

# Cluster Algorithms for $CP(N - 1)$ Models in the Large $N$ Limit

Diplomarbeit

der Philosophisch-naturwissenschaftlichen Fakultät  
der Universität Bern

vorgelegt von

**Samuel Hertig**

2007

Leiter der Arbeit

**Prof. Uwe-Jens Wiese**

Institut für Theoretische Physik, Universität Bern

## Abstract

This thesis is about numerical simulations for the  $CP(N-1)$  model, which can be considered a toy model for QCD. The  $CP(N-1)$  target theory emerges from an  $SU(N)$  ferromagnet via D-theory. Physical observables are measured by generating a Markov chain of configurations using the Monte Carlo method. New configurations are generated by forming clusters of spins and changing their orientation. The main focus of the present study is to examine the behavior of the algorithm in the  $N \rightarrow \infty$  limit.



In the beginning the Universe was created.  
This has made a lot of people very angry  
and been widely regarded as a bad move.

Douglas Adams



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Standard Formulation of <math>CP(N - 1)</math> Models</b>	<b>3</b>
2.1	$CP(N - 1)$ Models in Continuous Space-Time . . . . .	3
2.2	$CP(N - 1)$ Models on the Lattice . . . . .	4
<b>3</b>	<b>D-Theory Formulation of <math>CP(N - 1)</math> Models</b>	<b>5</b>
3.1	$SU(N)$ Quantum Spins . . . . .	5
3.2	Magnons . . . . .	6
3.2.1	Calculation of $E_0$ . . . . .	6
3.2.2	Calculation of $E_p$ . . . . .	7
3.3	Spontaneous Symmetry Breaking from $SU(N)$ to $U(N - 1)$ . . . . .	9
3.4	Low-Energy Expansion . . . . .	10
3.5	Dimensional Reduction . . . . .	12
<b>4</b>	<b>Path Integral Representation of <math>SU(N)</math> Quantum Spin Systems</b>	<b>15</b>
4.1	Step 1: Trotter Decomposition of the Hamiltonian . . . . .	15
4.2	Step 2: Insertion of Complete Sets of Spin States . . . . .	16
4.3	Step 3: Calculating the Boltzmann-Weights of a Single Plaquette . . . . .	17
<b>5</b>	<b>Cluster Algorithm for <math>CP(N - 1)</math> Models with small <math>N</math></b>	<b>21</b>
5.1	The Monte Carlo Method . . . . .	21
5.2	Cluster Rules . . . . .	22
5.3	Observables and Improved Estimators . . . . .	25
5.3.1	Uniform Magnetization . . . . .	25
5.3.2	Uniform Magnetic Susceptibility . . . . .	26
5.3.3	Correlation Function . . . . .	27
5.3.4	Correlation Length and the Second Moment Method . . . . .	27
5.4	Summary . . . . .	28
<b>6</b>	<b>The Algorithm in the Large <math>N</math> Limit</b>	<b>29</b>
6.1	Getting Rid of Spins . . . . .	29
6.2	Artificial Weights . . . . .	31
6.3	The t'Hooft limit . . . . .	32
6.3.1	Increasing $n$ . . . . .	32
6.3.2	Increasing $J$ . . . . .	32
6.3.3	Increasing $\beta$ . . . . .	32

---

<b>7</b>	<b>Results</b>	<b>33</b>
7.1	Simulations with moderate $N$ . . . . .	33
7.2	Simulations with $N = \infty$ . . . . .	34
<b>8</b>	<b>Conclusion and Outlook</b>	<b>39</b>
<b>A</b>	<b>Proof of the <math>SU(N)</math>-invariance of <math>H</math></b>	<b>41</b>
A.1	Proof of $[H, T^j] = 0$ . . . . .	41
A.2	Proof of $H' = H$ . . . . .	42
	<b>Acknowledgments</b>	<b>43</b>
	<b>Bibliography</b>	<b>44</b>

# Chapter 1

## Introduction

Shortly after its appearance, quantum chromodynamics (QCD) was found to be a highly non-trivial theory to solve analytically. QCD is one of the main pillars of today's standard model of particle physics. It describes particle interactions due to the strong force, which is one of the four fundamental forces of nature. Even today, an analytical breakthrough for QCD seems far from possible. Fortunately, there exist several alternative methods to tackle it. One standard way is the perturbative approach, which yields results that have been verified experimentally with high precision. However, perturbation theory is only applicable at high energies. Effective theories like chiral perturbation theory or heavy quark theory are also a general approach to QCD, but again limited to a specific energy range. Another ansatz is the  $1/N$ -expansion, which makes use of the fact that QCD would be less problematic if there were an infinite number of quark colors. But even for  $N = \infty$ , QCD cannot be solved analytically. Of course, corrections have to be made since in nature, we know that  $N = 3$ .

Among the non-perturbative methods, lattice QCD is probably the only first principles approach. The basic idea of lattice QCD is to replace continuous space-time by a discrete set of space-time points. The theory is then examined by carrying out numerical simulations, which are often very time consuming and resource intensive. To even further simplify the theory, various toy models for QCD can be formulated. Although they do not describe our physical world, they can help us to understand QCD. They are formulated to be much simpler than QCD, while nevertheless sharing some of the important characteristics of QCD. Two examples of toy models are the non-linear sigma model and the  $CP(N - 1)$  model (the latter discussed for the first time in [1]). However, we know that nowadays even toys can get quite complex, which is especially true in physics. Unfortunately, the  $CP(N - 1)$  model has so far proven to be resistant against all attempts to solve it analytically. Moreover, a no-go theorem described in [2] states that it is impossible to construct a cluster algorithm for  $CP(N - 1)$  models in the standard Wilson formulation.

The only efficient  $CP(N - 1)$  algorithm so far besides the one presented in this thesis is the multigrid algorithm in [3]. However, it is restricted to zero vacuum angle  $\theta = 0$ . An efficient numerical approach for non-zero vacuum angles of  $CP(N - 1)$  models has not been found until recently: Beard, Pepe, Riederer and Wiese were able to construct an efficient cluster algorithm in [4, 5]. Their work was only possible because they used a different method of lattice regularization than the standard Wilson method: D-theory. In D-theory, continuous



classical fields are replaced by quantum spins and an additional dimension is introduced temporarily. At the end of the procedure, the additional dimension disappears via dimensional reduction and the theory reduces to the desired  $CP(N - 1)$  model. This master thesis is largely based on [4, 5] (which is explained in more detail in [6]), but then takes a look at the behavior of the algorithm at large  $N$ . The goal is to decide whether the algorithm would still work if  $N = \infty$ . Possible results could then be compared with theoretical results of the  $1/N$ -expansion for the  $CP(N - 1)$  model.

In chapter 2, we take a look at the standard formulation of the  $CP(N - 1)$  model, both in continuous space-time and on the lattice. Chapter 3 introduces the concept of D-theory applied to a  $SU(N)$  ferromagnet, which results in an alternative formulation of the  $CP(N - 1)$  model. Chapter 4 shows how to reformulate the partition function of our system. These results are then used in chapter 5, where we first take a look at Monte Carlo simulations in general, and then move on to the construction of the algorithm. Chapter 6 deals with the limit as  $N$  approaches infinity, which is the main part of the present study. Finally, the results obtained by the simulations are presented in chapter 7. In chapter 8, we conclude the thesis with some final comments.

## Chapter 2

# Standard Formulation of $CP(N - 1)$ Models

### 2.1 $CP(N - 1)$ Models in Continuous Space-Time

The manifold  $CP(N - 1)$  is the  $(2N - 2)$ -dimensional coset space  $SU(N)/U(N - 1)$ . In chapter 3, we will see how this manifold arises by symmetry breaking from  $SU(N)$  to  $U(N - 1)$ . The fields living in this coset space are represented by  $P(x) \in CP(N - 1)$  which have the following properties:

$$[P(x)]_{ij} \in \mathbb{C}, \text{ where } i, j \in \{1, \dots, N\}, \quad (2.1)$$

$$P(x)^2 = P(x), \quad (2.2)$$

$$P(x)^\dagger = P(x), \quad (2.3)$$

$$\text{Tr}P(x) = 1. \quad (2.4)$$

Equations (2.1) and (2.2) explain the meaning of the name  $CP(N - 1)$ , which stands for Complex Projective space. From equation (2.2) it follows that

$$\det P = \det(P^2) = (\det P)^2,$$

thus  $\det P = 1$  or  $0$ . If we assume  $\det P = 1$ , it follows that  $P^{-1}$  exists. Multiplying equation (2.2) by  $P^{-1}$  we see that  $P$  is just the identity matrix, which is not acceptable because then  $\text{Tr}P(x) = N$ . Therefore we also have

$$\det P = 0. \quad (2.5)$$

When trying to construct a Euclidean action, we have to make sure that it has the following properties:

1. Locality.

2. Lorentz invariance.
3.  $SU(N)$  invariance.
4. The smallest number of derivatives as possible.

The Euclidian action of the  $CP(N - 1)$  model in two dimensions is defined as

$$S[P] = \int d^2x \frac{1}{g^2} \text{Tr}[\partial_\mu P(x) \partial_\mu P(x)], \quad (2.6)$$

which is the simplest non-trivial option. The coupling constant  $g$  is dimensionless. Terms of higher order in  $\partial_\mu P(x)$  are omitted, since they would require a dimensionful constant preceding them. This would result in a non-renormalizable theory. Because of the properties of the trace, the action is invariant under global transformations acting on  $P$  as  $U \in SU(N)$ :

$$P(x)' = UP(x)U^\dagger. \quad (2.7)$$

## 2.2 $CP(N - 1)$ Models on the Lattice

Now we replace continuous space-time by a  $d$ -dimensional lattice. The distance between two lattice sites is denoted by  $a$  and the variable  $x$  is now a site on the lattice and thus discrete. The fields  $P(x)$  are written as  $P_x$  and the action is given by

$$\begin{aligned} S[P] &= \sum_{x,\mu} a^2 \frac{1}{g^2} \text{Tr} \left[ \left( \frac{P_{x+\hat{\mu}} - P_x}{a} \right)^2 \right] \\ &= \sum_{x,\mu} \frac{1}{g^2} \text{Tr} [P_{x+\hat{\mu}}^2 + P_x^2 - 2P_{x+\hat{\mu}}P_x] \\ &= \sum_{x,\mu} \frac{2}{g^2} [-\text{Tr}(P_{x+\hat{\mu}}P_x) + 1], \end{aligned}$$

where  $\hat{\mu}$  is the unit-vector in the  $\mu$ -direction. The additive constant can be dropped and thus the standard lattice action in two dimensions is given by

$$S[P] = -\frac{2}{g^2} \sum_{x,\mu} \text{Tr}(P_{x+\mu}P_x). \quad (2.8)$$

The continuum limit is approached by decreasing the coupling constant  $g \rightarrow 0$ . Due to asymptotic freedom of the  $CP(N - 1)$  models, the correlation length for small  $g$  is exponentially large, i.e. we have

$$\xi \propto \exp(4\pi/Ng^2) \gg a. \quad (2.9)$$

## Chapter 3

# D-Theory Formulation of $CP(N - 1)$ Models

So far, it has been impossible to construct an efficient cluster algorithm for  $CP(N - 1)$  models in the standard Wilson formulation of lattice field theory. *D-theory* is an alternative non-perturbative way of quantizing field theories and makes the construction of cluster algorithms possible. It is a formulation of field theory using *discrete* variables undergoing *dimensional* reduction, hence its name. In the present study, we will concentrate on the application of D-theory as a tool for  $CP(N - 1)$  models only. For further applications, the reader should consult [6, 7, 8].

### 3.1 $SU(N)$ Quantum Spins

In our case, the discrete variables mentioned above are  $SU(N)$  quantum spins located on the sites of a  $d$ -dimensional periodic hypercubic lattice of length  $La$ , thus forming a quantum ferromagnet with a total of  $L^d$  sites. Every spin is represented by a Hermitean operator  $T_x^j$ , where  $x$  is a site on the lattice and  $j = 1, \dots, N^2 - 1$ . The  $T_x^j$  are the generators of the group  $SU(N)$ :

$$\begin{aligned} [T_x^j, T_y^k] &= i\delta_{xy}f_{jkl}T_x^l, \\ \text{Tr}(T_x^j T_y^k) &= \frac{1}{2}\delta_{xy}\delta_{jk}. \end{aligned} \quad (3.1)$$

In the case of  $N = 2$ , the  $T_x^j$  are the three Pauli matrices (multiplied by  $\frac{1}{2}$ ), whereas the generators of  $SU(3)$  are the eight Gell-Mann matrices. The Hamilton operator is

$$H = -J \sum_{\langle xy \rangle} T_x^j T_y^j = -J \sum_{x,i} T_x^j T_{x+i}^j, \quad (3.2)$$

where we have used the implicit summation convention. The unit vector in the  $i$ -direction is  $\hat{i}$  with  $i = 1, \dots, d$ . For a ferromagnet, the exchange coupling  $J$  is positive. A proof that

this Hamiltonian is invariant under a global  $SU(N)$  transformation  $U$  is given in appendix A. The quantum partition function  $Z$  describing such a system takes the form

$$Z = \text{Tr} \exp(-\beta H), \quad (3.3)$$

where  $\beta = 1/kT$  with  $T$  being the temperature of the system. However, the direction of the finite extent  $\beta$  can also be regarded as an extra dimension, in which the evolution of the configuration of the ferromagnet takes place, as described by the Hamiltonian (3.2). This extra dimension will ultimately disappear after dimensional reduction. In the D-theory interpretation, it is not the Euclidean time of the target  $CP(N-1)$  theory.

