Helsinki University of Technology
Faculty of Information and Natural Sciences
Department of Information and Computer Science

Petri Savola

Applications of Combinatorial Algorithms in Statistical Physics

Master's thesis submitted in partial fulfilment of the requirements for the degree of Master of Science in Technology

Espoo, 25th May 2009

Supervisor: Prof. Pekka Orponen
Instructor: Prof. Pekka Orponen

HELSINKI UNIVERSITY OF TECHNOLOGY
Faculty of Information and Natural Sciences
Degree Programme of Computer Science and Engineering

ABSTRACT OF MASTER'S THESIS

Author: Petri Savola

Date: 25th May 2009

Pages: vii + 49

Title of thesis: Applications of Combinatorial Algorithms in Statistical Physics

Professorship: Theoretical Computer Science

Code: T-79

Supervisor: Prof. Pekka Orponen

Instructor: Prof. Pekka Orponen

One of the fundamental models in statistical physics is the Ising model. The model consists of spins, which can take value 1 or -1 and interactions between the spin pairs. The energy landscape of the spin glass systems is very complex and the problem of finding the spin configuration with the minimal number of unsatisfied interactions is very difficult.

In this thesis, we will introduce a method to generate all the metastable states of a spin glass system. This method can be used for example to explore the energy landscape of the system. The method is based on a binary search tree, which is generated in such a way that each path from the root to a leaf corresponds to an unique spin configuration. By considering simple spin glass systems, such as the hexagonal lattice, we can efficiently prune the search tree in such a way that a random local minimum can be found without backtracking.

We will show that the number of metastable states is exponential as a function of spins by using a classical search tree size estimation algorithm. Also, the reliability of the estimate is considered both analytically and empirically.

Some other applications, such as uniform sampling of the metastable states, are considered. An algorithm for almost uniform sampling is presented and it is used to obtain the energy distribution of the metastable states. Finally, we apply our method to check whether the phase transition between spin glasses and ferromagnetic systems affects the number of metastable states.

Keywords: Spin glass, Hexagonal lattice, Search tree size estimation, Sampling

TEKNILLINEN KORKEAKOULU
Informaatio- ja luonnontieteiden tiedekunta
Tietotekniikan koulutusohjelma/tutkinto-ohjelma

DIPLOMITYÖN TIIVISTELMÄ

Tekijä	Petri Savola	Päiväys	25. toukokuuta 2009
		Sivumäärä	vii + 49
Työn nimi	Kombinatoristen algoritmien soveltaminen tilastolliseen mekaniikkaan		
Professuuri	Tietojenkäsittelyteoria	Koodi	T-79
Työn valvoja	Prof. Pekka Orponen		
Työn ohjaaja	Prof. Pekka Orponen		

Yksi keskeisistä tilastollisen mekaniikan malleista on Ising-malli. Malli koostuu spineistä, jotka voivat saada arvon 1 tai -1, ja spin-parien välisistä vuorovaikutuksista. Spin-systeemin energiamaasto on monimutkainen ja sellaista systeemin tilaa, joka minimoi tyytymättömien vuorovaikutusten määrän, on erittäin vaikea löytää.

Tässä diplomityössä esitellään uusi menetelmä, jolla voidaan generoida kaikki annetun spin-systeemin metastabiilit tilat. Menetelmää voidaan käyttää esimerkiksi systeemin energiamaaston tutkimiseen. Menetelmä perustuu binääripuuhun, joka generoidaan siten, että jokainen polku juuresta lehteen vastaa yksikäsitteisesti yhtä tilaa. Kun tarkastellaan yksinkertaisia spin-systeemejä, kuten kaksiulotteista heksagonaalista hilaa, voidaan hakupuuta karsia tehokkaasti siten, että kukin metastabiili tila voidaan löytää ilman peruuttamista.

Työssä osoitetaan numeerisesti, että metastabiilien tilojen lukumäärä on eksponentiaalinen spinien lukumäärän funktiona käyttämällä klassista hakupuun koon arviointimenetelmää. Lisäksi tulosten luotettavuutta arvioidaan sekä analyyttisesti että kokeellisesti.

Työssä tarkastellaan myös joitain muita menetelmän käyttökohteita, kuten metastabiilien tilojen tasaista otantaa. Esitetään algoritmi, jolla voidaan generoida tiloja melkein tasaisesti ja tutkitaan sen avulla tilojen energiajakaumaa. Lopuksi tutkitaan vielä, vaikuttaako spin-lasisysteemien ja ferromagneettisten systeemien välinen faasitransitio esimerkiksi metastabiilien tilojen lukumäärään.

Avainsanat

Spin-lasi, Heksagonaalinen hila, Hakupuun koon arviointi, Otanta

Preface

This master's thesis has been written at the Department of Information and Computer Science. First of all, I would like to thank my instructor and supervisor Prof. Pekka Orponen for having the patience and time to read and comment my work numerous times, and for the helpful discussion sessions concerning the work.

I would also like to express my gratitude to Prof. Alexander Hartmann for his hospitality and many helpful suggestions during my stay in Oldenburg in July 2008. Using the method proposed in this thesis to study the spin glass - ferromagnet phase transition was just one of his magnificent ideas and contributions.

Then, I would like to thank my colleagues, friends, family and specially my wife Annika for the great support during the year.

May 2009
Espoo, Finland

Petri Savola

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Many problems in statistical physics, including the Ising spin glass ground state problem, are difficult to solve. In this thesis, we will present a new combinatorial approach for exploring the energy landscape of spin glass systems.

Spin glasses are frustrated magnetic systems with random distributions of ferromagnetic and antiferromagnetic interactions. They have complicated energy landscapes with an exponential number of local minima. We present a method for generating all the local minima of a given spin glass model and consider the problem of counting and sampling the energy minima. The counting is based on a classical search tree size approximation method by Knuth [9] and we will use the same idea recursively to obtain local minima samples almost uniformly.

The general case of spin glass ground state problem is known to be NP-hard and only a few special cases have been solved analytically [7,20]. Due to the complexity of the general case, we will mainly be considering a special case, the hexagonal lattice.

1.1 Outline

In Chapter 2 we introduce the basics of spin glass models and the definitions which will be used throughout the thesis. Additionally, we will show the equivalence between spin configurations and edge colorings and prove that the spin glass ground state problem is NP-hard.

In Chapter 3 the hexagonal lattice model is more carefully explained. The algorithm for generating the local energy minima is introduced and the main ideas behind the algorithm are explained and proven. In the end of the chapter, a couple of results are shown to verify that the number of energy minima really grows at an exponential rate in the size of the system.

In Chapter 4 a few methods to improve the performance of the algorithm are introduced. We will also show how much the improvements actually affect the results. Finally, a formula for the variance of the sampling method is derived analytically.

In Chapter 5 we will solve analytically the number of energy minima for two simple spin glass systems and compare the real results with the search tree leaf size estimates obtained from the algorithm.

In Chapter 6 we will consider the problem of uniform sampling of the energy minima. We will present a way to achieve almost uniform samples and use it to

obtain an energy distribution histogram, which shows that the energy per spin value of the minima seems to concentrate on a small interval as the lattice size grows. This indicates that only a very small proportion of the minima have energy close to the ground state energy.

In Chapter 7 we will introduce the phase transition phenomenon between spin glasses and ferromagnetic systems. We will use our method to check whether the number of energy minima changes at the phase transition point. We will also consider whether the phase transition can be noticed by other means.

In Chapter 8 we will take a look at what else is known about spin glasses. A lot of research has been done on this topic recently, and some of the research is related quite closely to the counting and sampling problems.

Finally, in Chapter 9 we will present brief conclusions, summarize the thesis and propose some new research topics based on this subject.

Chapter 2

Spin Glasses and Energy Minima

2.1 The Ising model

The first spin glass model was introduced by Ernst Ising in 1925 to model the phenomenon of ferromagnetism [3]. The magnetic properties of matter are caused by groups of electrons spinning in the same direction [26], but naturally there is no interaction between the electrons on different sides of a magnet, so how do the electrons know their spinning direction? This is explained by the Ising model, which demonstrates how the local forces between the electrons affect the global properties of the system.

Let J_{ij} denote the interaction between two spins i and j. In the Ising model, for each spin pair i, j one defines:

$$J_{ij} = \begin{cases} 1 & \text{if the interaction is ferromagnetic} \\ -1 & \text{if the interaction is antiferromagnetic} \\ 0 & \text{if there is no interaction.} \end{cases}$$

If we consider a system of N spins, which can take values 1 or -1 (up or down) and denote the value of spin i by σ_i, the *energy* can be defined by summation over all the interactions

$$H(\sigma) = -\sum_{ij} J_{ij}\sigma_i\sigma_j. \tag{2.1}$$

Instead of calculating the energy as a function of spins, it is often useful to consider the energy as a function of satisfied and unsatisfied interactions. To minimize the energy, a pair of spins prefer to be similarly oriented if the interaction between them is ferromagnetic and differently oriented if the interaction between them is antiferromagnetic. This follows from the fact that the energy is minimized by setting $J_{ij}\sigma_i\sigma_j = 1$ for as many pairs as possible.

We consider the spin glass system as a graph where nodes are the spins and edges are the interactions. An edge $\{i, j\}$ is $satisfied$ $(green)$ if $J_{ij}\sigma_i\sigma_j = 1$ and and $unsatisfied$ (red) if $J_{ij}\sigma_i\sigma_j = -1$. Denote the number of green edges by H_G and the number of red edges by H_R. Then the energy can then be expressed as a function of the edge coloring

$$H = H_R - H_G. \tag{2.2}$$

Valid edge colorings are the ones that are induced by some spin configuration. There exist also edge colorings which do not correspond to any spin configuration; these colorings are considered to be invalid. Each spin configuration induces an edge coloring and conversely, each valid edge coloring of the graph represents two spin configurations due to the up-down symmetry of the spins, so there is an equivalence between spin configuration pairs and valid edge colorings. If the graph has M edges, equation (2.2) can be simplified to

$$H = H_R - (M - H_R) = 2H_R - M = -2H_G + M \tag{2.3}$$

because an edge is always either red or green. This is a useful result as the energy can be expressed as a function of red or green edges only.

2.2 Energy landscapes

Often one is interested in finding the minimum energy or a *ground state* of a spin glass system. The ground state is a spin configuration which maximizes the number of satisfied edges or minimizes the number of unsatisfied edges. In a system which contains no antiferromagnetic interactions, finding a ground state is easy, because a configuration with all spins up or all spins down would satisfy all the edges with a ground state energy of $-M$. Systems with only these kinds of interactions are called *ferromagnetic*.

However, if we introduce antiferromagnetic interactions, finding the ground state becomes much more difficult because of an effect called *frustration*. It is no longer possible to satisfy all of the interactions and there will exist cycles with frustrated edges so the ground state energy is much more difficult to determine. Note that the consistency of spin orientations requires that each cycle of the graph with an odd number of antiferromagnetic edges must have an odd number of red edges and each cycle with an even number of antiferromagnetic edges must have an even number of red edges, i.e. the parities of antiferromagnetic and red edges need to agree. In all other cases the edge coloring does not represent any spin configuration and is considered to be invalid.

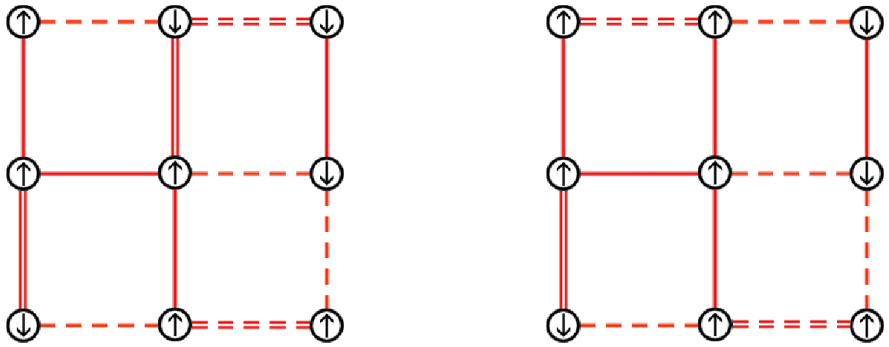

Figure 2.1: An example of a small spin glass system

Figure 2.1 demonstrates the edge coloring idea. Solid lines represent ferromagnetic edges, dashed lines represent antiferromagnetic edges and double lines represent the

unsatisfied edges. The left spin configuration does not represent a local minimum, because one of the spins (top middle) could be flipped to decrease the number of unsatisfied edges. The resulting configuration after the spin flip can be seen on the right. Note that it is still not the ground state because there exist configurations with fewer unsatisfied edges. This spin system has several different ground states, which have only two unsatisfied edges.

Even if we find a state with very low energy, it is difficult to prove that it is a global minimum. Additionally, a simple greedy search cannot be used to find the ground state because there exist an exponential number of local minima and the algorithm would get stuck in one of those. A *local minimum* is a state where no single spin flip decreases the energy; they are also called *metastable states*. Furthermore, a *proper* local minimum is a state where any single flip increases the energy.

In ferromagnetic lattices the low energy states are such that there are many large clusters of similarly oriented spins. If we consider the edge coloring, these states are characterized by islands of green edges, separated by series of red edges. If there is a single red edge somewhere in the lattice, there must be a lot of other red edges, because any path between the endpoints of a red edge must contain an odd number of red edges. The red edges separate the clusters from each other in such a way that they either form cycles or paths through the lattice. The idea is demonstrated in Figure 2.2.

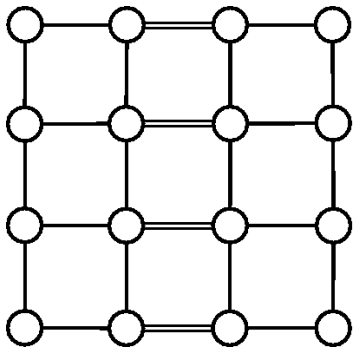

Figure 2.2: Example of a ferromagnetic lattice, where a path of red edges (double lines) goes through the lattice and separates two clusters of similarly oriented spins

Spin glass lattices, which consist of both ferromagnetic and antiferromagnetic edges, are a bit more complex, but they behave similarly as ferromagnetic lattices if we consider that red ferromagnetic and green antiferromagnetic edges separate the spin clusters while green ferromagnetic and red antiferromagnetic edges form clusters. However, it is very often the case that there are not many edges of same type near each other, so usually big clusters of similarly oriented spins do not form. The visualization and analysis of spin glass lattices is more complex than the analysis of ferromagnetic lattices.

In this thesis we will introduce methods to count and sample the local minima of spin glass systems to learn more about the complex energy landscape. To get things started we will prove in the next section that finding a ground state of a spin glass system is an NP-hard problem. This means that any problem in the complexity class

NP is reducible to the ground state problem, but the ground state problem itself is not necessarily in NP.

2.3 Computational complexity

If any instance of problem A can be easily transformed into some instance of problem B, we say that A *reduces* to B, denoted by $A \leq_P B$ if the reduction can be done in polynomial time. This means that problem B is at least as hard as problem A, because we can solve any instance of problem A by first mapping it into a problem B instance and then applying an algorithm which solves the problem B. A very typical way of proving complexity results is to construct reductions of this type.

Although the spin glass model seems very simple, we shall prove that the problem of maximizing the number of satisfied edges of a spin glass system (finding the ground state) is difficult. Earlier it has been proven that the ground state decision problem is NP-complete by showing that any instance of the MAX-CUT problem can be mapped in polynomial time into a ground state problem instance [21].

We will present two new reductions to show that the MAX-2-SAT problem and the ground state problem are equally hard, and to show that the ground state decision problem is actually NP-complete because MAX-2-SAT decision problem is known to be NP-complete [2, 16]. The reductions are interesting theoretic results but they can also be useful in practice because if there exists an effective tool for solving one of the given problems, the tool can be used to solve the other problem as well. In fact, many problems are solved by first mapping them into a SAT-instance and then applying one of the efficient SAT-solvers [1].

