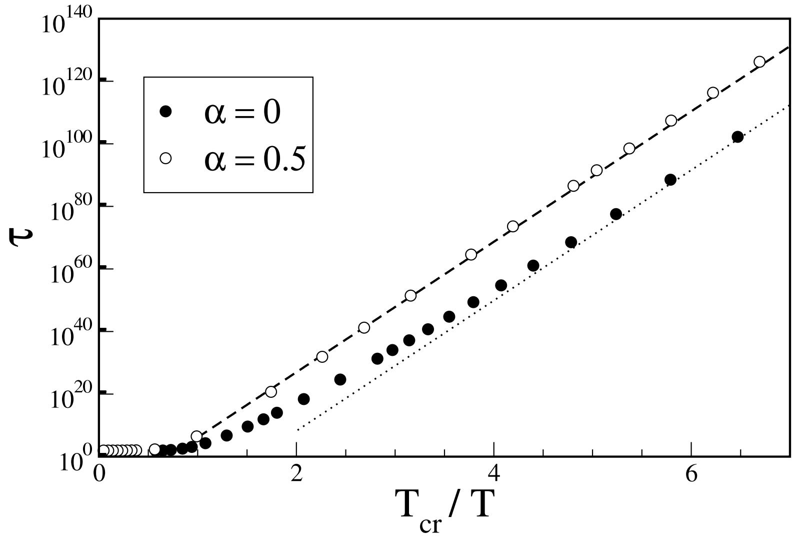

Figure 5.8: Average reversal time τ calculated from Eq. (5.31) as a function of the rescaled temperature T_{cr}/T for different interaction range $\alpha = 0$ (full circles) and $\alpha = 0.5$ (open circles) and $N = 24$. Dashed and dotted lines represent their asymptotic value, as given by Eq. (5.42).

real potential barrier felt by the magnetization.

In same way, the critical temperature T_{cr} has the physical meaning of the specific energy barrier. It is interesting to note that the condition $T \ll T_{cr}$ is not too restrictive, at least for long-range systems. Indeed, taking into account that for large N [49]:

$$T_{cr} = r\ \frac{|E_{min}|}{2N} \simeq \frac{2 - 2^{\alpha}}{2(2-\alpha)(1-\alpha)}\ N^{1-\alpha}, \tag{5.43}$$

only at criticality ($\alpha = 1$) it does not depend on N (and $T_{cr} = \ln 2$), while generally it grows with the number of particles. This is not at all surprising for a long-range systems; indeed, if we make the energy of the system extensive, (multiplying the Hamiltonian by $N/|E_{min}|$), we have $T_{cr} = r/2$, which is finite for any interaction range.

5.4 Power-law corrections to Arrhenius's law

It should be clear from the previous sections that, depending on the particular shape of the density of states and on the interaction range, power law corrections to the exponential behavior can appear (as shown in Fig. 5.7).

Just to give a concrete example let us consider a density of states

$$\rho(E) \sim \begin{cases} (E - E_{min})^N & \text{for} \quad E_{min} \leq E < E_{tnt} \\ \\ \rho_1(E) & \text{elsewhere,} \end{cases} \tag{5.44}$$

where $\rho_1(E)$ is a smooth function with a negative second derivative, and

$$P_E(m=0) \sim (E - E_{tnt})^\gamma \,. \tag{5.45}$$

In the limit $T \ll |E_{tnt}|$, the integrals in Eq. (5.31) can be computed, thus obtaining

$$\tau_T \sim T^{-\gamma} \frac{e^{\Delta/T}}{\sum_{k=0}^{N} \frac{\Gamma(N+\gamma-k+1)}{k!(N-k)!} \left(\frac{\Delta}{T}\right)^k} \tag{5.46}$$

where $\Gamma(x)$ is the Gamma function.

If we further assume $\gamma = N$ as we have found for long range interacting systems, Eq. (5.46) becomes:

$$\tau_T \sim \frac{e^{\Delta/T}}{\sum_{k=0}^{N} \frac{(2N-k+1)!}{k!(N-k)!} \left(\frac{T}{\Delta}\right)^{N-k}} . \tag{5.47}$$

Needless to say, for long-range interacting systems, the temperature region where appreciable deviations to the Arrhenius law occurs for $T_{cr} < T < |E_{tnt}|$. It is far from obvious that short-range interaction system will present the same temperature dependence as long range ones. Indeed, in that case, usually one has $E_{tnt} \sim E_{min}$, so that the evaluation of Eq. (5.31) calls for different approximations.

Last, but not least, let us remark that the model given by Hamiltonian (5.2) at criticality ($\alpha = 1$) is very interesting. Indeed, for $\alpha = 1$ the parameter r (the ratio between the disconnected portion of energy space compared to the full one) goes to zero in the thermodynamical limit. The difference with the short-range case is that r goes to zero logarithmically, instead of a power law: $r \sim 1/\ln N$. This simply means the existence of an effective phase transition for finite systems at criticality.

In their seminal 1966 paper, Griffiths *et al[55]* showed that transition times between metastable states, as given by the minima of the free energy $A(m,T)$ depends on the free energy barrier, δ, as:

$$\tau_G = e^{\beta\delta} \tag{5.48}$$

where, in our case:

$$\delta = A(0,T) - \min_m [A(m,T)] \tag{5.49}$$

On the other hand, the free energy barrier can be given by:

$$\delta = \frac{P_{max}(T)}{P_0(T)} \tag{5.50}$$

where $P_{max}(T)$ is the temperature dependent probability of the most likely value of the magnetization. Near TNT, $P_{max}(T) \gg P_0(T)$ and can be considered as roughly constant, which leads us to:

$$\tau \sim P_0(T)^{-1} \tag{5.51}$$

that we can easily obtain from out the results in the previous section.

5.5 Publications

[59] F. Borgonovi, G.L.Celardo, B. Gonçalves and L. Spadafora, Magnetic Reversal Time in Open Long Range Systems, *e-print cond-mat/0710.3935*, (2007)

Chapter 6

Hanoi Networks

”Save that from yonder ivy-mantled tower,

The moping owl does to the moon complain.”

– Thomas Gray - American Poet

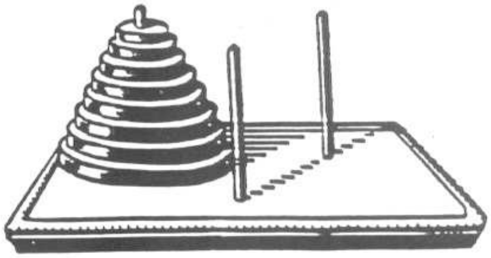

Figure 6.1: Illustration of the Tower of Hanoi puzzle.

According to legend, there is temple in the city of Hanoi in Vietnam that contains a series of $k = 64$ gold disks and 3 pegs. The priests of the Indian deity Brahma, were ordered by an ancient prophecy to move all the disks from the original peg to another, always in increasing order of size, and without ever placing a larger disk on top of a smaller one. The third peg is provided as overflow. If you number the disks from smallest to largest, the task proceeds as follows; First, disk 1 moves to the overflow and disk 2 to the second stack, followed by disk 1 on top. Now, disk 3 can move to the overflow, disk 1 back onto disk 4, disk 2 onto disk 3, and 1 onto 2. Now we have a new stack of disks 1, 2, and 3 in perfect order, and only 61 more disks to go! According to the prophecy, the end of this Sisyphean

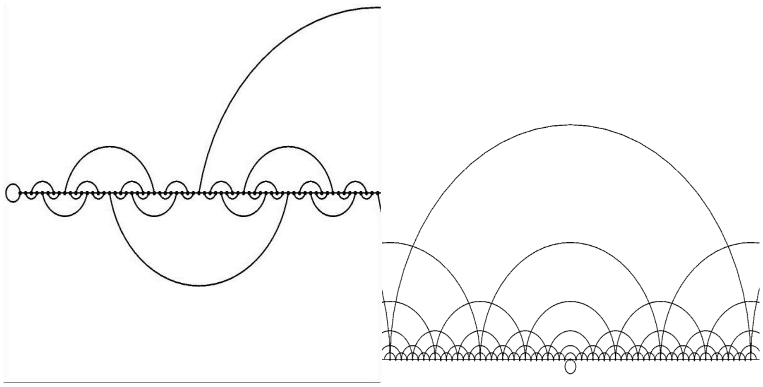

Figure 6.2: left) Hanoi network of degree 3 ($HN3$) right) Hanoi network of degree 4 ($HN4$).

task will signal the end of the world. Luckily, if we assume that the priests perform one move per second, 24 hours a day, they would take roughly 600 billion years to complete the $2^{64} - 1$ moves required.

This problem can, of course, be generalized to any number of disks and is normally used as a simple example of a recursive algorithm in introductory Computer Science classes.[60, 61]. But notice the sequence of labels, i, for the disks moved:

$$1 - 2 - 1 - 3 - 1 - 2 - 1 - 4 - 1 - 2 - 1 - 3 - 1 - 2 - 1 - 5 - \ldots \tag{6.1}$$

We can use this sequence to create a class of networks as follows. First, we lay out this sequence on a $1d$-line of nearest neighbor connected sites labeled from $n = 1$ to $n = N \equiv 2^k - 1$(the number of moves required to finish the problem). In general, any site on this line can be described uniquely by:

$$n = 2^{i-1}\left(2j + 1\right) \tag{6.2}$$

where i corresponds to the label of the disk moved at step n in the sequence above and $j = 0, 1, 2, \ldots$. If we now connect each site n to one (two) sites that are at distance 2^i away and share the same value of i we obtain the three (four) connected Hanoi Network[1]. We will refer to these networks has $HN3$ and $HN4$ respectively, and we illustrate their structure in Fig. 6.2.

In summary, all odd numbered nodes are connected to their second nearest neighbors

[1] In the case of the three connected network, we always choose the neighbor with a site of value at most $i + 1$ between them.

and the odd multiples of powers of two get exponentially long links where the length of a nodes link is given by 2^i where i is the same as in Eq. 6.2. This value is given by counting how many times the number can be divided by 2 and adding 1.

One can think of this class of networks as models of a situation where everybody knows their neighbors, but an exponentially decreasing number of individuals have some friends exponentially far away. Since everybody has exactly the same number of connections, the behavior of any node will be determined by the distance at which it is connected to other nodes.

6.1 Geometry

We start the study of this system by looking at some of its geometrical properties, so as to get some insight that will prove useful when analyzing a wealth of other characteristics of this apparently simple class of graphs.

For $HN3$, it is simple to determine geometric properties, for instance, its diameter d, the longest of the shortest paths between any two sites. Using a sequence of networks for $k = 2, 4, 6, \ldots$, the diameter-path looks like a Koch curve, see Fig. 6.4. We can define the path Π_k as a sequence of jumps reaching from one end of the $N = 2^k$-long graph to the other via

$$\begin{aligned}
\Pi_0 &= 1, \\
\Pi_2 &= 1 - 2 - 1 = \Pi_0 - 2 - \Pi_0, \\
\Pi_4 &= 1 - 2 - 1 - 8 - 1 - 2 - 1 = \Pi_2 - 8 - \Pi_2, \\
&\cdots , \\
\Pi_{k+2} &= \Pi_k - 2^{k+1} - \Pi_k, \qquad (6.3)
\end{aligned}$$

using obvious notation. The length d_k of each marked path is given by $d_{k+2} = 2d_k + 1$ for $N_{k+2} = 4N_k$, hence

$$d \sim \sqrt{N}. \qquad (6.4)$$

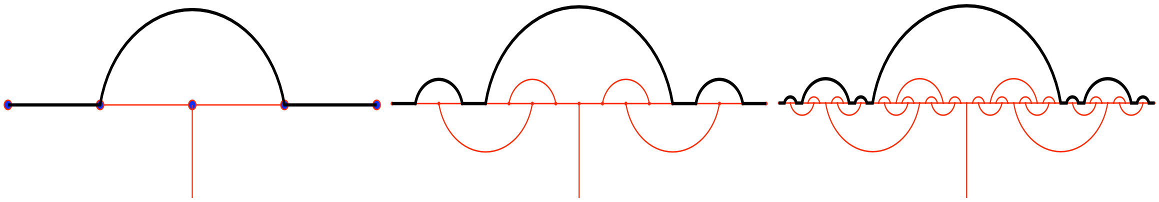

Figure 6.3: (Color Online) Sequence of shortest end-to-end paths (=diameter, thick lines) for $HN3$ of size $N = 2^k$, $k = 2, 4, 8$. Whenever the system size N increases by a factor of 4, the diameter d increases by a factor of ~ 2, leading to Eq. (6.4).

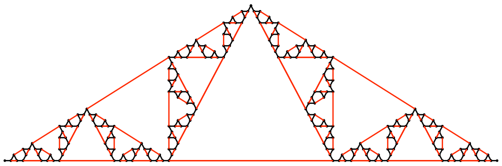

Figure 6.4: Alternative view of the 3-regular graph in Fig. 6.2 as a Koch curve. In this view, the end-to-end shortest marked in Fig. 6.3 is in fact simply the base line here.

This property is reminiscent of a square-lattice of N sites, whose diameter (=diagonal) is also $\sim \sqrt{N}$. $HN3$ is thus far from true SW behavior where $d \sim \ln^\alpha N$, with $\alpha \equiv 1$.

The geometry of $HN4$ is more subtle. We consider again the shortest path between the origin $n = 0$ and the end $n = N = 2^k$. Using the notation from Eq. (6.3), we have

$$\begin{aligned}
\Pi_0 &= 1, \\
\Pi_1 &= 1-1, && (6.5) \\
\Pi_2 &= 1-2-1 = \Pi_0 - 2 - \Pi_0, \\
\Pi_3 &= 1-2-2-2-1 = \Pi_0 - 3\times 2 - \Pi_0, && (6.6) \\
\Pi_4 &= 1-1-4-4-4-1-1 = \Pi_1 - 3\times 4 - \Pi_1, \\
\Pi_5 &= 1-2-1-8-8-8-1-2-1 = \Pi_2 - 3\times 8 - \Pi_2, \\
\Pi_6 &= 1-2-1-7\times 8-1-2-1 = \Pi_2 - 7\times 8 - \Pi_2, && (6.7)
\end{aligned}$$

and so on. Due to degeneracies at each level, one has to probe many levels in the hierarchy to discern a pattern. In fact, any pattern evolves for an increasing number of levels before it gets taken over by a new one, with two patterns creating degeneracies at the crossover.

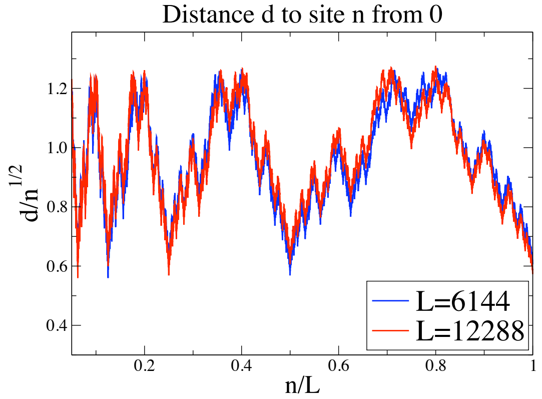

Figure 6.5: Plot of the shortest path length between the origin of HN3 and the nth site on two networks of extend $L = 3\,2^{11}$ and $L = 3\,2^{12}$. In both sets of data, we plot the path-distance relative to the root of the separation between site n and the origin ($n = 0$) along the linear backbone. Then, all rescaled distances fluctuate around a constant mean. Those fluctuations are very fractal, their self-similarity becoming apparent when super-imposing the data for both sizes L on a relative distance scale with n/L.

Finally, we get (putting the degeneracies aside)

$$\Pi_k = \begin{cases} \Pi_{k-2} - 1 \times 2^{k-1} - \Pi_{k-2}, & (k = 2), \\ \Pi_{k-3} - 3 \times 2^{k-2} - \Pi_{k-3}, & (2 < k \leq 5), \\ \Pi_{k-4} - 7 \times 2^{k-3} - \Pi_{k-4}, & (5 < k \leq 9), \\ \Pi_{k-5} - 15 \times 2^{k-4} - \Pi_{k-5}, & (9 < k \leq 14), \\ \ldots, \end{cases} \qquad (6.8)$$

and so on. We find that the paths here do *not* search out the longest possible jump, as in Fig. 6.3. Instead, the paths reach quickly to some intermediate level and follow *consecutive* jumps at that level before trailing off in the end. This is a key distinguishing feature between $HN3$ and $HN4$: Once a level is reached in $HN4$, the entire network can be traversed at that level, while in $HN3$ one *must* switch to lower levels to progress (see the left side of Fig. 6.2). Specifically, the first pattern holds for $k = l_1 = 1$, the 2nd for $k = l_2 = 2$, the 3rd for $l_2 = 2 < k \leq 5 = l_3$, and so on, with $l_g = l_{g-1} + g$ (initiated by $l_1 = 1$, $l_2 = 2$) demarking the crossover points between patterns. Hence, $l_g = \frac{1}{2}g\,(g+1) - 1$, $(g \geq 2)$. Within each

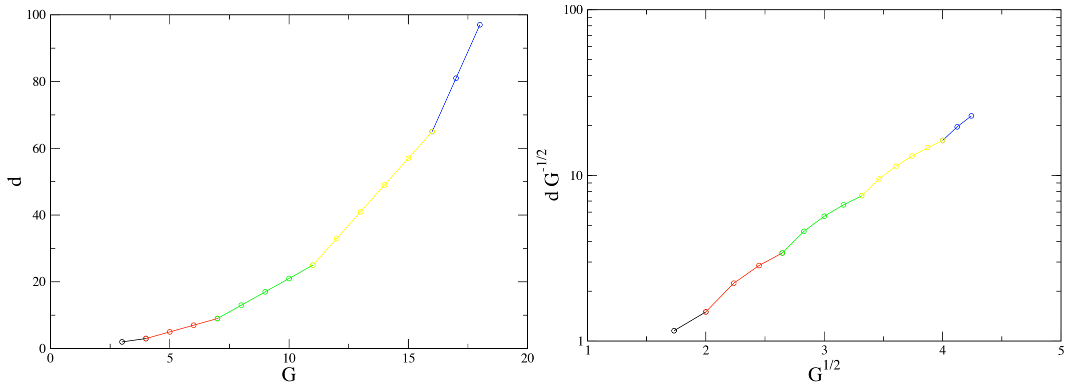

Figure 6.6: left)Ring diameter as a function of $G \equiv \log_2 N$. right) Scaling behavior of d with $\sqrt{G}$.

pattern g, we observe for the end-to-end diameters d_k:

$$d_k^{(g)} = 2d_{k-g}^{(g-1)} + \left(2^{g-1} - 1\right), \quad \left(l_{g-1} < k \leq l_g\right). \tag{6.9}$$

defining $l_1 = 1$, where in general

$$l_g = l_{g-1} + g, \qquad l_2 = 2, \tag{6.10}$$

demarks the crossover point between the generations, see Fig. 6.6. Eq. (6.10) easily yields

$$l_g = \frac{1}{2} g\,(g+1) - 1 \qquad (g \geq 2)\,. \tag{6.11}$$

To obtain the asymptotic behavior for d_k, instead of solving Eq. (6.9) for all k, we note that exactly on the crossover points $k = l_g$ (i. e. $k - g = l_{g-1}$) we have

$$d_{l_g}^g = 2d_{l_{g-1}}^{g-1} + \left(2^{g-1} - 1\right). \tag{6.12}$$

Defining $e_g = 2^g d_{l_g}^g$, we get

$$e_g = e_{g-1} + \frac{1}{2} - 2^{-g},$$

which is easily summed up to give

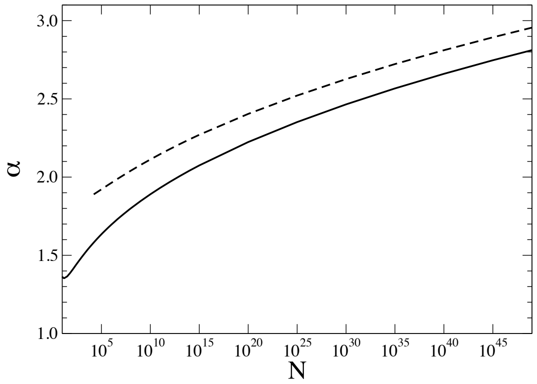

Figure 6.7: Plot of the system-size dependence of the exponent $\alpha = \alpha(N)$ defined in Eq. (6.15). The solid curve is the exact value based on Eq. (6.13), and the dashed curve is the asymptotic approximation also given in Eq. (6.15).

$$d_{l_g}^{(g)} \quad = \quad (g-1)\,2^{g-1}+1. \tag{6.13}$$

With $k = l_g$ and $g \sim \sqrt{2l_g} \sim \sqrt{2k} \sim \sqrt{\log_2 N^2}$, it is

$$d_k \quad \sim \tfrac{1}{2} \quad \sqrt{\log_2 N^2}\; 2^{\sqrt{\log_2 N^2}} \qquad (N \to \infty) \tag{6.14}$$

for the diameter of $HN4$. Expecting the diameter of a small world to scale as $d \sim \log N$, we rewrite Eq. (6.14):

$$d_k \quad \sim \quad (\log_2 N)^{\alpha} \quad \text{with} \quad \alpha \sim \frac{\sqrt{\log_2 N^2}}{\log_2 \log_2 N^2} + \frac{1}{2}. \tag{6.15}$$

Technically, α diverges with N and the diameter grows faster than any power of $\log_2 N$ [but less than *any* power of N, unlike Eq. (6.4)]. In reality, though, α varies only very slowly with N, ranging merely from $\alpha \approx 1.44$ to ≈ 2.8 over *fifty* orders of magnitude, $N = 10 - 10^{50}$, as shown in Fig. 6.4.

Another topological property of interest is the average distance, $\langle d\rangle$, between two nodes. This quantity gives us a sense of how many hops, on average, are required to travel between one node and another along the shortest path connecting them, which has obvious important consequences for the transport properties of the network, since in networks with smaller

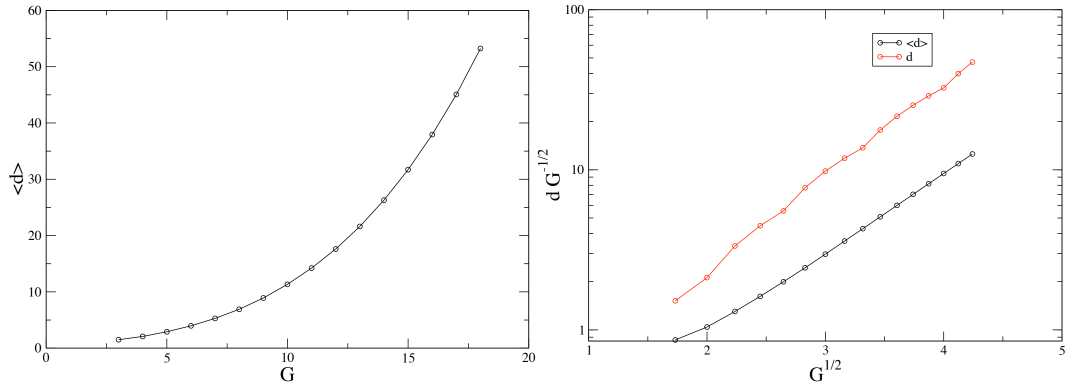

Figure 6.8: left) Average distance between nodes as a function of G for $HN4$. right) Scaling comparison of $\langle d\rangle$ and d vs G. The functional form is similar, but the overal constant is different.

average distance between nodes it's easier to send messages between any two points. The left side of Fig. 6.8 plots this quantity as a function of G. We find that it possesses a similar dependency as the maximal quantity d, but with different values of A and B, as shown on the right side of 6.8, where we plot both functions scaled by $\sqrt{G}$. As expected, the slope of d (≈ 0.6) is larger than the slope of $\langle d\rangle$ (≈ 0.5).

6.2 Diffusion

To model diffusion on these networks, we study simple random walks with nearest-neighbor jumps along the available links. Here, we focus merely on the asymptotic mean-square displacement

$$\left\langle r^2\right\rangle \sim t^{2/d_w}, \tag{6.16}$$

which defines the fractal dimension of the walk in terms of the exponent d_w.

First, we consider a random walk on HN4. The "master-equation" [62] for the probability of the walker to be at site n, at time t is given by

$$\begin{aligned}\mathcal{P}_{n,t} &= \frac{1-p}{2}\left[\mathcal{P}_{n-1,t-1}+\mathcal{P}_{n+1,t-1}\right]\\ &\quad+\frac{p}{2}\left[\mathcal{P}_{n-2^i,t-1}+\mathcal{P}_{n+2^i,t-1}\right],\end{aligned} \tag{6.17}$$

where p is the probability to make a long-range jump. (Throughout this Chapter, we considered p uniform, independent of n or t). A detailed treatment of this equation is quite involved and will be given elsewhere. Instead, we note that the long-time behavior is dominated by the long-range jumps, as discussed below for HN3. To simplify matters, we set $p = 1/2$ here, although any finite probability would lead to the same conclusions. We make an "annealed" approximation, i. e., we assume that we happen to be at some site n in Eq. (6.2) with probability $1/2^i$, corresponding to the relative frequency of such a site, yet independent of time or history. This ignores the fact that in the network geometry a long jump of length 2^i can be followed *only* by another jump of that length or a jump of unit length, and that many intervening steps are necessary to make a jump of length 2^{i+1}, for instance. Here, at each instant the walker jumps a distance 2^i left or right irrespectively with probability $1/2^i$, and we can write

$$\mathcal{P}_{n,t} \quad = \quad \sum_{n'} T_{n,n'} \mathcal{P}_{n',t-1} \tag{6.18}$$

with

$$T_{n,n'} \quad = \quad \frac{a-1}{2a} \sum_{i=0}^{\infty} a^{-i} \left(\delta_{n-n',b^i} + \delta_{n-n',-b^i} \right), \tag{6.19}$$

where $a = b = 2$. Eqs. (6.18-6.19) are identical to the Weierstrass random walk discussed in Refs. [63, 64] for arbitrary $1 < a < b^2$. There, it was shown that $d_w = \ln(a)/\ln(b)$, which leads to the conclusion that $d_w = 1$ in Eq. (6.16) for $HN4$. Simulations for $HN4$, evolving some 2×10^7 walks for 10^6 time steps each, yield a value of $d_w = 1$ with high confidence, see Fig. 6.9. Hence, a walk on $HN4$ proceeds effectively ballistic, but hardly with linear motion: widely fluctuating jumps conspire just so that a single walker extends outward with an on-average constant velocity in *both* directions. Clearly, it is easier to traverse $HN4$ than $HN3$ because of the above-stated fact that on $HN4$ a walker can progress *repeatedly* within a hierarchical level.

For HN3, the master-equation in the bulk reads for

$$\begin{aligned} \mathcal{P}_{n,t} &= \frac{1-p}{2}\left[\mathcal{P}_{n-1,t-1}+\mathcal{P}_{n+1,t-1}\right]+p\,\mathcal{P}_{n',t-1}, \\ n' &= \begin{cases} n+2^i, & j \text{ even}, \\ n-2^i, & j \text{ odd}, \end{cases} \end{aligned} \tag{6.20}$$

In the RG [62, 65] solution of Eq. (6.20), at each step we eliminate all odd sites, i. e., those sites with $i = 0$ in Eq. (6.2). As shown in Fig. 6.10, the elementary unit of sites effected is centered at all sites n having $i = 1$ in Eq. (6.2). We know that such a site n is surrounded by two sites of odd index, which are mutually linked. Furthermore, n is linked by a long-distance jump to a site also of type $i = 1$ at $n \pm 4$ in the neighboring elementary unit, where the direction does not matter here. The sites $n \pm 2$, which are shared at the boundary between such neighboring units also have even index, but their value of $i \geq 2$ is undetermined and irrelevant for the immediate RG step, as they have a long-distance jump to some sites $m_\pm$ at least eight sites away.

