A new Cluster Algorithm for the Ising Model

Diplomarbeit
der Philosophisch-naturwissenschaftlichen Fakultät
der Universität Bern

vorgelegt von
Matthias Nyfeler

November 2005

Leiter der Arbeit:
Prof. Dr. U.-J. Wiese

Institut für theoretische Physik, Universität Bern

Abstract

Using D-theory we construct a new efficient cluster algorithm for the Ising model. The construction is very different from the standard Swendsen-Wang algorithm and related to worm algorithms. With the new algorithm we have measured the correlation function with high precision over a surprisingly large number of orders of magnitude.

There is nothing better for a man,
than that he should eat and drink,
and that he should make his soul enjoy good in his labour.

Ecclesiastes 2,24

Contents

Introduction

Numerical methods are a widely used and irreplaceable tool for solving complex systems and theories. For example, in condensed matter systems as well as in non-perturbative particle physics, analytical methods often fail and thus numerical methods, such as Monte Carlo simulations, are very often the only way to tackle these theories.

Even a simple system, such as the Ising model, can be analytically solved only in one or two dimensions, whereas in the 3-dimensional model one has to rely upon numerical simulations such as the Metropolis algorithm [1] or the much more efficient cluster algorithms [2].

Cluster algorithms are a powerful tool for efficient simulations of statistical systems and have been very successful in numerically solving some interesting field theoretic models [3]. These algorithms have the potential to drastically reduce or even eliminate critical slowing down. The Swendsen-Wang [4] and Wolff [5] algorithms for the Ising model were the first cluster algorithms, but soon this idea has been extended to other classical and quantum spin systems [6, 7] with continuous symmetries. However, the success of cluster algorithms has been limited to spin models without frustration. It has not yet been possible to develop cluster algorithms for Wilson's formulation of gauge theories, but there is hope to reach that goal in the framework of D-theory.

D-theory is an alternative non-perturbative approach to quantum field theory formulated in terms of discrete quantized variables instead of classical fields [8]. Classical scalar fields are replaced by generalized quantum spins and classical gauge fields are replaced by quantum links. It is obvious that discrete variables are much easier to handle in numerical simulations. This method allows us to develop cluster algorithms. It can also be used to describe and simulate simple spin models.

For this purpose, we reanimated the cluster algorithm for the $U(1)$ gauge theory studied in the PhD thesis of Tsapalis [9], which suffers from an inefficiency in updating a certain winding number. We soon realised that there is still a lot we have not yet understood when it comes to cluster algorithms in general. We made a survey of all available cluster algorithms in all different models and bases of the physical Hilbert space. That way we got a table spanning from spin models such as the Ising model to gauge theories like the $U(1)$ gauge theory.

With this table in hand, where the Ising model was on the very top of the list, we suspected that it should be possible to construct an algorithm for the Ising model in a different basis using the ideas borrowed from D-theory. Thus we returned to this simplest

of all spin-models and reformulated it as a quantum model with an additional Euclidean time dimension. We did this in a different basis of the Hilbert space than the basis the standard Swendsen-Wang algorithm is constructed in. In this way we were able to develop a new efficient cluster algorithm which is very different from the Swendson-Wang or Wolff algorithms. In our basis we could also reformulate our cluster algorithm as a worm algorithm and managed in this way to easily measure the correlation function over more than hundred orders of magnitude. We performed these measurements in the one-, two- and three-dimensional Ising model and we have also constructed an improved estimator for the susceptibility. As expected, this method works correctly and even proves to be highly efficient.

Also this algorithm can be set up as a meron-cluster algorithm [10, 11] in the standard basis. We have also constructed this algorithm in order to understand better the technical challenges of algorithms in different bases.

In this work we review the classical Ising model and its solution in chapter 1 and then resolve it from the perspective described above in chapter 2 and 3. In chapter 4 we reconstruct the algorithm in the standard basis as a meron-cluster algorithm. The results are presented in chapter 5.

Chapter 1

The Ising Model

1.1 Standard Formulation and Observables

The Ising model is the simplest classical spin model with variables $s_x = \pm 1$. The spins live on a d-dimensional lattice (below we mainly consider $d = 1$ and $d = 2$). Its Hamilton function is a sum of nearest neighbour contributions

$$\mathcal{H}[s] = -J \sum_{\langle xy \rangle} s_x s_y - \mu B \sum_x s_x, \tag{1.1}$$

with a ferromagnetic coupling constant $J > 0$ that favours parallel spins (with $J < 0$ we would have an antiferromagnet) and an external magnetic field B. The classical partition function is then given by

$$Z = \sum_{[s]} \exp(-\beta \mathcal{H}[s]) = \sum_{s_1=\pm 1} \sum_{s_2=\pm 1} \cdots \sum_{s_N=\pm 1} \exp(-\beta \mathcal{H}[s]) = \prod_x \sum_{s_x=\pm 1} \exp(-\beta \mathcal{H}[s]), \tag{1.2}$$

with the number of sites $N = L^d$. The sum over all spin configurations corresponds to an independent summation over all possible orientations of individual spins. Thermal averages are computed by inserting appropriate observables. The magnetisation $\mathcal{M}$ is equal to

$$\mathcal{M}[s] = \sum_x s_x, \tag{1.3}$$

and its thermal expectation value is given by

$$\langle \mathcal{M} \rangle = \frac{1}{Z} \prod_x \sum_{s_x=\pm 1} \mathcal{M}[s] \exp(-\beta \mathcal{H}[s]). \tag{1.4}$$

The magnetic susceptibility is defined as

$$\chi = \frac{1}{L^d} \left[\langle \mathcal{M}^2 \rangle - \langle \mathcal{M} \rangle^2 \right] = \frac{\partial^2 \log Z}{\partial (\beta \mu B)^2}. \tag{1.5}$$

With no external magnetic field, it can also be defined as the sum over the correlation function (because $\langle \mathcal{M} \rangle = 0$)

$$\chi = \frac{1}{L^d}\langle \mathcal{M}^2 \rangle = \frac{1}{L^d}\langle \sum_x s_x \sum_y s_y \rangle = \frac{1}{L^d}\sum_{x,y}\langle s_x s_y \rangle = \sum_x \langle s_0 s_x \rangle. \tag{1.6}$$

Similarly, the spin correlation function is defined as

$$\langle s_x s_y \rangle = \frac{1}{Z}\prod_x \sum_{s_x=\pm 1} s_x s_y \exp(-\beta \mathcal{H}[s]), \tag{1.7}$$

which at large distances decays exponentially

$$\langle s_x s_y \rangle - \langle s_x \rangle \langle s_y \rangle \sim \exp(-|x-y|/\xi), \tag{1.8}$$

where ξ is the so-called correlation length. The average internal energy per spin — the energy density — is defined as

$$E = \frac{1}{L^d d}\langle \mathcal{H} \rangle = -\frac{1}{L^d d}\frac{\partial}{\partial \beta}\log Z = -\frac{1}{L^d dZ}\sum_{[s]} \mathcal{H}[s]\exp(-\beta \mathcal{H}[s]). \tag{1.9}$$

Without an external magnetic field B, it can be rewritten in terms of the correlation function

$$\begin{aligned} E &= -\frac{J}{L^d dZ}\sum_{[s]}\sum_{\langle xy \rangle} s_x s_y \exp(-\beta \mathcal{H}[s]) = -\frac{J}{L^d d}\sum_{\langle xy \rangle}\frac{1}{Z}\sum_{[s]} s_x s_y \exp(-\beta \mathcal{H}[s]) \\ &= -\frac{J}{L^d d}\sum_{\langle xy \rangle}\langle s_x s_y \rangle = -J\langle s_0 s_1 \rangle. \end{aligned} \tag{1.10}$$

1.2 Analytic Results

In the following sections and chapters all the calculations and simulations will be done without an external magnetic field, i.e. $B = 0$.