We define that 2-SAT formula is a Boolean formula which consists of disjunctions of literal pairs connected by conjunctions. The MAX-2-SAT and ground state decision problems can be formally presented as follows.

MAX-2-SAT
Given a 2-SAT formula F consisting of N variables, M clauses and a constant k, is there any truth assignment that satisfies k clauses?

GROUND STATE
Given a spin glass system of N spins, interaction matrix J (elements $J_{ij} \in \{-1, 0, 1\}$) and a constant H, is the energy of the ground state at most H?

Theorem 1. *GROUND STATE* $\leq_P$ *MAX-2-SAT:*

Proof. Let J be an interaction matrix for a spin glass system with M edges. We shall construct a 2-SAT formula F with $2M$ clauses such that for every spin configuration of J with energy of at most H there exists a truth assignment for F which satisfies at least $(3M - H)/2$ of the clauses and vice versa.

Assume that spin i pointing upwards or downwards represents σ_i being true or false respectively. For each ferromagnetic interaction between spins i and j (spins prefer to be similarly oriented) we include in the formula the disjunction

$$(\sigma_i \wedge \sigma_j) \vee (\neg\sigma_i \wedge \neg\sigma_j)$$

which is equivalent to the CNF-formula

$$(\neg\sigma_i \vee \sigma_j) \wedge (\sigma_i \vee \neg\sigma_j)$$

And similarly for each antiferromagnetic interaction we obtain

$$(\sigma_i \vee \sigma_j) \wedge (\neg\sigma_i \vee \neg\sigma_j).$$

Thus, each edge in the spin glass system corresponds to two 2-SAT clauses. If an edge is satisfied in a spin configuration, both of these clauses are satisfied by the corresponding truth assignment. If an edge is not satisfied, then exactly one of the clauses is satisfied. If we consider a spin glass system consisting of M edges and a spin configuration with energy $H = H_R - H_G$, the number of satisfied edges will be $H_G = (M - H)/2$ because $H_R = M - H_G$. The number of unsatisfied edges is then $H_R = (M + H)/2$. A spin configuration with energy H thus corresponds to a truth assignment which satisfies $H_R + 2H_G = (M + H)/2 + 2 \cdot (M - H)/2 = (3M - H)/2$ of the 2-SAT clauses. Hence, a spin configuration with energy of at most H corresponds to a truth assignment which satisfies at least $(3M - H)/2$ clauses and vice versa.

Hence, the ground state of the system is H_{min} if and only if the maximum number of satisfied edges is $(M - H_{min})/2$ which holds if and only if the maximum number of satisfiable 2-SAT clauses is $(3M - H_{min})/2$. Any spin glass ground state instance can be transformed into a MAX-2-SAT instance, and the truth assignment which satisfies the maximum number of clauses represents the optimal spin configuration. If the maximum number of the $2M$ clauses that can be satisfied is k then the ground state energy is $H_{min} = 3M - 2k$. □

Theorem 2. *MAX-2-SAT* $\leq_P$ *GROUND STATE:*

Proof. Let F be a 2-SAT formula with M clauses and N variables. We shall construct a spin glass system with $3(M + N + 1)$ spins and interaction matrix J such that for every truth assignment of F which satisfies at least k of the clauses, there exists a pair of metastable spin configurations of J with energy of at most $-6M - 4k$ and vice versa.

There are four possible clauses with two literals: $a \vee b$, $\neg a \vee b$, $a \vee \neg b$ and $\neg a \vee \neg b$. Each of these clauses is mapped to a triangle of three spins as demonstrated in Figure 2.3 where solid and dashed lines represent ferromagnetic and antiferromagnetic bonds respectively. The triangles are designed with the idea that the third spin without label (from now on these nodes will be called x-nodes) always points upwards. Then exactly two of the edges are satisfied if the corresponding clause is satisfied and if the clause is not satisfied, all of the edges of the triangle will be unsatisfied. These triangles are called *clause-triangles* as they correspond to the clauses.

The idea of the reduction is demonstrated in Figure 2.4. We construct an additional triangle with only ferromagnetic edges for each variable which exists in the formula, including the x-variable. We call these triangles *variable-triangles* as they correspond to the variables. The variable-triangles are then linked to the clause-triangles in such a way that if a variable label exists in a clause-triangle, we introduce ferromagnetic edges from all nodes in the corresponding variable-triangle. The subgraph consisting of a variable-triangle and the respective variable-nodes is called a *variable-cluster*.

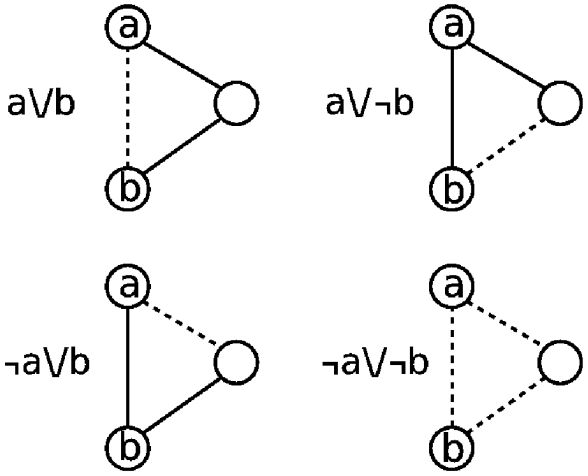

Figure 2.3: 2-SAT to spin glass transformation

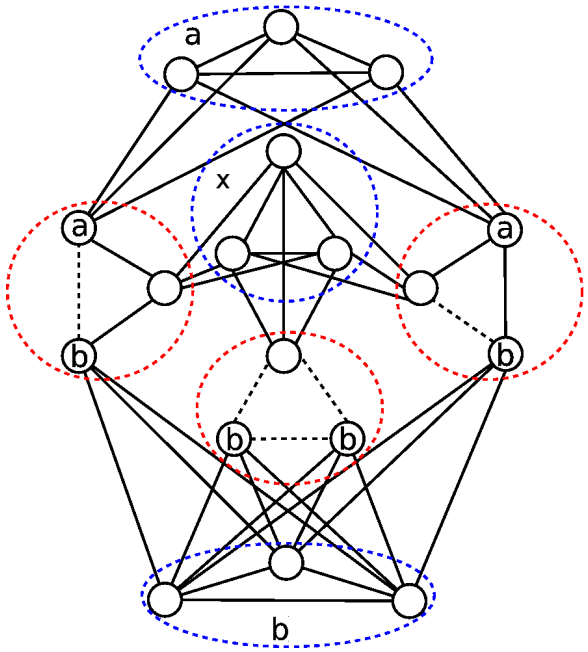

Figure 2.4: Spin glass system representing the 2-SAT formula $(a \vee b) \wedge (\neg b \vee \neg b) \wedge (a \vee \neg b)$

Why do we need such a complex structure? In principle we could replace the variable-triangles by simply introducing edges directly between the variables or by replacing the triangles with singleton nodes, but the ground state of this system would not necessarily correspond to the truth assignment which satisfies a maximal number of clauses. And the structure we introduced has a nice property that each symmetric pair of local minima of the system correspond to a unique truth assignment. This follows from the fact that all the variable-nodes have degree of 5 so we have a local minimum regardless of the edges of clause-triangles if all other edges are satisfied.

It is very easy to satisfy all the non-clause edges by simply assigning the spins of variable-clusters either up or down. This choice corresponds to the variable being

true or false in the truth assignment. The variable cluster triangles are big enough to ensure all the spins corresponding to a given variable must be similarly oriented in all local minima. This can be verified by considering the variable-cluster and its neighbor nodes and checking that the number of unsatisfied edges of the cluster is always minimal in the case where all spins in the whole cluster have similar orientation regardless of the orientation of the neighboring nodes.

Note that the nodes in the x-triangle have degree of $2+M$ and generally, the degree of a node in a variable-triangle will be two plus the number of times the variable occurs in the formula.

To remove the up-down symmetry of the spin glass system we can decide to set the orientation of the spins in the x-triangle, say, upwards. After this, we can easily read the truth assignment from the spin configuration by looking at the orientations of the variable-clusters.

If the original 2-SAT formula has M clauses and N variables, the corresponding spin glass system has $12M$ edges and $3(M+N+1)$ nodes. For each clause-triangle it is only possible to satisfy two of the edges due to frustration. Each clause-triangle with two satisfied edges represents a satisfied clause and clause-triangles with no satisfied edges represent the unsatisfied clauses. All other edges of the spin glass system are satisfied in the ground state and the number of these edges is $9M$. Thus, the number of satisfiable clauses in the original formula is at least k if and only if the number of satisfied edges in the spin glass is at least $9M + 2k$, which holds if and only if the energy is at most $H_R - H_G = M + 2(M - k) - (9M + 2k) = -6M - 4k$.

Hence, any 2-SAT formula can be transformed into a spin glass system and by finding the ground state we can find the maximum number of satisfied clauses as well. If the ground state of the spin glass instance is H_{min} then the maximum number of satisfied clauses is $(-H_{min} - 6M)/4$.

□

Chapter 3

Counting and Sampling of Metastable States

3.1 Hexagonal lattice

It is known that the number of metastable states of a spin glass system is exponential as a function of the number of spins [23], but it is interesting to know something more precise about the metastable states. For example, we will consider whether the number of metastable states is a reason why finding the ground state is so difficult. In this chapter we will introduce methods for counting and sampling them. Let N be the number of spins and denote the number of metastable states by N_S. Then it is natural to assume

$$N_S = e^{\alpha N} \tag{3.1}$$

where α is a constant which depends on the spin glass system. Values of α are typically very difficult to calculate, but have been calculated and estimated for some spin glass systems [13, 14]. We applied our method for the hexagonal lattice and the result was quite close to the existing estimates.

In principle we could consider any spin glass system, but for an arbitrary system, it is very difficult to estimate or calculate the number of local minima. Due to the complexity of the general case we consider a regular hexagonal lattice. It is a good candidate for several reasons: The structure is simple, but the number of local minima is still not easy to calculate. Another nice property is that the local minima of the system are also proper local minima because except for the boundaries all nodes in the lattice have degree of three. This is one of the main reasons why we decided to consider the hexagonal lattice instead of the square lattice, which is a more commonly used structure. The hexagonal lattice is also known as the *honeycomb lattice* [12].

3.2 Search tree generation

Let the number of edges in a spin glass system be M. Our method for sampling the metastable states is based on a binary search tree of height M. For each edge which exists in the spin glass system we will make a decision whether we satisfy the edge or not. If we decide to satisfy the edge, we will pick the left branch in the tree and

otherwise we will pick the right branch. By performing M decisions like this we have actually chosen an edge coloring for the system and as we discussed in the previous chapter, the edge coloring represents a spin configuration. Each path from the root to a leaf corresponds to a unique edge coloring.

Then we will introduce pruning rules for the tree in such a way that we prune out all the branches which do not lead to metastable states. For spin glass systems with simple structure, such as the hexagonal lattice, the pruning can be done quite efficiently without needing to backtrack.

Additionally, we will need to prune out the branches which would produce invalid edge colorings. This can be done by selecting a spanning tree of the graph instead of selecting all the edges and by coloring the edges of the spanning tree only, because the other edge colors will be determined by the coloring of the spanning tree. This follows from the fact that fixing one of the spins and the coloring of the spanning tree actually determines the whole spin configuration, which then determines the edge coloring as we have discussed before.

In the end of this chapter we will introduce a technique which can be used to estimate the size of a search tree. In our case, this technique will be used to estimate the number of leaves as we are interested in the number of local minima. But first, we will explain the pruning process in more detail.

3.3 Pruning rules

A hexagonal lattice consists of nodes with degree three and two. Nodes of degree two exist along the borders if we do not apply any periodical boundary conditions. A spin configuration is a local minimum if for each node there is at most one adjacent red edge. If there would be more adjacent red edges, we could simply flip the spin to obtain a state with lower energy and thus the original state would not be a local minimum.

To avoid invalid colorings of the edges, we will need to select a spanning tree as discussed in the previous section. Basically, any spanning tree would be fine, but the pruning process will be much easier if the spanning tree is chosen carefully. One of the spanning trees which allows easy pruning is a spiral shaped spanning path which starts from the center of the lattice, demonstrated in Figure 3.1. The main reason to choose this kind of spanning path is the ability to check efficiently whether a consistent partial coloring of the path can be completed to a full coloring. From this it follows that at each branching point we can immediately prune out the branches which do not lead into an edge coloring representing a local minimum.

Note that a local minimum of the spanning tree is not necessarily a local minimum of the underlying lattice, because the additional edges which belong to the lattice, but not to the tree, will not necessarily be green. Their color is determined in such a way that the parities of antiferromagnetic and red edges will agree for each cycle in the lattice.

Figure 3.2 demonstrates the search tree idea. It contains a search tree for a single ferromagnetic hexagon with six spins. The height of the search tree is five, because the number of the edges in the spanning path is five. In this case, the local minima

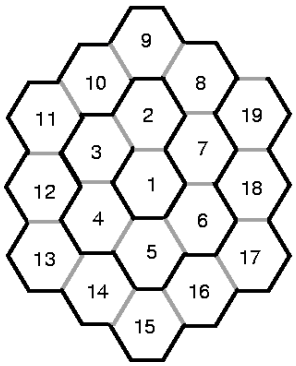

Figure 3.1: Spanning path for a hexagonal lattice

have either two red edges or no red edges at all and there cannot be two red edges in a row. Also note that if the sixth edge is required to be red to even out the parity of red edges, neither the first nor the fifth edge can be red. Solid lines represent the green edges (left branch - we decide to satisfy the edge) and dashed lines represent the red edges (right branch - we decide to leave the edge unsatisfied).

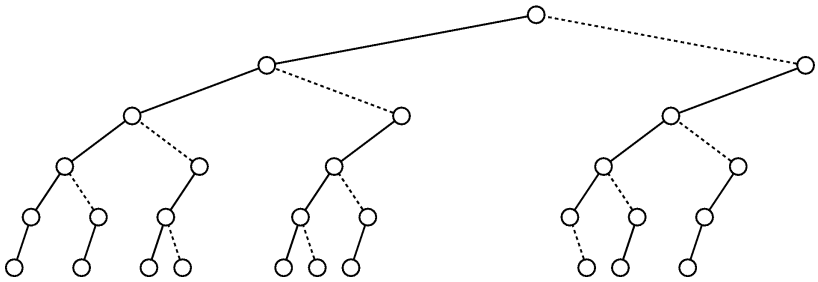

Figure 3.2: The search tree for a single ferromagnetic hexagon

The search tree looks slightly different if an odd number of the edges would be antiferromagnetic. In this case, the number of red edges would need to be odd as well and the resulting search tree would look like the tree in Figure 3.3. In this case the minima have either one or three red edges and the number of leaves is different, indicating that the number of metastable states differs between ferromagnetic systems and spin glass systems.

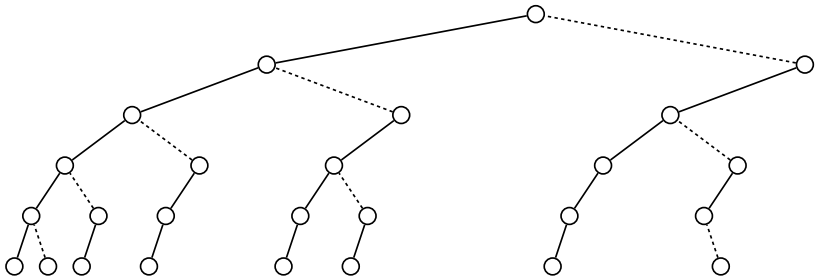

Figure 3.3: The search tree for a single hexagon with an odd number of antiferromagnetic bonds

The idea is not to generate the whole search tree because its size is exponential.

Instead, we generate the search tree on the fly while descending by pruning out the branches that will lead into a dead end. Typically this kind of pruning is very hard to do because arbitrarily long look-ahead is required but the spiral shaped spanning path seems to solve this problem for the hexagonal lattice. The idea is to color one hexagon at a time, starting from the central hexagon. Once a hexagon has received a valid coloring, we will move on to the next hexagon via the path, give it a valid coloring and so on.