Using a standard generating function [62],

$$x_n(z) = \sum_{t=0}^{\infty} \mathcal{P}_{n,t}\, z^t, \tag{6.21}$$

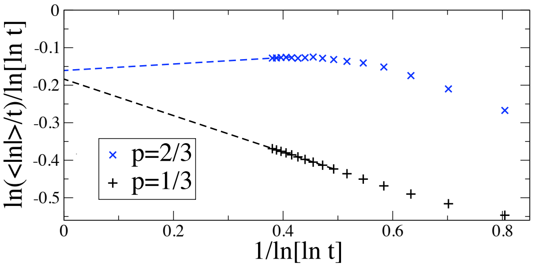

Figure 6.9: Rescaled plot of the mean distance $\langle |n| \rangle$ in $HN4$ for walks up to $t = 10^6$. We demonstrate that $d_w = 1$ but with log-corrections by rewriting Eq. (6.16) as $\langle |n| \rangle / t \sim V\,[\ln t]^{\beta}$. Then we obtain $\ln\left(\langle |n| \rangle / t\right) / \ln\left[\ln t\right] \sim \beta + \ln V / \ln\left[\ln t\right]$ and linearly extrapolate (dashed lines) $1/\ln\left[\ln t\right] \to 0$, estimating $\beta \approx -0.18$ at the intercept, independent of p. An effective "velocity" V could be extracted from the slope. For any value besides $d_w = 1$, these extrapolations would *not* converge.

yields for the five sites inside the elementary unit centered at n:

$$\begin{aligned}
x_n &= a\left(x_{n-1}+x_{n+1}\right) \\
&\qquad +c\left(x_{n-2}+x_{n+2}\right)+p_2\, x_{n\pm4}, \\
x_{n\pm1} &= b\left(x_n+x_{n\pm2}\right)+p_1\, x_{n\mp1}, \\
x_{n\pm2} &= a\left(x_{n\pm1}+x_{n\pm3}\right) \\
&\qquad +c\left(x_n+x_{n\pm4}\right)+p_2\, x_{m_\pm},
\end{aligned} \tag{6.22}$$

where we have absorbed the parameters p and z into general “hoping rates”, which are initially $a^{(0)}=b^{(0)}=\frac{z}{2}(1-p)$, $c^{(0)}=0$, and $p_1^{(0)}=p_2^{(0)}=zp$.

The RG update step consist of eliminating from these five equations those two that refer to an odd index, $n\pm1$. We obtain

$$\begin{aligned}
x_n &= b'\left(x_{n-2}+x_{n+2}\right)+p_1'\, x_{n\pm4}, \\
x_{n\pm2} &= a'\left(x_n+x_{n\pm4}\right) \\
&\qquad +c'\left(x_{n\mp2}+x_{n\pm6}\right)+p_2'\, x_{m_\pm},
\end{aligned} \tag{6.23}$$

with

$$\begin{aligned}
a' &= \frac{\left[ab+c\left(1-p_1\right)\right]\left(1+p_1\right)}{1-p_1^2-2ab}, \\
b' &= \frac{ab+c\left(1-p_1\right)}{1-p_1-2ab}, \\
c' &= \frac{abp_1}{1-p_1^2-2ab}, \\
p_1' &= \frac{p_2\left(1-p_1\right)}{1-p_1-2ab}, \\
p_2' &= \frac{p_2\left(1-p_1^2\right)}{1-p_1^2-2ab}.
\end{aligned} \tag{6.24}$$

If we further identify for all sites $l=n,n\pm2,n\pm4,\dots$ in Eq. (6.23)[2] $x_l=Ax'_{l/2}$, we note that the primed equations coincide with the unprimed ones in Eqs. (6.22). Hence, the RG

[2] The constant A is determined when initial and boundary conditions are considered, essential for the case of first passage and return times [62].

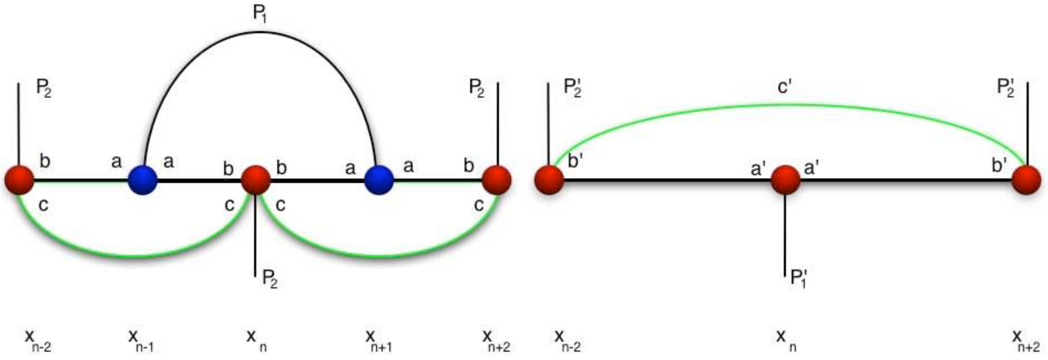

Figure 6.10: Depiction of the (exact) RG step for random walks on $HN3$. Hopping rates from one site to another along a link are labeled at the originating site. The RG step consists of tracing out odd-labeled variables $x_{n\pm1}$ in the top graph and expressing the renormalized rates (a', b', c', p_1', p_2') on the right in terms of the previous ones (a, b, c, p_1, p_2) on the bottom. The node x_n, bridged by a (dotted) link between x_{n-1} and x_{n+1}, is special as it *must* have $n = 2(2j+1)$ and is to be decimated at the following RG step, justifying the designation of p_1'. Note that the original graph does not have the green, links with hopping rates (c, c'), which *emerge* during the RG recursion.

recursion equations in (6.24) are *exact* at any step k of the RG, where unprimed quantities refer to the kth recursion and primed ones to $k+1$.

Solving Eqs. (6.24) algebraically at infinite time [which corresponds to the limit $z \to 1$, see Eq. (6.21)] and for $k+1 \sim k \to \infty$ (by dropping the prime on all left-hand parameters), we – apparently – obtain only two fixed points at $a = b = 1/2$ and $c = p_1 = p_2 = 0$, and $a = b = c = 0$ and $p_1 = p_2 = 1$. The first fixed point corresponds to an ordinary $1d$ walk without long-range jumps, in the second there is no hopping along the $1d$-backbone at all and the walker stays *confined,* jumping back-and-forth within a single, long-range jump. Yet, both fixed points are *unstable* with respect to small perturbations in the initial parameters.

Starting with any positive probability p for long-range jumps, those dominate over the $1d$ walk at long times. Paradoxically, exclusive long-range jumps found at the 2nd fixed point lead to confinement, itself undermined by *any* positive probability to escape along the $1d-$line, allowing to reach even longer jumps. Instead, the process gets attracted to a third, stable fixed point hidden inside a singular *boundary layer*[66] in the renormalization group equations (6.24) near the confined state.

We have to account for the asymptotic boundary layer with the Ansatz $\{a, b, c, 1 -$

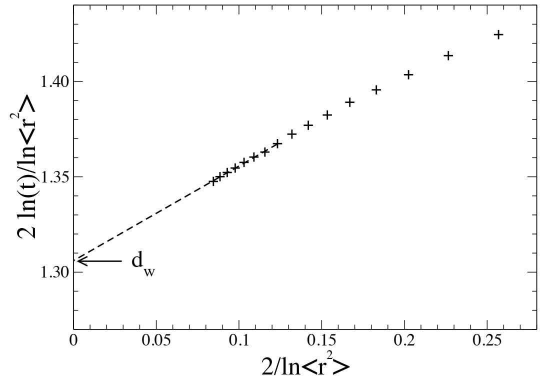

Figure 6.11: Plot of the results from simulations of the mean-square displacement of random walks on $HN3$ displayed in Fig. 6.2. More than 10^7 walks were evolved up to $t_{\max} = 10^6$ steps to measure $\langle r^2 \rangle_t$. The data is extrapolated according to Eq. (6.16), such that the intercept on the vertical axis determines d_w asymptotically. The exact result from Eq. (6.27) is indicated by the arrow.

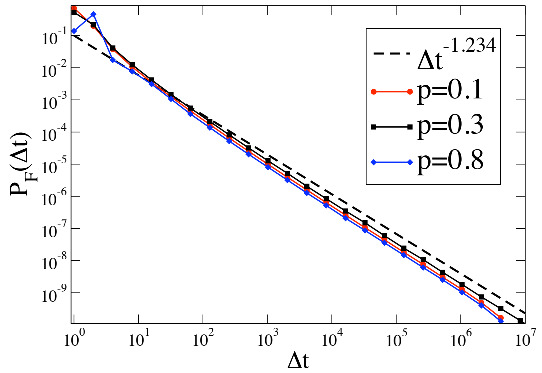

Figure 6.12: Plot of the probability $P_F(\Delta t)$ of first returns to the origin after Δt update steps on a system of unlimited size. Data was collected for three different walks on $HN3$ with $p = 0.1$ (circles), $p = 0.3$ (squares), and $p = 0.8$ (diamonds). The data with the smallest and largest p exhibit strong transient effects. The exact result in Eq. (6.28), $\tau_F = 1.234\ldots$, is indicated by the dashed line.

$p_1, 1-p_2\} \propto \alpha^{-k} \to 0$, where $k \to \infty$ refers to the kth RG step. The largest eigenvalue, *self-consistently* determined from Eqs. (6.24) and the only one satisfying the requirement $\alpha > 1$ in the Ansatz, is $\alpha = 2/\phi$. Here, $\phi = \left(\sqrt{5}+1\right)/2 = 1.6180\ldots$ is the legendary "golden section" [67], first defined by Euclid [68]. Hence, every renormalization of network size, $L \to L' = 2L$, has to be matched by a rescaling of hopping rates with $\alpha = 2/\phi$ to keep motion along the $1d$-backbone finite and prevent confinement (which in light of Eq. (6.26) below corresponds to $\alpha = 0$, its only other physical eigenvalue).

Extending the analysis to include finite-time corrections (i. e., $1 - z \ll 1$), we extend the above Ansatz to

$$y^{(k)} \quad \sim \quad A_y \alpha^{-k} \left\{ 1 + (1-z)\, B_y \beta^k + O\left[(1-z)^2\right] \right\} \tag{6.25}$$

for all $y \in \{a, b, c, 1-p_1, 1-p_2\}$. In addition to the leading-order constants A_y and α, also the next-leading constants are determined self-consistently, and we extract uniquely $\beta = 2\alpha$. Accordingly, time rescales now as

$$T \quad \to \quad T' = 2\alpha T, \tag{6.26}$$

and we obtain from Eq. (6.16) with $T \sim L^{d_w}$ for the diffusion exponent for $HN3$

$$d_w \quad = \quad 2 - \frac{\ln \phi}{\ln 2} = 1.30576\ldots. \tag{6.27}$$

The result for d_w is in excellent agreement with our simulations, as shown in Fig. 6.11.

Using the methods from Ref. [62], we can further show that the walk is recurrent and that the exponent τ_F for the probability distribution, $P_F(\Delta t) \sim \Delta t^{-\tau_F}$, of first-return times Δt is given by

$$\tau_F \quad = \quad 2 - \frac{1}{d_w} = 1.2342\ldots. \tag{6.28}$$

The relation between τ_F and d_w is typical also for Lévy flights [69], and the result is again borne out by our simulations, see Fig. 6.12.

We conclude by contrasting the behavior of $HN4$ discovered before with the analysis of $HN3$. Clearly, when long-range jumps are interconnected as in $HN4$, there is no confinement, the boundary layer disappears [which would be similar to $\alpha = 1$ in Eqs. (6.25-6.26) for $HN3$], and diffusion spreads ballistically, $d_w = 1$. Our numerical studies, and the similarity to Weierstrass random walks [63], further supports that τ_F for walks on $HN4$ is also given by Eq. (6.28), leading to $\tau_F = 1$. Since this value of the exponent matches the embedding dimension, $d = 1$, of the network, the walk is marginally recurrent, which explains the strong logarithmic corrections observed in the simulations.

6.3 Ising Model

In this section, we study Ising spin models on the 3- and 4-regular graphs, both with Monte-Carlo simulations and with the renormalization group.

6.3.1 Renormalization Group

First, we consider the renormalization group for the Ising model on $HN3$. In this case, all steps can be done exactly but the result turns out to be trivial in the sense that there are no finite-temperature fixed points of the RG flow. Yet, the calculation is instructive, highlighting the large family of models access through the hierarchical nature of the process, and is almost identical in outcome to the approximate treatment below for the non-trivial $HN4$.

We ignore the distinction between semi-infinite or infinite lattice, boundary conditions, and the special site $n = 0$, as we are merely interested in bulk properties. The renormalization consists of recursively tracing out spins level-by-level in the hierarchy. In terms of Eq. (6.2), we start by tracing out all sites with n odd, i. e. $i = 0$, then those n which are divisible by 2 only once, i. e. $i = 1$, and so on. We can always relabel all sites n after any renormalization step by $n \to n/2$, so that we trace out odd-relabeled sites at any level. It is apparent, for instance from Fig. 6.2, that odd-labeled sites are connected to their even-labeled nearest neighbors on the lattice backbone, say, by a coupling K_0. At any level, each odd-labeled site $x_{n\pm1}$ is also connected to one other such site $x_{n\mp1}$ across an even-labeled

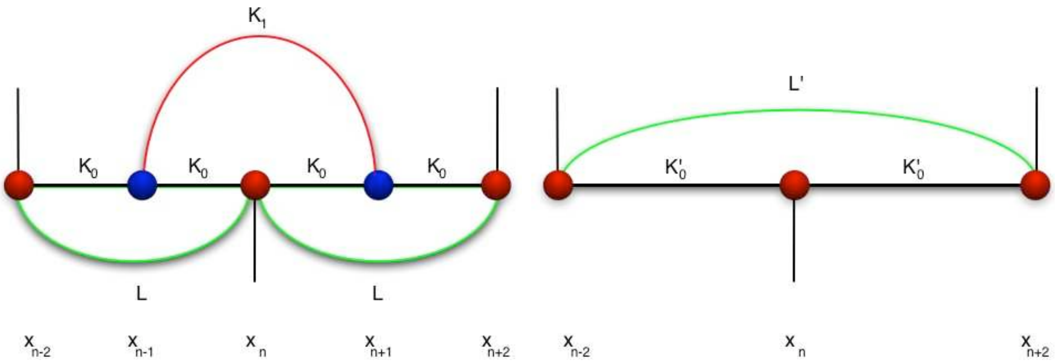

Figure 6.13: Depiction of (exact) renormalization group step for the Ising model on the $HN3$. The step consists of tracing out odd-labeled variables $x_{n\pm1}$ in the left plot and expressing the renormalized couplings (L', K_0') on the right in terms of the old couplings (L, K_0, K_1). Note that the original graph in Fig. **??** does not contain couplings of type (L, L'), but that they certainly become relevant during the process.

site x_n with $n = 2\,(2j+1)$ that is exactly *once* divisible by 2. Let us call that coupling K_1. The basic renormalization step is depicted in Fig. 6.13 and consists of tracing out the two sites $x_{n\pm1}$ neighboring the site x_n for all j with $n = 2\,(2j+1)$. While not present initially, during the process a new coupling L between next-nearest even-labeled neighbors emerges. We can section the Hamiltonian

$$-\beta\mathcal{H} \;=\; \sum_{\{n=2(2j+1)\}} \left(-\beta\mathcal{H}_n\right) + \mathcal{R}\left(K_2, K_3, \ldots\right), \tag{6.29}$$

where $\mathcal{R}$ contains all coupling terms of higher level in the hierarchy, and each sectional Hamiltonian is given by

$$\begin{aligned} -\beta\mathcal{H}_n \;=\;& I + K_0\left(x_{n-2}x_{n-1} + x_{n-1}x_n + x_n x_{n+1} + x_{n+1}x_{n+2}\right) \\ & + K_1 x_{n-1}x_{n+1} + L\left(x_{n-2}x_n + x_n x_{n+2}\right), \end{aligned} \tag{6.30}$$

where (K_0, K_1, L) are the unrenormalized couplings defined in Fig. 6.13 and I is constant that fixes the overall energy scale. Tracing out the odd-labeled spins, we have to evaluate

$$\begin{aligned}\sum_{\{x_{n-1}=\pm 1\}}\sum_{\{x_{n+1}=\pm 1\}} e^{-\beta\mathcal{H}_n} &= 2e^{I+L(x_{n-2}x_n+x_nx_{n+2})}\\ &\quad\times\left\{e^{K_1}\cosh\left[K_0\left(x_{n-2}+2x_n+x_{n+2}\right)\right]\right.\\ &\quad\left.+\,e^{-K_1}\cosh\left[K_0\left(x_{n-2}-x_{n+2}\right)\right]\right\} \qquad (6.31)\\ &= e^{I'+K_0'(x_{n-2}x_n+x_nx_{n+2})+L'x_{n-2}x_{n+2}} \qquad (6.32)\end{aligned}$$

for the remaining spins $x_{n-2}, x_n, x_{n+2} = \pm 1$. After some algebra, we find for the renormalized quantities

$$\begin{aligned}K_0' &= L+\frac{1}{2}\ln\cosh\left(2K_0\right)+\frac{1}{4}\ln\left[1+\tanh\left(K_1\right)\tanh^2\left(2K_0\right)\right],\\ L' &= \frac{1}{4}\ln\left[1+\tanh\left(K_1\right)\tanh^2\left(2K_0\right)\right],\\ I' &= I+2\ln 2+\ln\cosh\left(K_1\right)+\ln\cosh\left(2K_0\right)\\ &\quad+\frac{1}{4}\ln\left[1+\tanh\left(K_1\right)\tanh^2\left(2K_0\right)\right], \qquad (6.33)\end{aligned}$$

and $K_i' = K_{i+1}$ for $i \geq 1$. Searching for fixed points $K_0' = K_0 = K_0^*$ and $L' = L = L^*$ immediately provides the trivial, high-temperature solution $K_0^* = L^* = 0$. Excluding that and eliminating L^*, further analysis yields

$$1 = \tanh\left(K_1\right)\tanh\left(2K_0^*\right), \qquad (6.34)$$

which, unfortunately, has no non-trivial (finite-T) fixed points aside from the $T_c = 0$ solution $K_0^* = \infty$. Note, however, that the renormalization recursion equations (6.33) have a remarkable property due to the hierarchical structure of the graph: The next-level coupling K_1 appears as a *free parameter* in the equations and acts as a kind of source term that could be chosen to represent physically interesting situations, such as a distance-dependent coupling and/or some form of disorder. Yet, that potential for (real) K_1 does not change much of the fact that the fixed point Eq. (6.34) has no finite-T solution for any choice. Thus, while the geometry of $HN3$ analyzed in Sec. 6.1 indicates a fractal dimension above $d = 1$, the ferromagnet on this graph, while exactly solvable, essentially acts as a 1d Ising model.

Again, $HN4$ in contrast provides a non-trivial solution for the Ising model. But, we fail here to obtain an exact result, although we hope that in light of the inherent symmetries such a solution should be possible. Instead, we proceed to a Niemeijer-van Leeuwen cumulant approximation and compare with our numerical results.

The Hamiltonian for the Ising model on $HN4$ indeed has a very hierarchical form:

$$\begin{aligned} -\beta\mathcal{H} &= \sum_{n=1}^{N} K_0 x_{n-1}x_n + \sum_{j=1}^{N/2} K_1 x_{2j-1}x_{2j+1} + \sum_{j=1}^{N/4} K_2 x_{2(2j-1)}x_{2(2j+1)} + \ldots, \\ &= \sum_{n=1}^{2^k} K_0 x_{n-1}x_n + \sum_{i=1}^{k}\sum_{j=1}^{2^{k-i}} K_i x_{2^{i-1}(2j-1)}x_{2^{i-1}(2j+1)}, \end{aligned} \tag{6.35}$$

where we have chosen $N = 2^k$. Note that the first sum represents the lattice backbone and the double-sum contains all couplings in the hierarchy. In preparation for the renormalization group, we rewrite Eq. (6.35) as

$$-\beta\mathcal{H} = -\beta\mathcal{H}_0 - \beta\mathcal{V} + \mathcal{R},$$

with

$$\begin{aligned} -\beta\mathcal{H}_0 &= \sum_{j=1}^{2^{k-1}} x_{2j-1}h_{2j-1} + \sum_{j=1}^{2^{k-1}} L x_{2j-2}x_{2j}, \qquad h_{2j-1} = K_0\left(x_{2j-2} + x_{2j}\right), \\ -\beta\mathcal{V} &= \sum_{j=1}^{2^{k-1}} K_1 x_{2j-1}x_{2j+1}, \\ \mathcal{R} &= \sum_{i=2}^{k}\sum_{j=1}^{2^{k-i}} K_i x_{2^{i-1}(2j-1)}x_{2^{i-1}(2j+1)}, \end{aligned} \tag{6.36}$$

where we have added new couplings L that emerge during renormalization, as in Eq. (6.30). In reference to the Niemeijer-van Leeuwen cumulant expansion, we write the partition function

$$\begin{aligned}
\mathcal{Z} &= \mathcal{N}\,\mathbf{Tr}\,e^{-\beta\mathcal{H}},\\
&= \mathcal{N}\,\mathbf{Tr}_{even}\,e^{\mathcal{R}}\,\mathbf{Tr}_{odd}\,e^{-\beta\mathcal{V}}e^{-\beta\mathcal{H}_0},\\
&= \mathcal{N}\,\mathbf{Tr}_{even}\,e^{\mathcal{R}}\left(Z_0\left\langle e^{-\beta\mathcal{V}}\right\rangle_0\right), \qquad (6.37)
\end{aligned}$$

summing over odd-labeled and even-labeled spins separately, with

$$\begin{aligned}
\left\langle e^{-\beta\mathcal{V}}\right\rangle_0 &= \frac{1}{Z_0}\mathbf{Tr}_{odd}\,e^{-\beta\mathcal{V}}e^{-\beta\mathcal{H}_0},\\
&\approx e^{\langle-\beta\mathcal{V}\rangle_0}e^{\frac{\beta^2}{2}\left(\left\langle\mathcal{V}^2\right\rangle_0-\langle\mathcal{V}\rangle_0^2\right)}\ldots, \qquad (6.38)
\end{aligned}$$

and

$$\begin{aligned}
Z_0 &= \mathbf{Tr}_{odd}\,e^{-\beta\mathcal{H}_0},\\
&= \exp\left[\sum_{j=1}^{2^{k-1}}Lx_{2j-2}x_{2j},\right]\prod_{j=1}^{2^{k-1}}\left\{\sum_{x_{2j-1}}\exp\left[x_{2j-1}h_{2j-1}\right]\right\},\\
&= \exp\left[\sum_{j=1}^{2^{k-1}}Lx_{2j-2}x_{2j},\right]\prod_{j=1}^{2^{k-1}}\left\{2\cosh\left[h_{2j-1}\right]\right\}. \qquad (6.39)
\end{aligned}$$

6.3.1.1 Leading order evaluation

In light of Eq. (6.38), to leading order, we need to evaluate

$$\begin{aligned}
\langle-\beta\mathcal{V}\rangle_0 &= \frac{1}{Z_0}\exp\left[\sum_{j=1}^{2^{k-1}}Lx_{2j-2}x_{2j},\right]\prod_{j=1}^{2^{k-1}}\left\{\sum_{x_{2j-1}}\left(\sum_{j=1}^{2^{k-1}}K_1x_{2j-1}x_{2j+1}\right)\exp\left[x_{2j-1}h_{2j-1}\right]\right\},\\
&= \frac{1}{Z_0}\sum_{j=1}^{2^{k-1}}K_1\partial_{h_{2j-1}}\partial_{h_{2j+1}}Z_0,\\
&= \sum_{j=1}^{2^{k-1}}K_1\tanh\left[K_0\left(x_{2j-2}+x_{2j}\right)\right]\tanh\left[K_0\left(x_{2j}+x_{2j+2}\right)\right]. \qquad (6.40)
\end{aligned}$$

Subsequently, we can rewrite

$$Z_0 e^{\langle -\beta\mathcal{V}\rangle_0} = \mathcal{N}' \exp\left[A\sum_{j=1}^{2^{k-1}} x_{2j-2}x_{2j} + B\sum_{j=1}^{2^{k-1}} x_{2j-2}x_{2j+2}\right],$$

and, correspondingly,

$$\begin{aligned}\mathcal{R} + \ln Z_0 - \langle\beta\mathcal{V}\rangle_0 &= \sum_{i=3}^{k}\sum_{j=1}^{2^{k-i}} K_i x_{2^{i-1}(2j-1)}x_{2^{i-1}(2j+1)} + K_2\sum_{j=1}^{2^{k-2}} x_{2(2j-1)}x_{2(2j+1)}\\ &\quad + \left[L + \frac{1}{2}\ln\cosh(2K_0) + \frac{K_1}{2}\tanh^2(2K_0)\right]\sum_{j=1}^{2^{k-1}} x_{2j-2}x_{2j}\\ &\quad + \frac{K_1}{4}\tanh^2(2K_0)\sum_{j=1}^{2^{k-1}} x_{2j-2}x_{2j+2}, \end{aligned} \tag{6.41}$$

where we dropped an overall constant term. Note that Eq. (6.41) only contains even-labeled variables, and we complete the renormalization step by replacing $x_{2n} \to x'_n$, $k \to k'+1$, to get the renormalized Hamiltonian

$$\begin{aligned}-\beta\mathcal{H}' &= \sum_{i=2}^{k'}\sum_{j=1}^{2^{k'-i}} K'_i x'_{2^{i-1}(2j-1)}x'_{2^{i-1}(2j+1)} + \sum_{j=1}^{2^{k'-1}} K'_1 x'_{2j-1}x'_{2j+1}\\ &\quad + \sum_{j=1}^{2^{k'-1}} K'_0 x'_{j-1}x'_j + \sum_{j=1}^{2^{k-1}} L' x'_{2j-1}x'_{2j+1}, \end{aligned} \tag{6.42}$$

where we identify

$$\begin{aligned} K'_0 &= L + \frac{1}{2}\ln\cosh(2K_0) + \frac{K_1}{2}\tanh^2(2K_0),\\ K'_1 &= K_2 + \frac{K_1}{4}\tanh^2(2K_0),\\ K'_i &= K_{i+1} \qquad (i \geq 2),\\ L' &= \frac{K_1}{4}\tanh^2(2K_0). \end{aligned} \tag{6.43}$$

(Note: These are exactly the same relation one would obtain for the $1d$ Ising model with *nnn* couplings $K_1 = L$, except there $K_i = 0$ for $i \geq 2$. In that case, one would find – correctly

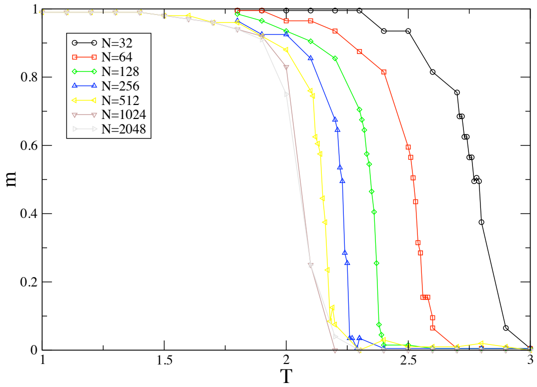

Figure 6.14: Magnetization as a function of temperature for the ferromagnetic ring.

– that there are *no* non-trivial fixed points, as with $K_2 = 0$ and assuming $K_1' = K_1 = K_1^*$ finite in the $2nd$ relation, $4 = \tanh^2 2K_0^*$ has no real solution.)