## 3.2 Magnons

Magnons or spin waves are disturbances of the uniform magnetization of the ground state  $|0\rangle = |uu\dots u\rangle$ . They have the form

$$|p\rangle = \sum_x T_x^- \exp(ipx) |0\rangle. \quad (3.4)$$

They can be considered as the first excited state of the quantum ferromagnet with energy-eigenvalue  $E_p$ :

$$H |p\rangle = E_p |p\rangle. \quad (3.5)$$

In order to verify the action of the low-energy effective theory derived later on, we have to calculate  $E_p$ . We set  $d = 2$  for all calculations in this section. First, we need to compute the energy  $E_{p=0} = E_0$ , since we will use the result for the calculation of  $E_p$  for arbitrary  $p$ .

### 3.2.1 Calculation of $E_0$

The Hamiltonian in equation (3.2) can be rewritten as follows:

$$H = -\frac{J}{4} \sum_{\langle xy \rangle} \left[ (T_x^+ T_{x+\hat{i}}^- + T_x^- T_{x+\hat{i}}^+ + U_x^+ U_{x+\hat{i}}^- + U_x^- U_{x+\hat{i}}^+ + \dots) + \sum_{j=1}^{N-1} \hat{T}_x^j \hat{T}_y^j \right]. \quad (3.6)$$

The matrices  $\hat{T}^j$  are the  $N-1$  diagonal generators of  $SU(N)$ :

$$\hat{T}^j = \sqrt{\frac{2}{j(j-1)}} \text{diag}(\underbrace{1, \dots, 1}_{j \text{ times}}, -j, 0, \dots, 0). \quad (3.7)$$

The sum in equation (3.6) also includes all the  $N(N-1)/2$  raising and lowering operators  $T^-$ ,  $T^+$ ,  $U^-$ ,  $U^+$  etc. Since there is always a raising operator combined with a lowering operator, their contribution to the eigenvalue  $E_0$  is zero. Therefore, we only consider

$$\begin{aligned}
H|0\rangle &= -\frac{J}{4} \sum_{\langle xy \rangle} \sum_{j=1}^{N-1} \hat{T}_x^j \hat{T}_y^j |0\rangle = -\frac{J}{4} (2L^2) \sum_{j=1}^{N-1} \hat{T}_x^j \hat{T}_y^j |0\rangle \\
&= -\frac{JL^2}{2} \sum_{j=1}^{N-1} \left( \frac{\sqrt{2}}{\sqrt{j(j+1)}} \right)^2 |0\rangle = -JL^2 \sum_{j=1}^{N-1} \frac{1}{j(j+1)} |0\rangle \\
&= -JL^2 \left( \frac{N-1}{N} \right) |0\rangle = E_0 |0\rangle.
\end{aligned} \tag{3.8}$$

In the second step, we have used the fact that we have four neighbours per site and thus a total of  $2L^2$  neighbour couplings. Hence, our result is

$$E_0 = -JL^2 \left( \frac{N-1}{N} \right). \tag{3.9}$$

### 3.2.2 Calculation of $E_p$

Let us take a detailed look at the definition (3.4). The operator  $T_x^-$  acts on the spin on the lattice site  $x$  and flips it to another quantum state. Without loss of generality, we choose

$$T_x^- = \frac{1}{\sqrt{2}} (T_x^1 - iT_x^2), \tag{3.10}$$

such that

$$T_x^- |u\rangle = \sqrt{2} |d\rangle. \tag{3.11}$$

We can now rewrite the definition (3.4) of the spin wave by introducing the state  $|x\rangle = |uu\dots udu\dots u\rangle$ , where the site  $x$  has spin  $d$  and all others have spin  $u$  :

$$|p\rangle = \sum_x \exp(ipx) |x\rangle. \tag{3.12}$$

Further, we decompose the Hamiltonian into two parts:  $H = H_x + H_{rest}$ . The contribution  $H_x$  acts only on the site  $x$  with its four neighbours (recall  $d = 2$ ) and  $H_{rest}$  acts on all the other  $2L^2 - 4$  sites. From equation (3.9) it follows that

$$H_{rest} |x\rangle = -J \left( \frac{2L^2 - 4}{2} \right) \left( \frac{N-1}{N} \right) |x\rangle = J(2 - L^2) \left( \frac{N-1}{N} \right) |x\rangle. \tag{3.13}$$

To calculate the eigenvalue produced by  $H_x$  we make a further decomposition, similar to the one in equation (3.6):

$$H_x = \sum_{\substack{y: \text{neigh-} \\ \text{bours of } x}} (H_{x,y}^{diag} + H_{x,y}^{shift}), \tag{3.14}$$

where

$$H_{x,y}^{diag} = -\frac{J}{4} \sum_{j=1}^{N-1} \widehat{T}_x^j \widehat{T}_y^j, \quad (3.15)$$

where the matrices  $\widehat{T}^j$  are again the  $N-1$  diagonal generators of  $SU(N)$ , as defined in equation (3.7). Since  $H^{diag}$  acts on two spins with flavors  $d$  at site  $x$  and  $u$  at the neighbouring site  $y$ , we define  $|u\rangle = (0, \dots, 0, 1, 0)$  and  $|d\rangle = (0, \dots, 0, 1)$  and obtain

$$\begin{aligned} H_{x,y}^{diag} |ud\rangle &= -\frac{J}{4} \sum_{j=1}^{N-1} \widehat{T}_x^j \widehat{T}_y^j |ud\rangle = -\frac{J}{4} \widehat{T}_x^{N-1} \widehat{T}_y^{N-1} |ud\rangle \\ &= -\frac{J}{4} (-N-1) \frac{2}{N(N-1)} |ud\rangle = -\frac{J}{2N} |ud\rangle. \end{aligned} \quad (3.16)$$

Hence,

$$\sum_{\substack{y: \text{neigh-} \\ \text{bours of } x}} H_{x,y}^{diag} |x\rangle = -\frac{2J}{N} |x\rangle. \quad (3.17)$$

The other part of  $H_x$  is

$$\sum_{\substack{y: \text{neigh-} \\ \text{bours of } x}} H_{x,y}^{shift} = H_{x,x+\hat{1}}^{shift} + H_{x,x-\hat{1}}^{shift} + H_{x,x+\hat{2}}^{shift} + H_{x,x-\hat{2}}^{shift}, \quad (3.18)$$

with

$$H_{x,x+\hat{i}}^{shift} = -\frac{J}{4} (T_x^+ T_{x+\hat{i}}^- + T_x^- T_{x+\hat{i}}^+ + U_x^+ U_{x+\hat{i}}^- + U_x^- U_{x+\hat{i}}^+ + \dots). \quad (3.19)$$

Now, we consider again the state  $|ud\rangle$ , with  $d$  being the spin flavour at the site  $x$  and  $u$  the spin flavour of the neighbour in the  $i$ -direction. Since we defined  $T$  as in equation (3.10), the only non-zero contribution to the eigenvalue is generated by the term  $T_x^+ T_{x+\hat{i}}^-$ . Therefore, the state  $|ud\rangle$  will turn into  $|du\rangle$  with an additional prefactor and we can write

$$H_{x,x+\hat{i}}^{shift} |x\rangle = -\frac{J}{4} T_x^+ T_{x+\hat{i}}^- |x\rangle = -\frac{J}{2} |x+\hat{i}\rangle. \quad (3.20)$$

Taking into account all four neighbouring sites of  $x$ , we obtain

$$\sum_{\substack{y: \text{neigh-} \\ \text{bours of } x}} H_{x,y}^{shift} |x\rangle = -\frac{J}{2} \sum_{\substack{y: \text{neigh-} \\ \text{bours of } x}} |y\rangle. \quad (3.21)$$

Putting it all together, we get

$$H |x\rangle = (H_x + H_{rest}) |x\rangle = J \left\{ \left[ \left( \frac{N-1}{N} \right) (2 - L^2) + \frac{2}{N} \right] |x\rangle - \frac{1}{2} \sum_{\substack{y: \text{neigh-} \\ \text{bours of } x}} |y\rangle \right\}. \quad (3.22)$$

The last step of the calculation is a Fourier transformation to momentum space. Let us see what happens with the third summand in equation (3.22) (with the sum over the neighbours). For example, it contains the state  $|x + \hat{1}\rangle = |x_1 + 1, x_2\rangle$ . Carrying out the Fourier transform for this state only yields

$$\begin{aligned} & \sum_x \exp(ipx) |x_1 + 1, x_2\rangle = \sum_{x_1, x_2} \exp(ip_1 x_1) \exp(ip_2 x_2) |x_1 + 1, x_2\rangle \\ = & \sum_{x'_1, x_2} \exp[ip_1(x'_1 - 1)] \exp(ip_2 x_2) |x'_1, x_2\rangle = \exp(-ip_1) |p\rangle, \end{aligned} \quad (3.23)$$

where we substituted  $x_1$  with  $x'_1 = x_1 + 1$ . Now we are able to compute  $E_p$  :

$$\begin{aligned} H |p\rangle &= \sum_x \exp(ipx) H |x\rangle \\ &= \sum_{x_1, x_2} \exp(ip_1 x_1) \exp(ip_2 x_2) J \left\{ \left[ \left( \frac{N-1}{N} \right) (2 - L^2) + \frac{2}{N} \right] |x\rangle - \frac{1}{2} \sum_{y: \text{neigh-}} |y\rangle \right. \\ &= \left. \left[ J \left( \frac{N-1}{N} \right) (2 - L^2) + \frac{2J}{N} \right] |p\rangle \right. \\ &- \frac{J}{2} [\exp(ip_1) + \exp(-ip_1) + \exp(ip_2) + \exp(-ip_2)] |p\rangle \\ &= \underbrace{J \left[ \left( \frac{N-1}{N} \right) (2 - L^2) + \frac{2}{N} - \cos p_1 - \cos p_2 \right]}_{E_p} |p\rangle. \end{aligned} \quad (3.24)$$

A quick check confirms that if  $p = 0$ , we have indeed  $E_0$  as in equation (3.9). The expansion of the difference  $E_p - E_0$  for small  $p$  takes the form

$$E_p - E_0 = J [(1 - \cos p_1) + (1 - \cos p_2)] \simeq \frac{J}{2} (p_1^2 + p_2^2) = \frac{J}{2} p^2. \quad (3.25)$$

This result is only valid for the fundamental (i.e.  $N$ -dimensional) representation of  $SU(N)$ . For higher dimensional irreducible representations, equation (3.25) is slightly modified:

$$E_p - E_0 \simeq \frac{Jn}{2} p^2, \quad (3.26)$$

where  $n$  is the number of boxes of the corresponding Young Tableau of the chosen  $SU(N)$  representation. For a complete derivation of (3.26), the reader should consult [6].

### 3.3 Spontaneous Symmetry Breaking from $SU(N)$ to $U(N - 1)$

In order to obtain the desired symmetry breaking from  $SU(N)$  to  $U(N - 1)$ , it is necessary to choose the completely symmetric representation of the group  $SU(N)$ . In this case, the  $L^d$ -dimensional vacuum vector can be chosen as  $|0\rangle = |uu \dots u\rangle$ , meaning that all spins of the



lattice have the flavour  $q = u$ . If  $T \rightarrow 0$ , the system spontaneously acquires the ground state  $|0\rangle$ . In other words, the average uniform magnetization vector  $\langle \vec{\mathcal{M}} \rangle$  has a non-zero value. Hence, if  $\beta \rightarrow \infty$ , the system is no longer invariant under  $SU(N)$  transformations. However, a  $SU(N-1)$  transformation that mixes all spins except those with flavour  $u$  leaves the ground state vector unchanged. In addition,  $|0\rangle$  is also invariant under a  $U(1)$  transformation. Therefore, the remaining symmetry of the system is  $SU(N-1) \otimes U(1) = U(N-1)$ . Since  $CP(N-1) = SU(N)/U(N-1)$ , we have arrived at the coset space we aimed at. The Goldstone theorem states that whenever a spontaneous breaking of a continuous symmetry occurs, a number of massless particles are generated: the Goldstone bosons. In our case, these are just the magnons of the  $SU(N)$  ferromagnet (introduced in the previous section 3.2). We will now construct an action for the magnons using the effective theory.