As can be seen from Figure 3.1 the first hexagon has five edges to be colored, the second hexagon has four, third hexagon has three (two of the edges do not belong to the path and one was already colored because it is also part of the first hexagon) and so on. All of the hexagons have at least two uncolored edges and thus two degrees of freedom which is exactly enough to guarantee a valid coloring of a metastable state. One degree of freedom is required because the previous edge in the path may be red. The second degree of freedom is needed to guarantee that the parities of antiferromagnetic and red edges agree.

3.4 Algorithm for sampling

The basic schema for sampling the local minima follows these steps: First, we need to choose a spanning tree and enumerate its edges, preferably in such a way that the pruning can be done efficiently. Then, we start choosing colors for the edges in the given order and for each edge, if there are two possible colors, flip a coin to decide the color and proceed to the next edge. If only one color is available, we choose that without flipping a coin.

In this section we will introduce, in more detail, an algorithm which is designed to produce local minima samples (or actually, edge colorings) from a given hexagonal lattice. The algorithm will follow the schema and ideas presented earlier, but before presenting the algorithm we will need to introduce the following theorem and define a grammar for hexagonal lattice generation.

Theorem 3. *Let R be the radius of the lattice, meaning the number of perfect circles of hexagons around the central hexagon. With $R = 1$ the lattice consists of a single hexagon, with $R = 2$ there are seven hexagons and so on. The lattice has $3R^2 - 3R + 1$ hexagons, $6R^2$ spins and $9R^2 - 3R$ edges.*

Proof. The calculation for the number of hexagons is quite simple. A perfect circle of radius i of hexagons has exactly 6 corner-hexagons. Additionally, there are $i - 2$ hexagons between the corner-hexagons. Thus, the total number of hexagons in the i:th circle is $6 + 6(i - 2) = 6i - 6$. By summing over all circles up to R we get

$$1 + \sum_{i=1}^{R} (6i - 6) = 1 + 6\frac{R(R+1)}{2} - 6R = 3R^2 - 3R + 1. \tag{3.2}$$

The calculation for the number of edges is similar. If we consider the number of edges bordering the $(i - 1)$:th full circle of hexagons the corner-hexagons contribute 3 edges and the other hexagons contribute 2 to the sum. Thus, the border length is

Grammar $G = (V, \Sigma, P, S)$
$S \to 54B \mid 5 \mid \epsilon$
$B \to AB \mid C$
$A \to 3333233D$
$D3 \to 23D$
$D2 \to 2D$
$DC \to C \mid \epsilon$

Table 3.1: Grammar for hexagonal lattice generation

$6 \cdot 3 + 6 \cdot 2(i-2) = 12i - 6$. Additionally there are edges which connect this border with the previous one. The number of these edges equals the number of hexagons, which is $6i - 6$.

$$\sum_{i=1}^{R}(12i - 6 + 6i - 6) = -12R + 18\frac{R(R+1)}{2} = 9R^2 - 3R \tag{3.3}$$

The calculation of the number of spins is almost same as the edge calculation. The total number of spins is the same as the total number of edges in the borders of the complete circles of hexagons.

$$\sum_{i=1}^{R}(12i - 6) = -6R + 12\frac{R(R+1)}{2} = 6R^2 \tag{3.4}$$

□

If we are given a lattice of N hexagons, using Theorem 3 we get the equation $3R^2 - 3R + 1 = N$ and by solving it we obtain

$$R = \frac{3 + \sqrt{9 - 12(1-N)}}{6} = \frac{3 + \sqrt{12N - 3}}{6}$$

Now $\lfloor R \rfloor$ is the number of complete circles of hexagons around the central hex.

In order to be able to color a hexagon in such a way that we do not run into dead ends, we must know the number of path edges remaining to complete the hexagon in question. The number of path edges of hexagon x_i follows the sequence $x = \{5, 4, 3, 3, 3, 3, 2, 3, 3, ...\}$. Knowing that each full circle of hexagons around the central hexagon has six more hexagons than the previous one, we can design a grammar G which generates the sequence x. The grammar G is defined in Table 3.1.

Example 1. We want to generate a hexagonal lattice with $N = 19$ hexagons, so we need to know the degrees of freedom for 19 first hexagons. This kind of lattice can be seen in Figure 3.1. Using the grammar G we can derive a string using production $B \to AB$ exactly $\lceil R \rceil - 1 = 2$ times. The grammar gives us a string a of length 22 as follows:

$S \to 54B \to 54AB \to 54AAB \to 54AAC \to 543333233DAC$
$\to 543333233D3333233DC$
$\to^* 5433332332323232322323DDC$
$\to^* 543333233232323232323 = a$

Thus, $a \in L(G)$ and the sequence represents the degrees of freedom encountered. We only need to use the N first numbers from the string. Intuitively, 3 represents a corner hexagon and 2 represents the other hexagons. As the circle grows, the corner hexagons will become more rare. The actual algorithm itself is presented in Table 3.2.

Require: the number of hexagons N
  Compute $R \leftarrow \lceil \frac{3+\sqrt{12N-3}}{6} \rceil - 1$
  Using the production $B \to AB$ in grammar G exactly R times, produce string a
  Consider the string as a table of numbers
  $i \leftarrow 0$
  while $i < N$ **do**
    while $a[i] > 0$ **do**
      if previous edge red **then**
        choose green edge
      else
        if $a[i] > 2$ **then**
          choose red or green edge
        else
          if legal coloring cannot be found by choosing red **then**
            choose green edge
          else
            if legal coloring cannot be found by choosing green **then**
              choose red edge
            end if
          else
            choose red or green edge
          end if
        end if
      end if
      $a[i] \leftarrow a[i] - 1$
    end while
    $i \leftarrow i + 1$
  end while

Table 3.2: Algorithm for generating a random metastable state in a hexagonal spin glass lattice

3.5 Leaf size approximation

The previous sections introduce an efficient method for generating a search tree. In this section we consider estimating the number of its leaves (its leaf size) by using a simple algorithm proposed by Knuth [9]. Generally, the idea is the following.

Theorem 4. *Consider a random descent from the root to a leaf p at depth h in a search tree. Denote the number of children of node at depth i of the descent by d_i. Then $\prod_{i=0}^{h-1} d_i$ is an unbiased estimate for the leaf size of the search tree.*

Proof. Let L be the set of leaves. Let us consider a particular leaf $p \in L$ of the search tree at depth h. Denote the degrees of the nodes in the path between root and p by d_0, d_1, ..., d_{h-1}. Define a random variable $X : L \to \mathbb{R}$ as $X(p) = \prod_{i=0}^{h-1} d_i$. Then

$$E[X] = \sum_{p \in L} Pr(p)X(p) = \sum_{p \in L} \frac{1}{\prod_{i=0}^{h-1} d_i} \prod_{i=0}^{h-1} d_i = |L|. \tag{3.5}$$

□

Hence, the product of the degrees in the path from root to a leaf can be used as an unbiased estimate for the leaf size of a search tree. Let D_p be the number of branching points encountered during a descent to leaf p in a binary tree. It is actually the logarithm of the random variable X:

$$\log_2(X(p)) = \log_2 \prod_{i=0}^{h-1} d_i = \sum_{i=0}^{h-1} \log_2 d_i = D_p \tag{3.6}$$

If we want to estimate the size of a given search tree, we will simply perform random sampling in the tree as described in Algorithm 1 and introduce a counter to keep track of the number of branching points. Let n be the number of samples (descents). Then our search tree leaf size estimate is

$$N_S \approx \frac{1}{n} \sum_{i=1}^{n} 2^{D_i}. \tag{3.7}$$

3.6 Results

The experiments seem to verify that the number of local minima is exponential as a function of the number of spins. We used lattices of sizes 160, 216, 294, 384, 486, 600, 726, 864, 1014, 1176, 1350 and 1536 spins. These are the numbers that form perfect circles of hexagons around the central hex of the lattice. For each system size we used an ensemble of 1000 lattices and calculated the logarithm of the size estimate for each of them. The search tree leaf size estimate itself was obtained using 10000 descents per tree. The averages of these logarithms are plotted in Figures 3.4 and 3.5 along with standard deviation and the line which fits best to the data. Figure 3.4 shows the results for a ferromagnetic lattice where all edges are positive and Figure 3.5 shows the results for a spin glass lattice where edges are positive or negative with equal probability.

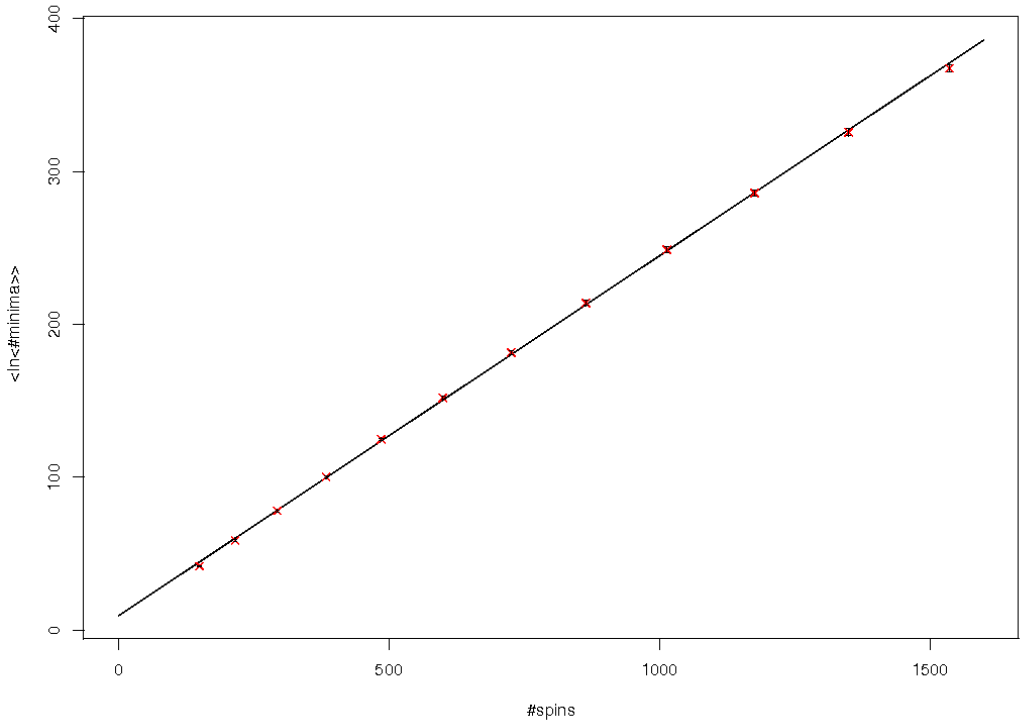

Figure 3.4: α estimation for ferromagnetic lattice using an ensemble of 1000 lattices and 10000 descents per system

By calculating the slope of the line we can find out the exponent α in the equation $N_S = e^{\alpha N}$. Our estimates for the value of α are 0.2354 for a ferromagnetic lattice and 0.2385 for a spin glass lattice. These results are in very good agreement with article [10]: their calculations for α resulted in values 0.2315 and 0.2350 respectively for a "brickwork lattice" of height 5. [1]

Although the fit seems to be linear, the slope of the line seems to decrease slowly as the number of spins grows. It is possible that our choice of lattices of up to 1536 spins is still not enough to represent an infinite lattice and the border effects cause a small error in the estimation of α. Small systems consisting of fewer than 160 spins were ignored to minimize the effect of the borders.

[1] Brickwork lattices are hexagonal lattices with unbounded width but finite height.

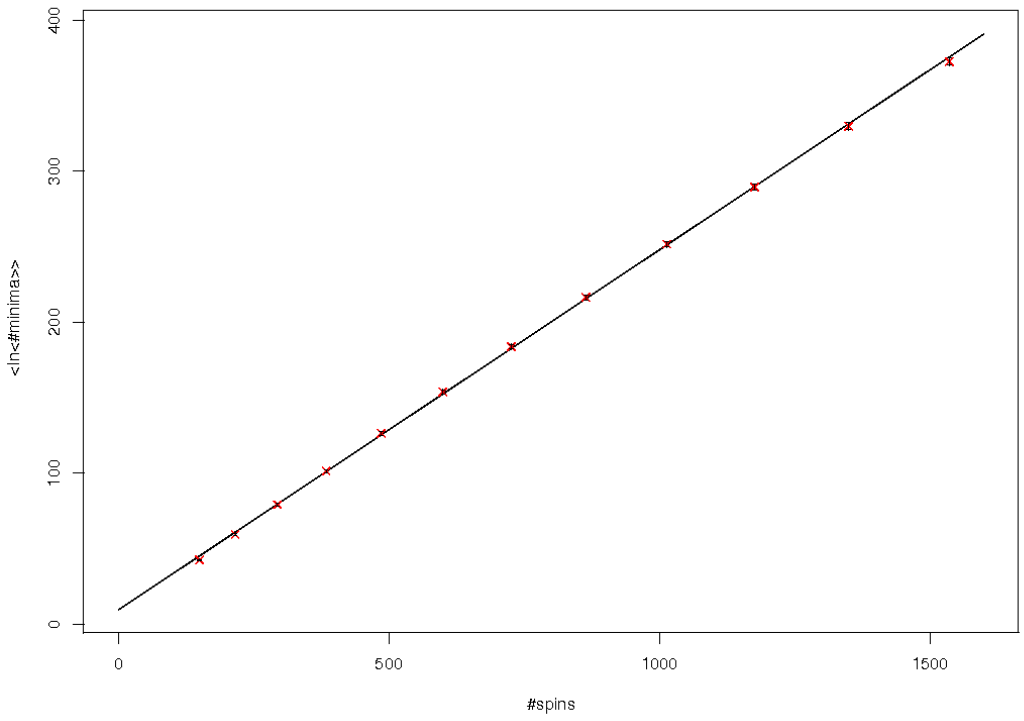

Figure 3.5: α estimation for spin glass lattice using an ensemble of 1000 lattices and 10000 descents per system

Chapter 4

Improvements and Analysis of the Sampling Method

4.1 Improved approximation

Knuth's method provides a simple way to calculate the search tree leaf size and it works surprisingly well, but obviously the method is quite rough and the estimates will be far from the true value if the tree happens to be unbalanced. In the original paper two simple methods to improve the reliability were introduced [9].

Instead of flipping a fair coin at each branching point, we can use our intuition about the search trees and perform a biased coin toss. For example if we assume that 70% of the leaves are in the left subtree, we may choose to pick the left branch with probability $p = 0.7$ and right branch with probability $1 - p = 0.3$. For each branching point we are allowed to choose different branching probabilities and if we managed to choose the probabilities perfectly we would be able to achieve an estimate with zero variance.

Theorem 5. *Consider a random descent from the root to a leaf p at depth h in a search tree. Denote the probability of choosing a branch at depth i by p_i. Then $(\prod_{i=0}^{h-1} p_i)^{-1}$ is an unbiased estimate for the leaf size of the search tree.*

Proof. Probability of ending up in a particular leaf will be $p = \prod_{i=0}^{j-1} p_i$ and the corresponding estimate will be $1/p$ so all leaves contribute one to the expected value and summing over all the leaves yields the claimed result. □

Note that Theorem 4 is a special case of Theorem 5. Next, we shall check whether using a biased coin improves the estimates when considering the search trees whose paths from the root to leaves correspond to the local minima of a hexagonal spin glass lattice. Clearly the search tree is not completely balanced because the left branch is often larger than the right branch. This is the case because choosing a green edge does not force as many other edge colors as a red choice potentially would. But the interesting question is, how unbalanced the search trees actually are?