6.3.2 Monte-Carlo Simulations

The next step is to study the magnetic consequences that the geometry of $HN3$ has, by placing Ising spins on the nodes and applying methods similar to the ones used above in the study of magnetic systems. Spins only interact along the existing edges and there are several possible ways of choosing the interaction energies, J_{ij}, so we start with the simplest choice, $J_{ij} \equiv +1$, corresponding to an Ising ferromagnet.

The ferromagnetic system was placed in contact with a heat bath at different temperatures, and the most likely magnetization at zero field was measured as a function of T. We find a transition from paramagnetic behavior, where $m = 0$ at $T > T_c$ to ferromagnetic behavior, where $m = \pm m_1 \neq 0$ at $T < T_c$, with m_1 reaching 1 at sufficiently low temperatures. We plot this behavior in Fig. 6.14, for different system sizes. Our preliminary results indicate that $2.0 \leq T_c \leq 2.2$, in reasonable agreement with the analytical results above.

6.4 Publications

[70] S. Boettcher and B. Gonçalves and H. Guclu, Hierarchical, Regular Small-World Networks, *e-print cond-mat/0712.1259*, (2007)

[71] S. Boettcher and B. Gonçalves, Anomalous Diffusion on the Hanoi Networks, *e-print* *cond-mat/0802.2757,* (2008)

Part II

Complex Networks

Chapter 7

Graph Bisection

A new algorithm to solve the graph bisection problem in $G(N, p)$ graphs for the $0 < p < 2$ region is introduced. The algorithm removes part or all of the graph's tree mantle until the remaining core is of size $N/2$. Other isolated components can be used to compensate for a small core size, if necessary. We explore the quality of the decompositions, the structure of the tree-mantle and the $2-$core, the possibility of cutting the core to extend the quality of the algorithm and remark briefly on its time complexity. The conclusions we reach should be applicable to other algorithms as well.

7.1 Introduction

A new Job Batching System is being developed by a .com company. The project requires an efficient way of dividing multiple interdependent jobs between two server machines in such a way that the network traffic between them is minimized in order to maximize overall performance. What is the minimum number of inter-server communications that is possible to achieve?

A Silicon Valley company is developing a new dual-core CPU for high performing computers. The complex design of the CPU requires that the two cores are capable of communicating with each other but since each connection between the two components is expensive a method is required to determine the best way of splitting the circuits in to the different cores without modifying the structural properties of the circuit. What is the least amount of connections that have to be maintained?

A physicist is interested in studying the ground state properties of a ferromagnetic Ising spin system constrained to have zero overall magnetization. The ferromagnetic interaction implies that it is energetically more favorable for nearest neighboring spins to have the same orientation, but the constraint on the magnetization forces half the spins to be oriented UP and the other half DOWN. What is the lowest energy that is accessible by the system?

All of the optimization problems mentioned can be reduced to a unique common form, commonly known in the literature as graph bisection. Here, we want to divide a given graph in half by cutting as few connections between the nodes as possible. The parallels can be quickly established if we identify the nodes in the graph with, respectively, jobs, circuit components and Ising spins, and the edges with, respectively, inter-job communication, wires and spin interactions. Due to all its possible applications, the problem of graph bisection has received a lot of attention over the years [72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83].

Mathematically, a graph, $\mathcal{G}$, is defined as a pair (V, E) of vertices V and edges E that connect them. There are many ways in which such a pairing can be constructed, one of the most common being the one devised by *Erdos* and *Renyi[4]* in 1959. We choose to use them due to the ease with which they are analyzed mathematically and to the fact that there is a huge amount of knowledge accumulated over the years on their properties[84, 85].

These graphs are constructed by connecting any pair of vertices (u, v) with a given probability p. In general, one should consider each edge to be directed and represented by an ordered pair, $(u, v) \neq (v, u)$, but we shall restrict ourselves to the undirected case where $(u, v) = (v, u)$. Graphs created by this procedure, are usually denoted by $\mathcal{G}(N, p)$ to represent the ensemble of graphs $\mathcal{G}$ with N nodes and where a given pair of nodes has a probability p of being connected. We further restrict ourselves by imposing that $u \neq v$ (no self-edges) and that (u, v) appears only once in V (no multiple edges).

The attentive reader will have noticed that the mean degree of these graphs is given by $\langle k \rangle \equiv p(N-1)$, so if p is held constant has N increases, the mean degree will diverge linearly. Since, many of the results we are interested in are mean degree dependent, we choose to keep $\langle k \rangle \equiv \alpha$ constant, thus defining $p \equiv \alpha/(N-1)$. It is also obvious from our definitions that there is no reason why we should expect for there to be a, possibly long, path of edges connecting any two pairs of nodes in any given realization of a $\mathcal{G}(N, p)$ graph. In general, this will depend on the value of p, with there possibly being many components for smaller values of p that will eventually merge in to a single one at some threshold, p_t. In fact, there are two particular points in the interval $[0, \alpha_c/N]$ (with $\alpha_c \approx 2.0636$) that we will be particularly interested in.

For $Np_{GCC} \equiv 2\log 2$, the size of the largest connected component in our graph will approach $N/2$ with high probability as $N \to \infty$. We call this sub-graph the Giant Connected Component (GCC) and below this threshold our "graph" is constituted by possibly many smaller and mutually disconnected subgraphs. At this point, the GCC is constituted by two distinct parts, denser central region composed exclusively by very interconnected nodes, to which several trees are attached to. To the central region we will refer, for reasons that will become apparent later, to as the "2-core", or simple, the core and the trees attached to it will be designated by the phrase "tree mantle". As we move away from $Np = 2\log 2$ towards higher values of Np, the GCC will continue to grow and become increasingly denser until, at $Np_{core} \approx 2$, the size of the 2-core becomes, with high probability, $N/2$. We will look in considerably more detail at what happens over this range of Np values in general and at these two points in particular later on.

7.2 Core Peeling Algorithm

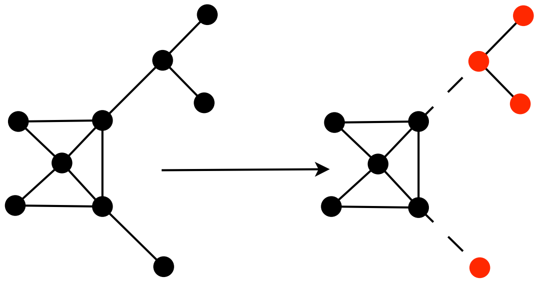

Figure 7.1: Illustration of core peeling. All non-cyclical components connected to the core are cut.

The algorithm we propose is very simple, but we hope it will provide us with some insight on the characteristics that a bisection algorithm must have in order to be successful. Let us represent the nodes on each section of the graph by two different colors, BLACK and RED.

Algorithm 4 Core peeling.

```
1. Start by making all nodes BLACK.
2. Order all components in the graph by their size.
3. Designate the largest component as the "Giant Connected Component" (GCC) and mark all its nodes as being RED
4. Distinguish 2 different possibilities:
    (a) If the size $S$ of the GCC is less than $N/2$ mark as RED exactly $(N/2 - S)$ nodes from the smallest components and return the solution found.
    (b) If the size $S$ is larger than $N/2$ reduce it to its 2-core (Fig. 7.1), by recoloring the mantle nodes BLACK. If the size of the core is less than $N/2$, perform step $4(a)$ using preferentially the trees we just removed as well as the other components. If the size is larger than $N/2$ abort and return the (unbalanced) solution found.
```

Before we carefully analyze the results of this algorithm, a few remarks should be made. The term Giant Connected Component is usually reserved for a component with a size that scales with N, but, for simplicity sake, we use this term for the largest component in the graph. This is justified by the fact that in most cases of interest to us $Np \in [2\log 2, 2]$ the largest component will indeed be the GCC. In step $4(a)$ we are reduced to a knapsack problem where we try to find $(N/2 - S)$ nodes from the remaining graph components. This might not always be possible, specially for smaller sized graphs, resulting in non-zero cut size.

Finally, there is no guaranty that the algorithm we described above will find a balanced solution to the bisection problem. In fact, from standard graph theory results [86] it should stop working when the size of the $2-$core is, with high probability, larger than $N/2$, that is, at $Np = \alpha_c \approx 2.06$. We show this in Fig. 7.2, where we plot the ratio between RED and BLACK nodes as a function of the mean degree Np for various values of N. We can clearly see that, specially for large N, we can always find perfect bisections up to $Np = 2$.

After $Np = 2$, we see a sharp decline in the quality of our solution. After the decline we see an increase of the ratio between RED and BLACK. A quick look at our description above explains this. For $Np > 2$, the size of the core is larger than $N/2$ and when we reach step $4(a)$ the algorithm aborts having marked all the nodes in the core as RED and all others as BLACK. The increase in "quality" is simply an artifact of the increase of the core

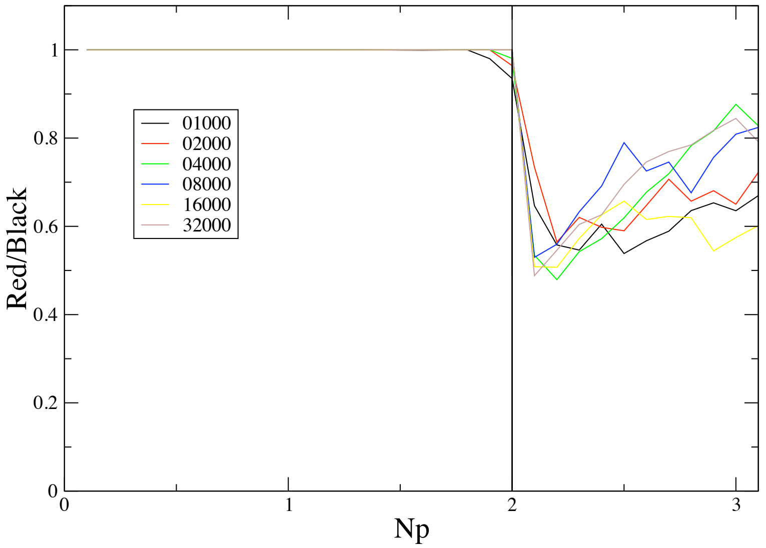

Figure 7.2: Quality of solution found as a function of Np. The value 1 represents a balanced bisection.

size (see Fig. 7.3).

From our previous discussion of the nature of $G(N,p)$ graphs, we should expect that to an increase in the mean degree there should correspond an increase in the number of edges we must cut in order to find a way to split the graph in two equal sized communities. When the GCC size becomes $N/2$, we start cutting trees, which should cause the required number of cuts to decrease until at $Np = 2$, the tree-mantle is no longer sufficient to compensate the "excessive" size of the GCC and our algorithm breaks. We illustrate this behavior on the left side of Fig. 7.5, with the two vertical lines clearly indicating the two points of particular interest. The monotonous decrease in the cut size we observe for higher values of the mean degree, is due to the increasingly denser nature of the GCC and consequentially smaller number of trees available to be cut (at this point we can only cut trees).

Since the tree mantle plays such an important role in the performance of our algorithm and limits the quality of the solutions found, we decided to try to understand some of its properties. We started by looking at the maximum tree size distribution as a function of Np and for various values of N, as shown on the right side of Fig. 7.5. Two features of this plot immediately grab our attention. First, the divergence at $Np \equiv 1$ signals the point in which the two largest components in the graph start merging to create what will eventually become the Giant Connected Component. Second, after $Np = \log 4$, the maximum tree size

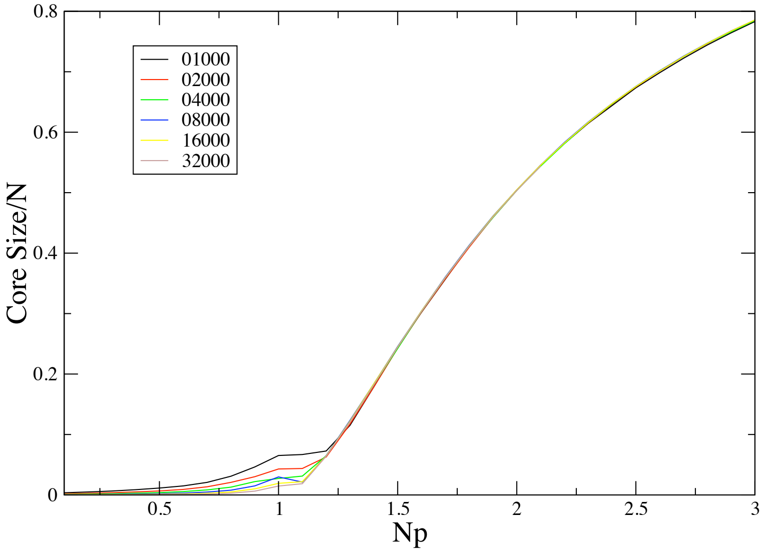

Figure 7.3: Average core size as a function of the mean degree.

scales with the log of N and displays a monotonous decay until it practically vanishes at $Np \equiv 2$.

The next step was then to measure the tree size distribution for different values of the mean degree. For low connectivities, the distribution is unremarkable and the high variability associated with bad statistics prevent us from drawing any conclusions about its shape. Starting at $Np \equiv 1.3$, the statistics become better and we can observe a clear exponential dependence becoming more and more obvious, as we show in Fig. 7.4. Analytical results seem to indicate that this distribution isn't truly exponential, but the accuracy of our numerical results does not allow us to distinguish these results from other similar distributions.

7.3 Core Cutting

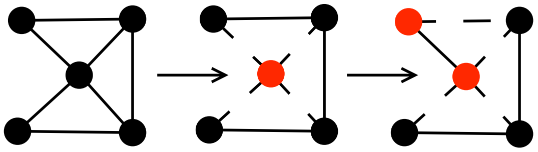

Figure 7.6: Two steps of the core cutting procedure. Nodes are randomly selected and "flipped", cutting any edges connecting nodes in different components, and reestablishing edges between nodes that are now on the same component.

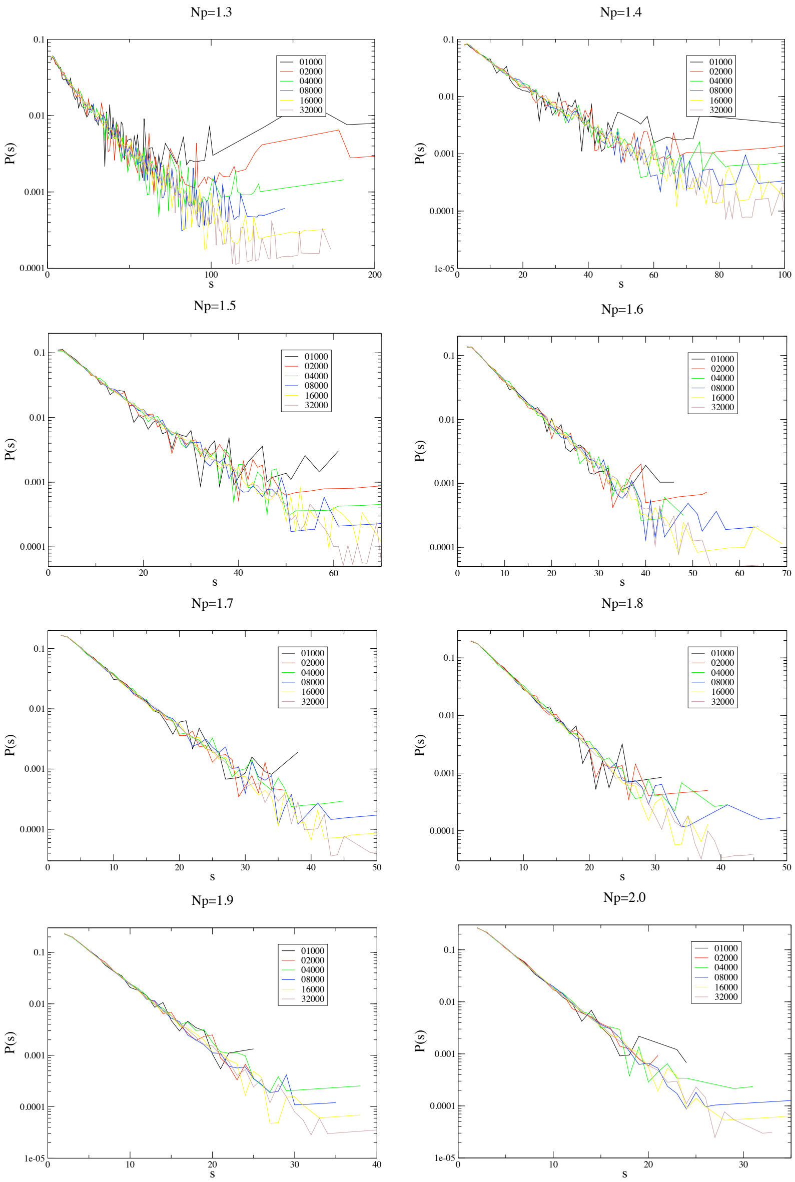

Figure 7.4: Distribution of tree sizes, s, for different values of the mean degree.

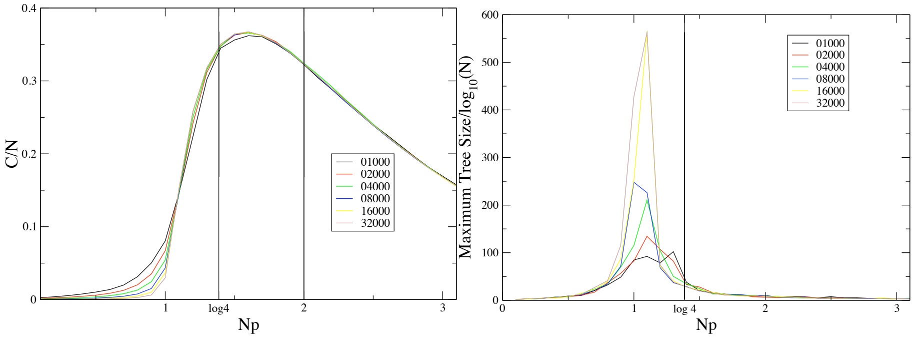

Figure 7.5: left) Number of cut edges as a function of Np. right) Maximum tree size as a function of the mean degree.

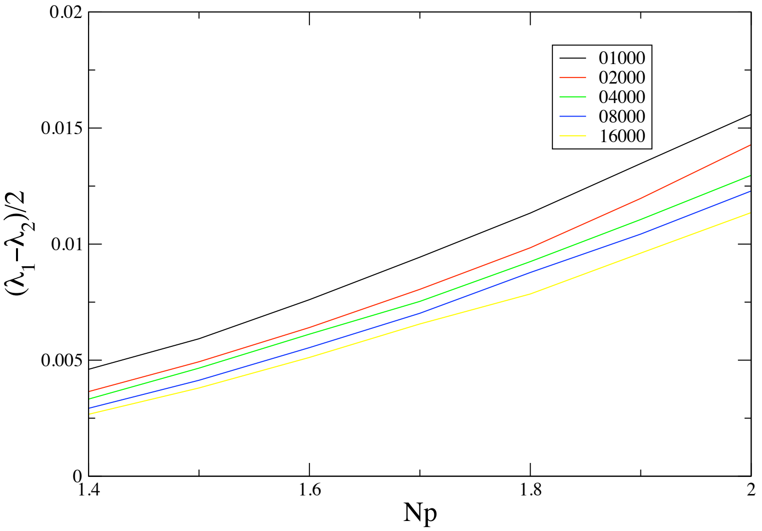

Figure 7.7: Spectral gap as a function of the mean degree.

Our algorithm seems to work reasonably well up until $Np = 2.0$, at which point it stops working altogether, since for larger values of the mean degree the core is simply too large for us to be able to obtain a perfect bisection without having to cut it. It seems, therefore, productive to look in more detail to the structure of the 2−core to see if we could extend the applicability of the algorithm by cutting the core. To see if by cutting the core we could obtain a good candidate for an optimal solution, we calculate the quantity r, defined as:

$$r \equiv \min \frac{e\left(S_1, S_2\right)}{S_1} \tag{7.1}$$

where $e\left(S_1, S_2\right)$ is the number of edges broken by setting S_1 randomly chosen nodes in the core to be red and the remaining S_2 nodes to black. We divide by the size of the smallest of the partitions, S_1 in order to define a quantity that would give us an indication of the average number of links that we must cut per each node we remove from this tightly connected 2−core. The minimization should, in principle, be made over all possible realizations of such partitions. In practice, however, it's impossible to sample all $\left(S_1 + S_2\right)^{(s_1+s_2)}$ combinations so we chose to perform 10 such random choices and average over 100 different network instances for each value of the mean degree considered, our results are displayed in Fig. 7.8, where we scaled the xx axis by the total number of nodes in the core, $S_1 + S_2$. Due to symmetry, we only need to look at partitions with $S_1 \leq N_{core}/2$.

As we can see, the cost in edges cut per each node that we remove from the core decreases linearly as the relative partition size approximates 1/2 but never dips below 1. The maximum cost per node is always for small partition size and varies with the system size, but both the slope of the decrease and the value for the maximum partition size seems to be independent of the graph size. This implies that the number of edges that need to be cut is always larger than the number of nodes we want to remove, thus indicating that cutting the core is always a worst option than tree cutting, where by cutting only one edge we can obtain a relatively large number of nodes of the order of 10^1 or even 10^2. Based on these results, we believe that optimal solutions with minimum possible cost will avoid cutting the core as much as possible, raising the possibility of replica symmetry in the graph bisection problem solution space, at least for $\mathcal{G}(N,p)$ graphs.

One should note that our search procedure in phase space is far from being exhaustive, so there might be a a region of solution space that would indeed provide for a small cost per node removed that we never reach. We assume that this is a possibility and that a better sampling procedure should be devised if we are to consider these results as final. We can however, interpret these results as upper bounds on the ideal case and that there might still be room for improvement.

In an attempt to clarify this situation we measured the spectral gap of the Lagrangian matrix of the core. The Lagrangian is defined as:

$$L \equiv (l_{ij}) = \begin{cases} \frac{1}{k_i} & i \rightarrow j \\ 0 & \text{otherwise} \end{cases} \tag{7.2}$$

where the value of element l_{ij} is $1/k_i$if there exists an edge between nodes i and j and 0 otherwise. Using this definition the largest eigenvalue λ_1 is guaranteed to be 1 and it is well know that if

$$\frac{\lambda_1 - \lambda_2}{2} < r \tag{7.3}$$

where r is the average cost per node as defined in Eq.7.1 and λ_2 is the second largest eigenvalue, the graph is an expander. In graph theory[87, 84, 88], a graph is said to be an expander if the number of edges between any possible subsection is a non-vanishing function

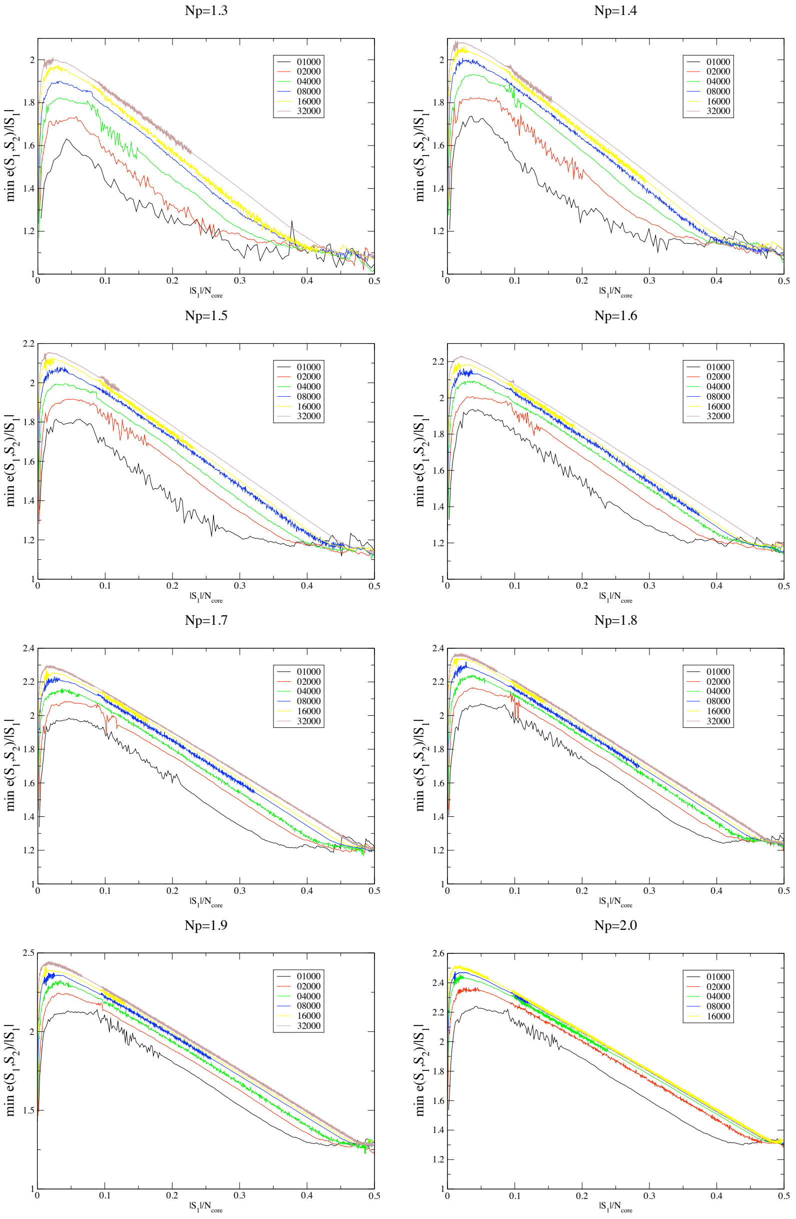

Figure 7.8: Average minimum cost per node removed from the core.

of the total number of nodes in that partition. We plot the spectral gap as a function of Np in Fig. 7.7 for different graph sizes. The size of the spectral gap is always significantly smaller than 1 for all values of Np considered, much smaller than our measured value for r, thus providing one more indication that the graph is highly connected and that an optimal solution will try to avoid cutting it at all costs.

7.4 Discussion

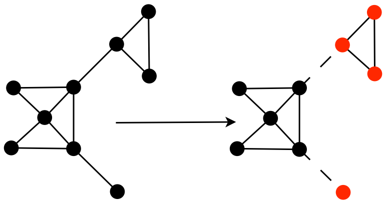

Figure 7.9: Dangling unicyclical components.

Graph bisection is an important problem with a wide variety of application in diverse technological and theoretical fields. The numerical measurements we have performed have given us a great deal of insight in to the structure of the phase space of this problem. The theoretical results we were able to confirm give us a good degree of confidence that our software is accurate and that the observations that we made that are not yet supported by analytical results are well grounded and indicate the true behavior of this type of system.

Our results indicate that our core peeling algorithm can provide a feasible solution to this problem for average connectivities less than 2 and that it breaks down shortly there after. This is due to the evolution of the GCC of $\mathcal{G}(N,p)$ graphs for increasing values of Np. For $Np \leq \log 4$ the GCC is relatively small and we must use other components in order to evenly split the graph in two, sometimes having to cut one of the smaller ones. Above $\log 4$and below 2 the GCC is larger than $N/2$ and we must cut some of the trees that it has surrounding the 2-core. Finally, above $Np \equiv 2$, the $2-$core becomes larger than $N/2$ and we can no longer find a feasible solution that cuts only trees.

We also considered the possibility of cutting the core as a way of both improving our solution in the interval $\log 4 \leq Np \leq 2$ and possibly extending it to larger values. The simulations indicate that the $2-$core is an graph expander which precludes its use as an alternative to tree cutting and that seems to indicate that all optimal solutions in this range will try not to split the core. For $Np > 2$ we must, in effect, cut the core in order to find a feasible solution, but the details of this regime have not been explored and should be the object of future work.

