The State Space of Complex Systems

von der Fakultät für Naturwissenschaften genehmigte
Dissertation
zur Erlangung des akademischen Grades

doctor rerum naturalium

(Dr. rer. nat.)

vorgelegt von Dipl.-Phys. Frank Heilmann

geboren am 18. Juni 1977 in Stollberg/E.

eingereicht am 26. Juli 2005
Gutachter: Prof. Dr. Karl Heinz Hoffmann
Prof. Dr. Michael Schreiber
Prof. Dr. Peter Salamon

Tag der Verteidigung: 14. Oktober 2005

http://archiv.tu-chemnitz.de/pub/2005/0138

Bibliographische Beschreibung

Heilmann, Frank
The State Space of Complex Systems
Technische Universität Chemnitz, Fakultät für Naturwissenschaften,
Dissertation, 2005 (in englischer Sprache)
110 Seiten, 21 Abbildungen, 2 Tabellen, 108 Literaturzitate

Referat

In dieser Arbeit wird eine Beschreibung von Monte-Carlo-Verfahren zur Lösung komplexer Optimierungsaufgaben mit Hilfe von Markov-Ketten durchgeführt. Nach einer kurzen Einführung werden Lösungsmenge solcher Aufgaben und der physikalische Zustandsraum komplexer Systeme identifiziert.

Zunächst wird die Dynamik von Zufallswanderern im Zustandsraum mit Hilfe von Master-Gleichungen modelliert. Durch Einführung von Performanzkriterien können verschiedene Optimierungsstrategien quantitativ miteinander verglichen werden. Insbesondere wird das Verfahren *Extremal Optimization* vorgestellt, dass ebenfalls als Markov-Prozess verstanden werden kann. Es wird bewiesen, dass eine im Sinne der genannten Kriterien beste Implementierung existiert. Da diese von einem sogenannten *Fitness Schedule* abhängt, wird dieser für kleine Beispielsysteme explizit berechnet.

Daran anschließend wird die Zustandsdichte komplexer Systeme betrachtet. Nach einem kurzen Überblick über vorhandene Methoden folgt eine detaillierte Untersuchung des Verfahrens von Wang und Landau. Numerische und analytische Hinweise werden gegeben, nach denen dieser Algorithmus innerhalb seiner Klasse wahrscheinlich der Optimale ist. Eine neue Methode zur Approximation der Zustandsdichte wird vorgestellt, die insbesondere für die Untersuchung komplexer Systeme geeignet ist. Abschließend wird ein Ausblick auf zukünftige Arbeiten gegeben.

Schlagworte

Statistische Physik, Globale Optimierung, Stochastisches Suchverfahren, Markov-Ketten-Monte-Carlo-Verfahren, Simulated Annealing, Extremal Optimization, Kontrolltheorie, Zustandsdichte, Verfahren von Wang und Landau, Transition Matrix Monte Carlo

Contents

Chapter 1

Introduction

The most important concept in statistical physics is the state space of a system. It is the set of all possible states the system can be found in. On the basis of the state space a dynamics can be formulated. In statistical physics this dynamics introduces a probabilistic view on the evolution of the system. By describing how the probability to find the system in a specific state changes in time it provides the opportunity to calculate physical quantities time-dependently. Most of these quantities are weighted averages, calculated over the whole state space.

The past decades have seen the successful application of the concepts of statistical physics to one of the most important research fields: combinatorial optimization. Here, the task is to extract those solutions from a very large set of possible ones which minimize a given energy function. The possibility to identify the set of possible solutions of such an optimization problem with a corresponding state space opens the door for the utilization of the rich set of methods of statistical physics to solve the task.

But not only physically inspired methods have been introduced. Also biologically, evolutionary and even co-evolutionary inspired ideas and concepts have been developed. Nowadays, it seems that the borders between these subjects become more and more fluent, and it is sometimes very hard to classify optimization algorithms accordingly. For example, a genetic algorithm – clearly initiated by biological and evolutionary principles – is describable easily by the standard mathematical tool of Markov chains, which is heavily used for modeling in statistical physics. Hence, one and the same algorithm could be seen as almost "pure biological with some probability in it", but also as purely probabilistic with

"biological dynamics". Indeed, it is much easier to classify optimization algorithms by another scheme: is the algorithm stochastic or not?

This introductory chapter starts with a short description of multi-objective optimization, and how it can be confined to a single-objective optimization. Chapter 2 includes the theoretical foundation for the results gathered in this work. State spaces are introduced, and a short outline of the most often used non-stochastic algorithms to solve optimization problems is presented. Mathematical tools are introduced to describe and characterize stochastic procedures. Random walks will be described in a framework given by the theory of MARKOV processes. The master equation will be introduced.

A description of stochastic optimization methods in general, and of a new method called *Extremal Optimization* in particular, will be given in Chapter 3. This optimization algorithm is some kind of mixture of physically and evolutionary inspired methods. One of the main questions to be answered here is how this algorithm can be implemented in an optimal way. Optimal control theory is used to apply the method to small complex systems.

Chapter 4 shows in which way the density of states of a complex system can be determined. Besides the most often used argument why this density is important – it makes the solution of the thermodynamics of the system possible – another one is given which is connected with optimization problems.

Conclusions are made in chapter 5, ending with some remarks about the work which still has to be done. Outlooks are given how research in this subject could be driven forward.

Most of the results presented here are results of computer simulations, some are analytical ones. Therefore, this work should be considered as a contribution to computational physics.

From Multi-objective to Single-Objective Optimization – How to Buy a Car the Theorist's Way

Suppose your family grows, and very soon you feel that a new car would come in handy. Further suppose that in order to not making a mistake you collect data on quality, maybe defined by repair costs, and price for 100 different car models available on the market. How to select the best one?

You know that there are huge differences in quality, but it can be observed that one and the same level of quality can be achieved by dif-

ferent prices. Therefore, you plot the data, quality vs. price. You define that a model a dominates another one b, $a \succ b$, if it offers at least the same quality at a lower price. You can decide whether models dominate each other for most of the pairs of cars. Obviously, it would be really a mistake to buy a dominated car, so you safely delete them from the dataset.

The cars which are left over are a little bit extraordinary. They offer a higher or lower quality than any other car, but at a higher and lower price, respectively. Hence $a \not\succ b$, but also $b \not\succ a$ for all remaining cars a, b. It is one of these so-called "PARETO" optimal cars you have to buy, but which one is so far simply a matter of taste. It might depend on how much money you are able or willing to pay, whether you prefer quality or price, or another objective like design and prestige of the producer of the models in question.

Since finding an optimal car in the described way depends on multiple objectives such an optimization task is called a multi-objective optimization. Its goal has been achieved if all PARETO optimal solutions are identified; the following decision which solution is to be chosen in the end is not computable within this framework. Of course, if there were only one objective, this decision would be very easy; then the solution which extremalizes the objective would be the only one of interest. In that sense a multi-objective optimization can easily be transformed into one with a single objective: by weighing every objective with a factor and summing them up we can reduce the task to a simple extremalization problem. Of course, dependent on the weighting factors different optimal solutions will be found.

In fig.1.1 a hypothetical data set for 100 cars is plotted. The PARETO optimal cars are given, together with a weighting which favors the quality of the car and a weighting which favors a low price. As can be seen, these different weightings lead to different optimal solutions. Of course, also other weightings, maybe nonlinear ones, are possible.

This very small example shows the principle possibility to transform every multi-objective into a single-objective optimization. As we will see in the next chapter, even a single-objective optimization is most often a computationally hard task due to the existence of lots of local extrema. These two facts can be seen as the reason why the present work is confined to this case.

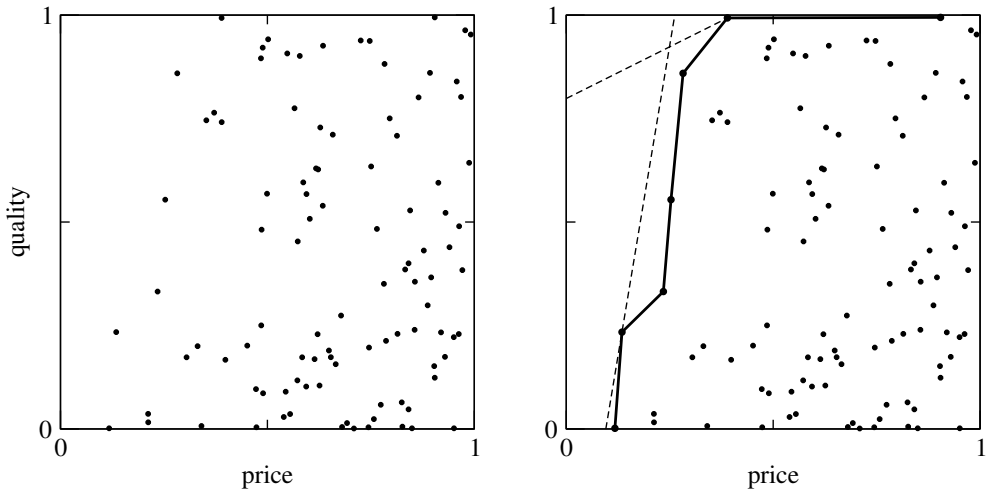

Figure 1.1: Left: The hypothetical data for quality and price for 100 cars. Right: The PARETO optimal cars are connected by the thick line. The broken lines denote linear functions which weigh quality and price differently.

Chapter 2

Dynamics in Complex State Spaces

In this chapter complex systems are discussed, focusing on the description of their state space and the simulation of dynamics within them. Some introduction about what makes a system complex in the sense of this thesis is made, and the search for finding the ground states and other low lying states is motivated. Some examples for complex systems are given. After that, the probably most important means for bringing dynamics into play is introduced: the tracing of random walkers. In the context of Markov processes and the probabilistic view an analytic description of the dynamics – the master equation – is outlined. The chapter closes with some facts about stochastic matrices and limiting distributions in general.

2.1 State Spaces of Complex Systems

In statistical physics a system is described by its state space Ω. This is simply the set of all microscopic states $s \in \Omega$ the system can be in. A microscopic state can be, e. g., the positions and velocities of the particles of a piece of matter, maybe further specified by the orientations of their magnetic moments in an magnetic field. Due to the normally large number of degrees of freedom a state of the system comprises the state space is generally of very high dimensionality. A gas of 10^{23} particles, e. g., may possess a state space with dimension $2 \cdot 3 \cdot 10^{23}$, because the x, the y and the z component of the position vector as well as those of the velocity vector of every particle might be changed independently.

The dimensionality could even raise if other degrees of freedom – like the mentioned magnetic moment – are taken into account.

Besides the high dimensionality there is another important property of the state space. The number of states $L = |\Omega|$ itself is extremely large, or even uncountable in the case of continuously varying degrees of freedom. If it is finite, the state space is said to be discrete. The particles of the mentioned gas, e. g., shall possess five orientations of a magnetic moment in an external field. If they are assumed to be fixed in three-dimensional space then they offer $5^{(10^{23})}$ different states. Their state space would be discrete.

Moreover, a function $H = H(s)$ is defined on the states s. It associates an energy with every state. Such an energy function for a state of the 10^{23} particles of the gas with magnetic moment $\boldsymbol{\mu}_j$ could read

$$H(s) = \sum_{j=1}^{10^{23}} \frac{m}{2} \boldsymbol{v}_j^2 - \sum_{j=1}^{10^{23}} \boldsymbol{\mu}_j \cdot \boldsymbol{B}, \tag{2.1}$$

if the particles of mass m are assumed to be independent and are moving in a magnetic field $\boldsymbol{B}$. With that, macroscopic quantities like inner energy, entropy or specific heat can be calculated as weighted means over the state space.

A dynamics can be introduced into such state space by the concept of random walks. We will have a detailed look on these in the next section, but to complete the description of the structure of the state space of complex systems we mention them here. Random walks are just a number of steps a walker takes in the state space of the system under consideration. Being in the current state α, a transition or step to a neighboring state $\beta \in \mathcal{N}(\alpha)$ is performed. The predefined so-called neighborhood relationship $\mathcal{N}(\alpha) \subset \Omega$ is the set of all states which are just one step away from α, and are therefore proposable as next step. The walks are called random because the state to be taken actually as next step is selected with some probability and therefore unknown *a priori*. In that sense the neighborhood relationship is also called "move class".

A local minimum is simply defined as a state the energy of which is smaller than that of all neighboring states. Barriers separate minima, and maxima are defined analogous to minima. A system is called *complex* if its state space contains lots of local minima, barriers and maxima. We speak about the bumpy, hilly or mountainous energy landscape of the complex system, because a mountain range offers in principle the same structure.

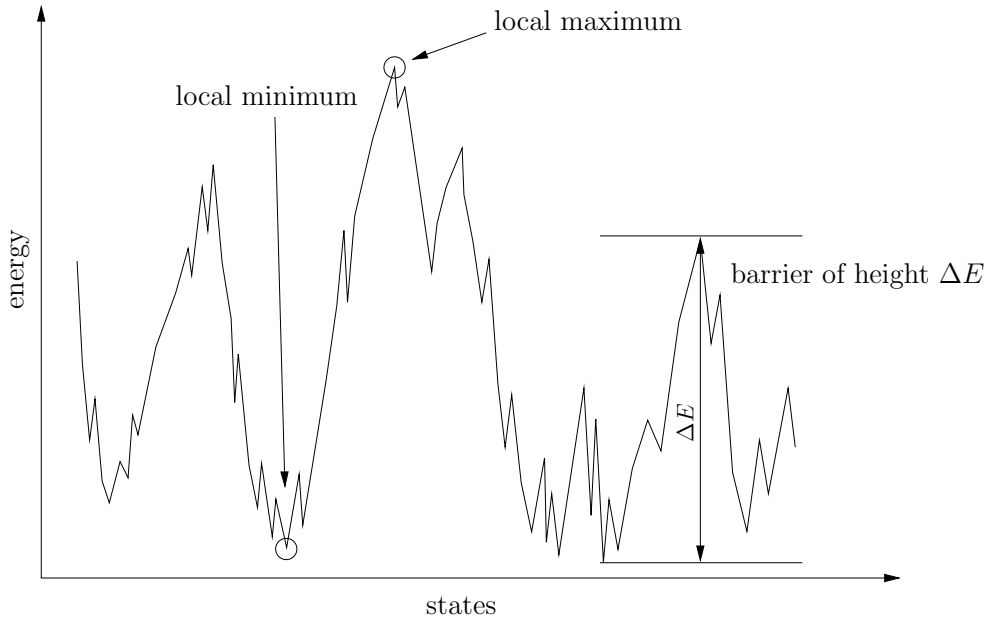

Figure 2.1: A typical random walk through the state space of a complex system. The walker experiences the energy function defined on the underlying state space; the visited states reflect local minima, separated by barriers, and local maxima. The minima are located at different energies.

A typical random walk through the energy landscape of a complex system is shown in fig. 2.1. Such a random walker experiences the energy function defined on the states, hence, the states which are visited during the walk reflect the complex structure of the underlying state space.

There are many examples for complex systems in physics. One of the most prominent is a spin glass. Such a system is a collection of atoms with a magnetic moment, e. g., caused by an unpaired spin. The spins – maybe those of manganese or nickel atoms – are located randomly in an non-magnetic matrix like gold or copper. There are many realizations of spin glasses, e. g., metallic compounds or insulators (see e. g. [1, 2]). The term spin glass is derived from the fact that in analogy to the unordered microscopic structure of glass the "magnetic structure" of a spin glass is unordered: the spins are spread randomly, and are randomly aligned.

This random alignment is caused by an interaction between two spins which is strongly influenced by the surrounding electrons. Depending on the distance between two spins this interaction has changing strength and sign, i. e., it can be ferro- or anti-ferromagnetic. This behavior was studied by Ruderman and Kittel [3] in the context of nuclear magnetism,

and further developed by KASUYA [4] and YOSIDA [5]. Therefore, it is known as the RKKY interaction.

Due to the random positions of the magnetic impurities this interaction can also be considered random in strength and sign. A simple model covering this feature has been introduced by EDWARDS and ANDERSON [6]. A number of spin vectors $\boldsymbol{s}_i$ are placed on a regular lattice with random values J_{ij} for the interaction between them. For a configuration S of such spins the energy is defined as

$$H(S) = -\sum_{\langle i,j \rangle} J_{ij} \cdot \boldsymbol{s}_i \cdot \boldsymbol{s}_j \tag{2.2}$$

with the summation performed over all pairs of neighboring spins. The choices for the distribution of the J_{ij} are, e. g., Gaussian, flat between $+I$ and $-I$, or $\pm I$, $I \in \mathbb{R}$. In the latter case the state space is highly degenerated.

Even simpler models are the xy-model and the ISING spin model. The former utilizes only the x- and the y-component of the spin, whereas the latter only treats the z-component. Here, the z-component only takes the values $+1$ and -1. But even this very simple model exhibits all the features a complex state space offers: lots of local minima, separated by barriers of different height. Long-range versions with summation over all spin pairs are called SHERRINGTON-KIRKPATRICK models [7].

One of the most important questions regarding the state space of complex systems is: What is the ground state of the system, and what other low lying states do exist? The so-called ground state is that one which minimizes the energy function globally. In low-temperature physics it is of general interest, because this state is the one the system is in with highest probability at very low temperatures. But other low lying states are of interest, too. They are those the system should be in at somewhat higher temperatures. This is closely related to questions about the structure and thermodynamic stability of substances [8, 9, 10]. In other fields of science the ground states simply realize the optimal solution of a given problem. Therefore, finding this state is called an optimization.

To underline this we have a glance on the wide class of complex combinatorial problems. These can be seen as the mathematical formulation of questions about the ground state originating from physical, engineering, economical and information technological investigations, and are describable by means of statistical physics. They form a kind of playground for the methods and optimization algorithms to be studied, because they

combine the advantage of being easy to implement with a rich set of effects observable in complex systems. For example, the state space of the ISING spin glass of n spins can be seen as the set of all vectors of length n which are combinations of plus and minus one – there are 2^n of them. Which combination of them is the ground state?

Other important combinatorial problems are:

- The traveling salesman problem (TSP). Given a set of n towns, located randomly, find the shortest tour which visits every town exactly once, starting and ending in the same town.

  The origins of this task can obviously be found in economics and engineering. Nowadays every car navigation system has to answer similar questions.

- The graph partitioning problem. Given is a set of n points, or vertices, in 2D space, n even. The vertices are randomly connected by edges. Find the two subsets each containing $n/2$ vertices such that the number of edges connecting a vertex of one subset with a vertex of the other subset is minimal.

  This question originates from chip design and layout: transporting electrons within a semiconductor chip (through the edges connecting vertices of the same subset) is faster and less dissipating compared to transporting electrons through inter-chip connecting wires (edges between vertices belonging to different subsets). Cluster computing has equivalent problems: inter-process communication within a node is much faster compared to inter-node communication.

- The graph coloring problem. Given a set of vertices, connected by edges, label each vertex with a color in such a way that no two connected vertices have the same color. Find the labeling which needs the minimum number of colors.

  A very active application for graph coloring is the allocation of registers (colors) of CPUs during the execution of a program, (represented by the graph). Variables (vertices) in registers can be accessed much quicker than those in RAM, but the number of registers is limited. In general, there are many more variables than registers, so multiple variables must be assigned to a register. Conflicts occur if one register is used for more than one variable at the same time (two connected vertices have the same color). These

have to be solved by assigning the variables that do not conflict in a way which minimizes RAM usage.

- A Combination. Find the shortest tour of a TSP with a program using as less RAM as possible, employing a compute cluster with as less inter-node communication as possible.

All of the mentioned combinatorial problems have complex state space with a huge number of states. Especially these examples motivate the search for well understood, efficient optimization algorithms.

2.2 Non-Stochastic Optimization

In order to find the ground states of complex optimization problems a lot of attempts have been made to introduce proper algorithms. Surely, the simplest approach – calculate the energy for every state and compare – is in general not feasible due to the very large number of states. Hence, other methods have been developed. Three of the most important will be shortly explained in the following.

Greedy Algorithms

Suppose a state is given. How to reduce the associated energy? One way would be to calculate the energy of all neighbors of the current state, and to take a step onto that with the lowest energy. Repeating these two steps results in finding the next local minimum.

Such an algorithm is called "greedy"; it always chooses the locally maximum possible gain. Generally, only sub-optimal solutions are located, but it can be shown that for some problems a greedy algorithm always finds the global minimum. There are some similarities to the "Steepest Descent" method explained below.

The Branch-and-Bound approach

Finding the ground state can also be seen as a moving on a decision tree described by two basic steps, "branching" and "bounding". The tree itself is created by branching the original problem; it is split into two or more subproblems which are easier to solve. Branching is done recursively, the created subproblems are again split by a branching step.

A solution of the original problem can be reconstructed from the sub-solutions found on the branches of the tree. In order to not having to look on every branch of the created tree a bounding step has to be implemented. The so-called "upper bound" is the best energy seen so far. Furthermore, a "lower bound" can be calculated by estimating how small the energy of the current state might become. If this lower bound is greater than the upper bound then the constructable state cannot be the ground state, and the whole branch can be truncated. This considerably saves computing time.

Conjugate Gradients

In this thesis, mostly discrete states spaces – i. e., such with a finite number of states – will be investigated. However, in a later section a generalization of an optimization method intended for discrete state spaces to the continuous case will be considered, and an algorithm which is able to deliver local minima in such state space will be needed. As it is the most commonly used technique a short description of the conjugate gradient method [11, 12] is given here.

Finding a (local) minimum of some energy function f defined on a continuous state space can easily be done by employing the gradient $\boldsymbol{f'} := \mathrm{grad} f$ of f. If at some position $\boldsymbol{x}$ in the state space $\boldsymbol{f'}(\boldsymbol{x}) = \boldsymbol{0}$ and $\boldsymbol{x} < \boldsymbol{y}$ for all $\boldsymbol{y}$ in a small neighborhood of $\boldsymbol{x}$ then $\boldsymbol{x}$ must be a local minimum. Hence, a very simple iterative gradient-based technique can be developed:

1. Start at some position $\boldsymbol{x}_i$ in the state space.

2. Calculate the "direction" $-\boldsymbol{f'}(\boldsymbol{x}_i)$.

3. Perform a line minimization along this direction to get $\boldsymbol{x}_{i+1}$, the corresponding minimum.

4. Set $\boldsymbol{x}_i = \boldsymbol{x}_{i+1}$, go to 2 as long a some convergence criterion is not met.

This method, called *Steepest Descend*, performs poorly, because successive gradients will be orthogonal to each other [12]. This leads to a large number of steps required to find the minimum in narrow "valleys" of the state space.

What really is needed are successive directions which are orthogonal to each other "with some respect to the shape of the energy landscape". Near a local minimum this "shape" can be expressed by a quadratic form

$$f(\boldsymbol{x}) \approx c + \boldsymbol{b} \cdot \boldsymbol{x} + \frac{1}{2}\left(\boldsymbol{x}^{tr} \cdot \boldsymbol{A} \cdot x\right) \tag{2.3}$$

with some special $\boldsymbol{A}$ and $\boldsymbol{b}$, and $(.)^{tr}$ denoting the transpose. Two vectors $\boldsymbol{v}_1, \boldsymbol{v}_2$ are said to be $\boldsymbol{A}$-orthogonal or conjugate if

$$\boldsymbol{v}_1^{tr} \cdot \boldsymbol{A} \cdot \boldsymbol{v}_2 = 0. \tag{2.4}$$

It can be shown that successive line minimizations along "conjugate" directions converge much faster to the local minimum. Algorithms to construct such directions are called *Conjugate Gradient Methods*. These try to circumvent the problem of the matrix $\boldsymbol{A}$ being unknown by – of course – calculating gradients of the energy function. A detailed description, together with some "canned algorithms", can be found in [11].

2.3 Random Walks and Markov Processes

In order to be able to describe stochastic optimization as a MARKOV process some basic definitions and facts have to be given. For a discrete state space the meaning of the move class $\mathcal{N}(\alpha) \in \Omega$ for a state α has already been defined. The probability to select a state $\beta \in \mathcal{N}(\alpha)$ is denoted with $\Pi_{\beta\alpha}$. Calculating and coding $\Pi_{\beta\alpha}$ might not be trivial as can be seen in the case of fractals [13, 14].

Whether a transition of a random walker from one state α to another one $\beta \in \mathcal{N}(\alpha)$ is performed depends on the so-called acceptance probability $P_{\beta\alpha}(t)$ which might be dependent on the time t. Generally, there exists no restriction how this probability can be chosen. But in physical investigations it is mostly somehow connected to the energies of the states α and β.

Let us restrict ourselves to discrete times t_k for the moment. Performing a step in a random walk is a two-step procedure of selecting a neighbor and accepting or rejecting it. Hence, the total transition probability $\Gamma_{\beta\alpha}$ is

$$\Gamma_{\beta\alpha}(t) = \Pi_{\beta\alpha} \cdot P_{\beta\alpha}(t) \tag{2.5}$$

We see that we have at least two major choices for how we could define $P_{\beta\alpha}$, hence $\Gamma_{\beta\alpha}(t)$: we could make $P_{\beta\alpha}$ "history-dependent", i. e.,

$P_{\beta\alpha}(t_k) = P_{\beta\alpha}(\{t_i\}, i = 0, \ldots, k)$, or not "history-dependent", $P_{\beta\alpha} = P_{\beta\alpha}(t_k)$. The latter choice, called MARKOV property, secures that the next state of the random walk only depends on the current state. Explicitly, it does not depend on states already visited. Such a special random walk produces MARKOV chains [15]. A MARKOV chain is the collection of the state probabilities which evolve due to successive transitions with MARKOV property.

The states can be classified according to how they can be populated by transitions. The most important are [15]

- **absorbing states**: they can never be left;
- **periodic and aperiodic states**: a periodic state is visited again with non-zero probability after a number of steps s, and $1 < s < \infty$. If $s = 1$ or $s = \infty$ the state is called aperiodic.
- **recurrent and transient states**: a recurrent state, once left, is visited again with certainty; a transient state, once left, is never again visited with non-zero probability. A positive-recurrent state has the property that the mean time for the first return to that state is finite.

Accordingly, the most important types of MARKOV chains are

- **irreducible chains**: in such chains every state is reachable from any other state by a finite number of transitions (such chains cannot contain absorbing states or sets of states that "absorb"),
- **aperiodic chains**: an aperiodic chain consists only of aperiodic states,
- **recurrent and transient chains**: a recurrent chain consists only of recurrent states, and a transient one only of transient states.

If $P_{\beta\alpha}$ is not a function of t at all, the MARKOV chain is called homogeneous, otherwise inhomogeneous.

In the next chapter we will see how understanding stochastic optimization as a MARKOV process opens the possibility to prove which type of processes are best suited for optimization in general. To do so, we need a mathematical tool to describe MARKOV processes.

2.4 The Master Equation

We assume an infinite number of random walks with MARKOV property, performed in parallel independently of each other. The question how the states of the system are populated as time progresses can be answered by setting up the master equation [15, 16] for the system.

2.4.1 Continuous Time

In a discrete state space Ω with $|\Omega| = N$ the states can be labeled $1, 2, \ldots, N$. Due to the infinite number of walkers involved a time dependent probability vector $p_i(t)$ can be introduced, giving the probability to be in state i at time t. This probability can change in time by the inflow of probability from other states and out-flow of probability to other states. If the probability transition rate between two distinct states i and j is denoted with $\Gamma_{ji}(t)$, and $\Gamma_{ij}(t) \geq 0$, the total change reads

$$\dot{p}_j(t) = \sum_{i=1, i\neq j}^{N} \Gamma_{ji}(t) p_i(t) - \sum_{i=1, i\neq j}^{N} \Gamma_{ij} p_j(t). \tag{2.6}$$

This can be written in vector form

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{p}(t) = \boldsymbol{\Gamma}(t) \cdot \boldsymbol{p}(t) \tag{2.7}$$

with a square matrix

$$\boldsymbol{\Gamma}(t) = [\Gamma_{ij}(t)] \in \mathbb{R}^{N\times N} \tag{2.8}$$

the diagonal elements of which are set to $\Gamma_{jj}(t) = -\sum_{i=1;i\neq j}^{N} \Gamma_{ij}(t)$. Therewith,

$$\sum_{j=1}^{N} \dot{p}_j(t) = 0, \tag{2.9}$$

i. e., normalization of $\boldsymbol{p}(t)$ is ensured for all times. The vector $\boldsymbol{p}(t)$ is the probability distribution over the state space at time t.