### 3.4 Low-Energy Expansion

We now leave the microscopic model and try to construct an action using the principles of effective field theory. Since we are looking at a low-energy expansion, we need the lowest possible order of derivatives. A possible candidate for  $N = 2$  might look as follows:

$$S_{AFM}[\vec{e}] = \int d^d x \int_0^\beta dt \frac{\rho_s}{2} \left( \partial_i \vec{e} \cdot \partial_i \vec{e} + \frac{1}{c^2} \partial_t \vec{e} \cdot \partial_t \vec{e} \right), \quad (3.27)$$

where we have introduced the direction of the local magnetization  $\vec{e}$  with length 1:  $|\vec{e}| = 1$ . It is related to the matrices  $P(x) \in CP(N-1)$  introduced in chapter 2 by  $P(x) = \frac{1}{2}(\mathbb{1} + \vec{e} \cdot \vec{\sigma})$ . The constant  $\rho_s$  is the spin stiffness and  $c$  is the magnon propagation velocity. Using the principle of least action, we obtain the following magnon dispersion relation for the effective theory:  $E_p^2 = c^2 p^2$ . This is clearly wrong, since we know from the microscopic model that the dispersion relation should be non-relativistic as in (3.26). In fact, equation (3.27) is the action of an antiferromagnet. For a ferromagnet, the action should contain only one derivative with respect to time in order to produce the correct dispersion relation. Hence we are forced to include a different term in our action, the Wess-Zumino-Witten term, in short WZW-term. Written for arbitrary  $N$  in terms of the  $P$ -fields, it has the following form:

$$S_{WZW}[P] = C \int d^d x \int_0^\beta dt \int_0^1 d\tau \text{Tr} [P(\partial_t P \partial_\tau P - \partial_\tau P \partial_t P)], \quad (3.28)$$

and thus the correct expansion for the action to leading order becomes

$$\begin{aligned} S[P] &= \int d^d x \int_0^\beta dt [\rho_s \text{Tr} (\partial_i P \partial_i P)] + S_{WZW} \\ &= \int d^d x \int_0^\beta dt \left\{ \rho_s \text{Tr} (\partial_i P \partial_i P) + C \int_0^1 d\tau \text{Tr} [P(\partial_t P \partial_\tau P - \partial_\tau P \partial_t P)] \right\}. \end{aligned} \quad (3.29)$$

The constant  $C$  preceding the term will now be determined. The WZW-term contains an integration over a deformation parameter  $\tau \in [0, 1]$ . Both dimensions  $t$  and  $\tau$  together can be considered as a circle  $S^1$  including its enclosed area (with radius  $\tau = 1$  and circumference  $t = \beta$ ), which is topologically equivalent to the hemisphere  $H^2$ . We have to interpolate the field  $P(t)$  on  $H^2$  such that we get a smooth function that maps  $(t, \tau) \in H^2$  to  $P(t, \tau) \in S^2$ .

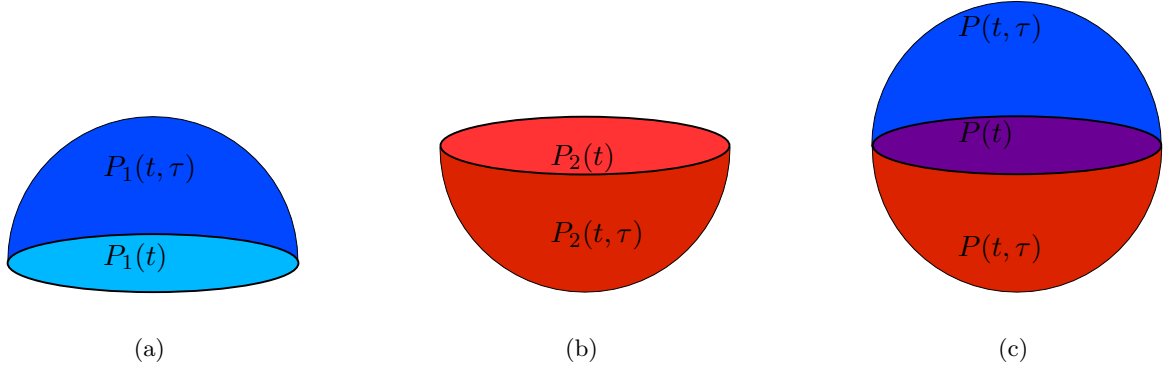


Figure 3.1: The interpolations  $P_1(t, \tau)$  on the hemisphere (a) and  $P_2(t, \tau)$  on the hemisphere (b) are fitted together to the interpolation  $P(t, \tau)$  on the sphere (c).

Since this choice of interpolation is arbitrary, it is instructive to consider the difference between two WZW-terms with different interpolations  $P_1(t, \tau)$  and  $P_2(t, \tau)$ .

$$\begin{aligned}
S_{WZW}(P_1) - S_{WZW}(P_2) &= C \int d^d x \int_{H^2} dt d\tau \operatorname{Tr} [P_1(\partial_t P_1 \partial_\tau P_1 - \partial_\tau P_1 \partial_t P_1)] \\
&\quad - C \int d^d x \int_{H^2} dt d\tau \operatorname{Tr} [P_2(\partial_t P_2 \partial_\tau P_2 - \partial_\tau P_2 \partial_t P_2)] \quad (3.30) \\
&= C \int d^d x \int_{S^2} dt d\tau \operatorname{Tr} [P(\partial_t P \partial_\tau P - \partial_\tau P \partial_t P)].
\end{aligned}$$

In the last step we defined  $P(t, \tau)$  as the combined interpolation of  $P_1(t, \tau)$  and  $P_2(t, \tau)$ . The integral now extends over the sphere  $S^2$ , because the integral in  $-S_{WZW}(P_2)$  extends over an upside-down hemisphere and thus the two  $H^2$  can be fitted together to  $S^2$  (see figure 3.1). Without proof we state that

$$\frac{1}{i\pi} \int_{S^2} dt d\tau \operatorname{Tr} [P(\partial_t P \partial_\tau P - \partial_\tau P \partial_t P)] = K_1, \quad (3.31)$$

where  $K_1 \in \mathbb{Z}$ . By including a factor of  $1/i\pi(La)^d$  in the constant  $C$ , we ensure that the difference between the two WZW-terms is an integer. The factor  $(La)^d$  is needed to compensate the integration over  $d^d x$ . Hence, the ambiguity of interpolations manifests itself in the value of  $K_1$ . The complete action  $S[P]$  in equation (3.29) will appear in the exponent of the Boltzmann weights in  $Z$ . What really matters is that the Boltzmann weights are invariant under different interpolations. Hence, we also include a factor  $2\pi i$  in  $C$ , because then we have factors in  $Z$  resulting from the WZW-part of the action which have the following form:

$$\exp(2\pi i K_1) = 1, \quad (3.32)$$

for all  $K_1 \in \mathbb{Z}$ . Putting it all together, we obtain

$$C = \frac{2}{(La)^d}, \quad (3.33)$$

as a prefactor for the difference of two WZW-terms. We can now choose

$$C = \frac{2K_2}{(La)^d}, \quad (3.34)$$

as the prefactor for a single WZW-term, where we have included another integer  $K_2 \in \mathbb{Z}$  such that the difference still has the properties mentioned above. In principle,  $K_2$  could take any integer value. However, it turns out that  $K_2$  is related to the magnetization  $\langle \vec{M} \rangle$ . An external magnetic field  $\vec{B}$  can be included by replacing the derivative  $\partial_t P$  with the covariant derivative  $D_t P = \partial_t P + \frac{1}{2}[\vec{B} \cdot \vec{\sigma}, P]$ . Then we obtain

$$\langle |\vec{M}| \rangle = \left. \frac{\partial}{\partial B \beta} \ln(Z) \right|_{B=0} = \frac{K_2}{2}. \quad (3.35)$$

Together with the equation  $s = n/2$  which relates the number of boxes  $n$  of the Young tableau of the corresponding  $SU(2)$ -representation to the value of a single spin  $s$  and the definition of the magnetization density  $m = |\vec{M}|/(La)^d = s/a^d$ , we get

$$C = \frac{2n}{a^d}. \quad (3.36)$$

Note that the relation  $m = n/2a^d$  is true for general  $N$ . Thus, the low energy expansion of the effective action for our ferromagnet is

$$S[P] = \int_0^\beta dt \int d^d x \operatorname{Tr} \left[ \rho_s \partial_\mu P \partial_\mu P + \frac{2n}{a^d} \int_0^1 d\tau P (\partial_t P \partial_\tau P - \partial_\tau P \partial_t P) \right]. \quad (3.37)$$

Again using the principle of least action, we derive the following dispersion relation:

$$E_p - E_0 = \frac{2\rho_s}{n} p^2. \quad (3.38)$$

This is non-relativistic and looks just like the result (3.26) obtained from the microscopic model. Comparing these two equations, we are able to relate the spin stiffness  $\rho_s$  from the effective theory to the exchange coupling  $J$ :

$$\rho_s = \frac{Jn^2}{4}. \quad (3.39)$$

### 3.5 Dimensional Reduction

First, we take a look at the case when  $d = 2$ . The correlation length  $\xi = 1/m$  is our physically relevant scale on the lattice. Since the Goldstone bosons are massless for  $\beta = \infty$ , we have  $\xi = \infty$ . What happens when  $\beta$  becomes finite? If we set  $L = \infty$ , the system is essentially only two-dimensional. However, the Mermin-Wagner-Coleman theorem states that a continuous symmetry can not be spontaneously broken in two dimensions. Hence there is no reason

anymore for the Goldstone bosons to be massless and the correlation length  $\xi$  becomes finite. Let us boldly assume for the moment that the correlation length is still much larger than the extent of the extra dimension  $\beta$ , i.e.  $\xi \gg \beta$  (we will see in a moment that this is true). In this case, the second term of the action (3.37) vanishes, because the Goldstone boson fields are essentially constant along the extra dimension of extent  $\beta$  and therefore  $\partial_t P = 0$ . The integration over the extra dimension can be performed trivially and thus the action having lost the WZW-term reduces to

$$S = \beta \rho_s \int d^d x \operatorname{Tr} [\partial_\mu P \partial_\mu P]. \quad (3.40)$$

A glance at the actions (2.6) and (3.40) tells us that we have to identify

$$\frac{1}{g^2} = \beta \rho_s, \quad (3.41)$$

in order to see that our system is now indeed a  $d$ -dimensional  $CP(N-1)$  model and has lost the extra dimension of extent  $\beta$ . To prove that our assumption  $\xi \gg \beta$  is correct, we have to make use of the fact that  $CP(N-1)$  models (like their big brother QCD) have the important property of asymptotic freedom. As already mentioned in chapter 2, as long as the coupling  $g^2$  is sufficiently small, the correlation length depends on  $g^2$  through

$$\xi \propto \exp(4\pi/g^2 N) = \exp(4\pi\beta\rho_s/N) = \exp(\pi\beta J n^2/N). \quad (3.42)$$

Hence, for large (but finite)  $\beta$ , we have indeed  $\xi \gg \beta$ , which is a necessary condition for dimensional reduction. Paradoxically, we can conclude that  $\beta$  vanishes (compared to  $\xi$ ) only if it is sufficiently large.

In case when  $d > 2$ , the Mermin-Wagner-Coleman theorem does not apply, and the spontaneous symmetry breaking occurs even for sufficiently large finite  $\beta > \beta_c$ . Therefore, the correlation length remains infinite. If  $\beta < \beta_c$  (which corresponds to a strong coupling according to equation (3.41)), the system changes into the unbroken phase, the correlation length becomes finite and the  $SU(N)$  symmetry is restored. However, it is possible to fine-tune the value of  $\beta$  from below such that  $\beta \approx \beta_c$ . If the phase transition at  $\beta_c$  is second order, we can obtain  $\xi \rightarrow \infty$ . Thus, the extra dimension of extent  $\beta$  is small compared to the correlation length and dimensional reduction takes place.



## Chapter 4

# Path Integral Representation of $SU(N)$ Quantum Spin Systems

The goal of this chapter is to obtain a path integral representation of the previously introduced quantum partition function

$$Z = \text{Tr} \exp(-\beta H), \quad (4.1)$$

with  $H$  being the Hamilton operator for an  $SU(N)$  ferromagnet,

$$H = -J \sum_{x,i} T_x^j T_{x+i}^j. \quad (4.2)$$

### 4.1 Step 1: Trotter Decomposition of the Hamiltonian

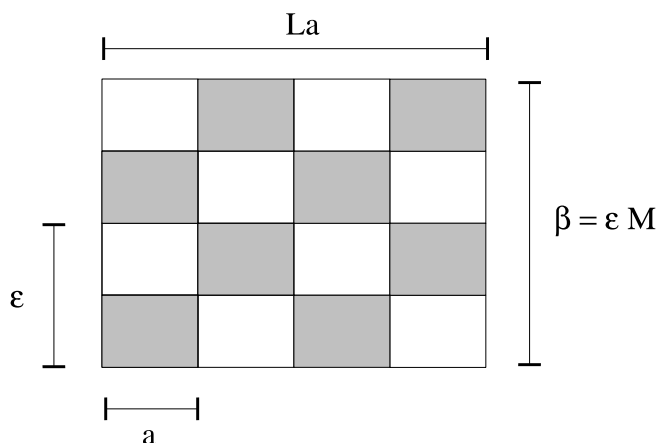


Figure 4.1: Trotterization of the lattice for  $d = 1$ .