1.2.1 Ising Model in one Dimension

In a 1-dimensional system of size L with periodic boundary conditions $s_0 = s_L$, we have the Hamilton function

$$\mathcal{H} = -J\sum_{i=1}^{L} s_i s_{i+1}, \tag{1.11}$$

which contributes to the partition function

$$Z = \sum_{[s]} \exp(-\beta H) = \sum_{[s]}\prod_{i=1}^{L} \exp(\beta J s_i s_{i+1}). \tag{1.12}$$

Every factor $\exp(\beta J s_i s_{i+1})$ can be written as a matrix element of an operator — the transfer matrix T — in a two-dimensional vector space with the basis

$$\begin{aligned} &| s\rangle \ (s = \pm 1), \ \langle s \mid s' \rangle = \delta_{ss'}, \\ T = &\begin{pmatrix} \langle +1 \mid T \mid +1 \rangle & \langle +1 \mid T \mid -1 \rangle \\ \langle -1 \mid T \mid +1 \rangle & \langle -1 \mid T \mid -1 \rangle \end{pmatrix} = \begin{pmatrix} e^{\beta J} & e^{-\beta J} \\ e^{-\beta J} & e^{\beta J} \end{pmatrix}. \end{aligned} \tag{1.13}$$

This implies that we can express the partition function as a trace

$$Z = \sum_{s_1 \ldots s_L} \langle s_1 \mid T \mid s_2 \rangle \langle s_2 \mid T \mid s_3 \rangle \ldots \langle s_L \mid T \mid s_1 \rangle = \sum_{s_1} \langle s_1 \mid T^L \mid s_1 \rangle = \mathrm{Tr}\ T^L. \tag{1.14}$$

Using a unitary transformation that diagonalises the transfer matrix, we obtain

$$T = UDU^\dagger \ \Rightarrow D = \begin{pmatrix} 2c & 0 \\ 0 & 2s \end{pmatrix}, \tag{1.15}$$

where

$$U = U^\dagger = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \tag{1.16}$$

and $c = \cosh(\beta J), \ s = \sinh(\beta J)$. The partition function is then obtained as

$$Z = \mathrm{Tr}\ T^L = \mathrm{Tr}(UDU^\dagger)^L = \mathrm{Tr}\ UD^L U^\dagger = \mathrm{Tr}\ D^L = (2c)^L + (2s)^L. \tag{1.17}$$

In this formulation, the calculation of the two-point correlation function is straightforward (we use σ^3 as the spin operator at position x: $s_x \mathrel{\widehat{=}} \sigma^3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$)

$$\begin{aligned} \langle s_0 s_x \rangle &= \frac{1}{Z} \mathrm{Tr}(s_0 T^x s_x T^{L-x}) = \frac{1}{Z} \mathrm{Tr}(s_0 U D^x U^\dagger s_x U D^{L-x} U^\dagger) = \frac{1}{Z} \mathrm{Tr}(\sigma_0^1 D^x \sigma_x^1 D^{L-x}) \\ &= \frac{s^x c^{L-x} + s^{L-x} c^x}{s^L + c^L} = \frac{t^x + t^{L-x}}{1 + t^L}, \end{aligned} \tag{1.18}$$

where $t = \tanh(\beta J)$ and the Pauli matrix $\sigma_x^1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ $(U^\dagger s_x U = \sigma_x^1)$. In the infinite volume limit $L \to \infty$ the correlation function is given by

$$\langle s_0 s_x \rangle = \tanh^x(\beta J). \tag{1.19}$$

Thus the correlation length is given by

$$\xi = -\frac{1}{\log(\tanh(\beta J))}. \tag{1.20}$$

Since ξ diverges for $\beta \to \infty$, there is a second order phase transition at zero temperature.

The energy density, which corresponds to $-J\langle s_0 s_1 \rangle$, is given by

$$E = -\frac{1}{L} \frac{\partial}{\partial \beta} \log Z = -J \frac{s c^{L-1} + s^{L-1} c}{s^L + c^L} = -J \frac{t + t^{L-1}}{1 + t^L}. \tag{1.21}$$

The susceptibility can be calculated as the sum over the correlation function

$$\chi = \sum_{i=1}^{L} \langle s_0 s_i \rangle = \sum_{i=1}^{L} \frac{t^i + t^{L-i}}{1 + t^L} = \frac{(1 - t^L)(1 + t)}{(1 + t^L)(1 - t)} = \frac{1 - t^L}{1 + t^L} \exp(2\beta J). \tag{1.22}$$

1.2.2 The Ising Model in two Dimensions

While the 1-dimensional Ising model is quite simple to solve analytically, the 2-dimensional model is highly non-trivial. Still in 1944, Onsager found an exact solution [12].

In two dimensions, the Hamilton function can be written as

$$\mathcal{H} = -J \sum_{i,j} (s_{i,j} s_{i+1,j} + s_{i,j} s_{i,j+1}), \tag{1.23}$$

where the spins are now indexed by two indices corresponding to a point on the 2-dimensional lattice of size L^2. Introducing a short hand notation for $\mathcal{H}$,

$$\mathcal{H} = \sum_{j=1}^{L} (E(\mu_j, \mu_{j+1}) + E(\mu_j)), \tag{1.24}$$

where

$$E(\mu_j, \mu_k) = -J \sum_{i=1}^{L} s_{i,j} s_{i+1,j}, \tag{1.25}$$

and $\mu_j = \{s_1, \ldots, s_L\}$ defined as a set of spins in a column. Here $E(\mu_j, \mu_k)$ denotes the energy between one column j and its neighbour k and $E(\mu_j)$ denotes the energy within a given column. One can again define a transfer matrix T with the elements

$$\langle \mu_j \mid T \mid \mu_k \rangle = \exp(-\beta[E(\mu_j, \mu_k) + \frac{1}{2}(E(\mu_k) + E(\mu_j))]), \tag{1.26}$$

which is now a $2^L \times 2^L$ matrix. The partition function is again given by

$$Z = \text{Tr } T^L. \tag{1.27}$$

The transfer matrix T can again be diagonalised as

$$D = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_{2^L}), \tag{1.28}$$

which leads us to the partition function containing the eigenvalues λ_α

$$Z(\beta) = \sum_{\alpha=1}^{2^L} (\lambda_\alpha)^L. \tag{1.29}$$

In the infinite volume limit, the partition function is dominated only by the largest eigenvalue λ_{max}

$$\begin{aligned} (\lambda_{max})^L \leq Z \leq 2^L (\lambda_{max})^L \quad &\Rightarrow \quad \frac{1}{L} \log \lambda_{max} \leq \frac{1}{L^2} \log Z \leq \frac{1}{L} \log \lambda_{max} + \frac{1}{L} \log 2 \\ &\Rightarrow \quad \lim_{N \to \infty} \frac{1}{N} \log Z(\beta) = \lim_{L \to \infty} \frac{1}{L} \log \lambda_{max}, \end{aligned} \tag{1.30}$$

where $N = L^2$ denotes the number of sites. The solution by Onsager [12] is given by

$$\lim_{N\to\infty} \frac{1}{N} \log Z(\beta) = \frac{1}{2} \log(2\cosh^2(2\beta J)) + \frac{1}{2\pi} \int_0^\pi d\phi \log \frac{1}{2}(1 + \sqrt{1 - \kappa^2 \sin^2 \phi}), \quad (1.31)$$

where

$$\kappa = \frac{2}{\cosh(2\beta J) \coth(2\beta J)}. \quad (1.32)$$

The energy density is thus given by

$$E = -\frac{1}{N} \frac{\partial}{\partial \beta} \lim_{N\to\infty} \frac{1}{N} \log Z(\beta) = -J \coth(2\beta J)(1 + \frac{2}{\pi})\kappa' K_1(\kappa), \quad (1.33)$$

with the elliptic function

$$K_1(\kappa) = \int_0^{\pi/2} \frac{d\phi}{\sqrt{1 - \kappa^2 \phi^2}} \quad (1.34)$$

and

$$\kappa' = 2\tanh^2(2\beta J) - 1. \quad (1.35)$$

(These calculations are taken from the book "Statistical Mechanics" by Huang [13]).

Furthermore, the correlation function along a lattice diagonal can be calculated using a recursion relation due to Jimbo and Miwa [14].

1.2.3 Self-Duality in two Dimensions

Because the 2-dimensional Ising model is self-dual (see below), we can exactly calculate its critical temperature T_c for the phase transition that now occurs at non-zero temperature. For this purpose we introduce bond variables

$$b_{\langle xy \rangle} = s_x s_y = \pm 1. \quad (1.36)$$

These bond variables around an elementary lattice square with four sites w, x, y and z satisfy the constraint

$$b_{\langle wx \rangle} b_{\langle xy \rangle} b_{\langle yz \rangle} b_{\langle zw \rangle} = 1. \quad (1.37)$$

For each lattice square we introduce a variable $m_\square$ that implements this constraint via

$$\delta_{b_{\langle wx \rangle} b_{\langle xy \rangle} b_{\langle yz \rangle} b_{\langle zw \rangle}, 1} = \frac{1}{2} \sum_{m_\square = 0,1} (b_{\langle wx \rangle} b_{\langle xy \rangle} b_{\langle yz \rangle} b_{\langle zw \rangle})^{m_\square}. \quad (1.38)$$

We then introduce the dual lattice with sites $\tilde{x}$ at the centers of the squares $\square$. The variable $m_\square$ can then be interpreted as a spin variable

$$s_{\tilde{x}} = 1 - 2m_\square = \pm 1 \quad (1.39)$$

on the dual lattice. Summing over the bond variable $b_{\langle xy\rangle}$ on the original lattice induces an interaction between the dual spins $s_{\tilde{x}}$ and $s_{\tilde{y}}$ at the centers of the two squares $\square_{\tilde{x}}$ and $\square_{\tilde{y}}$ that share the bond $\langle xy\rangle$. We have

$$\sum_{b_{\langle xy\rangle}=\pm 1} \exp(\beta J b_{\langle xy\rangle}^{m_{\tilde{x}}+m_{\tilde{y}}}) = \exp(-\tilde{\beta}\tilde{h}(s_{\tilde{x}}s_{\tilde{y}})), \tag{1.40}$$

which defines a Hamilton function

$$\tilde{\mathcal{H}}[s] = \sum_{\langle\tilde{x}\tilde{y}\rangle} \tilde{h}(s_{\tilde{x}}s_{\tilde{y}}). \tag{1.41}$$