If $\boldsymbol{\Gamma}$ is not explicitly dependent on t, then the solution of this system of linear differential equations is formally given by

$$\boldsymbol{p}(t) = \exp(t \cdot \boldsymbol{\Gamma}) \cdot \boldsymbol{p}(0). \tag{2.10}$$

The vector $\boldsymbol{p}(0)$ is given by initial conditions. The exponential of the matrix $t \cdot \boldsymbol{\Gamma}$ can be defined as a Taylor expansion at $t = 0$ of the exponential function,

$$\exp(t \cdot \boldsymbol{\Gamma}) := \sum_{i=0}^{\infty} \frac{1}{i!} \left[t \cdot \boldsymbol{\Gamma}\right]^i . \tag{2.11}$$

This expansion becomes especially simple if $\boldsymbol{\Gamma}$ can be transformed into diagonal form, because then the matrix powers can be computed very easily. For that, we determine the eigenvectors $\boldsymbol{e}_i$, use them as the columns of a matrix $\boldsymbol{E}$ and perform the similarity transformation

$$\boldsymbol{E}^{-1}\boldsymbol{\Gamma}\boldsymbol{E} = \mathrm{diag}\{e_1, e_2, \ldots, e_N\} =: \boldsymbol{D} \tag{2.12}$$

with e_i being the eigenvalue to eigenvector $\boldsymbol{e}_i$. Then, due to (2.11)

$$\exp(t \cdot \boldsymbol{D}) = t \cdot \mathrm{diag}\{\exp(e_1), \exp(e_2), \ldots, \exp(e_N)\}, \tag{2.13}$$

and (2.7) can be written as

$$\begin{aligned} \boldsymbol{E}^{-1}\left(\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{p}(t)\right) &= \left(\boldsymbol{E}^{-1} \cdot \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{p}(t)\right) && (2.14)\\ &= \left(\boldsymbol{E}^{-1}\boldsymbol{\Gamma}\boldsymbol{E}\right)\left(\boldsymbol{E}^{-1} \cdot \boldsymbol{p}(t)\right) && (2.15)\\ &= \boldsymbol{D}\left(\boldsymbol{E}^{-1} \cdot \boldsymbol{p}(t)\right). && (2.16) \end{aligned}$$

Therefore, with $\tilde{\boldsymbol{p}}(t) = \boldsymbol{E}^{-1} \cdot \boldsymbol{p}(t)$ we have a set of simple differential equation of first order for every entry of $\tilde{\boldsymbol{p}}$ which can be solved independently of each other,

$$\tilde{p}_i(t) = \tilde{p}_i(0) \cdot \exp(e_i \cdot t). \tag{2.17}$$

This solution can than be re-transformed by left multiplication with $\boldsymbol{E}$, yielding the solution $\boldsymbol{p}(t)$. Of course, the integration constants $\tilde{p}_i(0)$ have to be chosen such that $\boldsymbol{p}(0)$ has the right value. This reduces to the solution of a linear system of equations.

But not every matrix can be transformed into such a simple diagonal form. This may happen if some of the eigenvalues appears multiple times, i. e., the characteristic polynomial

$$p(e) = \det(\boldsymbol{\Gamma} - e\boldsymbol{I}) \tag{2.18}$$

has roots e_i which appear multiple times. Then the corresponding matrix might not be diagonalizable, and in that case the way to solve (2.7) must be changed slightly.

It can be shown that every square matrix can be transformed into the so-called JORDAN normal form [17]. This form is probably the one which can most easily be dealt with if the matrix cannot be diagonalized. As a detailed description would be beyond the scope of the present work, only the following short notes are given. A JORDAN block $J_k(e)$ to eigenvalue e is a k-by-k upper triangular matrix of the form

$$J_k(e) = \begin{bmatrix} e & 1 & 0 & 0 & \dots & 0 \\ & e & 1 & 0 & \dots & 0 \\ & & \ddots & \ddots & \ddots & \vdots \\ & & & e & 1 & 0 \\ & \mathbf{0} & & & e & 1 \\ & & & & & e \end{bmatrix}. \tag{2.19}$$

Such a block has non-zero entries only on the main diagonal and on the superdiagonal. The $\mathbf{0}$ represents a 0 for all entries not explicitly given.

The JORDAN normal form $\boldsymbol{J}$ of a matrix $\boldsymbol{\Gamma} \in \mathbb{R}^{N\times N}$ is a direct sum of JORDAN blocks,

$$\boldsymbol{J} = \begin{bmatrix} J_{k_1}(e_1) & & & \mathbf{0} \\ & J_{k_2}(e_2) & & \\ & & \ddots & \\ \mathbf{0} & & & J_{k_n}(e_n) \end{bmatrix}, \; k_1 + \dots + k_n = N \tag{2.20}$$

The k_i need not be distinct, and the e_i need not be distinct. That means, one and the same eigenvalue might be represented by multiple JORDAN blocks of different size. But the JORDAN form is unique, apart from permutations of the Jordan blocks along the "JORDAN-block diagonal" of $\boldsymbol{J}$. A diagonal matrix is simply a special JORDAN form.

With the corresponding similarity matrix $\boldsymbol{S}$ and

$$\boldsymbol{J} = \boldsymbol{S}^{-1} \cdot \boldsymbol{\Gamma} \cdot \boldsymbol{S} \tag{2.21}$$

it is possible to solve the system of linear differential equations (2.7) with a non-diagonalizable $\boldsymbol{\Gamma}$ not dependent on t in the following way. We have again

$$\begin{aligned} \boldsymbol{S}^{-1}\left(\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{p}(t)\right) &= \left(\boldsymbol{S}^{-1}\cdot\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{p}(t)\right) && (2.22)\\ &= \left(\boldsymbol{S}^{-1}\boldsymbol{\Gamma}\boldsymbol{S}\right)\left(\boldsymbol{S}^{-1}\cdot\boldsymbol{p}(t)\right) && (2.23)\\ &= \boldsymbol{J}\left(\boldsymbol{S}^{-1}\cdot\boldsymbol{p}(t)\right). && (2.24) \end{aligned}$$

With $\tilde{\boldsymbol{p}}(t) = \boldsymbol{S}^{-1} \cdot \boldsymbol{p}(t)$ we can solve the coupled system of linear differential equations

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{\boldsymbol{p}}(t) = \boldsymbol{J} \cdot \tilde{\boldsymbol{p}}(t) \tag{2.25}$$

by recursively solving all subsystems corresponding to a JORDAN block. This is possible, as every subsystem is independent from all other subsystems. We denote the subset of the $\tilde{p}_i(t)$ corresponding to the m-th JORDAN block $J_{k_i}(e_j)$ with $\tilde{p}_1^m(t), \ldots, \tilde{p}_{k_i}^m(t)$. Then

$$\begin{aligned}
\tilde{p}_{k_i}^m(t) &= \tilde{p}_{k_i}^m(0) \cdot \exp(e_j \cdot t) && (2.26)\\
\tilde{p}_{k_i-1}^m(t) &= \tilde{p}_{k_i-1}^m(0) \cdot \exp(e_j \cdot t) + t \cdot \tilde{p}_{k_i}^m(0) \cdot \exp(e_j \cdot t) && (2.27)\\
\tilde{p}_{k_i-2}^m(t) &= \tilde{p}_{k_i-2}^m(0) \cdot \exp(e_j \cdot t) + t \cdot \tilde{p}_{k_i-1}^m(0) \cdot \exp(e_j \cdot t) \\
&\quad + \frac{1}{2}t^2 \cdot \tilde{p}_{k_i}^m(0) \cdot \exp(e_j \cdot t) && (2.28)\\
&\vdots \\
\tilde{p}_1^m(t) &= \sum_{l=1}^{k_i} \frac{1}{(l-1)!} \cdot \tilde{p}_l^m(0) \cdot t^{l-1} \cdot \exp(e_j \cdot t). && (2.29)
\end{aligned}$$

Carrying out this construction for every JORDAN block yields the complete solution for the system of coupled differential equations. The N integration constants $p_i^m(0)$ can again be chosen such that the solution of the original system $\boldsymbol{p}(t) = \boldsymbol{S} \cdot \tilde{\boldsymbol{p}}(t)$ obeys the given initial conditions $\boldsymbol{p}(0)$.

2.4.2 Discrete Time

Very often a description of a random walk is needed which does not show a continuous change of the state probabilities. For example, a walk which performs a series of "hops" from one state to another at discrete times changes the probability of a state abruptly. To describe such a process a discrete master equation can be used.

A Taylor expansion of $\boldsymbol{p}(t + \Delta t)$ for small Δt to first order yields

$$\boldsymbol{p}(t + \Delta t) \approx \boldsymbol{p}(t) + \Delta t \cdot \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{p}(t). \tag{2.30}$$

A combination with (2.7) results in an approximated, discrete-in-time master equation,

$$\begin{aligned}
\boldsymbol{p}(t + \Delta t) &= \boldsymbol{p}(t) + \Delta t \cdot \boldsymbol{\Gamma}(t) \cdot \boldsymbol{p}(t) && (2.31)\\
&= [\mathbf{1} + \Delta t \cdot \boldsymbol{\Gamma}(t)] \cdot \boldsymbol{p}(t) && (2.32)\\
&= \boldsymbol{\Gamma}^d(t) \cdot \boldsymbol{p}(t). && (2.33)
\end{aligned}$$

The diagonal elements of the new transition matrix $\mathbf{\Gamma}^d(t)$ are

$$\Gamma^d_{jj}(t) = 1 - \Delta t \cdot \sum_{i=1, i\neq j}^{N} \Gamma_{ij}(t), \quad \text{hence,} \quad \sum_{i=1}^{N} \Gamma^d_{ij} = 1 \,\forall j. \tag{2.34}$$

Therefore, the sum of the elements of $\boldsymbol{p}(t)$ remains the same after an application of $\mathbf{\Gamma}^d$. In other words, normalization is ensured here, too.

We are free to choose $\Delta t = 1$ to define our time scale, but consequently we are forced to scale the entries of $\mathbf{\Gamma}^d$ according to (2.34). They must obey

$$\Gamma^d_{ij} \geq 0 \quad \text{and} \quad \sum_{i=1}^{N} \Gamma^d_{ij} = 1 \,\forall j, \quad \text{i. e.,} \quad \Gamma^d_{ij} \leq 1. \tag{2.35}$$

A matrix with such entries is called stochastic. Furthermore, setting $\Delta t = 1$ has the effect that (2.33) is only a good approximation for the continuous master equation (2.7) if the non-diagonal entries of $\mathbf{\Gamma}^d$ are very small, leading to only small changes of the state probabilities per time step.

Generally, $\mathbf{\Gamma}^d$ can be used to model any MARKOV process due to these processes being inherently discrete in time. Absorbing states are represented by a corresponding column with all entries equal to zero except the one on the main diagonal. Transient states are represented by a corresponding row with all entries equal to zero.

From now on the superscript $(.)^d$ will be suppressed, because this discrete-time description is the one which will be used throughout the rest of the present work. This is due to the fact that the dynamics which will be used is merely discrete in time: the transitions from one state to another are never continuous in problems investigated here, but can be considered to be jumps at certain points in time. The master equation then reads

$$\boldsymbol{p}(t+1) = \mathbf{\Gamma}(t+1) \cdot \boldsymbol{p}(t). \tag{2.36}$$

The labeling of the transition matrices has also been shifted by 1. This does not affect the time development of the state probabilities, but is the notation used in the literature. This shift refers to practical implementations in which at every time step the new transition matrix is calculated first and then applied to the vector of the state probabilities.

If $\mathbf{\Gamma}$ is not a function of t at all there is again the possibility to solve (2.36) by employing the eigenvalues e_i and corresponding eigenvectors $\boldsymbol{e}_i$

of $\mathbf{\Gamma}$. We have

$$\boldsymbol{p}(t+1) = \mathbf{\Gamma} \cdot \boldsymbol{p}(t), \quad \text{hence,} \quad \boldsymbol{p}(t) = \mathbf{\Gamma}^t \cdot \boldsymbol{p}(0) \tag{2.37}$$

and, assuming that $\mathbf{\Gamma}$ is diagonalizable,

$$\boldsymbol{D} = \boldsymbol{E}^{-1} \cdot \mathbf{\Gamma} \cdot \boldsymbol{E} \tag{2.38}$$

$$\boldsymbol{D}^2 = (\boldsymbol{E}^{-1} \cdot \mathbf{\Gamma} \cdot \boldsymbol{E})(\boldsymbol{E}^{-1} \cdot \mathbf{\Gamma} \cdot \boldsymbol{E}) = \boldsymbol{E}^{-1} \cdot \mathbf{\Gamma}^2 \cdot \boldsymbol{E} \tag{2.39}$$

$$\vdots \tag{2.40}$$

$$\boldsymbol{D}^t = \boldsymbol{E}^{-1} \cdot \mathbf{\Gamma}^t \cdot \boldsymbol{E} \tag{2.41}$$

Again, with $\tilde{\boldsymbol{p}}(0) = \boldsymbol{E}^{-1} \cdot \boldsymbol{p}(0)$, we can transform the original system of equations into

$$\tilde{\boldsymbol{p}}(t) = \boldsymbol{D}^t \cdot \tilde{\boldsymbol{p}}(0) = \mathrm{diag}(e_1^t, e_2^t, \ldots, e_N^t) \cdot \tilde{\boldsymbol{p}}(0), \tag{2.42}$$

which enables an immediate solution for every time step t. Furthermore, the $\tilde{p}_i(0)$ are just the coefficients of a linear combination of the eigenvectors to represent $\boldsymbol{p}(0)$.

If $\mathbf{\Gamma}$ is not diagonalizable there is again the possibility to transform it into JORDAN normal form $\boldsymbol{J}$. With the appropriate similarity matrix $\boldsymbol{S}$ we have again

$$\boldsymbol{J}^t = \boldsymbol{S}^{-1} \cdot \mathbf{\Gamma}^t \cdot \boldsymbol{S}. \tag{2.43}$$

The system

$$\tilde{\boldsymbol{p}}(t) = \boldsymbol{J}^t \cdot \tilde{\boldsymbol{p}}(0) \tag{2.44}$$

with $\tilde{\boldsymbol{p}}(0) = \boldsymbol{S}^{-1}\tilde{\boldsymbol{p}}(0)$ can be solved by evaluating the t-th power of $\boldsymbol{J}$. This can be carried out by first evaluating the t-th power of the JORDAN blocks, and then forming the direct sum of these powers. For a JORDAN block $J_{k_i}(e_j)$ to eigenvalue e_j

$$J_{k_i}(e_j) \cdot J_{k_i}(e_j) = \begin{bmatrix} e_j & 1 & & & \mathbf{0} \\ & \ddots & \ddots & & \\ & & \ddots & \ddots & \\ & & & \ddots & 1 \\ \mathbf{0} & & & & e_j \end{bmatrix}^2 \tag{2.45}$$

$$= \begin{bmatrix} e_j^2 & 2e_j & 1 & & \mathbf{0} \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & 1 \\ & & & \ddots & 2e_j \\ \mathbf{0} & & & & e_j^2 \end{bmatrix} \tag{2.46}$$

and all higher powers can be computed by a repeated matrix multiplication. The first row of $[J_{k_i}(e_j)]^t$ reads

$$[J(e_j))]_1^t = \left(\begin{array}{cccc} e_j^t & te_j^{t-1} & \frac{1}{2}t(t-1)e_j^{t-2} & \dots \quad \frac{1}{(k_i-1)!}\prod_{l=0}^{k_i-2}(t-l)e_j^{t-1} \end{array} \right) \tag{2.47}$$

All following rows can be constructed by shifting the first row to the right and filling the leading entry with 0, i. e.,

$$\begin{array}{rcl} [J(e_j)]_2^t & = & (0 \quad e_j^t \quad te_j^{t-1} \quad \frac{1}{2}t(t-1)e_j^{t-2} \quad \dots) \\ [J(e_j)]_3^t & = & (0 \quad 0 \quad e_j^t \quad te_j^{t-1} \quad \dots) \\ & \vdots & \end{array} \tag{2.48}$$

This construction, performed for every JORDAN block, enables the immediate computation of arbitrary powers $\boldsymbol{J}$. Hence, with appropriate $\tilde{\boldsymbol{p}}(0)$, the system (2.37) has been solved.

Some remarkable facts about stochastic matrices in general should be given here. We denote the spectrum of a matrix $\boldsymbol{A}$ with $\sigma(\boldsymbol{A})$, and the spectral radius with $\rho(\boldsymbol{A}) = \max_i(|e_i| : e_i \in \sigma(\boldsymbol{A}))$. It can be shown that the spectral radius $\rho(\boldsymbol{B})$ of a nonnegative matrix $\boldsymbol{B}, B_{ij} \geq 0$, is an eigenvalue of $\boldsymbol{B}$, and the corresponding eigenvector $\boldsymbol{b}$ can be chosen to be nonnegative, too, $b_i \geq 0$ [17]. Hence, in this case the eigenvalue with the largest absolute value is a nonnegative real number.

Stochastic matrices are, by definition, nonnegative matrices. That means, for a stochastic matrix $\boldsymbol{C}$ and the eigenvalue/eigenvector pair $\rho(C)/\boldsymbol{c}$ we have

$$\boldsymbol{C} \cdot \boldsymbol{c} = \rho(\boldsymbol{C}) \cdot \boldsymbol{c} \tag{2.49}$$

We scale $\boldsymbol{c}$ in such a way that the sum of its entries equals 1. After an application of $\boldsymbol{C}$ the sum must be 1 again. As $\boldsymbol{c}$ is an eigenvector, we can therefore state

$$\boldsymbol{C} \cdot \boldsymbol{c} \stackrel{!}{=} 1 \cdot \boldsymbol{c}, \tag{2.50}$$

hence,

$$\rho(\boldsymbol{C}) = 1. \tag{2.51}$$

Therefore, the largest eigenvalue of a stochastic matrix is 1. Due to this fact there always exist stationary distributions, which are linear combinations of the eigenvectors to eigenvalue 1. If, in addition, the MARKOV

chain is irreducible then the eigenvalue 1 is simple. The corresponding unique eigenvector with component sum 1 is also called the Perron vector $\boldsymbol{p}^*$ [17]. An initial distribution might develop into a stationary distribution $\boldsymbol{p}^*$, but this need not be the case [15].

If the stationary distribution can be guaranteed to be the limit of (2.36) for $t \to \infty$ and an arbitrary initial distribution the MARKOV chain produced by the transition probabilities is called *ergodic*. For example, irreducible, aperiodic and positive-recurrent MARKOV chains are ergodic [15].

In general, whether a system is ergodic or not can be decided by calculating the n-th power of its transition matrix $\boldsymbol{\Gamma}$. If

$$\Gamma_{ij}^n > 0 \text{ for some } n \geq n_0 \in \mathbb{N} \text{ for all } i, j, \tag{2.52}$$

i. e, $\boldsymbol{\Gamma}^n$ is a positive matrix, then it can be shown that the limit $\lim_{n\to\infty} \boldsymbol{\Gamma}^n$ exists, and

$$\boldsymbol{\Gamma}^\infty := \lim_{n\to\infty} \boldsymbol{\Gamma}^n = [\boldsymbol{p}^*|\boldsymbol{p}^*|\ldots|\boldsymbol{p}^*] \tag{2.53}$$

Hence, if $\Gamma_{ij}^\infty > 0$ for all i, j then the system is ergodic.[1] To show this we consider an arbitrary initial distribution $\boldsymbol{i}$ to be a linear combination of probability vectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_N$ with $\boldsymbol{v}_i = (\delta_{1i}, \delta_{2i} \ldots, \delta_{Ni})^{tr}$,

$$\boldsymbol{i} = \sum_{i=1}^{N} a_i \cdot \boldsymbol{v}_i \quad \text{with } a_i \geq 0, \ \sum_{i=1}^{N} a_i = 1. \tag{2.54}$$

Then

$$\begin{aligned} \boldsymbol{\Gamma}^\infty \cdot \boldsymbol{i} &= \boldsymbol{\Gamma}^\infty \cdot \left(\sum_{i=1}^{N} a_i \cdot \boldsymbol{v}_i \right) = \sum_{i=1}^{N} a_i \cdot \boldsymbol{\Gamma}^\infty \cdot \boldsymbol{v}_i = \sum_{i=1}^{N} a_i \cdot \boldsymbol{p}^* \\ &= \boldsymbol{p}^*, \end{aligned} \tag{2.55}$$

any initial distribution converges to the limit distribution.

The stationary distribution should not be confused with an *equilibrium* distribution (see also [18, 19, 20]). In equilibrium there is no net flow of probability between any two microscopic states; the system is said to obey *detailed balance*

$$\Gamma_{ji} \cdot p_i^* = \Gamma_{ij} \cdot p_j^*. \tag{2.56}$$

[1] Sometimes, this condition is relaxed a little bit: at least in [15] $\Gamma_{ij}^\infty \geq 0$ is also allowed. In that case, the stationary solution might also contain entries equal to 0.

This property need not be fulfilled by a stationary distribution. Therefore, all equilibrium distributions are stationary distributions, but not vice versa.

Let us close with four small systems to explain the given definitions. In fig. 2.2 a system of four states is given. Every state is represented by a large dot, and the transition probabilities are represented by an arrow with the corresponding value. The transition probabilities have been varied from (a) to (d), giving the transition matrices

$$\mathbf{\Gamma}_{(a)} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad \mathbf{\Gamma}_{(b)} = \begin{bmatrix} 5/6 & 0 & 0 & 0 \\ 1/6 & 1/3 & 3/4 & 3/4 \\ 0 & 1/3 & 1/4 & 0 \\ 0 & 1/3 & 0 & 1/4 \end{bmatrix}$$
$$\mathbf{\Gamma}_{(c)} = \begin{bmatrix} 5/6 & 1/6 & 0 & 0 \\ 1/6 & 1/6 & 1/2 & 3/4 \\ 0 & 1/3 & 1/4 & 0 \\ 0 & 1/3 & 1/4 & 1/4 \end{bmatrix} \quad \mathbf{\Gamma}_{(d)} = \begin{bmatrix} 5/6 & 1/6 & 0 & 0 \\ 1/6 & 1/6 & 3/4 & 3/4 \\ 0 & 1/3 & 1/4 & 0 \\ 0 & 1/3 & 0 & 1/4 \end{bmatrix} \quad . \tag{2.57}$$

The corresponding matrix powers $\mathbf{\Gamma}^{\infty}_{(\cdot)}$ can be calculated by diagonalization,

$$\lim_{n\to\infty} \mathbf{\Gamma}^{n}_{(a)} \text{ does not exist} \qquad \mathbf{\Gamma}^{\infty}_{(b)} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 9/17 & 9/17 & 9/17 & 9/17 \\ 4/17 & 4/17 & 4/17 & 4/17 \\ 4/17 & 4/17 & 4/17 & 4/17 \end{bmatrix}$$
$$\mathbf{\Gamma}^{\infty}_{(c)} = \begin{bmatrix} 27/82 & 27/82 & 27/82 & 27/82 \\ 27/82 & 27/82 & 27/82 & 27/82 \\ 6/41 & 6/41 & 6/41 & 6/41 \\ 8/41 & 8/41 & 8/41 & 8/41 \end{bmatrix} \quad \mathbf{\Gamma}^{\infty}_{(d)} = \begin{bmatrix} 9/26 & 9/26 & 9/26 & 9/26 \\ 9/26 & 9/26 & 9/26 & 9/26 \\ 2/13 & 2/13 & 2/13 & 2/13 \\ 2/13 & 2/13 & 2/13 & 2/13 \end{bmatrix} \quad . \tag{2.58}$$

As can be seen, system (a) is not ergodic, and only evolves to its stationary distribution $(0, 1/3, 1/3, 1/3)^{tr}$ if a proper initial condition is chosen. System (b) always converges to its stationary state, but is not ergodic, because the probability to be in state #1 tends to 0. System (c) is ergodic, but does not obey detailed balance; it has a stationary but no equilibrium distribution. System (d) is ergodic and has detailed balance; its stationary distribution is also an equilibrium distribution.

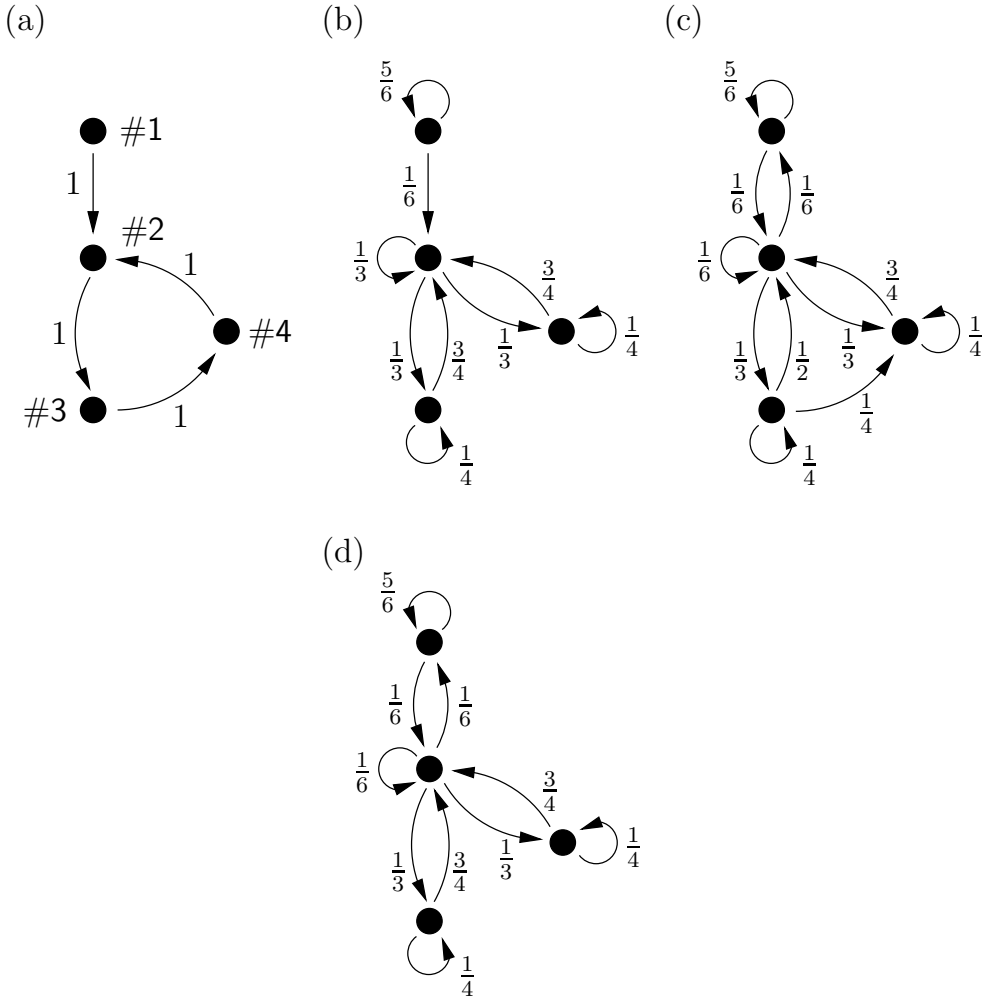

Figure 2.2: Diagrams of a system with four states. The numbering of the states #1 to #4 is given in (a). The different transition probabilities are represented by arrows. System (a) is non-ergodic, with no limiting distribution. System (b) is also non-ergodic, but converges to a limiting distribution regardless of the initial distribution. System (c) is ergodic, but its stationary distribution does not obey detailed balance. System (d) is ergodic, and the stationary distribution obeys detailed balance.