As a final idea, we also tried looking for unicyclical components connected to the $2-$core as shown in Fig. 7.9. Although we can remove each of these components by cutting a single edge, the are still part of the core since all the nodes that are part of them have mean degree greater or equal to 2. Also, since they should be relatively rare, our minimization procedure described above might have missed them, but we were hopeful that they might still make a relevant contribution. The truth, however, is much different. In our simulations we were unable to find a sufficient number of them to make any type of significant difference. Since this was our most recent attempt, we are still looking at the source code to make sure that it is completely bug free and that this absence is not an artifact of deficient programming.

Finally, we also analyzed the time complexity of our implementation as shown in Fig. 7.10. Both the tree cutting procedure (red line) and the core cutting analysis (black line) seem to have run times of $O\left(N^2\right)$ as expected from our description of the respective algorithms. We should not be able to make significant improvements on this overall behavior due to the nature of the heuristics used, but there might be room for improvement in the constant multiplicative factors that might allow us to explore larger system sizes.

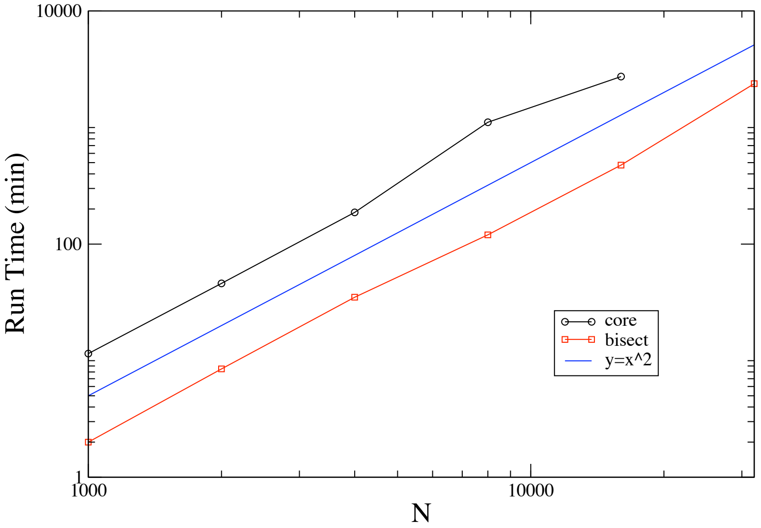

Figure 7.10: Time complexity. The central line has slope 2 and is meant as a guide to the eye.

Chapter 8

Weight Correlations

Complex networks have proved to be useful tools to explore natural or man-made phenomena as diverse as the Internet [89], human societies [90], transport patterns between airports [91, 92] or even metabolic reactions in the interior of cells [93]. The vertices in the networks represent the elements of the system and the edges the interactions between them. The study of the topology of the network provides valuable information on how the basic components interact. While the existence or not of an edge is already informative, in many cases, as those listed above, the interactions can appear on different levels. The bandwidth between two servers on the Internet, for instance, is not a flat quantity equal for all pairs, it depends on the importance of the servers as well as on the traffic expected. This fact led to the introduction of weighted graphs as a more accurate way to describe real networks [94, 95, 96, 97]. Weighted graphs are complex networks where the edges have a magnitude associated, a *weight*. The weight accounts for the quality of the connection between two vertices. The existence of a distribution of weights dramatically alters transport properties of networks like the geometry of the optimal paths [98, 99, 83, 100], the spreading of diseases [101] or the synchronizability of oscillators [102]. Most of these studies have been carried out on networks with uncorrelated weights on neighboring edges (those arriving at the same node) even though most real cases possess correlation. Our aim here is to check how the presence of correlations can influence these results.

There may be several kinds of correlations in random graphs [103]. Recently, it has been shown that the edge weights in some real-world networks are related to other properties of the graph such as the degree (the number of connections a vertex has) [91, 104]. The weights were found to follow, on average, a power law dependence on the degree. Several theoretical mechanisms have been proposed to generate networks of this type [105, 99, 106]. In this case, a clear correlation is introduced between the weight of neighboring edges but one may ask if that is the only possible kind of weight correlations. If not, which type of structures can be observed? How can the correlations be quantitatively characterized? And most importantly, which is the influence that they have on the transport properties of the graph?

In this chapter, we address these questions. First, we construct a model that allows us to explore the different possibilities for the weight correlations independently of other properties of the network. Next, we study how their presence affects the transport properties. And finally, we show that these correlations are present in two examples of real-world networks: the IMDB actor collaboration network and the traffic network between US airports.

8.1 Weight Correlation Model

Let us start by defining a mathematical framework for the weight correlations. From the point of view of an edge of weight w with vertices with degree k and k' at its extremes, the joint probability that its neighboring edges have a certain weight is given by

$$P_{k\,k'}(w, w_1, \ldots, w_{k-1}, w'_1, \ldots, w'_{k'-1}) \tag{8.1}$$

These functions contain all the information about both degree and weight distributions and correlations. Here, we will focus only on the simplest possible scenario. In the same way the Markovian condition is a simplifying assumption for stochastic processes, we will consider only correlations generated by two-point joint probability functions $P_{k\,k'}(w, w')$, and, among those, initially only the ones that are degree independent given by functions of the type $P(w, w')$. There can be, of course, other effects that are missed by this simplification but the results within this restricted framework are already interesting. It should be noted that other possibilities, such as wieght degree correlations, have been already studied in the

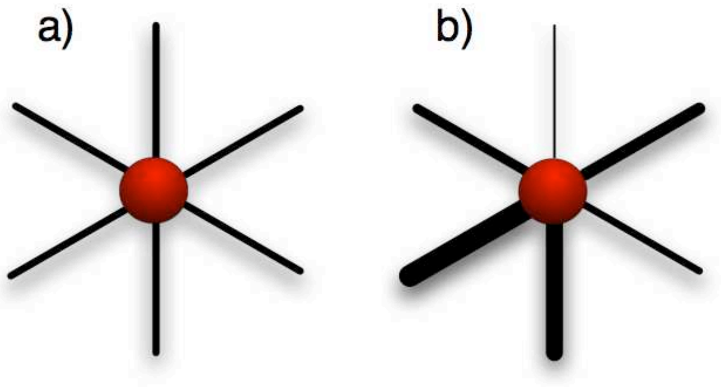

Figure 8.1: Two possible cases in networks with correlation in the link weight: a) positively correlated nets and b) negatively correlated networks. The width of the line of the link represents the value of the weight

literature[91, 106, 103, 100, 107].

In order to construct weighted networks along these lines, we use the so-called Barabási-Albert (BA) model [7], where new nodes entering the network connect to old ones with a probability proportional to their degree [1]. The networks generated by this model are scale-free (their degree distribution goes as $P_k(k) \sim k^{-3}$), have no degree-degree correlations, and their clustering coefficient (probability of finding triangles) tends to zero when the system size tends to infinity. All this makes them ideal null models to test correlations between edge weights. Once the network is grown, a joint probability distribution for the link weights $P(w, w')$ is needed. With this function one can calculate the weight distribution:

$$P\left(\omega\right) = \int d\omega' P\left(\omega, \omega'\right) \tag{8.2}$$

and the conditional probability of having a weight w' provided that a neighboring link has a weight w,

$$P\left(\omega' | \omega\right) = \frac{P\left(\omega, \omega'\right)}{P\left(\omega\right)} \tag{8.3}$$

We start by choosing an edge at random and giving it a weight obtained from $P\left(\omega\right)$. Then we move to the nodes at its extremes and assign weights to the neighboring links. To do this, we follow a recursive method: if the edge from which the node is accessed has a weight

[1] We also tested other growth models but the results do not change significantly.

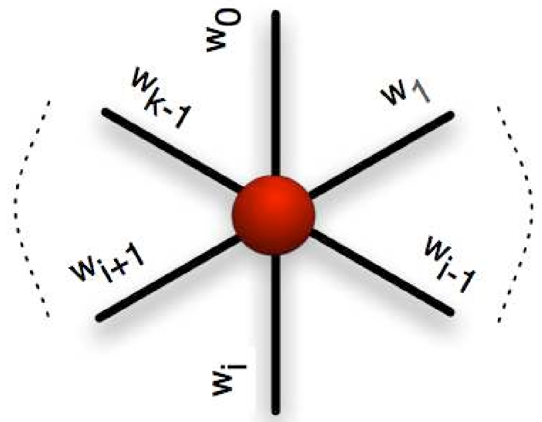

Figure 8.2: Assigning correlated weights to a network.

w_0, the rest, $w_1, \ldots, w_{k-1}$, are obtained from the conditional distributions $P\left(\omega_i|\omega_{i-1}\right)$. The recursion is necessary to increase the variability in case of negative correlations (see below). If any of the links, j, has already a weight, it remains without change and its value affects the subsequent edges $j+1, \ldots k-1$. We repeat this process until all the edges of the network have a weight [2]. For $P\left(\omega, \omega'\right)$, we consider three possibilities:

$$\begin{aligned} &P_C\left(\omega, \omega'\right) \sim \left(\omega + \omega'\right)^{-2-\alpha}, \\ &P_A\left(\omega, \omega'\right) \sim \left(\omega\, \omega' + 1\right)^{-1-\alpha}, \\ &P_U\left(\omega, \omega'\right) \sim \left(\omega\, \omega'\right)^{-1-\alpha} \end{aligned} \tag{8.4}$$

These functions are normalized in the domain $(1, \infty)$ and generate a weight distribution that asymptotically, decays as $P(w) \sim w^{-1-\alpha}$, for $\alpha > 0$. The reason to use power-law decaying functions is that this is the type of distributions most commonly observed in empirical networks. We named the distributions C (correlated), A (negatively correlated) and U (uncorrelated) because the average weight, $\langle w' \rangle(w) = \int dw'\, w'\, P(w'|w)$, obtained with the conditional probabilities from a certain seed w grows as $\langle w' \rangle_C(w) \sim w$, decreases as $\langle w' \rangle_A(w) \sim \alpha/(\alpha - 1) + 1/w$ and remains constant, respectively. This means that in C networks the links of each node tend to be relatively uniform in the weights (see Fig. 1a), with separate areas of the graph concentrating the strong or the weak links, while in the A

[2] After the assignment process is over, we have verified that the weights are distributed as $P(\omega)$ for every case.

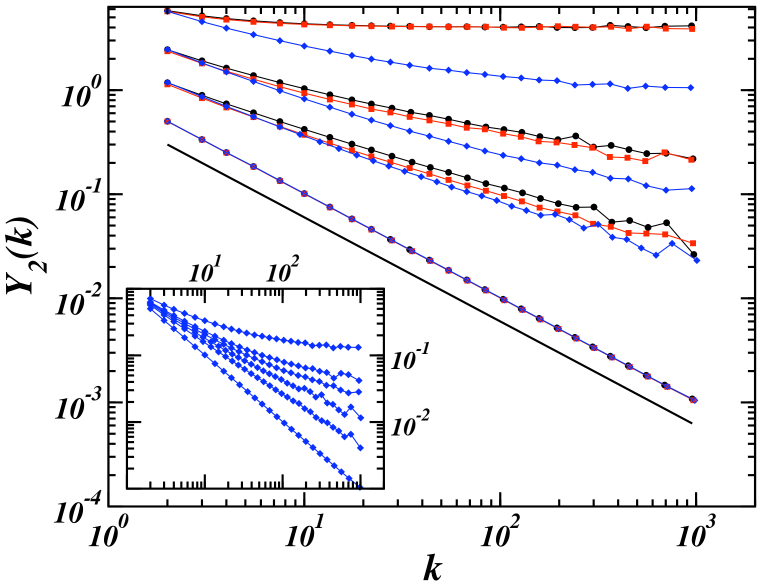

Figure 8.3: Plot of the disparity as a function of the degree, the triplets represent the graphs generated using the correlated (blue diamonds), the uncorrelated (red squares) and negatively correlated (black circles) distributions. Each triplet corresponds to a different value of α, from bottom to top $\alpha = 20, 1.5, 1.2$ and 0.5. The straight line has slope -1 and is meant as a guide to the eye. The triplets have been shifted upwards to improve visibility. The inset shows the disparity versus k for the correlated case and for the values of α from bottom to top $\alpha = 20, 2, 1.5, 1.2, 1$ and 0.5

case links with high and low weights are heavily mixed (more than randomly, see Fig. 1b).

From a numerical point of view, we have checked how the variables to measure vary with the network size N. In the following, we show the results for graphs with $N = 10^5$, which is big enough to avoid finite size effects. For each value of the exponent α (from Eq. (8.4) and for each type of correlations, we have averaged over 300 realizations. Note that we use α as a control parameter for the strength of the correlations. For high values of α, $P(w)$ decays very fast and the correlations become negligible, all links have essentially the same weight. For $\alpha < 2$ however, the second moment of $P(w)$ diverges and the effect of correlations becomes stronger, increasing in intensity when $\alpha \to 0$ (see inset Fig. 3).

8.2 Correlation Measurements

To gain insight on the correlations, we need to compare magnitudes measured in correlated graphs with those obtained from uncorrelated networks. Two magnitudes are specially suited to study weight correlations. The first, the *disparity*, introduced by Barthélemy *et al.* [108], which for a node i is defined as

$$Y_2(i) = \frac{\sum_{j\in\nu(i)} w_{ij}^2}{(\sum_{j\in\nu(i)} w_{ij})^2}, \tag{8.5}$$

where $\nu(i)$ is the set of neighbors of i. This quantity measures the diversity of the edge weights. If all the links of a node have a similar weight, their value must be $w \approx s/k$, and therefore the disparity decays as $Y_2(k) \sim 1/k$. On the other hand, if the vertex strength is essentially due to the weight of a single edge, $Y_2(k)$ must tend to a constant. In Figure 2, $Y_2(k)$ is displayed versus k for a set of values of α and for the three cases A, U and C. For very high α, the correlations are irrelevant, the three curves fall on top of each other. When α decreases below 2, we observe that $Y_2(k)$ separate for C, A and U networks. Below $\alpha \approx 1$, A and U cases are seldomly differentiable though C lies far below. The second magnitude that can be used to estimate the type and intensity of correlations is the *range* defined for a node i as

$$r_i = \frac{w_{max}(i) - w_{min}(i)}{w_{max}(i) + w_{min}(i)}, \tag{8.6}$$

where $w_{max}(i)$ and $w_{min}(i)$ are respectively the maximum and minimum weights of the edges of i. One can then average r_i and $Y_2(i)$ over all the nodes of the network and obtain $\langle r\rangle$ and $\langle Y_2\rangle$. The value of these two parameters must be compared with those from uncorrelated networks to quantify the intensity of the correlations. The most accurate procedure is to disorder the original network: randomly interchange the values of the edge weights preserving thus the weight distribution $P(w)$, measure then $\langle r\rangle_{rand}$ and $\langle Y_2\rangle_{rand}$ and estimate

$$\rho = \langle r\rangle_{orig}/\langle r\rangle_{rand} \ \ , \ \ \Upsilon = \langle Y_2\rangle_{orig}/\langle Y_2\rangle_{rand} \ . \tag{8.7}$$

If $\rho < 1$ ($\Upsilon < 1$), the network displays positive weight correlations. The stronger the correlations are, the smaller ρ becomes. Otherwise, if $\rho > 1$ ($\Upsilon > 1$), the weights are negatively correlated. $\rho = \Upsilon = 1$ is the limit of uncorrelated networks. The values of ρ and Υ for some examples are shown in Table 8.1. In general ρ seems more sensitive than Υ.

As previously mentioned, networks with weights generated as a function of the degree

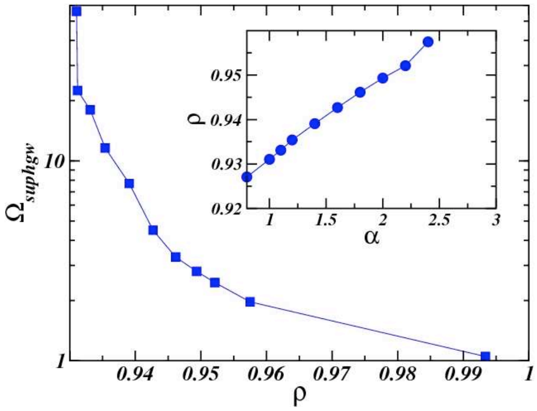

Figure 8.4: Relation between the intensity of the correlations, estimated as $\rho = \langle r\rangle(C)/\langle r\rangle(rand)$, and the change in the weight of the superhighways for positively correlated networks. In the inset, variation of the correlations with the exponent α.

may display quite different transport properties from their unweighted counterparts [101, 109, 110]. Usually, the weights of these networks are obtained by means of a relation of the kind $w_{ij} \sim (k_i\, k_j)^{\delta}$ [104]. The question is thus in which way a relation between weight and degree affects the weight-weight correlations. The answer (see Table 8.1) is that provided that the degree is an *a priori* characteristic that equally influence all the edges of a vertex, the weight of the links is positively correlated. The networks created in this way show correlations similar to our C case (regardless of the sign of the exponent δ) .

8.3 Transport In Networks

Let us now focus on the transport properties of the networks and how they vary with weight correlations. Several measures have been proposed to study transport [83, 111], here we will analyze two of them. First we calculate the weight of an equivalent of the Minimum Spanning Tree (MST) for each kind of network. If the number of vertices in the network is N, the MST is defined as the tree composed by $N - 1$ edges connecting all the nodes that has the minimum total weight. We have considered the maximum weight instead of the minimum when choosing the edges. There are two reasons for this: (i) our weight distributions are bounded from below, $w \geq 1$, but not from above, and (ii) in many real-world networks as the Internet where the weight represents the bandwidth, or the airport connection networks where w_{ij} is the number of seats available or passengers traveling from destination i to j, the bigger the weight is, the more efficient the transport becomes. To obtain the tree, we used an algorithm derived from Prim's method but adapted for searching the maximum weight. The rate $\Omega_{MaxST} = W_{MaxST}(orig)/W_{MaxST}(rand)$, where W_{MaxST} stands for the weight of the MST, is shown in Table 8.1 for some examples. The effect of the correlations is subtle but consistent, positively correlated networks score higher than uncorrelated or negatively correlated ones.

A second aspect related to the transport that we have studied is the size and weight of the "*superhighways*" as recently introduced by Wu *et al.* [83]. If the edges of an uncorrelated network are severed following an increasing order from small to higher weights, one eventually reaches the percolation threshold (when $\langle k^2\rangle/\langle k\rangle \approx 2$ [111]). The remaining connected graph after the process is over, the incipient percolation cluster or superhighways, holds most of the traffic of the original network. In practice, we start by disordering the weights of each correlated network, estimating then the superhighways of the randomized graphs and measuring how many edges must be cut on average to attain the percolation threshold. Next, we cut the same number of links in the correlated network (again going from lower to higher values of w) and compare the size and the weight of the biggest remaining connected cluster (W_{suphgw}) with those of the randomized graphs ($\Omega_{suphgw} = W_{suphgw}(orig)/W_{suphgw}(rand)$). The results, displayed in Table 8.1, show that, in general, positive correlations play a decisive role on the value of W_{suphgw}, increasing it by orders of magnitude. In Fig. 8.4, we show how the Ω_{suphgw} of correlated networks enhances with the intensity of correlations measured using ρ. When ρ departs from one, Ω_{suphgw} sky rockets. This fact agrees with the results obtained with Ω_{MaxST} and may be understood by keeping the analogy with the roads: the transport improves if the highways are connected together forming a communication backbone as large as possible. This result applies to our model where the weight correlations are independent of other structural factors. In general, there may be other aspects influencing the transport properties. If several compete, as it happens in the case of $w_{ij} = (k_i\, k_j)^{-1/2}$ regarding the degree distribution and the weight correlations, the transport capability of the network may suffer.

8.4 Real World Networks

Finally, we have studied a couple of real-world examples. First the IMDB movie database with 383640 actors that are connected together whenever they have shared a common movie. The weight of an edge represents here the number of times the partnerships has been repeated [7]. The second network is formed by 1278 US airports. A directed edge connects two of them whenever a flight traveled without stops between them in 2005, the weight corresponds to the number of passengers doing that travel in 2005 [112]. Since this

Network	type	$\langle Y_2 \rangle$	$\langle r \rangle$	W_{MaxST}	W_{suphgw}
$P(w, w')$	C (orig)	0.595	0.688	$5.83\,10^{13}$	$8.3\,10^{11}$
	C (rand)	0.675	0.746	$5.81\,10^{13}$	$8.0\,10^{9}$
$\alpha = 0.5$	A (orig)	0.665	0.727	$7.37\,10^{13}$	$6.3\,10^{10}$
	A (rand)	0.662	0.723	$7.37\,10^{13}$	$3.7\,10^{11}$
$\alpha = 1$	C (orig)	0.507	0.554	$6.6\,10^{6}$	$1.9\,10^{6}$
	C (rand)	0.543	0.595	$6.1\,10^{6}$	$5.8\,10^{4}$
	A (orig)	0.531	0.568	$4.6\,10^{6}$	$3.9\,10^{4}$
	A (rand)	0.528	0.562	$4.6\,10^{6}$	$3.9\,10^{4}$
$\alpha = 1.2$	C (orig)	0.486	0.515	$2.0\,10^{6}$	$3.0\,10^{5}$
	C (rand)	0.513	0.551	$1.9\,10^{6}$	$2.0\,10^{4}$
	A (orig)	0.501	0.522	$1.44\,10^{6}$	$1.3\,10^{4}$
	A (rand)	0.498	0.517	$1.45\,10^{6}$	$1.5\,10^{4}$
$\alpha = 1.5$	C (orig)	0.463	0.466	$8.37\,10^{5}$	$2.8\,10^{4}$
	C (rand)	0.468	0.496	$8.24\,10^{5}$	$5.7\,10^{3}$
	A (orig)	0.471	0.467	$6.58\,10^{5}$	$4.1\,10^{3}$
	A (rand)	0.468	0.462	$6.60\,10^{5}$	$4.4\,10^{3}$
$\alpha = 1.7$	C (orig)	0.451	0.439	$6.28\,10^{5}$	$1.2\,10^{4}$
	C (rand)	0.465	0.465	$6.20\,10^{5}$	$3.7\,10^{3}$
	A (orig)	0.456	0.436	$5.01\,10^{5}$	$2.9\,10^{3}$
	A (rand)	0.454	0.431	$5.02\,10^{5}$	$2.9\,10^{3}$
$\omega_{ij} = (k_i\, k_j)^{1/2}$	orig	0.431	0.383	$1.37\,10^{6}$	$5.2\,10^{5}$
	rand	0.459	0.461	$1.34\,10^{6}$	$6.9\,10^{3}$
$\omega_{ij} = (k_i\, k_j)^{-1/2}$	orig	0.417	0.383	$2.88\,10^{4}$	0.83
	rand	0.440	0.461	$2.89\,10^{3}$	57
actors	orig	0.060	0.102	$1.33\,10^{6}$	$2.4\,10^{5}$
	rand	0.066	0.407	$1.29\,10^{6}$	$7.4\,10^{4}$
airports	orig	0.623	0.956	$3.20\,10^{8}$	$3.70\,10^{8}$
	rand	0.648	0.972	$3.13\,10^{8}$	$3.06\,10^{8}$

Table 8.1: Topological and transport parameters of the networks for $N = 10^5$.

network is directed, the magnitudes shown in Tab. I refer only to the outgoing direction. To calculate the superhighways, one has to generalize the method to directed graphs and study the incipient strongly connected component (SCC) instead of the incipient percolation cluster. Both networks exhibit positive correlations on the edge weights and their transport properties are significantly affected by the correlations (see Ω values).

In summary, we have explored how correlations between edge weights can occur in random networks. For the simplest case of correlations between pair of neighboring links, there are two possible scenarios apart from the absence of correlation altogether. We have shown how those correlations can be characterized by the use of the disparity and the range. We have also studied the kind of weight correlations introduced by degree dependent weights, positive and equivalent to our C model. In all cases, the transport properties of the networks have been analyzed, showing that in general positive correlations improve the transport properties. Finally, we have also presented data from two real-world networks on professional collaborations and on airport connections, indicating that weight-weight correlations are an important factor to take into account in the study of real networks.

8.5 Publications

[113] J. J. Ramasco and B. Gonçalves, Transport on weighted Networks: when correlations are independent of degree, *Phys. Rev. E*, 76:066106 (2007)

Chapter 9

Web Surfing

9.1 The World Wide Web (WWW)

In 1969, the U.S Department of Defense (DoD) secretly conceived a feasibility plan whose main purpose was to use dynamic rerouting of messages to enable the few computer networks in existence at the time to survive a direct military (in particular nuclear) attack. What at the time was called ARPAnet (Advanced Research Projects Agency network) was created by interconnecting three networks in California and one in Utah using the Internet Protocol (IP). Three years later, in 1972, when this project was revealed to the public, it already included 50 different institutions among universities and research organizations involved in defense projects.

A year later, England and Norway established the first international connections to ARPAnet. A decade later, the National Science Foundation (NSF) created the NSFnet to interconnect five supercomputing centers, using a new version of the IP protocol known as TCP/IP (Transmission Control Protocol/Internet Protocol). By the end of the 1980s, the DoD decided that the network was sufficiently developed and had achieved its initial purposes. It decided to stop funding of its backbone infrastructure that was gradually shut down, until the last node went offline in 1989. NFS then replaced ARPAnet as the fundamental backbone of what was already known as the Internet.

In 1989, Tim Berners Lee at the Centre Européenne pour la Recherche Nucléaire (CERN) started working on what was originally referred to as a ”large hypertext database” as a way of allowing high energy physicists to easily share data among themselves. A year later, ”WorldWideWeb”, the first browser and editor and ”info.cern.ch”, the first server were up and running within the confines of the research institution. After it became publicly available through the Internet, in 1991, the ”World Wide Web” exploded to become a global network with several billion web pages and hundreds of millions of users scattered all over the world. It became a fundamental part of the way western society functions, permitting a broader exchange of ideas than ever before. In this chapter we study the behavior of users as they visit web pages hosted by a University web server.

9.2 Network Structure

Each time a user clicks on a link or types in the address on address bar of an Internet browser, a request for that web page is sent to the corresponding web server which then returns the appropriate page to the requesting browser. For optimization and accounting reasons, most web server software keep track of which pages were requested, when, by whom and, in some cases, the page that contained the link that the user clicked[1].

Our dataset is constituted by the log files generated by the Apache web server running in the *emory.edu* machine and responsible for all of Emory University's web content. The system administrators made available to us the complete set of page access logs for the period between April 1st, 2005 and January 17, 2006. Due to privacy concerns, each of the IP addresses in each of the 294 log files was replaced by a private IP of the form 10.x.x.x, in such a way that the same original IP always maps to the same private IP and that no two original IPs are mapped to the same private one (and vice-versa). This ensures that we can still analyze the behavior of each individual IP without introducing any systematic errors. The final dataset had $3,179,671$ IPs, $2,562,398$ unique URL's from the *emory.edu* domain, $3,293,916$ referring URL's from all over the WWW and $53,582,121$ individual page requests, each with the date and time at which it occurred.

To the best of our knowledge, this is the largest dataset of this type that has been studied in the literature, and this is the first time this type of study has been conducted.

[1] This information must be supplied by the users browser and, as such, is unreliable and not always available to the web server.