We then obtain

$$\begin{aligned} \exp(-\tilde{\beta}\tilde{h}(1,1)) = \exp(-\tilde{\beta}\tilde{h}(-1,-1)) = 2\cosh(\beta J) = 2c, \\ \exp(-\tilde{\beta}\tilde{h}(1,-1)) = \exp(-\tilde{\beta}\tilde{h}(-1,1)) = 2\sinh(\beta J) = 2s, \end{aligned} \tag{1.42}$$

where c and s are the same factors as above. In the original Ising model the ratio of the two Boltzmann factors was

$$\exp(-\beta h(1,-1))/\exp(-\beta h(1,1)) = \exp(-2\beta J) \tag{1.43}$$

with $h(s_x, s_y) = \exp(\beta J s_x s_y)$. Similarly, the ratio of the two dual Boltzmann factors is

$$\frac{\exp(-\tilde{\beta}\tilde{h}(1,-1))}{\exp(-\tilde{\beta}\tilde{h}(1,1))} = \tanh(\beta J) = \exp(-2\tilde{\beta}\tilde{J}). \tag{1.44}$$

This equation determines the coupling constant $\tilde{J}$ of the dual Ising model. When the original Ising model is in the high-temperature phase (βJ small), the dual Ising model is in the low-temperature phase ($\tilde{\beta}\tilde{J}$ large) and vice versa. The exact critical temperature $T_c = 1/k_B\beta_c$ for the second order phase transition of the 2-d Ising model follows from the self-duality condition

$$\tanh(\beta_c J) = \exp(-2\beta_c J). \tag{1.45}$$

The solution of this is equation is

$$\beta_c J = \frac{1}{2}\log(1+\sqrt{2}) \simeq 0.4406. \tag{1.46}$$

1.3 Monte Carlo Method

In a spin model one could calculate the partition function Z by calculating Boltzmann factors for all possible 2^{L^d} spin configurations. Already in a not too small system 2^{L^d} is an immense number, which makes this practically impossible (unless it can be analytically solved like in the 1-dimensional Ising model). That is where Monte Carlo simulations

enter the stage. The idea behind Monte Carlo is importance sampling. In other words, we generate a configuration C with a probability p_C which corresponds to the Boltzmann weight of C

$$p_C = \frac{1}{Z} \exp(-\beta \mathcal{H}[C]). \tag{1.47}$$

One generates a set of configurations, a so-called Markow chain

$$C^1 \to C^2 \to \cdots \to C^N,$$

in which one finds C^i with the appropriate probability p_{C^i}. An observable can then be measured by averaging over the set of configurations in the Markow chain

$$\langle \mathcal{O} \rangle = \lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} \mathcal{O}[C^i]. \tag{1.48}$$

Due to importance sampling, a moderately large set of N configurations is sufficient to get an accurate measurement. In order to generate configurations with the correct probability, one has to ensure that the transfer matrix $w[C \to C']$ has eigenvalue 1 for a stationary probability distribution with the eigenvector p_C in equilibrium. A sufficient but not necessary condition to achieve this is detailed balance

$$\frac{p_{C^i}}{p_{C^j}} = \frac{w[C^j \to C^i]}{w[C^i \to C^j]}. \tag{1.49}$$

Here $w[C^i \to C^j]$ denotes the probability to go from C^i to C^j. (Of course these transitions have to be properly normalised, i.e. $\sum_j w[C^i \to C^j] = 1$). With detailed balance it is easy to show that the above condition is met, i.e. p_C is indeed an eigenvector with eigenvalue 1

$$p_{C^j} = \sum_C w[C^i \to C^j] p_{C^i} = \sum_i p_{C^i} w[C^j \to C^i] \frac{p_{C^j}}{p_{C^i}} = p_{C^j} \sum_C w[C^j \to C^i] = p_{C^j}. \tag{1.50}$$

Furthermore we demand ergodicity, i.e.

$$w^n[C^i \to C^j] > 0, \ \forall \ C^i, \ C^j \, n < \infty. \tag{1.51}$$

Ergodicity ensures that it is possible to reach any possible configuration in a finite number of steps. If ergodicity is not satisfied, certain configurations can never be reached. Thus the algorithm does not sample the whole partition function and is not correct.

As a Monte Carlo algorithm generates a new configuration out of a given one, these configurations are not statistically independent. That is why one should consider the autocorrelation times to estimate the efficiency. The autocorrelation time τ of an observable $\mathcal{O}$ — i.e. the Monte Carlo time it takes to get to a statistically independent measurement — is measured by

$$\langle \mathcal{O}(t) \mathcal{O}(t + \Delta t) \rangle - \langle \mathcal{O}(t) \rangle \langle \mathcal{O}(t + \Delta t) \rangle \sim \exp(-\Delta t / \tau). \tag{1.52}$$

Close to a second order phase transition the autocorrelation time is proportional to the physical correlation length to the power of a factor z

$$\tau \sim \xi^z. \tag{1.53}$$

At a second order phase transition ξ diverges. Because of that, critical slowing down occurs, i.e. the autocorrelation time grows with the volume, which makes it very hard — in other words it takes a huge effort to generate enough statistics — to measure observables close to the critical temperature in large volumes. Thus the factor z determines the quality of the algorithm and it is, of course, the goal to construct algorithms with very small z factors.

1.3.1 Metropolis Algorithm

The Metropolis algorithm was already developed in 1953 [1] and is still widely in use nowadays in cases where cluster-algorithms have not yet been developed, e.g. in gauge theories. It is a local algorithm, i.e. an algorithm that updates only a single site at a time. In this algorithm a new configuration C' is randomly chosen based on the old configuration C. If the energy of the new configuration is smaller than the energy of the old configuration, the new configuration is accepted, and on the other hand, if the new energy is larger, the new configuration is accepted only with a certain probability, i.e.

$$w[C \to C'] = \min(1, \exp(-\beta(\mathcal{H}[C'] - \mathcal{H}[C]))). \tag{1.54}$$

This indeed obeys detailed balance

$$\begin{aligned} \frac{p_C}{p_{C'}} &= \frac{\exp(-\beta\mathcal{H}[C])}{\exp(-\beta\mathcal{H}[C'])} = \frac{w[C' \to C]}{w[C \to C']} \\ &= \frac{\min(1, \exp(-\beta(\mathcal{H}[C] - \mathcal{H}[C'])))}{\min(1, \exp(-\beta(\mathcal{H}[C'] - \mathcal{H}[C])))} = \frac{\exp(-\beta\mathcal{H}[C])}{\exp(-\beta\mathcal{H}[C'])}. \end{aligned} \tag{1.55}$$

In the Ising model one picks a site x and flips it with the right probability. It is obvious to see that ergodicity is satisfied because it is possible to reach any configuration in L^d steps with non-zero probability.

Because the Metropolis algorithm performs only local updates and thus small steps in configuration space, it is not very efficient and has large autocorrelation times close to criticality. The z factor is around the value of 2.

1.4 Standard Cluster Algorithms for the Ising Model

1.4.1 Swendsen-Wang Cluster Algorithm

The inefficiency of the Metropolis algorithm motivates the development of cluster algorithms. In the Ising model, Swendsen and Wang constructed the first cluster algorithm [4]

which runs in multi-cluster mode. Instead of flipping single spins, whole areas called clusters are being flipped with 50 percent probability. The algorithm creates so-called bonds that glue neighbouring spins together with a probability p. This maintains the constraint that these certain spins keep their relative orientation in the next cluster-update. Bonds are set with probability ($p = 1 - \exp(-2\beta J)$) for parallel neighbouring spins, and no bonds are set for anti-parallel spins.

To show detailed balance, it is sufficient to consider just one pair of neighbouring spins. If the two are anti-parallel they necessarily belong to different clusters. After a cluster flip they will be parallel with 50 percent probability. In the next sweep the bond will be activated with probability p and deactivated with probability $1 - p$. The probability to turn back to the original antiparallel configuration is $\frac{1}{2}(1 - p)$. The detailed balance relation then takes the form

$$\frac{\exp(-\beta J)}{\exp(\beta J)} = \frac{\frac{1}{2}(1-p)}{\frac{1}{2}} = \frac{\frac{1}{2}\exp(-2\beta J)}{\frac{1}{2}} = \exp(-2\beta J). \tag{1.56}$$

From the fact that it is always possible to set no bonds at all, which creates L^d clusters, each containing one site, one can see that it is possible to reach any configuration in one step and thus ergodicity is shown.