Chapter 3

Stochastic Optimization as a Markov Process

In this chapter a special view on stochastic optimization processes is developed: they can be described as MARKOV processes. Together with control theory it is possible to deduce an optimal steering of them. We start with a description how this can be carried out in the case of Simulated Annealing and Threshold Accepting. Numerical evidence that Threshold Accepting outperforms Simulated Annealing leads to an analytical proof that this is always the case.

Furthermore, this proof is extended to cover even Extremal Optimization. A special implementation – *Fitness Threshold Accepting* – is applied to real systems, and the optimal control formalism is able to deliver optimal fitness threshold schedules.

3.1 Annealing-Like Dynamics

Historically, stochastic optimization had been introduced as a means to circumvent the enormous effort to enumerate a combinatorial optimization problem. Instead of being sure to find an optimal solution it is hoped for that the probability to find the ground state or other low lying states can be made very high. Indeed, a very large number of investigations proved that stochastic optimization delivers very good solutions in acceptable time.

With the seminal work of KIRKPATRICK et al. [21, 22] and ČERNY [23] a first physically motivated stochastic optimization scheme – dubbed Simulated Annealing (SA) – was developed. It is based on the transition

rates given by the METROPOLIS sampling technique [24]. This sampling is widely used in the simulation of thermal properties of physical systems, its acceptance probability is

$$P_{\beta\alpha}^{Me} = \begin{cases} 1 & \text{if } H(\beta) - H(\alpha) \leq 0, \\ \exp\left(-\frac{1}{T}(H(\beta) - H(\alpha))\right) & \text{otherwise,} \end{cases} \tag{3.1}$$

with a temperature T scaled such that the BOLTZMANN constant k_B equals one. A random walker explores the state space according to (3.1), but the temperature is time-dependent and lowered at every step according to a prescribed schedule $T = T(t)$. The walker falls down into local minima, but does not get stuck in these. Instead, he is able to climb barriers with some probability. The idea is to make the probability to be in the ground state at the end of the run as high as possible by a very careful adjusting of $T(t)$. This is analogue to the annealing of a, say metal melting, which can also be turned into a mono-crystal – the configuration or state with the lowest energy – by careful lowering the temperature.

The computation of the exponential function in (3.1) is relatively costly. To speed up the algorithm, DUECK et al. [25] and MOSCATO and FONTANARI [26] changed it into the evaluation of a step function,

$$P_{\beta\alpha}^{TA} = \begin{cases} 1 & \text{if } \Delta E \leq T, \\ 0 & \text{if } \Delta E > T. \end{cases} \tag{3.2}$$

with $\Delta E = H(\beta) - H(\alpha)$. An algorithm with these transition probabilities is called Threshold Accepting (TA). Another technique, called TSALLIS statistics, has also been introduced [27, 28, 29]. Its transition probabilities depend on another parameter q,

$$P_{\beta\alpha}^{TS} = \begin{cases} 1 & \text{if } \Delta E \leq 0, \\ \left(1 - \frac{1-q}{f(q)}\frac{\Delta E}{T}\right)^{\frac{1}{1-q}} & \text{if } \Delta E > 0 \text{ and } \frac{1-q}{f(q)}\frac{\Delta E}{T} \leq 1 \\ 0 & \text{if } \Delta E > 0 \text{ and } \frac{1-q}{f(q)}\frac{\Delta E}{T} > 1. \end{cases} \tag{3.3}$$

Originally, $f(q) = 1$ was chosen. With a slight change given in [30],

$$f(q) = \begin{cases} 2 - q & \text{if } q < 2, \\ 1 & \text{if } q \geq 2, \end{cases} \tag{3.4}$$

this modified TSALLIS statistics gives the original TSALLIS statistics for $q \geq 2$, the limit $q \to 1$ yields (3.1), and $q \to -\infty$ gives (3.2) [31].

Therefore, (3.3) serves as generalization of all three strategies, which are selectable by the corresponding value of q.

In this sense stochastic optimization is just a MARKOV process as described the previous chapter. From a current state a neighboring one is selected with some probability and accepted with some probability. The most important feature of stochastic optimization is that the MARKOV chains produced are inhomogeneous. The transition matrix $\mathbf{\Gamma}$ depends on the time due to the acceptance probabilities being time-dependent. This dependence is coded as the prescribed, more or less arbitrary schedule $T(t)$. The outcome of an optimization depends heavily on a good selection of $T(t)$ [32]. Most often used are linear-in-time falling and exponential decreasing temperatures. The most important observation is that with a temperature schedule which cools too fast the system freezes: the random walkers then tend to get stuck in possibly high lying local minima, because the probability to escape these becomes too small too rapidly.

3.2 Stochastic Tunneling

The annealing-like algorithms described above try to circumvent the vulnerability of random walkers to get stuck in local minima by a temperature dependent acceptance probability. Another idea is to transform the energy landscape in such a way that local minima are flattened out, and barriers can be overcome easier. A corresponding method is Stochastic Tunneling (STUN) [33].

Instead of moving random walkers on the original energy landscape $H(s)$ the landscape is transformed into $H_{\mathrm{STUN}}(s)$ due to

$$H_{\mathrm{STUN}}(s) = 1 - \exp(-\gamma[H(s) - H_0]), \tag{3.5}$$

preserving the locations of the original minima. An adjustable tunneling parameter γ is introduced. The energy of the lowest minimum found so far is denoted with H_0. Random walkers moving on H_{STUN} seem to "tunnel" barriers of the original landscape. A fixed temperature is employed, hence, the whole process can be interpreted as an METROPOLIS procedure on the original landscape with an energy-dependent temperature. In fig. 3.1 the effect of the non-linear transformation (3.5) is shown for a one-dimensional toy landscape.

Stochastic Tunneling has been applied with great success to spin glasses, TSP problems and "low-autocorrelation binary sequences". Such

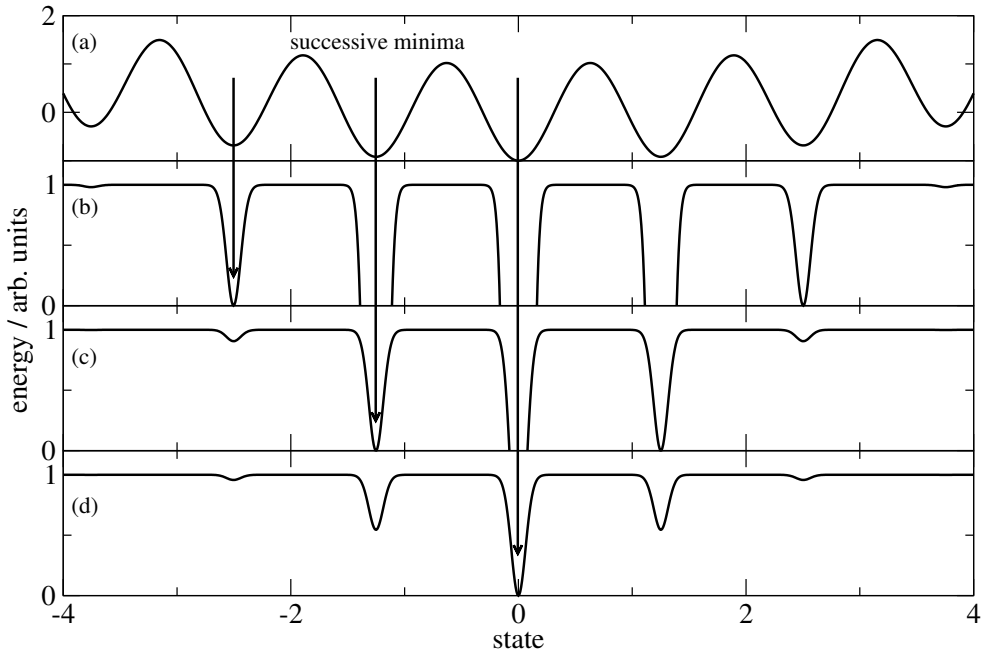

Figure 3.1: Schematic plot of Stochastic Tunneling. The original 1D energy landscape in (a) is explored by a random walker coming in from the left. The transformation due to better and better successive minima is shown in (b) - (d). The location of the minima is preserved; still unknown minima are enhanced. In (d) the ground state has been found. Then the entire landscape is mapped onto the interval $[0, 1]$.

sequences are equivalent to a one-dimensional spin-1/2 chain of length N with energy

$$E = \frac{1}{N} \sum_{k=1}^{N} \left[\sum_{j=1}^{N-k} s_j s_{j+k} \right]^2 . \tag{3.6}$$

3.3 Discrete Control Theory and Optimal Schedules

The question arises which schedule $T(t)$ gives the best optimization results. Before an argument about that can be done a way of measuring the performance of an optimization algorithm must be given. Assuming a total of S optimization steps made we describe the MARKOV chain of the

optimization process on the discrete state space with the master equation (2.36). The state probabilities of the last step are $\boldsymbol{p}(S)$. We define an objective function which depends linearly on that state probabilities,

$$o := \boldsymbol{E} \cdot \boldsymbol{p}(S), \tag{3.7}$$

which is just a scalar, and try to minimize it by selection of the proper sequence of transition matrices, i. e., the proper temperature schedule. The numbers in the vector $\boldsymbol{E}$ depend on the type of measure we are interested in. The most frequently used objectives are the following [31, 34, 35].

(O1) The final mean energy should be as small as possible.

(O2) The final probability of ending up in the ground state should be as large as possible.

(O3) The expected number of visits to the ground state should be as large as possible.

(O4) The probability of visiting the ground state during the optimization run should be as large as possible.

(O5) The mean final best-so-far (BSF) energy E_{bsf} [36, 37, 38] should be as small as possible. The best-so-far energy describes the lowest energy found during a random walk.

The first two objectives are easy cope with. If the mean energy has to be minimized, then $\boldsymbol{E}$ contains the state energies $H(\alpha)$; if we want to maximize the probability to be in the ground states, then all numbers are zero except the ones corresponding to the ground states, which are -1. The remaining three objectives require a more cumbersome description of the random walks to be performed. To describe them we introduce extended Markov chains.

3.3.1 An Extension of the Markov Chains – Absorbing States

In order to be able to determine objectives (O3) – (O5) we turn all states at or below an energy E into absorbing states [39]. The transition matrix is modified to be

$$\Gamma_{\beta\alpha;E}(t) = \begin{cases} \delta_{\beta\alpha} & \text{if } E \leq H(\alpha), \\ \Gamma_{\beta\alpha;E}(t) & \text{if } E > H(\alpha). \end{cases} \tag{3.8}$$

Here, $\delta_{\beta\alpha}$ denotes KRONECKER's delta. A random walker reaching a state with or below energy E is trapped at that state. The associated probability distribution

$$p_{\beta;E}(t) = \sum_{\beta\in\Omega} \Gamma_{\beta\alpha;E}(t) \cdot p_{\alpha;E}(t-1) \tag{3.9}$$

denotes the probability of being in state α of the modified chain after t steps. If $H(\alpha) > E$ it is the same as the probability of being in state α in the unmodified walk and *not* having visited any states with energy less than or equal to E before t. Therefore,

$$B_E(S) = \sum_{\beta:H(\beta)\leq E} p_{\beta;E}(S) \tag{3.10}$$

gives the probability of having visited a state with energy less than or equal to E up to time S.

Before we express objectives (O3) – (O5) we introduce a compact matrix notation for the master equation for our modified chains. Due to the finiteness of the state space under consideration we can sort all possible energies and label them E_r, $r = 0, \ldots, R$. The label $r = 0$, which we introduce for convenience only, denotes an arbitrary energy lower than the ground state energy E_1. We denote all probability vectors with $\boldsymbol{p}_{E_r}(t) = \{p_{\beta;E_r}(t)\}$. Their time development is again

$$\boldsymbol{p}_{E_r}(t) = \boldsymbol{\Gamma}_{E_r}(t) \cdot \boldsymbol{p}_{E_r}(t-1). \tag{3.11}$$

These $R+1$ matrix equations can be combined into one, written as

$$\boldsymbol{q}(t) = \begin{pmatrix} \boldsymbol{p}_{E_0}(t) \\ \boldsymbol{p}_{E_1}(t) \\ \vdots \\ \boldsymbol{p}_{E_R}(t) \end{pmatrix} \tag{3.12}$$

$$= \begin{pmatrix} \boldsymbol{\Gamma}_{E_0}(t) & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Gamma}_{E_1}(t) & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{\Gamma}_{E_R}(t) \end{pmatrix} \cdot \begin{pmatrix} \boldsymbol{p}_{E_0}(t-1) \\ \boldsymbol{p}_{E_1}(t-1) \\ \vdots \\ \boldsymbol{p}_{E_R}(t-1) \end{pmatrix} \tag{3.13}$$

$$= \hat{\boldsymbol{\Gamma}}(t) \cdot \boldsymbol{q}(t-1) \tag{3.14}$$

As we have $L = |\Omega|$ states and $R+1$ different energies we have now a linear system of $L(R+1)$ equations. The time development of the

unmodified walk is contained in the first L entries $q_\gamma(t), \gamma = 1, \dots L$ of $\boldsymbol{q}(t)$.

Now the distribution of the mean BSF energy $\langle E_{\mathrm{bsf}} \rangle$ can be expressed as follows. We calculate the probabilities

$$b_{E_r}(S) = B_{E_r}(S) - B_{E_{r-1}}(S), \quad r = 1, \dots, R \tag{3.15}$$

within this extended random walk formalism. We have $B_{E_0} = 0$, since no walker can reach a state with energy below the ground state energy, and

$$\begin{aligned}
\langle E_{\mathrm{bsf}}(S) \rangle &= \sum_{r=1}^{R} b_{E_r}(S) \cdot E_r && (3.16)\\
&= \sum_{r=1}^{R} E_r \left[B_{E_r}(S) - B_{E_{r-1}}(S) \right] && (3.17)\\
&= \sum_{r=1}^{R} E_r \left(\sum_{\substack{\beta: \\ H(\beta) \le E_r}} p_{\beta;E_r}(S) - \sum_{\substack{\beta: \\ H(\beta) \le E_{r-1}}} p_{\beta;E_{r-1}}(S) \right) && (3.18)\\
&= \sum_{r=1}^{R} E_r \left(\sum_{\substack{\beta: \\ H(\beta) \le E_r}} q_{Lr+\alpha}(S) - \sum_{\substack{\beta: \\ H(\beta) \le E_{r-1}}} q_{L(r-1)+\alpha}(S) \right). && (3.19)
\end{aligned}$$

We define an extended objective function $\tilde{o}$ to measure the performance of an optimization run which is described by the just introduced extended Markov chains,

$$\tilde{o} = \sum_{t=1}^{S} \tilde{\boldsymbol{E}}(t) \cdot \boldsymbol{q}(t) \to \min . \tag{3.20}$$

The scalar $\tilde{o}$ is a function of all vectors $\boldsymbol{q}(t)$. The sequence of vectors $\tilde{\boldsymbol{E}}(t)$, each an $L(R+1)$-tuple of numbers, measures the performance of the chain, and is in general arbitrarily selectable. As the $\boldsymbol{q}(t)$ depend on the sequence of transition matrices, the minimum of $\tilde{o}$ has to be taken over all possible sequences.

Due to the unmodified walk being contained in the extended dynamics employing absorbing states, objectives (O1) and (O2) are describable with an appropriate sequence $\tilde{\boldsymbol{E}}(t)$, too. But now also objectives (O3) – (O5) are measurable; the corresponding numbers $\tilde{E}_\gamma$ are:

(O1) (minimizing the mean final energy)

$\tilde{E}_\gamma(t) = 0$ for $t < S$, $\tilde{E}_\gamma(S) = H(\gamma)$ for $\gamma \leq L$, $\tilde{E}_\gamma(S) = 0$ for $\gamma \geq L$

(O2) (maximizing the final ground state probability)

$\tilde{E}_\gamma(t) = 0$ unless $t = S$, $\gamma \leq L$ and $H(\gamma) = E_1$, in which case $\tilde{E}_\gamma(S) = -1$

(O3) (maximizing the expected number of visits to the ground state)

$\tilde{E}_\gamma(t) = 0$ unless $\gamma \leq L$ and $H(\gamma) = E_1$, in which case $\tilde{E}_\gamma(t) = -1$

(O4) (maximizing the probability of visiting the ground state)

Maximize $b_{E_1}(S)$: $\tilde{E}_\gamma(t) = 0$ unless $t = S$ and $L < \gamma \leq 2L$, in which case $\tilde{E}_\gamma(S) = -1$

(O5) (minimizing the mean BSF energy)

$\tilde{E}_\gamma(t) = 0$ for $t < S$,

$\tilde{E}_{Lr+\alpha}(S) = 0$ for $r \in \{0, \ldots, R-1\}$, $\alpha \in \{1, \ldots, L\}$ and $H(\alpha) > E_r$,

$\tilde{E}_{Lr+\alpha}(S) = E_r - E_{r+1}$ for $r \in \{0, \ldots, R-1\}$, $\alpha \in \{1, \ldots, L\}$ and $H(\alpha) \leq E_r$,

$\tilde{E}_{LR+\alpha}(S) = E_R$ for $\alpha \in \{1, \ldots, L\}$

With these definitions statements about how good or bad an algorithm performs can easily be made. Although these do not form a complete set of possible measures, they cover the most often used.

The question remains how $\tilde{o}$ can be minimized technically. It is easy to see that testing all possible sequences of transition matrices $\hat{\mathbf{\Gamma}}$, i. e. enumerating the set of all sequences, is impossible due to the huge number of sequences. But it has been shown that the development of an iterative algorithm which successively improves a first "sequence guess" is possible.

3.3.2 Optimal Sequences of Transition Matrices

The task to minimize $\tilde{o}$ with respect to a measure (O1)-(O5) in the present context can be understood as a discrete control problem. We need to choose a control – a schedule $T(t)$ – leading to a minimal $\tilde{o}$, prescribing the optimal sequence of transition matrices for our stochastic optimization and giving the corresponding sequence $\boldsymbol{q}(t)$.

We have to minimize $\tilde{o}$. In the minimum the first variation has to be zero. Surely, the first variation of the objective function can only depend on the entries of $\boldsymbol{q}(t)$ due to the numbers E_γ being fixed,

$$\delta\tilde{o} = \sum_{t=1}^{S} \sum_{\gamma=1}^{L(R+1)} \frac{\mathrm{d}\tilde{o}}{\mathrm{d}q_\gamma(t)} \delta q_\gamma(t), \tag{3.21}$$

Furthermore, the vectors $\boldsymbol{q}(t)$ have to be the solution of the master equation (3.14), the transition matrices of which are given by the schedule. We introduce LAGRANGE parameters [31, 40] $\boldsymbol{\Lambda}(t)$, transforming the original objective $\tilde{o}$ into

$$\begin{aligned}
\tilde{o} &= \sum_{t=1}^{S} \tilde{\boldsymbol{E}}(t) \cdot \boldsymbol{q}(t) \\
&\quad + \sum_{t=0}^{S-1} [\boldsymbol{\Lambda}(t+1)]^{tr} \cdot \Big[\hat{\boldsymbol{\Gamma}}(T(t+1)) \cdot \boldsymbol{q}(t) - \boldsymbol{q}(t+1) \Big] && (3.22) \\
&= \sum_{t=1}^{S} \tilde{\boldsymbol{E}}(t) \cdot \boldsymbol{q}(t) \\
&\quad + \sum_{t=0}^{S-1} [\boldsymbol{\Lambda}(t+1)]^{tr} \cdot \hat{\boldsymbol{\Gamma}}(T(t+1)) \cdot \boldsymbol{q}(t) - \sum_{t=1}^{S} [\boldsymbol{\Lambda}(t)] \cdot \boldsymbol{q}(t) && (3.23) \\
&= \sum_{t=1}^{S} \Big(\tilde{\boldsymbol{E}}(t) - [\boldsymbol{\Lambda}(t)] \Big) \cdot \boldsymbol{q}(t) \\
&\quad + \sum_{t=0}^{S-1} [\boldsymbol{\Lambda}(t+1)]^{tr} \cdot \hat{\boldsymbol{\Gamma}}(T(t+1)) \cdot \boldsymbol{q}(t) && (3.24)
\end{aligned}$$

Our control is the schedule $T(t)$, which is also subject to variation. Therefore, the first variation of $\tilde{o}$ reads

$$\begin{aligned}
\delta\tilde{o} &= \sum_{t=1}^{S} \Big(\tilde{\boldsymbol{E}}(t) - \boldsymbol{\Lambda}(t) \Big) \cdot \delta\boldsymbol{q}(t) && (3.25) \\
&\quad + \sum_{t=1}^{S-1} [\boldsymbol{\Lambda}(t+1)]^{tr} \cdot \hat{\boldsymbol{\Gamma}}(T(t+1)) \cdot \delta\boldsymbol{q}(t) && (3.26) \\
&\quad + \sum_{t=0}^{S-1} \frac{\partial \Big([\boldsymbol{\Lambda}(t+1)]^{tr} \cdot \hat{\boldsymbol{\Gamma}}(T(t+1)) \cdot \boldsymbol{q}(t) \Big)}{\partial T(t+1)} \cdot \delta T(t+1) && (3.27)
\end{aligned}$$

Note that the starting distribution is fixed, $\delta \boldsymbol{q}(0) = 0$. As $\delta\tilde{o} = 0$ must hold for any arbitrary $\delta \boldsymbol{q}(t)$ we finally get

$$\boldsymbol{\Lambda}(S) = \tilde{\boldsymbol{E}}(S), \quad \boldsymbol{\Lambda}(t) = \tilde{\boldsymbol{E}}(t) + \left[\hat{\boldsymbol{\Gamma}}(T(t+1))\right]^{tr} \cdot \boldsymbol{\Lambda}(t+1), \quad (3.28)$$

and $[\boldsymbol{\Lambda}(t+1)]^{tr} \cdot \hat{\boldsymbol{\Gamma}}(T(t+1)) \cdot \boldsymbol{q}(t)$ must be in a minimum according to $T(t+1)$.

An iterative procedure to calculate the optimal schedule $T(t)$ for the unmodified chain has been developed [40] an shown to be convergent, but specialized for the case $\tilde{\boldsymbol{E}}(t) = \boldsymbol{0}$ unless $t = S$. It is easily extendable for the modified version of the walks. This algorithm starts with an arbitrary schedule, and goes as follows, with i denoting the iteration step:

1. Compute $\boldsymbol{q}^{i=0}(t+1) = \hat{\boldsymbol{\Gamma}}(T^{i=0}(t+1)) \cdot \boldsymbol{q}^{i=0}(t)$, $t = 0, \ldots, S-1$.

2. Compute $\boldsymbol{\Lambda}^i(t-1) = \tilde{\boldsymbol{E}}(t-1) + \left[\hat{\boldsymbol{\Gamma}}(T^i(t)\right]^{tr} \cdot \boldsymbol{\Lambda}^i(t)$, $t = S, \ldots, 2$. The Lagrange parameters get shifted by the fixed numbers $E_\gamma(t)$, compared with the original version of this algorithm.

3. Compute $T^{i+1}(t+1)$ such that $[\boldsymbol{\Lambda}^{i+1}(t+1)]^{tr} \cdot \hat{\boldsymbol{\Gamma}}^{i+1}(T(t+1)) \cdot \boldsymbol{q}^{i+1}(t)$ has a minimum, determine $\boldsymbol{q}^{i+1}(t+1) = \hat{\boldsymbol{\Gamma}}(T^{i+1}(t+1)) \cdot \boldsymbol{q}^{i+1}(t)$, $t = 0, \ldots, S-1$. Compute $\tilde{o}^{i+1}$ via (3.20).

4. Compare $\tilde{o}^{i+1}$ with the previous value $\tilde{o}^i$, check whether the difference is smaller than a chosen accuracy. If not, go back to 2.

We are now able to compute an optimal sequence of transition matrices depending on a control $T(t)$ for a given performance measure.

3.4 Tree Dynamics

The master equation (3.14) in combination with the algorithm given in sec. 3.3.2 is a powerful means to describe a stochastic optimization procedure. But due to the very large cardinality even of small optimization problems the calculation of the control $T(t)$ is not possible for the original corresponding state space. Therefore, to make theoretical investigations possible coarse grained models of state spaces have been developed.

It has been shown that many of the properties of a complex state space can be modeled by tree-like structures [41]. These consist of a set of nodes which are connected by edges. The nodes represent the local

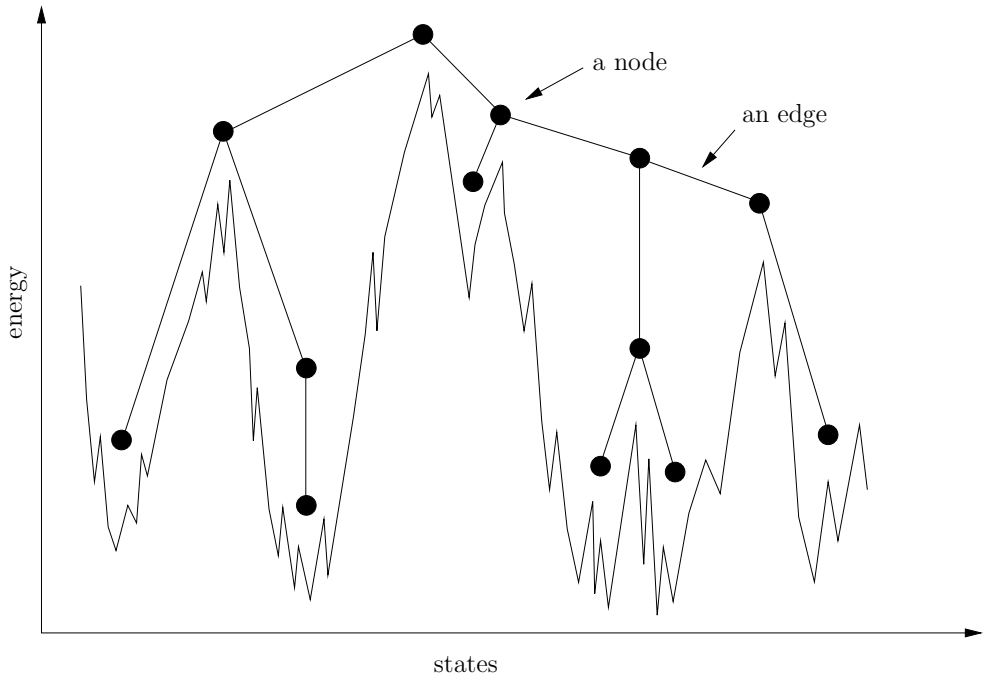

Figure 3.2: A landscape can be modeled by tree-like structures. Nodes lump microstates together and represent local minima and barriers. Edges connect nodes, representing possible paths in the original state space. Such trees can be extracted according to some predefined rules from the original state space, or set up in an empirical way.

minima as well as barriers between them, whereas the edges represent possible paths in the state space from one minima to another over the barriers. Random walkers – i. e., probability – populating a node are allowed to transit or hop to connected nodes. The transition rate is adjustable after every time step. In fig. 3.2 a possible tree representing the landscape of fig. 2.1 is depicted.