First, let us recall a few remarks on dimensions. The spacial extent of our  $(d+1)$ -dimensional lattice is denoted by  $L^d$ , where  $L$  is an even integer. The  $\beta$ -direction represents the extra time-like direction (which should not be confused with the deformation parameter  $\tau$  in the WZW-action-term). It is now split up in  $M$  Euclidean time-slices, each of them with a length

$\epsilon = \beta/M$ . Figure 4.1 gives a schematic overview of the lattice for  $d = 1$ . As mentioned earlier, the dimension which is referred to as the time here will ultimately disappear after the dimensional reduction of the system and should not be confused with the Euclidean time of the  $CP(N-1)$  model. We now split up the Hamiltonian into  $2d$  terms

$$H = H_1 + H_2 + \dots + H_{2d}, \quad (4.3)$$

where the  $H_i$  are defined as

$$H_i = \sum_{\substack{x=(x_1, x_2, \dots, x_d) \\ x_i \text{ even}}} h_{x,i}, \quad H_{i+d} = \sum_{\substack{x=(x_1, x_2, \dots, x_d) \\ x_i \text{ odd}}} h_{x,i}. \quad (4.4)$$

The  $h_{x,i}$  represent the interaction between two neighbouring spins. Every shaded plaquette in figure 4.1 corresponds to such an  $h_{x,i}$ . They are defined as

$$h_{x,i} = -JT_x^j T_{x+\hat{i}}^j. \quad (4.5)$$

Hence, we made a further decomposition; every time-slice has been divided into  $2d$  smaller slices. The partition function can now be rewritten as follows:

$$\begin{aligned} Z &= \text{Tr} \exp(-\beta H) \\ &= \lim_{M \rightarrow \infty} \text{Tr} \exp(-\epsilon M H) \\ &= \lim_{M \rightarrow \infty} \text{Tr} \left\{ \left[ \exp \left( -\epsilon \sum_{i=1}^{2d} H_i \right) \right]^M \right\} \\ &= \lim_{M \rightarrow \infty} \text{Tr} \left\{ \left[ \prod_{i=1}^{2d} \exp(-\epsilon H_i) \right]^M \right\}, \end{aligned} \quad (4.6)$$

where in the last step we have used the Baker-Campbell-Hausdorff formula and neglected higher-order terms in  $\epsilon$ . The smaller the quantity  $\epsilon J$  is, the better is the above approximation (4.6). Therefore, it is important to use small values of  $\epsilon J$  in the algorithm. This implies large values for  $M$ , which is equivalent to a very fine segmentation of the  $\beta$ -direction.

## 4.2 Step 2: Insertion of Complete Sets of Spin States

Now we have to insert a complete set of spin states between every exponential factor in (4.6). They have the following form

$$\mathbb{1} = \sum_{n=1}^{NL^d} |n\rangle \langle n|, \quad (4.7)$$

where  $|n\rangle$  is a state with a number of  $L^d$  spins that can have  $N$  different flavors  $q$ . After having inserted a total number of  $2dM$   $\mathbb{1}$ -operators, we encounter terms of the following form:

$$\langle m | \exp(-\epsilon JT_x^j T_{x+\hat{i}}^j) | n \rangle = \exp\{-s[q(x,t), q(x+\hat{i},t), q(x,t+1), q(x+\hat{i},t+1)]\}. \quad (4.8)$$

These are the transfer matrix elements describing the evolution of two spins from one time-slice to the next. They are equal to the Boltzmann-weight of the corresponding plaquette, i.e. of the four spins connected by the corresponding  $h_{x,i}$ . We denote the action of a single plaquette by  $s$ . The Boltzmann factor of the entire lattice action  $S$  is now expressed as a product of single plaquette contributions:

$$\begin{aligned} \exp(-S[q]) = & \prod_{p=0}^{M-1} \prod_{i=1}^d \prod_{\substack{x=(x_1, x_2, \dots, x_d) \\ x_i \text{ even}, t=2dp+i-1}} \exp\{-s[q(x,t), q(x+i,t), q(x,t+1), q(x+i,t+1)]\} \\ & \times \prod_{\substack{x=(x_1, x_2, \dots, x_d) \\ x_i \text{ odd}, t=2dp+d+i-1}} \exp\{-s[q(x,t), q(x+i,t), q(x,t+1), q(x+i,t+1)]\}, \end{aligned} \quad (4.9)$$

where  $[q]$  stands for a certain configuration of all the spins on the lattice. The partition function now simply is

$$Z = \sum_{[q]} \exp(-S[q]). \quad (4.10)$$

### 4.3 Step 3: Calculating the Boltzmann-Weights of a Single Plaquette

The calculations in this section are done for  $N = 2$ . For a derivation for general  $N$ , the interested reader should consult [6]. For two sites each carrying an  $SU(2)$ -spin, we have four different states:

$$\phi_1 = |u, u\rangle, \quad \phi_2 = |u, d\rangle, \quad \phi_3 = |d, u\rangle, \quad \phi_4 = |d, d\rangle. \quad (4.11)$$

In a 4-dimensional Hilbert-space, they can be written as follows:

$$\phi_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \phi_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad \phi_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \quad \phi_4 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}. \quad (4.12)$$

Since  $N = 2$ , the Hamiltonian in (4.2) is written as

$$H = -J \vec{S}_x \cdot \vec{S}_y, \quad (4.13)$$

with

$$\vec{S}_x \cdot \vec{S}_y = \frac{1}{4} \sum_{j=1}^3 \sigma_x^j \sigma_y^j, \quad (4.14)$$

where the  $\sigma_x^j$  and  $\sigma_y^j$  are the three Pauli-matrices acting at the sites  $x$  and  $y$  respectively:



$$\sigma^1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma^2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma^3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \quad (4.15)$$

Since we know how the Pauli-matrices act on a single spin  $|u\rangle = (1,0)$  or  $|d\rangle = (0,1)$ , it is rather straightforward to rewrite our Hamilton operator (4.13) in such a way that it can act on the spin states (4.12) directly:

$$H = -\frac{J}{4} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 2 & 0 \\ 0 & 2 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (4.16)$$

Multiplying  $H$  by  $-\epsilon$  and exponentiating yields the transfer matrix  $T$ :

$$T = \exp\left(\frac{1}{4}\epsilon J\right) \begin{pmatrix} \exp\left(-\frac{1}{2}\epsilon J\right) & 0 & 0 & 0 \\ 0 & \cosh\left(\frac{1}{2}\epsilon J\right) & -\sinh\left(\frac{1}{2}\epsilon J\right) & 0 \\ 0 & -\sinh\left(\frac{1}{2}\epsilon J\right) & \cosh\left(\frac{1}{2}\epsilon J\right) & 0 \\ 0 & 0 & 0 & \exp\left(-\frac{1}{2}\epsilon J\right) \end{pmatrix}. \quad (4.17)$$

Now we are ready to calculate the transfer matrix elements using equation(4.8):

$$\exp(-s[\phi_i, \phi_j]) = \phi_i^T T \phi_j. \quad (4.18)$$

The results are as follows:

$$\begin{aligned} \exp(-s[\phi_1, \phi_1]) &= \exp(-s[\phi_4, \phi_4]) = \exp\left(\frac{1}{4}\epsilon J\right), \\ \exp(-s[\phi_2, \phi_2]) &= \exp(-s[\phi_3, \phi_3]) = \frac{1}{2} [1 + \exp(-\epsilon J)] \exp\left(\frac{1}{4}\epsilon J\right), \\ \exp(-s[\phi_2, \phi_3]) &= \exp(-s[\phi_3, \phi_2]) = \frac{1}{2} [1 - \exp(-\epsilon J)] \exp\left(\frac{1}{4}\epsilon J\right). \end{aligned} \quad (4.19)$$

All other Boltzmann factors are zero. Multiplying this by  $\exp(-\frac{1}{4}\epsilon J)$  and using the notation from above, we obtain:

$$\begin{aligned} \exp(-s[u, u, u, u]) &= \exp(-s[d, d, d, d]) = 1, \\ \exp(-s[u, d, u, d]) &= \exp(-s[d, u, d, u]) = \frac{1}{2} [1 + \exp(-\epsilon J)], \\ \exp(-s[u, d, d, u]) &= \exp(-s[d, u, u, d]) = \frac{1}{2} [1 - \exp(-\epsilon J)]. \end{aligned} \quad (4.20)$$

It turns out that the above result is independent of  $N$  (see again [6] for details). The flavours  $u$  and  $d$  can be permuted to other values. These Boltzmann weights have direct implications for the cluster rules, which will be examined in the next chapter. Figure 4.2 summarizes the results of equation (4.20).





Plaquette Spin Configurations				
Transfer Matrix Elements	$\exp(-s[u,u,u,u]) = 1$	$\exp(-s[d,d,d,d]) = 1$	$\exp(-s[u,d,u,d]) = \frac{1}{2} [1 + \exp(-\epsilon J)]$	$\exp(-s[u,d,d,u]) = \frac{1}{2} [1 - \exp(-\epsilon J)]$

Figure 4.2: *Non-zero transfer matrix elements of different plaquette configurations. A “x” stands for the spin flavour “down”, a “•” stands for “up”.*



## Chapter 5

# Cluster Algorithm for $CP(N - 1)$ Models with small $N$

Before we can explore the limit  $N \rightarrow \infty$ , we must understand how the algorithm works for finite  $N$ . We will focus on the multi-cluster algorithm, since there seems to be no way of constructing a single-cluster algorithm for  $N \rightarrow \infty$ . However, for the simulation results with moderate  $N$  in chapter 7, we have used the single-cluster algorithm described in [4, 5, 6].

### 5.1 The Monte Carlo Method

As an introduction to this chapter, we will quickly review the concepts of the Monte Carlo simulation applied to the  $SU(N)$  quantum ferromagnet. In our case, the initial configuration  $[C^{(1)}]$  will be the one where all the spins have the same flavour (if we describe configurations by their spins, we have  $[C^i] = [q^i]$ ). Some spins will be flipped according to certain rules (see below) and a new configuration  $[C^{(2)}]$  is generated. This is done repeatedly to form a Markov chain. The probability for a certain configuration  $[C^{(i)}]$  to appear in the chain is  $\exp(-S[C^{(i)}])$ . After a certain number of iterations  $N_{equi}$ , the system will not “remember” its initial state and is said to have reached equilibrium. In  $[C^{(N_{equi})}]$ , almost no traces of the form of the initial spin configuration are left. This is the point where we can begin to measure observables. Now we perform  $N_{meas}$  more iterations and measure the desired quantities  $\mathcal{O}[C^{(i)}]$ ,  $i > N_{equi}$ , after every cluster update. Averaging over  $N_{meas}$  contributions gives the expectation value  $\langle \mathcal{O} \rangle$  :

$$\langle \mathcal{O} \rangle \approx \frac{1}{N_{meas}} \sum_{i=N_{equi}+1}^{N_{equi}+N_{meas}} \mathcal{O}[C^{(i)}]. \quad (5.1)$$

The standard deviation of the observable  $\mathcal{O}$  is

$$\Delta \mathcal{O} = \frac{1}{\sqrt{N_{meas} - 1}} \sqrt{\langle \mathcal{O}^2 \rangle - \langle \mathcal{O} \rangle^2}. \quad (5.2)$$

Hence, the error of  $\mathcal{O}$  decreases by a factor of  $1/\sqrt{N_{meas} - 1}$  for increasing  $N_{meas}$  and (5.1) becomes exact for  $N_{meas} \rightarrow \infty$ . Unfortunately, the standard deviation is only a lower bound for the error. This is due to the fact the two subsequent configurations  $[C^{(i)}]$  and  $[C^{(i+1)}]$  are correlated, meaning that  $[C^{(i+1)}]$  depends on  $[C^{(i)}]$ . Of course, this correlation also affects

$\mathcal{O}[C^{(i)}]$  and  $\mathcal{O}[C^{(i+1)}]$ ; thus the statistical fluctuation of these observables is reduced. However, this problem of incorrect error estimation can be avoided by binning the data: We generate a new data set  $\widehat{\mathcal{O}}(j)$  by reducing the number of measured observables  $\mathcal{O}[C^{(i)}]$  by a factor of  $N_{bin}$  such that

$$\widehat{\mathcal{O}}(j) = \frac{1}{N_{bin}} \sum_{k=(j-1)N_{bin}+1}^{jN_{bin}} \mathcal{O}[C^{(k)}]. \quad (5.3)$$

Now we have a total of  $N_{meas}/N_{bin}$  new observables  $\widehat{\mathcal{O}}(j)$  that can be considered statistically independent for a sufficiently large  $N_{bin}$ . It is obvious that  $\langle \widehat{\mathcal{O}} \rangle = \langle \mathcal{O} \rangle$ . However, the standard deviation will now be greater:  $\Delta(\widehat{\mathcal{O}}) \geq \Delta(\mathcal{O})$ . In order to find an appropriate value for  $N_{bin}$  we gradually increase it until  $\Delta(\widehat{\mathcal{O}})$  has reached a plateau. Only then will we achieve a correct estimate for the error of  $\mathcal{O}$ . It should be noted that this does not represent a general solution to the problem of autocorrelation.

