Martin Raifer, BSc

Quantum Monte Carlo Analysis of the Heisenberg Chain

MASTER THESIS

for obtaining the academic degree
Diplom-Ingenieur

Master Programme of
Technical Physics

Graz University of Technology

Supervisor:
Ao.Univ.-Prof. Dr. Hans Gerd Evertz
Institut für Theoretische Physik - Computational Physics

Graz, July 2014

EIDESSTATTLICHE ERKLÄRUNG

AFFIDAVIT

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Masterarbeit identisch.

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis.

Datum / Date

Unterschrift / Signature

Abstract

The *Heisenberg* model is investigated using a Quantum Monte Carlo method. The Heisenberg model represents the interaction of quantum mechanical spins on a lattice using a linear nearest neighbour approximation. An efficient Monte Carlo method to simulate this model is found to be the *Directed Loop* algorithm which operates in the *world line* representation of the partition sum. It is discussed how (using a *perturbation expansion*) this algorithm can be extended to simulate more complex extensions of the Heisenberg model, for example the *Spin-Peierls* model which studies the interaction between the spins and lattice vibrations (*phonons*).

Kurzfassung

Das *Heisenberg*-Modell wird mit Hilfe einer Quanten-Monte-Carlo Methode untersucht. Das Heisenberg-Modell beschreibt die Wechselwirkung von quantenmechanischen Spins auf einem Gitter mit Hilfe einer linearen, nächste-Nachbar Näherung. Eine effiziente Monte-Carlo Methode um solche Modelle zu studieren ist der sogenannte *Directed-Loop* Algorithmus. Dieser arbeitet in der *Weltlinien*-Darstellung der Zustandssumme. Untersucht wird in dieser Arbeit, wie mit Hilfe einer *Störungsentwicklung* dieser Algorithmus erweitert werden kann, um komplexere Systeme basierend auf dem Heisenberg-Modell zu untersuchen, zum Beispiel das *Spin-Peierls*-Modell welches die Wechselwirkung zwischen Spins and Gitterschwingungen (*Phononen*) beschreibt.

Contents

1. Introduction

In physics, often simplified models of natural objects are studied to learn about their behaviour and to better understand real phenomena. One such model is the Heisenberg model, which approximates the interactions of quantum-mechanical spins on a lattice. This model concentrates on magnetic effects such as (anti-) ferromagneticity. Sometimes, these simple models are extended and combined with other models to study how different effects interact and change the overall behaviour of the systems. Using this approach one can potentially understand complex behaviour in real life materials, for example.

Despite their simplicity, most of the properties of these systems are too complicated to be investigated using exact calculations, especially for "large" system sizes. Therefore, often stochastical approaches are used instead. One popular choice for studying such models are Quantum Monte Carlo methods. They are relatively simple to formulate, give some direct insight to the systems to be studied and are also relatively easy and straight-forward to extend to more complex models.

This thesis discusses how the Heisenberg model can be simulated using a Quantum Monte Carlo method and how this simulation can be extended to simulate more complex systems.

2. Quantum Spin Chain Models

2.1. The Heisenberg Model

The Heisenberg model describes a lattice of quantum mechanical spins that are coupled with a nearest neighbour interaction. In its most general for it can be written as follows:

$$\hat{H} = \sum_{<i,j>} J^x \hat{S}_i^x \hat{S}_j^x + J^y \hat{S}_i^y \hat{S}_j^y + J^z \hat{S}_i^z \hat{S}_j^z - \sum_i \mathbf{B} \cdot \hat{\mathbf{S}}_i \tag{2.1}$$

Here, $\sum_{<i,j>}$ stands for the sum over all possible nearest neighbour pairs, $J^{(\cdot)}$ is the coupling constant in the respective direction and the $S_i^{(\cdot)}$ are the components of the standard spin $\frac{1}{2}$ operator at location i. $\mathbf{B}$ is an external magnetic field.

The Heisenberg model describes magnetic lattices which show ferromagnetic behaviour for negative couplings J and antiferromagnetic behaviour for $J > 0$.

Often special, less general forms of this Hamiltonian are analysed. In this work we will leave out external fields and only consider isotropic couplings: $J^x = J^y = J^z \equiv J$. Than, the model can be written as:

$$\hat{H} = J \sum_{<i,j>} \hat{\mathbf{S}}_i \hat{\mathbf{S}}_j \tag{2.2}$$

Or, after applying the transformation $\hat{S}^{\pm} = \hat{S}^x \pm i\hat{S}^y$:

$$\hat{H} = J \sum_{<i,j>} \frac{1}{2} \left(\hat{S}_i^+ \hat{S}_j^- + \hat{S}_i^- \hat{S}_j^+ \right) + \hat{S}_i^z \hat{S}_j^z \tag{2.3}$$

Furthermore, we will consider linear spin chains with periodic boundary conditions, which can be written in the above equations by replacing the sums with $\sum_{i=1}^{N}$ and the

substitution $j \to i+1$:

$$\hat{H} = J \sum_{i=1}^{N} \hat{\mathbf{S}}_i \hat{\mathbf{S}}_{i+1} \tag{2.4}$$

The periodic boundary conditions are ensured when $\hat{S}_{N+1} = \hat{S}_1$.

2.2. The Spin-Peierls Model

The Heisenberg model can be expanded to simulate so called Spin-Peierls Transition.[1] In this model, additional degrees of freedom are introduced which describe lattice vibrations (phonons). These are then coupled to the spins of the Heisenberg model:

$$\hat{H} = \sum_{i=1}^{N} f(\hat{x}_i) \hat{\mathbf{S}}_i \hat{\mathbf{S}}_{i+1} + \frac{1}{2} \sum_{q} \hat{p}_q^2 + \omega^2(q) \hat{x}_q^2 \tag{2.5}$$

$f(\hat{x}_i)$ describes the spin-phonon coupling, q is a reciprocal lattice vector and $\omega(q)$ is the bare dispersion of the non-interacting phonons.

The bare phonon dispersion relation $\omega(q)$ can be chosen to reflect the internal structure of the crystal to be simulated. Often, one only considers one of the following two different types:

- Dispersion-less ("Einstein") phonons

  $$\omega(q) = \omega_0 \tag{2.6}$$

  A constant dispersion relation is often taken because it is relatively easy to handle in numerical and analytical calculations. It can also be seen as an approximation of an optical phonon branch.

- Acoustic phonons

  $$\omega(q) = \frac{\omega(\pi)}{\sqrt{2}} \sqrt{1 - \cos(q)} \tag{2.7}$$

  This is the (acoustic) phonon branch of a one-dimensional single-atom chain with nearest neighbour interaction: $\sim (x_i - x_{i+1})^2$.

Also for the spin-phonon interaction different types of interaction can be simulated:

- site phonons

$$f(x_i) = 1 + g(x_i - x_{i+1}) \tag{2.8}$$

  This represents a system where the spins live on the same *sites* as the phonons. Thus, the distances between neighbouring spins fluctuate as phonons pass by and so does the respective spin-spin interaction. A simple linear model results in the above equation.

- bond phonons

$$f(x_i) = 1 + gx_i \tag{2.9}$$

  For this model, the spin positions are held constant, while other atoms affecting the coupling perform transverse oscillations. This changes the super-exchange coupling along the respective *bond*.

With this model at hand, one can investigate how different couplings between spins and phonons modify the phonon dispersion relation and properties of the uncoupled Heisenberg Hamiltonian.

3. Quantum Monte Carlo Simulations

Quantum Monte Carlo (QMC) methods can be used to simulate systems with strong correlations. It is applicable to a wide variety of quantum many particle systems and doesn't rely on systematic approximations. QMC methods can be used if the system doesn't show the so called sign problem. Errors are only of statistical nature.