We performed a small test for ferromagnetic hexagonal lattices with one, two and three hexagons. We chose such small systems, because it is possible to generate the whole search tree and exactly calculate its leaf size. If we know the real leaf size, we

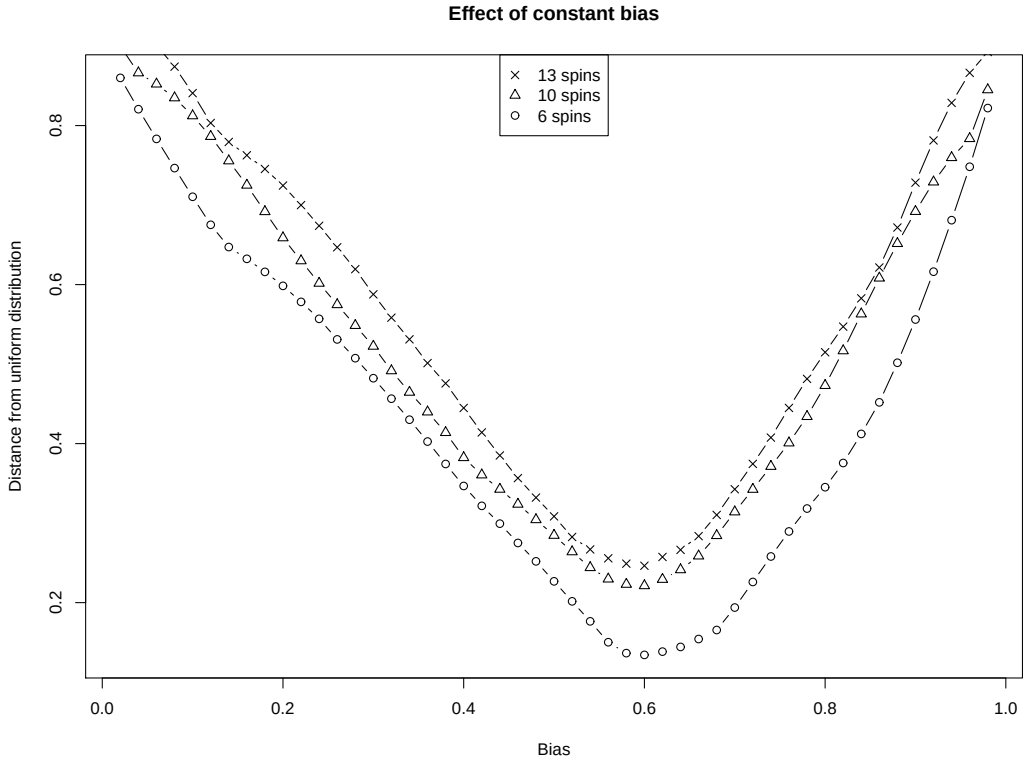

Figure 4.1: The distance from the uniform distribution as a function of bias

will be able to calculate how far the distribution of samples we get is from the uniform distribution. We use the following *total variation distance* [15] as a distance measure between two distributions π and ρ,

$$d_v(\pi, \rho) = \frac{1}{2} \sum_i |\pi_i - \rho_i|. \tag{4.1}$$

To check how balanced the tree is, we select a constant bias p and calculate the distance between the resulting samples and the uniform distribution. The results can be seen in Figure 4.1. Clearly, the values around $p = 0.6$ seem to produce the best samples, which indicates that the search tree is unbalanced approximately in such a way that 60% of the leaves are in the left subtree. This is a very rough analysis but it gives us a good overview on the search trees and it is a proof that the search tree is significantly unbalanced.

Another way to get better estimates is to readjust the tree in some way to decrease the variance. One way to do this is to compress the tree to decrease its height. This can often be done by considering the structure of the original problem and the tree generation process itself. For example, instead of performing binary choices whether to color an edge red or green, we can consider coloring entire hexagons at once. The number of possible hexagon colorings depends on the earlier choices but it is relatively simple to perform a case analysis and then pick one of the feasible colorings at random. This increases the branching factor of the tree and the estimate will converge towards the real value faster. However, it will be harder to retrieve other useful information

such as the energy of the minima from the process and it is a bit more difficult to analyze the tree structure or variance as well.

As a conclusion, the results are slightly better by using the improved techniques, but the drawback is the difficult analysis and more complex algorithms. The best option depends on the resources available and the accuracy criteria. Figure 4.2 and Figure 4.3 demonstrate the behavior of the estimate as a function of iterations (descents) for ferromagnetic lattices of 271 and 631 hexagons. It can be seen that the estimates seem to settle down quite quickly, and converge towards the same value. The introduction of bias ($p = 0.6$) seems to help a bit, because the convergence seems faster. The tree modification does not seem to help as much.

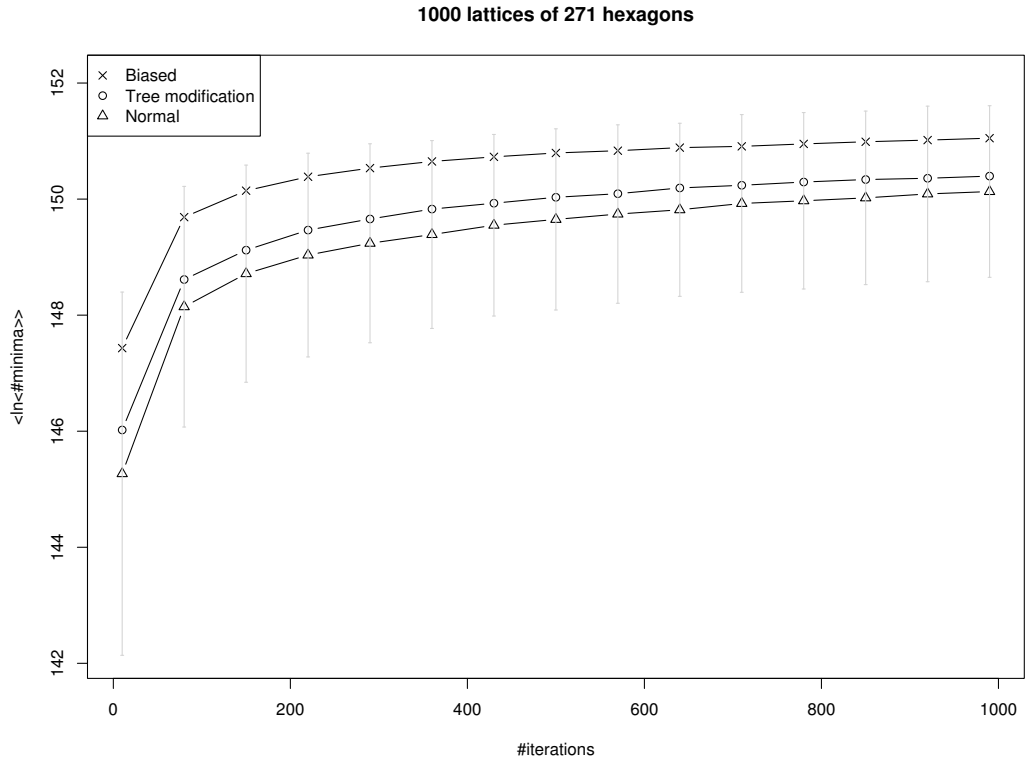

Figure 4.2: The leaf size estimate as a function of descents using a lattice of 271 hexagons

A quite straightforward way to reduce the variance is to simply increase the number of descents. As we noticed from Figures 4.2 and 4.3, we get results of the correct order of magnitude even with a small number of iterations, but what happens if we perform longer runs?

Consider an ensemble of n lattices and let X_i be the leaf size estimate of lattice i. The leaf size estimate of the system can be calculated by averaging over the n lattices

$$X = \frac{1}{n} \sum_{i=1}^{n} X_n. \tag{4.2}$$

However, this estimate is based on just a single descent per lattice. We can repeat the process to get more accurate results. Consider random variables Y_i where $i = \{1, ..., I\}$

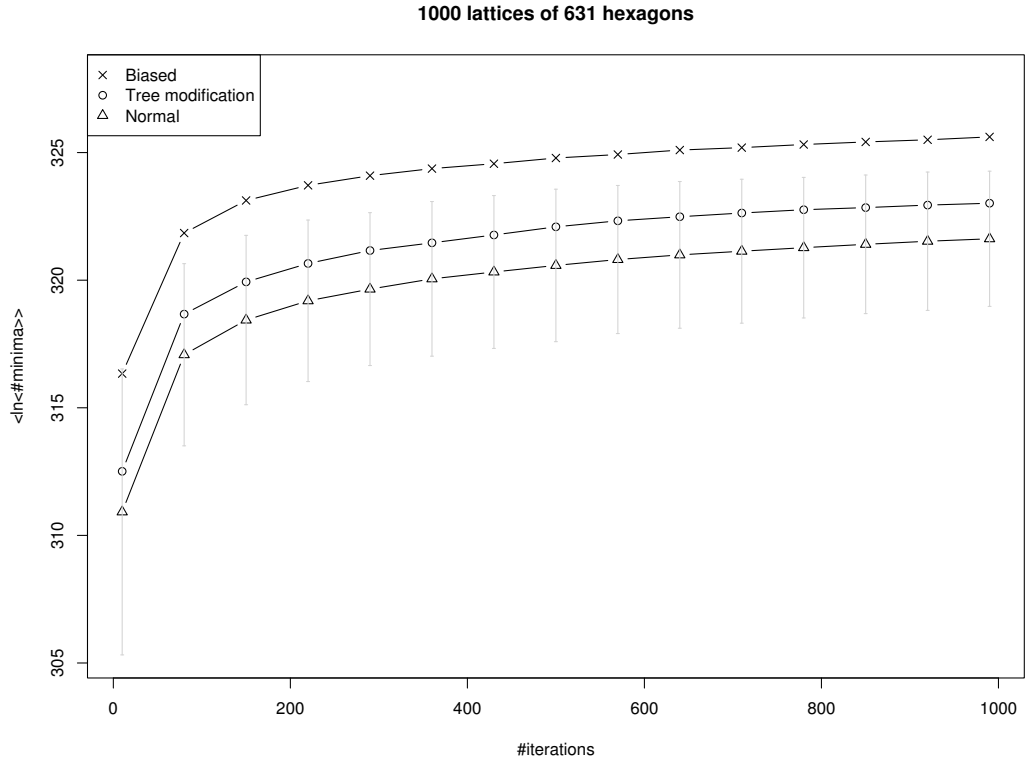

Figure 4.3: The leaf size estimate as a function of descents using a lattice of 631 hexagons

and $Y_i \sim X$. The improved estimate would then be

$$Y = \frac{1}{I}\sum_{i=1}^{I} Y_i. \tag{4.3}$$

Note that the sum of a large number of random variables from the same distribution is normally distributed. Also, we know that the random variables Y_i have the same variance σ^2, they are uncorrelated and $Var(aX) = a^2 Var(X)$. Hence, the variance of Y is

$$Var(Y) = Var(\frac{1}{I}\sum_{i=1}^{I} Y_i) = \frac{1}{I^2}\sum_{i=1}^{I} Var(Y_i) = \frac{1}{I^2} I\sigma^2 = \frac{\sigma^2}{I}. \tag{4.4}$$

So, by performing I descents we can reduce the standard deviation to $\sigma/\sqrt{I}$ and basically we can calculate I based on how accurate results we want. Also, the size of the ensemble and the number of iterations are closely related: Taking a large ensemble will yield more accurate results just like increasing the number of iterations would. The best performance is achieved by taking a large ensemble and using many iterations.

4.2 Reliability of the results

We have introduced a method for estimating the search tree size and proposed a couple of improvements, but it is interesting to know how good the method actually is. We analyze the variance of the most simple version of the algorithm. The improved version will have a slightly higher reliability, but the analysis would be much harder.

We will propose a formula, which is slightly more intuitive than the bound given by Knuth [9], for the variance of the search tree size estimate and prove it, but we will need the following lemma during the proof so it will be presented first.

Lemma 1. *Let T be an arbitrary binary search tree where each node that is not a leaf has exactly two children (subtrees L and R). Denote the leaf size estimate of a tree T by S_T. Then*

$$ES_T^2 = 2(ES_L^2 + ES_R^2). \tag{4.5}$$

Proof. Assume that with probability p_i the leaf size estimate of the left subtree is 2^i and let the height of the subtree be m. Then for the expected value of the square of the leaf size estimate of the left subtree it holds that

$$ES_L^2 = p_0(2^0)^2 + ... + p_m(2^m)^2.$$

Similarly for the right subtree (probability for estimate 2^i being q_i, and let the height of the tree be n)

$$ES_R^2 = q_0(2^0)^2 + ... + q_n(2^n)^2.$$

The expected value of the square of the leaf size of the whole tree can then be calculated as follows

$$\begin{aligned} ES_T^2 &= \frac{p_0}{2}(2 \cdot 2^0)^2 + ... + \frac{p_m}{2}(2 \cdot 2^m)^2 + \frac{q_0}{2}(2 \cdot 2^0)^2 + ... + \frac{q_n}{2}(2 \cdot 2^n)^2 \\ &= 2(p_0(2^0)^2 + ... + p_m(2^m)^2) + 2(q_0(2^0)^2 + ... + q_n(2^n)^2). \end{aligned}$$

By applying the equations for ES_L^2 and ES_R^2 we conclude the proof

$$ES_T^2 = 2(ES_L^2 + ES_R^2).$$

□

Theorem 6. *Let T be an arbitrary binary search tree where each node i that is not a leaf has exactly two children (subtrees $L^{(i)}$ and $R^{(i)}$). Let the vertex set of the tree be $V(T)$ and denote the depth of node i by d_i. Then it holds that:*

$$Var(S_T) = \sum_{i \in V(T)} 2^{d_i}(S_{L^{(i)}} - S_{R^{(i)}})^2. \tag{4.6}$$

Proof. For an unbiased estimate for the leaf size, it holds that

$$ES_T = ES_L + ES_R. \tag{4.7}$$

Then it immediately follows that

$$(ES_T)^2 = (ES_L + ES_R)^2 = (ES_L - ES_R)^2 + 4ES_L ES_R$$

and based on the previous equation we get

$$Var(S_T) = ES_T^2 - (ES_T)^2 = ES_T^2 - 2(ES_T)^2 + 4ES_L ES_R + (ES_L - ES_R)^2.$$

We can rewrite the equation by expanding the term $2(ES_T)^2$ using equation (4.7)

$$Var(S_T) = ES_T^2 - 2(ES_L)^2 - 2(ES_R)^2 + (ES_L - ES_R)^2.$$

By applying Lemma 1 we arrive at

$$Var(S_T) = 2(ES_L^2 + ES_R^2) - 2(ES_L)^2 - 2(ES_R)^2 + (ES_L - ES_R)^2$$

and we can write this in a different form as

$$Var(S_T) = 2(Var(S_L) + Var(S_R)) + (ES_L - ES_R)^2. \tag{4.8}$$

The derivation of equation (4.8) was required as we will use it to prove the original claim by induction.

If tree T is a single node then it has no subtrees and the estimated leaf size will always be 1. Our formula (6) gives $Var(S_T) = 2^0 \cdot (0 - 0) = 0$ because the subtrees do not exist.