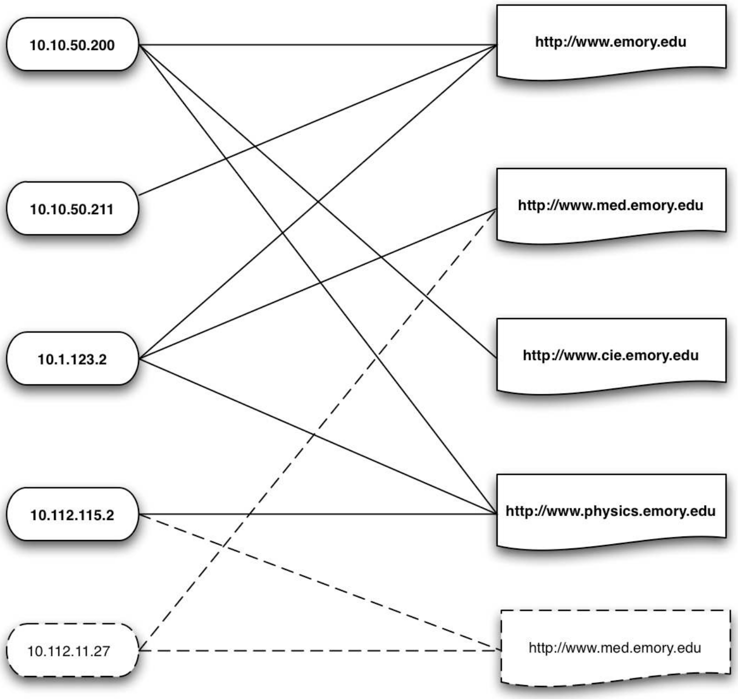

Figure 9.1: Bipartite structure of the IP-URL interaction graph. Dashed lines represent new nodes and edges that will be created on the next step.

The predominantly academic nature of the server that generated these logs results in a few inevitable ideossyncracies in the data. In particular, we expect outbursts of activity in periods corresponding to the beginning of semesters and decreases during weekends and holidays. However, we don't expect these peculiarities to affect the quality of the results and conclusions that might be drawn from this analysis.

One should also note that we make no effort to completely map the network structure generated by the links between the web pages in the server. Instead, we rely solely on the behavior of the different individuals to "show" us which pages exist by visiting them directly. If for some reason an existing web page receives no visits during the sampling period, then we do not include it in the analysis. This implies, among other things, that web pages are added to our view of the network when they are visited for the first time and not when they are initially created. As noted in the literature [114, 115] sampled networks can display different properties than the complete ones, but in this as in most cases, there is no other choice than to use the data available to us. We are also more interested in understanding the average behavior of individual web surfers than in thoroughly studying the properties

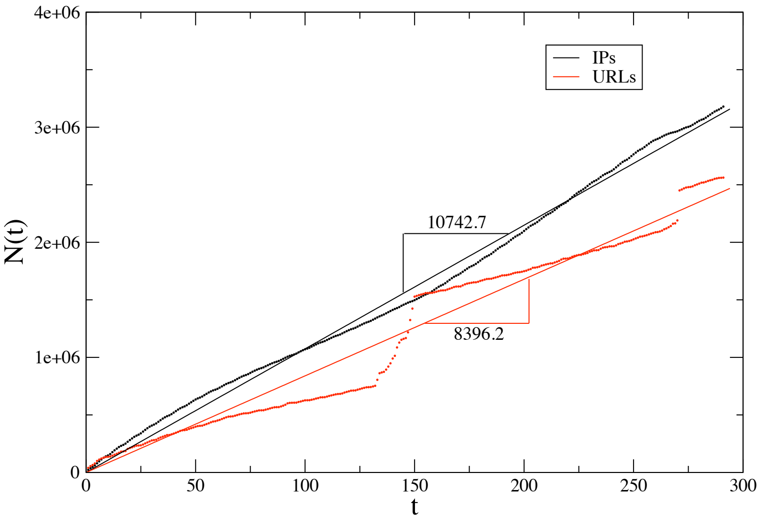

Figure 9.2: Number of IPs and URLs present in the graph during the observation period. Linear fits for both curves are also displayed.

of the resulting network, which gives us an extra measure of confidence on the results we might obtain.

Our interaction graph has two distinct types of nodes, as shown in Fig. 9.1. On one hand, we have IP nodes representing individual users[2], and on the other we have URL nodes representing individual web pages. The first time a user visits a web page a new link is added to the graph connecting them with unit weight and on subsequent visits, the weight is incremented by one. New pages are added to the network when they are visited by the first time and a new user is created when he first visits one of our pages. This structure is represented in Fig. 9.1. From this description, it is clear that our network is continuously growing with new IPs and URLs being added. The evolution of the number of IPs and URLs is shown in Fig. 9.2. Clearly, the number of IPs grows faster than the number of URLs, throughout most of the sample period, but we have two bursts of web page creation corresponding to the beginning and end of the summer semester, respectively. This is one of the ideossyncracies we were expecting to observe. We smooth out this behavior by performing a linear fit to the full data set, obtaining, in effect, an average number of pages created per day.

[2] Throughout this work, we make the underlying assumption that each user has a unique IP address that does not change over the sampling period. However, some ISPs assign a different IP address each time a user connects to the Internet for privacy reasons and the problem of uniquely identifying a user across multiple addresses is still subject to much debate and study[116].

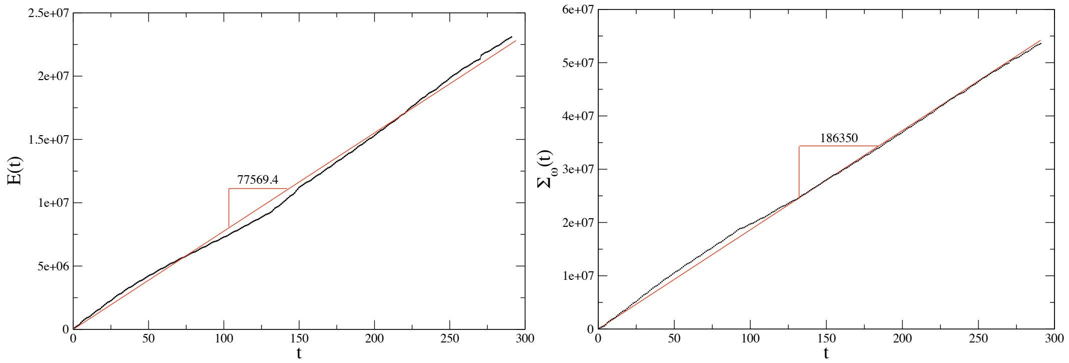

Figure 9.3: left) Evolution of the number of edges connecting IPs and URLs. right) Evolution of the total weight in the system.

New connections between users and pages are also added on a daily basis as users visit different sites on different days, and we represent the total number of edges as a function of time on the left side of Fig. 9.3. We account for the possibility of a user visiting the same page several times by attributing weights to the links connecting them, where the weight on an edge connecting an IP with an URL is simply the number of times that page was visited by that machine[3]. By comparing the number of links with the sum of all the weights in the network, one can get a measure of the relative importance of these two alternative processes, the creating of new links and the reusing of previously existing ones. If both quantities behave similarly, than connections are usually used only once, but if the total weight, $\Sigma_\omega\left(t\right)$, grows faster than the number of links, then we can safely conclude than connections are reused several times. We plot $\Sigma_\omega\left(t\right)$ on the right side of Fig. 9.3. The slope of this line is ≈ 2.4 times the slope of $E\left(t\right)$ indicating that each page is viewed, on average, 2.4 times by each of its visitors, over the period of time considered. This surprisingly low number is another signature of the specific nature of the web pages considered and we should see much larger average weight, $\langle\omega\rangle$, on other types of web servers[4].

The weight frequency distribution, $P\left(\omega\right)$, displays a clear power law form with an exponent, $\delta \approx 2.$, for over two decades. This indicates that, although, the vast majority of users

[3] This is equivalent to allowing multiple links between pairs of nodes and measuring the importance of the connection by simple counting how many links connect a given pair, but has the advantage of being more memory efficient. Both alternatives are used in the literature with the choice between one and the other being one determined by convenience of analysis in the situation of interest.

[4] Consider how many times you are likely to visit *www.google.com* or *www.cnn.com* over the course of almost a year.

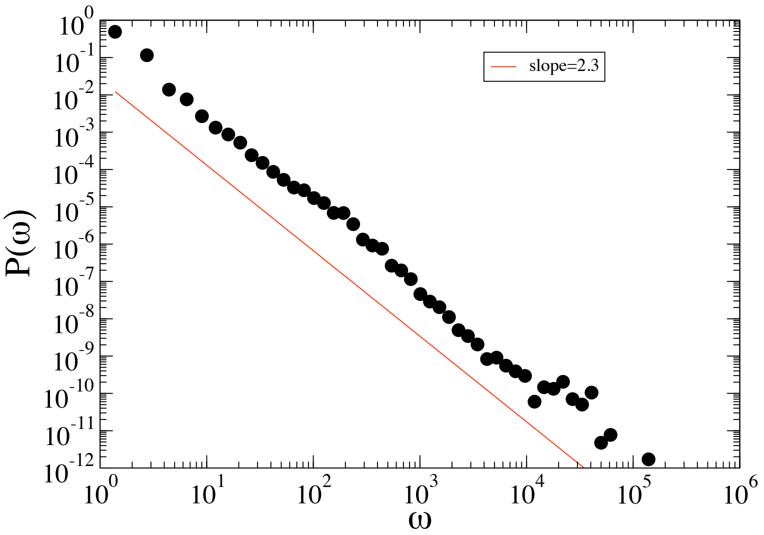

Figure 9.4: Probability distribution of the weights.

visits each page less than 10 times, some call up the same URL on their web sites several hundreds or even thousands of times. These are probably users who set their browsers home page to the university's or their departments web sites, with the largest weights being attributable to more active web surfers.

Even though the weight distribution can provide us with some information about some aspects of the behavior of some users, it is of limited use to understand the structure of the network. We can define a few other quantities that are more informative and of more practical use for web masters and web server administrators. Of particular practical interest are, how many unique users visit a given web page, how many web pages does a given user visit, how many total visits does a page receive (how many "page views"), and how many total pages are viewed by a user. These quantities correspond, in graph theory jargon, to the connectivity of URLs and IPs and the strengths, or total weight, of each URL and IP, respectively. We represent the connectivity distribution, $P(k)$, for both IPs and URL's on the left side of Fig. 9.5 and the corresponding strength distributions on the right side of Fig. 9.5. All of these distributions clearly display a scale-free behavior with exponents of 2.2 and 2.0, respectively.

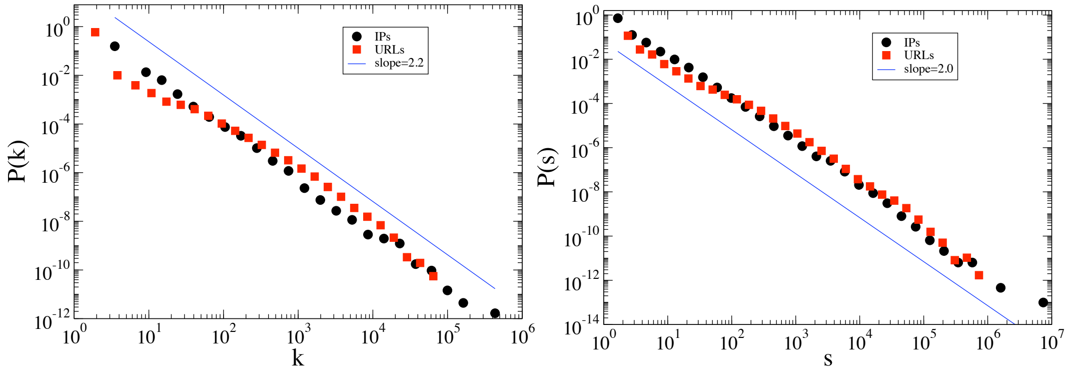

Figure 9.5: left) Distribution of the number of unique visitors per web page and number of unique web pages visited per user. right) Distribution of the total number of hits per web page and the total number of page requests per user.

9.3 Preferential Linking

Another aspect that is worth to explore in the dynamics of our database is whether the new connections or new clicks follow a preferential rule or not. Preferential linking or the "rich get richer" effect is a relatively old concept considered previously in a socio-economical context by E.H. Simon [117] in 1955 and as part of a theory of evolution by G. U. Yule[118] in 1924. In the area of graphs theory, it was introduced in 1999 [7] with a problem inspired precisely in the hyperlinks of the WWW (see also [119, 120]). A few years have passed, and although several attempts have been made to check the existence of preferential linking in a variety of systems [121, 122], as far as we know, a systematic study of preferentiality on the user-web relationship is still missing. To be precise, if the variable under consideration x can change in time for each element of the system, it is said that it shows preferentiality if the variation follows on average an expression of the type

$$\langle \Delta x_i \rangle \approx Ax + B, \tag{9.1}$$

where x_i is the value of x for the element i, the average $\langle . \rangle$ is taken over all i, and A and B are constants. This mechanism supposes that if the update refers to quantities such as number of connections or number of clicks of a site, the probability that a particular site is chosen to update is proportional to the number of connections or clicks that it has previously accumulated. The more popular sites concentrate thus higher attention leading

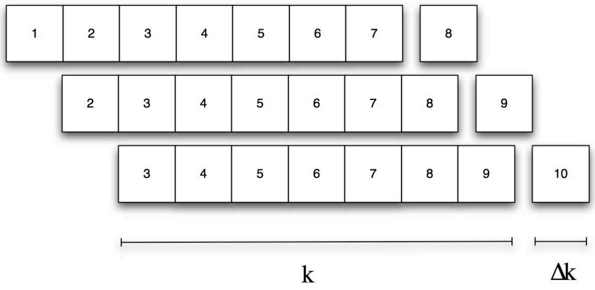

Figure 9.6: Running average calculation of the variation of the weight as a function of the weight. Each numbered box corresponds to a day of data. We use a sequence of 7 days as our initial network and measure the variation due to adding the next day.

to an agglomeration process that, after a while, produces a very wide distribution of values of x in the system. Provided that the relation of Eq. (9.1) is linear, the distribution $P_x(x)$ can be approached by a decaying power-law function with an exponent depending on the values of A and B [123]. If the relation is not linear, on the other hand, two scenarios can be found. Either $\Delta_t x$ grows with x faster than linear and the most popular element will eventually congregate a finite fraction of all the value of x available in the system, or it is sublinear and the distribution of values of x will not be so wide (stretched exponential instead of a power-law) [124, 125, 123].

To verify whether or not this process occurs in our system, we measure the average variation of degree as a function of the original degree, $\Delta k\left(k\right)$, by considering 7 consecutive days of data to be our initial network and calculating the variation caused by adding in the next day. This is done for all consecutive week long periods, as shown in Fig. 9.6. The reason to use a period of a week is simple. A close look at Fig. 9.3 reveals a variation between business days and weekends, with less traffic during the weekend than during the week. To avoid skewing the results, we use a period of time that is maximally uniform by always including exactly one weekend.

In the form introduced by *Barabasi et al*, preferential attachment consists of having the probability of a node with degree k attracting a new link to be of the form ak, where a is a normalization constant which results, necessarily, on a degree distribution $P\left(k\right) \sim k^{-3}$. However, as we saw previously, our degree distribution follows a different exponent. We plot in Fig. 9.7 our measurements for $\Delta k\left(k\right)$ using logarithmic bins. The attachment function obtained is of the form $ak + b$, thus deviating from the original *Barabasi* model and its predictions, but maintaining the basic principle that ”popularity is attractive”, or, in other words, that the more users a web site has, the more rapidly it increases that number, leading to an explosive effect. To illustrate this, consider one simple example. Two new web sites come online at the same time, all things, being equal, one would expect that they would attract users at the same pace. If however, one of the web sites gets a slightly larger number of users, that small initial difference will become increasingly larger and larger as time goes on. For a web based company, this can mean the difference between profit and bankruptcy. One also should notice an important detail, that for URLs, the constant b is positive, while for IPs, the constant is negative[126]. This apparently insignificant difference between the two types of nodes, signals an important difference in behavior. A positive constant, as in the case of the URLs, represents an intrinsic ”attractiveness” of the node, or a certain degree of quality that attracts new visitors. On the other hand, the negative constant exhibited by the IP data represents a threshold that must be overcome. IPs with degree less than $|b|$ are unlikely to create new connections, while IPs with connectivities above this value continuously increase it in a linear preferential fashion.

Another quantity we are interested in analyzing due to its practical relevance is the ”stickiness” of a web page, or in other words, how many times you will revisit a page, given that you have already visited it x times. As we noted earlier, the number of times a user visits a page is the weight of that edge, so ”stickiness” is the variation of the weight as a function of the weight, $\Delta\omega\left(\omega\right)$. Following the same approach we used for $\Delta k\left(k\right)$, illustrated

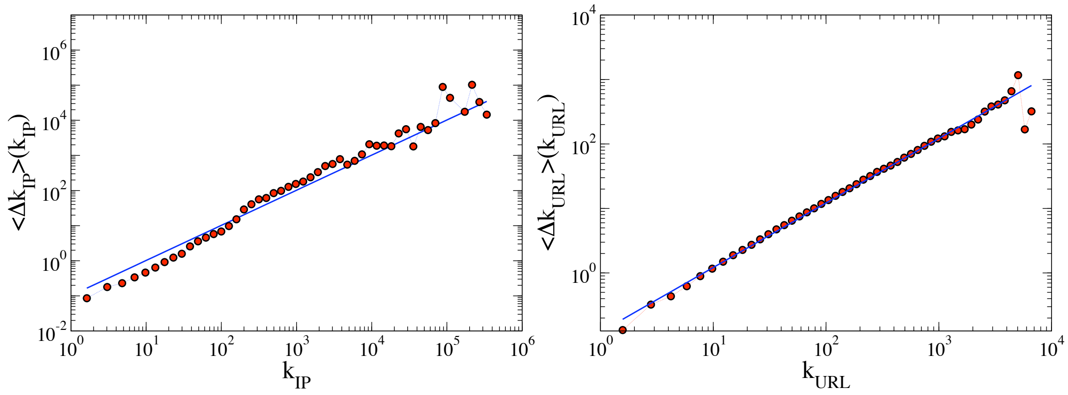

Figure 9.7: Change in connectivity as a function of connectivity. left) IPs right) URLs

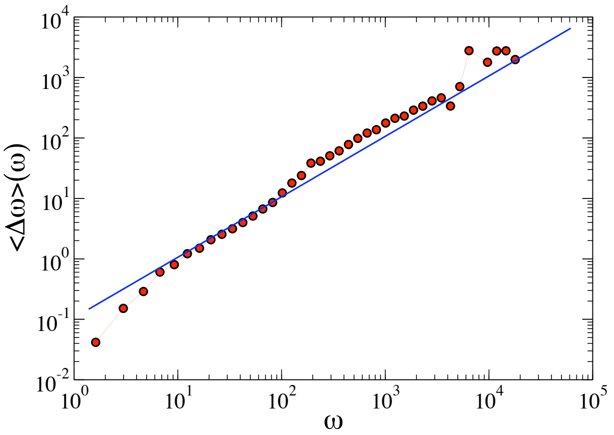

Figure 9.8: Variation of the weight as a function of the weight.

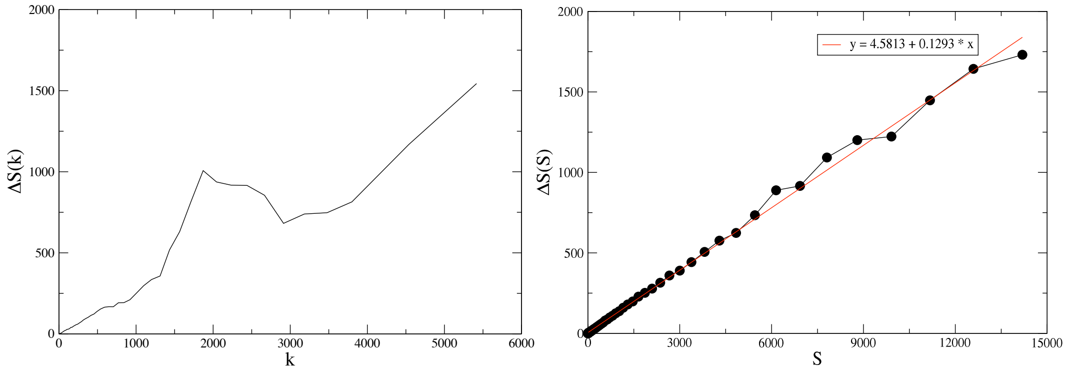

Figure 9.9: Variation of a web page's traffic. left) As a function of the number of unique visitors. right) As a function of the total traffic.

in Fig. 9.6, and applying a logarithmic binning, we obtain the results shown in Fig. 9.8. where the linear behavior is clear.

This linear dependency implies a sort of habit forming in users. The higher the number of times a user has visited a page in the recent past, the more times he will visit it in the future, which might help explain why web sites that become dominant are rarely overcome by others, even if the new comers are ”better” in some sense. In web surfing, as in other things, people are creatures of habit.

9.4 Non-stationarities

The linear relations we saw in the previous section have important implications for the structure of the navigation patterns, turning the distribution of values of k_{IP}, $P_{IP}(k_{IP})$,

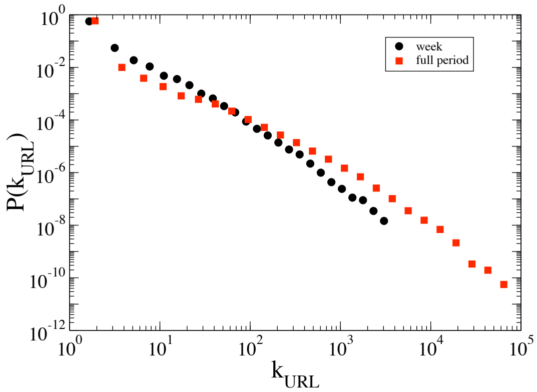

Figure 9.10: URL degree distribution for a period of a week and for the full period. The red points are the same as in Fig. 9.5.

quite wide and well fitted by the power-law $\sim k^{-2.2}$. In addition, the distribution is stationary: Its functional form does not vary if k_{IP} is calculated from the data in time windows comprehending one week or if its value is collected for the full 292 days. Such time stability is an important feature that renders easier the modeling of the system, although, as we will see next, it only holds for k_{IP}. It is not valid if the focus is moved onto the Web sites. As shown in Fig. 9.7, the variation of the number of visitors a site receives follows as well a linear behavior with the number of visitors already captured during the previous week. The distribution $P_{URL}(k_{URL})$ is also wide, although it can be hardly considered as a power-law but, more importantly, its functional form strongly depends on the time window during which the number of visitors is measured. In Fig. 9.10 we compare this distribution measured using the whole network, and using only one week periods and averaging over the 41 weeks in our dataset. This clearly shows that the distribution is not stationary.

Such effect can be understood by taking into account that web sites typically have an average life time. In an university, the time in which a site, or more specifically its contents, is relevant closely tracks the evolution of the academic year. In general, a similar rule should apply to all the Web sites. The life time can be more flexible, depending also on the number of visitors, but a certain loss of interest as the time passes since the first online publication can be expected [127]. A similar scenario is also observed for the average variation of the number of clicks a user makes on the same URL (see Fig. 9.10). In this case, the distribution $P_w(w)$ is not stationary either, with some user-URL connections getting an

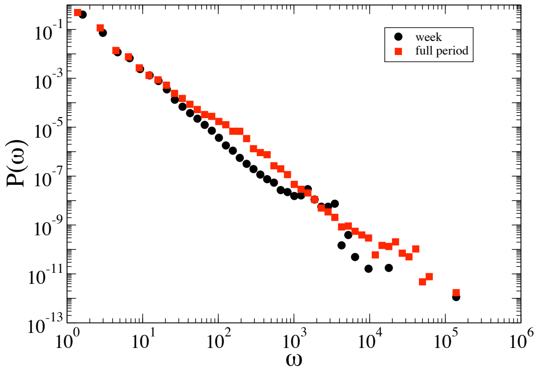

Figure 9.11: Weight distribution for a period of a week and for the full period. The red points are the same as in Fig. 9.5.

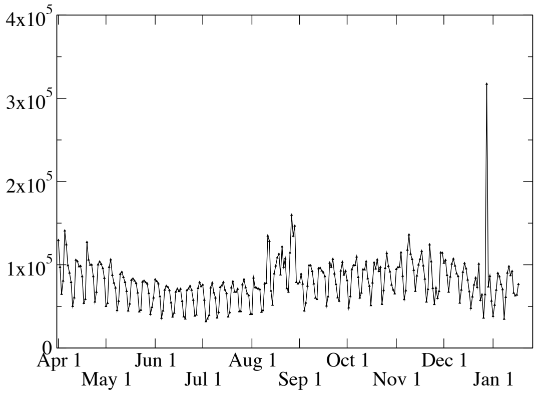

Figure 9.12: Time evolution of several quantities. The black curve represents the total number of page requests registered in a particular day. The red and green curves correspond, respectively, to the number of new users (IPs) and new web pages (URLs) accesed that day. Finally, the blue curve represents the number of people that access pages for the first time.

ever higher number of requests.

These are most likely automatic processes instructed to keep requesting the same URL over time. It is also important to note that in measuring these quantities, we have not made any special effort to separate the activity coming from humans or that from programs. The long extension of the linear increase of $\langle \Delta k_{IP} \rangle$, $\langle \Delta k_{URL} \rangle$ and $\langle \Delta w \rangle$ may imply that, in fact, no difference should be expected if the discrimination is done since humans and automatic process are majority in different parts of the domain of k_{IP}, k_{URL}, w (low values for humans, high for computer programs).

9.5 Activity Patterns of the Population

Let us also take a view of the collective behavior of the entire population during the time period we have data for. Intuitively, we expect the activity on a domain to vary from day to day, week to week and even month to month. In particular, it should be possible to observe variations in the activity, seen as the number of clicks or page searches, due to weekends, holidays and other major events that disrupt the normal life of the University. The traffic at Emory is dominated by students and teachers in the course of their professional activities and hence the major events in the course of the school year, such as the beginning and end of a semester, breaks or holidays, should be noticeable in the web traffic. In order to check this idea, the number of page requests detected per day is shown in Fig. 9.12 as a function of the observation date. One obvious feature of the figure is a clear oscillatory behavior with a period of a week. It also displays different trends for two special time of the year: one at the later part of the month of August, corresponding to the beginning of the school year, and the other at the end of December, when the semester finishes. This interpretation is also corroborated by the corresponding spikes in the number of detected pages, shown in Fig. 9.17, since at the end of August most faculty and students update their respective web sites. These new pages become less attractive after some days when they have been visited a few times.

Since accesses to this particular domain are mostly work related, we believe it justified to use of the overall traffic as a measure of the University wide "productivity". Busier days would result in larger amounts of traffic, while during holidays and weekends the number of page requests is overall smaller, thus rendering only the relative changes in traffic significant.

The averages over the page requests day by day of the week during the complete observation period are plotted together with their corresponding error bars ($2\,\sigma$, approximately. 95% confidence interval) on Figure 9.13. Our results support the old adage that after Wednesday, the hardest part of the week is already behind us, with the productivity decreasing from then on to the weekend, with Sunday being the least active day of the week. It is also interesting to note the sharp decrease of page requests on Friday and the not-so

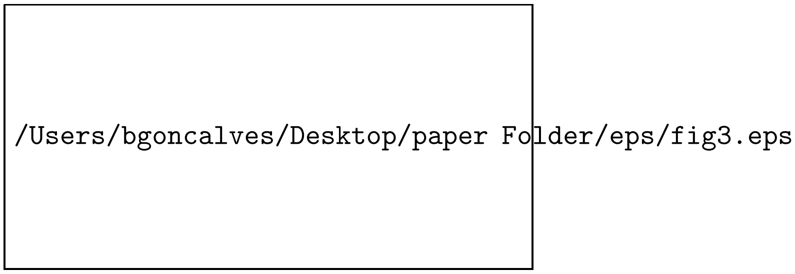

Figure 9.13: Comparison between the average week productivity and productivity during thanks giving week. The green vertical lines represent the beginning and end of the official thanks giving break at Emory University. The error bars for the average are calculated as two times the standard deviation, 2σ, or the 95% confidence interval.

active behavior of Mondays. Armed with an estimate of how productivity evolves over the week, we are now in a position to evaluate the effects of a break. In the same Figure, we also represent the data for the days surrounding Thanksgiving, arguably, one of the major holidays in the US. Traditionally, Thanksgiving recess is established to be from Thanksgiving Thursday till Sunday, so one might expect any decreases in activity to be most noticeable during that period. Indeed, this is what we observe, but we find other effects as well. Both the Wednesday before and the Monday immediately after Thanksgiving seem to be less productive than normal. This is however complemented with a slightly busier than usual days on the day before (Tuesday) and after (Tuesday) of the break.