The first part of this chapter closely follows the review paper about world line quantum monte carlo methods by F. F. Assaad and H. G. Evertz [2] while the later sections are influenced by the dissertations of Franz Michel [1] and Peter Pippan [3]. Towards the end of this chapter, own modifications of the simulation algorithms are discussed.

3.1. World Line Representation

$$\langle \hat{O} \rangle = \frac{1}{Z} \operatorname{Tr} \left(\hat{O} e^{-\beta \hat{H}} \right) \tag{3.1}$$

This defines an observable $\langle \hat{O} \rangle$ for the hamiltonian $\hat{H}$ and the inverse temperature β. To make this observable accessible, we will transform the trace into a classical partition sum from which we can draw configurations in an ordinary Monte-Carlo scheme. For this we will ultimately introduce an additional dimension to the system: the imaginary time axis.

Let us consider the isotropic Heisenberg model, as described in equation 2.4. We will use the set of basis vectors produced by the tensor product of N spin $\frac{1}{2}$ states:

$$|\boldsymbol{\sigma}\rangle = |\sigma_1, \sigma_2, \ldots, \sigma_N\rangle \tag{3.2}$$

Now, we will split the hamiltonian into two parts, each of which consists only of terms

which commute with each other:

$$\begin{aligned}\hat{H} &= \hat{H}_e + \hat{H}_o \\ &= J \sum_{i \text{ even}} \hat{\mathbf{S}}_i \hat{\mathbf{S}}_{i+1} + J \sum_{i \text{ odd}} \hat{\mathbf{S}}_i \hat{\mathbf{S}}_{i+1}\end{aligned} \tag{3.3}$$

The matrix elements of the individual parts $\hat{H}_e$ and $\hat{H}_o$ of this hamiltonian are easily evaluated, because they consist only of independent two-particle problems. The partition sum can be divided into a sequence of successive applications of the $\hat{H}_e$ and $\hat{H}_o$.

$$\begin{aligned}\mathrm{Tr}\,(e^{-\beta\hat{H}}) &= \mathrm{Tr}\,(e^{-\Delta\tau\hat{H}})^M \\ &= \mathrm{Tr}\,(e^{-\Delta\tau(\hat{H}_e+\hat{H}_o)})^M \\ &= \mathrm{Tr}\,(e^{-\Delta\tau\hat{H}_e}e^{-\Delta\tau\hat{H}_o})^M + \mathcal{O}(\Delta\tau)\end{aligned} \tag{3.4}$$

Here, the system is split into M slices of *imaginary time* $\Delta\tau$ (with $\Delta\tau M = \beta$). In the last step a Suzuki-Trotter decomposition is used to split the terms. This introduces an error in the order of $\Delta\tau$, which we will later bring down to zero when we perform the limit $\Delta\tau \to 0$.

We can now write out the trace and add unity operators $\sum_{|\boldsymbol{\sigma}\rangle} |\boldsymbol{\sigma}\rangle \langle\boldsymbol{\sigma}|$ between each imaginary time slice:

$$\mathrm{Tr}\,(e^{-\beta\hat{H}}) = \sum_{|\boldsymbol{\sigma}_1\rangle,\ldots,|\boldsymbol{\sigma}_{2M}\rangle} \underbrace{\langle\boldsymbol{\sigma}_{2M}|e^{-\Delta\tau\hat{H}_e}|\boldsymbol{\sigma}_1\rangle \langle\boldsymbol{\sigma}_1|e^{-\Delta\tau\hat{H}_o}|\boldsymbol{\sigma}_2\rangle \ldots \langle\boldsymbol{\sigma}_{2M-1}|e^{-\Delta\tau\hat{H}_o}|\boldsymbol{\sigma}_{2M}\rangle}_{W_{WC}} + \mathcal{O}(\Delta\tau) \tag{3.5}$$

For each possible permutation of $|\boldsymbol{\sigma}_1\rangle \ldots |\boldsymbol{\sigma}_{2M}\rangle$, which we will call a *world line* configuration (see below), one gets an associated weight W_{WC}. With this one can get observables (3.1) by evaluating $\langle O\rangle = \frac{\sum_{WC} O_{WC} W_{WC}}{\sum_{WC} W_{WC}}$ using a classical Monte Carlo approach.

Let us take a look at one of the matrix elements: $\langle\boldsymbol{\sigma}_1|e^{-\Delta\tau\hat{H}_o}|\boldsymbol{\sigma}_2\rangle$. Since it is a sum of independent two-particle problems, we can write:

$$\langle\boldsymbol{\sigma}_1|e^{-\Delta\tau\hat{H}_o}|\boldsymbol{\sigma}_2\rangle = \prod_{i \text{ odd}} \langle\sigma_{1,i}, \sigma_{1,i+1}|e^{-J\Delta\tau\hat{\mathbf{S}}_i\hat{\mathbf{S}}_{i+1}}|\sigma_{2,i}, \sigma_{2,i+1}\rangle \tag{3.6}$$

This means that the calculation of the total weight only depends on the solution of the two-particle matrix elements (and of course on the individual state of the world lines). Here are the solutions for the isotropic Heisenberg Hamiltonian:

$$\begin{aligned} a &= \langle\uparrow\uparrow|\, e^{-J\Delta\tau\hat{\mathbf{S}}_1\hat{\mathbf{S}}_2}\, |\uparrow\uparrow\rangle = \langle\downarrow\downarrow|\, e^{-J\Delta\tau\hat{\mathbf{S}}_1\hat{\mathbf{S}}_2}\, |\downarrow\downarrow\rangle = e^{-\Delta\tau J/4} \\ b &= \langle\uparrow\downarrow|\, e^{-J\Delta\tau\hat{\mathbf{S}}_1\hat{\mathbf{S}}_2}\, |\downarrow\uparrow\rangle = \langle\downarrow\uparrow|\, e^{-J\Delta\tau\hat{\mathbf{S}}_1\hat{\mathbf{S}}_2}\, |\uparrow\downarrow\rangle = -e^{\Delta\tau J/4}\sinh(\Delta\tau J/2) \\ c &= \langle\uparrow\downarrow|\, e^{-J\Delta\tau\hat{\mathbf{S}}_1\hat{\mathbf{S}}_2}\, |\uparrow\downarrow\rangle = \langle\downarrow\uparrow|\, e^{-J\Delta\tau\hat{\mathbf{S}}_1\hat{\mathbf{S}}_2}\, |\downarrow\uparrow\rangle = e^{\Delta\tau J/4}\cosh(\Delta\tau J/2) \end{aligned} \tag{3.7}$$

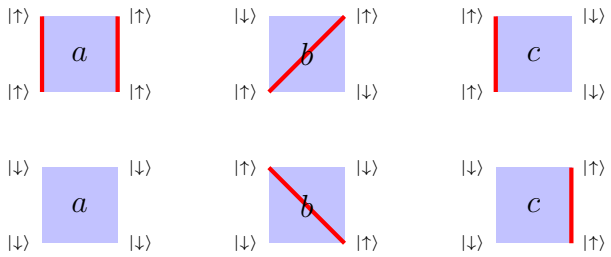

Figure 3.1.: Graphical representation of the matrix elements in equation 3.7. Spin-up sites are connected with bold (world) lines.

All other matrix elements are zero which means that the respective spin configurations cannot be part of a valid configuration, because the total world line configuration weight would be zero as well.

The negative sign of the matrix elements b can be got rid of by performing a π rotation about the S^z axis for spins on one sub-lattice of a bipartite lattice.[4]

3.1.1. The World Line Picture

There is a graphical representation of the above mentioned method of transforming the quantum mechanical trace into a classical partition sum, which makes the term *world line* more obvious.

Instead of applying the whole hamiltonian at once, we apply it M times for an *imaginary time*[1] span of $\Delta\tau$. In each of these slices we apply first one half of the Hamiltonian then the other half. The first part $\hat{H}_e$, by construction, only affects the interaction between spins along even numbered bonds (e.g. spins 2 and 3, 4 and 5, etc.). The second half does the same for the odd spin interactions.