Such models mimic the dynamics of the original state space very well [42], but are numerically much cheaper. It is possible to create coarse grained, tree-like structure automatically from a given state space based on a set of rules [42, 43, 44], but here it suffices to employ even simpler models. We use them to model stochastic optimization processes, and to calculate optimal controls. Following [31] we employ hierarchical barrier systems. Such models exhibit an important property of the original state space. They offer energy barriers of different height. These can be considered the obstacles in the way of the random walker searching

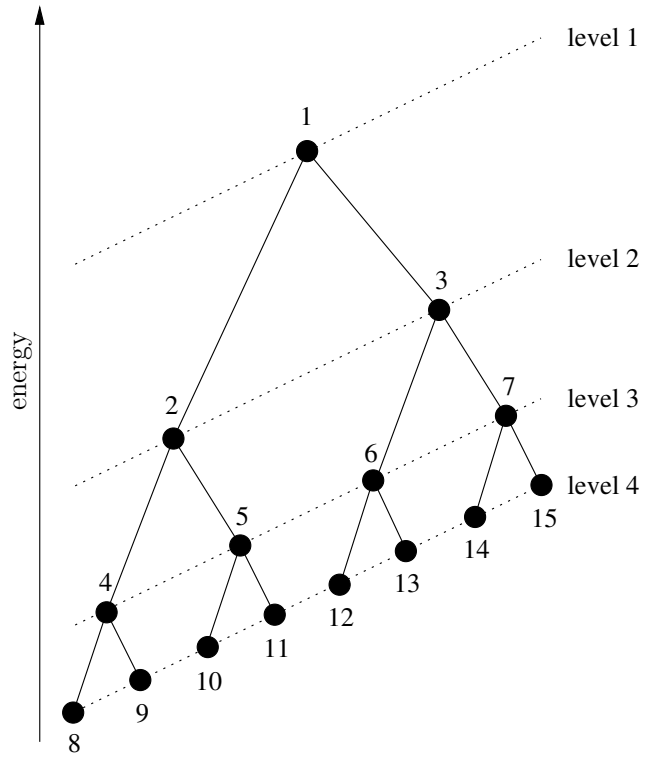

Figure 3.3: The tree-like structures can be further simplified to hierarchies, consisting of different barrier heights. The nodes are numbered in way which simplifies setting up the probabilities to select a neighboring node $\mathbf{\Pi}$. This tree has four levels.

for the ground state. In fig. 3.3 such a hierarchy of nodes is shown, together with a numbering which makes it easy to set up the corresponding probabilities $\Pi_{\beta\alpha}$.

Each node is situated on a so-called level. The top node on level $l = 1$ has label 1. The left connected node on the next deeper level has label $2l$, the right one $2l + 1$. Furthermore, every node α has an energy E_α. It lumps together many of the states of the original state space, therefore we define a degeneracy g_α for each. For many systems, an exponentially increasing degeneracy has been found [45], so we choose

$$g_\alpha := 2^{E_\alpha}. \tag{3.29}$$

An often used choice for $\Pi_{\beta\alpha}$ is

$$\Pi_{\beta\alpha} = c \cdot \begin{cases} 0 & \text{if } \beta \notin \mathcal{N}(\alpha), \\ g_\beta & \text{if } \beta \in \mathcal{N}(\alpha), \end{cases} \tag{3.30}$$

which we will also use here.

With this description we are able to calculate optimal schedules $T(t)$ for hierarchical binary trees of different node numbers for various values of the parameter q of the modified TSALLIS statistics (3.3). As we will see later, two values for q are the most interesting: $q \to 1$, as this denotes the original SA and is therefore of special interest, and $q \to -\infty$, as this denotes TA, which is within the framework developed so far the best possible strategy. A proof for this statement will be given later. Therefore, we will only deal with this two "extremes".

3.5 Optimal Schedules for Simulated Annealing and Threshold Accepting

To get a feeling for the shape of optimal schedules we have a look on some selected examples. For a tree of four levels optimal schedules for SA as well as TA and varying performance measures have been calculated. Following [31] the temperature interval $T \in [0, \infty]$ has been mapped onto $x \in [0, 1]$ by

$$x := \exp(-1/T) \tag{3.31}$$

with $x \to 0$ for $T \to 0$ and $x \to 1$ for $T \to \infty$. The possible range for x has been divided into 1000 parts; the minimization in the third step of the given algorithm to compute the optimal schedule was done by testing all 1001 possible values.

In fig. 3.4 the optimal schedule for maximizing the probability to be in the ground state after of 100000 steps has been calculated. In the case of TA a "hopping" between different temperatures can be observed. This is due to the nature of the process of calculating the optimal schedule: for TA there are many different optimal schedules, the one chosen by the algorithm simply depends on the initial guess. The optimal SA does not show this behavior.

For comparison, the same procedure has been performed for 1000000 steps in fig. 3.5. In principle, the schedules look the same, but here it

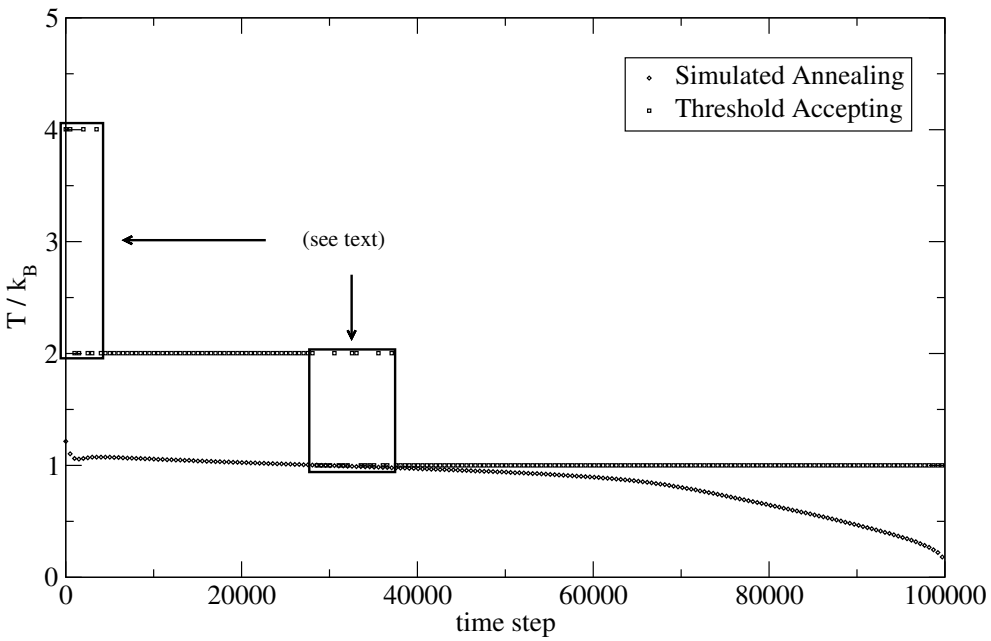

Figure 3.4: Optimal temperature schedules for Simulated Annealing and Threshold Accepting for a tree of four levels, maximizing the final probability to be in the ground state with 100000 steps. (For clarification only every 500th step has been plotted.) The hopping between different temperatures in the framed areas is due to the nature of the algorithm employed to calculate the optimal schedule. In the case of TA (and for long run times), there are many optimal algorithms; the algorithm simply finds one of them.

becomes evident that the optimal schedule for SA is a series of $1/\ln(T)$-like schedules. To see this, an appropriate fit to the SA data for steps $700000 - 900000$ has been done. The same – not shown here – can also be done for steps $100000 - 300000$. For infinite run times it can be shown that with such schedules the ground state can be found with certainty [46].

In all cases TA outperforms SA, i. e., the measured probabilities to be in the ground state at the end of the runs were larger in the case of TA than in the case of SA (see also [31]). The question whether it can be proved mathematically that TA offers highest performance within this class of optimization heuristics is answered in [34, 35].

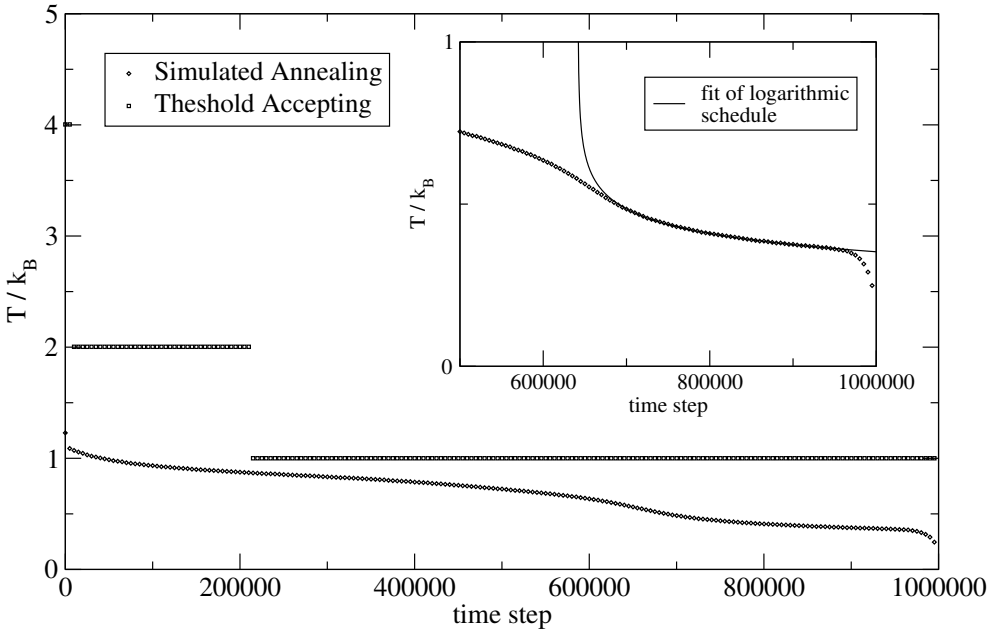

Figure 3.5: Optimal temperature schedules for Simulated Annealing and Threshold Accepting for a tree of four levels, maximizing the final probability to be in the ground state with 1000000 steps. (For clarification only every 5000th step has been plotted.) The hopping observed in the previous figure is present here, too, but not noticeable at this scale. The inset shows a fit of a $1/\ln(T)$-like schedule adapted to the SA data for times $700000 - 900000$. The same (not shown) can be done for the time span $100000 - 300000$; the optimal schedule is a chain of such truncated schedules (see also [47]).

3.6 Extremal Optimization as a Markov Process

Annealing-like optimization algorithms are in principle generic methods. Only two ingredients are needed: a move class and a time-dependent acceptance probability. These rather mild prerequisites are met in most of the optimization tasks. But surly, a generic method need not be the one with highest performance for a special problem. Therefore, many problem-specific algorithms have been developed, taking into account special properties of the underlying state space and energy function in question.

One of these algorithms is the *Extremal Optimization* (EO) heuristics of BOETTCHER and PERCUS. It is of special interest because it combines enhanced performance with a still more or less general stochastic approach to explore the state space. Before being concerned with open questions about the algorithm we will have a look on the general idea and the class of problems which can be solved with EO.

3.6.1 Basic Idea

In many of the combinatorial optimization problems the states offer an additional structure. They consist of many small parts which contribute to the total energy. Some of them contribute large amounts, whereas others contribute small amounts. For example, a traveling salesman's tour is a collection of short and long parts. We would consider the short parts of the total tour "well adapted" or "fit", and the long parts "unadapted" or "unfit". Another example is that of the state of a spin glass model. All spins experience a local field, given by the current orientations of the neighboring spins and the interaction constants. As all spins have only two possible orientations some spins will be aligned parallel to the local field, others anti-parallel. Hence, they will contribute small or large amounts of energy to the total energy of the state, respectively.

In the context of optimization it is clear that we are interested in states which consist only of "fit" parts. Those states are the ones with an all-in-all small, if not the smallest, energy. In order to take advantage of that additional feature BOETTCHER and PERCUS developed EO: changing the current state during an optimization process is done by only changing an *extremal* part of the state [48, 49, 50, 51].

Before the algorithm itself is outlined a few words about the definition of the fitness of a part of the solution must be given. First of all, the energy of the whole state i has to the sum of the contributions ϵ_k of the individual parts or degrees of freedom (DOF) k, $k = 1, \ldots, n$,

$$E_i = \sum_{k=1}^{n} \epsilon_k. \tag{3.32}$$

Each value ϵ_k might depend on the current state, i. e., it is related to the value of the other degrees of freedom (DOF). A DOF k is fit if ϵ_k is small, otherwise it is unfit. Therefore, we define the fitness λ_k to be

$$\lambda_k := -\epsilon_k. \tag{3.33}$$

For example, in the graph partitioning problem the fitness of a vertex of a subset could be defined as $\lambda_k = -b_k/2$, with b_k being the number of connections to vertices of the other subset. A spin s_i in the configuration S of a spin glass has a fitness

$$\lambda_{s_i} := B_{\text{local}}(S) \cdot s_i, \tag{3.34}$$

which is the negative of the potential energy of the spin due to the current local field $B_{\text{local}}(S)$ the spin experiences.

With this preparation the first version of EO went as follows:

1. Select an initial state.

2. For the current state, calculate the fitnesses λ_k. Create a ranking by sorting all DOF by their fitness in ascending order. DOF with low fitness have a low rank, those with high fitness have a high rank.

3. Select a new state so that the DOF with lowest rank *must* change. Accept this state *unconditionally*.

4. Measure the best-so-far energy. Iterate by going to step 2 until some stopping criterion is met.

Originally the idea of simply changing the worst part of the state was due to the BAK-SNEPPEN model of evolution [52]. In contrast to the idea behind evolutionary algorithms, namely "breeding the good", the BAK-SNEPPEN model formulates a co-evolutionary process by "eliminating the bad". Some specialties of the BAK-SNEPPEN model are also very attractive for a general purpose optimization algorithm (taken and adapted from [52, 53]):

- The system evolves into a self-organized critical state. Almost all species then have a much better than random fitness.

- Most species offer a good fitness even for long times. Unless they are connected to poorly adapted species, they do not go extinct.

- Perhaps most important is the observation that the system "retains a potential for large, hill-climbing fluctuations at any stage" and "the model accomplishes these features without any control parameters."

Indeed, the requirement to accept all proposed states unconditionally causes high fluctuations at every time. In contrast, a SA run shows equivalent fluctuations only at high temperatures. In this sense EO circumvents the problem of freezing observed in conventional SA. EO, due to the large fluctuations, easily climbs barriers, and probes many local minima.

The MARKOV chains of the original EO method are reducible, as some states might not be reachable by others due to the given transition probabilities. Another specialty of EO is that all diagonal elements of the corresponding transition matrix are zero, because a random walker is always forced to leave the current state in the next step.

But the procedure, implemented in this way, has a drawback. If, for some reason, the worst DOF can only be changed into one new value then the whole EO process might become a deterministic search. An example is the spin glass model with two possible values for a spin. Changing the worst spin can only be done in one way, which makes the chain of subsequent states predictable. This leads to "dead ends" in the sense of the search for the ground state, as the system might periodically come back to one and the same state.

3.6.2 Avoiding Dead Ends

In order to avoid the dead end effect, EO has been developed further in the following way. Instead of always selecting the least rank also higher ranks are selected to be changed. At each step a probability distribution over the ranks is employed. Originally, a distribution

$$d_k \sim k^{-\tau}, \;\; \tau > 0 \tag{3.35}$$

was used, introducing a single parameter τ. This parameter is unknown *a priori*, and has to be fine-tuned. An EO method employing such a distribution is called τ-EO [54, 55, 56, 57].

The choice of functional shape of this distribution is arbitrary, but a motivation is given in [54]. Selecting a power-law distribution secures that every regime of fitness is included in the further evolution, because d_k varies in a gradual scale-free manner over the ranks.

With such a distribution over the ranks the EO heuristics is able to produce irreducible chains. In this case any state is reachable from any other state, given an appropriate move class. But τ-EO need not be ergodic.

EO and τ-EO have been applied to a large testbed of problems, including spin glasses, graph bi-partitioning, traveling salesman problems, graph coloring and image alignment [58]. Where tested against it, τ-EO seems to outperform SA, even with fine-tuned schedules. Nevertheless, the question arises whether there exists an analytical proof that a special implementation of EO offering highest performance. Of course, there must be a parameter τ which does best, but is the power law distribution itself really a good choice?

3.7 A Provably Optimal Implementation

We have seen that in order to avoid a deterministic search for the ground state a probability distribution over the ranks has to be employed. Obviously, the performance of the whole EO process must depend on that distribution. Immediately the question arises whether there is a best one, and if so, how it looks like. Chances are that the originally chosen distribution $\sim k^{-\tau}$ might not be the optimal one.

Based on numerical studies similar to the ones performed in sec. 3.5 Franz et al. were able to realize a proof showing that indeed TA is the best possible algorithm to find ground states within the class of "annealing-like" methods [34, 35]. This result is especially interesting, as mathematically proved results are very rare among the publications in the field. Could this proof be generalized to cover even such a "unusual" dynamics like the one introduced by EO?

It is indeed possible, as we have shown in [59, 60]. Here, an arbitrary distribution $d(k)$ over the n ranks k is considered a vector $\boldsymbol{d}^t$ of dimension n. Of course, this vector is dependent on the time t. Each entry d_i^t, representing the probability to select and change the corresponding rank, can vary between 0 and 1. Furthermore, it seems reasonable to select low ranks with more probability than high ranks, so

$$1 \geq d_1^t \geq d_2^t \geq \ldots d_n^t \geq 0 \ \ \forall t. \tag{3.36}$$

As all vectors $\boldsymbol{d}^t$ represent a probability distribution we also have

$$1 = \sum_{i=1}^{n} d_i^t \ \ \forall t. \tag{3.37}$$

Consequently, the set of all admissible vectors I is defined by the linear inequalities (3.36) and the linear equality (3.37).

To find the extreme points or vertices $V = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n\}$ of I the inequalities must be set to equalities. They are exactly those vectors with an initial sequence of i entries equal to $1/i$, followed by the remaining $n - i$ entries equal to zero. Explicitly, the vertices are

$$\begin{aligned} \boldsymbol{v}_1 &= (1, 0, 0, \ldots, 0)^{tr} && (3.38)\\ \boldsymbol{v}_2 &= (1/2, 1/2, 0, 0, \ldots, 0)^{tr} && (3.39)\\ &\vdots \\ \boldsymbol{v}_i &= (1/i, 1/i, \ldots, 1/i, 0, 0, ...0)^{tr} && (3.40)\\ &\vdots \\ \boldsymbol{v}_n &= (1/n, 1/n, ...1/n)^{tr}. && (3.41) \end{aligned}$$

These vectors are linearly independent.

The set I is exactly the convex hull $C(V)$ of V,

$$C(V) = \left\{ \begin{array}{l} \sum_{i=1}^{n} a_i \boldsymbol{v}_i = a_1 \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + a_2 \begin{bmatrix} 1/2 \\ 1/2 \\ \vdots \\ 0 \end{bmatrix} + \ldots a_n \begin{bmatrix} 1/n \\ 1/n \\ \vdots \\ 1/n \end{bmatrix}; \\ \\ a_i \in [0, 1];\ \sum_{i=1}^{n} a_i = 1 \end{array} \right\}, \tag{3.42}$$

which is a simplex. To show this let us consider the l^{th} row of an element d^t of $C(V)$,

$$d_l^t = \sum_{i=l}^{n} a_i \frac{1}{i} = \sum_{i=l+1}^{n} a_i \frac{1}{i} + a_l \frac{1}{l} = d_{l+1}^t + a_l \frac{1}{l} \geq d_{l+1}^t. \tag{3.43}$$

so (3.36) is fulfilled. Summing up the rows of $C(V)$ results in

$$\sum_{l=1}^{n} d_l^t = \sum_{l=1}^{n} \sum_{i=l}^{n} a_i \frac{1}{i} = \sum_{l=1}^{n} l a_l \frac{1}{l} = \sum_{l=1}^{n} a_l = 1, \tag{3.44}$$

showing that (3.37) is also fulfilled. Thus $C(V) \subset I$.

Conversely, let us denote an arbitrary point with $\boldsymbol{p} \in I$. Since the vertices $\boldsymbol{v}_i$ are linearly independent, we can use them as a basis and write $\boldsymbol{p}$ as a linear combination

$$\boldsymbol{p} = \sum_{i=1}^{n} b_i \boldsymbol{v}_i. \tag{3.45}$$

For the l^{th} component p_l

$$p_l = \sum_{i=l}^{n} b_i \frac{1}{i} = p_{l+1} + b_l \frac{1}{l} \tag{3.46}$$

which implies by (3.36)

$$p_l \geq p_{l+1} \Rightarrow p_l - p_{l+1} = b_l \frac{1}{l} \geq 0 \Rightarrow b_l \geq 0. \tag{3.47}$$

Summing up all p_l, then using (3.37) yields

$$\sum_{l=1}^{n} p_l = \sum_{l=1}^{n} l b_l \frac{1}{l} = \sum_{l=1}^{n} b_l = 1 \Rightarrow b_l \leq 1. \tag{3.48}$$

So we have $b_l \geq 0$ and $b_l \leq 1$, therefore $\boldsymbol{p} \in C(V) \, \forall \boldsymbol{p} \in I$, hence, $I \subset C(V)$.

What does this mean for the optimization process given by τ-EO? Following the approach in [34], let us apply the Bellman principle of dynamic programming [61]. To find the best possible probability distribution over the ranks we have to work our way backwards, starting with the last step. The output of the last step $\boldsymbol{q}(S)$ is used to determine the optimality criterion (3.20).

In the last step S, we have to solve the optimization problem (3.20) for a given input $\boldsymbol{q}(t), 1 \leq t \leq S-1$. Using (3.14) we get

$$\tilde{o} = \sum_{t=1}^{S} \tilde{\boldsymbol{E}}(t) \cdot \hat{\boldsymbol{\Gamma}}(S) \cdot \boldsymbol{q}(t) \rightarrow \min . \tag{3.49}$$

Since the matrix entries of $\hat{\boldsymbol{\Gamma}}(S)$ depend linearly on the distribution $\boldsymbol{d}^S$ we have to find the minimum of a linear function on a simplex. Its minimum must be found on one of the vertices of the simplex. Therefore, for best performance we have to take a distribution which is represented by one of the elements of V. We denote the corresponding optimal transition probabilities by $\hat{\boldsymbol{\Gamma}}_{opt}(S)$.

Now we consider the second to last step $S-1$. For any given input $\boldsymbol{q}(t), 1 \leq t \leq S-2$ we have to solve an analogue optimization problem by defining a new objective

$$\tilde{\boldsymbol{E}}_2(t) := \tilde{\boldsymbol{E}}(t) \cdot \hat{\boldsymbol{\Gamma}}_{opt}(S) \;\; \forall t. \tag{3.50}$$

By the same arguments we find that the optimal probability distribution over the ranks for the second to last step is found on an element of V. All remaining time steps can be processed in that manner, resulting in an optimal distribution

$$\boldsymbol{d}^t \stackrel{!}{\in} V \ \ \forall t. \tag{3.51}$$

This proof shows that a uniform distribution over some of the "least fit" ranks gives the best implementation of EO. The resulting algorithm is called Fitness Threshold Accepting (FTA) [59], because in analogy to TA all moves triggered by selecting ranks which lie under some fitness threshold are selected with equal probability.

So far, the proof is based on the fundamental theorem of linear programming: a linear function defined on a simplex assumes its minimum at a vertex. The proof does not state that *all* optimal strategies are of the given form. In particular, other strategies may do equally well, but certainly not better.

This proof works with all of the measures (O1) – (O5) given above. Especially the original measure, namely minimizing the best-so-far energy, is covered. Nevertheless, it is not constructive: neither do we know how an optimal schedule looks like, nor how we should calculate it *a priori*.

3.8 Fitness Threshold Accepting for 1D Ising Spin Systems

To get a feeling what an optimal fitness threshold schedule looks like the methodology used for calculating the schedules for SA and TA has been employed. But here, coarse grained models, like the trees described above, which could be investigated with the EO dynamics are not present at the moment. But what can be done is to calculate exact EO transition matrices for small systems, which are still manageable on the computers we have access to, and use them for extremalizing selected performance measures.

To do so, one-dimensional ISING spin systems with random couplings $J_{ij} \in [-1, 1]$ have been used. The maximum number of spins used was 16; more spins were not manageable due to memory restrictions. Of course, sparse matrix techniques have been used throughout. In fig. 3.6 various optimal schedules for the fitness threshold have been calculated.

The probability to have seen the ground state was to be maximized, and different numbers of spins as well as total run times were employed.

The most apparent difference to the schedules we are used to in SA and TA are the jumps between very high and low thresholds from the very beginning of the optimization even to the very end. We would have expected some kind of monotonic decreasing schedule, but instead these very discontinuous ones have to be used. The numerical experiments performed did not show a systematic dependence of the schedule on the initial conditions. In this sense, the presented schedules are a "typical" example of optimal fitness thresholds for the systems which have been investigated. The FTA methodology with optimal schedules seems to be a kind of alternating sequence: probability is periodically allowed to flow into high energy regions of the state space and 'quenched' into low lying regions.

The development of the corresponding probabilities to have seen the ground state is shown in fig. 3.7. Of course, with growing run times the probability can be made larger and larger. The most exciting and interesting fact is that in spite of these unexpected FT schedules we can fit an exponential to the evolving probability to have seen the ground state. To see this in fig. 3.7 a function $\sim (1 - \exp(-\lambda t))$ has been fitted to the data for the 16 spin system. As can be seen this exponential behavior fits nicely. This is equivalent to the fact that the probability for having *not* seen the ground state vanishes exponentially. Therefore, it seems to be the case that despite the rather unusual dynamics of FTA the method is indeed a very good optimization algorithm.

This can be further underlined by investigating the mean best-so-far energy. But here, we are even more restricted in the size of the calculable system sizes: due to the extended transition matrices which have to be created the demand on available RAM grows very fast. For example, an ISING spin glass of 10 spins has originally a transition matrix $\mathbf{\Gamma}_{10} \in \mathbb{R}^{1024\times1024}$. If every entry is represented by a `double` of 8 bytes then we need 8MB for this matrix. Within the extended walk formalism we need to make successively every two states belonging to one and the same energy absorbing, leading to a total of $513 \cdot 8\text{MB} \approx 4\text{GB}$ RAM needed to hold the extended matrix. Furthermore, we need 10 of them, one for each of the ten possible distributions over the ranked spins.

Of course, these 40GB are reduced dramatically by using sparse matrix techniques, but the largest calculation possible at the moment[1] is exactly these 10 spins paired with 100 steps of optimization. Hence, in fig. 3.9 best possible fitness schedule for the 8 and the 10 spin system – minimizing the best-so-far energy – has been calculated only for the small number of 100 optimization steps. Also shown is the development of the best-so-far energy. Again it is possible to fit an exponential $\sim \exp(-\lambda t)$ to the data. In this case only the last 90 steps have been used for fitting, the first ten can be considered a warm-up phase.

[1] The `Riesen` cluster – 24 nodes with Intel Xeon processors, 4GB RAM per node – was used.