If tree T is not a single node then it has two subtrees L and R (and each subtree that is not a leaf has two subtrees and so on). We assume that the variance for the leaf size estimate of L and R can be calculated as follows

$$Var(S_L) = \sum_{i \in V(L)} 2^{d_i} (ES_{L^{(i)}} - ES_{R^{(i)}})^2.$$

$$Var(S_R) = \sum_{i \in V(R)} 2^{d_i} (ES_{L^{(i)}} - ES_{R^{(i)}})^2.$$

We can apply equation (4.8) to get

$$Var(S_T) = 2(Var(S_L) + Var(S_R)) + (ES_L - ES_R)^2 =$$

$$\sum_{i \in V(L)} 2^{d_i+1} (ES_{L^{(i)}} - ES_{R^{(i)}})^2 + \sum_{i \in V(R)} 2^{d_i+1} (ES_{L^{(i)}} - ES_{R^{(i)}})^2 + (ES_L + ES_R)^2.$$

We know that $V(L) \cap V(R) = \emptyset$ and the depth of node i increases by one as the new root is added from which we can obtain

$$Var(S_T) = \sum_{i \in V(L) \cup V(R)} 2^{d_i} (ES_{L^{(i)}} - ES_{R^{(i)}})^2 + (ES_L + ES_R)^2.$$

Finally, by noticing that the depth of the root node is 0, we can simplify the equation to conclude the proof

$$Var(S_T) = \sum_{i \in V(T)} 2^{d_i} (ES_{L^{(i)}} - ES_{R^{(i)}})^2.$$

And specially, if the search tree estimates are unbiased we get

$$Var(S_T) = \sum_{i \in V(T)} 2^{d_i} (S_{L^{(i)}} - S_{R^{(i)}})^2.$$

□

As an immediate consequence of Theorem 6 we can see that any difference in the subtree leaf sizes will result in an exponential variance. Also, the depth of the branching point where the subtree sizes differ affects the variance greatly. This sounds bad at first, because even a small difference in the subtree sizes affects the variance by a lot if the branch is located deep in the tree. However, one needs to remember that the estimates are also of exponential size and that is why we can manage with the variance.

Chapter 5

Exact Calculations

Some exact calculations were performed in order to be able to say something more concrete on the reliability of the search tree size estimates. The number of local minima for a ferromagnetic square lattice and Bethe lattices are solvable with the help of simple recurrence equations. The real values are then compared with the estimates.

5.1 Ferromagnetic square lattice

It is relatively easy to calculate the number of proper local minima of a ferromagnetic square lattice with or without periodic boundary conditions. By proper local minima we mean the spin states that do not have any neighboring states with equal or lower energy. A square lattice is demonstrated in Figure 5.1.

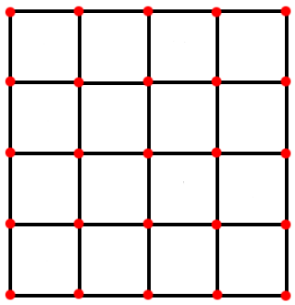

Figure 5.1: Square lattice

In terms of coloring, the local minima of a ferromagnetic square lattice look like red "ladders" through the lattice. If one edge of a square is red, the edges next to it must be green, because otherwise the state would not be a proper local minimum, and thus the edge opposite to it is forced to be red as well, due to validity of coloring. The process continues like this and forces all of the edges of a column or row to be red if at least one of them is red. The ladders can be either horizontal or vertical, but not both because the ladders may not cross. If there are no periodic boundary conditions, the edges next to the lattice corners may not be red.

Let x_n represent the number of arrangements of horizontal ladders through an $n \times n$ square lattice. Then we obtain the recursion formula

$$x_n = x_{n-1} + x_{n-2} \tag{5.1}$$

where $x_1 = 1$ and $x_2 = 1$ (only the colorings with no ladders are possible). The idea behind the equation is the following: The second to last row either has a ladder or not. If there is a ladder, the number of ladder combinations remaining is x_{n-2} because there cannot be a ladder in the next row. If there is no ladder in the second to last row, there are no restrictions for the row next to it and thus x_{n-1} possibilities remain.

Recurrence (5.1) gives rise to the Fibonacci numbers. There exists a closed form solution

$$x_n = \frac{\phi^n - (1-\phi)^n}{\sqrt{5}} \tag{5.2}$$

where $\phi = \frac{1+\sqrt{5}}{2}$ [8]. As discussed before, x_n represents the number of possibilities for inserting horizontal ladders through an $n \times n$ lattice. The ladders may also be vertical, but if there are no ladders at all (ground state) we calculate this twice so it must be taken into account in the final formula for the number of proper local minima for a ferromagnetic square lattice of size $n \times n$:

$$y_n = 2(2x_n - 1) = 4x_n - 2 \tag{5.3}$$

The up-down symmetry of spins is also taken into account in the previous equation. It is interesting that the number of proper local minima is not exponential as a function of the number of spins but instead $\mathcal{O}(e^{\ln \phi \sqrt{N}})$.

5.2 Bethe lattice

Introduced by Hans Bethe in 1935, the Bethe lattice is a connected cycle-free graph where each node that is not a leaf has d neighbors [5]. Due to the absence of cycles it is possible to exactly solve the number of local minima of a Bethe lattice of given d. Let $d = 3$ in this example case. The lattice is demonstrated in Figure 5.2.

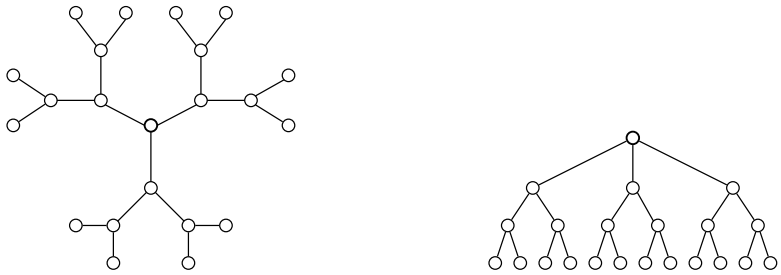

Figure 5.2: Two ways to represent the Bethe lattice with $d = 3$

A local minimum corresponds to an edge coloring where the central nodes have at most one adjacent red edge and the only edges adjacent to the leaf nodes are green.

The number of such colorings can be calculated recursively. The number of local minima a_n for a binary tree of depth n can be calculated by formula

$$a_n = a_{n-1}^2 + 2a_{n-1}a_{n-2}^2 \tag{5.4}$$

where $a_0 = 1$ and $a_1 = 1$. The idea of the recursion is the following: A node might have no descending red edges in which case the children are independent of each other and have no restrictions at all. The other case is that one of the two descending edges is red. In this case the red child must have two descending green edges but the green child does not have any restrictions. To finish up the calculations we must use the formula

$$b_n = a_{n-1}^3 + 3a_{n-1}^2a_{n-2}^2 \tag{5.5}$$

to obtain the number of local minima starting from the root node because the root node has three children. The equation is derived in a similar fashion as (5.4). In the end, one needs to remember to take the up-down symmetry of the spins into account, thus the total number of local minima for a Bethe lattice of radius i is $2b_i$. The idea behind the recurrence relations is demonstrated in Figure 5.3.

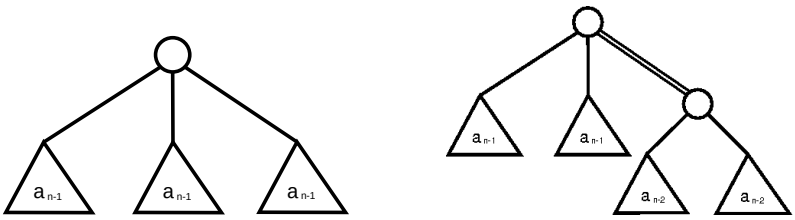

Figure 5.3: Idea behind the recurrence relations.

Bethe lattices are interesting for two reasons: They have similar local properties as the hexagonal lattice because the degree distribution of the vertices is quite similar and thus the differences in the results are likely to be caused by the existence of cycles in the hexagonal lattice. Also, it is interesting to be able to test the accuracy of the search tree estimates in a case where most of the choices are almost independent of each other. There are no global dependencies because there are no cycles. If the colorings would have no dependencies at all, the size estimates of the search tree would follow a log-normal distribution.

Assume that the coloring of an edge i is independent of the other choices. Let $d_i \in \{0, 1\}$ be a random variable which takes value 0 if there is just one possible color and value 1 if there are two possible colors for the edge. The sum

$$D = \sum_{i=1}^{M} d_i \tag{5.6}$$

of independent binary random variables is normally distributed and it is actually the logarithm of the size estimate. Because the logarithm is normally distributed, the size estimate itself follows the log-normal distribution.

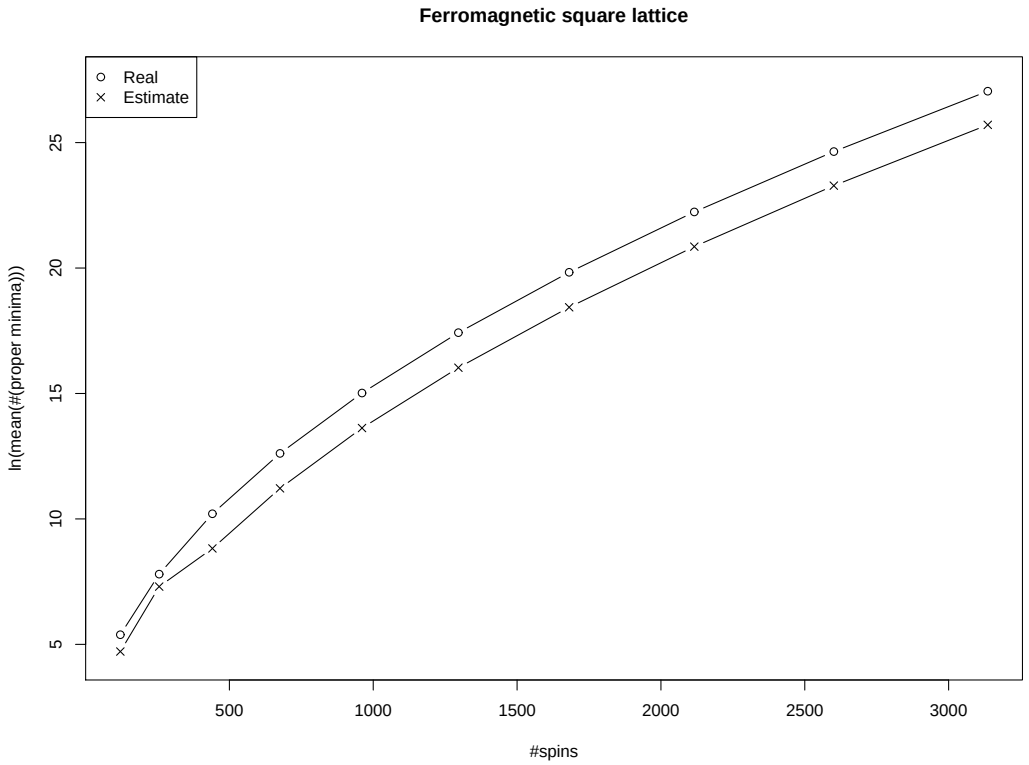

Figure 5.4: Ferromagnetic square lattice

5.3 Conclusions

The ferromagnetic square lattice is a hard case for the estimation algorithm because the search tree is very biased. By choosing a green branch no restrictions are introduced but by choosing a red branch a lot of edge colors are forced immediately. As demonstrated in Figure 5.4 the results seem very accurate up to about $e^5 \approx 144$ spins and after that the estimates tend to be below the actual value.

The Bethe lattice is a rather easy case for the algorithm because the search tree is quite balanced due to almost no dependencies in the edge colorings. And in this case the algorithm works very well up to around 1000 spins as can be seen from Figure 5.5.

The estimates are not perfect but they are quite accurate given the size of the trees. In both cases the estimates tend to be underestimates. This is caused by the fact that the original algorithm assumes no bias in the tree and gives accurate results for completely balanced trees. In our case the trees are certainly biased because the green branch is almost always larger than the red branch. If we introduce a bias to each branching point we will be able to get estimates closer to the real value but if we increase the bias too much we will get overestimates for the tree size. It is not very simple to come up with a perfect bias for each branching point but good guesses can be made to improve the results if needed.

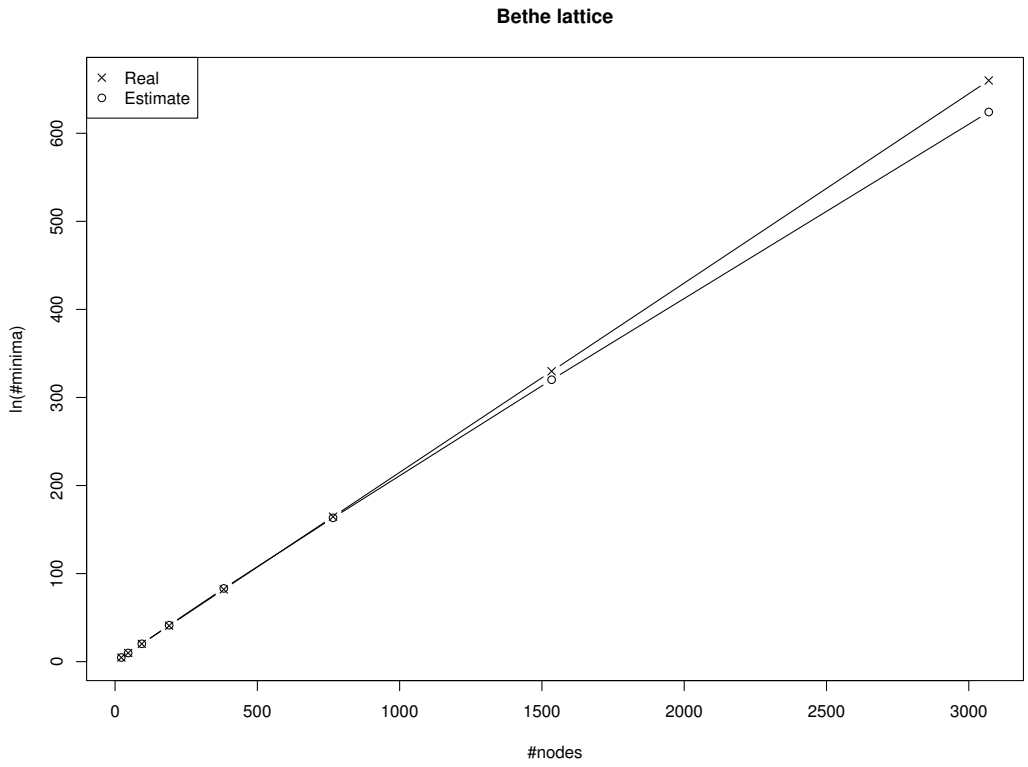

Figure 5.5: Bethe lattice with $d = 3$

Chapter 6

Almost Uniform Sampling of Metastable States

The method for estimating the search tree leaf size can also be used for sampling. A trivial way to perform sampling would be to simply make random descents in the search tree. However, the goal is to achieve a uniform sampling method which means performing descents in such a way that each of the leaves is reached with equal probability. Such a method would allow us for example to gather interesting information on the energy distribution of the spin configurations. Uniform distribution is not achieved by random descents and that is why we present an alternative approach based on the search tree size estimation idea.

6.1 Subtree size approximation

Our approach is to use the search tree leaf size estimates recursively. At each branching point we measure a size estimate for both subtrees and choose a branch with biased probability. If the left subtree estimate is S_L and the right subtree estimate is S_R then the probability of choosing the left branch will be $p_l = S_L/(S_L + S_R)$ and the probability of choosing the right branch will be $p_r = 1 - p_l = S_R/(S_L + S_R)$.

The computational cost of this sampling method is quite high. If the search tree height is n, then in the worst case we encounter n branches during a descent and we must perform a total number of $2n$ subtree leaf size estimations. Each estimate can be done in linear time, but if we use i iterations for each estimate, the total time complexity of the recursive algorithm is $\mathcal{O}(in^2)$.

Clearly the algorithm should produce the uniform distribution if i is large enough, because then the subtree estimates will converge towards the real values and the choice of correct probabilities p_l and p_r at each branching point will result in zero variance as discussed in the previous chapter. But using large enough i will cost a lot of time so we need to consider the trade-off between time and accuracy. Using any constant i will result in a slightly biased distribution, as will be shown in the following examples.