[1] The term *imaginary time* comes from the similarity of the time propagation operator $e^{-it\hat{H}}$ with the Boltzmann-factor $e^{-\beta\hat{H}}$.

We can draw this as a chessboard where shaded sites indicate the active operators at each time step. For a particular configuration state $|\boldsymbol{\sigma}_1\rangle \dots |\boldsymbol{\sigma}_{2M}\rangle$ one can connect (for example) all spin-up states along the imaginary time evolution and finally get to *world lines*. The world lines exchangeably define a spin configuration.

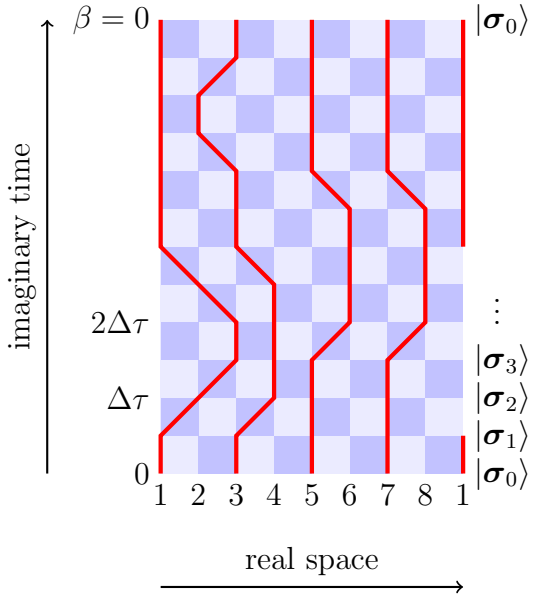

Figure 3.2.: An example world line configuration. Shaded squares indicate acting two-site operators. Sites in a spin-up state are connected via bold world lines.

Note that the world lines are continuous lines because of the conservation of S^z of the Heisenberg Hamiltonian 2.1. There are also periodic boundary conditions in space (introduced by the model hamiltonian) and time (introduced by the trace in 3.1).

3.1.2. Monte Carlo Methods

Now we have to construct a Markov Chain Monte Carlo method to draw random configurations with respect to the global weight (see equation 3.5).

A simple approach would be a simple local Metropolis-Hastings algorithm like the following: Start at some configuration C, propose a new configuration C' (based on small modifications of the current state). Now accept the new configuration with probability $\sim W(C')/W(C)$ or stay with the old configuration.

This approach is suboptimal, because on the one hand if one tries to propose new configurations that are largely different from the previous state, one will get extremely

low acceptance probabilities (and thus very large autocorrelation times). If, on the other hand, one only proposes small changes between the configurations, one will still get large autocorrelation times because the world lines need very long simulation times to change their shape significantly.

This means that we need a better algorithm to draw new configurations for the Markov Chain. The Loop Algorithm [2] and its variants are cluster algorithms that achieve this goal. Particularly, in this thesis the *Directed Loop* algorithm is used, a thorough description of which can be found in [4].

The general idea of the Directed Loop method is to work in an extended configuration space. Not only closed world lines are allowed, but additionally also exactly one "loose" world line (i.e. one source and one drain - which correspond to a $\hat{S}^+$ and a $\hat{S}^-$ operator acting at particular sites and imaginary times). At the start of each update, one puts a source/drain pair at some random site at some random time. Then, one repeatedly applies *local* Monte Carlo updates (that is, the position of one of the operators S^+ or S^- is changed by flipping neighbouring spins). The partial world line created by the movement of this point is called a worm. Note that the updates are still made according to detailed balance of the weights of the overall model to be studied. When at some time the two worm ends meet each other again, the source and drain operators annihilate and what remains is a new configuration for the main simulation. Because in this process the worm can scan over large areas, the new configuration will potentially be much different from the previous one, thus the autocorrelation times of the simulation will be small.

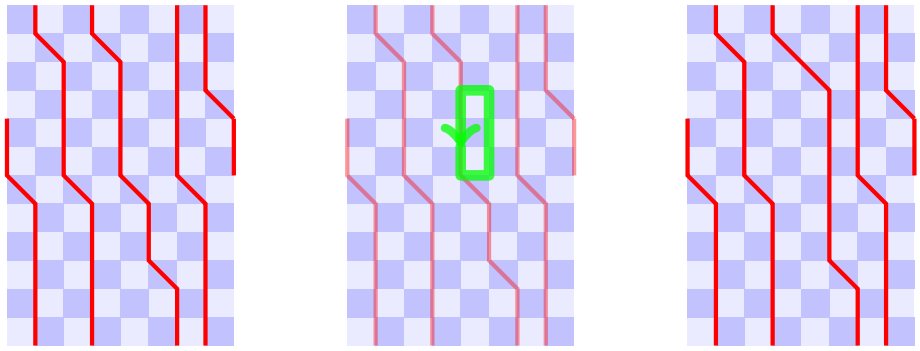

Figure 3.3.: A worm alters a world line configuration: On the left the initial world line configuration is shown. In the centre, one worm has been created on this configuration (the worm's end point is denoted by the arrow head). And on the right the final resulting world lines is displayed.

For the movement of the worm head, which has to satisfy the global weights, one possibility would be to choose new spin configurations according to the heat-bath probability $W(C_{\text{new}})/\sum_i W(C_i)$. Here $W(C_{\text{new}})$ is the weight (3.7) of a particular new configuration and the sum $\sum_i$ functions as a normalization factor. This has the drawback of relatively large probabilities for the worm head to backtrack (that is a particular spin configuration to remain the same), thus getting relatively small worms and longer simulation times. But, one has more options to fulfil detailed balance. Reference [4] shows how a set of probabilities can be constructed that minimize the probability of backtracking. This method is called *Directed Loops*. For the isotropic Heisenberg model without magnetic fields, backtracking can be avoided altogether by using the following movement rules:

- If the world line configuration at the worm head position is of type a (3.7), convert it to a configuration of type c with unit probability by flipping the spin at the worm head position and moving along vertically (in the imaginary time direction).
- If the world line configuration at the worm head position is of type b, convert it to a configuration of type c with unit probability by flipping the spin at the worm head position and moving along horizontally (and flipping the worm head's time direction).
- If the world line configuration at the worm head position is of type c, convert it to a configuration of type b with probability $\tanh(\Delta\tau J/2)$ by flipping the spin at the worm head position and moving along horizontally (and flipping the worm head's time direction). Otherwise, convert the configuration into type a by moving along vertically (in the imaginary time direction).