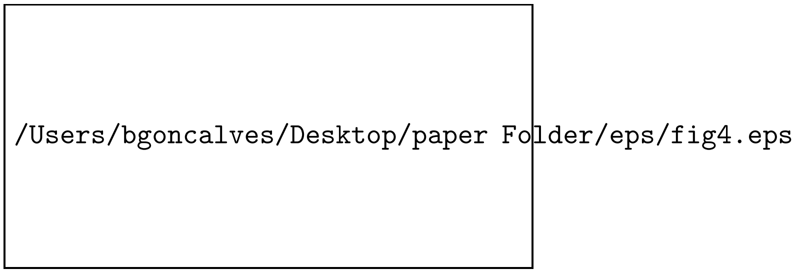

Figure 9.14: Average hourly activity in the complete Emory domain as a function of the hour of the day.

Intra day variations, with some times of the day being busier than others are also seen. By averaging the activity observed at a given hour over all the days in our data set, we obtain Fig. 9.14. Clearly, the most active period is between 7AM and 6PM. The large dip between 11AM and 14PM is due to the lunch break. The 3 hour width of that interval shows a relative wide distribution in the preferences on when to take that break. After lunch, the activity peaks reaching the higher level of the day. After 6PM activity levels off until 10PM, possibly marking the end of the workday for most students, following which it steadily decreases throughout the rest of the night, until the beginning of the new day. It is important to note that similar plots as Figs 2 and 4 have been recently reported on social network emailing (Facebook) and on cell phone usage [128]. Such ubiquity indicates the existence of important universal features (profiles) regarding human habits.

9.6 Individual Activities

Although interesting, the analysis of averages taken over the entire population has some limitations. The statistics of activity of single users are typically very wide, being in some cases well-modeled by power-law distributions with exponents smaller than 2. When this happens, it is difficult to identify a "typical" user based on such metrics[5]: while most users only visit the domain sites a few times, a significant fraction of individuals (as identified by their IP addresses) accumulate large numbers of page requests. This variability deserves a higher attention by itself since it can carry important information on the users. Figure 9.15 shows the activity patterns of three of these users. Due to privacy concerns, we cannot know the actual IPs but it is possible to deduce in part the intention of the visit based on the particular URL accessed and on the profile of the activity. In Figures 9.15*a* and 9.15*b*, the users are computer programs. One, the case shown in *a*), corresponds to a malicious attack on the payroll service of Emory. It took place on April the 4th. The profile of the number of access attempts per unit of time displays a very peculiar shape, quite regular as occurs for most automatic navigators, with a very high number of requests concentrated in a short period of time. Other, more friendly, cybernetic users are those corresponding to updating programs. An example can be seen in Figure 9.15*b* where a software site in Emory is regularly visited presumably in search for new updates. Unfortunately, this could not be verified as this particular URL no longer exists. Finally, human users show a very different activity profile from that of the machines. The activity of a human user selected at random can be seen in Figure 9.15*c*. In this case the URL is an administrative site that demands

[5] A power-law distribution decaying with an exponent equal or smaller than 2 does not have a finite average value [129]

a manual introduction of data. The activity congregates in some days followed by relative long periods of time without any request.

Given the strong variability in the activity of the human users, it is interesting to measure some statistics of it. In Figure 9.16, we have represented the histograms of the duration of the periods between requests for two different scenarios: in Fig. 9.16a for the time between consecutive visits of the same user to the same URL, $P_v(\tau_v)$, and, in Fig 9.16b, for the time between clicks by the same user to any of the sites in Emory's domain (not necessarily to the same URL), $P_c(\tau_c)$. Both distributions are rather wide.

Figure 9.15: Activity history of several individuals: a) what seems to be a malicious attack on a finance web page of the University, b) an automatic software update program, and c) a human user filling data in an administration site. The red curves represent the cumulative number of clicks. To facilitate the visualization, the scale of the cumulative and temporal number of clicks are different. The axis on the right side of each plot displays the scale for the cumulative number of clicks.

Indeed, the distribution $P_c(\tau_c)$ can be well fitted by a power-law decaying function of the type $P_c(\tau_c) \sim \tau_c^{-1.25}$. The distribution of time between consecutive visits, $P_v(\tau_v)$, decays still slower with an exponent of value -1. This latter value can be understood thanks to a model on human dynamics recently proposed by A.-L. Barabási [130] (see also [128, 131, **?**, 132, 133]). In the model, an agent has to perform a set of tasks each with a random priority assigned. A step consists in the selection of the task with the highest priority with probability p or of a random one with probability $1-p$. After the execution, a new tasks occupies the free spot in the queue. This group of rules is extremely simple but is able to reproduce a distribution of waiting times for the tasks in the queue that, in the limit of small p, decays as $\sim 1/\tau$. It can be argued that consecutive visits to the same site in Emory are equivalent to one of these tasks since many of the visits are related to work or studies, and probably bear an inherent sense of priority for each user. Also returning immediately to the same URL and reloading it is not the most common practice,

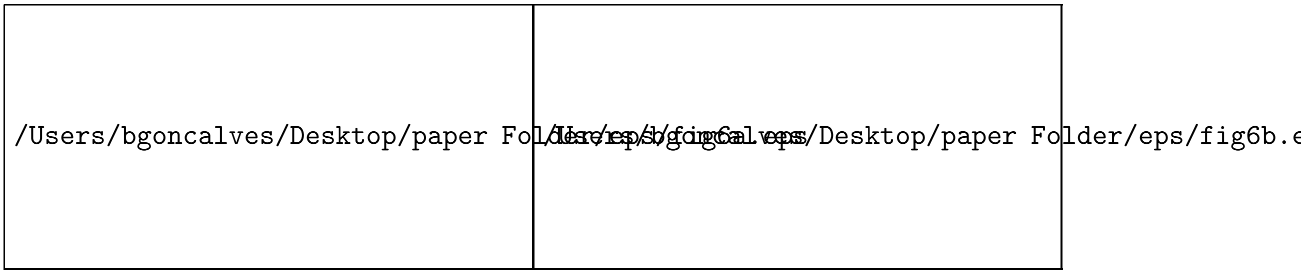
/Users/bgoncalves/Desktop/paper Folder/eps/fig6a.eps
/Users/bgoncalves/Desktop/paper Folder/eps/fig6b.e

Figure 9.16: Distribution of times between consecutive clicks: a) visits of the same user to the same URL, and b) the same user to any page of the Emory domain. The straight lines correspond to the power-law $f(\tau) \sim \tau^{-1}$ in a) and to $f(\tau) \sim \tau^{-1.25}$ in b). In the inset of b), the distribution of time in the queue is plotted for a variation of Barabási's model with a number of executed tasks per unit of time of $\nu = 3$, with $p = 0.99999$, a total of 100 tasks and 10^7 time steps. The line of the inset corresponds also to a power-law with exponent -1.25.

at least not among humans. Actually, it is important to note that if the user pushes the "back"-bottom in the browser, we are not able to detect such a move because it does not leave a trace in the weblogs. If each entrance is thus seen as a fresh start of a different task, this would explain why $P_v(\tau_v)$ decays as $\sim 1/\tau$ in the range of large values of τ that is likely populated mostly by human users. The question is then whether the shape of $P_c(\tau_c)$ admits also a simple explanation. In fact, the answer is yes, if one considers that a single click on the domain does not necessarily have to be related to the realization of a single task. Many tasks will require a (fast) succession of clicks on different sites of the domain for their completion. If the model is modified to admit the execution of ν items per iteration, the resulting distribution for the time of permanence in the queue decays as $\sim \tau_c^{-1.25}$ provided that $\nu > 2$ (see the inset of Fig. 9.16*b* for an example with $\nu = 3$). These models are oversimplifications but seem able to capture some of the essential features present in the dynamics of a large community of users leading to the existence of universal exponents.

9.7 Discussion

Weblogs have proven to be an important source of information regarding human dynamics. Here we have offered an extensive study on the medium size Web domain of Emory University, *www.emory.edu*, tracking the users in a consistent way for almost a year. A clear signal of human biorythms has been obtained as well as activity patterns that seem to be

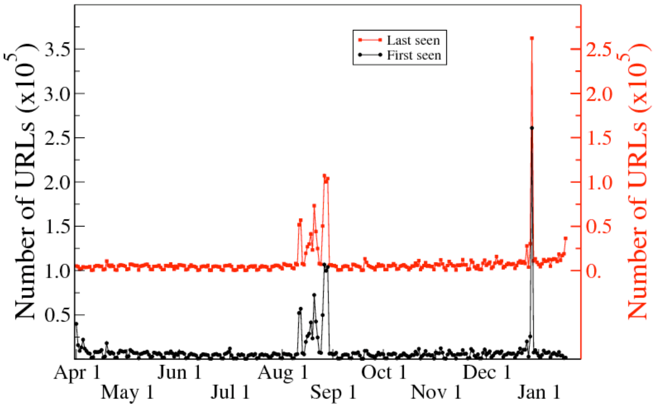

Figure 9.17: The number of URLs that are first seen, last seen, and seen for the first time in any given day. The two major "extinction" and creation events, correspond to the beginning and end of the semester as in Fig. 9.12.

universal since they are in agreement with previous results on mobile phone records or email posting in social sites. In addition, in this case, the online traffic can be directly related to the productivity of the members of the University, namely, students, professors and staff. The comparison between the activity of an ideal average week and the week containing Thanksgiving Day is revealing in this sense, with some days concentrating a large fraction of the total traffic, much higher than the average, and others falling clearly behind.

After the characterization of activity at the whole University scale, we have moved our focus down on the study of statistics of single users. The difference in the navigation patterns between humans and automatic processes, either malicious or friendly, has been highlighted. Humans are in general more unpredictable, although a similar behavior might be always reproduced by automatic means. In particular for human users, it is important to analyze the statistics of the times between events (clicks) and compare them with recently introduced models based on priority queues. We have shown that indeed such models are able to explain the inter-clicks periods distribution if the dyad user-site is considered. Furthermore, a simple modification, in which the number of tasks to execute in a short interval of time is higher than one, can also account for the statistics of times between requests of the same user on the whole Emory domain.

Finally, we have also explored another mechanism that has been proposed as an important ingredient in the development of the WWW, namely "preferential attachment". Apart from in the way the sites attract new hyperlinks, linear preferential attractiveness is in fact detected in all the aspects of the traffic contemplated: the rate of exploration of new sites by the users, the capture of new visitors by the sites or the new clicks received on each connection user-Web page. In all these cases, the linear relation holds only in short period of time. For longer periods, the life-time of the Web pages must be also taken into account, changing and complicating substantially the scenario. Preferential linking, priority queuing and Web page aging seem thus to be essential factors for any model aimed to characterize Web surfing.

9.8 Publications

[**134**] B. Gonçalves and J. J. Ramasco, Human dynamics revealed through Web analytics, *e-print cond-mat/0803.4018*, (2008)

Chapter 10

Information propagation in networks

> ”Epidemics have often been more influential than statesmen and soldiers in shaping the course of political history, and diseases may also color the moods of civilizations.”
>
> – Benjamin D'Israeli - XIX century british novelist and statesman

10.1 Epidemics

Epidemics have always had an important role in shaping the course of human history. In 1347 the bubonic plague ravaged the towns and villages of Europe causing, according to some estimates [135], the demise of one third of the continents population. Almost 200 years later, in the 1520s when the Spanish Conquistador, Francisco Pizarro faced off against the Incan empire in South America, his task was made much easier by the introduction among the natives of the European diseases such as smallpox for which their immune systems where utterly unprepared.

More recently, in the early 1980s the first cases of what was to become one of the biggest threats of the following decades were detected in the Los Angeles male homosexual community. The origins of the virus responsible for the Acquired Imuno-Deficiency Syndrome, or AIDS for short, have been traced back to Guine-Bissau in Africa [136], but its prevalence is far from local. It quickly spread to all corners of the world resulting in the infection of an estimated 50 million people of all origins, races and sexual orientations with an annual death toll of about 3 million people every year according to the World Health Organization[137].

The understanding of how diseases spread within a structured population is a fundamental requirement to stop them. Epidemiologists have been struggling with this problem for centuries, going as far back as the Greek Thucydides that described the Plague of Athens in $430-428$ B.C.[135]. In recent years, the floury of interest in Complex Networks in the physics community and its many applications has led physicists and Epidemiologists alike to become interested in the study of network based epidemiological models.

In classical epidemiological models, the total population would be divided in several classes of individuals (such as Healthy and Sick or Susceptible and Infected, respectively, in technical jargon) with all the individuals within a given class being indistinguishable. Although this approach has provided a wealth of insight in to the spreading of certain diseases in human populations, it has some limitations. In particular, it does not account for geographic and social constraints that can limit the interactions between the individuals within each of the classes considered, and only the mean field behavior can be observed.

In more recent models, and following the floury of interest in the study of complex networks, each individual can be seen as constituting its own mini-class with interactions with a given, and always individual dependent, subset of other individuals. The natural representation for this system is then that of a graph or network with a complex structure where context dependent weights can be assigned to each edge.

10.2 Broadcasting

With the coming of age of the Internet the problem of broadcasting information to all the nodes in the network as quickly and efficiently as possible gained a wide appeal with both legitimate and illegal applications. Software updates must be sent to all the computers managed by a given ISP, routing information must be assembled from all the machines available and sent to all the routers within an Autonomous Server, the start time of a distributed Denial Of Service (DOS) attack must be sent to all the drone machines constituting the bot net, a list of available files must be compiled and made available to all the users of a Peer-To-Peer (P2P) network, a new set of instructions must be sent to all the drones in a DOD surveillance network.

Since no single computer has a physical connection to all the computers in the network, and only has a limited amount of bandwidth available, what is the most efficient way to transmit a given information to as large a fraction of the network as possible in the shortest amount of time conceivable? Or, as in the case of less legitimate applications, what can be done to impair or, if possible, completely block that transmission?

10.3 Modeling

The attentive reader will by now have noticed the similarities between the two realms of application mentioned in this introduction. In epidemiology, the holy grail is to limit the propagation of disease between individuals in a population, in computer science the objective can be either to promote or impair the propagation of information among computers in a network. Even though the two realms are very distinct, the problem can be formulated in a similar fashion and the results applicable to both cases. In the rest of this Chapter we will focus primarily on the, perhaps more complex, case of disease propagation in human populations with special emphasis on HIV spread due to its relevance, but parallels will be drawn to the field of Computer Science whenever deemed necessary.

In the graph approach each of the nodes in the network represents an individual which can be in one of several states and each node can interact only with nodes with which it has a common edge. This approach has the advantage of allowing a much more detailed analysis of the spreading of the virus from person to person and to isolate individuals or groups that play a predominant role in spreading the illness and should be targeted preferentially if we aim at hindering its destructive progress.

The interactions between individuals within a given population is commonly represented as in Fig. 10.1 with squares representing males, circles representing females and edges connecting them whenever a pair of nodes interacts in a way that can result in infection. The weight of each edge can be seen as representing the importance or frequency of that contact.

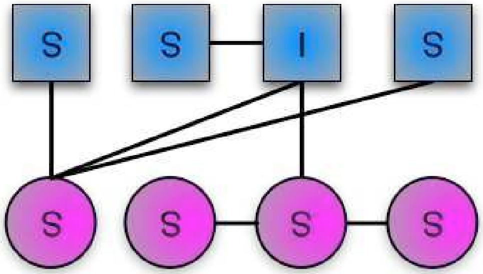

Figure 10.1: Structured interactions between individuals. Blue squares represent males and Pink circles correspond to females. Each individual only interacts directly through the connections it has.

If we consider only relatively short time periods or diseases with long permanence time, such that the likelihood of one of the nodes dying of becoming immune from the disease, we can restrict ourselves to two possible states for each node, Susceptible (or healthy) and Infected (or sick),[1] and deal only with transitions between the Susceptible and Infected states. This type of model is usually referred to in the literature as SI (for Susceptible-Infected) models and is usually take as a first step towards understanding the basic mechanisms that influence the epidemic in its earlier stages.

Following this formulation of the problem, a generic algorithm can be quickly formulated to serve as a template for numerical studies, such an algorithm is defined in Alg. 5. Briefly, we use the weights along each edge as a sort of contact frequency between the nodes and after each interaction, a random number drawn from an appropriate distribution tells us when the next exchange between these two individuals occurs. It should be clear that the model we have just formulated is fairly generic and we expect that for a given interaction network we should observe very different epidemic behaviors by using different sets of weights, w_{ij}, probability distribution functions, $\mathcal{E}\left(\omega_{ij}, t\right)$, and infection probabilities, p_{ij}. Our goal is to try to identify what effect does the combination of these two factors has on the way the infection progresses and to gain insight that might be applicable to real world processes.

The different parameters for the model, such as the time distribution function, infection probabilities and weight distributions must be chosen for each specific case. Before we can

[1] In the case of update broadcasting in Computer Science, the Susceptible and Infected states would correspond, respectively, to Outdated and Updated.

Algorithm 5 Susceptible-Infected model

```
Start with an initial set of infected and susceptible individuals (nodes) connected along the edges of a weighted network.
1. Assign interaction times, $t_{ij}$, to each edge drawn randomly from a distribution $\mathcal{E}\left(\omega_{ij}, t\right)$, where $\omega_{ij}$ is the weight of the edge connecting individuals $i$ and $j$, and the distribution becomes increasingly narrower as $\omega_{ij}$ increases.
2. For the edge with the smallest value of $t_{ij}$:
    (a) An interaction is assumed to occur.
    (b) Set the system's clock time to $t_{ij}$.
    (c) If only one of the two individuals is infected, it infects the other individual with probability $p_{ij}$.
    (d) If both individuals are both infected or both susceptible, nothing happens.Increment the interaction time, $t_{ij}$, by $\Delta t_{ij}$ obtained from the same distribution.
3. Repeat step 3 for a specified number of times or until the systems clock reaches a given time.
```

start analyzing the effect that each of these factors plays in the spreading, we analyses a null model that will give us a baseline to which we might compare our future results. For our null model we chose 50 instances of *Erdos-Renyi* with average connectivity 6.0 and 50 instances of *Barabasi-Albert* networks with 10^4 nodes each.

We started by assigning to each edge a weight obtained from a power law distribution of degree 2 and a uniform infection probability $p_{ij} \equiv p = 0.01$. The wide distribution of weights insures that most frequencies are relatively low but that there are a small number of higher frequencies as well, and the uniform probability signifies that the disease has a relatively low infectivity and is democratic, in the sense that it is as likely to infect any given individual. Finally, for the time distribution function, $\mathcal{E}\left(\omega_{ij}, t\right)$, we chose an exponential $\sim e^{-\omega_{ij} t}$. thus treating the weights in the network as characteristic rates with which a given event occurs. In the context of human interactions, ω_{ij} can be the frequency with which you physically interact with a given person in your daily life and the Computer Science equivalent would be the frequency with which that link is used to transmit packets between two computers.

Given these choices, we expect the main differences between the two datasets to be due

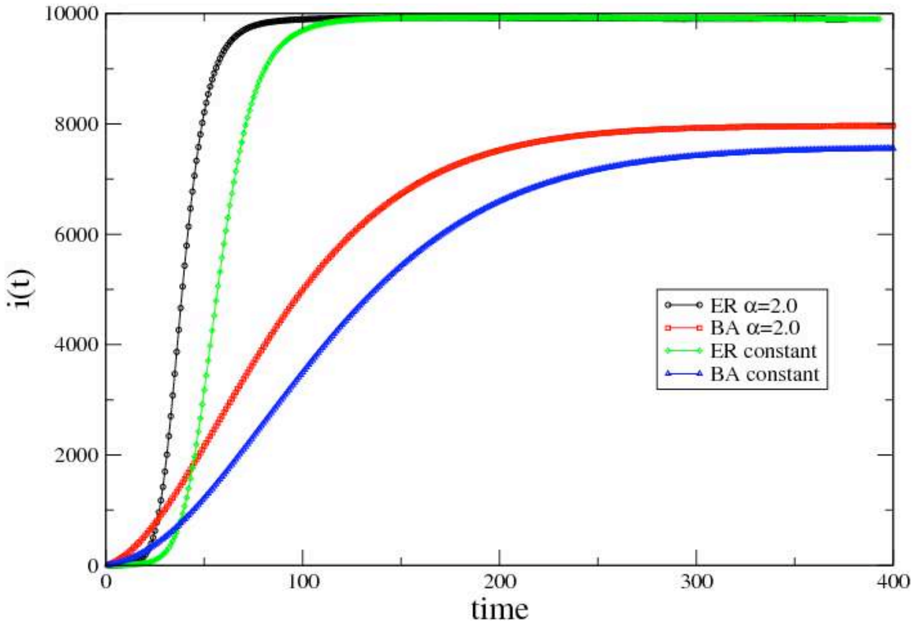

Figure 10.2: Number of Infected individuals as a function of time.

to the topological differences between the two types of networks. The epidemic prevalence, or the number of infections as a function of time, $i\left(t\right)$, for both cases is shown in Fig. 10.2. We also present the results for the case with all the weights set equal to 1 for comparison purposes.

The most obvious difference between the two classes of networks is the speed at which the infection spreads. In the *ER* networks, after a first period of modest growth, $i\left(t\right)$ quickly explodes to encompass a large fraction of the network. On the other hand, the growth is much smoother in *BA* networks, and even after 2 or 3 times longer the fraction of infected individuals is still far from 1. This picture can, however, be deceiving for one would expect that with increasingly larger number of infected individuals, the number of new infections at each time step would also grow. To account for this effect, we follow the approach introduced by Crepey *et al*[138] and define the relative variability of the number of infections as the ratio between the standard deviation and the average number of infections:

$$\Delta\left[i\left(t\right)\right]=\frac{\sqrt{\left\langle i\left(t\right)^{2}\right\rangle-\left\langle i\left(t\right)\right\rangle^{2}}}{\left\langle i\left(t\right)\right\rangle}\tag{10.1}$$

The evolution of this quantity is shown on the left side of Fig. 10.3. The figure clearly

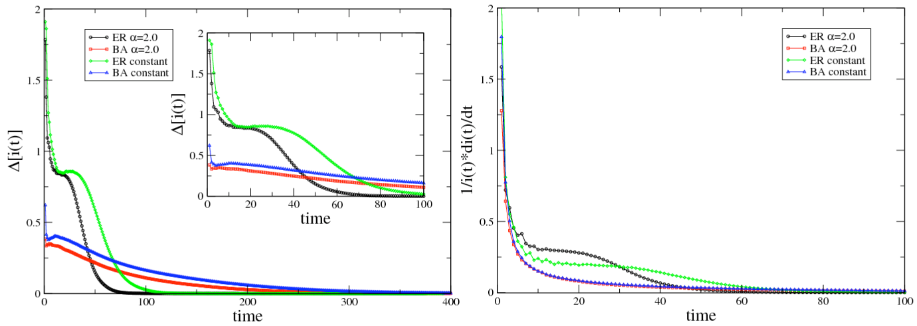

Figure 10.3: left-main) Relative variation of the number of Infected individuals as a function of time. left-inset) Detail of the range time$\in [0, 100]$. right) Logarithmic derivative of the prevalence.

shows that, despite what we expected, the scaled variation in the number of infections becomes smaller over time, but with different behaviors for each class of network and even with differences within the same class between the set with unit weights and the one with power-law distributed weights.

A quantity of great interest to Epidemiologists, is the average number of infections caused by each infected individual. To study this quantity we define $\mathcal{D}\left[i\left(t\right)\right]$, the logarithmic derivative of the prevalence[2]:

$$\mathcal{D}\left[i\left(t\right)\right] = \frac{1}{i\left(t\right)}\frac{di\left(t\right)}{dt}, \tag{10.2}$$

and plot it on the right hand side of Fig. 10.3. This derivative has indeed an interesting behavior for, while it decreases monotonically in all cases, it stabilizes for a short period of time for *ER* networks with the longer stable period corresponding to the networks with unit weights. This seems to indicate that the variation in the weights leads, as one would expect, to larger variations in the number of infections that can be caused by each infected individual and to a faster spread of the disease. In the case of *BA* networks, the two curves are practically indistinguishable beyond the first few stages, both decreasing monotonically, but, in opposition with what is observed in the *ER* case, with unit weighted networks having slightly larger infection ratios than the power-law ones.

An attentive eye will have noticed that the curves for each type of network look remarkably similar, raising the possibility of a scaling law connecting them. The importance of

[2] Numerically, this quantity is calculated by taking the forward difference of $i\left(t\right)$, $i\left(t+1\right)$-i(t) and dividing it by $i\left(t\right)$.

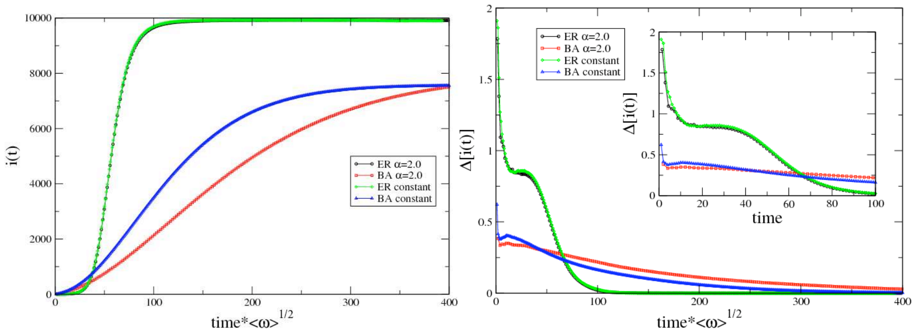

Figure 10.4: Scaled prevalence (left) and relative prevalence variance (right).

this is undeniable, as it would indicate that the exact weights are not important and that we may focus more on just the unweighted version of the model. There is indeed a easy scaling for the *Erdos-Renyi* case, as we show in Fig. 10.4, where we have simply multiplied the time by the square root of the average weight. This scaling, however, does not work for the *Barabasi-Albert* case, thus rendering such scaling useless in this type of network. After trying multiple scalings of both the xx and the yy axis with different functions of $\langle\omega\rangle$, $\left\langle\omega^2\right\rangle$ and σ, it is our belief that the BA does not obey a simple scaling law, thus making an analysis of the effect of the weights an important step in understanding spreading phenomena.

Chapter 11

Final Remarks

> ”Dissertations are not finished; they are abandoned”
>
> – Fred Brooks

Over the previous Chapters we analyzed two different classes of problems. Magnetism, with its long history within physics already possesses the basic tools and frameworks required for its study. Networks, on the other hand, being a much younger field (particularly within Physics), is still at a stage where much has already been achieved, but much more still remains to be done. However different these two problems might seem at first, the set of tools and concepts that they share proves that they are indeed similar. In both cases, the way the components of the system are connected and the strength of each connection plays a major role in the way the system behaves.

In the case of Spin Glasses, the connectivity is one of the determining factors that control the size of avalanches with important consequences for optimization and magnetic recording. Potts spins, lying in a non-geometric regular lattice of fixed connectivity display unintuitive behavior such as negative specific heat. This illustrates that not only the number of connections, but also the way those connections are distributed within the system is also important. We demonstrate this also for the dynamics of the system with the introduction of the Hanoi Networks with their hierarchy and analytical tractability. Even in relatively small systems of Heisenberg spins, you can obtain interesting results by introducing a long range interaction among all of them.