Example 2. Consider the tree in Figure 6.1. We start from the root and measure the subtree leaf sizes. Assuming that we use only a single descent to estimate the

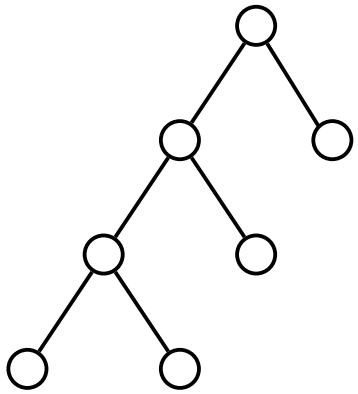

Figure 6.1: Linear tree of height 3

subtree sizes, for the left subtree we obtain size 2 with probability 0.5 and size 4 with probability 0.5. For the right subtree we always obtain the size 1. Thus, with probability 0.5 we will choose the left branch with probability $2/(2+1) = 2/3$ and with probability 0.5 we will choose it with probability $4/(4+1) = 4/5$. So the probability to choose the left branch will be $1/2 \cdot 2/3 + 1/2 \cdot 4/5 = 1/3 + 2/5 = 11/15$. We should choose the left branch with probability 3/4, because it contains 3 of the 4 leaves, but $11/15 \approx 0.7333 \neq 3/4$ so we do not obtain the uniform distribution.

Example 3. Assuming that we use two descents to estimate the subtree sizes, the calculations of the previous example would go differently: Left subtree size is 2 with probability 0.25, 3 with probability 0.5 and 4 with probability 0.25. The probability of choosing the left branch is $1/4 \cdot 2/3 + 1/2 \cdot 3/4 + 1/4 \cdot 4/5 = 1/6 + 3/8 + 1/5 = 20/120 + 45/120 + 24/120 = 89/120 \approx 0.7417$. Note that the probability is closer to the real value 0.75 but still not there. Increasing the number of iterations will improve the results, but a very large value for i is required to achieve the uniform distribution.

If the algorithm fails to find the uniform distribution even with such a simple tree, how will it manage with larger trees? Clearly, the worst case for the algorithm is a "linear tree" which has N leaves, height $N-1$ and the right subtree always has size 1. Considering the root and using a single descent for the subtree size estimation we obtain size 2 with probability 0.5, size 4 with probability 0.25 and so on. The probability of choosing the left branch from the root is

$$\frac{1}{2} \cdot \frac{2}{3} + \frac{1}{4} \cdot \frac{4}{5} + ... + \frac{1}{2^{N-2}} \cdot \frac{2^{N-2}}{2^{N-2}+1} + \frac{1}{2^{N-2}} \cdot \frac{2^{N-1}}{2^{N-1}+1}$$

$$= \frac{2}{2^{N-2}+1} + \sum_{i=1}^{N-3} \frac{1}{2^i+1}$$

This probability converges to 0.7645 as N grows, but the real probability $(N-1)/N$ converges to 1 as N grows. From this we can see that the maximum error in the probability using a single descent for subtree size estimation is 0.2355. If this big error is done multiple times, the sampling technique does not work very well. However, in the real situation the search tree is not linear and thus the algorithm will perform

better. Experiments also show that the quality of the distribution is not dramatically increased by using many descents to estimate the subtree leaf sizes.

6.2 Energy distribution results

To obtain the energy distribution we collect n samples by using the idea presented in the previous section. While we descend in the search tree, we calculate how many edges are colored red in each sample. The energy of the sample can then be simply calculated by equation $H = -M + 2H_R$ because we know H_R and $M = 9R^2 - 3R$ for a hexagonal lattice of radius R. We consider hexagonal lattices of sizes $5, 10$ and 15. The number of spins in these lattices are $150, 600$ and 1350 respectively. The energy per spin for each minimum is thus

$$\frac{H}{N} = \frac{-9R^2 + 3R + 2H_R}{6R^2} = -\frac{3}{2} + \frac{1}{2R} + \frac{H_R}{3R^2}. \tag{6.1}$$

It is quite obvious that the energy per spin of local minima must be in the interval $[-1.5, 0]$ and we divide this interval into 150 equidistant parts (bins). Then for each bin we calculate the frequency of minima and normalize it by dividing the frequency by the total number of descents. In this experiment we used 1000 descents per lattice and for the subtree leaf size approximation we used one descent per subtree. The resulting histogram for the spin glass lattices can be seen in Figure 6.2.

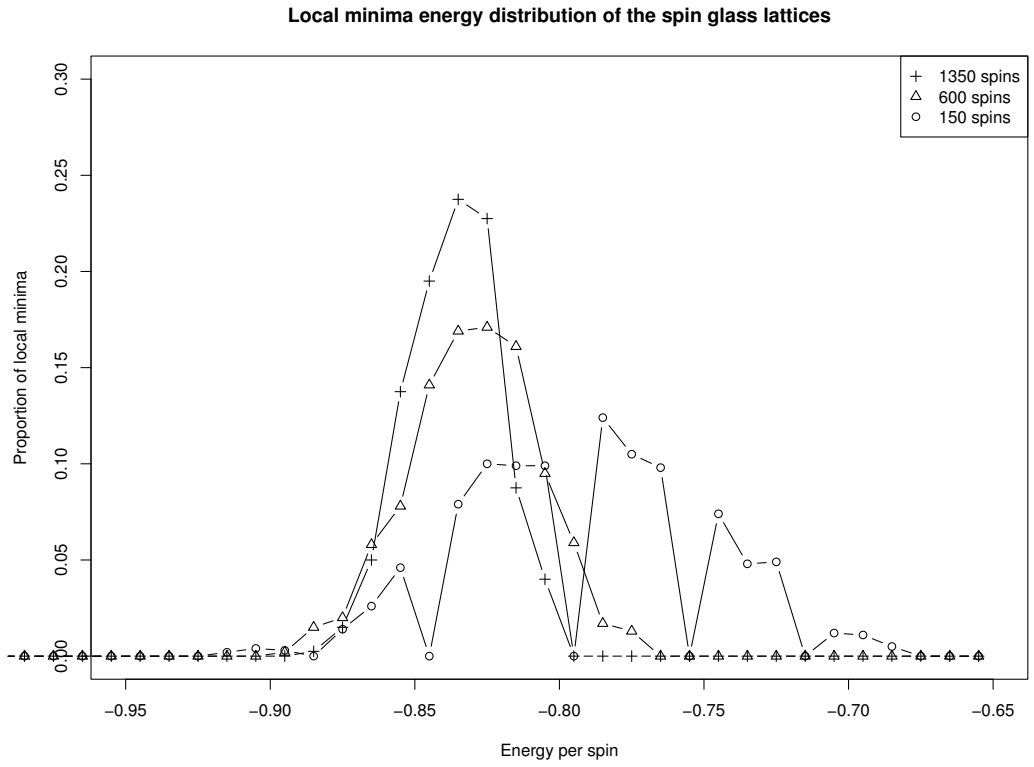

Figure 6.2: Local minima energy distribution of a spin glass lattice

The energy density (energy per spin value) of the local minima seems to concentrate on a smaller and smaller interval as the number of spins grows. If the lattice would be infinite, almost all the local minima should have the energy per spin value of around -0.84, very roughly. Note that the systems with 1350 spins and 600 spins form quite smooth bell-shaped curves while the system with 150 spins forms a curve which is not nearly as smooth. The reason for this is simply that there exist no local minima in the intervals $[-0.85, -0.84]$ or $[-0.80, -0.79]$ because the number of red edges is a discrete variable.

Similar histograms for the ferromagnetic lattices are shown in Figure 6.3. The asymptotical energy density for the ferromagnetic lattice would seem to be around -0.78 per spin.

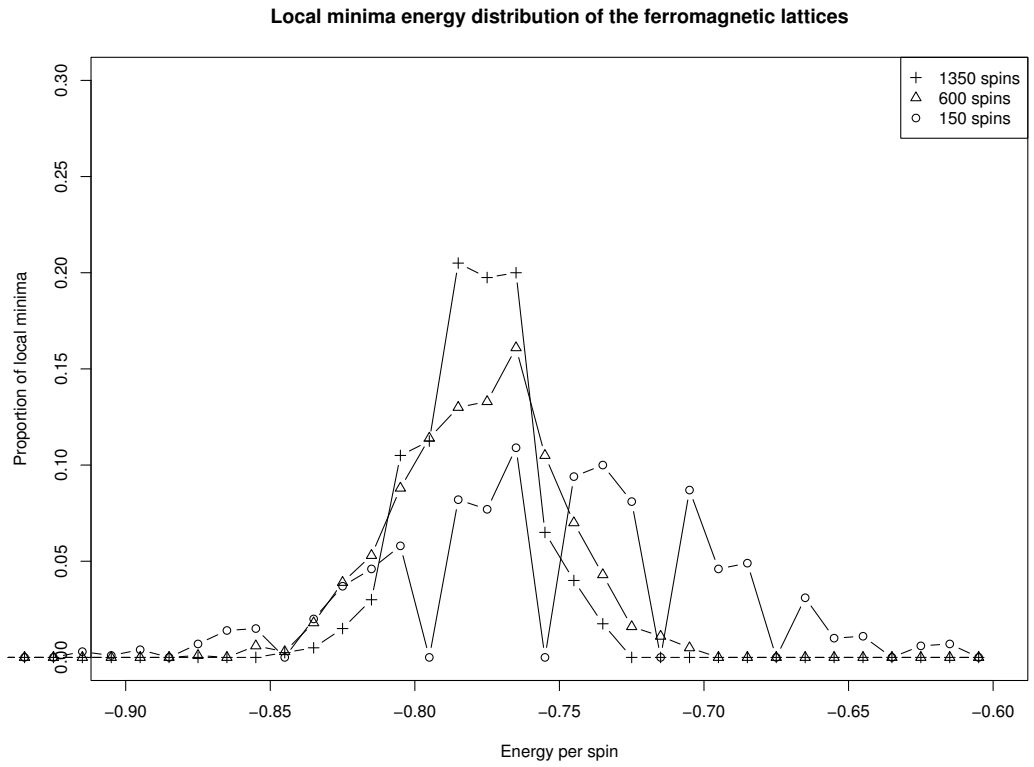

Figure 6.3: Local minima energy distribution of a ferromagnetic lattice

The asymptotical energy density of the hexagonal lattice was not very clearly visible in the previous figure. Therefore we consider another type of normalization: Instead of calculating the energy per spin values, we calculate the proportion of red edges. Now, we divide the interval $[0, 1]$ into 100 bins and plot a similar histogram as before using the different normalization. The results for the spin glass lattices can be seen in Figure 6.4 and for the ferromagnetic lattices in Figure 6.5. Now the concentration seems much more clear and we can quite confidently state that the proportion of the red edges is asymptotically in the interval $[0.21, 0.22]$ in the spin glass system and in the interval $[0.23, 0.24]$ in the ferromagnetic system. Also the smoothness of the curves is fixed if we perform this kind of normalization.

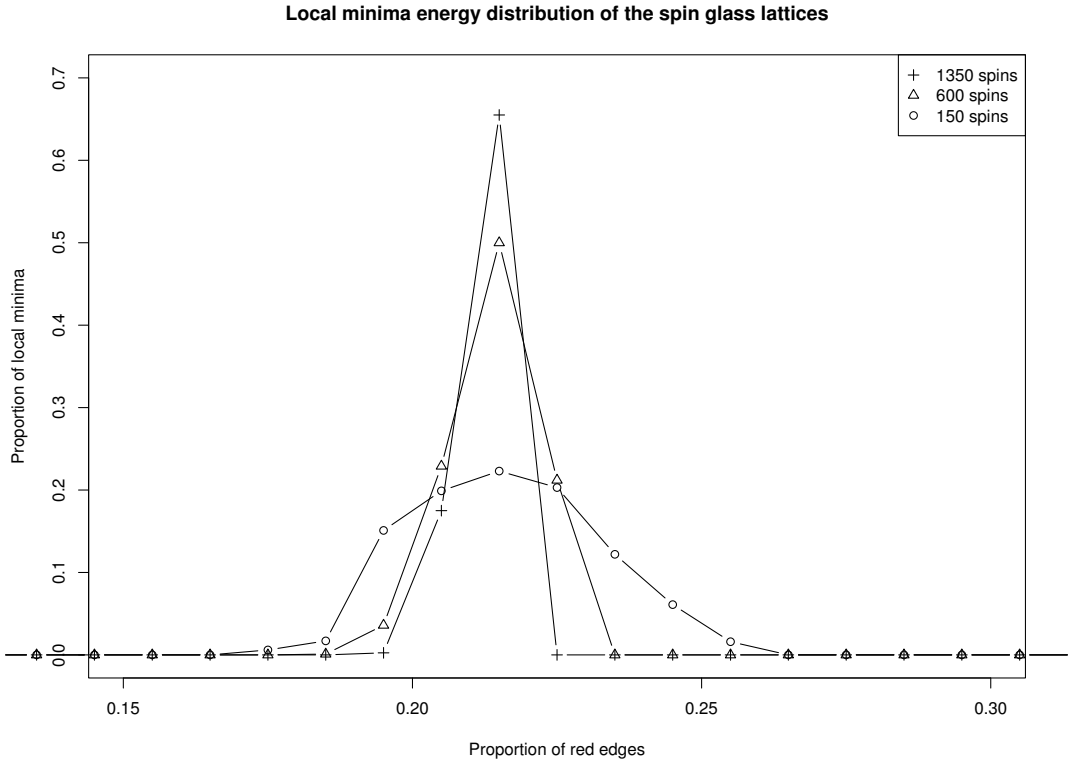

Figure 6.4: Concentration of the local minima of a spin glass lattice

6.3 Sampling of the full distribution

As can be seen from the results in previous section, most of the local minima have relatively high energy. Even if one would be able to achieve completely uniform sampling, the rare local minima would appear rarely if at all, unless exponentially many samples were collected. It is useful to be able to find the local minima at the low energy levels too, for example to understand the change of behavior at the phase transition point, which shall be explained more carefully in the next chapter. One approach to achieve this goal would be to favor green branches with a fixed probability because the low energy states are more likely to be the leaves of a green branch than a red one. In the end this sampling technique would produce a distribution where the minima with a high number of green edges are favored, but in the end this bias could be corrected, because we know how much bias was introduced in the search process.

If we end up in a leaf using a biased descent we can easily calculate what the probability of ending up in the leaf would have been without the bias because we know how many branches we encountered. Using this idea we are able to reach a larger portion of the minima but we still have the clue whether the minimum is one of the typical ones using different sampling or not. It would have been very nice to perform a couple of tests to see how well this idea works in practice, but this idea has not been implemented yet.

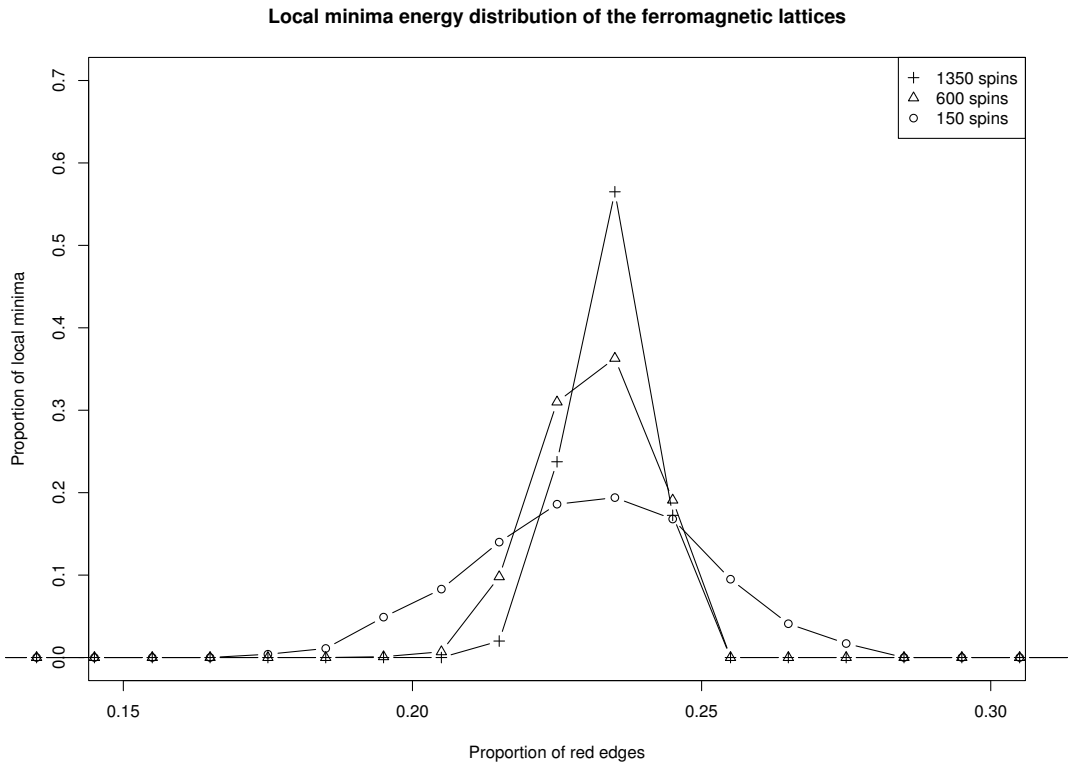

Figure 6.5: Concentration of the local minima of a ferromagnetic lattice

Chapter 7

Spin Glass - Ferromagnet Phase Transition

Another application where our search tree idea turns out to be useful is the study of the spin glass - ferromagnet phase transition. The phase transition occurs when the system has a certain critical proportion $p = p_c$ of ferromagnetic bonds [13].