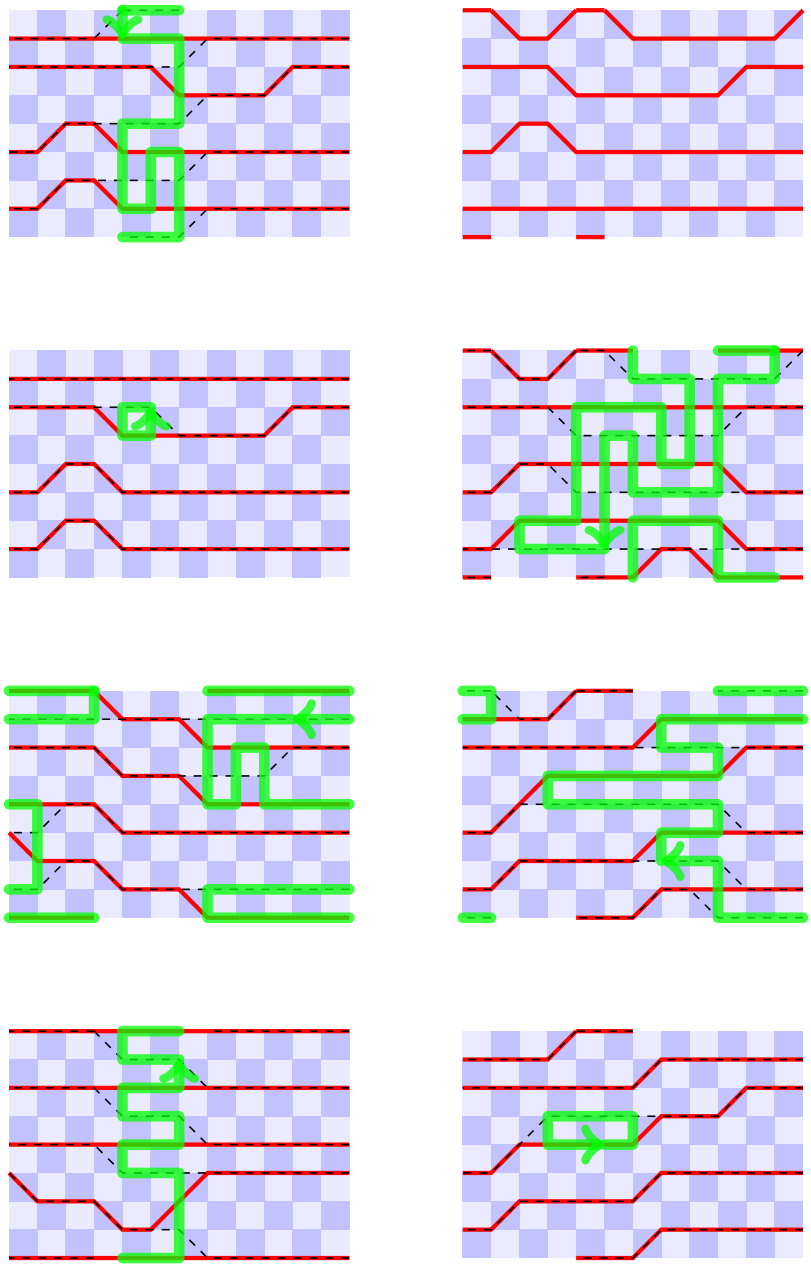

Figure 3.4.: A series of worms. The initial world line state at each iteration is shown in red, the worm in green and the resulting world line configuration as dashed black lines (which is the starting configuration for each following iteration).

3.2. Continuous Time Limit

In order to get more accurate results, we want to take the limit $\Delta\tau \to 0$. Instead of spins on a space-time grid we will store only the sites and imaginary times where a spin flips. As the probability for such a spin flip (i.e. a configuration of type b in equation 3.7) is proportional to $\Delta\tau$, there is only a finite number of such *kinks* to be stored even in the continuous time limit $\Delta\tau \to 0$.

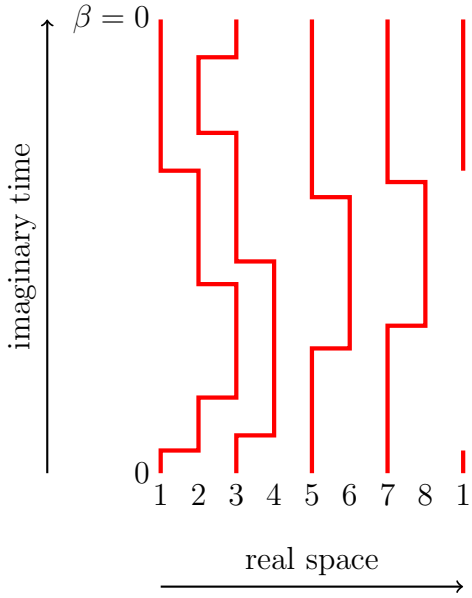

Figure 3.5.: An example world line configuration in continuous time.

The Directed Loop algorithm depicted above can also be adapted to continuous time: The decision to introduce a new kink (i.e. converting a configuration of type c to one of type b) corresponds to a Poisson process (the probability density per time is constant). Distances between Poisson events are exponentially distributed, therefore we can randomly draw a time at which a new kink can be inserted. Additionally, we must also fulfil the remaining conditions of the algorithm, i.e. that a new kink cannot be inserted if the neighbouring spin is equal to the current spin (this corresponds to a type a configuration in the discrete time model), and that in cases where the worm head hits an already existing kink, we must follow along it horizontally.

New world line configurations are constructed in the following manner:

1. Choose a random site, imaginary time and direction. Insert a worm head and worm end point there.

2. Calculate the imaginary time τ_{end} of the next kink (or the worm tail) on the current site in the chosen direction.

3. For each neighbour of the current site, propose a specific worm head move:
   a) Draw a random number $\delta\tau$ from an exponential distribution with rate $J/2$.
   b) If no kink is allowed because of the neighbour's spin configuration, advance the worm head to the neighbour's next kink time.
   c) If the worm head position moved by $\delta\tau$ would cross the initially calculated τ_{end}, propose to follow a kink at τ_{end}.
   d) Calculate the imaginary time distance to the neighbour's next kink.
   e) If $\delta\tau$ is smaller than this distance, propose a new kink at the worm head's time plus $\delta\tau$.
   f) Otherwise subtract this distance from $\delta\tau$ and continue at 3b.

4. Choose the worm move which is the closest to the start point in terms of imaginary time.

5. If the worm head has hit its tail, abort with a new world line configuration.

6. Insert new kinks at the the proposed time on the current site and the respective neighbour. Move the worm head's position to the neighbour site and reverse its direction. Continue at 2.

This algorithm is implemented by storing a doubly linked list of kinks for each lattice site. Additionally, each spin's state at imaginary time $\tau = 0$ is stored. To get the state of a particular spin at a particular imaginary time, one has to transverse this linked list up to the desired time. This introduces an additional cost of $\mathcal{O}(\beta J)$ for each such lookup. But, of course, other variants of this representation are also possible. For example, if one would store the world line configuration as a complete graph of interlinked kinks, instead of a set of independent links, the above mentioned additional cost for neighbour site lookups could be omitted.

3.2.1. Measurements

Measurements of observables that only contain diagonal operators $\hat{S}^z$ (such as the correlation function $\langle s_i^z s_j^z \rangle$ or the energy $\langle \hat{H} \rangle$) can be measured directly by slicing

through the world line configuration at some value of imaginary time τ and simply read by the values of the spins (see equation 3.2) at each site. To improve the statistics, several measurements at different values of τ can be performed after a single Monte Carlo update.

Similarly, also the greens function $\langle s_i^z(\tau) s_j^z(0)\rangle$ can be measured directly. For practical reasons, these measurements are done on a finite set of imaginary times. The spacing Δt of these times limits the maximum energy that can be resolved $E_{\max} \sim 1/\Delta t$ in the measurement, but it does not otherwise affect the accuracy of the simulation.

Measurements of the propagator $\langle \hat{S}^+(\tau)\hat{S}^-(0)\rangle$ are done *during* each Monte Carlo update, during the construction of the worm (see chapter 3.1.2): Given the worm head at site i and imaginary time τ_1 and the worm end at site j and time τ_2, we get one contribution to the propagator $\langle \hat{S}_i^+(\tau_2 - \tau_1)\hat{S}_j^-(0)\rangle$. These measurements are typically also done using a discrete time grid.

3.3. Simulating the Spin-Peierls Model

The above described Monte Carlo algorithms can be easily generalized and expanded to simulate more complicated models, for example the Spin-Peierls Model (introduced in chapter 2.2):

$$\hat{H} = \sum_{i=1}^{N} f(\hat{x}_i, \hat{x}_j)\hat{\mathbf{S}}_i\hat{\mathbf{S}}_{i+1} + \frac{1}{2}\sum_i \hat{p}_i^2 + \omega_0^2\hat{x}_i^2 \tag{3.8}$$

In order to simulate this model with its additional degrees of freedom (i.e. the phonon elongations x_i), one performs two steps in each Monte Carlo update: First, one holds all phonon elongations constant and updates the spins using the Directed Loop method. Second, one holds all spins constant and performs an update of the phonon degrees of freedom. The two steps are independent of each other and are described in chapters 3.3.2 and 3.3.3, respectively. Together, these two steps make sure that the overall Markov chain is ergodic.