Furthermore, we must ensure that both detailed balance and ergodicity are satisfied for our algorithm. Ergodicity means that our algorithm is capable of reaching every possible spin configuration with non-zero Boltzmann weight in a finite amount of time. Detailed balance guarantees that the probability distribution of configurations converges to the correct distribution where  $p_C = Z^{-1} \exp(-S[C])$ . In other words,  $p_C$  in equilibrium is an eigenvector with eigenvalue 1 of the transition probability matrix  $w$ . The elements  $w[C^{(i)} \rightarrow C^{(j)}]$  of  $w$  are the probabilities to go from the configuration  $[C^{(i)}]$  to  $[C^{(j)}]$ . Their normalization is  $\sum_j w[C^{(i)} \rightarrow C^{(j)}] = 1$ . Using the condition of detailed balance, namely

$$p_{C^{(i)}} w[C^{(i)} \rightarrow C^{(j)}] = p_{C^{(j)}} w[C^{(j)} \rightarrow C^{(i)}], \quad (5.4)$$

it is easy to show that  $w p_C = p_C$ :

$$\begin{aligned} p_{C^{(j)}} &= \sum_i w[C^{(i)} \rightarrow C^{(j)}] p_{C^{(i)}} = \sum_i p_{C^{(i)}} w[C^{(j)} \rightarrow C^{(i)}] \frac{p_{C^{(j)}}}{p_{C^{(i)}}} \\ &= p_{C^{(j)}} \sum_i w[C^{(j)} \rightarrow C^{(i)}] = p_{C^{(j)}}. \end{aligned} \quad (5.5)$$

## 5.2 Cluster Rules

This section explains exactly how a new configuration is generated by the multi-cluster algorithm. Let us consider an arbitrary spin configuration  $[C^{(i)}]$  on the  $d+1$ -dimensional lattice. Figure 5.1(a) shows an example of a possible configuration for  $d=1$ ,  $L=4$ ,  $M=2$  and  $N=2$  (the same sample lattice is used for all subsequent figures in this section). Again, the colored regions indicate the positions of the plaquettes, i.e. four lattice sites affected by the Hamiltonian interaction. It is now useful to introduce the concept of bonds: Each plaquette is tagged with either a parallel bond or a cross bond according to rules we will discuss below. This is shown in figure 5.1(b). These bonds are to the cluster what the chain links are to a chain; they connect two lattice sites that are now part of the cluster. Hence, after having visited every plaquette and having tagged it with a certain bond, every lattice site belongs to one and only one cluster. A cluster contains spins that have the same flavour. Since both

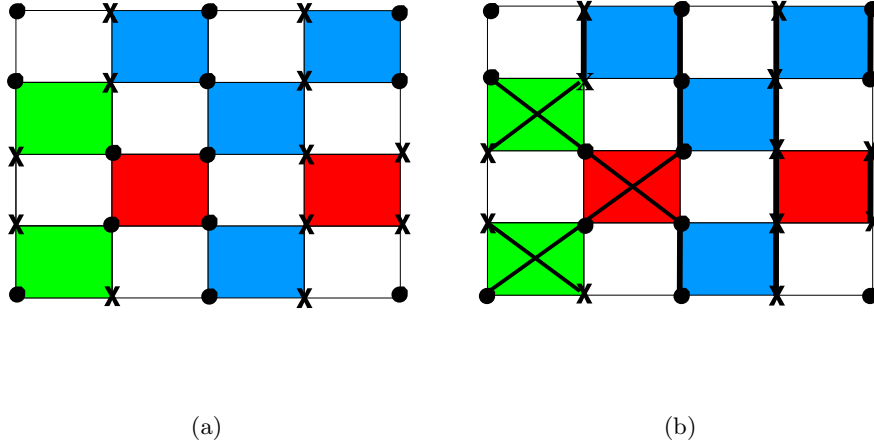


Figure 5.1: (a) A possible spin configuration of the lattice. Different plaquette colors indicate different spin configurations. (b) A possible bond configuration of the same lattice. Note that the plaquette with four spins of the same flavour (red) can have either a cross-bond or a parallel bond.

bond types connect sites belonging to different time slices, clusters can close only by wrapping around the  $\beta$ -direction due to periodic boundary conditions. Furthermore, we note that  $n_c \in \{1, \dots, L^d\}$ , where  $n_c$  is the number of clusters. In case  $n_c = 1$ , all sites belong to the same cluster. See figure 5.2(a) for details. After all clusters have been formed, the algorithm decides which spins to change. This is done by deciding for every cluster to either flip all spins belonging to it, or none. The flavour of the new spin is selected randomly among the  $N$  possibilities. If  $N = 2$ , the spin is flipped with a probability of 0.5. We can see immediately that the multi-cluster algorithm is not ideal, since there is a slight possibility that no spins are flipped, and thus  $[C^{(i+1)}] = [C^{(i)}]$ . Of course, the bonds have been placed in such a way that only allowed configurations appear in the Markov chain. In fact, the probabilities of those two bond types are directly related to the transfer matrix elements in equation (4.20). This means that only the following combinations of spin flavours of the four plaquette neighbours are possible:

**Case 1** : Consider the plaquette configuration  $C^1 = [q^a, q^b, q^a, q^b]$  with  $a \neq b$ . In this case, the only possibility is a parallel bond. If we tagged the plaquette with a cross-bond, it is possible that a configuration  $C^* = [q^a, q^b, q^a, q^a]$  would be generated. The corresponding matrix element is zero, and we would end up with a physically forbidden configuration.

**Case 2** : The second possibility is when the configuration has the form  $C^2 = [q^a, q^b, q^b, q^a]$  with  $a \neq b$ . Here we need a cross-bond for our plaquette. Placing a parallel bond would again result in a forbidden spin configuration.

**Case 3** : All four sites of the plaquette carry the same spin flavour:  $C^3 = [q^a, q^a, q^a, q^a]$ . Now, both bond types are possible. Placing a cross-bond results either in case 1 or again in case 3. Placing a parallel bond results either in case 2 or again in case 3. We

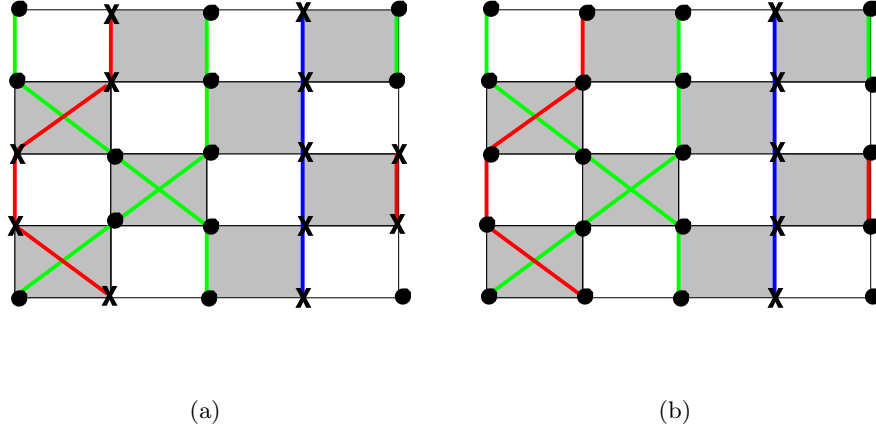


Figure 5.2: (a) Identifying the clusters. Different colors mean different clusters. (b) The spins on the sites belonging to the red cluster have been flipped from “down” to “up”.

now set the probability for a parallel bond to

$$p = \frac{1}{2}[1 + \exp(-\epsilon J)]. \quad (5.6)$$

Thus, the probability for a cross-bond is

$$1 - p = \frac{1}{2}[1 - \exp(-\epsilon J)]. \quad (5.7)$$

We note that both bond-types connect spins of the same flavour. Hence, all spins belonging to the same cluster indeed have the same spin flavour. Moreover, our choice of bond weights ensures detailed balance. According to equation (4.20), the probabilities for the three different plaquette spin configurations are:

$$\begin{aligned} p_{C^1} &= \exp(-s[q^a, q^b, q^a, q^b]) = 1, \\ p_{C^2} &= \exp(-s[q^a, q^b, q^b, q^a]) = p, \\ p_{C^3} &= \exp(-s[q^a, q^a, q^a, q^a]) = 1 - p. \end{aligned} \quad (5.8)$$

We chose the transition probabilities to be:

$$\begin{aligned}
w[C^1 \rightarrow C^2] &= 0, \\
w[C^1 \rightarrow C^3] &= \frac{1}{N^2}, \\
w[C^2 \rightarrow C^3] &= \frac{1}{N^2}, \\
w[C^2 \rightarrow C^1] &= 0, \\
w[C^3 \rightarrow C^1] &= p \frac{1}{N^2}, \\
w[C^3 \rightarrow C^2] &= (1-p) \frac{1}{N^2}.
\end{aligned} \tag{5.9}$$

The factor  $N^{-2}$  arises because we have to flip two clusters per plaquette, so the new spin flavour of the sites could take any of the  $N$  possibilities. Indeed, the detailed balance condition (5.4) is satisfied:

$$\begin{aligned}
p_{C^1} w[C^1 \rightarrow C^2] &= p_{C^2} w[C^2 \rightarrow C^1] = 0, \\
p_{C^1} w[C^1 \rightarrow C^3] &= p_{C^3} w[C^3 \rightarrow C^1] = p \frac{1}{N^2}, \\
p_{C^2} w[C^2 \rightarrow C^3] &= p_{C^3} w[C^3 \rightarrow C^2] = (1-p) \frac{1}{N^2}.
\end{aligned} \tag{5.10}$$

In the case that all four sites of the configuration  $C^3$  belong to the same cluster, we have  $w[C^i \rightarrow C^j] = 0$  for all  $i, j \in 1, 2, 3$ , and detailed balance is satisfied trivially. The algorithm is also ergodic, since every possible configuration can be generated. Having flipped the spins belonging to the clusters we chose to change, we arrive at the new configuration  $[C^{(i+1)}]$ , illustrated in figure 5.2(b). Before starting the whole updating process again, we measure some observables.

## 5.3 Observables and Improved Estimators

Our algorithm measures three different physical quantities after each cluster update: The uniform magnetization, the uniform magnetic susceptibility and the two-point correlation function. The first two are mainly used as a method of comparison to crosscheck the data with already existing algorithms. The purpose of measuring the correlation function is explained below.

### 5.3.1 Uniform Magnetization

The uniform magnetization in the  $u/d$ -direction can be defined as

$$\mathcal{M} = \sum_{x,t} [\delta_{q(x,t),u} - \delta_{q(x,t),d}]. \tag{5.11}$$

We limit ourselves to the calculation of the uniform magnetization in the  $u/d$ -direction. Equation (5.11) tells us that this is a very straightforward task: We simply count the number of sites with spin  $q = u$  and subtract the number of sites with  $q = d$ .



### 5.3.2 Uniform Magnetic Susceptibility

The uniform susceptibility is related to the magnetization by

$$\chi = \frac{1}{\beta L^d} \langle \mathcal{M}^2 \rangle. \quad (5.12)$$

We could measure the susceptibility directly with the formula (5.12). However, there is a much more efficient way to obtain  $\chi$ . First, let us rewrite the total magnetization  $\mathcal{M}$  as a sum over all cluster magnetizations  $\mathcal{M}_C$ :

$$\mathcal{M} = \sum_C \mathcal{M}_C. \quad (5.13)$$

This is correct since every lattice site belongs to exactly one cluster. The cluster magnetization is simply the magnetization of all the spins belonging to the cluster:

$$\mathcal{M}_C = \sum_{(x,t) \in C} [\delta_{q(x,t),u} - \delta_{q(x,t),d}]. \quad (5.14)$$

Now we can rewrite equation (5.12) as follows:

$$\chi = \frac{1}{\beta L^d} \langle \mathcal{M}^2 \rangle = \frac{1}{\beta L^d} \langle (\sum_C \mathcal{M}_C)^2 \rangle = \frac{1}{\beta L^d} \langle \sum_{C_1, C_2} \mathcal{M}_{C_1} \mathcal{M}_{C_2} \rangle. \quad (5.15)$$

The following relation helps us to simplify the above expression: If  $C_1 \neq C_2$ , we have

$$\langle \mathcal{M}_{C_1} \mathcal{M}_{C_2} \rangle = 0. \quad (5.16)$$

This is true because two magnetizations from different clusters are independent and thus their product averages to zero. The algorithm does not have to explicitly create these configurations, measure their magnetizations and average them. Thus we are able to improve the statistics by a factor of  $2^{n_c}$ . And since  $n_c \propto L^d$ , the improvement is very significant for big volumes. The susceptibility now takes the form

$$\chi = \frac{1}{\beta L^d} \langle \sum_C \mathcal{M}_C^2 \rangle. \quad (5.17)$$

Since a cluster contains only sites with equal spin, the absolute value of the cluster magnetization  $|\mathcal{M}_C|$  is just the size of the cluster, i.e. the number of sites belonging to the cluster:

$$|\mathcal{M}_C| = |\mathcal{C}| = \sum_{(x,t) \in C} 1. \quad (5.18)$$

Finally, we have

$$\chi = \frac{1}{\beta L^d} \langle \sum_C |\mathcal{C}|^2 \rangle. \quad (5.19)$$

### 5.3.3 Correlation Function

The two-point correlation function is defined as

$$G(x - y) = \frac{1}{Z} \text{Tr} \left[ \int_0^\beta dt_1 T_{x,t_1}^j \int_0^\beta dt_2 T_{y,t_2}^j \exp(-\beta H) \right]. \quad (5.20)$$

For  $N = 2$ , the improved estimator for the correlation function is derived as follows:

$$\begin{aligned} G(x - y) &= \langle \vec{S}_x \cdot \vec{S}_y \rangle \stackrel{*}{=} 3 \langle S_x^3 S_y^3 \rangle = 3 \langle S_o^3 S_{y-x}^3 \rangle \\ &= \frac{3}{L^d \beta} \langle \sum_z S_z^3 S_{y-x+z}^3 \rangle = \frac{3}{L^d \beta} \sum_{\mathcal{C}} \langle \sum_{z \in \mathcal{C}} S_z^3 S_{y-x+z}^3 \rangle \\ &\stackrel{**}{=} \frac{3}{L^d \beta} \langle \sum_{\mathcal{C}} \sum_{z \in \mathcal{C}} \frac{1}{2} \delta_{z, y-x+z}^{\mathcal{C}} \rangle = \frac{3}{2L^d \beta} \langle \sum_{\mathcal{C}} |\mathcal{C}| \delta_{z_0, y-x+z_0}^{\mathcal{C}} \rangle, \end{aligned} \quad (5.21)$$

where  $\vec{S}_x \cdot \vec{S}_y$  is defined in equation (4.14) and  $\delta_{x,y}^{\mathcal{C}}$  is equal to one if  $x$  and  $y$  are on the same cluster and zero otherwise. Equality “\*” is due to symmetry properties and “\*\*” is true because all sites on the same cluster have the same spin direction. The variable  $z_0$  is an arbitrary site on the cluster. The result of equation (5.21) can be generalized to arbitrary  $N$ . Instead of  $S^3$ , we choose one of the  $N - 1$  diagonal generators  $\widehat{T}^j$ . The only thing that changes is the prefactor. However, this is irrelevant, since we are mainly interested in the correlation length  $\xi$ .