Here we present a very rough example which probably makes the theory behind phase transition more concrete. In the ferromagnetic phase the lattice has only ferromagnetic bonds so all elementary cycles (the cycles consisting of a single hexagon, square or triangle) of the graph contain an even number of antiferromagnetic bonds. If all cycles contain an even number of antiferromagnetic bonds, the ground state of the system remains such that all bonds can be satisfied. Once we start to replace ferromagnetic bonds with antiferromagnetic ones, all bonds cannot be satisfied anymore because some of the cycles in the graph become frustrated.

In the beginning, most replacements change the parity of two cycles from even to odd, meaning that an odd number of the edges must be unsatisfied. In a hexagonal lattice of radius R there are $3R^2 - 3R + 1$ elementary cycles and $9R^2 - 3R$ edges (Theorem 3, page 13). With large R, the ratio of these is $1/3$. If we consider that each replacement increases the number of odd cycles by two and in the spin glass phase on average half of the cycles have odd parity, then we can estimate that we will need approximately proportion $1/12 \approx 0.0833$ of antiferromagnetic edges to achieve the spin glass phase. After this the changes will not have an effect anymore because we are already in the spin glass phase and on average, the total parity of cycles is not affected. This very rough calculation gets us quite close to the real critical point $1 - p_c \approx 0.0673$ which is estimated in articles [13] and [14].

7.1 Behavior of the number of metastable states

The phase transition does not have a major effect on the number of metastable states. With large lattices the number clearly seems to decrease, but the order of magnitude remains the same. From this we can conclude that the change of behavior is not caused by the number of metastable states although it is possible and likely that the number of metastable states with low energy differs greatly between these two phases. Figure 7.1 shows that the phase transition between spin glass and ferromagnet does not affect the number of metastable states greatly.

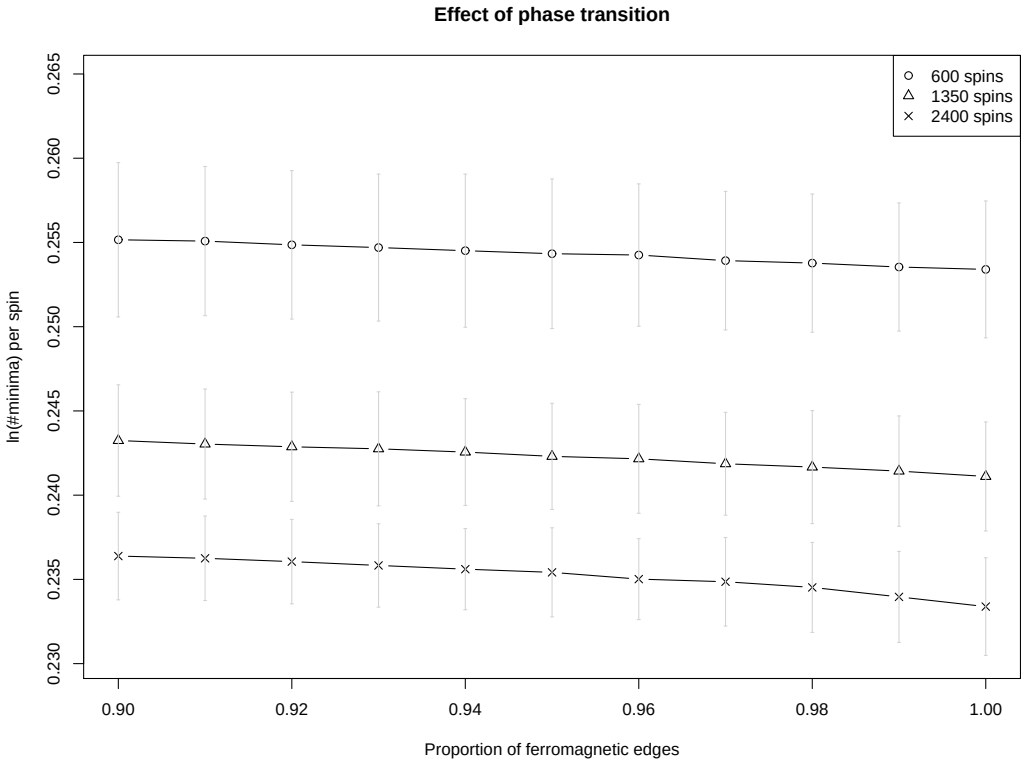

Figure 7.1: Ensembles of 1000 systems of different size

7.2 Exploration of very large lattices

In many articles the energy landscape of a spin glass system has been explored for lattices of quite limited size (such as 100 or maybe 1000 spins). Our method works reasonably well even with lattices up to 10000 or even 100000 spins. However, one problem with huge lattices is that the search tree size estimates will be exponentially large. As presented in the earlier chapters, one way to tackle this problem is to consider the logarithm of the estimate

$$\log_2(S_T) = \log_2(\prod_{i=0}^{n-1} d_i) = \sum_{i=0}^{n-1} \log(d_i) = \sum_{i=0}^{n-1} (d_i - 1)$$

which is very easy to calculate by simply taking the sum of the points where branching was allowed during a random descent. This technique allows us to produce scatter plots like Figure 7.2, and can be used for example to visualize that there is no sudden change of behavior in the number of local minima near the known phase transition point. Let p be the proportion of ferromagnetic bonds. In Figure 7.2 an ensemble of 1000 lattices with p-values $\{1, 0.99, 0.98, 0.96, 0.92, 0.5\}$ are used.

To ensure that a lattice with for example $p = 0.99$ really has more ferromagnetic bonds than a lattice with $p = 0.98$ we generate 1000 (size of the ensemble) random seeds and using each of these seeds we assign random numbers from uniform distribution $[0, 1]$ to each edge in the corresponding lattice. If p is greater than the value of the edge, then the edge is considered to be ferromagnetic. So with $p = 0$ all edges would be antiferromagnetic and by increasing p the number of ferromagnetic edges either stays the same or increases, but does not decrease. This construction is crucial and now the clouds (different p values) in Figure 7.2 can easily be separated.

If we assume that the branching factors are independent of each other, then $\log_2(S_T)$ is basically a sum of random binary variables. Since this follows normal distribution, then S_T actually follows a log-normal distribution. The mean for a log-normal distribution can be calculated with formula

$$ES_T = e^{\mu + \sigma^2/2}$$

where μ and σ^2 are the mean and variance of the underlying normal distribution [24]. Using this technique it is possible to gain rough estimates on the search tree sizes without the need to use large numbers at all. The problem is of course the assumption of independence of the binary variables which is certainly not the case. The more cycles the system has the more the branching factors depend on each other, but even for systems with small number of cycles or no cycles at all, the variables are not completely independent, because many unsatisfied edges are not allowed to be adjacent to each other.

The log-normality assumption seems to hold surprisingly well for the spin glass lattices, as can be seen from Figure 7.3. In a Q-Q plot theoretical quantiles from normal distribution are plotted against the quantiles of the sum of branching points, which is assumed to be normally distributed. If the result is a line, the two distributions are similar. For the ferromagnetic case, the log-normality assumption clearly

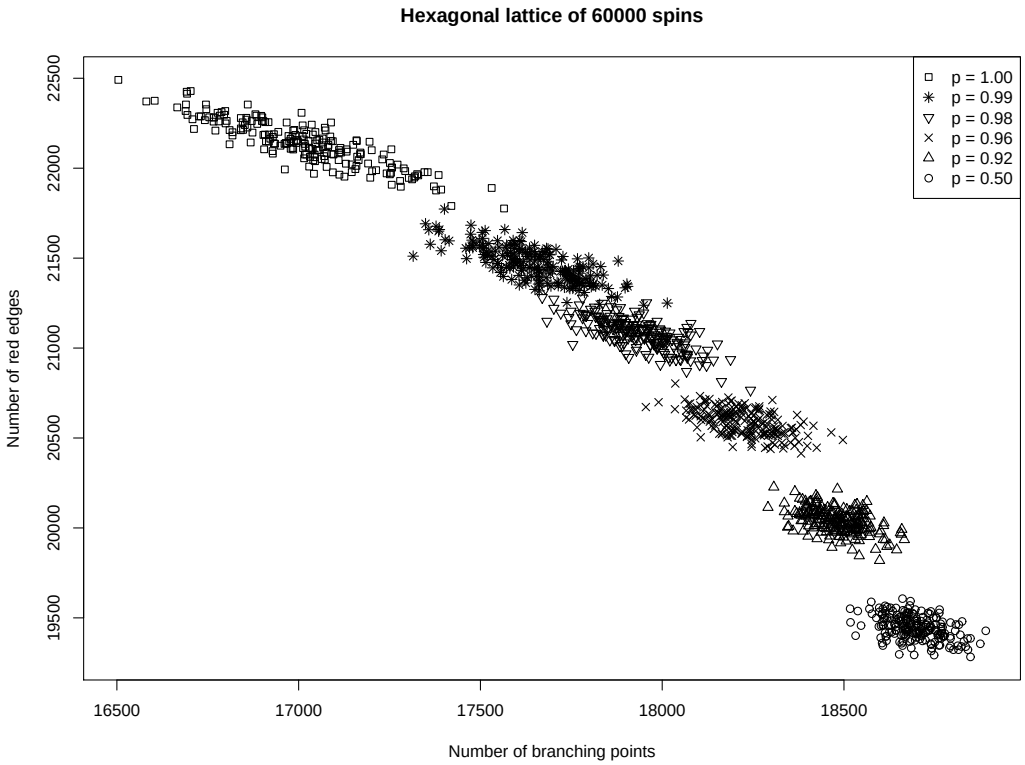

Figure 7.2: Scatter plot using systems of 60000 spins and different values of p

does not hold as can be seen from the Figure 7.4. For the spin glass case, the log-normality assumption gives a reasonable approximation, but should not be used if accurate results are required.

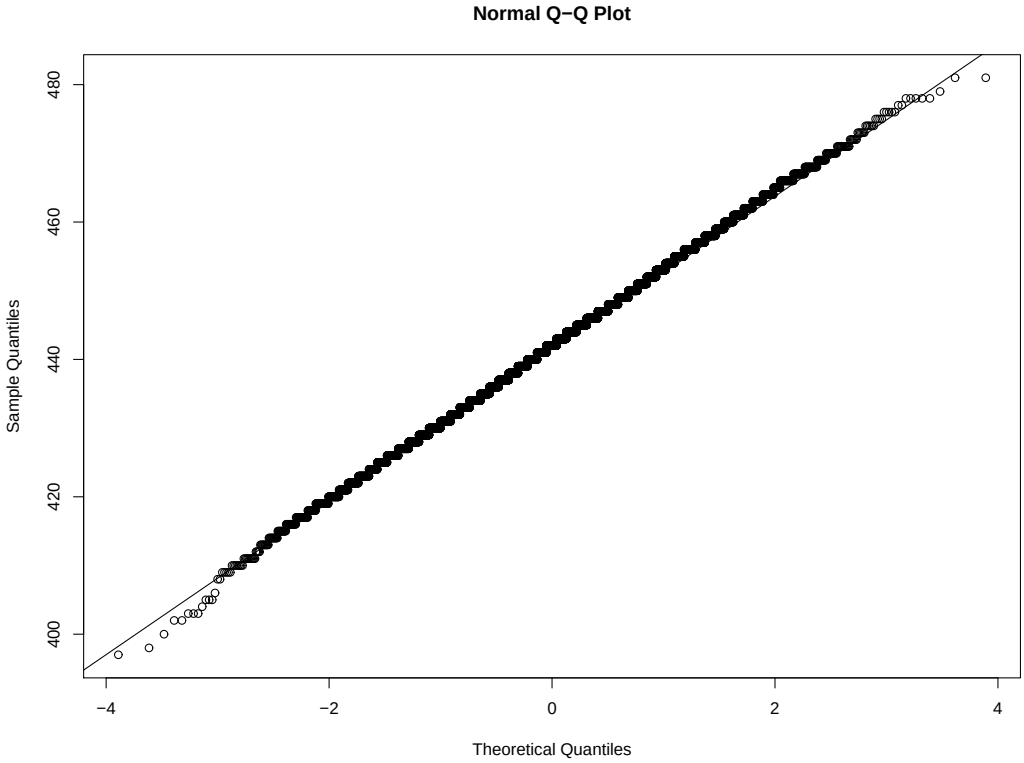

Figure 7.3: Q-Q plot of a spin glass-lattice of 1350 spins

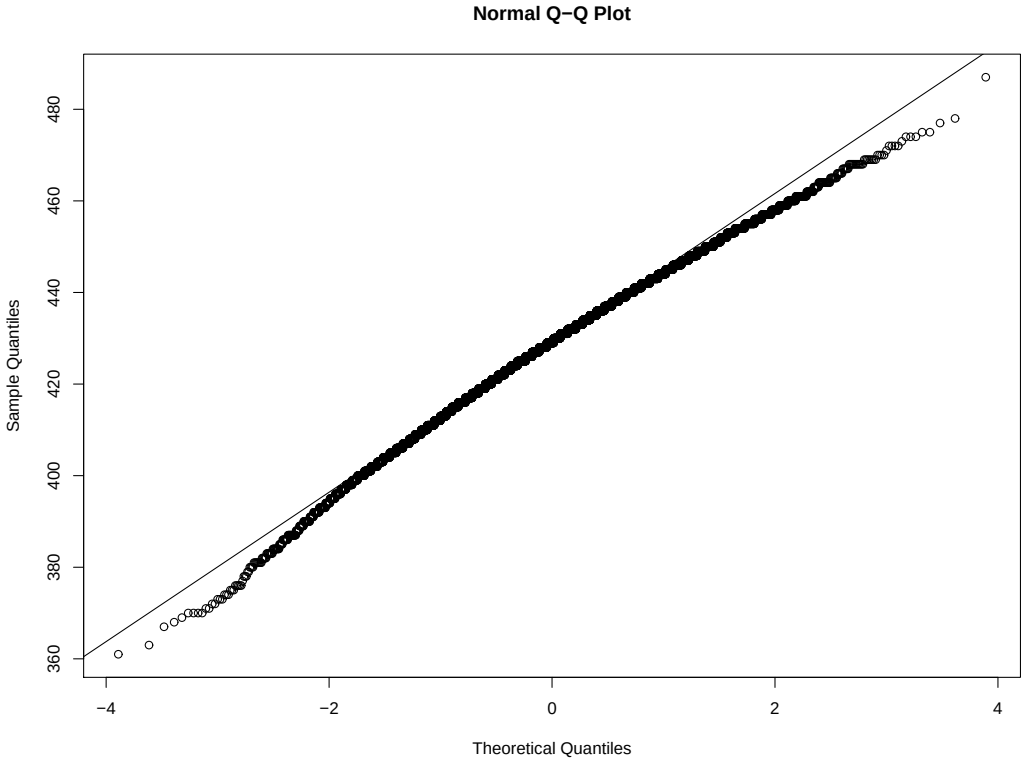

Figure 7.4: Q-Q plot of a ferromagnetic lattice of 1350 spins

Chapter 8

Related Work

In 1975 another fundamental spin glass model (SK model) was introduced by David Sherrington and Scott Kirkpatrick [20]. Their model has infinite-ranged random interactions with the advantage that the model is solvable in the general case. The Ising model has been solved in the one-dimensional case by Ernst Ising himself in 1925 and later on, the two-dimensional square lattice has been solved by Lars Onsager in 1942 [3]. The behavior of the two-dimensional spin glass system is significantly different from the one-dimensional case. The reason for the change is the number of paths between an arbitrary pair of spins.