A general introduction of this algorithm can be found in the dissertation of Franz Michel [1], where a perturbation expansion of the partition sum is used to simulate the Spin-Peierls model. The idea was revisited in the dissertation of Peter Pippan [3] in which the Holstein model (i.e. spin-less fermions coupled to dispersion-less phonons)

is studied using a world-line QMC method.

3.3.1. Phonon Subspace

The phonon elongations are stored straight forwardly in the real space basis x_i. Just like the spins, the elongations are also extended into the imaginary time space using a path integral representation:

$$Z = \mathrm{Tr}_s \int \mathcal{D}\boldsymbol{x} \exp\left(- \int_0^\beta \hat{H}[\{x_i(\tau)\}] \mathrm{d}\tau \right) \tag{3.9}$$

Here, Tr_s stands for the trace over all spin states and $\int \mathcal{D}\boldsymbol{x}$ is the path integral over all phonon configurations $\{x_i(\tau)\}$. $\hat{H}[\{x_i(\tau)\}]$ is the Hamiltonian operator for the respective phonon configuration, where the position operators $\hat{x}_i$ are replaced by the classical variables $x_i(\tau)$ and $p_i(\tau)$.[5]

Unlike for the spins, it is not possible to find a continuous time representation of the phonon subspace. Therefore, the imaginary time is discretised into a grid with some small spacing $\delta\tau$.

This introduces a systematic error to the simulation which has to be checked to be acceptably small after simulating (for example by comparing different simulation runs with varying $\delta\tau$). An alternative representation of the phonon subspace would involve storing it in momentum and frequency space. Then, a maximum cut-off frequency has to be introduced to perform actual simulations (as described in reference [1]). This does also introduce similar errors to the simulation.

3.3.2. Spin Updates

The spin updates work quite similarly to what has been described in chapter 3.2: When one compares the simple Heisenberg model (equation 2.2) with the Spin-Peierls model (3.8) here, and one notes that for the spin update all phonon elongations are held constant, one immediately sees that the phonons only alter the coupling J to some effective coupling $J_{\mathrm{eff}}(\tau) := f(x_i(\tau), x_j(\tau))$.

As the effective coupling is now a function of the imaginary time τ, we cannot continue to simply draw new times from an exponential distribution in the Directed Loop algorithm. This is because the local probability density to insert a new kink is

not constant any more, but a function of imaginary time. Reference [3] shows how this hurdle can be overcome by drawing new times from a more complex probability distribution. This distribution looks like the following:

$$p(\tau' \to \tau)\mathrm{d}\tau = \lambda(\tau) e^{-\int_{\tau'}^{\tau} \lambda(t)dt} \mathrm{d}\tau \tag{3.10}$$

Here τ' and τ are the worm head's start and end points and $\lambda(t)$ is the local probability density for introducing a kink at time t. Given τ and $\lambda(t)$ (which is a function of the phonon coordinates $x_i(t)$ and spins $s_i(t)$), we need to draw a new imaginary time τ'. This can be achieved by using the rejection method: One draws random numbers τ_p from a simplified distribution $m(\tau_p) = \lambda_{\max} e^{-\lambda_{\min}\tau_p}$ (which in this case is just an exponential distribution with a modified normalization factor) and accepts the proposed number with probability $p(\tau_p)/m(\tau_p)$, otherwise the whole step is repeated. For this to work, $\lambda(\tau) > 0$ needs to hold for every τ.

When one carries this out for the Spin-Peierls model (for example with bond-phonons $f(x_i) = 1 + g x_i$) in the Directed Loop algorithm, the local probability density is equal to $\lambda(\tau) = \frac{1}{2} f(x_i(\tau)) V(\tau)$. The function V indicates whether a kink towards a specific neighbour at a specific imaginary time is allowed ($V = 1$) or not ($V = 0$). As one can see, there are regions in imaginary time where λ regularly becomes zero. This renders the rejection method described above inapplicable (since $\lambda_{\min} = 0$).

Therefore, a different approach is used to draw times from distribution 3.10: Imagine the imaginary time axis was mapped to a new time variable $\bar{\tau}$, which has all regions where $V(\tau)$ is zero are cut out. Also, the new time variable would be stretched or compressed locally proportionally to the value of $\lambda(\tau)$. As is shown below this can be done in a way such that the probability to add a new kink is now constant again in the variable $\bar{\tau}$. Then, we can simply draw an exponentially distributed time for $\bar{\tau}$ and map it back to the initial imaginary time τ.

We define $\bar{\tau}$ to be equal to the exponent in equation 3.10 (for simplicity, we set the worm start time to zero in the following derivation):

$$\bar{\tau}(\tau) := \int_0^{\tau} \lambda(t)\mathrm{d}t \tag{3.11}$$

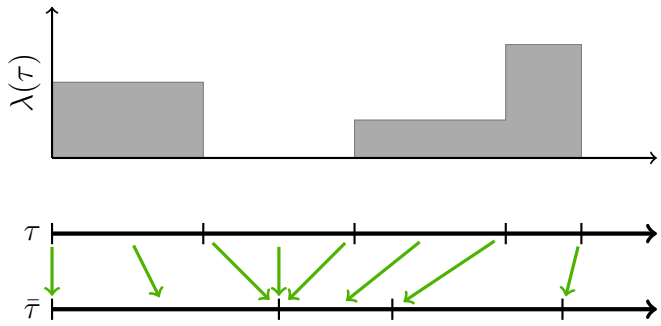

Figure 3.6.: The imaginary time τ is mapped onto a new time scale $\bar{\tau}$ according to the local probability density $\lambda(\tau)$.

The differential for this reads

$$\frac{\mathrm{d}\bar{\tau}}{\mathrm{d}\tau} = \lambda(\tau) \Rightarrow \mathrm{d}\tau = \frac{\mathrm{d}\bar{\tau}}{\lambda(\tau)} \tag{3.12}$$

We substitute the new variable in equation 3.10 and get

$$\begin{aligned} p(\tau)\mathrm{d}\tau &= \lambda(\tau) e^{-\int_0^\tau \lambda(t)\mathrm{d}t} \mathrm{d}\tau \\ &= \cancel{\lambda(\tau)} e^{-\bar{\tau}} \mathrm{d}\bar{\tau} / \cancel{\lambda(\tau)} \\ p(\bar{\tau})\mathrm{d}\bar{\tau} &= e^{-\bar{\tau}} \mathrm{d}\bar{\tau} \end{aligned} \tag{3.13}$$

Thus, we can draw $\bar{\tau}$ from a basic exponential distribution. All we have left to do is to transform the random value back to the imaginary time scale τ by inverting equation 3.11 numerically.