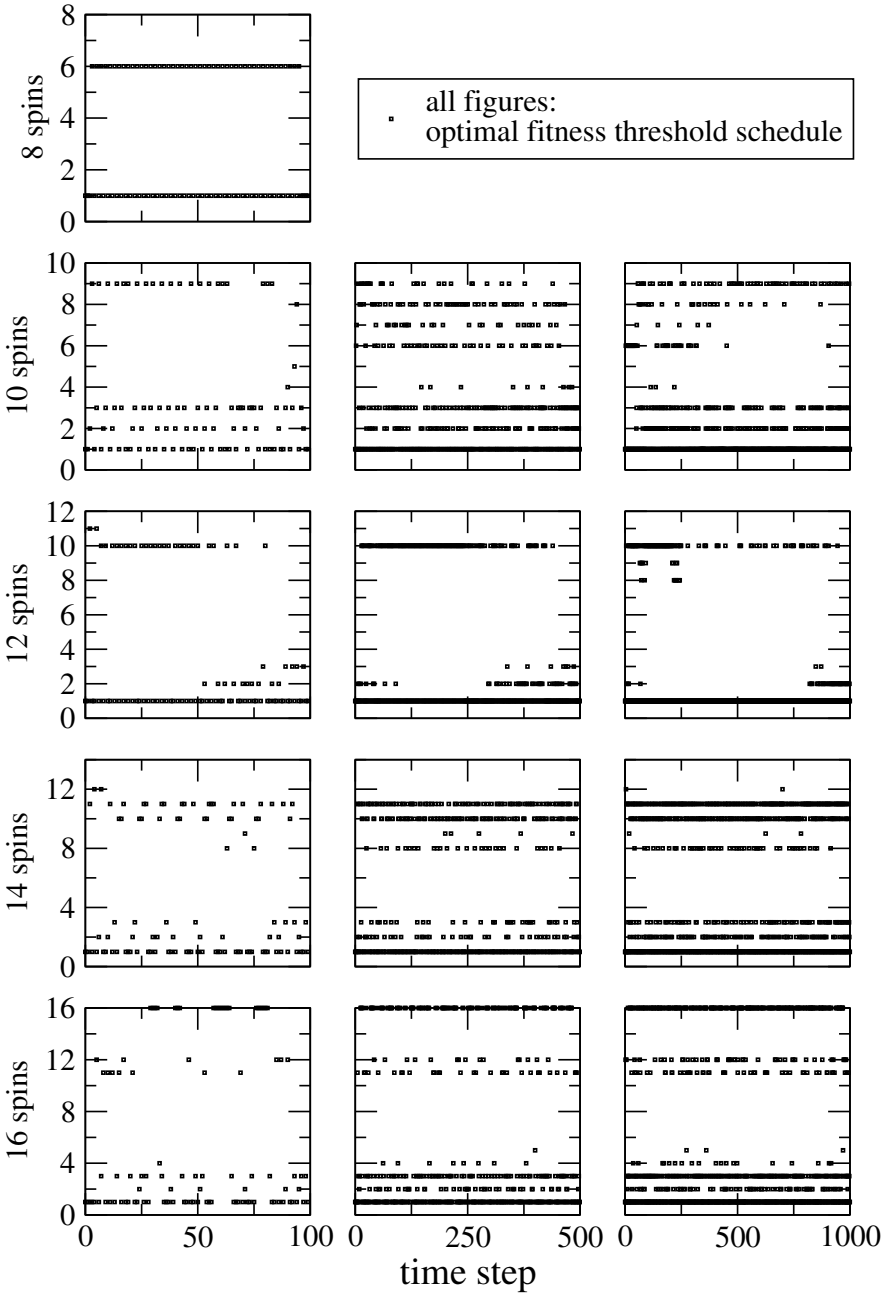

Figure 3.6: Optimal Fitness Threshold schedules for small 1D Ising spin glasses. For a system of 8, 10, 12, 14 and 16 spins the optimal schedule for maximizing the probability to have seen the ground state has been calculated. For all but the 8 spin system different run times have been investigated.

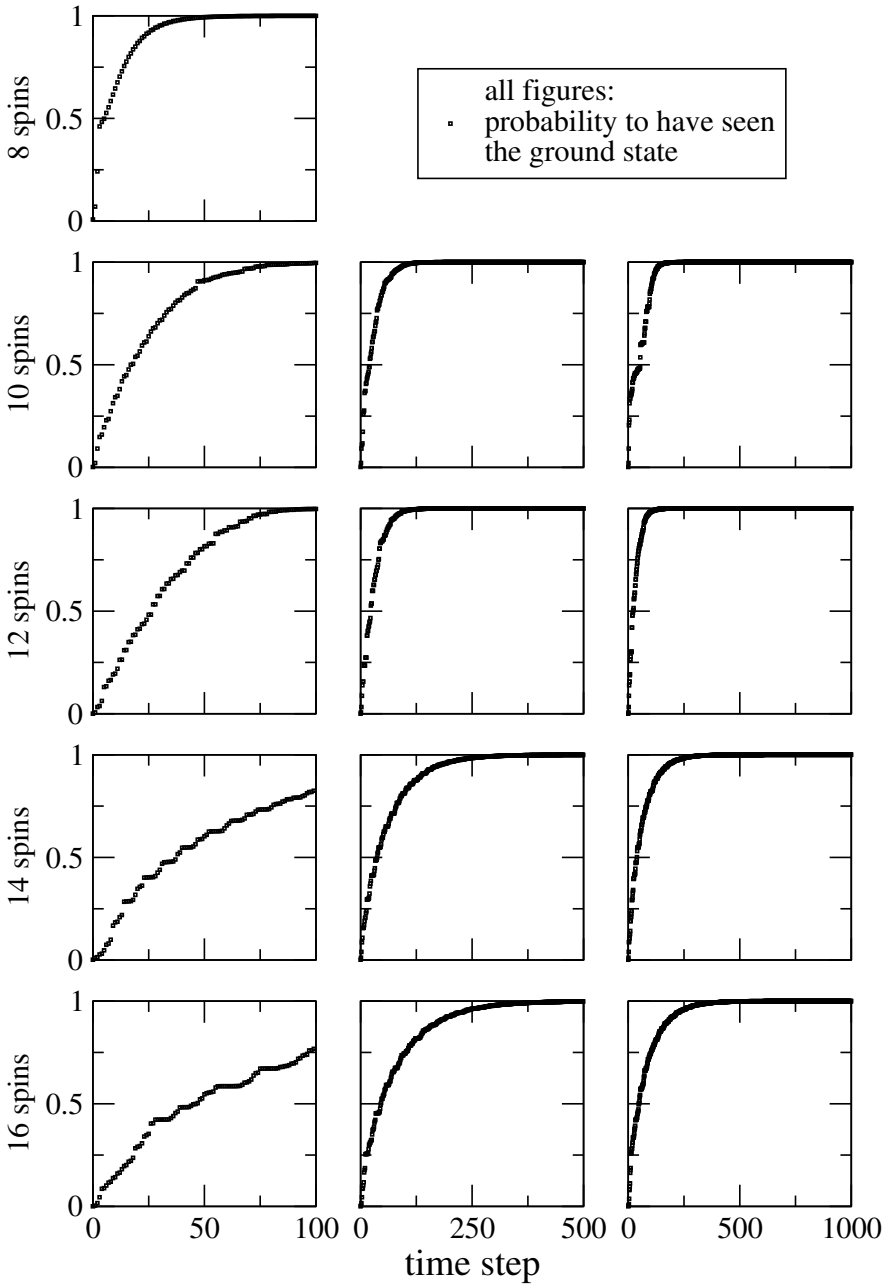

Figure 3.7: The corresponding probabilities to have seen the ground state of the ISING spin glasses of fig. 3.6. The probability for having not seen the ground state vanishes exponentially (see next figure).

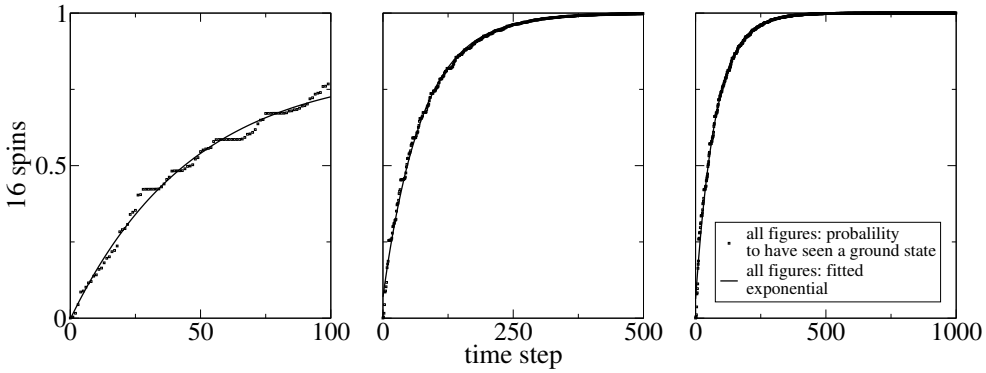

Figure 3.8: Fit of exponential functions to the data of the 16 spin system of fig. 3.7. The probability of not having seen the ground state becomes exponentially small with growing optimization time.

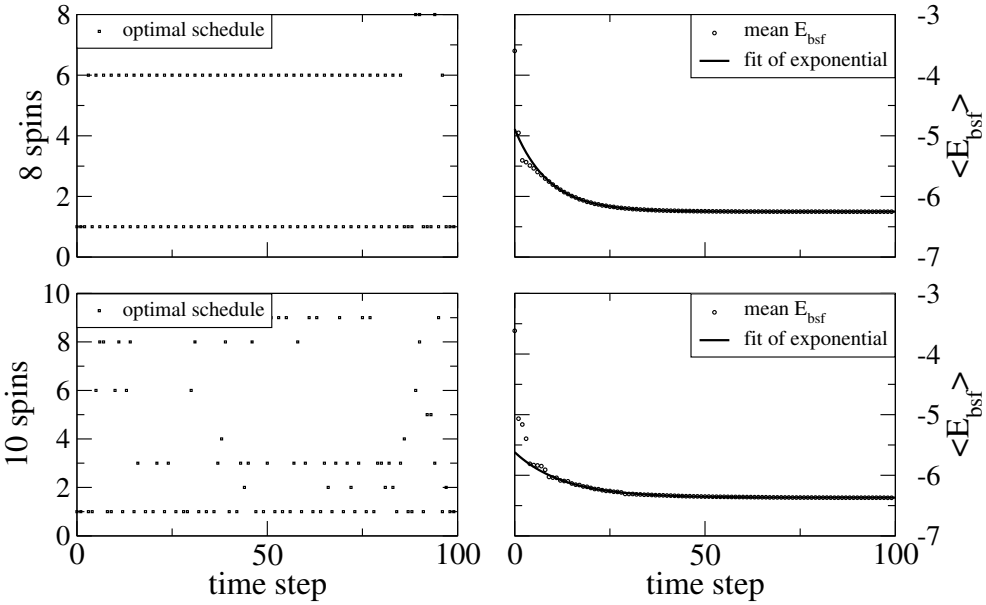

Figure 3.9: The optimal schedule for minimizing the mean best-so-far energy for the 8 and the 10 spin system, together with the development of this performance measure. Fitting an exponential is again possible, underlining the very good performance of FTA.

Towards Non-Ergodic Optimization

A deeper understanding of the superior performance of FTA (as well as τ-EO) and TA compared with SA might be attained by the following reasoning. This chain of arguments is mainly inspired by [55].

The SA algorithm, based on the METROPOLIS acceptance rules, is an ergodic process for a fixed temperature. Given that temperature the produced Markov chains always develop towards the limiting BOLTZMANN distribution. In particular, even the energetically highest states are populated with some probability unless $T = 0$. Of course, this probability vanishes exponentially with $T \to 0$, but unless cooling is not done very carefully the BOLTZMANN distribution cannot be realized. This is due to the inherently bumpy energy landscape, which originally was the reason to introduce stochastic optimization. Hence, even for long simulation times the ground state is not populated with the highest possible probability: given a temperature, the actual probability to be in the ground state will almost always be lower than that given by the BOLTZMANN weights.

Moreover, probability already gathered in low lying states might flow back into higher energy regions. So why should this be allowed? The answer is well-known, and a has many times been given as a reason for the success of SA: if such a reflux is explicitly allowed than a corresponding random walker is able to overcome barriers, and might be able to explore possibly lower lying minima behind these barriers. But this intuition only works under the assumption that there are always local minima behind the next barrier which lie lower indeed. This is simply *not true*. Instead, random walkers being currently in a good local minimum might step into regions of the state space which contain only higher lying minima. As a result we have to use a temperature schedule [46]

$$T(t) \geq \frac{c}{\ln(1+t)} \tag{3.52}$$

with some constant c. This schedule cools very slowly; nevertheless, for $t \to \infty$ it secures [46] that all of the probability flows into the global minimum. Corresponding random walkers would then be able to escape every local minimum.

But infinite run times would let stochastic optimization algorithms appear senseless – why should we introduce such sophisticated methods when we have *enough time* to enumerate the state space? Of course, we cannot optimize *infinitely* long. But as soon as the time we have at

our disposal is *finite* we seem to be forced to use at least non-ergodic algorithms: TA in the case of annealing-like methodologies, FTA in the EO framework. Based on this reasoning we should expect such types of algorithms to be those with highest performance. Hence, in the future such methodologies should be invented and further investigated.

3.9 Continuous Extremal Optimization

Let us close this chapter with an interesting generalization of the EO heuristics. Recently, it has been generalized to cover also continuous state spaces. Zhou et al. studied ground states of Lennard-Jones clusters with a corresponding adaption dubbed Continuous EO (CEO) [62].

A Lennard-Jones cluster is simply a number of mass points, interacting via the (parameter free) potential

$$V(r_{ij}) = \left(\frac{1}{r_{ij}}\right)^{12} - \left(\frac{1}{r_{ij}}\right)^{6}. \tag{3.53}$$

This empirical potential, dependent only on the distance r_{ij} of two mass points, reflects, e. g., the interaction of atoms of noble gases. The energy of a cluster of n particles at rest is simply the sum of all pairs of potential energies,

$$E_{LJ} = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} V(r_{ij}). \tag{3.54}$$

A state is an ordered set of all positions of all mass points, and a local minimum is defined in the known way.

If we have a local minimum of E_{LJ} we can again define a fitness for each mass point,

$$\lambda_i := -\frac{1}{2} \sum_{j=1, j\neq i}^{n} V(r_{ij}). \tag{3.55}$$

We consider mass points with a low interaction energy to be fit, whereas mass points of high energy are unfit. With this definition jumping from one local minimum to another one could be performed easily by an EO, τ-EO or FTA algorithm. But unfortunately, we are *not* able to define a corresponding move class! Of course, proposing a new state can simply

be done by varying the actual position of one or more mass points. But chances are very small that this new state is indeed a local minimum.

Otherwise, we have to be in a local minimum to succeed with an EO run based on such a definition of fitness. Simply performing EO with fitnesses defined by (3.55) *without* being in a local minimum would lead to a more or less unbiased random displacement of the most unfit mass points. It seems reasonable to assume that such an approach would offer only poor performance. To bypass this difficulty ZHOU et al. mix in a local search algorithm: after the proposal of a new state and its unconditioned acceptance they employ a limited-memory BFGS method [63]. This forces the system into the next nearby local minimum. Therefore, a general CEO scheme runs as follows:

1. Create an initial local minimum.
2. Set up fitnesses. Rank the DOF.
3. Based on the ranking do an EO step; i. e., select a DOF, propose and accept a move.
4. Use a local search algorithm to find the next nearest local minimum.
5. Measure E_{bsf}.
6. Iterate by going to step 2 as long as desired.

The authors successfully find global minima of clusters of up to 100 mass points. Nevertheless, they do not explain the move class they used. But this is the crucial point of the algorithm. If we define, e. g., a uniform random displacement between 0 and a radius r of an unfit mass points as our move class, we observe the following: Taking to small values of r results in a local search finding again and again the same local minimum, as the proposed random displacement does not push the system far away from it. Too large values of r push the system far away from the current local minimum, which leaves a lot of not investigated local minima. Hence, finding a good move class might not be trivial.

In fig. 3.10 and fig. 3.11 an FTA scheme was used to optimize the geometry of 10 and 20 mass points, respectively. The schedule was held fixed at 3 and 6, respectively, and the Conjugate Gradient scheme was used for the local search. The global minima found at -28.422532 and -77.177042, respectively, are consistent with [64]. In both cases FTA, after the first visit, hits the global minimum again and again. The move

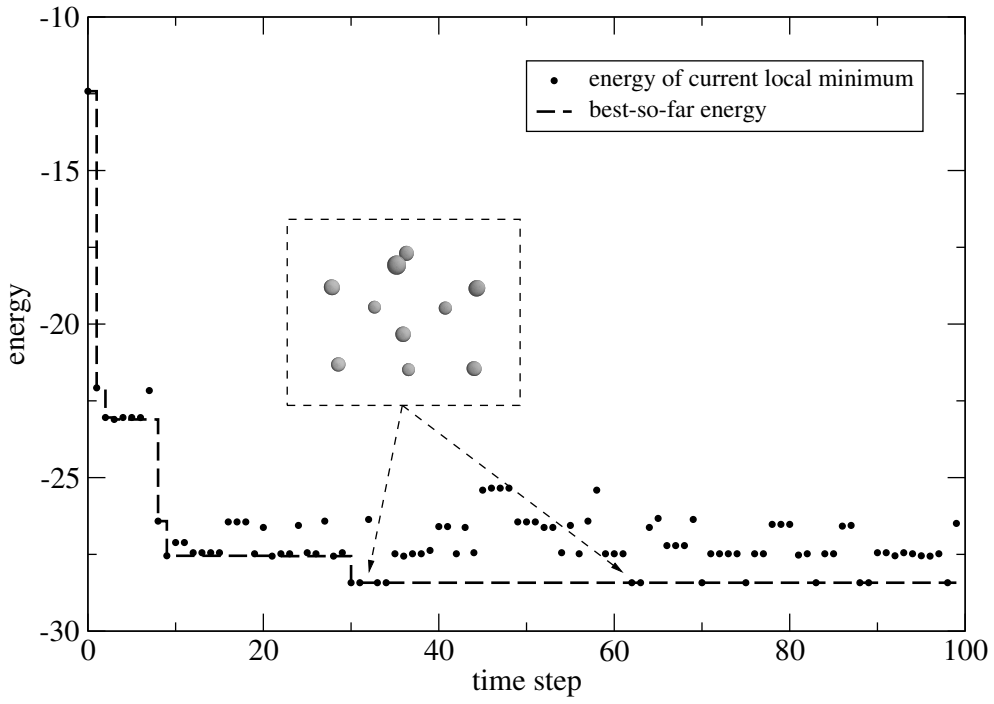

Figure 3.10: Continuous EO for a LENNARD-JONES cluster of 10 mass points. FTA was used, together with Conjugate Gradient and a move class described in the text. The inset shows the global minimum configuration of the cluster in real space.

class used here randomly displaces an unfit mass point, but on a sphere of radius 1.0 centered at its current position. This choice has two reasons: it forces the system into a far enough new state with a probably new nearby local minimum, and for small numbers of mass points this is *roughly* the distance in state space to the next local minimum. The two figures give an impression of how CEO converges to the minimum and the large fluctuations taking place during the optimization.

The most interesting theoretical point of this method is the fact that LJ clusters have – up to some energy threshold – a finite number of local energy minima [65]. If we assume that the local search algorithm finds the local minimum with certainty[2] then the whole dynamics can be considered to take place in a *discrete* state space. Therefore, the proof given in sec. 3.7 might be also applied here, leading to the insight that also here an FTA approach could be one with highest performance. Moreover,

[2]The CG method is such an algorithm.

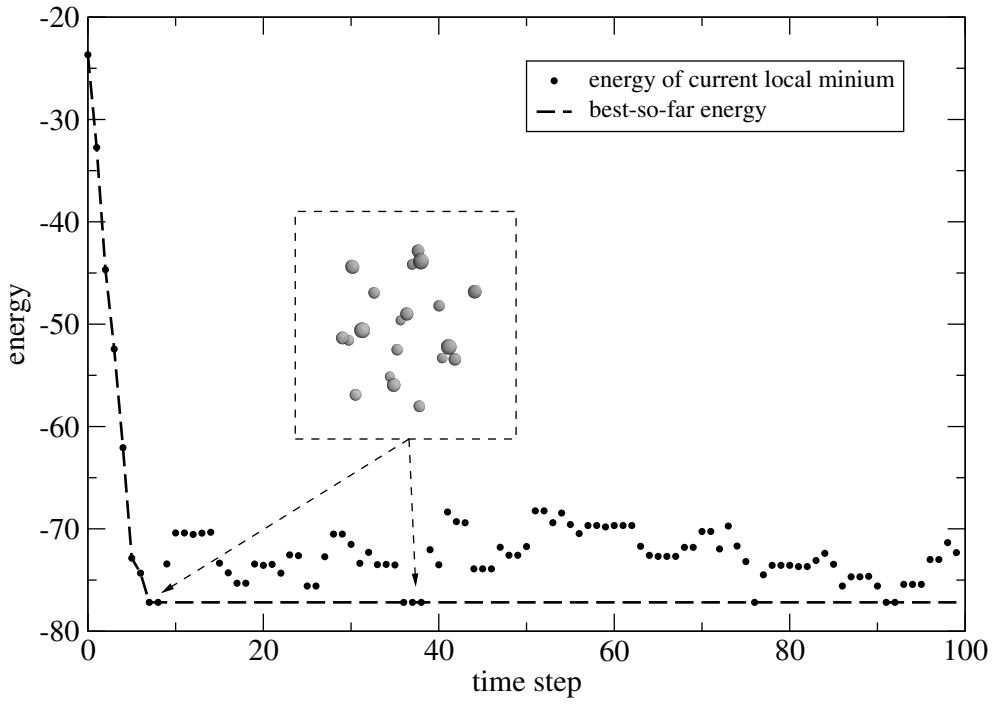

Figure 3.11: The same FTA scheme for a LENNARD-JONES cluster of 20 mass points. The inset again shows the global minimum configuration.

this description might be a promising starting point to generalize the proof to continuous state spaces.

Chapter 4

Algorithms to Calculate the Density of States

The calculation of the density of states is an important means to compute equilibrium thermodynamic quantities. Knowing it implies knowledge of the partition function and consequently all quantities which can be written as a functional of it.

But the density of states has also gained general interest in the field of optimization techniques. It can be used as a measure for the difficulty of optimization problems. Furthermore, extracting the density of states at low energies is closely related to finding ground states, i. e., solving optimization problems.

4.1 Thermodynamics and the Density of States

The probably most important variable in equilibrium thermodynamics is the canonical partition function $Z = Z(T)$. This is due to the fact that every equilibrium thermodynamic quantity – a macrostate X – can be written as a functional $X = X(Z(T))$ [66]. These functionals are simply weighted means over the states s of the system in question, and the weights are given by $\exp(-\beta H(s))$. Here, $\beta = 1/(k_B T)$, and $H(s)$ is the Hamiltonian, or energy, of the state s.

The thermodynamics of the system is considered solved if $Z(T)$ is known. We have

$$Z(\beta) := \sum_{s \in \Omega} \exp(-\beta H(s)). \tag{4.1}$$

Here, Ω denotes all states of the system under consideration. Of course, the $\sum$ sign can only hold in discrete state spaces, e. g. the classical combinatorial systems we investigate here. Otherwise, we have to integrate over the state space.

As an example, consider the calculation the specific heat $c(T)$, defined as the first derivative of the inner energy $U(T)$. We have

$$c(T) = \frac{\mathrm{d}}{\mathrm{d}T}U(T) \quad \text{and} \quad U(\beta) = \langle H \rangle = -\frac{\mathrm{d}}{\mathrm{d}\beta}\ln Z(\beta). \tag{4.2}$$

Therefore, $c(T)$ reads

$$\begin{aligned} c(T) &= \frac{1}{k_B T^2}\frac{\mathrm{d}}{\mathrm{d}\beta}\left(\frac{1}{Z(\beta)}\frac{\mathrm{d}}{\mathrm{d}\beta}Z(\beta)\right) && (4.3)\\ &= \frac{1}{k_B T^2}\left(-\frac{1}{[Z(\beta)]^2}\left[\frac{\mathrm{d}}{\mathrm{d}\beta}Z(\beta)\right]^2 + \frac{1}{Z(\beta)}\frac{\mathrm{d}^2}{\mathrm{d}\beta^2}Z(\beta)\right) && (4.4)\\ &= \frac{1}{k_B T^2}\left(-\langle H\rangle^2 + \left\langle H^2\right\rangle\right). && (4.5) \end{aligned}$$

This short recall outlines how equilibrium thermodynamic quantities are connected to the partition function by derivations with respect to β.

The question is how to sample the partition function and related quantities accurately. One of the most commonly used strategies is importance sampling [67]. By performing a Markov chain like random walk with transition probabilities given by Metropolis et al. [24] it is possible to extract a representative subset of the set of states the system is in at a certain temperature. Averages of quantities like the inner energy can than be calculated as averages over the chain. They tend to the equilibrium value with a growing number of steps.

But this strategy has some well known drawbacks. Firstly, in order to study equilibrium thermodynamics only "equilibrated" subsets of the original state space must be used. But due to the typical rough energy landscape of the systems in question very long run times are needed to realize such a subset; random walks tend to get stuck in local minima and cannot explore the whole state space easily. Secondly, especially at temperatures near to or at those where the system undergoes phase transitions the Metropolis sampling needs an overwhelming number of steps to give an "equilibrated" value of the quantity in question. This so called critical slowing down [68] might make this approach even unusable at all[1]. Thirdly, studying temperature dependencies, of course,

[1] A nice web page offering a JAVA applet dedicated to this effect for 2D Ising models can be found at [69].

requires the sampling of the partition function to be done at a multitude of different temperatures.

There are many more or less system specific solutions, e. g., for the case of ISING ferro-magnets the cluster flip algorithm [68]. But there is another very general approach for circumventing this problem. We can rewrite (4.1) in the following way. Firstly, we classify every microstate s by its energy $E = H(s)$. Secondly, for every energy the system can have we count all states with that energy to give a number $g = g(E)$. This function $g(E)$ is called the density of states (DOS), or degeneracy in quantum mechanics: it gives information about how dense in energy the microstates are situated.

The summation over the states s can then be transformed into a summation over the energies,

$$Z(T) = \sum_{s \in \Omega} \exp(-\beta H(s)) = \sum_{E_G}^{E_{AG}} g(E) \cdot \exp(-\beta E). \tag{4.6}$$

The smallest E, represented by the ground states, is denoted with E_G; the largest, realized by the anti ground states, with E_{AG}. These values are always finite in discrete and finite state spaces.

It is clear that if the DOS would be known, the partition function would be computable *at every temperature* – leading to particular low numerical load at critical temperatures. This is one of the reasons we are interested in $g(E)$. But there is another one. The DOS can be seen as measure of "how difficult" an optimization problem is [70]. If $g(E)$ is large even at E_G – i. e. the system has a lot of ground states compared to the number of non-ground states – we expect few difficulties to find one of them. But if $g(E)$ is small at E_G we expect a lot of computational effort: we will have to make our way through a lot of non-ground states until we find a ground state. In this sense $g(E)$ might even allow a decision about which type of optimization algorithm we should use: the former case would suggest a quench or gradient based scheme, the latter a more sophisticated technique like SA or EO. But how to get $g(E)$?

There have been, of course, efforts to calculate the density of states exactly, but the majority of these methods is bound to specific models. For instance, the method of BEALE [71] is only applicable to $n \times n$ ISING models with ferro-magnetic interactions; the method applied in [72] and [73] is only suitable for two-dimensional $n \times n$ $\pm J$ spin glasses with periodic boundary conditions. As there is still a "general purpose",

"polynomial-in-time" algorithm missing, Monte Carlo methods are still widely used and systematically studied.

In the following we will see how the development of histogram based, efficient algorithms to calculate $g(E)$ evolved. Besides a short description of some of methods developed and used up to now we will have a detailed look on the WANG-LANDAU sampling scheme. This scheme serves as some kind of "industry standard" nowadays. Matrix based methods follow; they can be seen as an enhanced version of histogram methods.

4.2 Reweighting and Histogram Methods

As complications with the traditional methods especially arise in studies of phase transitions, the ideas to overcome them have been developed in exactly this context. But in spite of the density of states being such a powerful means to "solve" thermodynamic models, an estimation of this quantity has not been the main goal of Monte Carlo simulations for a long time. Instead, there have been efforts to improve the information which could be gained by traditional methods. Some selected examples are outlined in the following.

Histogram Reweighting

One of these methods is the histogram reweighting algorithm, introduced by FERRENBERG and SWENDSEN [74, 75]. It aims to achieve fewer computational effort for scanning a temperature region near a phase transition, compared with a multiple partition function sampling.