Another influential paper which appeared in 1975 was written by Sam Edwards and Philip Warren Anderson [6]. This paper is the reason why the random bond models are sometimes called the EA models. Another well known spin glass model are the Potts model [25] and the Heisenberg model [4], where the spins are not restricted to pointing up or down, but can take any direction instead.

8.1 The number of metastable states

As stated before, the number of metastable states of a spin glass system is exponential as a function of the number of spins. Basically this means that equation

$$N_S = e^{\alpha N} \tag{8.1}$$

holds where N_S is the number of metastable states, N is the number of spins and α is a variable which characterizes the spin glass system.

Estimating values of α for different systems is an interesting task. In article [10] the values of α are calculated for honeycomb lattices with different bonds. This is done by considering the honeycomb lattice as a brickwork lattice of limited height and generating a set of equations which determine the possible edge configurations at the end of the lattice. The set of equations is generated by considering which edge configurations fit together in such a way that the result of the combination represents a local minimum. The largest eigenvalue of the matrix which comes from these equations can be estimated by iterating over equation $A^n x = \lambda^n x$ until convergence and the number of local minima is then calculated from the eigenvalue. This is a very nice technique, but the article presents results for lattices of heights up to only five,

probably due to the rapidly increasing size of the equation system. Thin lattices of infinite width do not necessarily have the same properties as normal infinite lattices which is where our method turns out to be useful.

There are several other articles which study the number of local minima for different kinds of systems. For example, in article [11] the values of α are calculated for Ising chains and strips with various different bonds and in article [23] the values of α are calculated for many different regular graphs. In article [22] the value of α is calculated for an infinite-ranged spin glass with Gaussian bond distribution. But to our knowledge, there exists no other result than the result of [10] for an infinite hexagonal lattice.

8.2 Spin glass energy landscape

In the article [17] the energy landscape of a square lattice with $\pm J$ interactions is explored by visiting local minima and maxima alternately by single spin flips. When searching for a minimum only the changes which decrease the total energy are allowed and vice versa when searching for a maximum. Additionally, a very small external magnetic field is introduced to get rid of the spin flips that do not change the energy at all. Once a minimum has been found the orientation of the field is changed several times until the energy converges into some value. If the reversing-field procedure is done long enough, the search will probably find its way from the energy plateau to a true local minimum. After the convergence into a minimum is noticed, the search continues on by looking for a maximum and so on.

The exploration of the landscape is used to store useful information about the minima and maxima, including the energy of the state. The energy parameter is normalized between 0 and 1 in such a way that in a zero-energy state all of the bonds are satisfied and if the energy of the system is 1, none of the bonds are satisfied. The ground state of a ferromagnetic lattice is the zero-energy state, but due to frustration, the spin glass ground state is strictly above zero. The energies of the visited states are then plotted as an energy distribution histogram which is roughly a bell-shaped curve whose peaks characterize the typical energy of a local minimum or maximum.

In the article, the curves are separated in two cases: even and odd. Odd curve means the case where an odd number of bonds are antiferromagnetic and even curve means the other case. This categorization is probably done because odd and even curves cannot have exactly the same energies because the number of unsatisfied edges is either odd or even depending on the bonds. If a system has an even number of unsatisfied edges to begin with, the property will not change because each spin has an even number of neighbors and thus by flipping a spin, the parity of unsatisfied bonds will not change.

The experiments are done with N $\times$ N lattices with periodic boundary conditions and with different N (8, 16, 32 and 64 were used). Both even and odd bond configurations were used and the histograms and their widths are calculated for each system size. From the result, the authors then try to extrapolate the peak location for a lattice of infinite size. Each histogram naturally has two peaks - one for the minima and one for the maxima. As one would assume these peaks are symmetrical and their

location from the average energy of 0.5 is equal.

The authors conclude that the width of the peak is $1/\sqrt{N}$ and thus the peak will become sharper and sharper as $n \to \infty$. Their extrapolation for the peak location gives value $\epsilon_\infty = 0.1633$. The authors used interesting techniques, especially the introduction of a very small external field seems like a good idea. Normally local search would be quite clueless while wandering around an energy plateau, but the alternating field will guide the search more quickly towards the true local minima and according to the article, the method seems to work reasonably well.

It is interesting to compare the results obtained by this kind of random walk with the results obtained by our sampling algorithm. Our method for the hexagonal spin glass lattice indicates that the peak location would be close to 0.215, which is naturally not the same as ϵ_∞ because the square lattice is a bit different from the hexagonal lattice. It would be very interesting to test how our algorithm would perform in the square lattice case.

8.3 Sampling techniques

Non-uniform sampling of the leaves of a search tree is very easy by just performing random descents. However, if we want a method which returns each of the leaves with equal probability the task is more difficult.

The most straightforward way to produce uniform sampling is to perform complete enumeration on the leaves. In article [19] a search tree with a hierarchical structure is traversed completely. However, the complete enumeration of the minima demands a lot of space and time. In the article the authors were able to perform the enumeration for square lattices of up to 27 squares, which means systems up to 39 spins. These systems are very small and although helical periodic boundary conditions are used to get rid of the border effects the applications are limited, because usually the interesting systems are larger.

Another approach which has been proposed in article [18] uses backtracking. The idea is to perform random descents in the tree, but in such a way that if we are likely to descend in a branch more often than we should, there is a certain probability that we will return back to the root instead of continuing. Although an improved version of this method was proposed in the article, backtracking does not seem like a good idea, because if the sampling process is performed on a very large search tree, the backtracking algorithm is very unlikely to be able to reach a leaf, resulting in a lot of iterations and a long running time, even if the samples produced are from the uniform distribution.

8.4 Location of the multicritical point

One application for our method is to study the properties of the lattice near the critical point where a phase transition occurs. Article [13] and a very recent article [14] present calculations for the spin glass - ferromagnet phase transition for triangular, hexagonal and square lattices. Starting from the ferromagnetic lattice and by turning ferromagnetic bonds into antiferromagnetic ones the behavior of the system changes

(phase transition occurs) when proportion p_c of the bonds are ferromagnetic. Despite the change in behavior the number of local minima does not seem to change significantly between these two phases which has been observed also in article [10]. Their article concludes that the number of minima actually slightly increases when going from spin glass phase to the ferromagnetic phase, but our results would suggest otherwise. Our explanation for the differing result is the limited height of lattices in their work.

It is true that for thin lattices, there exist more local minima in the ferromagnetic phase, but the behavior should change as the lattice height grows. This happens because just a single unsatisfied bond in a ferromagnetic lattice immediately implies a series of unsatisfied bonds in such a way that the series either forms a cycle or goes through the whole lattice. With respect to spin configuration this means that the local minima have quite large areas of similarly oriented spins. In a thin lattice this is not a huge restriction and thus does not affect the number of local minima very much which is why the number of local minima seems to be relatively high for the thin ferromagnetic lattices.

The location of the critical point depends on the lattice structure. Let p be the fraction of ferromagnetic bonds in the lattice. Some of the known locations are presented in Table 8.1. Both of the references [13] and [14] agree on the 3 first decimals.

lattice	p_c
square	0.891
hexagonal	0.933
triangular	0.836

Table 8.1: Location of the critical point for different lattices

8.5 External magnetic field

Typically the spin glass problem includes external magnetic field such that the energy can be calculated from formula

$$H(\sigma) = -\sum_{ij} J_{ij}\sigma_i\sigma_j + \sum_i \sigma_i h_i \tag{8.2}$$

where h_i is the contribution of the external field. For example with $h_i > 0$ the spin i tends to prefer the opposite direction $\sigma_i < 0$ to minimize the energy function. We are considering a system where $J_{ij} \in \{-1, 0, 1\}$.

Finding the ground state of a spin glass system with an external magnetic field is known to be an NP-complete problem, but even without the field the problem remains NP-complete in the general case where the system is not necessarily a regular lattice. However, counting the number of metastable states is an even more difficult problem and to keep things reasonably simple we will be considering mostly the case without the external magnetic field. If the magnetic field would exist, the problem of

estimating the number of local minima becomes harder, but it is possible to present some general statements about the hexagonal lattice case.

If $0 < |h_i| < 1$ for all i then the magnetic field will have a minor effect on the results, because only the spins in the lattice borders are affected because only those spins have even number of neighbors and those spins will start to prefer an orientation. With $1 < |h_i| < 3$ the magnetic field will begin to affect the whole lattice. The effect of interactions can still be stronger than the magnetic force, but none of the bonds may be unsatisfied if the orientation of the spin conflicts with the magnetic field. With $|h_i| > 3$ the magnetic field completely dominates the lattice and there is just one metastable state - the ground state.

Figure 8.1 demonstrates how a magnetic field may affect the system. In the general case the magnetic force may be arbitrarily strong.

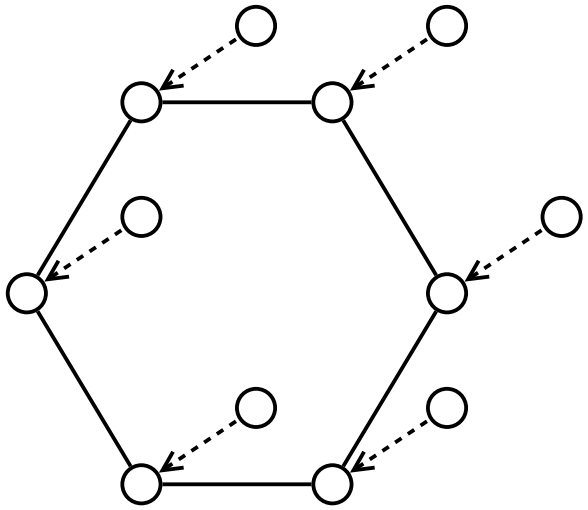

Figure 8.1: A spin glass in an external magnetic field.

Chapter 9

Conclusions

In this thesis, we have introduced an efficient algorithm for counting and sampling the energy minima of the hexagonal spin glass lattice, explained how the algorithm works and estimated the quality of the results. The search tree leaf size approximation method by Knuth seems a reasonable tool when one is interested in the logarithm of the leaf size estimates. However, the variance of the estimates is exponential, so a large number of iterations is required to obtain reliable results. And if we have to deal with very unbalanced trees, the method is clearly not very good.

The introduction of bias improved the estimates slightly. This leads to the following question: If constant bias improves the results quite a bit, would some kind of heuristic for calculating the biases improve the results even more? It could perhaps be an interesting research topic, because the search tree size approximation problem is very common in computer science.

To keep things simple, we considered only the hexagonal spin glass lattice, but the same method would work with the square lattice as well. The generalization for all kinds of systems would however require further thought. In principle the idea works, but in the general case, pruning is very hard to do without backtracking. This is essential, because we want to keep the algorithm as efficient as possible.

Another interesting thing is to consider whether the method could be generalized to other spin glass models, such as the Potts model, SK model, Heisenberg model and so on. This also does not seem very straightforward task, but would perhaps be doable.

Another approach for obtaining the local minima samples would be to consider a random spin configuration and perform greedy spin flips until we find a local minimum. This method cannot be used to calculate the number of local minima, but it can be used for sampling. It would be interesting to see whether the energy distribution from these samples would look similar to the distributions we obtain using the sampling algorithm proposed in this thesis.

Bibliography

[1] C. Ansótegui and F. Manyà. Mapping Problems with Finite-Domain Variables into Problems with Boolean Variables. In *The Seventh International Conference on Theory and Applications of Satisfiability Testing*, 2004.

[2] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi. *Combinatorial Optimization Problems and Their Approximability Properties*. Springer, 2003.

[3] R. Z. Bariev. Two-dimensional Ising model and local nonuniversality of critical exponents. *Brazilian Journal of Physics*, 30(4):680–681, 2000.

[4] R. J. Baxter. *Exactly Solved Models in Statistical Mechanics*. Academic Press, 1982.

[5] H. A. Bethe. Statistical theory of superlattices. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 150(871):552–575, 1935.

[6] S. F. Edwards and P. W. Anderson. Theory of spin glasses. *Journal of Physics F: Metal Physics*, 5(5):965–974, 1975.

[7] E. I. Hauge. Lars Onsager: The NTH student who became one of the greatest scientists of the 20th century. *Energy*, 30(6):787–793, 2005.

[8] S. K. Kivelä. *Reaalimuuttujan Analyysi*. Otatieto, 1992.

[9] D. E. Knuth. Estimating the efficiency of backtrack programs. *Mathematics of Computation*, 29:121–136, 1975.

[10] S. Masui, T. Li, B. W. Southern, and A. E. Jacobs. Metastable states of Ising models on the honeycomb lattice. *Physical Review B*, 40:7096–7100, 1989.

[11] S. Masui, B. W. Southern, and A. E. Jacobs. Metastable states of Ising spin glasses and random ferromagnets. *Physical Review B*, 39:6925–6933, 1989.

[12] M. E. J. Newman and G. T. Barkema. *Monte Carlo Methods in Statistical Physics*. Clarendon Press, Oxford, 1999.

[13] H. Nishimori and M. Ohzeki. Location of the Multicritical Point for the Ising Spin Glass on the Triangular and Hexagonal Lattices. *Journal of the Physical Society of Japan*, 75:034004, 2006.

[14] Masayuki Ohzeki. Precise locations of multicritical points for spin glasses on regular lattices. *arXiv:0811.0404*, 2008.

[15] P. Orponen. T-79.5204 Combinatorial Models and Stochastic Algorithms Lecture Notes. Retrieved 27.5.2009.

[16] C. H. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[17] E. S. Rodrigues and P. M. C. Oliveira. Spin-Glass Energy Landscape. *Journal of Statistical Physics*, 75(5-6), 1994.

[18] P. R. Rosenbaum. Sampling the Leaves of a Tree With Equal Probabilities. *Journal of the American Statistical Association*, 88(424):1455–1457, 1993.

[19] H. Seyed-Allaei, H. Seyed-Allaei, and M. R. Ejtehadi. Energy-landscape networks of spin glasses. *Physical Review E*, 77(3):031105, 2008.

[20] D. Sherrington and S. Kirkpatrick. Solvable model of a spin-glass. *Physical Review Letters*, 35:1792–1796, 1975.

[21] C. Simone, M. Diehl, M. Jünger, P. Muntzel, G. Reinelt, and G. Rinaldi. Exact ground states of Ising spin glasses: New experimental results with a branch-and-cut algorithm. *Journal of Statistical Physics*, 80(1):487–496, 1995.

[22] F. Tanaka and S. F. Edwards. Analytic theory of the ground state properties of a spin glass. I. Ising spin glass. *Journal of Physics F: Metal Physics*, 10:2769–2778, 1980.

[23] B. Waclaw and Z. Burda. Counting metastable states of Ising spin glasses on arbitrary graphs. *Physical Review E*, 77:041114, 2008.

[24] E. W. Weisstein. Log Normal Distribution. From MathWorld–A Wolfram Web Resource. Retrieved 26.5.2009.

[25] F. Y. Wu. The Potts model. *Reviews of Modern Physics*, 54(1):235–268, 1982.

[26] H. D. Young, R. A. Freedman, F. W. Sears, and M. W. Zemansky. *Sears and Zemansky's University Physics*, pages 976–977. Pearson Addison-Wesley, San Francisco (CA), 2008.