3.3.3. Phonon Updates

For the phonon updates we will generally follow the method introduced in references [1] and [3]. The idea is to write the partition sum using an imaginary time path integral for the phonon degrees of freedom and a perturbation expansion for the spin degrees of freedom. As can be shown (see appendix A), both expansions can be applied at the

same time and lead to the following partition sum:

$$\begin{aligned} Z = & \sum_{n=0}^{\infty} \sum_{|\alpha\rangle} \sum_{T_n} \int_0^{\beta} d\tau_1 \int_0^{\tau_1} d\tau_2 \ldots \int_0^{\tau_n - 1} d\tau_n \int \mathcal{D}\boldsymbol{x} \\ & \langle \alpha | \prod_{l=0}^{n} f\left(x_{k_l^1}(\tau_l), x_{k_l^2}(\tau_l) \right) \frac{1}{2} \hat{\mathrm{V}}_{k_l^1, k_l^2}(\tau_l) \, |\alpha\rangle \\ & e^{-\int_0^{\beta} \mathrm{d}\tau \sum_i \frac{1}{2} \left(p_i(\tau)^2 + \omega_0^2 x_i(\tau)^2 \right)} \end{aligned} \tag{3.14}$$

Here we sum over all possible operator sequences T_n. Each operator in the sequence has an associated time τ_l and two sites k_l^1 and k_l^2 at which the individual operators $\hat{V}$ act. The actual operators follow from an alternative notation of the Heisenberg Hamiltonian:

$$-\hat{\mathbf{S}}_i \hat{\mathbf{S}}_j + \frac{1}{4} = \frac{1}{2} \left(|\uparrow\downarrow\rangle + |\downarrow\uparrow\rangle \right) \left(\langle\uparrow\downarrow| + \langle\downarrow\uparrow| \right) \tag{3.15}$$

Each of the four non-zero matrix element of this hamiltionian represents a situation where a worm either created a new kink in the world line representation or annihilated a previously existing kink.

Just as in reference [1], the perturbation expansion of the partition sum is formally done in both the off-diagonal and diagonal parts of the Hamiltonian.

In the language of the deterministic loop algorithm [6], the stochastic step takes place when inserting or deleting the "diagonal" operators, whereas the "non-diagonal" updates are then already determined. Here, we use the language of the Loop algorithm [2] directly and use operator (3.15). This results in the same loop updates. Another difference to reference [1] is SSE versus continuous time:

We use a continuous time world line representation to do our Spin updates instead of an SSE (stochastic series expansion, [7]) method. For a in-depth discussion about how the SSE and continuous time world line methods are related via the perturbation expansion, see reference [1], chapter 2.

The time integral in equation 3.14 will now be approximated by a discrete sum $\int \mathrm{d}\tau \to \sum \delta\tau$ (with $\delta\tau = \beta/N$) in our case (see chapter 3.3.1). As a next step, the momentum $p_i(\tau)$ is replaced by the difference quotient $(x_i(\tau) - x_i(\tau + \delta\tau))/\delta\tau$ and a Fourier transformation of the phonon coordinates $x_i(\tau) \to \tilde{x}_{k,n}$ and spins $s_i(\tau) \to \tilde{s}_{k,n}$

is performed. The kinetic term of the Hamiltonian now becomes

$$\begin{aligned}\sum_{i,\tau} \frac{1}{2} \left(\frac{x_i(\tau) - x_i(\tau + \delta\tau)}{\delta\tau} \right)^2 &= \frac{N^2}{2\beta^2} \sum_{k,n} \left| 1 - e^{-2\pi i n/N} \right|^2 |\tilde{x}_{k,n}|^2 \\ &= \frac{N^2}{2\beta^2} \sum_{k,n} 4 \sin^2 \left(\frac{\pi n}{N} \right) |\tilde{x}_{k,n}|^2 \end{aligned} \tag{3.16}$$

The exponential in equation 3.14 contains now only terms which are up to second order in $x_{k,n}$ (note that we hold the spins constant during the phonon update). This means that the we could draw Gaussian distributed values for each $x_{k,n}$ (with mean and variance depending on the pre-factors of the terms $x_{k,n}^2$ and $x_{k,n}$). But, there is still the additional multiplicative term $f(x_{k_l^1}(\tau_l), x_{k_l^2}(\tau_l))$ next to the operators $\hat{V}$ in the spin-trace part of the partition sum that has to be accounted for in the simulation. These terms can formally be raised to the exponential:

$$Z = \ldots \exp \left(- \int \ldots \mathrm{d}\tau + \sum_{l=1}^{N} \ln \left(f(x_{k_l^1}(\tau_l), x_{k_l^2}(\tau_l)) \right) \right) \tag{3.17}$$

One can now expand the logarithm up to second order, so that the exponent stays quadratic. We use this form to draw proposals for new phonon configurations, which are later accepted or rejected using a traditional Metropolis-Hastings step:

One does the Fourier transformation as mentioned before, calculates new pre-factors for the $x_{k,n}$ above and thus gets different parameters for the Gaussians to draw from (see reference [1], Appendix C for detailed instructions about how to calculate them). The series expansion of the logarithm does of course introduce unwanted uncontrollable systematic errors to the simulation. But this problem can be overcome by adding a final Metropolis-Hastings step after drawing a new phonon configuration by comparing the quotients of the approximated and real partition sums before and after a specific phonon configuration proposal. A detailed description of this algorithm can be found in reference [1] (chapter 3.3.1.1) and shall not be repeated here.

4. Results

The simulation algorithms described earlier in this thesis have been implemented using the C++ ALPS (Algorithms and Libraries for Physics Simulations – http://alps.comp-phys.org/) framework. In particular, the program is based on a stripped-down version of ALPS' *worm* application [8] and includes portions of code developed by Peter Pippan in [3] and Franz Michel in [1].

The simulations were continuously tested and verified against analytical calculations (for small systems performing a full diagonalization is sometimes possible) and numerical results found in external research papers. Later, the simulation was used to measure the Greens function of the systems and to derive spectral functions from that.

Unfortunately, due to time constraints, it was not possible to carry out verified simulations of the full Spin-Peierls model, therefore Greens function and thus spectral functions could not be measured for it. The results in this chapter are presented solely for the isotropic antiferromagnetic Heisenberg model.

4.1. Scaling Relation

Here, a simulation was carried out to measure equal-time correlation functions $\langle \hat{S}_i^z \hat{S}_j^z \rangle$. The measurements were made to be compared with results from DMRG calculations in reference [9].

The measured correlation functions are normalized using the following steps:

$$\begin{aligned} \omega(l) &= (-1)^l \langle \hat{S}_n^z \hat{S}_{n+l}^z \rangle \\ \bar{\omega}(l) &= \frac{1}{4}[\omega(l-1) + 2\omega(l) + \omega(l+1)] \end{aligned} \tag{4.1}$$

According to reference [9] this function satisfies the scaling relation

$$\frac{\bar{\omega}(l)|_N}{f(l/N)} = \bar{\omega}(l)|_{N=\infty} \tag{4.2}$$

where the scaling function is given by

$$f(x) = \left(1 + 0.288\,22 \sinh^2(1.673x)\right)^{1.805} \tag{4.3}$$

In figure 4.1, the quantity $l\,\bar{\omega}(l)/f(l/N)$ is plotted over l for different system sizes ($N \in \{50, 70, 90\}$). The simulations were carried out at an inverse temperature of $\beta = 512$ (i.e. at very low temperature).

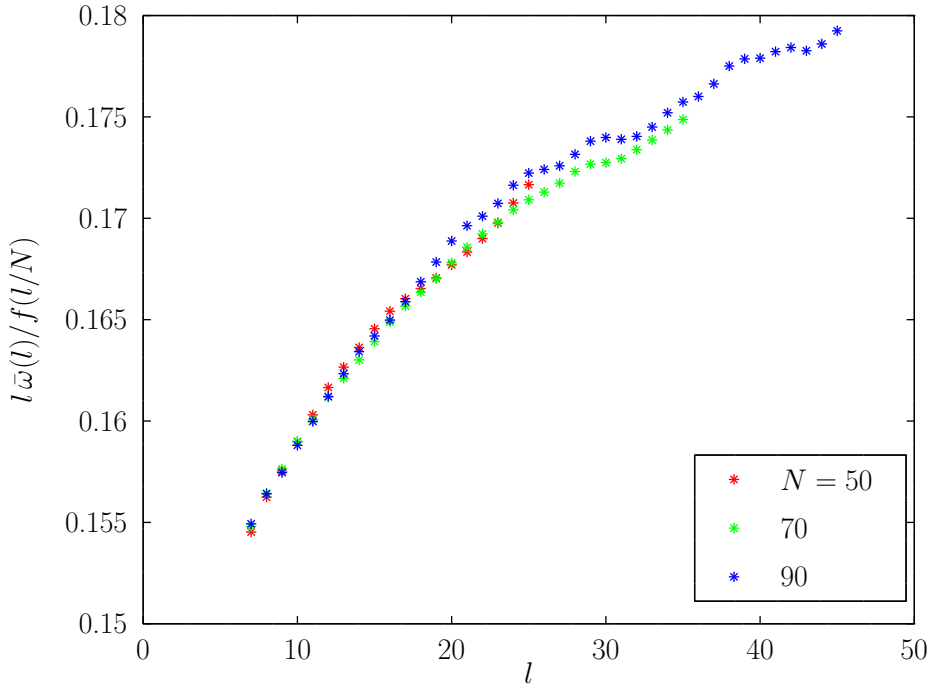

Figure 4.1.: The measured correlation function, as a scaled quantity $l\,\bar{\omega}(l)/f(l/N)$ for different system sizes N.