As a benefit of the cluster algorithm we can make use of the cluster representation of the susceptibility

$$\begin{aligned} \chi &= \frac{1}{L^d}\langle \mathcal{M}^2 \rangle = \frac{1}{L^d}\langle (\sum_C \mathcal{M}_C)^2 \rangle = \frac{1}{L^d}\langle (\sum_{C_1, C_2} \mathcal{M}_{C_1}\mathcal{M}_{C_2}) \rangle \\ &= \frac{1}{L^d}\langle \sum_C \mathcal{M}_C^2 \rangle = \frac{1}{L^d}\langle \sum_C |C|^2 \rangle \end{aligned} \tag{1.57}$$

in order to obtain a so-called improved estimator. Instead of measuring just $\mathcal{M}[s]^2$ for the given spin configuration, we sum the squares of all cluster sizes $|C|$. Effectively, this increases the statistics by the factor 2^{N_C}, where N_C is the number of clusters in the configuration.

This algorithm is indeed very efficient because it quickly generates a lot of quite uncorrelated configurations. Swendsen and Wang measure a z factor of 0.35 [4].

1.4.2 Wolff Cluster Algorithm

The Wolff cluster algorithm [5] is a variant of the Swendsen-Wang algorithm. It runs in single-cluster mode, i.e. at a random site x a cluster is built and always flipped. After a single-cluster update a new random site is chosen and the whole procedure repeated. Sometimes this is even more efficient than the Swendsen-Wang algorithm, because no bonds are activated without flipping them.

As for the Swendsen-Wang algorithm, for the Wolff cluster algorithm one can also construct an improved estimator for the susceptibility. While in multi-cluster mode all

clusters contribute $|C|^2$ to the susceptibility, in the single-cluster algorithm the cluster is selected with a probability $|C|/L^d$, i.e. proportional to its size $|C|$. Hence, bigger clusters are selected more frequently than smaller ones and one must correct for this bias. Hence, in the single-cluster algorithm the improved estimator for the susceptibility takes the form

$$\chi = \frac{1}{L^d}\langle |C|^2 \frac{L^d}{|C|}\rangle = \langle |C|\rangle. \tag{1.58}$$

Chapter 2

Reformulation of the Ising Model

2.1 Hamiltonian and Trotter Decomposition

We rewrite the Ising model with its Hamilton function as a quantum system with the Hamiltonian

$$H = -J \sum_{x,\mu} \sigma_x^3 \sigma_{x+\hat{\mu}}^3. \tag{2.1}$$

It is now formulated as a quantum system, but still describes the same system as the classical Ising model. Due to the trace-structure of the partition function $Z = \mathrm{Tr} \exp(-\beta H)$, we can perform a unitary transformation rotating to a basis containing σ_1,

$$H = -J \sum_{x,\mu} \sigma_x^1 \sigma_{x+\hat{\mu}}^1. \tag{2.2}$$

The basis choice is completely arbitrary. The system resembles the Ising model in any basis. However, the corresponding algorithms will look quite different from the usual ones.

We then perform a Trotter decomposition of the Hamiltonian. In one dimension this implies the following splitting into two parts,

$$H = H_1 + H_2,\ H_1 = -J \sum_{x \in (2m)} \sigma_x^1 \sigma_{x+1}^1,\ H_2 = -J \sum_{x \in (2m+1)} \sigma_x^1 \sigma_{x+1}^1, \tag{2.3}$$

while in two dimensions one needs four parts

$$H_1 = \sum_{x \in (2m,n)} h_{x,\hat{1}},\ H_2 = \sum_{x \in (m,2n)} h_{x,\hat{2}},\ H_3 = \sum_{x \in (2m+1,n)} h_{x,\hat{1}},\ H_4 = \sum_{x \in (m,2n+1)} h_{x,\hat{2}}, \tag{2.4}$$

where m, $n \in \mathbb{N}$.

The partition function is represented as a path integral by introducing an additional Euclidean time dimension with a finite extent $\beta = \epsilon M$, which determines the inverse temperature. For example, in the 1-dimensional Ising model, the Trotter decomposition of the partition function takes the form

$$Z = \mathrm{Tr} \exp(-\epsilon H_1 - \epsilon H_2)^M = \mathrm{Tr} \exp(-\epsilon H_1)^M \exp(-\epsilon H_2)^M. \tag{2.5}$$

The contributions to the sums in H_1 and H_2 commute. We first let H_1 and then H_2 act on the system, which leads to a checkerboard Trotter decomposition where the interactions reside only on the shaded plaquettes. The resulting space-time lattice has the form shown in figure 2.1.

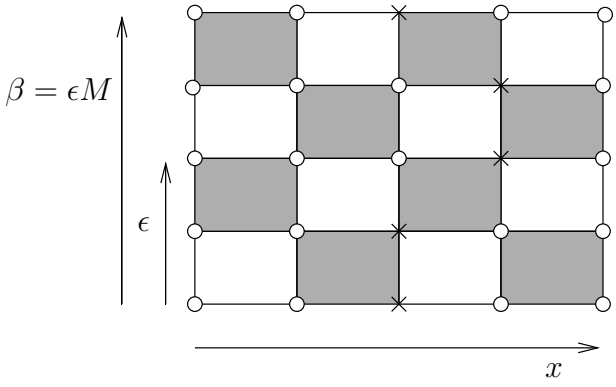

Figure 2.1: *Trotter decomposition in one dimension with a spin-configuration: crosses and circles represent the two spin states.*

In general, the Trotter decomposition is affected by an order ϵ^2 error. Unlike for genuine quantum spin systems, in the Ising model H_1 and H_2 commute and thus it is an exact rewriting of the partition function. Thus even with the minimal number of time-slices one obtains the continuous time limit.

2.2 Observables

We are going to measure the same observables as in the standard formulation of the Ising model. For example the magnetisation $\mathcal{M}$ takes the form

$$\langle \mathcal{M} \rangle = \frac{1}{Z} \mathrm{Tr} \left[\sum_x s_x \exp(-\beta \mathcal{H}[s]) \right] = \frac{1}{Z} \mathrm{Tr} \left[\sum_x \sigma_x^1 \exp(-\beta \mathcal{H}[s]) \right]. \tag{2.6}$$

This would be hard to measure and not very interesting, because $\langle \mathcal{M} \rangle = 0$. Rather we are interested in the correlation function

$$\langle s_x s_y \rangle = \frac{1}{Z} \mathrm{Tr} \left[s_x s_y \exp(-\beta H) \right]. \tag{2.7}$$

In our basis, where the spin-operators s_x are replaced by σ_x^1 matrices, the correlation function takes the form

$$\langle \sigma_x^1 \sigma_y^1 \rangle = \frac{1}{Z} \mathrm{Tr} \left[\sigma_x^1 \sigma_y^1 \exp(-\beta H) \right]. \tag{2.8}$$

We also want to measure the susceptibility

$$\chi = \sum_x \langle s_0 s_x \rangle = \sum_x \langle \sigma_0^1 \sigma_x^1 \rangle, \tag{2.9}$$

as well as the energy density E which is equivalent to $-J$ times the correlation function at distance 1

$$E = -J\langle s_0 s_1 \rangle = -J\langle \sigma_0^1 \sigma_1^1 \rangle. \tag{2.10}$$

Chapter 3

Algorithm for the reformulated Ising Model

3.1 The new Cluster Algorithm

In our basis the transfer matrix of a single plaquette $T = \exp(-\epsilon\sigma_x^1\sigma_{x+1}^1)$ takes the form

$$T = \exp(-\epsilon H) = \begin{array}{c} \begin{array}{cccc} \uparrow\uparrow & \uparrow\downarrow & \downarrow\uparrow & \downarrow\downarrow \end{array} \\ \begin{pmatrix} c & 0 & 0 & s \\ 0 & c & s & 0 \\ 0 & s & c & 0 \\ s & 0 & 0 & c \end{pmatrix} \end{array} \begin{array}{c} \uparrow\uparrow \\ \uparrow\downarrow \\ \downarrow\uparrow \\ \downarrow\downarrow \end{array} , \tag{3.1}$$

with $c = \cosh(\epsilon J)$ and $s = \sinh(\epsilon J)$. It is obvious, that the transfer matrix only allows certain possible plaquette configurations, shown in figure 3.1, which the algorithm has to pay tribute to.

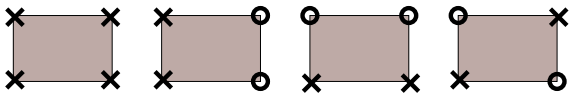

Figure 3.1: *Allowed plaquette configurations.*

In order to imply constraints on the relative orientation of the spins on a plaquette, we propose plaquette breakups that bind spins together on these plaquettes. These constraints must be maintained in the cluster update. The breakups give rise to clusters and these clusters can only be flipped as a whole. In this way, by construction, one indeed maintains the constraints implied by the cluster breakups. We choose to use the A, B_1, and B_2 breakups shown in figure 3.2 where

- A binds two spins on a plaquette that are constant in the Euclidean time direction,
- B_1 binds two spins on a plaquette that are constant in a spatial direction, and
- B_2 binds two spins on a plaquette that are opposite in a spatial direction.

It is also possible to propose further breakups shown in figure 3.3, but the ones from

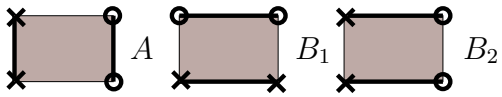

Figure 3.2: *The selected cluster breakups.*

above are sufficient.