### 5.3.4 Correlation Length and the Second Moment Method

For  $d = 2$ , the Fourier transform of  $G$  is

$$\tilde{G}(p, x_2) = \sum_{x_1} \exp(ipx_1) G(x, t), \quad (5.22)$$

where  $x = (x_1, x_2)$ . For  $p = 0$ , we have

$$\tilde{G}(p = 0, x_2) \propto \exp[-x_2 E(p = 0)] = \exp(-x_2/\xi). \quad (5.23)$$

The distance  $\xi = 1/m = E(0)$  is the correlation length. It is a measure how far the spins can influence each other. On a lattice with periodic boundary conditions, we have a cosh-function instead of the exponential function:

$$\tilde{G}(0, x_2) \propto \cosh[(x_2 - L/2)/\xi]. \quad (5.24)$$

The algorithm produces a numerical output of the correlation function in form of a histogram. The correlation length can be extracted from this function by two different ways: Either by making a fit with a cosh-function or by using the second moment method. For the results listed in chapter 7, we have used the latter method. Both procedures would yield the same value for  $\xi$  if the output generated by the algorithm was exactly a cosh-function. The formula of the second moment method for  $d = 2$  is

$$\xi = \frac{L}{2\pi} \sqrt{\frac{\tilde{G}(0)}{\tilde{G}(2\pi/L)} - 1}. \quad (5.25)$$

## 5.4 Summary

The following list is a summary of the steps performed by the algorithm during one iteration:

1. Visit all plaquettes and place bonds according to the bond rules described above.
2. Identify clusters.
3. Flip the spins to a randomly selected new flavour (if  $N = 2$ , the spins are flipped with a probability of  $1/2$ ).
4. Measure and store observables.
5. Reset clusters and bonds.

## Chapter 6

# The Algorithm in the Large $N$ Limit

The previous chapter discussed the algorithm for finite  $N$ . There are two reasons why it would not work for  $N \rightarrow \infty$ : The obvious reason is simply the fact that a computer program cannot handle arbitrarily high values of  $N$ . Second, the algorithm would be very inefficient for large  $N$ . Let us consider for example an initial configuration where all bonds are parallel, but all clusters have a different spin. This corresponds to case 1 in section 5.2 for every plaquette. The next configuration again will have only cross-bonds, but the clusters will have changed spin. However, the chance that two neighbouring clusters now have the same spin and therefore enabling a cross-bond in the third configuration is suppressed by the factor  $1/N$ . For  $N \rightarrow \infty$  the system becomes frozen and it requires an infinite amount of time to reach a different configuration. The following two sections provide us with the solution to these problems. Another obstacle which is of importance for large  $N$ , the t'Hooft limit, is investigated in section 6.3.

### 6.1 Getting Rid of Spins

It is possible to rewrite the partition function (4.10) as a sum over bond breakup types instead of a sum over spin configurations. What we had so far in (4.9) and (4.10) was

$$Z = \sum_{[q]} \prod_{\text{plaquettes}} \exp(-s[q^a, q^b, q^c, q^d]), \quad (6.1)$$

where  $s[q^a, q^b, q^c, q^d]$  is the action of a single plaquette connecting the sites  $(x, t)$ ,  $(x + \hat{1}, t)$ ,  $(x, t + 1)$  and  $(x + \hat{1}, t + 1)$ . Using the notation in (6.1), we distinguish configurations by their spins  $[q]$ . However, configurations can also be characterized by their bond types. For this purpose, let us consider an arbitrary configuration with certain bond types on every plaquette. Since we have  $n_c$  clusters that can have  $N$  possible spins each, we have a total of  $N^{n_c}$  possibilities for a configuration with a specific ensemble of bond types. Hence we can write

$$Z = \sum_{[b]} \prod_{\text{plaquettes}} \exp[-s(b)] N^{n_c}. \quad (6.2)$$

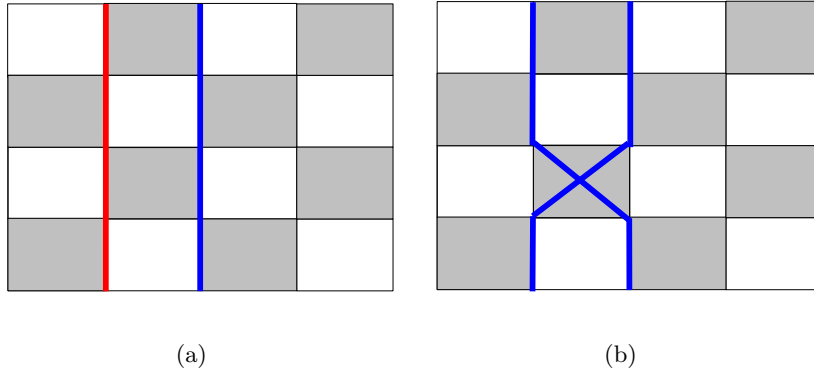


Figure 6.1: *Example of a possible plaquette update: Two different clusters (a) are combined to one (b) by switching the parallel bond to a cross-bond.*

Now, the sum runs over all possible bond type configurations  $[b]$ . A single bond type  $b$  is either a cross-bond or a parallel bond:  $b \in \{\times, \parallel\}$ . The action of such a bond type is denoted as  $s(b)$ , and we have

$$\exp -[s(\parallel)] = p = \frac{1}{2}[1 + \exp(-\epsilon J)], \quad \exp -[s(\times)] = 1 - p = \frac{1}{2}[1 - \exp(-\epsilon J)]. \quad (6.3)$$

Using the idea behind (6.2), we modify our multi-cluster algorithm in the following way: The lattice sites do not carry spins anymore. The fact that we have  $N$  different spins is implicitly included in the bond weights. Again, in the initial configuration all bonds are parallel. The algorithm now wanders from plaquette to plaquette and sets the bond type. Once more, let us consider just a single plaquette which is about to be tagged with a bond type. Let us assume that the sites  $(x, t)$  and  $(x, t + 1)$  belong to the same cluster, as do  $(x + \hat{i}, t)$ , and  $(x + \hat{i}, t + 1)$ . A cross-bond would then reduce the number of clusters by one, as can be seen in figure 6.1. Therefore, due to the factor  $N^{n_c}$  in (6.2), the possibility for a parallel bond  $p$  needs to be increased by a factor of  $N$ :

$$\hat{p} = \frac{Np}{Np + 1 - p}, \quad (6.4)$$

whereas the possibility of a cross bond is  $1 - \hat{p}$ . In fact, we have a total of four different cases that result in two different bond weights (see figure 6.2 for details). After these weights have been set, we roll the dice and tag the plaquette with the bond. This is done for every plaquette on the lattice. Under this aspect, the algorithm contains a Metropolis-like subalgorithm executed in every iteration step. In the end of the process, the clusters are identified and the observables introduced in section 5.3 are measured. Naturally, bonds and clusters are not reset since they form the starting position for the next updating process.









Plaquette Bond Configuration before Update				
Possible Plaquette Bond Configuration after Update				
Probability to set a Parallel Bond	$\hat{p} = Np / (Np+1-p)$		$\hat{p} = p / (N-Np+p)$	

Figure 6.2: Overview of the bond probabilities for different bond configurations. The probability for a parallel bond is  $\hat{p}$ , for a cross-bond it is  $1 - \hat{p}$ . The quantity  $p$  was defined in equation (3.6). Note that this choice of bond weights favours the configuration with more clusters by a factor  $N$ .

## 6.2 Artificial Weights

What happens now if we take the limit  $N \rightarrow \infty$ ? The bond type which would result in a configuration with fewer clusters than our current one would be equal to zero. So we are still in the situation where the whole configuration is unable to change its bonds. Talking in terms of configuration space, we need to leave the subspace where  $n_c = L^d$  in order to generate a different configuration. To arrive at new configurations, we should at least visit the region where  $n_c = L^d - 1$  from time to time. This detour is necessary to re-enter the  $n_c = L^d$  region at a different place. We do this by introducing artificial weights. So far, as soon as the algorithm proposed to enter the  $n_c = L^d - 1$  region, the corresponding weight was set to zero. We now redefine this weight and artificially increase it by a factor of  $N$ . A proposal to enter the  $n_c = L^d - 2$  subspace is left untouched (i.e. the corresponding probability remains zero), thus we never have less than  $n_c = L^d - 1$  clusters<sup>1</sup>. Also, as long as we stay in the region where  $n_c = L^d$ , nothing needs to be changed. However, since we meddled with the weights that lead us into the  $n_c = L^d - 1$  subspace, we must remember to reweight the observables measured in that configuration. In our new Markov chain, the configurations having  $n_c = L^d - 1$  clusters appear  $N$  times too often. Thus, if we want to measure in a configuration where  $n_c = L^d - 1$ , we need to multiply our measured observables by a value of  $1/N$  (if  $N = \infty$ , we simply do not measure them at all). This process is called reweighting. Note that the algorithm is now no longer ergodic, since we do not sample the entire configuration space anymore. However, those configurations which can not be generated anymore have a Boltzmann-weight which is suppressed by a factor  $N^{(n_c - L^d)}$ . Hence, for  $N = \infty$ , the reweighted bond weights are  $p$  and  $1 - p$ , just like for formulation of the algorithm for small  $N$ . Also note that in this case, the

<sup>1</sup>In chapter 7, we modify the algorithm such that it updates five different configuration space regions.

value for the magnetic susceptibility  $\chi$  given in equation (5.19) can be calculated directly:

$$\chi = \frac{1}{\beta L^d} \langle \sum_{\mathbf{c}} |C|^2 \rangle = \frac{1}{\beta L^d} \sum_1^{n_c=L^d} \beta^2 = \beta. \quad (6.5)$$

### 6.3 The t'Hooft limit

The question arises whether we are allowed to take  $N = \infty$  directly. According to the t'Hooft limit, we are not allowed to increase the number of flavours  $N$  without decreasing the coupling strength  $g^2$ . This means that while taking the limit  $N \rightarrow \infty$ , we have to ensure that the combination  $Ng^2$  is kept fixed. Using the formulas (3.41) and (3.39), we obtain

$$Ng^2 = \frac{N}{\rho_s \beta} = \frac{4N}{Jn^2 \beta} = \text{fixed}. \quad (6.6)$$

Condition (6.6) tells us that we have several options how to approach the t'Hooft limit. Note that the t'Hooft condition can be seen directly in equation (3.42). If  $N$  grows, the correlation length decreases which destroys the mechanism of dimensional reduction (since then  $\xi < \beta$ ).

#### 6.3.1 Increasing $n$

If we increase  $n \propto \sqrt{N}$  and keep all the other variables in (6.6) constant, the t'Hooft condition is satisfied. For  $SU(2)$ , this could be realized by having  $n$  spins with spin-value  $1/2$  on each lattice site to simulate a quantum spin system with  $s = n/2$ . However, generalized to arbitrary  $N$ , this would increase the computational effort by a factor  $N$ , since a nearest neighbour coupling would now consist of the  $n^2$  interaction of a total of  $2n$   $1/2$ -spins. It is clear that by choosing this way of satisfying the t'Hooft limit, it is impossible to directly set  $N = \infty$ . This method is not investigated in this work.

#### 6.3.2 Increasing $J$

We could also keep  $n$  and  $\beta$  fixed and increase  $J \propto N$ . Again, this has a consequence on the computational effort: Since we used the Baker-Campbell-Hausdorff approximation in equation (4.6), we have to decrease the length of a time-slice  $\epsilon \propto 1/J$ . Since  $\beta = \epsilon M$  should remain fixed, we conclude that  $M \propto N$ . Just as above, the computational effort would increase by a factor  $N$  and it would not be possible to perform simulations at  $N = \infty$ .

#### 6.3.3 Increasing $\beta$

The last option would be to increase  $\beta$ . Since we do not want a coarser lattice, we would have to increase again  $M$  proportional to  $N$ . The question then arises whether the continuum limit  $\beta \rightarrow \infty$  and the large  $N$  limit commute. By setting  $N = \infty$  directly and setting  $\beta$  sufficiently large to approximate the continuum limit, the t'Hooft condition might be satisfied automatically. However, this is most likely not the case and should be investigated more carefully. Since we have not completely understood how the two limits need to be approached, we decided to increase  $\beta$  only to reach the continuum limit and satisfy the t'Hooft condition by increasing  $J$  and  $M$ .