The simulations have been performed using approximately $4\,10^6$ worm updates. about 5% of these updates were used in a thermalisation phase. Measurements were taken after each eighth spin update.

The results turn out to be in good accordance with the findings in reference [9] (see figure 4.2).

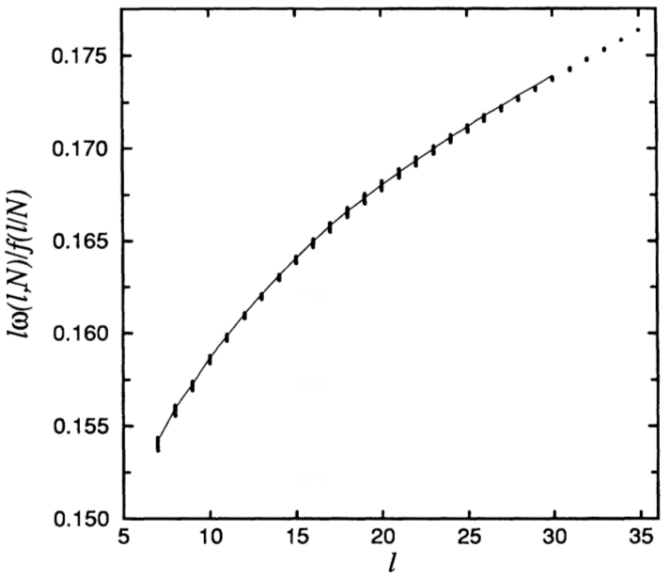

Figure 4.2.: The original results of the scaling analysis by Hallberg et al. in [9].

4.2. Spectral Function

The spectral density function (see reference [10]) is an important quantity which can be used to describe excitation processes of a system, for example. Unfortunately, this property can not be measured directly during a Quantum Monte Carlo calculation. But it is closely related to the Greens function via the *spectral theorem*:

$$G_{\hat{O},\hat{O}'}(\tau) = \int_{-\infty}^{\infty} \frac{e^{-\tau\omega}}{e^{-\beta\omega} - \epsilon} A^{\epsilon}_{\hat{O},\hat{O}'}(\omega)\,\mathrm{d}\omega + C(\epsilon) \tag{4.4}$$

$G_{\hat{O},\hat{O}'}$ is the Greens function of an Operator pair $(\hat{O}, \hat{P})$ and $A_{\hat{O},\hat{O}'}$ is the spectral density function for that. ϵ defines whether the operators do follow bosonic ($\epsilon = 1$) or fermionic ($\epsilon = -1$) rules. C is a constant which can occur in the bosonic case when degenerate states contribute to the observable.

We want to measure the spectral function of the spin creation and annihilation operators $\hat{S}^+\hat{S}^-$. Here, $\epsilon = 1$. The spectral theorem reads

$$\langle \hat{S}^+_{-k}(\tau)\hat{S}^-_k(0)\rangle = \int_{-\infty}^{\infty} \frac{e^{-\tau\omega}}{e^{-\beta\omega} - 1} S_{+-}(k,\omega)\,\mathrm{d}\omega \tag{4.5}$$

This equation can be inverted numerically using a probabilistic algorithm, the so called maximum entropy method.[11] Measuring the greens function is done during the worm update (see section 3.2.1).

The result is shown in figure 4.3, where a simulation of a system with $N = 64$ was performed. As in the previous chapter, the simulation used approximately 10^6 iterations (worm updates) of which about 5% is used for thermalization and measurements were taken after every eighth step.

The results turn out to be compatible with the findings in reference [12] (see figure 4.4) which were obtained using a *Bethe ansatz* calculation.

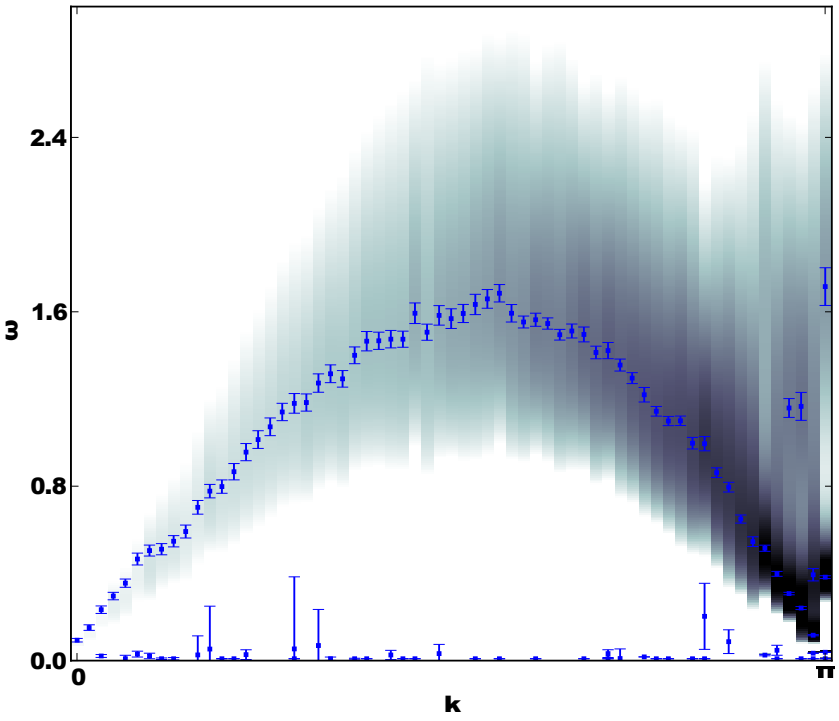

Figure 4.3.: The measured spectral function of the antiferromagnetic Heisenberg model.

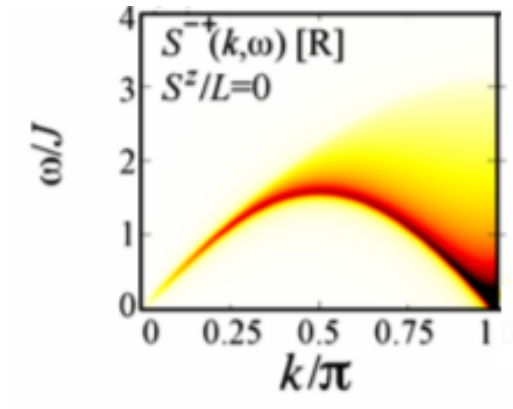

Figure 4.4.: The original findings of the spectral function analysis of the Heisenberg chain by Kohno in [12].

5. Conclusions

The Heisenberg chain was investigated using the directed loop algorithm. The general approach to this kind of quantum Monte Carlo simulation is shown in chapter 3; the directed loop algorithm specifically is explained in chapter 3.2. The results of these simulations can be found in chapter 4 and turned out to be in good accordance with existing research.