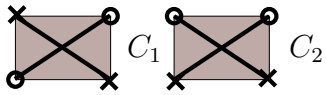

Figure 3.3: *Other possible plaquette breakups.*

The breakups above decompose the transfer matrix as follows

$$T = A \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} + B_1 \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix} + B_2 \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{3.2}$$

From eq.(3.2) one obtains the Boltzmann weights $B_1 = B_2 = s$ and $A = c - s$. Using these cluster breakups we can now simulate the system. In figure 3.4 we show an example of a cluster update, in multi-cluster mode, in which the dashed cluster has been flipped.

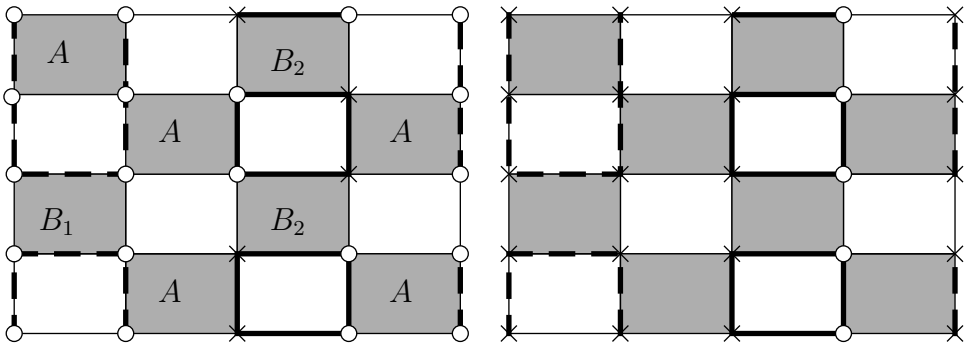

Figure 3.4: *A configuration with A, B_1 and B_2 breakups with a multi-cluster update.*

The chosen breakups can be rotated back to the basis containing σ^3. In that basis, using the breakups, we can from the reference configuration (see below) get to any possible configuration in a finite number of steps. Also this is the case in the opposite direction where we can get from any possible configuration to the reference configuration in a finite

number of steps. In this way ergodicity can be shown. (With the reference configuration we denote a perfectly ordered state. In our case i.e. all spins pointing in one direction.)

The energy density E is related to the fraction P of interaction plaquettes with a transition (spins opposite to their Euclidean time partners on a plaquette). In a given configuration $[s]$ we have $N[s]$ transition- and $(MV - N[s])$ non-transition-plaquettes $(P[s] = N[s]/(MV))$. The weight of a configuration is $\exp(-\beta H[s]) = c^{MV-N[s]}s^{N[s]}$, with $t = \tanh(\epsilon J)$, and MV the total number of plaquettes. Thus the energy density takes the form

$$\begin{aligned} E &= -\frac{1}{VZ}\frac{\partial Z}{\partial \beta} = -\frac{1}{VZ}\frac{\partial}{\partial \beta}\sum_{[s]}\exp(-\beta H[s]) = -\frac{1}{VZ}\sum_{[s]}\exp(-\beta H[s])\frac{\partial}{\partial \beta}(-\beta H[s]) \\ &= -\frac{1}{VZ}\sum_{[s]}\exp(-\beta H[s])\frac{\partial}{\partial \beta}\log(c^{MV-N[s]}s^{N[s]}) \\ &= -\frac{1}{VZ}\sum_{[s]}\exp(-\beta H[s])\left[(MV - N[s])\frac{s}{c}\frac{J}{M} + N[s]\frac{c}{s}\frac{J}{M}\right] \\ &= -J\left[t - \langle P\rangle t + \langle P\rangle\frac{1}{t}\right] = -J\left[t - \frac{1-t^2}{t}\langle P\rangle\right]. \end{aligned} \tag{3.3}$$

3.1.1 Measurement of the 2-Point Correlation Function

In our basis the correlation function is defined as $\langle \sigma_0^1 \sigma_x^1 \rangle$. The σ^1 matrices are off-diagonal and flip the spins they act on. Thus they manifest themselves as violations of spin conservation, i.e. opposite spins reside on the same site (see figure 3.1.1).

$$\sigma^1 = \begin{array}{c} \begin{array}{cc} \uparrow & \downarrow \end{array} \\ \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \end{array} \begin{array}{c} \uparrow \\ \downarrow \end{array}$$

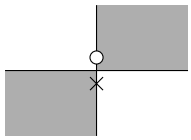

Figure 3.5: *The action of the σ^1 operator on a site, seen as a violation of spin conservation.*

In our algorithm we introduce two violations on a random site (where they cancel each other), build the clusters, and flip them with 50 percent probability, unless it is a cluster with one or two violations. These clusters are flipped partially in order to move the violations to different positions. For example, in figure 3.6 the dashed part of a cluster with violations has been flipped.

When the violations are on the same site, we introduce them randomly somewhere else. By histogramming the spatial distance between the two violations, we get a very accurate measurement of the correlation function. One then normalises the correlation function to 1 at zero distance.

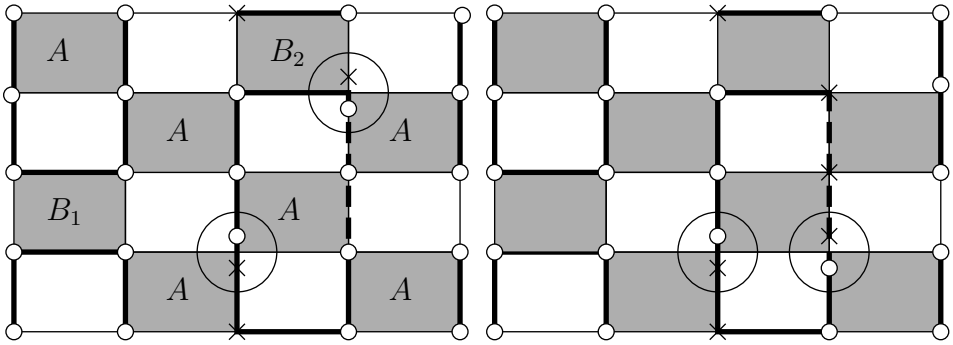

Figure 3.6: *Partially flipping a cluster and thus moving a violation (the violations of spin conservation are encircled).*

3.1.2 Improved Estimator for the Susceptibility

The susceptibility can be measured by summing over the correlation function. This is already an improved estimator because one gets only positive contributions. Because the correlation function is a normalised histogram, one can only measure the normalisation coefficient, i.e. the average frequency of reaching spatial distance zero. This corresponds to the inverse susceptibility. In a large system at high temperature this is not too efficient yet and can be further improved.

We project the clusters with violations to the spatial volume and measure their overlap. This gives us the probability that two violations are at zero spatial distance in the next cluster update and, again, corresponds to measuring the inverse susceptibility. In this way we gain statistics by a factor of the cluster-size squared $|C|^2$ with an effort proportional to the cluster-size $|C|$.

(One could also construct an improved estimator for the energy density in a similar way, where one measures the probability to be at distance one in the next cluster update. This directly corresponds to the correlation at distance one and thus to $-J$ times the energy density).

3.2 Worm Algorithm

Instead of using clusters, we can — with the same breakups — move the two violations locally with a worm-type Metropolis algorithm [15]. We start the worm by introducing two violations on the same site x. We assign them to the two different plaquettes attached to this specific site. These violations are called heads of the worm. With the same probabilities as for the A and B breakups, the heads are moved in the Euclidean time or the spatial direction on the plaquettes they currently reside on. We then assign each violation to the other plaquette attached to its site and start all over. In that way the two heads are moved until they happen to be on the same site and thus close the worm. We then reintroduce

the worm on a different site and start all over. A closed worm is very similar to a cluster apart from the fact that the worm can intersect itself. An example of a worm is shown in figure 3.7.

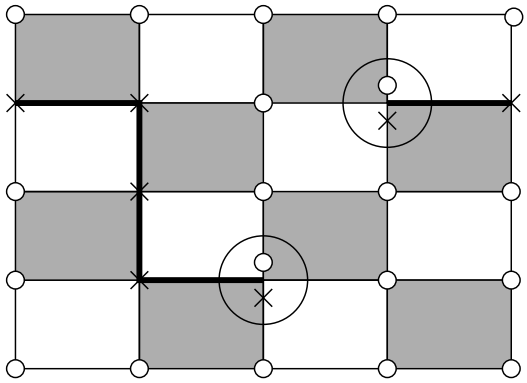

Figure 3.7: *Worm moving the violations and thus updating a given configuration.*

Interestingly we can also measure n-point functions by introducing n violations. These violations are moved in the same way as above.