For a quantity $X = X(s)$ we have

$$\langle X(\beta)\rangle = \frac{1}{Z(\beta)} \sum_{s\in\Omega} X(s) \cdot \exp(-\beta H(s)). \tag{4.7}$$

If we sample a subset of configurations $\{m\} = M \subset \Omega$ with probability $p_m = \exp(-\beta H(m))$, then

$$\langle X(\beta)\rangle \approx \frac{1}{|M|} \sum_{m\in M} X(m). \tag{4.8}$$

Information for a nearby temperature β' can be obtained by resampling the already sampled configurations M according to

$$\langle X(\beta')\rangle \approx \frac{\sum_{m\in M} X(m) \exp(-\beta' H(m))/\exp(-\beta H(m))}{\sum_{m\in M} \exp(-\beta' H(m))/\exp(-\beta H(m))} \tag{4.9}$$

According to [74, 76] a histogram $\mathcal{H}(E)$ can be stored which counts the number of occurrences of energy $E = H(m)$ during the simulation time. Another histogram holds the averages $\langle X(E)\rangle$,

$$\langle X(E)\rangle = \frac{1}{\mathcal{H}(E)} \sum_{m:H(m)=E} X(m). \tag{4.10}$$

With these histograms, the value $\langle X(\beta')\rangle$ can be calculated as a sum over the energies,

$$\langle X(\beta')\rangle = \frac{\sum_E \mathcal{H}(E)\,\langle X(E)\rangle \exp(-(\beta'-\beta)E)}{\sum_E \mathcal{H}(E)\exp(-(\beta'-\beta)E)} \tag{4.11}$$

Hence, as long as β' is close to β we can obtain information at β' from a run at temperature β by simply reweighting the simulation data. The method has been shown to work well for the second order phase transition of the two-dimensional Ising ferro-magnet, using a β near the critical temperature.

If β' is not close to β then the errors introduced by the reweighting become large. To broaden the accessible temperature region a little improvement can be made by using simulation data from different temperatures. An appropriate choice of the temperatures allows to create histograms which provide data useful for an overlapping of neighboring temperature regions.

Simulated and Parallel Tempering

In order to destroy the effect of random walks getting stuck in local minima, Marinari and Parisi and Geyer and Thompson proposed a technique dubbed simulated tempering (ST) [77, 78]. To simulate at temperature β a family of probability distributions $\boldsymbol{\pi} = \{\pi_i(s) \sim \exp(-\beta_i H(s)), i = 0,\ldots,n\}$ is constructed first, with $\beta_0 = \beta$ and $\beta_i > \beta_{i+1}$. Instead of simulating only at β_0, which corresponds to a unmodified random walk with Metropolis acceptance rates, the system is also allowed to transit to temperatures $\beta_{i\pm1}$ from the actual temperature β_i at each step. This is formalized by introducing an extended distribution $\pi_{st} \sim c_i \exp(-\beta_i H(s))$, defined on the augmented space $\Omega \times \mathcal{I}, \mathcal{I} = \{0,\ldots,n\}$. The c_i are tunable parameters which are subject to some pilot studies.

Of course, only the fraction of moves at β_0 is of interest for calculating averages subsequently, but the frequent visits of the system to the

lower β_i help to escape from local minima. The probability to change the temperature is only non-negligible if the difference between two temperatures is not too large. Otherwise, temperatures which differ only slightly will not be very helpful in crossing barriers.

A kind of twist of ST is the parallel tempering method (PT), proposed by GEYER [79] and later reinvented by HUKUSHIMA and NEMOTO [80]. Instead of augmenting Ω to $\Omega \times \mathcal{I}$, this method deals with the product space $\Omega_1 \times \cdots \times \Omega_n$ of n identical copies of Ω. With the vector of states $\boldsymbol{s} = \{s_1, \ldots, s_n\}$ a joint probability distribution π_{pt} can be defined on the product space, $\pi_{pt}(\boldsymbol{s}) \sim \prod_{i=1}^{n} \pi_i(s)$.

Chains of states are created in parallel in every copy of the original state space, but additional "index swaps" are possible. That means, with some probability the entries s_i and s_{i+1} of $\boldsymbol{s}$ are swapped at simulation step. Hence, states sampled at higher temperatures are able to become part of the chains sampled at lower temperatures, again helping to cross barriers.

Multicanonical and 1/k sampling

So far, the canonical weighting factor $\exp(-\beta H(s))$ for a microstate s has been used extensively in simulating a complex system. This factor can be derived for canonical ensembles.[2] But what about different weighting functions? Might changing the weights solve some of the described problems?

From a physical point of view such a change implies changing or generalizing the ensemble conditions. So it might be hoped that an investigation of a "proper generalized" ensemble can be performed easier, compared with the original one. For example, a probability distribution over the microstates which produces a uniform distribution of the sampled energies $E = H(s)$ or entropies $S(E) = \ln g(E)$ could be used.

In the canonical ensemble the probability to be in an arbitrary state of energy E in equilibrium is

$$\pi(E, \beta) = \frac{1}{Z(\beta)} g(E) \exp(-\beta E). \tag{4.12}$$

In order to flatten this distribution the multicanonical ensemble method was developed by BERG and NEUHAUS [81, 82]. They employ a modified

[2] This refers to sets of systems with a constant number of particles, volume and temperature.

distribution

$$\pi_{mu}(s) \sim \exp(-S(H(s)) \quad \text{with } S(H(s)) = \ln g(H(s)). \tag{4.13}$$

for sampling the microstates, using the microcanonical entropy $S(H(s))$. Now, in the ensemble with these weighting factors the probability to be in a state of energy E is simply

$$\pi'_{mu}(E) \sim c \quad \text{for all E.} \tag{4.14}$$

with some constant c. Central to the method is the attempt to obtain an accurate estimate of the density of states. This is again realized by the use of energy histograms; moves are performed due to the Metropolis acceptance rates. Details can be found, e. g., in [83]. In addition, a method dubbed entropic sampling has been proposed independently [84]. The method has also been applied to LENNARD-JONES glasses [85]

In the $1/k$ ensemble method of HESSELBO and STINCHCOMBE [86] a distribution is produced so that the entropy is uniform. Then, the probability to be in a state of energy E is

$$\pi'_{1/k}(E) \sim \frac{\mathrm{d} \ln g(E)}{\mathrm{d}E}. \tag{4.15}$$

The Wang-Landau sampling scheme

The WANG-LANDAU sampling scheme (WL) [87, 88] exploits the idea of creating a flat histogram to a very high degree. Setting up a random walk in energy space can be identified with the movement on an energy histogram, see fig. 4.1. Each time the random walker hits an energy level the entry of the corresponding histogram is increased by one. Denoting the probability to be in an arbitrary state of energy E with $\pi(E)$,

$$\pi(E) = \sum_{\alpha:H(\alpha)=E} p_\alpha = g(E) \cdot \sum_{\alpha:H(\alpha)=E} p_\alpha / g(E) \tag{4.16}$$

holds. Hence, sampling a state with probability $p_\alpha \sim 1/g(H(\alpha))$ for all possible energies would result in a constant $\pi(E)$ for *each* energy E,

$$\pi(E) = g(E) \cdot \sum_{\alpha:H(\alpha)=E} p_\alpha / g(E) \sim g(E) \cdot \frac{g(E)}{[g(E)]^2} = c \quad \text{for all } E. \tag{4.17}$$

The random walk would produce a "flat" histogram. As RATHORE and DE PABLO point out [89], WL has some similarities to the multicanonical method (see also [90]).

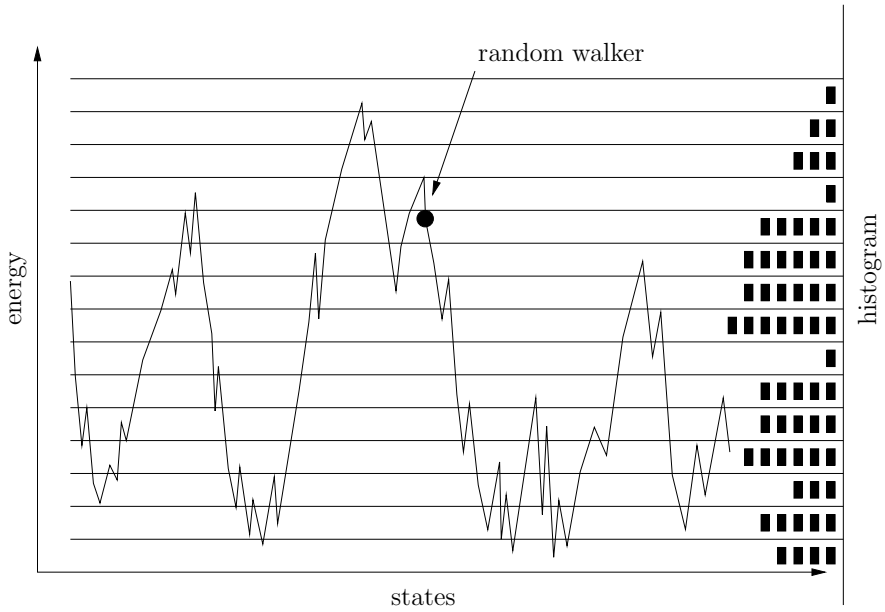

Figure 4.1: A random walk in the state space of a system hits different energies from move to move. Hence, it can be identified with a walk in a corresponding energy histogram. Each time an energy i is visited the histogram entry $h(i)$ is increased by one.

To sample the states in such a way we would like to have some rule how to make moves from a microstate β to another α: we need transition rates Γ_{ji} that secure the correct $1/g$ sampling of the microstates. Let us assume that detailed balance is valid,

$$\Gamma_{\beta\alpha} \cdot p_\alpha = \Gamma_{\alpha\beta} \cdot p_\beta \tag{4.18}$$

Then we have

$$\frac{\Gamma_{\alpha\beta}}{\Gamma_{\beta\alpha}} = \frac{\Pi_{\alpha\beta} \cdot P_{\alpha\beta}}{\Pi_{\beta\alpha} \cdot P_{\beta\alpha}} = \frac{p_\alpha}{p_\beta} = \frac{1/g(H(\alpha))}{1/g(H(\beta))}. \tag{4.19}$$

The neighborhood relationship $\Pi_{\beta\alpha}$ is fixed. In the following we assume $\Pi_{\beta\alpha} = \Pi_{\alpha\beta}$.

As this equation only determines the ratio of the acceptance probabilities between microstates there is – in principle – a wide variety to choose from. For instance, the rates

$$P_{\beta\alpha} = 1/g(H(\beta)) \tag{4.20}$$

or the rates

$$P_{\beta\alpha} = \frac{g(H(\alpha))}{N} \tag{4.21}$$

could be selected. But due to $g(H(\alpha))/N \ll 1$ for most of the microstates a trial move would very often be rejected, leading to a random walk sticking to one and the same energy level for large numbers of trials. This is extremely undesirable in a walk intended to hit a large number of different energies with relatively few steps. Hence, a slightly better choice would be to use the maximum density of states

$$g_{max} = \max_E[g(E)] \tag{4.22}$$

instead of the factor N.

Wang and Landau suggested transition rates [87, 88] of the form

$$P^{WL}_{\beta\alpha} = \min\left[\frac{g(H(\alpha))}{g(H(\beta))}, 1\right] \tag{4.23}$$

which also have the property (4.19), if $\Pi_{\beta\alpha} = \Pi_{\alpha\beta}$,

$$\frac{\Gamma^{WL}_{\alpha\beta}}{\Gamma^{WL}_{\beta\alpha}} = \frac{\min\left[\frac{g(H(\beta))}{g(H(\alpha))}, 1\right]}{\min\left[\frac{g(H(\alpha))}{g(H(\beta))}, 1\right]} = \frac{g(H(\beta))}{g(H(\alpha))}. \tag{4.24}$$

But here, substantially more trials are accepted compared to the previous transitions rates: we have an overall higher acceptance probability as in the case $g(H(\beta)) > g(H(\alpha))$

$$\frac{g(H(\alpha))}{g(H(\beta))} \geq \frac{g(H(\alpha))}{g_{max}} > \frac{g(H(\alpha))}{N} \text{ and } \frac{g(H(\alpha))}{g(H(\beta))} \geq \frac{1}{g(H(\beta))} \tag{4.25}$$

is valid, and a large fraction of proposed moves is accepted with certainty.

As an illustration we will have a look on a simple test system. The model consists of $n = 10$ spins s_i, hence $N = 2^{10}$ states α, with a long-range interaction

$$H(\alpha) = \frac{1}{n}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} s_i s_j. \tag{4.26}$$

Due to this special interaction, we can calculate the density of states by successively choosing $0, 1, 2, \ldots, 10$ spins pointing up, the rest down. As

number of states	energy of a state
$\binom{10}{0} + \binom{10}{10} = 2$	9/2
$\binom{10}{1} + \binom{10}{9} = 20$	27/10
$\binom{10}{2} + \binom{10}{8} = 90$	13/10
$\binom{10}{3} + \binom{10}{7} = 240$	3/10
$\binom{10}{4} + \binom{10}{6} = 420$	-3/10
$\binom{10}{5} = 252$	-1/2

Figure 4.2: States per energy for the long-range ISING model.

as state with i spins pointing up has an energy equal to a state with $10-i$ spins down, we get the densities given in fig. 4.2. With these values the transition matrices for the WL rates and the three non-WL rates can be calculated.

For fig. 4.3 the four different transition matrices have been calculated for this system. As the performance measure the standard deviation of the energy histograms produced has been chosen. Due to the initial distribution evolving towards the limiting one, which produces a perfect flat histogram, the standard deviation must approach zero.

The plot shows the superior performance of the WL acceptance rule for our simple test system. All standard deviations vanish exponentially, but quickest with the WL rates. Immediately the question arises if these rules are the best possible ones generally. It turns out that this question cannot be fully answered yet, but some ideas to start with can be developed.

Let us assume that the stationary distribution $\boldsymbol{p}^*$ corresponding to a perfect flat histogram is known. We want to approximate $\boldsymbol{p}^*$ by vectors $\boldsymbol{p}(t)$ which are constructed by iteratively applying a constant transition matrix $\boldsymbol{\Gamma}$ to an initial vector $\boldsymbol{p}(0)$. Hence, as we have an ergodic random walk due to detailed balance we also have a *rate of convergence* for $\boldsymbol{p}(t) \to \boldsymbol{p}^*$ [91]. This rate should be as large as possible, which is equivalent to the demand that the value of the second largest eigenvalue e_2 of $\boldsymbol{\Gamma}$ should be as small as possible.

Assuming $\Pi_{\beta\alpha} = \Pi_{\alpha\beta}$ we have due to (4.19)

$$P_{\beta\alpha} = \frac{g(H(\alpha))}{g(H(\beta))} \cdot P_{\alpha\beta}. \tag{4.27}$$

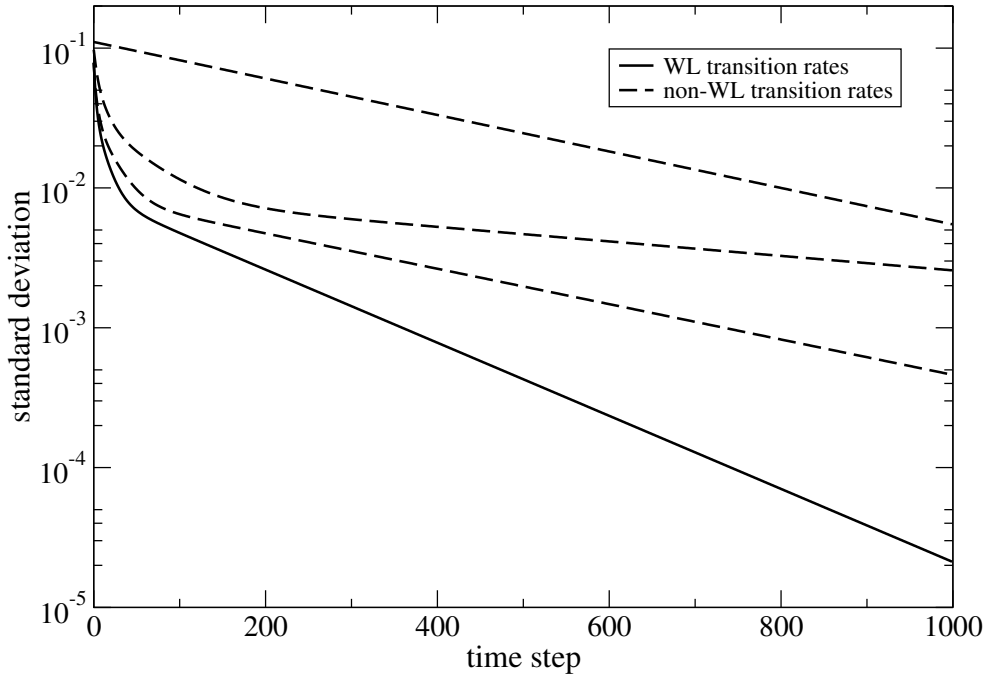

Figure 4.3: Performance of the given transition rates for the long-range Ising model. Shown is the standard deviation of the energy histogram for every time step. The histograms become flatter with growing time, leading to a monotonic falling standard deviation. The WL transition rates outperform the three other suggested rates (4.20), (4.21) and (4.22) (broken lines, from top).

As $P_{\beta\alpha}$ and $P_{\alpha\beta}$ are acceptance probabilities,

$$0 \leq P_{\beta\alpha} \leq 1 \quad \text{as well as} \quad 0 \leq P_{\alpha\beta} \leq 1 \tag{4.28}$$

must be valid, yielding an allowed range of

$$P_{\alpha\beta} \in \left\{ \begin{array}{ll} [0,1] & \text{if } g(H(\alpha)) \leq g(H(\beta)) \\ [0, g(H(\beta))/g(H(\alpha))] & \text{if } g(H(\alpha)) > g(H(\beta)) \end{array} \right\} \tag{4.29}$$

to choose acceptance rates from. This is equal to

$$P_{\alpha\beta} \in \left[0, \min\left[\frac{g(H(\beta))}{g(H(\alpha))}, 1\right]\right]. \tag{4.30}$$

Hence, (4.30) establishes an one-dimensional range for each of the transition rates $\Gamma_{\alpha\beta}$, the maximum of which is given by the WL transition rates.

If we could perform an calculation of the value of e_2 as a function of the transition rates for every pair $\Gamma_{\alpha\beta}/\Gamma_{\beta\alpha}$ then we could easily determine the transition rates with highest performance. But this seems to be nearly impossible to do in general, i. e., for arbitrary complex systems. What can be done is the following.

We know that the largest eigenvalue of the stochastic matrix describing the WL process must be equal to one. Furthermore, we can get a feeling for the allowed region the remaining eigenvalues are located in by calculating GERŠGORIN disks [17]. From every column a disk can be derived. Each disk is situated in the complex plane $(x, \mathrm{i}y)$, and centered around the diagonal element of the corresponding column. The radius of each disk is the sum of the absolute values of the non-diagonal elements. Therefore, the line $x = 1$ in the complex plane is tangent to every GERŠGORIN disk.

For stochastic matrices and ergodic systems we have, besides $e_1 = 1$, $e_1 > e_2 \geq e_3$. The second largest eigenvalue can therefore only be as small as possible if the allowed region for all remaining eigenvalues is as large as possible, hence, if each disk has the largest possible radius. This is indeed given in the case of the WL transition rates, as these represent the largest possible off-diagonal elements. But this is *not* a sufficient condition.

Whether e_2 is indeed as small as possible cannot be derived analytically from that reasoning. But we can still give a *physically* motivated plausible argument: if the process of converging to the limit distribution should be as fast as possible then low acceptance probabilities are counterproductive. The random walkers must not stick to one and the same energy level for long times; rather they have to jump between the levels quickly in order to allow for a large convergence rate. In this sense the WL transition rates might indeed be an optimal choice. —

But so far, we have put the cart before the horse: the density of states needs to be known to set up transitions according to eq. (4.23) and to produce flat histograms. To circumvent this, in the WANG-LANDAU sampling scheme the following procedure has been developed. Although tailored to ISING and POTTS models first, it is also suitable for other systems [76, 92].

1. Create a histogram h covering the energy region of interest. Its bins $h(i)$ are initially set to 0. Create an array $g_{appr}(i)$ of the same size, holding the so far unknown density of states. Initially, set

$g_{appr}(i) = 1 \forall i$. Hold a factor $f > 1$, initially set to $f \approx 2.7$ in the original work.

2. Perform a random walk using the transition rates given by (4.23), employ the values $g_{appr}(i)$: each time a trial move is made and an energy level is proposed to be hit accept or reject it, but update the corresponding $h(i)$ and $g_{appr}(i)$ in any case. Rejecting means "updating the old level". Add 1 to the histogram bin $h(i)$, and perform $g(i) \to f \cdot g(i)$.

3. Check whether the histogram is flat. Originally, flatness meant that every bin of the histogram is not less then 80% of the average histogram. If the histogram is flat, reset every bin to 0, and modify the factor $f \to \sqrt{f}$.

4. If f falls below a prescribed value > 1, such as $\exp(10^{-8})$, the simulation finished, g_{appr} holds approximate relative values for $g(E)$ of the system. If not, go to step 2.

5. To get absolute values, rescale g_{appr} by using the total number of states, or the (known) value for a bin.

This scheme has been proved to be suitable for a large number of models offering discrete [87, 88] state spaces. It has also been extended to cover continuous state spaces [93, 94] like those of Lennard-Jones fluids. Moreover, in a certain sense it might be the optimal choice of transition rates within local-update flat-histogram methods [95]. Measuring the tunneling time of random walkers, i. e., the time to pass from a ground state to an anti-ground state, shows that WL might approach a lower bound of these times. The lower bound has been calculated by a perfect flat histogram method, which employs known densities of states for small systems.

4.3 Matrix Based Methods and Par*Q*

As a further development matrix based methods for calculating the density of states have been developed. The main advantage compared with histogram based methods is that not only *visits* to energy levels are recorded, but *proposed transitions* between energy levels. It is hoped for that this approach delivers better statistics, e. g., better accuracy with the same number of Monte Carlo Sweeps (MCS) performed. All of the

methods rely on the fact that approximating the infinite temperature transition matrix between energy levels of the system under consideration is quite easy.

4.3.1 The Wang-Landau Transition Matrix Method

The idea of recording proposed transitions between energy levels to approximate the density of states is quite old [96, 97, 98, 99, 100]. Basically, it is very simple: every time a transition is proposed an one is added to the corresponding entry of a prepared matrix. At the end of the simulation these entries are post-processed to extract, e. g., the density of states. The methods described in the literature differ in how to get the entries, and how to post-process them. As a detailed description of all of the methods is out of the present scope only one of the latest and most performant methods is introduced here. This methodology, dubbed Wang-Landau Transition Matrix Method (WL-TM) [101], combines the idea of using transition matrices with the sampling of the state space proposed by the WL method.

In WL-TM random walkers are employed which propose moves according to the WL scheme described above. Every time a move from one energy level to a new, not necessarily different energy level is proposed the corresponding entry of a prepared matrix is increased by one. This matrix, zeroed in the beginning, holds the numbers of all proposed transitions in the end of the simulation. Broad phase space sampling is secured by the WL sampling scheme, hence, most if not all of the energy levels have been visited. Of course, the ratios of the visits are determined by the WL scheme.

Post-processing of these data begins by making the filled matrix stochastic. Thereto every entry is divided by the sum of all entries of the corresponding column. After that the density of states can be approximated by minimizing a variance, given in [101].

The method is applicable to lattice and continuous systems. It seems to offer good convergence properties, and is one of the most performant methods to date. A generalization to grand-canonical simulations is also possible [102, 103].

4.3.2 The Q Method and ParQ

Another matrix based method which has been developed here is par$Q$ [104]. Its fundamentals – the Q method – have already been shown some

years ago [96] in the context of lumped or coarse grained models. But implementations and applications to real systems did not exist up to now.

The Q method is again a random walk based algorithm. But in spite of focussing on microstate transitions, transitions from one to another energy level of the system under consideration are described by a master equation. The probability of a random walker to be in a state of energy $j = E_G, \ldots, E_{AG}$ can be written as

$$p_j(t+1) = \sum_{i=E_G}^{E_{AG}} \hat{\Gamma}_{ji}(\beta) \cdot p_i(t) \text{ with } \sum_k \hat{\Gamma}_{ki}(\beta) = 1 \, \forall i. \tag{4.31}$$

Here, $\hat{\Gamma}_{ji}(\beta)$ denotes transitions between energy levels. This transition matrix is a function of the inverse temperature β. The stationary solution p_j^* of (4.31) is the right eigenvector to the largest eigenvalue 1. If we assume a Boltzmann distribution, then

$$p_j^* = \frac{1}{Z(\beta)} g(j) e^{-\beta j}, \tag{4.32}$$

and for $t \to \infty$ we get

$$g(j)e^{-\beta j} = \sum_{i=E_G}^{E_{AG}} \hat{\Gamma}_{ji}(\beta) g(i) e^{-\beta i}. \tag{4.33}$$

Taking the limit $\beta \to 0$ results in an eigenvector equation for the DOS,

$$1 \cdot g(j) = \sum_{i=E_G}^{E_{AG}} Q_{ji} g(i), \tag{4.34}$$

with Q_{ji} denoting the stochastic infinite-temperature transition matrix between the energy levels of the system. Rescaling can easily be done if $g(E)$ is known at some E, or $\sum_E g(E)$ is known. To outline in detail how this method can be applied to real systems we have a look on three rather simple, but analytically solvable examples.

At first we analyze the binary tree of three levels given if fig. 4.4. The degeneracy of the nodes is again chosen to be 2^{E_α}, resulting in a

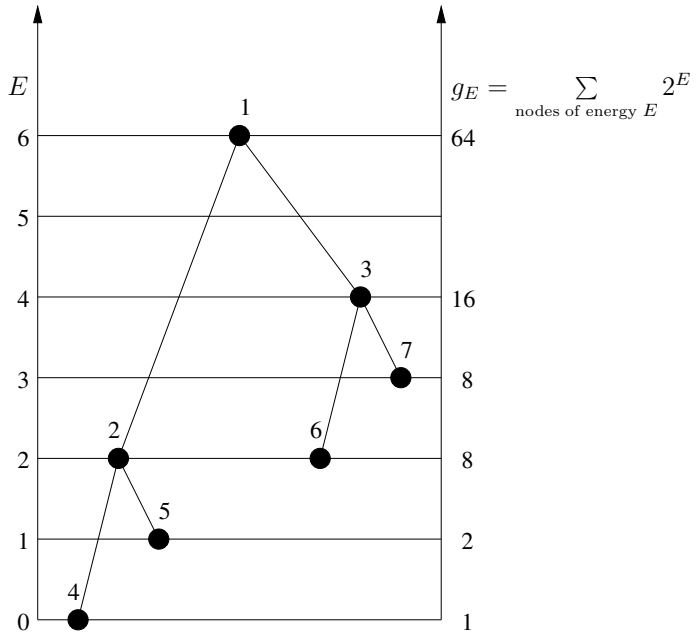

Figure 4.4: A binary tree of three levels. Given are the energy levels which the nodes are situated in and determine their degeneracy, as well as the density of states.

neighborhood relationship

$$\mathbf{\Pi} = c \cdot \begin{bmatrix} 0 & 64 & 64 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 4 & 4 & 0 & 0 \\ 16 & 0 & 0 & 0 & 0 & 16 & 16 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 8 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.35}$$

due to the energy levels given in 4.4. At infinite temperature, every move is accepted, regardless of which energy change it involves. Hence, the Metropolis rules are

$$P_{\beta\alpha}^{Me} = 1, \tag{4.36}$$

yielding a transition matrix

$$\mathbf{\Gamma} = \begin{bmatrix} 1-20c & 64c & 64c & 0 & 0 & 0 & 0 \\ 4c & 1-67c & 0 & 4c & 4c & 0 & 0 \\ 16c & 0 & 1-76c & 0 & 0 & 16c & 16c \\ 0 & 1c & 0 & 1-4c & 0 & 0 & 0 \\ 0 & 2c & 0 & 0 & 1-4c & 0 & 0 \\ 0 & 0 & 4c & 0 & 0 & 1-16c & 0 \\ 0 & 0 & 8c & 0 & 0 & 0 & 1-16c \end{bmatrix}. \tag{4.37}$$

We can use the transfer ratios calculated so far to construct the matrix $\boldsymbol{Q}$ for this tree. Every edge of the tree corresponds to a transfer from one node to another, and possibly from one energy level to another.[3] The edge connecting node 1 with 2, e. g., corresponds to transfers from energy 6 to 2 and vice versa. As we have 6 different energies we can choose $\boldsymbol{Q} \in \mathbb{R}^{6\times 6}$. Creating first a matrix $\tilde{\boldsymbol{Q}}$ by summing up all transfers from an energy level to another gives

$$\tilde{\boldsymbol{Q}} = \left[\begin{array}{c||cccccc} \text{energy} & 0 & 1 & 2 & 3 & 4 & 6 \\ \hline\hline 0 & 1-4c & 0 & c & 0 & 0 & 0 \\ 1 & 0 & 1-4c & 2c & 0 & 0 & 0 \\ 2 & 4c & 4c & 2-83c & 0 & 4c & 4c \\ 3 & 0 & 0 & 0 & 1-16c & 8c & 0 \\ 4 & 0 & 0 & 16c & 16c & 1-76c & 16c \\ 6 & 0 & 0 & 64c & 0 & 64c & 1-20c \end{array}\right]. \tag{4.38}$$

These transfers are the ones triggering a change of the node number – represented by the edges – and the ones which do not correspond to a change of the node number, i. e. which correspond to staying on a node. In our tree, e. g. $\tilde{Q}_{2,2}$, which holds the sum of all transfers from energy 2 to 2, is $\tilde{Q}_{2,2} = \Gamma_{2,2} + \Gamma_{6,6}$.