In the better known model of graphs, the Erdos Renyi model, as you increase the average degree of each node you cross two transitions that delimit intrinsically different regions, as can be seen by the efficacy of the tree cutting algorithm we introduced. In more realistic models, correlations between edge weights can significantly improve transport along those edges as can be seen in several cases of real world networks.

The same framework can also be applied to study a much more complex system, society. By analyzing the way that a group of people interact with a web site one can gain insight into the generic mechanisms behind how individuals behave and how these behaviors can have an impact in technology and everyday life. On the other hand, if we build a network of sexual interactions between individuals we can use much of the knowledge gained in the previous chapters to study the way in which a STD spreads within a population and what measures can be taken to combat it. If, however, we replace our labels, turning people in to machines and STDs in to computer viruses, the system remains unchanged and all the results are preserved.

Much more work could be done by applying similar ideas to other problems and exploring other connections along the same lines. While we were able to find solutions to several problems, many other questions were left unanswered. However, we believe that this dissertation illustrates the way these two apparently disjoint areas can cooperate to great mutual benefit, with important implications in a wide range of applications. We hope to be able to proceed within this multi disciplinary framework in the future.

Part III

Appendices

Appendix A

Logarithmic Binning

'An important problem when dealing with experimental data, whether this data originated in a "real" experiment or in a numerical one, is to reduce the amount of stochastic noise, specially when trying to determine the distribution, $P(x)$, of a given quantity x. A common approach is apply some form of logarithmic binning to the data and them proceed to analyze the cleaner "binned" data. Contrarily to what is done for histograms, for example, the bin size is not constant, instead it is given by:

$$\Delta = \frac{1}{N} \log \left(\frac{b}{a} \right) \tag{A.1}$$

where N is the desired number of bins, and b and a are, respectively, the maximum and the minimum of our distribution. Each data point x_i is assigned to bin:

$$n_i = \frac{1}{\Delta} \log \left(\frac{x_i}{a} \right) \tag{A.2}$$

to which corresponds the new xx value X_i given by:

$$X_i = a e^{\Delta (n_i + 1)} \tag{A.3}$$

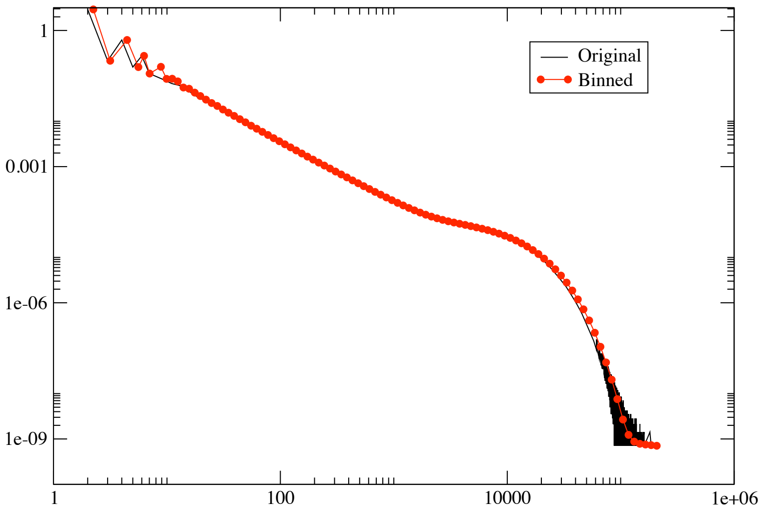

Figure A.1: Logarithmic Binning.

and the distribution value $P\left(X_i\right)$ is simply the arithmetic average of all the points in the bin:

$$P\left(X_i\right) = \frac{1}{N_i} \sum_{x_i \in \{n_i\}} x_i \tag{A.4}$$

where N_i is the number of data points in bin n_i and the sum is taken over all x_i in the bin. This method preserves the functional behavior of the distribution which averaging out fluctuations which are specially dominant in regions with poorer statistics. An example is shown in Fig. A.1 where we clearly see that the statistical fluctuations in the tail are smoothed out.

Appendix B

Generating Function Formalism

In this appendix we provide a brief review of the generating function formalism [139], extensively used in the literature to study stochastic processes [140, 141, 142, 143, 144]. The generating function of a stochastic system with probability $P_n(t)$ of being in state n, at time t, is defined as:

$$G(x,t) = \sum_{n=0}^{\infty} P_n(t)\, x^n \tag{B.1}$$

the derivatives with respect to t and x are, respectively:

$$\frac{\partial G(x,t)}{\partial t} = \sum_{n=0}^{\infty} \frac{\partial P_n(x)}{\partial t} x^n \tag{B.2}$$

$$\frac{\partial G(x,t)}{\partial x} = \sum_{n=0}^{\infty} n P_n(t)\, x^{n-1} \tag{B.3}$$

which we can use to simplify and solve differential equations involving $P_n(t)$. We can then solve to obtain a closed form for $G(x,t)$ and recover the original probabilities by expanding $G(x,t)$ as a Taylor series in x around $x \equiv 0$.

$$P_n(t) = \frac{1}{n!} \left. \frac{\partial^n G(x,t)}{\partial t^n} \right|_{x=0} \tag{B.4}$$

We can also easily see that,

$$G(1,t) = \sum_{n=0}^{\infty} P_n(t) \equiv 1 \tag{B.5}$$

if the probabilities $P_n(t)$ are properly normalized and that:

$$\langle n \rangle = \sum_{n=0}^{\infty} n P_n(t) = \left. \frac{\partial G(x,t)}{\partial x} \right|_{x=1} \tag{B.6}$$

and

$$\sigma^2 = \langle n^2 \rangle - \langle n \rangle^2 = \left. \frac{\partial^2 G(x,t)}{\partial x^2} \right|_{x=1} + \left. \frac{\partial G(x,t)}{\partial x} \right|_{x=1} - \left. \left(\frac{\partial G(x,t)}{\partial x} \right)^2 \right|_{x=1} \tag{B.7}$$

The fact that Generating Functions are, in fact, Formal Power Series has some useful consequences as well. In particular:

$$G^{(k)}(x,t) = (G(x,t))^k \tag{B.8}$$

where $G^{(k)}(x,t)$ is the generating function for k independent samplings of the $P_n(t)$ distribution and

$$G(x,t) H(y,t) = \sum_n P_n(t) x^n \sum_m Q_m(t) y^m = \sum_n R_n(t) z^n \tag{B.9}$$

with

$$R_n(t) = \sum_k P_k(t) Q_{n-k}(t) \tag{B.10}$$

We also enumerate some useful generating functions and refer to the literature[145, 146, 147, 148] for more examples.

Appendix C

Renormalization Group For Random Walks

As a pedagogical example, we present here the theory as it will be applied in Chap. 6. We consider a biased random walk on a finite one-dimensional line. In particular, we consider the questions of the average transit and first return times for starting at the left-most site "0" and either being absorbed at the right-most site or returning to "0", respectively. The master equations for such a random walk on a lattice of length $N = 2^{K+1}$ with reflecting boundaries are given by

$$\begin{aligned}
P_{0,t+1} &= p\,P_{1,t}, \\
P_{1,t+1} &= P_{0,t} + p\,P_{2,t}, \\
P_{n,t+1} &= (1-p)\,P_{n-1,t} + p\,P_{n+1,t} \qquad (2 \leq n \leq N-2), \\
P_{N-1,t+1} &= (1-p)\,P_{N-2,t} + P_{N,t}, \\
P_{N,t+1} &= (1-p)\,P_{N-1,t},
\end{aligned} \tag{C.1}$$

where $P_{n,t}$ denotes the probability of a walker to be at site n at time t, p is the probability expressing the biasing for left or right hops. Since we want to start the walks at time $t = 0$

at the origin $n=0$, these equations have the initial condition

$$P_{n,0} \quad = \quad \delta_{n,0}. \tag{C.2}$$

To facilitate renormalization, we introduce a generating function

$$\tilde{P}_n(z) \quad = \quad \sum_{t=0}^{\infty} P_{n,t} z^t \qquad (0 \leq n \leq N).$$

Incorporating the initial condition in Eq. (C.2), Eqs. (C.1) transform into

$$\begin{aligned} \tilde{P}_0 &= a\,\tilde{P}_1 + 1, \\ \tilde{P}_1 &= c\,\tilde{P}_0 + a\,\tilde{P}_2, \\ \tilde{P}_n &= b\,\tilde{P}_{n-1} + a\,\tilde{P}_{n+1} \qquad (2 \leq n \leq N-2), \\ \tilde{P}_{N-1} &= b\,\tilde{P}_{N-2} + d\,\tilde{P}_N, \\ \tilde{P}_N &= b\,\tilde{P}_{N-1}, \end{aligned} \tag{C.3}$$

where we have inserted generalized hopping rates in preparation for the RG. Initially, at the $k=0$th RG step, it is

$$\begin{aligned} a^{(0)} &= p\,z, \\ b^{(0)} &= (1-p)\,z, \\ c^{(0)} &= z, \\ d^{(0)} &= z, \end{aligned} \tag{C.4}$$

which provides a sufficient number of renormalizable parameters that are potentially required to consider special sites at both boundaries.

A single step of applying the RG consists of solving Eqs. (C.3) for $\tilde{P}_n$ with odd values of n (which is trivial here, as they are already expressed explicitly in terms of even ones) and eliminating them from the equations for the even n. After that elimination, we can rewrite the equations for even n as

$$\begin{aligned}
\tilde{P}_0 &= \frac{a^2}{1-ac}\,\tilde{P}_2 + \frac{1}{1-ac}, \\
\tilde{P}_2 &= \frac{bc}{1-2ab}\,\tilde{P}_0 + \frac{a^2}{1-2ab}\,\tilde{P}_4, \\
\tilde{P}_{2n} &= \frac{b^2}{1-2ab}\,\tilde{P}_{2n-2} + \frac{a^2}{1-2ab}\,\tilde{P}_{2n+2} \qquad \left(2 \le n \le \frac{N}{2}-2\right), \\
\tilde{P}_{N-2} &= \frac{b^2}{1-2ab}\,\tilde{P}_{N-4} + \frac{ad}{1-2ab}\,\tilde{P}_N, \\
\tilde{P}_N &= \frac{b^2}{1-bd}\,\tilde{P}_{N-2}.
\end{aligned} \tag{C.5}$$

Comparing these equations with Eqs. (C.3) allows to extract the RG recursion equations. [Note that superscripts referring the kth RG step have been suppressed thus far in Eqs. (C.3) and (C.5).]

Before we analyseui the first return time at the boundary specifically, we can use the equation for bulk sites n in (C.5) to extract already the diffusion exponent d_w. A comparison of the respective coefficients in Eqs. (C.3) and (C.5) yields

$$a^{(k+1)} = \frac{\left(a^{(k)}\right)^2}{1-2a^{(k)}b^{(k)}}, \qquad b^{(k+1)} = \frac{\left(b^{(k)}\right)^2}{1-2a^{(k)}b^{(k)}}. \tag{C.6}$$

These recursions converge for $k \to \infty$ towards fixed points (a^*, b^*) that characterize the dynamics in the infinite time limit (which corresponds to the limit of $z \to 1^-$). The trivial fixed point $a^* = b^* = 0$ is unphysical, as it can not be reached from the initial conditions in (C.4) for any choice of p (at $z = 1$). The physical fixed points are $(a^*, b^*) = (1, 0)$, which is reached for any bias $p > \frac{1}{2}$, or $(a^*, b^*) = (0, 1)$, reached for $p < \frac{1}{2}$; finally, $(a^*, b^*) = \left(\frac{1}{2}, \frac{1}{2}\right)$ can only be reached by entirely unbiased walks for $p = \frac{1}{2}$. To explore the behavior for large but finite times, we expand the RG recursions in (C.6) to first order in $\epsilon = 1 - z$ by writing for $y \in \{a, b\}$:

$$y^{(k)} \quad \sim \quad y^* + y_1^{(k)}\epsilon + \ldots. \tag{C.7}$$

Inserting the Ansatz in Eq. (C.7) into the recursions in Eqs. (C.6), we obtain near the fixed point with $a^* = b^* = \frac{1}{2}$:

$$\begin{aligned} a_1^{(k+1)} &= 3a_1^{(k)} + b_1^{(k)}, \\ b_1^{(k+1)} &= a_1^{(k)} + 3b_1^{(k)}, \end{aligned} \tag{C.8}$$

with the result that

$$a_1^{(k)} = b_1^{(k)} \propto 4^k.$$

This implies that as space rescales by a factor of 2 (i. e. eliminating all odd-index sites), time rescales by a factor of 4 for an unbiased random walk, leading to $d_w = 2$. The same analysis for either of the biased fixed points yields that , say, $a^{(k)} \equiv 0$ beyond any power of ϵ and, with the Ansatz $b^{(k)} \sim 1 + b_1^{(k)}\epsilon$ (and $a \equiv 0$) in Eqs. (C.6), $b_1^{(k)} \propto 2^k$. Following this interpretation, this would imply $d_w = 1$ and we find the familiar result that with the slightest bias, $p < \frac{1}{2}$ or $p > \frac{1}{2}$, the motion at large length and time scales is dominated by the constant-velocity drift upon reaching the bulk.

In this scenario, average first-return times are clearly system-size independent constants: A walker with a bias towards the origin ($p > \frac{1}{2}$) will drift back recurrently after only small excursions; a walker with a bias away from the origin ($p < \frac{1}{2}$) returns at most a few times in short order until the drift eventually carries it away without further recurrence. In the following, we therefore focus exclusively on the unbiased case $p = \frac{1}{2}$. Then, we can equate $a = b$ at every step, to get

$$a^{(k+1)} = \frac{\left(a^{(k)}\right)^2}{1 - 2\left(a^{(k)}\right)^2}. \tag{C.9}$$

To derive the return-time behavior, we have to examine Eqs. (C.5) more closely. Comparing coefficients also in the boundary terms leads to

$$c^{(k+1)} = \frac{a^{(k)}c^{(k)}}{1 - 2\left(a^{(k)}\right)^2}, \qquad d^{(k+1)} = \frac{a^{(k)}d^{(k)}}{1 - 2\left(a^{(k)}\right)^2}. \tag{C.10}$$

For large k, both $c^{(k)}$ and $d^{(k)}$ are entrained with $a^{(k)}$, and we obtain a consistent and closed set of relations for all coefficients in Eqs. (C.3) and (C.5) by identifying $c = d = 2a$. Further renormalizing

$$\tilde{P}_n^{(k+1)} = \left[1 - 2\left(a^{(k)}\right)^2\right] \tilde{P}_{2n}^{(k)} \tag{C.11}$$

ensures invariance of the constant term at the lower boundary that originated from the unit initial condition in Eq. (C.2).

After $k = K$ RG steps, the system has reduced to

$$\begin{aligned} \tilde{P}_0^{(K)} &= a^{(K)} \tilde{P}_1^{(K)} + 1, \\ \tilde{P}_1^{(K)} &= 2a^{(K)} \tilde{P}_0^{(K)} + 2a^{(K)} \tilde{P}_2^{(K)}, \\ \tilde{P}_2^{(K)} &= a^{(K)} \tilde{P}_1^{(K)}, \end{aligned}$$

which yields

$$\tilde{P}_0^{(K)} = \frac{1 - 2\left(a^{(K)}\right)^2}{1 - 4\left(a^{(K)}\right)^2}. \tag{C.12}$$

Using Eq. (C.11) in turn obtains

$$\tilde{P}_0^{(0)} = = \frac{\tilde{P}_0^{(K)}}{\prod_{k=0}^{K-1}\left[1 - 2\left(a^{(k)}\right)^2\right]} = \frac{1 - 2\left(a^{(K)}\right)^2}{1 - 4\left(a^{(K)}\right)^2} \prod_{k=0}^{K-1} \frac{1}{\left[1 - 2\left(a^{(k)}\right)^2\right]}. \tag{C.13}$$

It is a well-known fact that the generating functions for being at the origin, $\tilde{P}_0^{(0)}$, and for the first-return probability to the same site, $\tilde{Q}_0$, satisfy the following simple relation[62]:

$$\tilde{Q}_0 = 1 - \frac{1}{\tilde{P}_0^{(0)}}. \tag{C.14}$$

Note that a recurrent walk (with $\tilde{Q}_0 = 1$) requires that $\tilde{P}_0^{(0)}$ diverges at long times (i. e. $z \to 1^-$). In our one-dimensional walk here, it is $a^{(K)} \to a^* = \frac{1}{2}$, on behalf of which the denominator of $\tilde{P}_0^{(0)}$ in Eq. (C.13) has a zero, making the walk recurrent. In more detail, it is

$$\tilde{Q}_0 \quad = \quad 1 - \frac{1-4\left(a^{(K)}\right)^2}{1-2\left(a^{(K)}\right)^2} \prod_{k=0}^{K-1}\left[1-2\left(a^{(k)}\right)^2\right]. \tag{C.15}$$

To extract the exponent μ for the first-return probability distribution,

$$Q_0(t) \quad \sim \quad t^{-\mu} \qquad (t \to \infty),$$

requires an expansion of $\tilde{Q}_0(z)$ to 2nd order in $\epsilon = 1 - z$.

Luckily, due to the leading zero in $1 - 4\left(a^{(K)}\right)^2$, any other factor in Eq. (C.15) only needs to be expanded to first order in ϵ. Extending the Ansatz in Eq. (C.7) for a to 2nd order, we obtain here:

$$\begin{aligned}
a^{(K)} \quad &\sim \quad \frac{1}{2} - \frac{1}{2} \times 4\epsilon + 5 \times 16^K \epsilon^2 + \ldots, \\
1-4\left(a^{(K)}\right)^2 \quad &\sim \quad 8 \times 4^K \epsilon + 116 \times 16^K \epsilon^2 + \ldots, \\
1-2\left(a^{(K)}\right)^2 \quad &\sim \quad \frac{1}{2} + 4 \times 4^K \epsilon + \ldots, \\
\prod_{k=0}^{K-1}\left[1-2\left(a^{(k)}\right)^2\right] \quad &\sim \quad \prod_{k=0}^{K-1}\left[\frac{1}{2} + 4 \times 4^k \epsilon + \ldots\right], \\
&\sim \quad 2^{-K}\left[1 + 8\epsilon \sum_{k=0}^{K-1} 4^k + \ldots\right], \\
&\sim \quad 2^{-K} + \frac{8}{3} 2^K \epsilon + \ldots.
\end{aligned}$$

Inserting these expressions into Eq. (C.15), we get

$$\tilde{Q}_0 \quad \sim \quad 1 - 16 \times 2^K \epsilon + \frac{952}{3} \times 8^K \epsilon^2 + \ldots. \tag{C.16}$$

Then, we can extract for the moments

$$\begin{aligned}
\langle t \rangle_K \quad &\sim \quad 2^K, \\
\left\langle t^2 \right\rangle_K \quad &\sim \quad 8^K,
\end{aligned}$$

which leads to the familiar[62] first-return exponent of a one-dimensional walk,

$$\mu \quad = \lim\nolimits_{K\to\infty} \quad 2 + \frac{1}{1 - \frac{\langle t^2 \rangle_K}{\langle t \rangle_K}} = \frac{3}{2}. \tag{C.17}$$

Appendix D

Condor and DRBL

Imagine, you have been given a modest budget for your scientific computation needs, and you want to get the most bang for the buck. Or you have accumulated a diverse bunch of work-stations, and it is turning into a pain to keep programs running on each one while keeping track of their output, not to mention the administrative overhead. Throw in the fact that your department may have a student lab with brand-new desktops, sitting idle part of the year. Pulling together a few freely available software packages, such a heterogeneous environment can be bundled into a scalable computer cluster that is easy to administer and provides basically the unlimited uptime needed for most scientific computations. This article explains how we proceeded.

To build our cluster we are going to need three pieces of software:

- Fedora Core 3[149]
- DRBL (Diskless Remote Boot on Linux)[150]
- Condor[151]

Fedora Core 3 (or its successor) will be our base distribution, since it's freely available and is well supported by both DRBL and Condor. For the purposes of this tutorial we will assume that you already have FC3 installed on the machine that is to become your server. The server machine will be responsible for storing and serving all the files necessary for the client machines to be able to function properly. It will also act as a firewall between the cluster and the outside world and should possess more than one Ethernet card. While not essential, it would be worthwhile to budget for some extra hardware in the server for speed and security such as Giga-LAN, RAID, DVD-Burner, etc.

DRBL on its own allows a set of "thin" clients to be booted off a server, say, in a classroom setting. Thus these clients do not require any disks or peripheral hardware. For instance, in our cluster each thin client box has only a motherboard, an Ethernet card (typically integrated in the motherboard), a CPU, and some memory, but no video, disk, etc. At the time of writing, such a node could cost below $250 with a 64-bit CPU and 1Gb of memory.

By adding the Condor clustering software, we turn this set of machines into a computing cluster that can perform high-throughput scientific computation on a large scale. You can submit serial or parallel computing jobs on the server, and Condor takes care of distributing the jobs to idle cluster machines, if any, or puts them in a queue until the required resources are available. Condor can also perform periodic checkpoints on the jobs and restart them if something causes the machine on which they are running to reboot. Even if the server itself has to reboot (say, when a disk in your RAID-5 array failed), the client jobs wait patiently for its return.

The end product is very stable. You can add and remove compute nodes from the system without having to reconfigure anything. Designating a machine as a member of the cluster requires just a small penalty of hard drive space on the server to store some configuration files; in our cluster, config-files for 72 machines use less than 3GB of space on the server.

Several similar clustering solutions are currently available, most of them being directly related to Cluster Knoppix [152], such as Parallel Knoppix [153], Quantian [154], BCCD [155], etc. But these are mostly geared for educational or demonstration purposes and run off a live-CD/DVD, which is not well suited for long term usage that is essential for most scientific computing. These solutions provide convenient ways to run short term jobs but are unsuitable for longer term projects. In fact, in our initial attempt to cluster our hardware, we have tested the hard-drive installation [156] provided with Cluster Knoppix (then version 3.4) [152]. The resulting cluster configuration had significant bugs; an indication that it is far less supported than the live-CD version. Once it was fixed, the installation provided a diskless boot function similar to DRBL and clustering provided by OpenMosix [157]. Unlike Condor, OpenMosix does not require special instructions to batch jobs. Instead, it makes it appear as if the user is sitting in front of a single computer that happens to have multiple CPU. Yet, OpenMosix is directly compiled into the Linux kernel, which made system administration a task beyond our expertise. Any upgrade to the cluster software essentially meant re-installation. Also, while the OS performed just as reliable as expected from any Linux system, in case of power-failure or server reboot, say, all jobs are lost.

The implementation we propose here requires a dedicated machine as server but it allows for months (years?) of continuous usage and for easy administration, upgrades and large volume backups. Adding new machines is trivial since it does not require a reboot or even a reconfiguration of the cluster and the jobs continue to run happily even if the master is brought down temporarily.

D.1 DRBL installation

In this example we will assume that our server has three network interfaces: eth0, to connect to the outside world, and eth1 and eth2 to connect to the client machines. We're using two different subnets for the client machines in order to spread the network traffic over two different routers. Before we start installing DRBL on the server, we configure eth1 and eth2 with private IP addresses (of the form 192.168.x.x).

Most of our client machines are thin. We set these clients' BIOSes to boot via the network, which obliged us to temporarily install a graphics card and access the BIOS menu. Since we only need to use the graphics card for a few seconds on each machine, we have just one card that was installed on each of the machines in turn. The clients in our student lab are fully configured, running a non-Unix OS. Using a single PXE boot floppy [158] allowed us to commandeer these clients *without* affecting their BIOS or disks.

DRBL is available as a small RPM package. To set it up we are going to be using two installation scripts: “drblsrv”, to configure the server, and “drblpush”, to configure the clients. Start with ” drblsrv” by running:

```
# /opt/drbl/setup/drblsrv -i
```

This Perl script installs the packages that DRBL requires, such as DHCP and TFTP, if they're not already installed, and asks if we want to update the system. If we choose to perform the update, it will run "yum" to update all the packages for which there is a newer version (similar to what would happen if you were to run "up2date"[1] manually). The default options work well for most cases. We just have to remember that if we downloaded the DRBL RPM from the testing or unstable directories to select "Yes" when it asks if we want to use the testing or unstable packages.

After " drblsrv" finishes (which can take a fair amount of time, especially if we have a slow Internet connection and asked to update the system) it's time to run " drblpush" :

```
# /opt/drbl/setup/drblpush -i
```

This second Perl script is the workhorse of the system. It configures and starts all the services required to make the cluster work. It automatically detects the network interfaces that have private IP addresses assigned to them and asks us how many clients we want to set up in each of them. Here we want to allow for more clients than the number of machines currently have available. This allows us to add more machines later without having to reconfigure the server at only a small penalty in hard drive space. For added security and manageability, we can bind the booting process (via DHCP) to the MAC addresses of our clients. This feature is useful if we are setting up our system in the open (in a classroom, for instance) where anyone can plug in a new machine without us knowing about it and for organization purposes, since this way we know to what machine each hostname corresponds to. We'll assume our system is behind closed doors and accessible to users only by SSHing[159] to the server, so we'll choose "No" for this option and simply tell DRBL how many clients we want on each network interface. They will all get IP addresses on a first-come, first-served basis at boot time.

At this stage of the configuration process DRBL will present us with a little graphical view of our setup:

[1] On Red Hat based systems, other distributions might use "apt-get" or "synaptic".

```
NIC NIC IP Clients
+-----------------------------+
| DRBL SERVER                 |
|                             |
|  +-- [eth0] xxx.xxx.xxx.xxx +- to WAN
|                             |
|  +-- [eth1] 192.168.110.254 +-
to clients group 1 [ 24 clients, their IP
|                             | from 192.168.110.1 -
192.168.110.24]
|  +-- [eth2] 192.168.120.254 +-
to clients group 2 [ 24 clients, their IP
|                             | from 192.168.120.1 -
192.168.120.24]
+-----------------------------+
```

In this example we made room for 24 machines on each subnet. If our configuration doesn't look the way we intend it, we can press Ctrl-C and restart from scratch; nothing has been changed in our system yet. After this point, the only option we might want to change from the default is the one referring to the boot mode. Since these machines will be used only for number-crunching and have no terminals, we can set the boot mode to " 2" for text mode so we don't waste resources starting an X server that will never be used. We can safely leave all the other options at default values, or modify them to serve our specific goals. If at a later date we change something on the server and we want those changes to propagate to the clients we simply need to re-run " drblpush" .

In this case we can choose to keep the old client settings and just export the changes or to redo everything from scratch. After DRBL finishes successfully, we can boot our client machines and if everything went according to plan we should now have a basic cluster system up and running. If we run in to any problems, we can use the project's support forum [160]. The experience has been that we typically get an answer back from one of the developers within about a day.

D.2 Condor installation

To make the most of our new Linux cluster we will install Condor, a batch system developed by the University of Wisconsin[151]. Download the latest tarball, uncompress it, and run

```
# ./condor_install
```

The installation script provides plenty of details on what each option means, so one shouldn't have many problems setting it up. We must, however, keep in mind that everything is shared between the server and the clients and answer accordingly when it asks us about the file systems, accounts, and password files. Also, if we want to be able to make use of Java support, we need to have Sun's Java[161] virtual machine installed prior to installing Condor. (If installed afterward one must sort through Condor's configuration files or redo the Condor installation.) Message Passing Interface (MPI)[162] Protocol support requires some tinkering with the configuration files after the installation. Both "mpich"[163] and "lam-mpi"[164] are supported. For further details on Condor installation and its many features, refer to the Condor manual[165] on-line and to the very active mailing list that one is invited to subscribe to on downloading the software.