3.2.1 Snake Algorithm

An efficient trick [16] allows us to measure an exponentially suppressed signal with a linear effort. The correlation function is exponentially suppressed in the high-temperature phase and is thus hard to measure at large distances. The correlation function is the ratio of two partition functions and can be rewritten as the product of many fractions which can be measured individually

$$\langle \sigma_0^1 \sigma_x^1 \rangle = \frac{Z(x)}{Z(0)} = \frac{Z(1)}{Z(0)} \frac{Z(2)}{Z(1)} \cdots \frac{Z(x)}{Z(x-1)}. \tag{3.4}$$

In practice we let an end of the worm move only in a certain region of the lattice around a desired distance y. In this way we get a lot of data for a precise measurement of the fraction $Z(y+1)/Z(y)$. One repeats this procedure for all y spanning from 1 to $(x-1)$ in order to completely reproduce $Z(x)/Z(0)$.

Chapter 4

The new Algorithm in the Standard Basis

The standard basis for the Ising model — the one the Swendsen-Wang algorithm is constructed in — is the basis with σ^3 in the Hamiltonian

$$H = -J \sum_{x,\mu} \sigma^3_x \sigma^3_{x+\hat{\mu}}. \tag{4.1}$$

The transfer matrix now takes the form

$$T = \begin{pmatrix} c+s & 0 & 0 & 0 \\ 0 & c-s & 0 & 0 \\ 0 & 0 & c-s & 0 \\ 0 & 0 & 0 & c+s \end{pmatrix}. \tag{4.2}$$

Of course, this could be decomposed as

$$T = A \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} + D \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{4.3}$$

which would be nothing but Swendsen-Wang dynamics with an additional Euclidean time dimension. The breakups A and D, with the weights $A = \exp(-2\epsilon J)$ and $D = (1 - \exp(-2\epsilon J))$, are shown in figure 4.1.

4.1 Construction of the Algorithm in the Standard Basis

We want to construct in this basis our algorithm rotated from the basis containing σ^1. We rotate the breakups with a unitary transformation

$$T' = U^\dagger T U = U^\dagger A U + U^\dagger B_1 U + U^\dagger B_2 U = A' + B'_1 + B'_2, \tag{4.4}$$

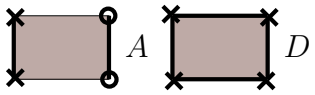

Figure 4.1: *Cluster breakups for the Swendsen-Wang like algorithm with an additional Euclidean time dimension.*

with

$$U = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}. \tag{4.5}$$

T is the transfer matrix from eq.(3.2) and A', B_1', and B_2' are the rotated breakups. For convenience we will call them again A, B_1, and B_2. The rotated breakups A, B_1, and B_2 take the form shown in figure 4.2 with the the same Boltzmann weights as $A = c - s$ and $B_1 = B_2 = s$ as in the other basis. The transfer matrix T decomposes as

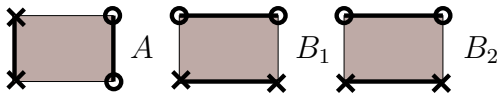

Figure 4.2: *Cluster breakups in the basis containing σ^3.*

$$T = A \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} + B_1 \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix} + B_2 \begin{pmatrix} 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix}. \tag{4.6}$$

From that we can construct the algorithm in a very similar way as in chapter 3.

4.2 The Sign Problem

Because B_2 contains a minus sign, the partition function takes the form

$$Z = \sum_{[s]} \mathrm{Sign}[s] \exp(-\beta \mathcal{H}[s]). \tag{4.7}$$

In order to measure an observable, one has to sample the average sign

$$\langle \mathcal{O} \rangle = \frac{1}{Z} \sum_{[s]} \mathcal{O}[s] \mathrm{Sign}[s] \exp(-\beta \mathcal{H}[s]) = \frac{\langle \mathcal{O} \mathrm{Sign} \rangle}{\langle \mathrm{Sign} \rangle}. \tag{4.8}$$

The average sign is exponentially small both with in the volume V and the inverse temperature β and thus it is nearly impossible to directly get an accurate measurement of an observable.

4.3 Meron-Cluster Solution

The sign problem can be solved using the meron-cluster concept [10, 11]. We classify clusters that contain an odd number of B_2 breakups, and thus change the sign of the configuration when being flipped, as meron-clusters.

If our system is in a configuration containing meron-clusters, the average contribution to the partition function is zero. This can be seen in a multi-cluster update, where the sign will be changed with 50 percent probability. Thus only configurations with no merons — i.e. the 0-meron sector of the system — contribute to the partition function. From this we get an improved estimator for the sign, given by the average time the system is in the 0-meron sector

$$\langle \text{Sign} \rangle = \langle \delta_{N,0} \rangle. \tag{4.9}$$

In a meron-cluster algorithm one does not build completely new clusters after every update. Starting with an initial configuration in the 0-meron sector, one visits every plaquette and proposes a new breakup according to the cluster rules. These proposals are rejected if the new configuration contains meron-clusters. In that way we are only in the 0-meron sector where the sign is always one.

To decide whether the meron number (i.e. the number of meron-clusters in the system) changes, one has to examine the clusters affected by a new connection. Although this requires a computational effort proportional to the cluster size, this is no problem, because one gains a factor that is exponentially large in the volume.

This algorithm now has the same average cluster distribution as the algorithm in the basis containing σ^1 without violations.

Observables such as the magnetisation, can now be efficiently measured.

4.3.1 2-Point Functions

Observables such as the correlation function, the energy, and the susceptibility, that contain products of different spins also get contributions from the 2-meron sector. Thus we have to sample both the 0- and the 2-meron sector of the system. This is done in the same way as above with the only difference that two sectors are now allowed. Observables are now measured by

$$\langle \mathcal{O} \rangle = \frac{\langle \mathcal{O}\delta_{N,0} \rangle + \langle \mathcal{O}\delta_{N,2} \rangle}{\langle \delta_{N,0} \rangle}. \tag{4.10}$$

However, because the 0-meron sector still is strongly suppressed, we use a re-weighting technique where we introduce a probability p_2 for the 2-meron sector. A bond that moves the configuration from the 0- to the 2-meron sector is accepted with probability p_2 and otherwise rejected. Thus eq.(4.10) takes the form

$$\langle \mathcal{O} \rangle = \frac{\langle \mathcal{O}\delta_{N,0} \rangle + \langle \mathcal{O}\delta_{N,2} \rangle / p_2}{\langle \delta_{N,0} \rangle}. \tag{4.11}$$

From the 2-meron sector the only, but always positive contributions to these observables are those where one spin is in meron-cluster C_1 and the other spin in the other meron-cluster C_2. That is because the two spins are either parallel when Sign $= 1$ or antiparallel when Sign $= -1$. Thus the estimator for an observable $\langle \mathcal{O} \rangle$ can be further improved.

The energy density is thus given by

$$E = -\frac{J}{2V} \frac{\langle \sum_{\langle xy \rangle} s_x s_y \delta_{N,0} \rangle + \langle \sum_{\langle xy \rangle} \delta_{N,2} \rangle_{x \in C_1, y \in C_2}}{\langle \delta_{N,0} \rangle}. \tag{4.12}$$

With this concept in hand, one can construct an improved estimator for the susceptibility given by

$$\chi = \frac{\langle \sum_C \mid C \mid^2 \delta_{N,0} \rangle + \langle 2 \mid C_1 \mid\mid C_2 \mid \delta_{N,2} \rangle}{V \beta \langle \delta_{N,0} \rangle}, \tag{4.13}$$

where C_1 and C_2 are again the tow meron-clusters. Using the re-weighting technique from above, the improved estimator for the susceptibility takes the form

$$\chi = \frac{\langle \sum_C \mid C \mid^2 \delta_{N,0} \rangle + \langle 2 \mid C_1 \mid\mid C_2 \mid \delta_{N,2} \rangle / p_2}{V \beta \langle \delta_{N,0} \rangle}. \tag{4.14}$$

The N-Point functions in general get contributions from all meron sectors with 0 up to n merons. Here n is always an even number. Thus all these meron sectors have to be sampled by the algorithm.

Chapter 5

Results

In this chapter we present the results obtained with the new algorithms presented in chapter 2 and 3. The rotated algorithm shown in chapter 4 also works correctly, but no extensive measurements have been performed.

5.1 Verification

In order to verify that our algorithm is indeed correct, we measured the correlation function and compared it to the analytic expression [14]. (In two dimensions we measured the correlation function along a lattice diagonal). Our measurements are always consistent with the analytic results within statistical errors. This can be seen below. We also observe the phase transition in the 2- and 3-dimensional model at the correct temperatures, which is a good indication that our method is indeed correct.

We measured the correlation function both with the cluster and the worm algorithm. A measurement around β_c is shown in figure 5.1. The measurements are right on top of the analytical recursion formula, apart from the measurement quite at the critical temperature. This discrepancy is due to finite size effects and it goes away with increasing volume shown in figure 5.2.