To get the right ratios between transfers from an energy level to another we divide every entry by the sum of all entries of the corresponding

[3] This need not be the case; trees with "horizontal" edges offer transfers which only change the node number.

column, getting the stochastic infinite temperature matrix

$$\boldsymbol{Q} = \begin{bmatrix} 1-4c & 0 & \frac{1}{2}c & 0 & 0 & 0 \\ 0 & 1-4c & c & 0 & 0 & 0 \\ 4c & 4c & \frac{1}{2}(2-83c) & 0 & 4c & 4c \\ 0 & 0 & 0 & 1-16c & 8c & 0 \\ 0 & 0 & 8c & 16c & 1-76c & 16c \\ 0 & 0 & 32c & 0 & 64c & 1-20c \end{bmatrix}. \tag{4.39}$$

The eigenvector to eigenvalue 1 with column sum 1 is

$$\hat{e}_1 = [1/99, 2/99, 8/99, 8/99, 16/99, 64/99], \tag{4.40}$$

which can be rescaled to give the density of states

$$g_E = 99 \cdot \hat{e}_1 = [1, 2, 8, 8, 16, 64]. \tag{4.41}$$

Indeed, this result coincides with the values depicted in fig. 4.4. This small example points out that the Q method is in principle able to cope with systems consisting of different energy barriers and arbitrary neighborhood relations.

The 3×3 ferro-magnet ($J_{ij} = 1 \forall i, j$) with periodic boundary conditions offers only 6 different energies, $E/J = \{-18, -10, -6, -2, 2, 6\}$. Therefore, we can create a matrix $\tilde{Q} \in \mathbb{R}^{6\times 6}$. Its entries are again the number of possible transfers from one energy level to another,

$$\tilde{Q} = \left[\begin{array}{c||cccccc} E/J & -18 & -10 & -6 & -2 & 2 & 6 \\ \hline\hline -18 & 0 & 18 & 0 & 0 & 0 & 0 \\ -10 & 18 & 0 & 72 & 72 & 0 & 0 \\ -6 & 0 & 72 & 72 & 216 & 72 & 0 \\ -2 & 0 & 72 & 216 & 864 & 504 & 126 \\ 2 & 0 & 0 & 72 & 504 & 360 & 360 \\ 6 & 0 & 0 & 0 & 126 & 360 & 432 \end{array}\right], \quad \sum_{i,j} \tilde{Q}_{ij} = 4608. \tag{4.42}$$

The entries have been collected by a computer program: we get these by classifying all transfers from all microstates s_i to their neighbors $N(s_i)$, defined by a single spin flip, by the proposed energy change. For example, going from the two ground states at $E = -18$ to a state with $E = -10$ is possible in 18 different ways, i. e. $\tilde{Q}_{-10,-18} = 18$. Note that the sum

of all possible transfers is $2^9 \cdot 9 = 4608$, as every microstate has nine neighbors.

Hence, the transition ratios Q for moving from one energy level to another at infinite temperature must be

$$\boldsymbol{Q} = \begin{bmatrix} 0 & 1/9 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1/6 & 4/99 & 0 & 0 \\ 0 & 4/9 & 1/6 & 4/33 & 1/18 & 0 \\ 0 & 4/9 & 1/2 & 16/33 & 7/18 & 7/51 \\ 0 & 0 & 1/6 & 28/99 & 5/18 & 20/51 \\ 0 & 0 & 0 & 7/99 & 5/18 & 8/17 \end{bmatrix} . \tag{4.43}$$

The eigenvector of Q to eigenvalue 1 with column sum 1 is

$$\hat{e}_1 = [1/256, 9/256, 3/32, 99/256, 9/32, 51/256]^{tr} . \tag{4.44}$$

We know that this sample has a total of $2^9 = 512$ microstates, therefore the DOS of that system can be calculated to give

$$g_E = 512 \cdot \hat{e}_1 = [2, 18, 48, 198, 144, 102]^{tr} \tag{4.45}$$

with the subscript E traversing $E = -18, -10, -6, -2, 2, 6$. Enumeration of the state space yields the same result.

A more sophisticated example is that of the symmetric 10 town travelling salesman problem. We assume that all tours are starting and ending in the same town. Then this problem has $(10-1)!/2 = 181440$ microstates. Furthermore, our move class shall be to "swap two towns" of the current tour, i. e., every state has $\binom{9}{2} = 36$ neighbors. The distribution of the towns of our example is sketched in fig. 4.5.

To calculate the infinite temperature transition matrix $\tilde{Q}^{tsp} \in \mathbb{R}^{n \times n}$ we create n "energy bins" of equal width: a state belongs to a bin if its energy lies within the bounds of that bin. Therefore, we can identify transitions from one state to another with a corresponding transition between bins. For our example ground and anti-ground state energy are about 3.38 and 8.02, respectively. If we choose $n = 8$ then the first bin covers the energy region $3.38 \ldots 3.96$.

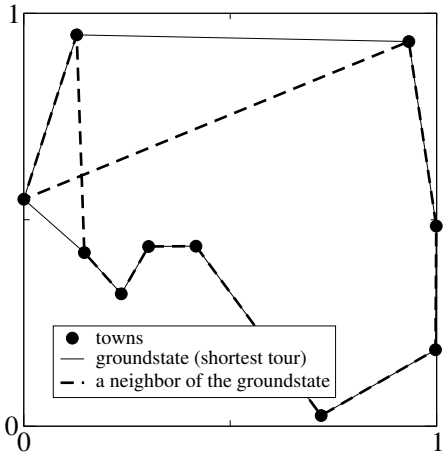

Figure 4.5: A traveling salesman problem: 10 towns, randomly distributed in the unity square. Shown are the ground state (shortest tour) and one of its neighbors. Here, the neighbors are those tours with two towns having been swapped.

For that value of n we collect the entries $\tilde{Q}^{tsp}_{ij}$ with some small computer programs. We get

$$\tilde{Q}^{tsp} = \left[\begin{array}{r||rrrrrrrr} \text{bin no.} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ \hline\hline 1 & 498 & 1521 & 1322 & 717 & 467 & 11 & 0 & 0 \\ 2 & 1521 & 19244 & 32163 & 20133 & 12594 & 2282 & 11 & 0 \\ 3 & 1322 & 32163 & 239688 & 229112 & 125084 & 35820 & 4035 & 0 \\ 4 & 717 & 20133 & 229112 & 783704 & 524604 & 163226 & 25834 & 2 \\ 5 & 467 & 12594 & 125084 & 524604 & 1180958 & 442820 & 70782 & 7 \\ 6 & 11 & 2282 & 35820 & 163226 & 442820 & 594828 & 110031 & 10 \\ 7 & 0 & 11 & 4035 & 25834 & 70782 & 110031 & 107710 & 17 \\ 8 & 0 & 0 & 0 & 2 & 7 & 10 & 17 & 0 \end{array}\right] . \tag{4.46}$$

The sum of the entries is $\sum_{i,j} \tilde{Q}^{tsp}_{i,j} = 6531840 = 36 \cdot 181440$.

The eigenvector of the corresponding stochastic transition matrix $Q^{tsp}_{ij} = \tilde{Q}^{tsp}_{ij} / \sum_{k=1}^{n} \tilde{Q}^{tsp}_{kj}$ to eigenvalue 1 with column sum 1 is

$$\hat{e}^{tsp}_1 = \left[\frac{1}{1440}, \frac{349}{25920}, \frac{3089}{30240}, \frac{5393}{20160}, \frac{21827}{60480}, \frac{12491}{60480}, \frac{1769}{36288}, \frac{1}{181440}\right]^{tr} , \tag{4.47}$$

yielding a DOS of

$$g_E = 181440 \cdot \hat{e}^{tsp}_1 = [126, 2443, 18534, 48537, 65481, 37473, 8845, 1] . \tag{4.48}$$

This is again consistent with an enumeration of the state space of our example. Of course, a larger number of bins gives the DOS at higher resolution in energy, as can be seen in fig. 4.6, but the calculation of

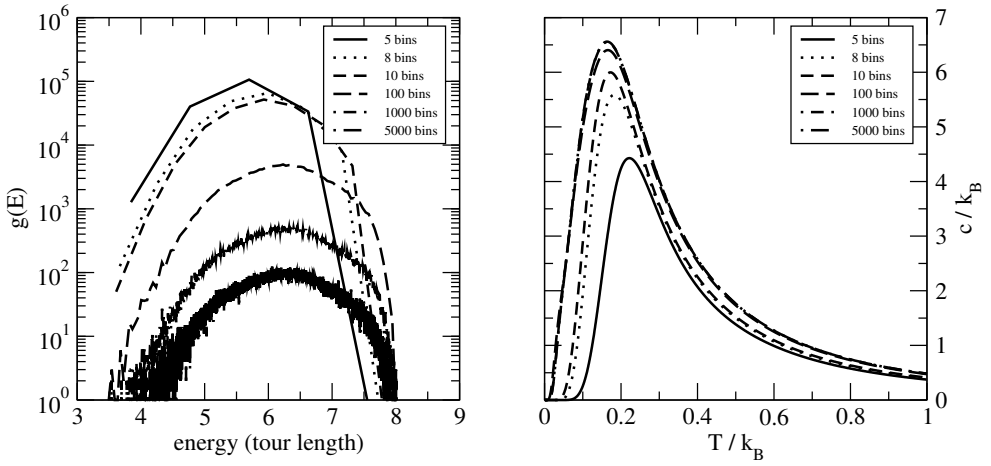

Figure 4.6: (Left) The density of states calculated for the TSP example. Growing numbers of bins $(5, \ldots, 5000)$ deliver a $g(E)$ of much better resolution, but calculating it as part of an diagonalization of Q becomes numerically very intensive. (Right) The specific heat $c(T)$ for the TSP example calculated with the $g(E)$ A higher bin number yields a better approximation of $c(T)$. The curves for a bin number of 1000 and 5000 coincide at this scale.

the eigenvector becomes numerically more intensive. Otherwise, a better resolved DOS yields better approximations of the thermodynamic quantities of interest. For example, the specific heat $c(T)$ of the TSP problem – calculated with (4.3) – is shown in fig. 4.6. A growing number of bins improves the approximation of $c(T)$. At the scale used in this figure about 1000 bins are necessary until no further improvement of $c(T)$ can be seen.

In general, the state space even of systems which are considered "small" cannot be enumerated, so $\boldsymbol{Q}$ must be approximated. The question has to be raised how this approximation can be carried out accurately and efficiently. A method to achieve this is par$Q$ [104]. This parallelized algorithm to approximate Q has two essential elements:

1. It employs random walkers to collect entries Q_{ji}. These are steered energetically broadly through the state space, using e. g. METROPOLIS sampling with changing temperature. The *proposed* moves from energy level or bin no. i to j contribute to Q_{ji} as these would be the moves taken in a METROPOLIS run at $\beta = 0$.

The eigenvector calculation is carried out as a simple iterated matrix/vector multiplication in the current implementation. This approach is very fast.

2. It is inherently parallelized. The walkers move independently from each other and are therefore easily distributable over the processors of a parallel computer or compute cluster. A very low communication time compared to the whole run time is needed, so par$Q$ offers a superb speedup. Moreover, due to the number of MC steps being selectable its run time in terms of MCS is known.

Technically, par$Q$ is a three-step procedure:

1. Collect entries for a non-stochastic matrix $\tilde{\boldsymbol{Q}} \in \mathbb{R}^{n \times n}$ by steering the random walkers through the state space. Sample energetically broadly; employ, e. g., METROPOLIS rules in combination with an exponential or linear schedule. Keep book of *proposed* moves from an energy level to another level.

2. Make $\tilde{\boldsymbol{Q}}$ stochastic by dividing each entry by the corresponding column sum, $\tilde{\boldsymbol{Q}}$ becomes $\boldsymbol{Q}_{appr}$. Columns with no entries do not contribute to the eigenvector, they can safely be ignored.

3. Perform the iterated matrix/vector multiplication: multiply a vector g_E with column sum 1 onto $\boldsymbol{Q}_{appr}$. Multiply the result onto $\boldsymbol{Q}_{appr}$. Iterate until some convergence criterion is met. Rescale the resulting $g(E)$.

The accuracy of par$Q$ has been shown considering ferro-magnets and spin glasses with random couplings [104]. Furthermore, highly frustrated $\pm J$ models have been investigated [105].

For the latter LUKIC et al. [73] provided exact densities for five 50×50 $\pm J$ spin glass samples. This opened the possibility to test the method with the largest systems which are exactly calculable to date. In fig. 4.7 par$Q$ has been employed to approximate the given exact densities. For every sample 10^8 MCS have been performed, using a linear temperature schedule. Although par$Q$ – implemented in this special way – delivers asymmetric walks in energy only the energy region $[-4000, 0]$ is shown for clarity[4]. For every sample the approximated and exact density of states coincides, so the relative error for the logarithm of the density of states is also shown.

[4]High energy regions can be sampled with inverse METROPOLIS moves.

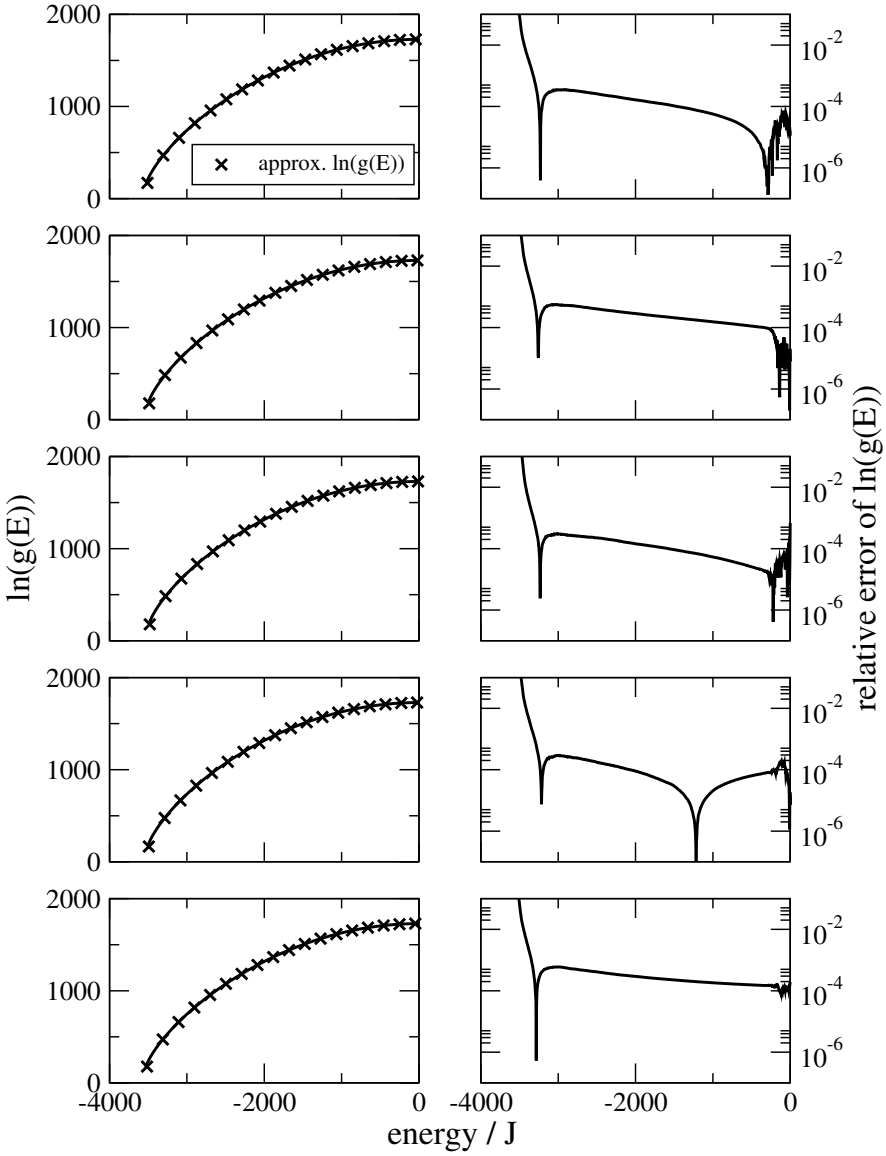

Figure 4.7: (Left column) Each row shows the exact (–) and approximated (×) density of states $g(E)$ for one of five different $\pm J$ ISING spin glasses of size 50×50. For clarity, only every 50th approximated value is shown. (Right column) As both curves coincide at this scale the corresponding relative errors $|\ln g_{\text{true}} - \ln g_{\text{approx}}|/\ln g_{\text{true}}$ are given. For each sample 10^8 MCS have been performed.

The method is very precise over a wide energy region. Only the density of states at low energies offers high relative errors. Of course, this is due to the low lying states of the samples being not easy to find, resulting in matrix entries of the corresponding $\boldsymbol{Q}$ near or equal to zero and – consequently – to a very bad approximated density of states in that energy region.

This is not a disadvantage of this particular method, but a general difficulty. Anyway, the par$Q$ methodology seems to outperform other state-of-the-art methods easily. As its "workhorse" is SA, i. e., an *optimization* algorithm, finding ground states can be done quicker than within WL and consequently the WL-TM scheme. This advantage leads to better sampled densities of states at low energies with the same number of MCS. Furthermore, energy regions of interest can easily be investigated by a simple tuning of the temperature schedule without introducing any numerical artefacts or difficulties. In contrast, WL walkers would have to be confined within an energy band, leading to systematic errors [106].

For fig. 4.8 the density of states for the five samples has again been approximated with par$Q$, and this time also with the original WL scheme. Two different run times in terms of MCS have been used. Shown are the best-so-far energies found by WL and par$Q$ for 10^4 and 10^6 MCS. The par$Q$ method always finds lower energies, and consequently approximates the corresponding density of states better than WL. In this sense it outperforms WL-TM too. Of course, it has to be said that the random walkers in WL have to explore the whole energy region, whereas the walkers in par$Q$ are systematically led to the low energies. But this is only an issue if high-temperature behavior is to be examined.

Another advantage of par$Q$ is that the evaluation of the density of states from the matrix entries is 'extremely' easy, compared to the extraction within the WL-TM scheme. Additionally, the method is very precise, even in the present first version.

Furthermore, a parallelization is straightforward. This has been done using MPI [107] and the CLiC [108] and Riesen clusters. The fig. 4.9 shows an example of the speedup behavior of the algorithm. Speedups have been calculated by fitting expected execution times of the form

$$t(n) = t_s + \frac{t_p}{n} + (n-1) \cdot t_c \tag{4.49}$$

with a serial, fractions of a basic parallel and multiples of a basic communication time t_s, t_p and t_c, respectively. The number of used nodes is denoted with n. This fit only works for the first 150 data points. So a

E_{bsf} for 10^4 MCS					
sample no.	1	2	3	4	5
par$Q$	-3442	-3430	-3426	-3422	-3514
WL	-3334	-3332	-3316	-3336	-3356

E_{bsf} for 10^6 MCS					
sample no.	1	2	3	4	5
par$Q$	-3498	-3478	-3470	-3478	-3514
WL	-3494	-3468	-3456	-3472	-3504

Figure 4.8: Best-so-far energies for WL and par$Q$ for two different numbers of MCS performed with the five $50 \times 50 \pm J$ samples. The par$Q$ method always finds lower energy levels, hence, approximates $g(E)$ better at low energies.

modified fit function of the form

$$t(n) = t_s + \frac{t_p}{n} + (n-1)^{\gamma} \cdot t_c \tag{4.50}$$

was fitted to all data points. The plateau effect observable on the left side can then be taken into account. This effect is mainly due to the MPI implementation used.

The par$Q$ method scales well as long as the number of random walkers is significantly larger than the number of nodes used: the number of walkers should be ten times the number processors in use. Furthermore, in some cases super-linear speedups have been observed [104] which are due to cache effects of the processors.

4.3.3 Systematic improvements

Although calculating the density accurately in its present form, par$Q$ still offers some possibilities for systematic improvements. For example, the schedule is – analog to Simulated Annealing – subject to optimization. There must be, of course, at least one schedule which minimizes the MC steps needed to sample $g(E)$ with a prescribed accuracy. Connected with this issue is the observation that the approximation of the entries of $\boldsymbol{Q}$ is not perfect yet. A schedule which performs relatively many steps at low temperature compared to high temperature approximates the entries with small indices better than those with large indices. A

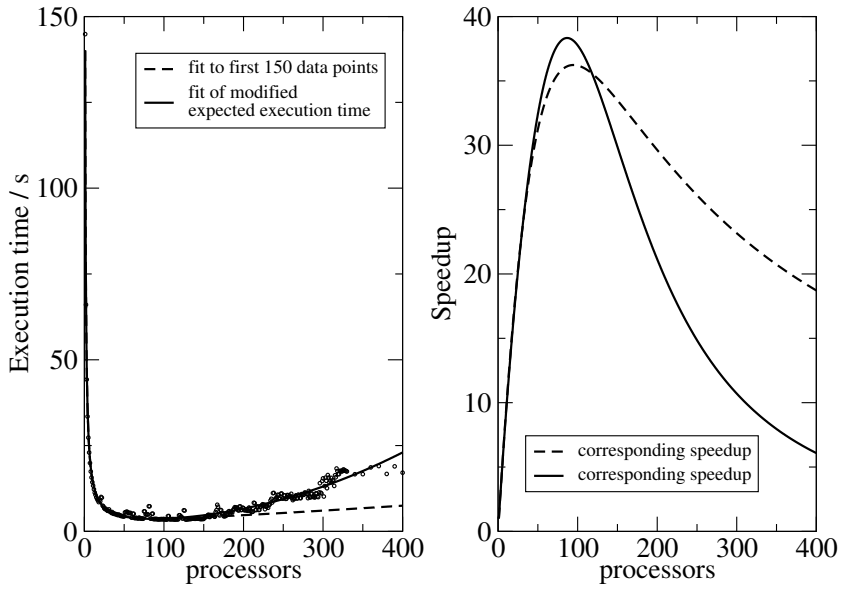

Figure 4.9: Execution time (left) and fitted speedups (right) for par$Q$, employing 1000 random walkers on up to 400 nodes. The fits correspond to two different expected execution times (see text).

good schedule should deliver equally well approximated entries Q_{ij} at every temperature. Furthermore, it is not clear how the calculation of the eigenvector is affected by this mixture of differently well approximated entries.

Chapter 5

Conclusions

The state space of complex systems is one of the most important means to describe ideas, methods and algorithms for finding ground states, or optimal solutions, of hard combinatorial problems. Based on a short description how this space is structured a methodology to model the dynamics in such a space has been recalled in Chapter 2. The movements of random walkers can be described by a master equation which is discrete in time.

With the help of this dynamics it was possible to characterize stochastic optimization algorithms in Chapter 3 as MARKOV processes. Furthermore, it enabled us to decide whether an optimization algorithm offers higher performance than another one by calculating optimal sequences of transition matrices.

The Extremal Optimization heuristics was introduced and explained in detail. This methodology could also be characterized as a MARKOV process. Moreover, a proof was given that the original implementation τ-Extremal Optimization cannot be optimal in general. Instead, a technique termed Fitness Threshold Accepting was developed and shown to be the best possible implementation. For this algorithm the optimal fitness threshold as a function of time has been calculated for small systems. Surprisingly these schedules do not show a monotonic decrease of the fitness threshold.

The Continuous Extremal Optimization algorithm outlined a recipe how the method could be applied to continuous state spaces. Besides that, it showed a possible starting point for a generalization of the given proof for continuous state spaces. A promising way might be the introduction of some "hybrid" dynamics into the continuous state space by mixing a local search algorithm with stochastic updates.

Connected with the search for ground states is the characterization of the state space as a whole by the density of states. Besides its central role in equilibrium thermodynamics it also serves as a means to describe the difficulty of an optimization problem. In Chapter 4 the WANG-LANDAU algorithm to approximate the density of states has been investigated in detail. Based on mild assumptions it could be characterized as the flat-histogram method with the highest probability flows between microstates. Analytical and numerical evidence was given which indicates that this method is indeed the fastest converging one in the class of histogram-based methods. Nevertheless, WANG-LANDAU sampling might fail to calculate the density of states of complex systems with a lot of local minima. Furthermore, parallelization is not straightforward.

Two matrix-based methods have been described. The first employed WANG-LANDAU like updates, and consequently suffers from the same disadvantages. As an alternative we developed a general-purpose algorithm, the par$Q$ method. This matrix-based algorithm circumvents the difficulties observed in the WANG-LANDAU sampling as it uses Simulated Annealing steps to explore the energetically low lying part of the state space. Moreover, it is inherently parallel, as non-interacting random walkers are employed. This gave rise to very efficient implementations even on compute clusters with a slow communication network.

Finally a summary of still open questions is given. They can be seen as a starting point for further investigations.

1. The structure of the given proof is very general. It should be possible to apply the given ideas to other classes of stochastic optimization strategies like Genetic Algorithms. As this strategy is also entirely Markovian it should be no surprise that some Θ-function-like distributions for the crossover, mutation or selection probability might do best.

2. In Continuous Extremal Optimization also a Fitness Threshold Accepting strategy might do best. To prove this a generalization should be possible by introducing a local non-stochastic search algorithm, reducing the dynamics from continuous to discrete state spaces.

3. An analytical proof of the optimality, or non-optimality, of WANG-LANDAU sampling in its self-consistent implementation is still unknown. To obtain it, maybe transition matrices with randomly dis-

tributed entries might be introduced. The allowed range to choose transition rates from has been given, but it is unclear whether choosing the highest rate for every entry is optimal within the proposed scheme of repeatedly updating a histogram.

4. The current implementation of par$Q$ can be improved systematically. The approximated entries should, of course, reflect the density of states well, resulting in equally small relative errors in the *whole* energy region. Also the second largest eigenvector of the transition matrix should be small, as this defines the rate of convergence of the proposed matrix/vector multiplication. Both requirements might be met by a self-adapting temperature schedule. This schedule could steer some random walkers into energy regions which have to be explored in order to improve the approximation of the corresponding matrix entries. The question is how this self-adaption could look like.

5. The density of states can serve as a measure for the difficulty of optimization problems. Therefore, it should be possible to introduce a new type of optimization methods which are not based on local information about the state space, but global information: it might be promising to propose steps in the random walk to find the ground state due to transition rates based on the density of states. In order to do so methodologies to extract more detailed information like the density of local minima and barriers must be introduced.

6. The Extremal Optimization methodology finds energetically low lying states with respect to one objective function very quickly. In the process of finding Pareto optimal solutions of a multi-objective optimization task such a search has to be performed with respect to each of the objectives. In this sense an application of Extremal Optimization to multi-objective optimization seems promising.

Appendix A

Implementation

The analytical and numerical results of this thesis have been produced by a combination of different methods. Analytical results were mostly verified with Mathematica. To produce and evaluate numerical data different computer languages have been used, e. g.,

- Java for computing the transition matrices of FTA,
- Mathematica for calculating optimal temperature schedules on the trees and optimal FTA,
- C/C++ for implementing CEO combined with CG for Lennard-Jones clusters, the Wang-Landau method on spin glasses and estimating the infinite temperature transition matrix,
- AWK for post-processing data.