# Chapter 7

## Results

This chapter is divided into two parts: In the first section, we explore whether the algorithm behaves as suggested by the t'Hooft condition. The second part presents the results obtained by the algorithm introduced in chapter 6 while disregarding the t'Hooft condition. All of the following simulations were carried out on a  $(2 + 1)$ -dimensional lattice, i.e.  $d = 2$ . The errors of  $\xi$  are the standard deviations obtained by performing five simulations each.

### 7.1 Simulations with moderate $N$

Since the t'Hooft condition forces us to increase  $M$  linearly with  $N$ , calculation time also increases. Thus, if we want to approach the t'Hooft limit correctly, we have to limit ourselves to moderate  $N$ . For low  $N$ , the formulation of the algorithm presented in chapter 5 is more efficient than the one for large  $N$  described in chapter 6. This is due to the fact that the latter has to visit the volume of the lattice roughly twice: First, every plaquette needs to be visited. Second, if two clusters are turned into one or vice versa, the new cluster(s) have to be renumerated in order to keep track of the number of clusters and their size. Hence, for moderate  $N$ , it is favorable to work with the algorithm that includes spins. In fact, the results presented in this section have been obtained by a single-cluster algorithm. The only difference to the multi-cluster algorithm is that during a Monte Carlo update, just one cluster is formed and flipped with a probability of 1. The formulas for the improved estimators change slightly, the interested reader should consult [4, 5, 6] for details. Relation (3.42) predicts that if we increase both  $N$  and  $J$ , the correlation length should stay roughly constant. To check this, we first performed a simulation at  $N = 3$  and measured the correlation length using the second moment method. Then we set  $N = 30$  and increased  $J$  by a factor of 10. As we can see in table 7.1, the correlation length did not change much.

$N$	$J$	$M$	$\epsilon$	$\beta$	$L$	$N_{equi}$	$N_{meas}$	$\xi$	$\Delta\xi$
3	1	45	0.1	45	20	$10^4$	$10^5$	8.83	0.03
30	10	450	0.01	45	20	$10^4$	$10^5$	11.91	0.06

Table 7.1:  $\xi(N)$  while adjusting  $J$ .



$N$	$J$	$M$	$\epsilon$	$\beta$	$L$	$N_{equi}$	$N_{meas}$	$\xi$	$\Delta\xi$
2	1	45	0.1	45	20	$10^4$	$10^5$	12.49	0.05
3	1	45	0.1	45	20	$10^4$	$10^5$	8.83	0.03
4	1	45	0.1	45	20	$10^4$	$10^5$	5.93	0.03
5	1	45	0.1	45	20	$10^4$	$10^5$	3.78	0.02
8	1	45	0.1	45	20	$10^4$	$10^5$	1.77	0.02
16	1	45	0.1	45	20	$10^4$	$10^5$	0.98	0.02
32	1	45	0.1	45	20	$10^4$	$10^5$	0.72	0.01
64	1	45	0.1	45	20	$10^4$	$10^5$	0.63	0.01

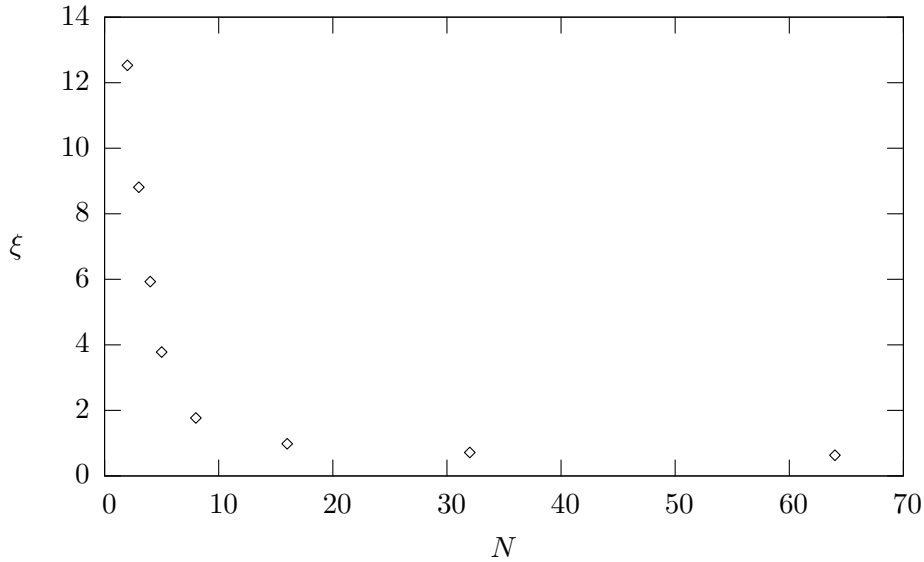
Table 7.2:  $\xi(N)$  with constant  $J$ .

Figure 7.1: Plot of the data in table 7.2.

By adjusting  $J$  only and leaving all other parameters (including  $M$ ) constant, we measured the correlation length given in table 7.2. This is plotted in figure 7.1; the data roughly indicates the  $\xi \propto \exp(1/N)$  behavior.

## 7.2 Simulations with $N = \infty$

Because of the t'Hooft limit, simulating directly at  $N = \infty$  is impossible even with the algorithm presented in chapter (6). Nevertheless, we would like to see whether the algorithm could principally operate at  $N = \infty$ . For this reason, we decided to simply ignore the t'Hooft limit. Let us briefly recall the steps performed by the algorithm during one update:

1. Metropolis-like subalgorithm: walk through all plaquettes and propose to change bonds, using the bond weights  $p$  and  $1 - p$  introduced in equation (5.6).
2. Identify clusters and determine  $n_c$ .

3. If  $n_c = L^d$ , measure and store observables.

The first few simulations revealed another problem: Although the algorithm did switch between the two different configuration space regions, this occurred far too rarely. Most of the time, the algorithm stayed in the region where  $n_c = L^d - 1$ . In other words, most of the generated configurations in the Markov chain contained an odd number of cross-bonds. Therefore, the number of configurations where observables could be extracted was very small. This is undesirable since we need to simulate for an overly long time to get meaningful expectation values. To solve this problem, we introduce a parameter  $t$  in the algorithm that controls the ratio of the number of configurations with  $n_c = L^d$  and  $n_c = L^d - 1$ . This can be done in the following way: Suppose the current configuration has  $n_c = L^d - 1$  and setting a cross-bond would take us back to the desired  $n_c = L^d$  sector. In this case, we would choose the ratio of the probability for a cross-bond to a parallel bond as  $t(1-p)/p$ . The parameter  $t$  is inserted always such that the configuration with  $n_c = L^d$  is favoured over the one with  $n_c = L^d - 1$ . The value for  $t$  is then found by tuning it such that the algorithm switches between the two sectors many times. A perfectly tuned  $t$  would result in a Markov chain consisting of configurations with alternating  $n_c$ . The value of  $t$  depends on the volume of the lattice and on  $\epsilon J$ . It might also be possible to find a formula so one could calculate the value of  $t$  directly. The data for two different values of  $t$  is presented in table 7.3.

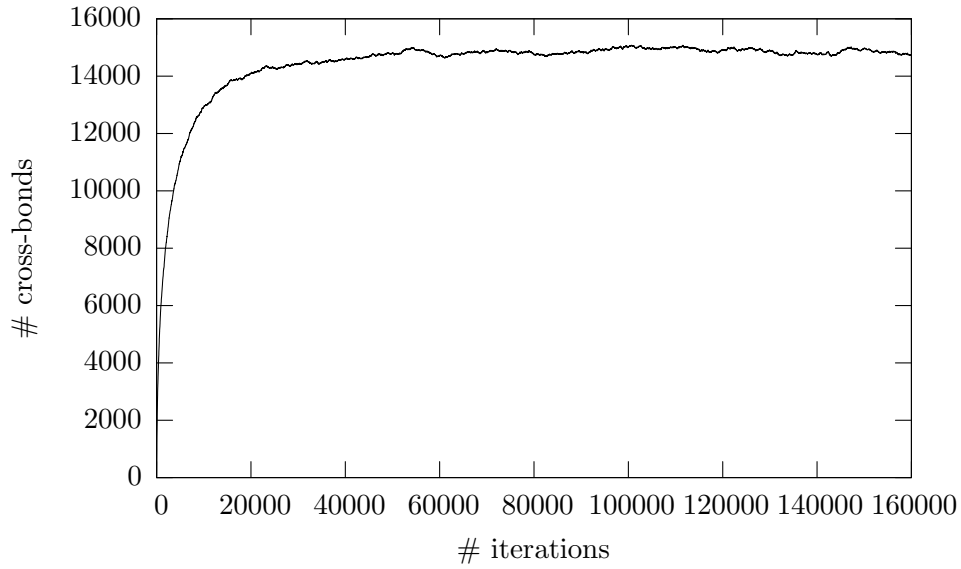
$N$	$J$	$M$	$\epsilon$	$\beta$	$L$	$N_{equi}$	$N_{meas}$	$t$	% $L^d$	# const.	$\xi$	$\Delta\xi$
$\infty$	1	400	0.1	40	20	$10^5$	$10^5$	1	$\sim 0.005$	$\sim 10^5$	1.95	0.03
$\infty$	1	400	0.1	40	20	$10^5$	$10^5$	0.00005	52	1.93	1.94	0.02

Table 7.3: *The effect of different values of  $t$ . The first row is the data for the untuned algorithm, while in the second row  $t$  has been tuned such that the Markov chain contains roughly the same number of configurations with  $n_c = L^d$  as with  $n_c = L^d - 1$ . The column “%  $L^d$ ” lists the percentage of configurations with  $n_c = L^d$ , and “# const.” lists the average number of consecutive configurations with constant  $n_c$ . Since we neglected the  $t$ ’Hooft limit, we have  $\xi < \beta$  and thus dimensional reduction did not take place.*

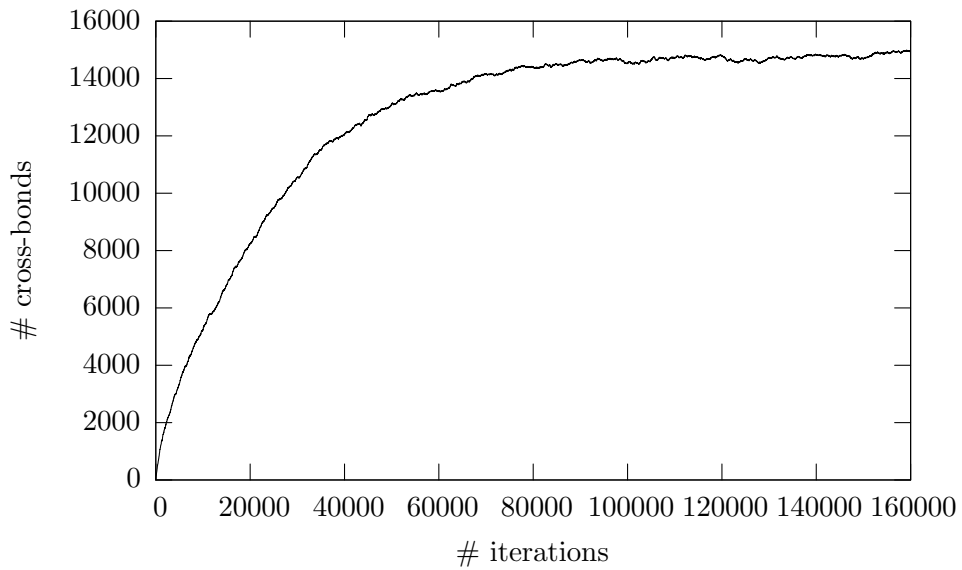
Unfortunately, the introduction of the tuning parameter  $t$  gives rise to a new problem: It seems that the well tuned algorithm takes more steps to reach the thermal equilibrium. An easy to measure quantity to compare the time required until thermal equilibrium is reached is the number of cross-bonds. Since its plateauing is a necessary but not a sufficient condition for thermal equilibrium, we use it only as method of comparison for different simulations. Figure 7.2 shows how the number of cross-bonds raises as the Monte Carlo time increases.

To accelerate thermalization, it might be useful to allow more than just the two configurations space regions with  $n_c = L^d$  or  $n_c = L^d - 1$ . Table 7.4 and figure 7.3 present the data obtained by an algorithm which allows five different configurations, namely  $n_c \in \{L^d - 4, \dots, L^d\}$ . Indeed, the number of cross-bonds plateaus faster than before.

Even after thermalization, the algorithm seems to be less efficient for  $N = \infty$  than for small  $N$ . All four plots in figures 7.2 and 7.3 show correlations of the number of cross-bonds extending over 1000 Monte Carlo iterations or more (this undesired effect is best seen in figure 7.2 (a) starting around step #55000).