5.2 Correlation function over 100 Orders of Magnitude

Using the snake algorithm we could even push forward to measurements of the correlation function that span over more than 100 orders of magnitude. In figure 5.3 we compare the numerical results at very high temperature with the analytical expression [14]. In the figure the analytical result and our measurement are right on top of each other. Of course, one cannot see an error because its on a logarithmic scale, but the errors in this measurement were less than four percent up to the values of 100 orders of magnitude, which is highly accurate. At diagonal distance 40 (which is the maximum distance on a 80^2 lattice with

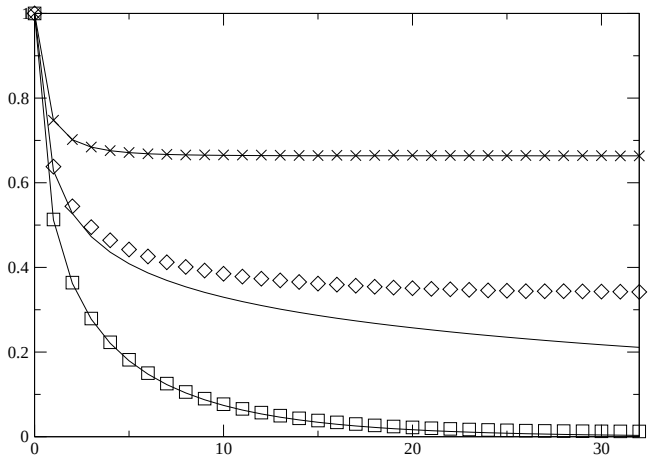

Figure 5.1: *Correlation function on a* 64^2*-lattice at* $\beta = 0.42$ *(squares),* $\beta = 0.44$ *(diamonds), and* $\beta = 0.46$ *(crosses) compared to the analytic results (lines).*

periodic boundary conditions) there are some visible finite size effects. The analytical expression in two dimensions considers the infinite volume limit.

To get a feeling of how gigantic these orders of magniture are, one could imagine that we were comparing the mass of an electron ($\sim 10^{-30}$ kg) with the mass of all approximately 10^{11} galaxies in our universe together ($\sim 10^{50}$ kg). This would be a comparison of the order of 10^{80}. We can span our measurement over an even larger number of orders of magnitude.

5.3 Autocorrelation Times

We performed autocorrelation time measurements for the improved estimator for the susceptibility and got an autocorrelation time smaller than 1 sweep. Also we measured the energy using eq.(3.3) with the cluster algorithm without any violations. In two dimensions it matters that one uses the Trotter decomposition from eq.(2.4) and the minimal number of time-slices. If one does so, the autocorrelation time is also very small which can be seen in figure 5.4. In this measurement on a 100^2 lattice we got an autocorrelation time $\tau \simeq 7$.

5.4 Cluster Diagnostics in two Dimensions

In the following subsection the cluster-size distribution will be discussed. Our measurements below are all done with the minimal number of four time-slices.

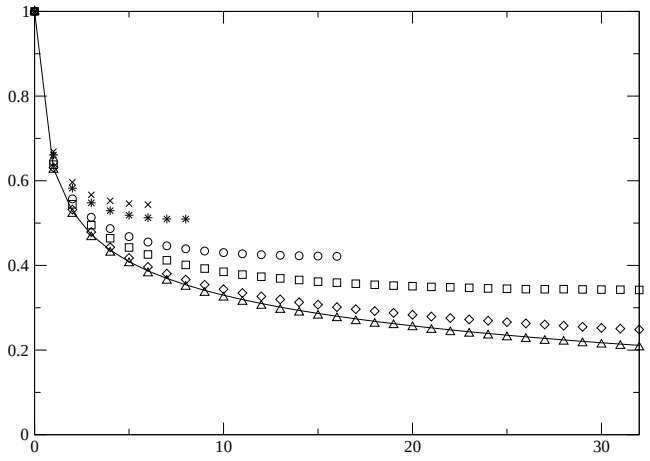

Figure 5.2: *Correlation fits analytical result (straight line) with increasing volume at* $\beta = 0.44$. *Measured on* 12^2, 16^2, 32^2, 64^2, 128^2, 256^2, *and* 1024^2 *lattices.*

5.4.1 Multi-Cluster Algorithm

Using our 2-dimensional Trotter decomposition from eq.(2.4) and the minimal number of four time-slices, we measured the average cluster-size with multi-cluster updates. From that we can see that our algorithm is indeed very different from the standard Swendsen-Wang-algorithm. Furthermore, we get a peak at the critical temperature which grows sharper with increasing volume. This can be seen in figure 5.5 compared to the Swendsen-Wang algorithm.

5.4.2 Average Cluster-Size with different Trotter Decompositions

Instead of using the Trotter decomposition proposed in eq.(2.4) — here denoted by A — one can also use different ones. The most obvious possibilities are the following
• Trotter decomposition B is equivalent to Trotter decomposition A apart from the fact that H_2 and H_3 are interchanged

$$H_1 = \sum_{x\in(2m,n)} h_{x,\hat{1}},\ H_2 = \sum_{x\in(2m+1,n)} h_{x,\hat{1}},\ H_3 = \sum_{x\in(m,2n)} h_{x,\hat{2}},\ H_4 = \sum_{x\in(m,2n+1)} h_{x,\hat{2}} \qquad (5.1)$$

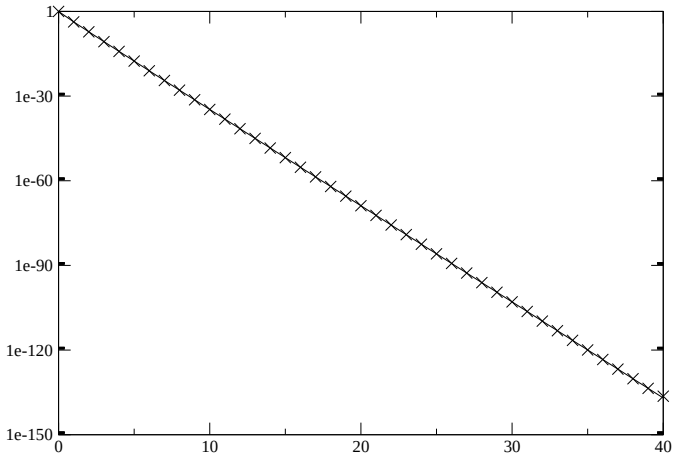

Figure 5.3: *Correlation function along a lattice diagonal on a* 80^2*-lattice at* $\beta = 0.01$*.*

with m, $n \in \mathbb{N}$.

• Trotter decomposition C is a full-fledged checkerboarding

$$H_1 = \sum_{x\in(mn)} h_{x,\hat{1}},\ H_3 = \sum_{x\in(mn+1)} h_{x,\hat{1}},\ H_2 = \sum_{x\in(mn)} h_{x,\hat{2}},\ H_4 = \sum_{x\in(mn+1)} h_{x,\hat{2}}. \tag{5.2}$$

• Trotter decomposition D is Trotter decomposition C with interchanged H_2 and H_3

$$H_1 = \sum_{x\in(mn)} h_{x,\hat{1}},\ H_2 = \sum_{x\in(mn)} h_{x,\hat{2}},\ H_3 = \sum_{x\in(mn+1)} h_{x,\hat{1}},\ H_4 = \sum_{x\in(mn+1)} h_{x,\hat{2}}. \tag{5.3}$$

The difference in the average cluster-size between these different Trotter decompositions is striking. One can again see the same as in figure 5.5 but this time with all these four Trotter decompositions (see figure 5.6).

5.4.3 Single-Cluster Algorithm

We can also measure the average cluster-size in single-cluster mode and get a similar result (see figure 5.7) as above with the Trotter decompositions A, B, C, D compared to the Wolff-algorithm. We can see that for small β we get practically the same average cluster-size for all different Trotter decompositions up to the phase transition at β_c. Above the situation is quite different. Also the average cluster-size distribution always has a peak with our algorithm whereas the size of the Wolff clusters always increases with β until they fill the whole volume.

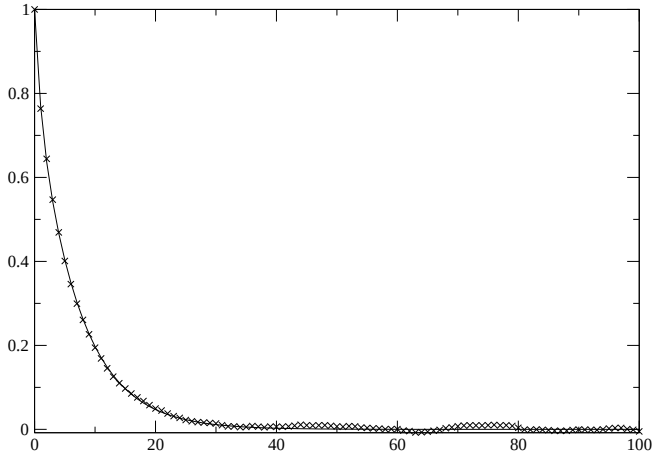

Figure 5.4: *Autocorrelation-time measurement (crosses) for the energy on a* 100^2 *lattice using no violations and the fit (straight line).*

5.5 Comparison of the two Bases

The new algorithm without violations has the same cluster-size distribution as the algorithm in the standard basis sampling only the 0-meron sector. This is a clear indication that the algorithm is indeed the same independent of the basis chosen. A comparison of the cluster-size distribution at different β is shown in figure 5.8.