Parallelization of par$Q$ was done using the object-oriented framework of MPI. The Gnu MP library has been used for arbitrary large floating point numbers.

All of these programs are an integral part of this thesis. They are available upon request.

Bibliography

[1] A. P. Young, editor. *Spin glasses and random fields.* World Scientific Publishing Co. Pte. Ltd., Singapore, 1997.

[2] H. Maletta and W. Felsch. *Insulating spin-glass system $Eu_xSr_{1-x}S$.* Phys. Rev. B, 20(3):1245–1260, 1979.

[3] M. A. Ruderman and C. Kittel. *Indirect Exchange Coupling of Nuclear Magnetic Moments by Conduction Electrons.* Phys. Rev., 96(1):99–102, 1954.

[4] T. Kasuya. *A Theory of Metallic Ferromagnetism and Antiferromagnetism on Zeners Model.* Prog. Theor. Phys., 16(1):45–57, 1956.

[5] K. Yosida. *Magnetic Properties of Cu-Mn Alloys.* Phys. Rev., 106(5):893–898, 1957.

[6] S. F. Edwards and P. W. Anderson. *Theory of Spin Glasses.* J. Phys. F: Met. Phys., 5(5):965–974, 1975.

[7] D. Sherrrington and S. Kirkpatrick. *Solvable Model of a Spin-Glass.* Phys. Rev. Lett., 35(26):1792–1796, 1975.

[8] J. C. Schön, M. A. C. Wevers, and M. Jansen. *'Entropically' stabilized region on the energy landscape of an ionic solid.* J. Phys.: Condens. Matter, 15(32):5479–5486, 2003.

[9] J. C. Schön and M. Jansen. *Determination, prediction, and understanding of structures, using the energy landscapes of chemical systems - Part I.* Zeitschrift für Kristallographie, 216(6):307–325, 2001.

[10] J. C. Schön and M. Jansen. *Determination, prediction, and understanding of structures, using the energy landscapes of chemical*

systems - Part II.
Zeitschrift für Kristallographie, 216(7):361–383, 2001.

[11] J. R. Shewchuk. *An Introduction to the Conjugate Gradient Method Without the Agonizing Pain.*
`http://www.bmen.tulane.edu/~rth/painless-conjugate-gradient.pdf`,
August 1994.

[12] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C.*
Cambridge University Press, Cambridge, 2nd edition, 1992.

[13] D. H. N. Anh, K. H. Hoffmann, S. Seeger, and S. Tarafdar. *Diffusion in disordered Fractals.*
Europhys. Lett., 70(1):109–115, 2005.

[14] S. Seeger, A. Franz, C. Schulzky, and K. H. Hoffmann. *Random Walks on Finitely Ramified Sierpinski Carpets.*
Comp. Phys. Comm., 134(3):307–316, 2001.

[15] P. Langrock and W. Jahn. *Einführung in die Theorie der Markovschen Ketten und ihre Anwendungen.*
BSB B. G. Teubner Verlagsgesellschaft, Leipzig, 1979.

[16] N. G. van Kampen. *Stochastic Processes in Physics and Chemistry.*
Elsevier, Amsterdam, 1997.

[17] R. A. Horn and C. R. Johnson. *Matrix Analysis.*
Cambridge University Press, Cambridge, UK, 1990.

[18] D. Ruelle. *Smooth Dynamics and New Theoretical Ideas in Nonequilibrium Statistical Mechanics.*
J. Stat. Phys., 95(1-2):393–468, 1999.

[19] D. J. Evans and L. Rondoni. *Comments on the Entropy of Nonequilibrium Steady States.*
J. Stat. Phys., 109(3-4):895–920, 2002.

[20] T. Hanney and M. R. Evans. *Einstein Relation for Nonequilibrium Steady States.*
J. Stat. Phys., 111(5-6):1377–1390, 2003.

[21] S. Kirkpatrick, C.D. Gelatt, and M.P. Vecchi. *Optimization by Simulated Annealing.*
Science, 220(4598):671–680, 1983.

[22] S. Kirkpatrick. *Optimization by simulated annealing: quantitative studies.*
J. Stat. Phys., 34(5/6):975–986, 1984.

[23] V. Černy. *Thermodynamical Approach to the Travelling Salesman Problem: An Efficient Simulation Algorithm.*
J. Optim. Theory Appl., 45:41–51, 1985.

[24] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. *Equations of state calculations by fast computing machines.*
J. Chem. Phys., 21:1087–1091, 1953.

[25] G. Dueck and T. Scheuer. *Threshold Accepting: A General Purpose Optimization Algorithm Appearing Superior to Simulated Annealing.*
J. Comput. Phys., 90:161–175, 1990.

[26] P. Moscato and J.F. Fontanari. *Stochastic versus Deterministic Update in Simulated Annealing.*
Phys. Lett. A, 146(4):204–208, 1990.

[27] C. Tsallis. *Possible Generalization of Boltzmann-Gibbs Statistics.*
J. Stat. Phys., 52(1/2):479–487, 1988.

[28] C. Tsallis and D. A. Stariolo. *Generalized Simulated Annealing.*
Physica A, 233(1-2):395–406, 1996.

[29] T. J. P. Penna. *Traveling salesman problem with Tsallis statistics.*
Phys. Rev. E, 51(1):R1–R3, 1995.

[30] A. Franz and K. H. Hoffmann. *Threshold accepting as limit case for a modified Tsallis statistics.*
Appl. Math. Lett., 16(1):27–31, 2003.

[31] A. Franz and K. H. Hoffmann. *Optimal Annealing Schedules for a Modified Tsallis Statistics.*
J. Comput. Phys., 176(1):196–204, 2002.

[32] P. Salamon, P. Sibani, and R. Frost. *Facts, Conjectures, and Improvements for Simulated Annealing*, volume 7 of *Monographs on Mathematical Modeling and Computation*.
SIAM, Philadelphia, USA, 1st edition, 2002.

[33] W. Wenzel and K. Hamacher. *Stochastic Tunneling Approach for Global Minimization of Complex Potential Energy Landscapes.*
Phys. Rev. Lett., 82(15):3003–3007, 1999.

[34] A. Franz, K. H. Hoffmann, and P. Salamon. *Best Possible Strategy for Finding Ground States.*
Phys. Rev. Lett., 86(23):5219–5222, 2001.

[35] K. H. Hoffmann, A. Franz, and P. Salamon. *Structure of best possible strategies for finding ground states.*
Phys. Rev. E, 66(4):046706/1–046706/7, 2002.

[36] M. O. Jakobsen, K. Mosegaard, and J. M. Pedersen. *Model Optimization in Exploration Geophysics II,* edited by A. Vogel.
Friedr. Vieweg & Son, Braunschweig, 1988.

[37] K. H. Hoffmann, P. Sibani, J.M. Pedersen, and P. Salamon. *Optimal Ensemble Size for Parallel Implementations of Simulated Annealing.*
Appl. Math. Lett., 3(3):53–56, 1990.

[38] Y. Nourani and B. Andresen. *A comparison of simulated annealing cooling strategies.*
J. Phys. A: Math. Gen., 31(41):8373–8385, 1998.

[39] J. G. Kemeny and J. L. Snell. *Finite Markov Chains.*
Van Nostrand, Princeton, 1960.

[40] K. Ergenzinger. *Optimale Kontrolltheorie für Simulated-Annealing-Schedules auf selbstähnlichen Strukturen.*
Diploma thesis, Universität Heidelberg, 1993.

[41] K. H. Hoffmann and P. Salamon. *The Optimal Simulated Annealing Schedule for a Simple Model.*
J. Phys. A: Math. Gen., 23:3511–3523, 1990.

[42] S. Schubert and K. H. Hoffmann. *Aging in enumerated spin glass state spaces.*
Europhys. Lett., 66(1):118–124, 2004.

[43] S. Schubert. *Random Walks in Complex Systems – Anomalous Relaxation.* PhD thesis, TU Chemnitz, Chemnitz, June 1999.
see also `http://archiv.tu-chemnitz.de/pub/1999/0017`.

[44] S. Schubert and K. H. Hoffmann. *The structure of enumerated spin glass state spaces.*
accepted by Computer Physics Communications, January 2004.

[45] J. C. Schön. *Preferential trapping on energy landscapes in regions containing deep-lying minima - the reason for the success of simulated annealing?*
J. Phys. A: Math. Gen., 30(7):2367–2389, 1997.

[46] S. Geman and D. Geman. *Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images.*
In IEEE Transanctions on Pattern Analysis and Machine Intelligence PAMI 6, pages 721–741, 1984.

[47] K. Ergenzinger, K. H. Hoffmann, and P. Salamon. *Optimal Simulated Annealing Schedules for Self Similar Systems.* J. Appl. Phys., 77(11):5501–5508, 1995.

[48] S. Boettcher and A. G. Percus. *Extremal optimization: Methods derived from co-evolution.*
In Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '99), pages 826–832, 1999.

[49] S. Boettcher. *Extremal optimization of graph partitioning at the percolation threshold.*
J. Phys. A: Math. Gen., 32(28):5201–5211, 1999.

[50] S. Boettcher and A. G. Percus. *Combining Local Search with Co-Evolution in a Remarkably Simple Way.*
In Proceedings of the 2000 Congress on Evolutionary Computation, pages 1578–1584. IEEE, 2000.

[51] S. Boettcher, A. G. Percus, and M. Grigni. *Optimizing through co-evolutionary avalanches.*
In Proceedings of the Sixth International Conference on Parallel Problem Solving from Nature, volume 1917 of *Lecture Notes in Computer Science*, pages 447–456, 2000.

[52] P. Bak and K. Sneppen. *Punctuated equilibrium and criticality in a simple model of evolution.*
Phys. Rev. Lett., 71:4083–4086, 1993.

[53] S. Boettcher. *Extremal Optimization: Heuristics via Co-Evolutionary Avalanches*
Computing in Science and Engineering, 2(6):75–82, 2000.

[54] S. Boettcher and A. Percus. *Nature's way of optimizing.*
Artificial Intelligence, 119:275–286, 2000. research note.

[55] S. Boettcher and A. G. Percus. *Extremal optimization for graph partitioning.*
Phys. Rev. E, 64(2):026114, 2001.

[56] S. Boettcher and M. Grigni. *Jamming Model for the Extremal Optimization Heuristic.*
LANL Preprint, cond-mat:0110165, 2001.

[57] S. Boettcher and A. G. Percus. *Optimization with Extremal Dynamics.*
Phys. Rev. Lett., 86(23):5211–5214, 2001.

[58] S. Meshoul and M. Batouche. *Robust Point Correspondence for Image Registration Using Optimization with Extremal Dynamics.*
Lecture Notes in Computer Science, 2449:330–337, 2002.

[59] F. Heilmann, K. H. Hoffmann, and P. Salamon. *Best possible probability distribution over extremal optimization ranks.*
Europhys. Lett., 66(3):305–310, 2004.

[60] K. H. Hoffmann, F. Heilmann, and P. Salamon. *Fitness threshold accepting over extremal optimization ranks.*
Phys. Rev. E, 70(4):046704–1 – 046704–6, 2004.

[61] R. E. Bellman and S. E. Dreyfus. *Applied Dynamic Programming.*
Princeton University Press, Princeton, 1962.

[62] T. Zhou, W.-J. Bai, L.-J. Cheng, and B.-H. Wang. *Continuous extremal optimization for Lennard-Jones clusters.*
Phys. Rev. E, 72(1):016702, 2005.

[63] D. C. Liu and J. Nocedal. *On the limited memory BFGS method for large-scale optimization.*
Math. Progr., 45(3):503–528, 1989.

[64] D. J. Wales, J. P. K. Doye, A. Dullweber, M. P. Hodges, F. Y. Naumkin, F. Calvo, J. Hernández-Rojas, and T. F. Middleton. *The Cambrigde Cluster Database.*
URL `http://www-wales.ch.cam.ac.uk/CCD.html`, 2005.

[65] J. A. Northby. *Structure and binding of Lennard-Jones clusters:* $13 \leq N \leq 147$.
J. Chem. Phys., 87(10):6166–6177, 1987.

[66] W. Nolting. *Statistische Physik*, volume 6 of *Grundkurs: Theoretische Physik*.
Verlag Zimmermann-Neufang, Ulmen, 1994.

[67] K. Binder and D. W. Heermann. *Monte Carlo Simulation in Statistical Physics*, volume 80 of *Springer Series in Solid-State Sciences*.
Springer-Verlag, 1992.

[68] R. H. Swendsen and J.-S. Wang. *Nonuniversal Critical Dynamics in Monte Carlo Simulations.*
Phys. Rev. Lett., 58:86–88, 1987.

[69] `http://www.ibiblio.org/e-notes/Perc/contents.htm`,
July 2004.

[70] H. Rosé, T. Asselmeyer, and W. Ebeling. *The Density of States - a Measure of the Difficulty of Optimisation Problems.*
In H.-M. Voigt, W. Ebeling, I. Rechenberg, and H.-P. Schwefel, (editors), *Parallel Problem Solving from Nature IV. Proceedings of the International Conference on Evolutionary Computation*, p. 208.
Springer-Verlag, Heidelberg, 1996.

[71] P. D. Beale. *Exact Distribution of Energies in the Two-Dimensional Ising Model.*
Phys. Rev. Lett., 76:78–81, 1996.

[72] A. Galluccio, M. Loebl, and J. Vondrák. *New Algorithm for the Ising Problem: Partition Function for Finite Lattice Graphs.*
Phys. Rev. Lett., 84(26), 2000.

[73] J. Lukic, A. Galluccio, E. Marinari, O. C. Martin, and G. Rinaldi. *Critical Thermodynamics of the Two-Dimensional $\pm J$ Ising Spin Glass.*
Phys. Rev. Lett., 92(11):11720, 2004.

[74] A. M. Ferrenberg and R. H. Swendsen. *New Monte Carlo Technique for Studying Phase Transitions.*
Phys. Rev. Lett., 61:2635–2638, 1988.

[75] A. M. Ferrenberg and R. H. Swendsen. *New Monte Carlo Technique for Studying Phase Transitions (ERRATA).*
Phys. Rev. Lett., 63(15):1658, 1989.

[76] M. Troyer, F. Alet, and S. Wessel. *Histogram Methods for Quantum Systems: from Reweighting to Wang-Landau Sampling.*
Brazilian Journal of Physics, 24(2A):377–383, 2004.

[77] E. Marinari and G. Parisi. *Simulated tempering: a new MonteCarlo scheme.* Europhys. Lett., 19:451, 1992.

[78] C. Geyer and E. Thompson. *Annealing Markov chain Monte Carlo with applications to ancestral interference.*
Journal of the American Statistical Association, 90:909–920, 1995.

[79] C. J. Geyer. *Markov chain Monte Carlo maximum likelihood.*
In E. Keramigas, (editor), *Computing Science and Statistics: The 23rd symposium on the interface*, pp. 156–163, Interface Foundation, Faifax, 1991.

[80] K. Hukushima and K. Nemoto. *Exchange Monte Carlo method and application to spin glass simulations.*
J. Phys. Soc. Japan, 65(4):1604–1608, 1996.

[81] B. A. Berg and T. Neuhaus. *Multicanonical Ensemble: A New Approach to Simulate First-Order Phase Transitions.*
Phys. Rev. Lett., 68:9–12, 1992.

[82] B. A. Berg and T. Celik. *New approach to spin-glass simulations.*
Phys. Rev. Lett., 69(15):2292–2295, 1992.

[83] J. S. Liu. *Monte Carlo Strategies in Scientific Computing.*
Springer-Verlag, New York, 2001.

[84] J. Lee. *New Monte Carlo Algorithm: Entropic Sampling.*
Phys. Rev. Lett., 71(2):211–214, 1993.

[85] K. K. Bhattacharya and J. P. Sethna. *Multicanonical methods, molecular dynamics, and Monte Carlo methods: Comparison for Lennard-Jones glasses.*
Phys. Rev. E, 57(3), 1997.

[86] B. Hesselbo and R. B. Stinchcombe. *Monte Carlo Simulation and Global Optimization without Parameters.*
Phys. Rev. Lett., 74(12):2151–2155, 1995.

[87] F. Wang and D. P. Landau. *Efficient, Multiple-Range Random Walk Algorithm to Calculate the Density of States.*
Phys. Rev. Lett., 86:2050–2053, 2001.

[88] F. Wang and D. P. Landau. *Determining the density of states for classical statistical models: A random walk algorithm to produce a flat histogram.*
Phys. Rev. E, 64:056101, 2001.

[89] N. Rathore and J. J. de Pablo. *Monte Carlo simulation of proteins through a random walk in energy space.*
J. Chem. Phys., 116(16):7225–7230, 2002.

[90] N. B. Wilding. *Computer simulation of fluid phase transitions.*
Am. J. Phys., 69(11):1147–1155.

[91] J. S. Rosenthal. *Convergence rates of Markov chains.*
SIAM Rev., 37:387–405, 1995.

[92] M. Troyer, S. Wessel, and F. Alet. *Flat Histogram Methods for Quantum Systems: Algorithms to Overcome Tunneling Problems and Calculate the Free Energy.*
Phys. Rev. Lett., 90(12):120201, 2003.

[93] M. S. Shell, P. G. Debenedetti, and A. Z. Panagiotopoulos. *Generalization of the Wang-Landau method for off lattice simulations.*
Phys. Rev. E, 66(5):056703, 2002.

[94] Q. Yan, R. Faller, and J. J. de Pablo. *Density-of-states Monte Carlo method for simulation of fluids.*
J. Chem. Phys., 116(20):8745–8749, 2002.

[95] P. Dayal, S. Trebst, S. Wessel, D.Würtz, M. Troyer, S. Sabhapandit, and S. N. Coppersmith. *Performance Limitations of Flat-Histogram Methods.*
Phys. Rev. Lett., 92(9):097201, 2004.

[96] B. Andresen, K.H. Hoffmann, K. Mosegaard, J. Nulton, J.M. Pedersen, and P. Salamon. *On lumped models for thermodynamic properties of simulated annealing problems.*
J. Phys. France, 49:1485–1492, 1988.

[97] J.-S. Wang, Tien Kiat Tay, and R. H. Swendsen. *Transition Matrix Monte Carlo Reweighting and Dynamics.*
Phys. Rev. Lett., 82:476–479, 1999.

[98] J.-S. Wang. *Transition Matrix Monte Carlo Method.*
Comp. Phys. Comm., 121-122:22–25, 1999.

[99] J.-S. Wang and L. W. Lee. *Monte Carlo algorithms based on the number of potential moves.*
Comp. Phys. Comm., 127:131–136, 2000.

[100] J.-S. Wang and R. H. Swendsen. *Transition Matrix Monte Carlo Method.*
J. Stat. Phys., 106:245–285, 2002.

[101] M. S. Shell, P. G. Debenedetti, and A. Z. Panagiotopoulo. *An improved Monte Carlo method for direct calculation of the density of states.*
J. Chem. Phys., 119(18):9406–9411, 2003.

[102] J. R. Errington. *Evaluating surface tension using grand-canonical transition-matrix Monte Carlo simulation and finite-size scaling.*
Phys. Rev. E, 67(1):012102, 2003.

[103] J. R. Errington. *Direct calculation of liquid–vapor phase equilibria from transition matrix Monte Carlo simulation.*
J. Chem. Phys., 118(22):9915–9925, 2003.

[104] F. Heilmann and K. H. Hoffmann. *ParQ – high-precision calculation of the density of states.*
Europhys. Lett., 70(2):155–161, 2005.

[105] F. Heilmann and K. H. Hoffmann. *ParQ for $\pm J$ Ising spin glasses.*
(unpublished).

[106] B. J. Schulz, K. Binder, M. Müller, and D. P. Landau. *Avoiding boundary effects in Wang-Landau sampling.*
Phys. Rev. E, 67(6):067102, 2003.

[107] The MPI Forum. *MPI-2: Extensions to the Message-Passing Interface.*
homepage: `http://www.mpi-forum.org`,
September 2001.

[108] Technische Universität Chemnitz.
homepage: `http://www.tu-chemnitz.de/urz/clic/`,
September 2003.

Zusammenfassung

Die Lösung kombinatorischer Optimierungsprobleme kann mit dem Auffinden des Grundzustandes komplexer physikalischer Systeme gleichgesetzt werden. Hierzu existiert eine Vielzahl von stochastischen Verfahren wie *Simulated Annealing*, *Threshold Accepting* und *Extremal Optimization*. Bei diesen Methoden werden Zufallswanderer im Zustandsraum zum Grundzustand hin gesteuert. Dies erlaubt die Beschreibung der Verfahren mit Hilfe von MARKOV-Ketten sowie die Aufstellung von Master-Gleichungen, welche die Dynamik der Wanderer modellieren.

Auf Basis dieser Beschreibung wurde ein analytischer Beweis durchgeführt, der innerhalb der Klasse der durch *Extremal Optimization* gegebenen Optimierungsverfahren eine bestmögliche Implementation charakterisiert. Dazu wurden Performanzkriterien eingeführt, die quantitative Vergleiche verschiedener Verfahren zulassen.

Die resultierende Methode *Fitness Threshold Accepting* hängt von einem *Fitness Schedule* ab. Der Verlauf dieses *Schedules* kann aus dem Beweis nicht abgeleitet werden; vielmehr muss dieser an das zu untersuchende komplexe System angepasst werden. Hierzu wurde eine auf diskreter optimaler Kontrolltheorie basierende Methode vorgestellt, und auf kleine ISING-Spinsysteme angewandt. Es wurden optimale *Fitness Schedules* berechnet, welche die Wahrscheinlichkeit, den Grundzustand des Systems im Verlauf der Optimierung erreicht zu haben, maximieren, sowie die mittlere *best-so-far*-Energie minimieren. Die *Schedules* zeigten sehr ungewöhnliche Abhängigkeiten von den Zeitschritten. Jedoch fällt die Wahrscheinlichkeit, den Grundzustand *nicht* zu erreichen, sowie die mittlere *best-so-far*-Energie exponenziell ab. Dies bedeutet insbesondere, dass es sich bei der vorgeschlagenen Methode tatsächlich um einen hochperformanten Optimierungsalgorithmus handelt.

Die Charakterisierung einer Optimierungsaufgabe kann durch die Approximation der Lösungs- bzw. Zustandsdichte erfolgen. Insbesondere kann diese Abschätzungen erlauben, wie schwierig es sein wird, die optimale Lösung bzw. den Grundzustand zu finden. Darüber hinaus erlaubt die Kenntnis der Zustandsdichte eines komplexen Systems sofort Aussagen zur Temperaturabhängigkeit von gleichgewichtsthermodynamischen Messgrößen. Nach einem Überblick über vorhandene Methoden wurde das Verfahren von WANG und LANDAU detailliert untersucht. Bei dieser Methode wird ein flaches Histogramm über die möglichen Energien des zu untersuchenden Systems erzeugt, mithin eine uneingeschränkte Zufallswanderung über diese Energien realisiert. Es zeigte sich, dass die

Herangehensweise von WANG und LANDAU die größtmöglichen Wahrscheinlichkeitsflüsse zwischen Mikrozuständen realisiert. Es scheint deshalb das beste Verfahren innerhalb der Flache-Histogramm-Methoden im Hinblick auf eine schnellstmögliche Konvergenz zu sein. Hierzu wurden numerische Indizien sowie Ansätze für einen entsprechenden analytischen Beweis gegeben.

Jedoch ist dieser Algorithmus auf komplexe Systeme mit vielen lokalen Minima nur bedingt anwendbar, da entsprechende Zufallswanderer lange in solchen Minima verweilen. Eine Approximation der Zustandsdichte bei kleinen Energien, die einer Suche nach Grundzuständen entspricht, wird somit numerisch aufwendig. Bei der vorgeschlagenen matrixbasierten Alternative *parQ* wird versucht, diesen Nachteil zu umgehen. Hier wird das Optimierungsverfahren *Simulated Annealing* benutzt, um zunächst mit Hilfe von Zufallswanderern Übergänge zwischen Energieniveaus vorzuschlagen, die dann durch eine Eigenvektorberechung ausgewertet werden. Diese Methode wurde an einigen der größten komplexen ISING-Spinsysteme getestet, für die zurzeit exakte Zustandsdichten bekannt sind. Eine sehr hohe Genauigkeit des Verfahrens konnte nachgewiesen werden.

Weiterhin zeigte sich ein Vorteil des Verfahrens bei der Umsetzung auf Parallelrechner. Die Methode *parQ* lässt sich aufgrund der voneinander unabhängigen Zufallswanderer äußerst leicht und effizient auf Cluster von Computern bzw. Rechenknoten verteilen. Dies wurde durch *Speedup*-Messungen an entsprechenden Prozessen belegt, bei denen mehrere hundert Rechenknoten benutzt wurden. Einige Messungen zeigten superlinearen *Speedup*, der durch Cache-Effekte auf den Prozessoren erklärt werden kann.

Erklärung gemäß Promotionsordnung vom 10. Oktober 2001, §6(2)4,5

Ich versichere hiermit, die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt zu haben. Ich erkläre außerdem, nicht bereits früher oder zur gleichen Zeit bei anderen Hochschulen oder an dieser Universität ein Promotionsverfahren beantragt zu haben. Die Promotionsordnung der Fakultät für Naturwissenschaften der Technischen Universität Chemnitz erkenne ich an.

Chemnitz, 26. Juli 2005

Frank Heilmann

Lebenslauf

Name:	Frank Heilmann
geboren am:	18. Juni 1977
Familienstand:	verheiratet, ein Kind

Schulbildung

1984 – 1991	Polytechnische Oberschule Neuwürschnitz
1991 – 1993	Spezialschule Chemnitz, mathematisch-naturwissenschaftlich-technische Richtung
1993 – 1996	Carl-von-Bach-Gymnasium Stollberg/E. Abschluss: Abitur mit Prädikat „sehr gut“ (1,4)

Studium und wissenschaftlicher Werdegang

1997 – 2002	Studium der Physik an der Technischen Universität Chemnitz, ab Hauptstudium gefördert durch die Studienstiftung des deutschen Volkes Abschluss: Diplom in Physik mit Prädikat „Mit Auszeichnung bestanden“ (1,0) Thema der Diplomarbeit: „Molekulardynamische Simulationen von mechanischen Deformationen“
Herbst 2000	Forschungsaufenthalt an der Universität Paderborn, AG Frauenheim
ab 2002	Doktorand an der Technischen Universität Chemnitz an der Professur „Theoretische Physik, insbesondere Computerphysik“, gefördert durch ein Promotionsstipendium der Edgar-Heinemann-Stiftung an der Technischen Universität Chemnitz

Publikationen

Heilmann, F.,
Molekulardynamische Simulationen von mechanischen Deformationen,
Diplomarbeit, TU Chemnitz, August 2002

Heilmann, F., Hoffmann, K. H. and Salamon, P.,
Best possible probability distribution over extremal optimization ranks,
Europhysics Letters **66**(3):305–310, 2004

Hoffmann, K. H, Heilmann, F. and Salamon, P.,
Fitness threshold accepting over extremal optimization ranks,
Physical Review E **70**(4):046704, 2004

Heilmann, F. and Hoffmann, K. H.,
ParQ – high-precision calculation of the density of states,
Europhysics Letters **70**(2):155–161, 2005

Tagungen / Workshops / Seminare

Heilmann, F.
Bestimmung der elastischen Eigenschaften kleiner Strukturen durch Vergleich von molekulardynamischen und kontinuumsmechanischen Rechnungen
Vortrag auf der Frühjahrstagung der DPG, Regensburg, März 2002

Heilmann, F.
Finding Minima in Complex State Spaces via Extremal Optimization
Vortrag auf dem Workshop "Relaxation Phenomena in Complex Systems", Chemnitz, März 2004

Seminare des SFB 393 „Parallele Numerische Simulation für Physik und Kontinuumsmechanik", TU Chemnitz