Chapter 3.3 explains how the algorithm can be extended to simulate the Spin-Peierls model which is an extension of the Heisenberg model that includes phonon degrees of freedom. In particular it is discussed how the Monte Carlo update can be split into two steps: a spin update and a phonon update. The algorithms for these two updates are explained in chapters 3.3.2 and 3.3.3, respectively.

A. Perturbation Expansion and Path Integrals

The partition sum $Z = \operatorname{Tr} e^{-\beta \hat{H}}$ can be written in a perturbation expansion [13]:

$$Z = \sum_{n=0}^{\infty} (-1)^n \int_0^{\beta} \mathrm{d}\tau_1 \int_0^{\tau_1} \mathrm{d}\tau_2 \ldots \int_0^{\tau_{n-1}} \mathrm{d}\tau_n \operatorname{Tr}\left(e^{-\beta \hat{D}} \hat{V(\tau_1)} \hat{V(\tau_2)} \ldots \hat{V(\tau_n)} \right) \tag{A.1}$$

Here the Hamiltonian is split into two parts $\hat{H} = \hat{D} + \hat{V}$ where $\hat{D}$ is diagonal in a basis $|\boldsymbol{\sigma}\rangle$ and $\hat{V}(\tau) = e^{\tau \hat{D}} \hat{V} e^{-\tau \hat{D}}$.

If $\hat{V}$ is composed of smaller terms (for example nearest neighbour interactions of the Heisenberg model), we can write this using a sum over all possible permutations of the individual terms. For this we define

$$\begin{aligned} &\hat{V} = \sum_{b=1}^{M} \hat{V}_b \\ &T_n = (b_1, \ldots, b_n) \quad b_i \in [1, M] \end{aligned} \tag{A.2}$$

and write down the trace in the basis $|\boldsymbol{\sigma}\rangle$.

$$Z = \sum_{n=0}^{\infty} \sum_{|\boldsymbol{\sigma}\rangle} \sum_{T_n} \int_0^{\beta} \mathrm{d}\tau_1 \int_0^{\tau_1} \mathrm{d}\tau_2 \ldots \int_0^{\tau_{n-1}} \mathrm{d}\tau_n \, (-1)^n \, e^{-\beta E_0} \prod_{p=1}^{n} e^{-\tau_p (E_p - E_{p-1})} \, \langle \boldsymbol{\sigma} | \prod_{p=0}^{n} \hat{V}_{b_p} | \boldsymbol{\sigma} \rangle \tag{A.3}$$

The E_p result from inserted complete basis sets $|\boldsymbol{\sigma}(p)\rangle$ between the operators $\hat{V}_{b_p}$ and $E_p = \langle \boldsymbol{\sigma}(p) | \hat{D} | \boldsymbol{\sigma}(p) \rangle$. By defining $\Delta\tau_p = \tau_p - \tau_{p+1}$ one can see that the exponential

pre-factors already look like a path integral:

$$Z = \sum \dots \int \dots (-1)^n \underbrace{\prod_{p=0}^{n} e^{-\Delta\tau_p E_p}}_{e^{-\sum_{p=0}^{n} \Delta\tau_p E_p} \to e^{-\int_0^\beta E(\tau)\mathrm{d}\tau}} \langle \boldsymbol{\sigma}| \prod_{p=0}^{n} \hat{V}_{b_p} |\boldsymbol{\sigma}\rangle \tag{A.4}$$

Now, one can use this perturbation expansion also on models that include further operators acting on degrees of freedom which require a path integral expansion, such as the phonon degrees of freedom of the Spin-Peierls model. Then, the terms E_p above will still contain these operators. To get rid of them, one can apply a path integral expansion [5] on each of the exponential terms $e^{-\Delta\tau_p \hat{E}_p}$ (see equation 3.9).

$$\langle \boldsymbol{x}| e^{-\Delta\tau_p \hat{E}_p} |\boldsymbol{x}'\rangle \to \int_{(\boldsymbol{x}',\tau_{p+1})}^{(\boldsymbol{x},\tau_p)} \mathcal{D}\boldsymbol{x} \exp\left(- \int_{\tau_{p+1}}^{\tau_p} E\left[\boldsymbol{x}(\tau)\right] \mathrm{d}\tau \right) \tag{A.5}$$

One can combine all individual path integrals to get the full picture:

$$Z = \sum_{n=0}^{\infty} (-1)^n \sum_{|\boldsymbol{\sigma}\rangle} \sum_{T_n} \int_0^\beta \mathrm{d}\boldsymbol{\tau} \int \mathcal{D}\boldsymbol{x}\ e^{-\int_0^\beta E[\boldsymbol{x}(\tau),\boldsymbol{\sigma}(\tau)]\mathrm{d}\tau} \langle \boldsymbol{\sigma}| \prod_{p=0}^{n} \hat{V}_{b_p}[\boldsymbol{x}(\tau_p)] |\boldsymbol{\sigma}\rangle \tag{A.6}$$

Here, $\int_0^\beta \mathrm{d}\boldsymbol{\tau}$ is a short hand notation for the time integrals in equation A.3 and $\int \mathcal{D}\boldsymbol{x}$ is the path integral over all trajectories $x_i(\tau)$.

Acknowledgments

I would really like to send out some huge thanks to all the people without whom completing this thesis certainly wouldn't have been possible for me:

- *To my family for their absolute support and love.*
- *To my sister especially for her drawing in the front pages of this thesis.*
- *To my flatmates and friends for their warm company.*
- *To my faculty's dean of studies for his motivating talks.*
- *To my supervisor for his endless patience and advice.*

Bibliography

[1] Franz Michel. *Phonon Spectra at Quantum Phase Transitions of Spin-Peierls Systems*. PhD thesis, Graz University of Technology, 2006.

[2] F. F. Assaad and H. G. Evertz. World line and determinantal quantum monte carlo methods for spins, phonons, and electrons. *Springer Lecture Notes in Physics*, 739, 2008.

[3] Peter Pippan. *Quantum Monte Carlo Simulations of Bosonic Systems*. PhD thesis, Graz University of Technology, 2010.

[4] Olav F. Syljuåsen and Anders W. Sandvik. Quantum monte carlo with directed loops. *Physical Review E*, 66:046701, Oct 2002.

[5] Hans Gerd Evertz. Lecture notes: Fortgeschrittene quantenmechanik.

[6] Anders W. Sandvik. Stochastic series expansion method with operator-loop update. *Physical Review B*, 59:R14157–R14160, Jun 1999.

[7] Anders W. Sandvik and Juhani Kurkijärvi. Quantum monte carlo simulation method for spin systems. *Physical Review B*, 43:5950–5961, Mar 1991.

[8] F. Alet et al., Report cond-mat/0410407, submitted to J. Phys. Soc. Jpn. / M. Troyer, B. Ammon and E. Heeb, Lecture Notes in Computer Science, Vol. 1505, p. 191 (1998).

[9] Karen A. Hallberg, Peter Horsch, and Gerardo Martínez. Numerical renormalization-group study of the correlation functions of the antiferromagnetic spin-1/2 heisenberg chain. *Physical Review B*, 52:R719–R722, Jul 1995.

[10] Wolfgang Nolting. *Grundkurs Theoretische Physik 7: Viel-Teilchen-Theorie*. Springer, 2009.

[11] W. von der Linden. Maximum-entropy data analysis. *Applied Physics A: Materials Science & Processing*, 60(2):155–165, 1995.

[12] Masanori Kohno. Dynamically dominant excitations of string solutions in the spin-1/2 antiferromagnetic heisenberg chain in a magnetic field. *Physical Review Letters*, 102:037203, Jan 2009.

[13] A. W. Sandvik, R. R. P. Singh, and D. K. Campbell. Quantum monte carlo in the interaction representation: Application to a spin-peierls model. *Physical Review B*, 56:14510–14528, Dec 1997.