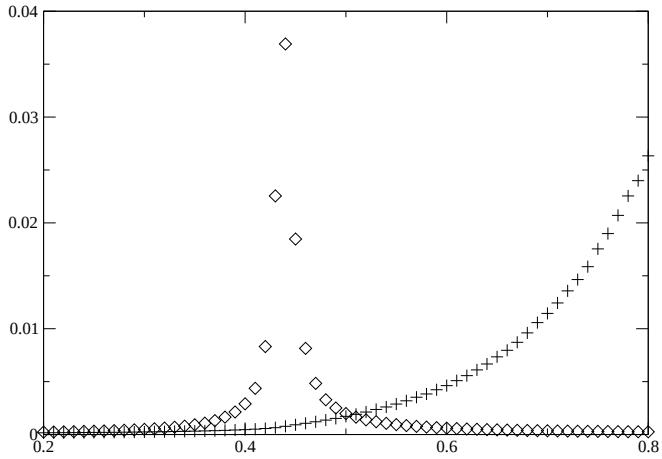

Figure 5.5: *Average cluster-size of the new algorithm (diamonds) compared to the Swendsen-Wang algorithm (crosses) on a* 100^2 *lattice.*

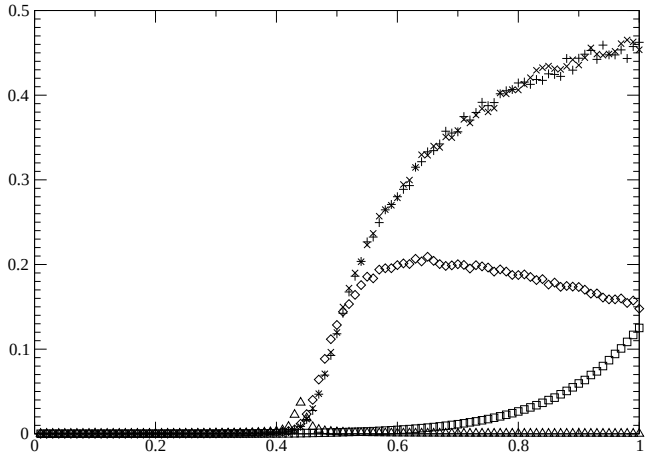

Figure 5.6: *Algorithm (using Trotter decompositions A,B,C,D — triangles, diamonds, crosses, diagonal crosses) and Swendsen-Wang (squares) average cluster-size per volume on a* 100^2 *lattice.*

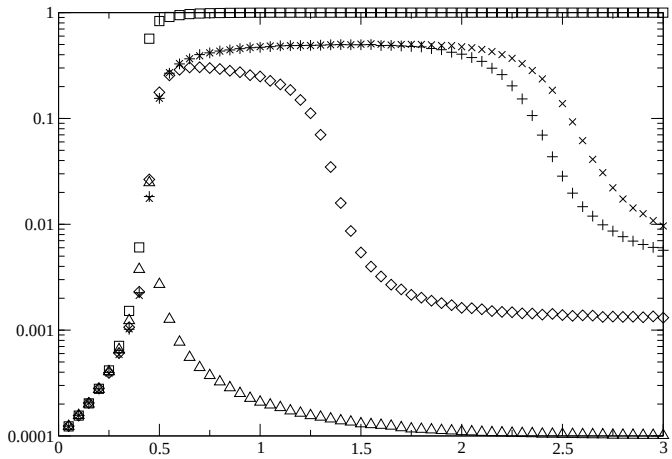

Figure 5.7: *Algorithm in single-cluster mode (using Trotter decompositions A,B,C,D — triangles, diamonds, crosses, and diagonal crosses) and Wolff (squares) average cluster-size on a* 100^2 *lattice*

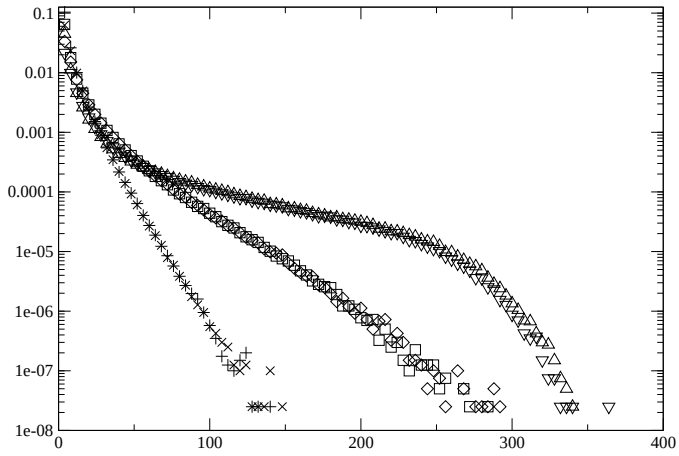

Figure 5.8: *Cluster distribution of the new algorithm at* $\beta = 0.2$ *(crosses),* $\beta = 0.3$ *(squares), and* $\beta = 0.5$ *(triangles up) compared to the algorithm in the standard basis at* $\beta = 0.2$ *(diagonal crosses),* $\beta = 0.3$ *(diamonds), and* $\beta = 0.5$ *(triangles down).*

Conclusions

We have seen that using the methods of D-theory an efficient loop cluster algorithm can be constructed that is very different from the standard Swendsen-Wang cluster algorithm.

We have an improved estimator for the susceptibility which is a very efficient method to measure this quantity. Furthermore, we are able to perform measurements of the two-point correlation function, using the snake algorithm, in regions previously inaccessible. In particular measuring a signal that is suppressed by a factor of hundred orders of magnitude.

We have also shown that this approach can be implemented in the standard basis as a meron-cluster algorithm.

This kind of approach provides deeper understanding of cluster algorithms and hopefully opens the door to tackling other theories, possibly gauge theories.

Acknowledgment

I really want to thank Prof. U.-J. Wiese whose support, advice and warmhearted help has inspired and motivated this work.

It has also been a great privilege to work with Michele Pepe who has always been very open to my questions and ideas.

I also want to thank my dear colleagues on the B-floor who also did help me a lot, or at least tried to do so. Namely Christoph Brügger, Christoph Weiermann, Jonas Belina, Stéphane Riederer, and Stefan Lanz.

Furthermore the whole institute deserves a big thankyou for the nice and warm atmosphere it provides.

Also an important thank-you goes to Ottilia Hänni for her support in the bureaucratic battle.

And indeed my parents got to be specially mentioned because their support and patience, without having a lot of insight on what I have been doing, has been the backbone of this now finished thesis.

Bibliography

[1] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *J. Chem. Phys.*, 21:1087–1092, 1953.

[2] F. Niedermayer. Cluster algorithms. 1996.

[3] H. G. Evertz. The loop algorithm. *Adv. Phys.*, 52:1, 2003.

[4] R. H. Swendsen and J.-S. Wang. Nonuniversal critical dynamics in monte carlo simulations. *Phys. Rev. Lett.*, 58:86–88, 1987.

[5] U. Wolff. Collective monte carlo updating for spin systems. *Phys. Rev. Lett.*, 62:361, 1989.

[6] H. G. Evertz, G. Lana, and M. Marcu. Cluster algorithm for vertex models. *Phys. Rev. Lett.*, 70:875–879, 1993.

[7] U.-J. Wiese and H.-P. Ying. Block spin cluster algorithms for quantum spin systems. 1992.

[8] R. Brower, S. Chandrasekharan, S. Riederer, and U.-J. Wiese. D-theory: Field quantization by dimensional reduction of discrete variables. *Nucl. Phys.*, B693:149–175, 2004.

[9] A. S. Tsapalis. Two topics in non-perturbative lattice field theories: The u(1) quantum link model and perfect actions for scalar theories. pages 135–171, 1998.

[10] W. Bietenholz, A. Pochinsky, and U.-J. Wiese. Meron cluster simulation of the theta vacuum in the 2-d o(3) model. *Phys. Rev. Lett.*, 75:4524–4527, 1995.

[11] S. Chandrasekharan and U.-J. Wiese. Meron-cluster solution of a fermion sign problem. *Phys. Rev. Lett.*, 83:3116–3119, 1999.

[12] L. Onsager. Crystal statistics. 1. a two-dimensional model with an order disorder transition. *Phys. Rev.*, 65:117–149, 1944.

[13] K. Huang. *Statistical Mechanics, Bd. 3.* Hochschultaschenbücherverlag, 1964.

[14] M. Jimbo and T. Miwa. Studies on holonimic quantum fields. xvii. *Proc. Japan Acad. A*, 56:405–410, 1980.

[15] N. Prokof'ev and B. Svistunov. Worm algorithms for classical statistical models. *Phys. Rev. Lett.*, 87:160601, 2001.

[16] P. de Forcrand, M. D'Elia, and M. Pepe. A study of the 't hooft loop in su(2) yang-mills theory. *Phys. Rev. Lett.*, 86:1438, 2001.