After the installation, we start Condor on the local machine by running:

```
# /usr/local/condor/bin/condor_master
```

This command should spawn all other processes that Condor requires. On the server we should be able to see five condor_ processes running:

```
# ps -ef | grep condor
condor 16418 1 0 14:18 ? 00:00:00 condor_master -f
condor 16477 16418 0 14:20 ? 00:00:00 condor_collector -f
condor 16478 16418 0 14:20 ? 00:00:00 condor_negotiator -f
condor 16479 16418 0 14:20 ? 00:00:00 condor_schedd -f
condor 16480 16418 0 14:20 ? 00:00:00 condor_startd -f
```

```
root 16503 16382 0 14:20 pts/4 00:00:00 grep condor
```

When starting Condor on the clients, we should see all of the above except condor_collector and condor_negotiator. Sometimes the installation program gets a bit confused with the fact that the server has several hostnames (the usual hostname for eth0, nfsserver_eth1, and nfsserver_eth2 respectively); in those cases, running condor_master on the server will start only the services that are used by a client machine. We can easily fix the problem by using condor_configure to force the current machine (with all its hostnames) to be the server. This will change the relevant configuration files, after which we must tell Condor to reread them:

```
# ./condor_configure --type=submit,execute,manager
# /usr/local/condor/sbin/condor reconfig
Sent " Reconfig"  command to local master
```

After that we should be able to see the server machine as part of our Condor cluster when you run condor_status.

```
# condor_status
Name          OpSys       Arch   State      Activity   Loa-
dAv Mem   ActvtyTime
underdark     LINUX       INTEL  Un-
claimed  Idle       0.240  1518  0+00:20:04
              Machines Owner Claimed Unclaimed Matched Preempting
INTEL/LINUX       1      0       0         1       0          0
Total             1      0       0         1       0          0
```

If we now run condor_master on all the nodes we should see the clients being added to this list within a few minutes (usually around five to 10 minutes, but the more machines there are the longer it takes for all of them to finish handshaking with the server). DRBL provides a simple mechanism to run a command on all the nodes. To start condor_master manually on all the clients we simply have to:

```
# /opt/drbl/bin/drbl-doit /usr/local/condor/sbin/condor_master
```

“drbl-doit” will run the command under the same user that calls it, so we should run this as root on the server machine. We can also have condor starting automatically when we boot the machines by adding condor_master to the rc.local file of the nodes. This can be done manually, or by editing the relevant section on the ” drblpush” Perl script and rerunning it; only then will we be able to add machines to our cluster without having to change anything.

D.3 Using the cluster

There are a number of tutorials[166] available on-line (not to mention the official manual[165]) that teaches the basics of using Condor, but the mechanics of it are fairly simple, especially if one has used other batching systems before.

To submit a job to the cluster we are going to need to create a submit script. This is basically a text file that in general would look something like this:

```
Executable = hello

Universe = Vanilla

Output = hello.out.$(PROCESS)

Input = hello.in.$(PROCESS)

Error = hello.err.$(PROCESS)

Log = hello.log

Queue 3
```

In this example our executable file is called hello (the traditional ” Hello World program”). We’re using the Vanilla universe (just your normal run of the mill executable, check the manual to find out about all the other universes). The Input, Output, Error and Log directives just tell condor which files to use for stdin, stdout, stderr and to log the jobs execution. Finally, the “Queue” directive specifies how many copies of the program to run. In this particular case we are starting 3 different jobs. The $(PROCESS) macro allows each job to use a different set of files, so job 0 would use hello.*.0, job 1 uses hello.*.1, etc.

After we have the submit file ready we can submit it to condor by running

```
    # condor_submit hello.sub
```

We can also check on the status of our job using

```
    # condor_q
```

which will tell us how many jobs are in the queue, their Id's and whether they're Running or Idle along with some statistics. Condor itself has many other features and provides for many possibilities, shown here are just the very basics of how to get it up and running. When reading the manual, one should pay particular attention to the Standard universe that allows to checkpoint jobs (at the cost of specially linking it) and the Java Universe that allows to run Java jobs seamlessly. We can also add condor to the boot sequence of our server and clients (just add it to the server as explained in the manual and rerun "drblpush"). By doing this and setting the DHCP server to allow more machines than what we currently have and just attribute IPs on a first come first serve basis (as suggested above) we can easily add machines to the cluster by simply connecting them to one of the subnets and booting them (if we configured the nodes to start condor at boot time). They will start condor on their own, connect to the central manager and become available to run jobs in just a few minutes. We can also shutdown machines and their jobs will continue (on the Standard Universe) or restart (on the Vanilla Universe) in a different machine. This allows for a lot of flexibility in managing the system. Also, by starting and stopping the DHCP server at a give time we can change the booting process for the clients. This is useful for the student lab full of machines that need to run Windows during the day but that are free to be added to the cluster at night. In that case, we just modify the boot device sequence of the clients BIOS to first boot from the network and only then boot from the hard-drive. If the DHCP server is on, they will boot from it and join the cluster, if not (in the morning for instance) the attempt to boot from the network will fail and they will simply boot from the hard-drive. This will allow us to put all those windows machines to good use without having to change their settings (other than the BIOS and possibly a Scheduler job to have them boot out of Windows at night).

Bibliography

[1] H. G. Katzgraber and A. P. Young. Monte carlo studies of the one-dimensional ising spin glass with power-law interactions. *Phys. Rev. B*, 67:134410, 2003.

[2] H. G. Katzgraber and A. P. Young. Probing the almeida-thouless line away from the mean-field model. *Phys. Rev. B*, 72:184416, 2005.

[3] P. Gambardella, A. Dallmeyer, K. Maiti, M. C. Malagoli, W. Eberhardt, K. Kern, and C. Carbone. Ferromagnetism in one-dimensional monatomic metal chains. *Naure*, 416:301, 2002.

[4] P. Erdös and A. Rényi. On random graphs. *Publications mathematicae*, 6:290, 1959.

[5] P. Erdös and A. Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci.*, 5:17, 1960.

[6] J. Travers and S. Milgram. An experimental study of the small world problem. *Sociometry*, 32:425, 1969.

[7] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286:509, 1999.

[8] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the internet topology. In *SIGCOMM*, pages 251–262, 1999.

[9] A. L. Barabaśi, R. Albert, and H. Jeong. Scale-free characteristics of random networks: The topology of the world wide web. *Physica A*, 281:2115, 1999.

[10] F. Liljeros, C. R. Edling, L. A. N. Amaral, H. E. Stanley, and Y. Abergmove. The web of human sexual contacts. *Nature*, 411:907, 2001.

[11] E. Ising. Beitrag zur theorie des ferromagnetismus. *Z. Phys.*, 31:253, 1925.

[12] L. Onsager. Crystal statistics i. a two-dimensional model with an order-disorder transition. *Phys. Rev.*, 65:117, 1944.

[13] D. Sherrington and S. Kirkpatrick. Solvable model of a spin-glass. *Phys. Rev. Lett.*, 35:1792, 1975.

[14] S. F. Edwards and P. W. Anderson. Theory of spin glasses. *Journal of Physics: F*, 5:965, 1975.

[15] J. C. Maxwell. *A Treatise On Electricity And Magnetism - Volume I.* Oxford, 1873.

[16] J. C. Maxwell. *A Treatise On Electricity And Magnetism - Volume II.* Oxford, 1873.

[17] H. Nishimori. *Statistical Physics of Spin Glasses And Information Processing - An Introduction.* Cradendon Press, 2001.

[18] K. Fisher and J. A. Hertz. *Spin Glasses.* Cambridge University Press, 1993.

[19] T. Castellani and A. Cavagna. Spin-glass theory for pedestrians. *e-print cond-mat/0505032*, 2005.

[20] D. Sherrington. Spin glasses. *e-print cond-mat*, page 9806289, 1998.

[21] P. E. Jönsson, R. Mathieu, P. Nordblad, H. Yoshino, H. Aruga Katori, and A. Ito. Spin glasses: A ghost story. *Phys. Rev. B*, 70:174402, 2004.

[22] N. Kawashima and H. Rieger. Recent progress in spin glasses. *e-print cond-mat/0312432*, 2003.

[23] F. Guerra. Spin glasses. *e-print cond-mat/0507581*, 2005.

[24] G. Zarand, F. Pazmandi, K.F. Pal, and G.T. Zimanyi. Using hysteresis for optimization. *Phys. Rev. Lett.*, 89:150201, 2002.

[25] K. F. Pál. Hysteretic optimization for the sherrington-kirkpatrick spin glass. *Physica A*, 367:261, 2006.

[26] K. F. Pal. Hysteretic optimization, faster and simpler. *Physica A*, 360:525, 2006.

[27] S. Boettcher and A. G. Percus. Extremal optimization: Methods derived from co-evolution. In *GECCO-99: Proceedings of the Genetic and Evolutionary Computation Conference*, page 825. Morgan Kaufmann, San Francisco, 1999.

[28] S. Boettcher and A. Percus. Nature's way of optimizing. *Artificial Inteligence*, 119:275, 2000.

[29] B. Goncalves and S. Boettcher. Hysteretic optimization for spin glasses. *J. Stat. Mech*, page P01003, 2008.

[30] G. S. Fishman. *Monte Carlo Concepts, Algorithms And Applications*. Springer, 1996.

[31] L. Zhu. *Nonparametric Monte-Carlo Tests*. Springer, 2005.

[32] R. M. Neal. *Probabilistic Inference Using Markov Chain Monte-Carlo Methods*. Department of Computer Science, University Of Toronto, 1993.

[33] J. E. Gentle. *Random Number Generation And Monte-Carlo Methods*. Springer, 2nd edition, 2003.

[34] K. Binder and D. W. Heermann. *Monte Carlo Simulation in Statistical Physics: An Introduction*. Springer Series in Solid State Sciences. Springer-Verlag, 1989.

[35] W. Krauth. Introduction to monte carlo algorithms. *e-print cond-mat*, page 9612186, 1996.

[36] G. Bhanot. The metropolis algorithm. *Rep. Prog. Phys.*, 51:429, 1988.

[37] H. J. Kushner and G. G. Yin. *Stochastic Approximation And Recursive Algorithms And Applications*. Springer, 2nd edition, 2003.

[38] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *J. Chem. Phys.*, 21:1087, 1953.

[39] M. Creutz. Microcanonical monte carlo simulation. *Phys. Rev. Lett.*, 50:1411, 1983.

[40] M. E. J. Newman and G. T. Barkema. *Monte Carlo Methods in Statistical Physics.* Claredon Press - Oxford, 1999.

[41] J. Hove. Density of states determined from monte carlo simulations. *Phys. Rev. E*, 70:056707, 2004.

[42] M. Mezard and G. Parisi. The cavity method at zero temperature. *J. Stat Phys*, 111:1, 2003.

[43] O. Rivoire. The cavity method for large deviations. *e-print cond-mat/0506164*, 2005.

[44] P. Maragos. Slope transforms: Theory and application to nonlinear signal processing. *IEEE Trans. Signal Proc.*, 43:864, 1995.

[45] R. Monasson A. Engel and A. Hartmann. On large deviation properties of erdös-rényi random graphs. *J. Stat. Phys*, 117:387, 2004.

[46] J. Barre and B. Goncalves. Ensemble inequivalence in random graphs. *Physica A*, 386:212, 2007.

[47] J. Barre', D. Mukamel, and S. Ruffo. Inequivalence of ensembles in a system with long range interactions. *Phys. Rev. Lett.*, 87:030601, 2001.

[48] F. Borgonovi, G. L. Celardo, M. Maianti, and E. Pedersoli. Broken ergodicity in classically chaotic spin systems. *J. of Stat. Phys.*, 116:1435, 2004.

[49] F. Borgonovi, G. L. Celardo, A. Musesti, R. Trasarti-Battistoni, and P.Vachal. Topological non-connectivity threshold in long-range spin systems. *Phys Rev E*, 73:026116, 2006.

[50] G. L. Celardo, J. Barre, F. Borgonovi, and S. Ruffo. The non-ergodicity threshold: Time scale for magnetic reversal. *Phys. Rev. E*, 73:011108, 2006.

[51] J. Barre', F. Bouchet, T. Dauxois, and S. Ruffo. Large deviation techniques applied to systems with long-range interactions. *J. Stat. Phys.*, 119:677, 2005.

[52] R. Salazar, R. Toral, and A. R. Plastino. Numerical determination of the distribution of energies for the xy-model. *Physica A*, 305:144, 2002.

[53] D. P. Landau, S.-H. Tsai, and M. Exler. A new approach to monte carlo simulations in statistical physics: Wang-landau sampling. *Am. J. Phys.*, 72:1294, 2004.

[54] L. D. Landau and E. M. Lifshitz. *Course Of Theoretical Physics - Volume V - Statistical Physics - Part I.* Pergamon Press, 3rd edition, 1980.

[55] R. B. Griffiths, C.-Y. Weng, and J. S. Langer. Relaxation times for metastable states in the mean-field model of a ferromagnet. *Phys. Rev.*, 149:301, 1966.

[56] V. P. Antropov, S. V. Tretyakov, and B. N. Harmon. Spin dynamics in magnets: Quantum effects and numerical simulations. *J. App. Phys.*, 81:3961, 1997.

[57] H. A. Kramers. *Physica*, 7:284, 1940.

[58] P. Hanggi, P. Talkner, and M. Borkovec. *Rev. Mod. Phys.*, 62:2, 1990.

[59] F.Borgonovi, G.L.Celardo, B. Goncalves, and L. Spadafora. Magnetic reversal time in open long range systems. *e-print cond-mat*, page 0710.3935, 2007.

[60] R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete Mathematics.* Addison-Wesley, 1989.

[61] R. Sedgewick. *Algorithms In Java - Part I-IV.* Addison-Wesley, 3rd edition, 2002.

[62] S. Redner. *A Guide To First-Passage Processes.* Cambridge University Press, Cambridge, 2001.

[63] B. D. Hughes, M. F. Shlesinger, and E. W. Montroll. Random walks with self-similar clusters. *Proc. Nat. Acad. Sci.*, 78:3287, 1981.

[64] M. F. Shlesinger, G. M. Zaslavsky, and J. Klafter. Strange kinetics. *Nature*, 363:31, 1993.

[65] B. Kahng and S. Redner. Scaling of the first-passage time and the survival probability on exact and quasi-exact self-similar structures. *J. Phys. A: Math. Gen.*, 22:887–902, 1989.

[66] C. M. Bender and S. A. Orszag. *Advanced Mathematical Methods for Scientists and Engineers*. McGraw-Hill, New York, 1978.

[67] M. Livio. *The Golden Ratio: The Story of PHI, the World's Most Astonishing Number*. Broadway Books, New York, 2003.

[68] Euclid. *Elements, Book VI, Definition 3*. c. 300BC.

[69] R. Metzler and J. Klafter. The restaurant at the end of the random walk: recent developments in the description of anomalous transport by fractional dynamics. *J. Phys. A: Math. Gen.*, 37:R161–R208, 2004.

[70] S. Boettcher, H. G. Katzgraber, and D. Sherrington. Local field distributions in spin glasses. *e-print cond-mat*, page 0711.3934, 2007.

[71] S. Boettcher and B. Goncalves. Anomalous diffusion on the hanoi networks. *e-print cond-mat*, page 0802.2757, 2008.

[72] T. Carson and R. Impagliazzo. Experimentally determining regions of related solutions for graph bisection problems.

[73] T. Dimitriou and R. Impagliazzo. Go with the winners for graph bisection. In *Proceedings of the ninth annual ACM-SIAM symposium on Discrete algorithms*, 1998.

[74] M. Jerruma and G. B. Sorkin. The metropolis algorithm for graph bisection. *Disc. App. Math.*, 82:155, 1998.

[75] G. R. Schreiber and O. C. Martin. Cut size statistics of graph bisection heuristics. *SCIAM J. on Opt.*, 10:231, 1999.

[76] D. Sherrington J. R. Banava and and N. Sourlas. Graph bipartitioning and statistical mechanics. *J. Phys. A: Math. Gen.*, 20:L1, 1987.

[77] S. Boettcher and A. G. Percus. Extremal optimization for graph partitioning. *Phys. Rev. E*, 64:026114, 2001.

[78] S. Boettcher. Extremal optimization of graph partitioning at the percolation threshold. *J. Phys. A: Math. Gen.*, 32:5201, 1999.

[79] Y. Fu and P. W. Anderson. Application of statistical mechanics to np-complete problems in combinatorial optimisation. *J. Phys. A: Math. Gen.*, 19:1605, 1986.

[80] J. Reichardt and S. Bornholdt. Partitioning and modularity of graphs with arbitrary degree distribution. *e-print cond-mat/0606295*, 2006.

[81] K. Y. M. Wong and D. Sherrington. Graph bipartitioning and spin glasses on a random network of fixed finite valence. *J. Phys. A: Math. Gen.*, 20:L793, 1987.

[82] K. Y. M. Wong, D. Sherrington, P. Mottishaw, R. Dewar, and C. De Dominicis. Graph partitioning and dilute spin glasses: the minimum cost solution. *J. Phys. A: Math. Gen.*, 21:L99, 1988.

[83] Z. Wu, L. A. Braunstein, S. Havlin, and H. E. Stanley. Transport in weighted networks: Partition into superhighways and roads. *Phys. Rev. Lett.*, 96:148702, 2006.

[84] B. Bollobás, editor. *Graph theory and combinatorics*. Academic Press, 1984.

[85] R. Diestel. *Graph theory*. Number 173 in Graduate Texts in Mathematics. Springer-Verlag, Heidelberg, 3rd edition, 2005.

[86] B. Pittel and N. C. Worlmald. Counting connected graphs inside-out. *J. Comb. Th. B*, 93:127, 2005.

[87] R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.

[88] Busacker and Saaty. *Finite graphs and networks*. McGraw-Hill Book Company, 1965.

[89] R. Pastor-Satorras and A. Vespignani. *Evolution and Structure of the Internet : A Statistical Physics Approach*. Cambridge University Press, 2004.

[90] M. E. J. Newman. *SCIAM Review*, 45:167, 2003.

[91] A. Barrat, M. Barthélemy, R. Pastor-Satorras, and A. Vespignani. The architecture of complex weighted networks. *PNAS*, 101:3747, 2004.

[92] R. Guimerà, S. Mossa, A. Turtschi, and L. A. N. Amaral. The worldwide air transportation network: Anomalous centrality, community structure, and cities' global roles. *Proc. Nat. Acad. Sci.*, 102:7794, 2005.

[93] E. Almaas, B. Kovács, T. Vicsek, Z. N. Oltvai, and A.-L. Barabási. Global organization of metabolic fluxes in the bacterium escherichia coli. *Nature*, 427:839, 2004.

[94] S. H. Yook, H. Jeong, A.-L. Barabási, and Y. Tu. Weighted evolving networks. *Phys. Rev. Lett.*, 86:5835, 2001.

[95] M. E. J. Newman. The structure of scientific collaboration networks. *Proc. Natl. Acad. Sci.*, 98:404, 2001.

[96] M. E. J. Newman. Scientific collaboration networks. i. network construction and fundamental results. *Phys. Rev. E*, 64:016131, 2001.

[97] M. E. J. Newman. Scientific collaboration networks. ii. shortest paths, weighted networks, and centrality. *Phys. Rev. E*, 64:016132, 2001.

[98] L. A. Braunstein, S. V. Buldyrev, R. Cohen, S. Havlin, and H. E. Stanley. Optimal paths in disordered complex networks. *Phys. Rev. Lett.*, 91:168701, 2003.

[99] K.-I. Goh, J. D. Noh, B. Kahng, and D. Kim. Load distribution in weighted complex networks. *Phys. Rev. E*, 72:017102, 2005.

[100] Z. Wu, L. A. Braunstein, V. Colizza, R. Cohen, S. Havlin, and H. E. Stanley. Optimal paths in complex networks with correlated weights: The world-wide airport network. *Phys. Rev. E*, 74:056104, 2006.

[101] Y. Gang, Z. Tao, W. Jie, F. Zhong-Qian, and W. Bing-Hong. Epidemic spread in weighted scale-free networks. *Chi. Phys. Lett.*, 22:510, 2005.

[102] C. Zhou and Kurths J. *Phys. Rev. Lett.*, 96:034101, 2006.

[103] M. A. Serrano, M. Boguna, and R. Pastor-Satorras. Correlations in weighted networks. *Phys. Rev. E*, 74:055101(R), 2006.

[104] A. Barrat, M. Barthélemy, and A. Vespignani. Weighted evolving networks: Coupling topology and weight dynamics. *Phys. Rev. Lett.*, 92:228701, 2004.

[105] S.N. Dorogovtsev and J.F.F. Mendes. Minimal models of weighted scale-free networks. *e-print cond-mat/0408343*, 2004.

[106] G. Bianconi. Emergence of weight-topology correlations in complex scale-free networks. *Europhys. Lett.*, 71:1029, 2005.

[107] M. Barthélemy and A. Flammini. Optimal traffic networks. *J. Stat. Mech*, page L07002, 2006.

[108] M. Barthelemy, B. Gondranb, and E. Guichard. Spatial structure of the internet traffic. *Phys. A*, 319:633, 2003.

[109] P. J. MacDonald, E. Almaas, and A. L. Barabási. *Euro. Phys. Lett.*, 72:308, 2005.

[110] L. Dall'Asta. Inhomogeneous percolation models for spreading phenomena in random graphs. *J. Stat. Phys.*, 08:P08011, 2005.

[111] M. Molloy and B. Reed. A critical point for random graphs with a given degree sequence. *Random Structures and Algorithms*, 6:161, 1995.

[112] http://www.transtats.bts.gov/.

[113] J. J. Ramasco and B. Goncalves. Transport on weighted networks: when correlations are independent of degree. *Phys. Rev. E*, 76:066106, 2007.

[114] M. Latapy and C. Magnien. Measuring fundamental properties of real-world complex networks. *e-print cs.NI*, page 0609115, 2006.

[115] S. H. Lee, P.-J. Kim, and H. Jeong. Statistical properties of sampled networks. *Phys. Rev. E*, 73:016102, 2006.

[116] B. Padmanabhan and Y. Yang. Clickprints on the web: Are there signatures in web browsing data? *e-print http://knowledge.wharton.upenn.edu/papers/1323.pdf*, 2006.

[117] H. A. Simon. On a class of skew distribution functions. *Biometrika*, 42:425, 1955.

[118] G. U. Yule. A mathematical theory of evolution, based on the conclusions of dr. j. c. willis, f.r.s. *Phil. Trans. of the Roy. Soc. of Lon. B, Bio*, 713:21, 1924.

[119] S. Bornholdt and H. Ebel. World wide web scaling exponent from simons 1955 models. *Phys. Rev. E*, 64:035104(R), 2001.

[120] S.N. Dorogovtsev, J.F.F. Mendes, and A.N. Samukhin. Www and internet models from 1955 till our days and the "popularity is attractive" principle. *cond-mat*, page 0009090, 2000.

[121] A. Capocci, V.D.P. Servedio, F. Colaiori, L.S. Burio, D. Donato, S. Leonardi, and G. Caldarelli. Preferential attachment in the growth of social networks: the case of wikipedia. *e-print physics*, page 0602026, 2006.

[122] A. L. Barabási, H. Jeonga, Z. Néda, E. Ravasz, A. Schubertd, and T. Vicsek. Evolution of the social network of scientific collaborations. *Phys. A*, 311:590, 2002.

[123] S. Dorogovtsev and J. F. F. Mendes. *Evolution of Networks: From Biological nets to the Internet and WWW.* Oxford University Press, 2003.

[124] P. L. Krapivsky, S. Redner, and F. Leyvraz. Connectivity of growing random networks. *Phys. Rev. Lett.*, 85:4629, 2000.

[125] P. L. Krapivsky and S. Redner. Organization of growing random networks. *Phys. Rev. E*, 63:066123, 2001.

[126] S. N. Dorogovtsev and J. F. F. Mendes. Scaling properties of scale-free evolving networks: Continuous approach. *Phys. Rev. E*, 63:056125, 2001.

[127] F. Wu and B. A. Huberman. Novelty and collective attention. *Proc. Nat. Acad. Sci.*, 104:17599, 2007.

[128] A. Vazquez. Impact of memory on human dynamics. *Phys A*, 373:747, 2007.

[129] F. Menczer M.R. Meiss and A. Vespignani. On the lack of typical behavior in the global web traffic network. In *Proc. WWW*, 2005.

[130] A.-L. Barabási. The origin of bursts and heavy tails in human dynamics. *Nature*, 435:207, 2005.

[131] A. Vázquez, J. G. Oliveira, Z. Dezso, K.-I. Goh, I. Kondor, and A.-L. Barabási. Modeling bursts and heavy tails in human dynamics. *Phys. Rev. E*, 73:036127, 2006.

[132] J. G. Oliveira and A. Vazquez. Impact of interactions on human dynamics. *e-print cond-mat*, page 0710.4916, 2007.

[133] J. G. Oliveira and A.-L. Barabasi. Human dynamics: Darwin and einstein correspondence patterns. *Nature*, 437:1251, 2005.

[134] B. Goncalves and J. J. Ramasco. Human dynamics revealed through web analytics. *e-print cond-mat*, page 0803.4018, 2008.

[135] J. D. Murray. *Mathematical Biology I: An Introduction*. Springer, 3rd edition, 2002.

[136] P. Lemey, O. G. Pybus, B. Wang, N. K. Saksena, M. Salemi, and A.-M. Vandamme. Tracing the origin and history of the hiv-2 epidemic. *PNAS*, 100:6588, 2003.

[137] WHO. *Advocacy & Activity Guide For HIV/AIDS*. World Health Organization, 2000.

[138] P. Crépey, F. P. Alvarez, and M. Barthélemy. Epidemic variability in complex networks. *Phys. Rev. E*, 73:046131, 2006.

[139] H. S. Wilf. *Generatingfunctionology*. Academic Press, Inc., 1994.

[140] M. E. J. Newman, S. H. Strogatz, and D. J. Watts. Random graphs with arbitrary degree distributions and their applications. *Phys. Rev. E*, 64:026118, 2001.

[141] M. E. J. Newman. Spread of epidemic disease on networks. *Phys. Rev. E*, 66:016128, 2002.

[142] L. Dall'Asta. *Dynamical Phenomena on Complex Networks*. PhD thesis, Universite Paris, 2006.

[143] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts. Network robustness and fragility: Percolation on random graphs. *Phys. Rev. Lett.*, 85:5468, 2000.

[144] C. M. Grinstead and J. L. Snell. *Introduction To Probability*. American Mathematical Society, 2nd edition, 1997.

[145] M. Abramowitz and I. A. Stegun, editors. *Handbook of Mathematical Functions*. Dover, 1970.

[146] G. A. Korn and T. M. Korn, editors. *Mathematical Handbook For Scientists And Engineers*. Dover Publications, Inc., 2000.

[147] M. R. Spiegel. *Mathematical Handbook Of Formulas And Tables*. McGraw-Hill, 2nd edition, 1998.

[148] T. M. Apostol. *Calculus - Volume II - Multi Variable Calculus and Linear Algebra With Applications To Differential Equations And Probability*. John Wiley & Sons, 1969.

[149] http://fedora.redhat.com.

[150] http://drbl.sourceforge.net/.

[151] http://www.cs.wisc.edu/condor/.

[152] http://clusterknoppix.sw.be/.

[153] http://pareto.uab.es/mcreel/parallelknoppix/.

[154] http://dirk.eddelbuettel.com/quantian.html.

[155] http://bccd.cs.uni.edu/.

[156] http://clusterknoppix.sw.be/knx-install.htm.

[157] http://openmosix.sourceforge.net/.

[158] http://rom-o-matic.net/.

[159] http://www.openssh.com/.

[160] http://sourceforge.net/forum/?group_id=73280.

[161] http://java.sun.com/j2se/1.5.0/.

[162] http://www-unix.mcs.anl.gov/mpi/.

[163] http://www-unix.mcs.anl.gov/mpi/mpich/.

[164] http://www.lam-mpi.org/.

[165] http://www.cs.wisc.edu/condor/manual/v6.7/.

[166] http://www.bgoncalves.com/online/condor/.

Index