(a)

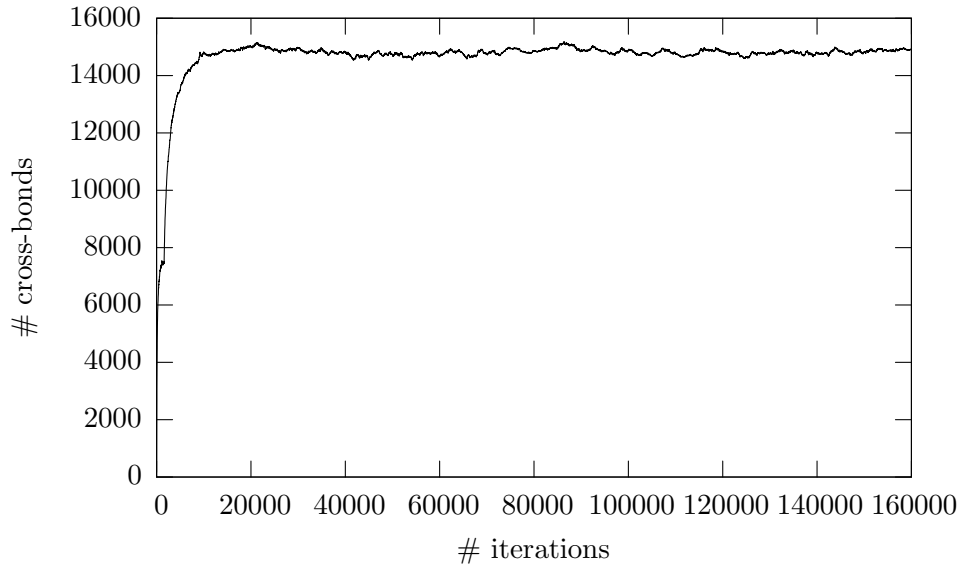


(b)

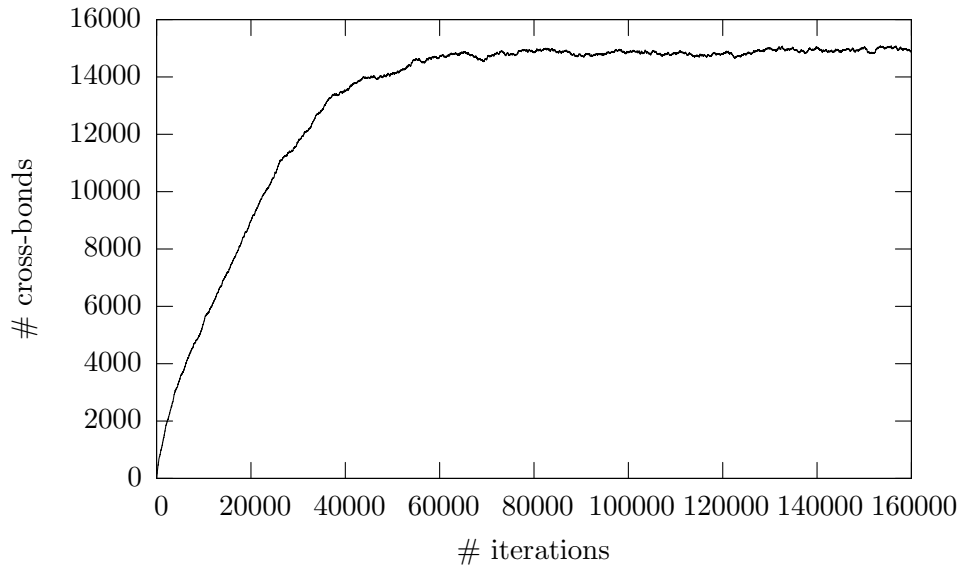
Figure 7.2: Number of cross-bonds vs. Monte Carlo time. Plot (a) corresponds to the data from the first row of table 7.3, (b) to the second row. Note that the tuned algorithm takes more time until thermal equilibrium is reached.

$t_1, t_2, t_3, t_4$	% $L^d$	% $L^d - 1$	% $L^d - 2$	% $L^d - 3$	% $L^d - 4$	# const.	$\xi$	$\Delta\xi$
1	0	0	0	$\sim 0.005$	$\sim 99.995$	$\sim 10^5$	-	-
0.00005	25	35	25	11	4	1.5	1.93	0.02

Table 7.4: *The effect of different values of  $t$  for the algorithm with five allowed configuration space regions. The first row is the data for the untuned algorithm with  $t_i = 1, i = 1, 2, 3, 4$ . Since we have no configurations with the maximal number of clusters, we can not measure any observables at all. In the second row, the  $t_i$  were set to a value which resulted in a more equal distribution of the five different configurations. Columns two to six give the percentage of the configurations, and the seventh column lists the average number of consecutive configurations with constant  $n_c$ . In principle, the  $t_i$  could be adjusted such that the five percentages would each be 20%.*



(a)



(b)

Figure 7.3: Number of cross-bonds vs. Monte Carlo time for the algorithm with five allowed configuration space regions. Plot (a) corresponds to the data from the first row of table 7.4, (b) to the second row. Again, the well-tuned algorithm takes more time until thermal equilibrium is reached. Compared to the algorithm with only two allowed sectors (figure 7.2), thermalization arises faster. However, we have less configurations with  $n_c = L^d$  where observables can be measured.

## Chapter 8

# Conclusion and Outlook

While aiming for large  $N$ , we encountered two main problems: the t'Hooft limit and the low efficiency of the algorithm. In the following concluding remarks, we summarize these problems and propose possible solutions.

If a thorough theoretical study would show that the only way to satisfy the condition would be to increase either  $J$  and  $M$  or  $n$ , it might not be possible to simulate at  $N = \infty$ . In this case, the value of  $N$  is limited by the amount of computational time required for the simulations: We have seen that  $M \propto N$ , and the larger the lattice, the larger the computational effort. However, a possible loophole might be the utilization of a continuous time algorithm similar to the one presented in [9]. It is possible to formulate the algorithm for continuous  $\beta$  (which corresponds to  $M \rightarrow \infty$  and  $\epsilon \rightarrow 0$ ) and therefore the t'Hooft limit could be carried out properly. However, at the time being, this is only speculation and a possible matter of future studies.

Even if we disregard the t'Hooft condition, a simulation at  $N = \infty$  is not free of troubles. We have to find the right balance between the thermalization speed (i.e. how long it takes the system to reach thermal equilibrium) and the number of configurations generated with the maximum number of clusters  $n_c = L^d$ . Observables can only be measured if  $n_c = L^d$ . Our aim is to have a high thermalization speed, many configurations with  $n_c = L^d$  but also a frequent back-and-forth switching between configurations with different  $n_c$ . These quantities can be controlled either by the number of allowed configurations or by the tuning parameters  $t_i$  which influence the relative frequency of occurrence of the corresponding configuration. Unfortunately, for the simulations performed in chapter 7, fast thermalization always means few configurations with  $n_c = L^d$  and vice versa. Moreover, we saw that the autocorrelation time after thermalization is very large. Maybe we could get rid of that by allowing a large number of configurations with different  $n_c$  to be sampled. But even if the efficiency of the algorithm would increase, we would inevitably have less of the desired configurations with  $n_c = L^d$  in the Markov chain.

As an overall conclusion, we can state that an increase of the computational effort as  $N$  goes to infinity is unavoidable. The increase is proportional to  $N$ , which is the same scaling as in an algorithm using the standard Wilson formalism. Still, for moderate  $N$ , it is probably the most efficient algorithm for  $CP(N-1)$  models we can presently think of.



# Appendix A

## Proof of the $SU(N)$ -invariance of $H$

### A.1 Proof of $[H, T^j] = 0$

The result of this section will be needed later to prove that the Hamiltonian for a ferromagnet is invariant under global  $SU(N)$  transformations. Recall that (equation (3.2))

$$H = -J \sum_{x,i} T_x^j T_{x+i}^j. \quad (\text{A.1})$$

The magnetization  $T^j$  is defined as the sum of all spins:

$$T^j = \sum_x T_x^j. \quad (\text{A.2})$$

The matrices  $T$  obey the following commutation relation (equation (3.1)):

$$[T_x^j, T_y^k] = i\delta_{xy} f_{jkl} T_x^l. \quad (\text{A.3})$$

Thus we have

$$\begin{aligned} [H, T^j] &= \left[ -J \sum_{x,i} T_x^k T_{x+i}^k, \sum_y T_y^j \right] \\ &= \sum_{x,y,i} \left[ -J T_x^k T_{x+i}^k, T_y^j \right] \\ &= -J \sum_{x,y,i} \left\{ T_x^k [T_{x+i}^k, T_y^j] + [T_x^k, T_y^j] T_{x+i}^k \right\} \\ &= -iJ \sum_{x,y,i} \left\{ T_x^k (\delta_{(x+i)y} f_{kjl}) T_{x+i}^l + (\delta_{xy} f_{kjl}) T_x^l T_{x+i}^k \right\} \\ &= -iJ \sum_{x,i} \left\{ f_{kjl} T_x^k T_{x+i}^l + f_{kjl} T_x^l T_{x+i}^k \right\} \\ &= -iJ \sum_{x,i} \left\{ f_{kjl} T_x^k T_{x+i}^l + f_{ljk} T_x^k T_{x+i}^l \right\} \end{aligned} \quad (\text{A.4})$$



$$\begin{aligned}
&= -iJ \sum_{x,i} T_x^k T_{x+i}^l (f_{kjl} + f_{ljk}) \\
&= 0,
\end{aligned}$$

since the structure constant  $f$  is completely antisymmetric in  $j, k, l$ .

## A.2 Proof of $H' = H$

A global  $SU(N)$  transformation acts on  $H$  like

$$H' = UHU^\dagger, \quad (\text{A.5})$$

with

$$U = \prod_x U_x = \prod_x \exp(i\alpha_j T_x^j) = \exp\left(i\alpha_j \sum_x T_x^j\right) = \exp(i\alpha_j T^j). \quad (\text{A.6})$$

The transformation matrix  $U$  is unitary:

$$U^\dagger = U^{-1} = \exp(-i\alpha_j T^j). \quad (\text{A.7})$$

We also know that

$$[(T^j)^k, H] = 0, \quad (\text{A.8})$$

for every  $k \in \mathbb{N}$ , which follows directly from the first section of this appendix. Then we have

$$\begin{aligned}
H' &= UHU^\dagger \\
&= UH \exp(-i\alpha_j T^j) \\
&= UH(1 - i\alpha_j T^j \pm \dots) \\
&= U(1 - i\alpha_j T^j \pm \dots)H \\
&= UU^\dagger H \\
&= H.
\end{aligned} \quad (\text{A.9})$$

Hence,  $H$  is indeed invariant under global  $SU(N)$  transformations.

# Acknowledgments

A lot of people have contributed to this work in many different ways. Despite his duties as head of the institute and his numerous diploma and PhD students, Prof. Uwe-Jens Wiese always managed to devote a lot of time for discussions on this thesis. Moreover, I would like to thank him for a very careful proofreading. Further thanks goes to Christine Bolliger, Matthias Nyfeler, Stéphane Riederer, Michele Pepe and Stefan Dürr for their much valued assistance. I have also benefited from numerous discussions with my colleagues in the neighbouring offices, especially Christoph Weiermann, Stefan Lanz and Lorenzo Mercolli. Certainly, those challenging iMaze battles fought in the digital realms of the institute shall not be forgotten. In case of emergency in computational matters, I truly appreciated the helpfulness of the system administrators. I am also very much indebted to Ottilia Hänni and Ruth Bestgen for taking care of administrative matters. Furthermore, I would like to thank Dr. Ferenc Niedermayer for always being there when I needed help in statistical thermodynamics. Another big thank you goes to my parents for their continuous support.



# Bibliography

- [1] A. D’Adda, P. Di Vecchia and M. Lüscher: A  $1/n$  Expandable Series of Non-Linear Sigma Models with Instantons. *Nucl. Phys. B146 (1978) 63*; *Nucl. Phys. B152 (1979) 125*.
- [2] S. Caracciolo et al.: Wolff-Type Embedding Algorithms for General Nonlinear  $\sigma$ -Models. *Nucl. Phys. B403 (1993) 475*.
- [3] M. Hasenbusch and S. Meyer: Multigrid Acceleration for Asymptotically Free Theories. *Phys. Rev. Lett. 68 (1992) 435*.
- [4] B. B. Beard, M. Pepe, S. Riederer and U. J. Wiese: Efficient Cluster Algorithm for  $CP(N - 1)$  Models. *Comput. Phys. Commun. 175 (2006) 629* [*arXiv:hep-lat/0602018*].
- [5] B. B. Beard, M. Pepe, S. Riederer and U. J. Wiese: Study of  $CP(N-1)$  Theta-Vacua by Cluster-Simulation of  $SU(N)$  Quantum Spin Ladders. *Phys. Rev. Lett. 94 (2005) 010603* [*arXiv:hep-lat/0406040*].
- [6] S. Riederer: D-Theory Formulation of Quantum Field Theories and Application to  $CP(N - 1)$  Models. *Ph.D. thesis*. 2006.
- [7] U.-J. Wiese: Quantum Spins and Quantum Links: The D-Theory Approach to Field Theory. *Reprint from Progress of Theoretical Physics, Supplement No. 131*. 1998.
- [8] U.-J. Wiese: D-Theory: A Quest for Nature’s Regularization. *Nuclear Physics B (Proc. Suppl.) 153 (2006) 336347*.
- [9] B. B. Beard and U.-J. Wiese: Simulations of Discrete Quantum Systems in Continuous Euclidean Time. *Nucl. Phys. Proc. Suppl. 53 838 (1997)* [*cond-mat/9602164*].