ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Roman Vetter

The Worm Algorithm for the $\mathbb{C}\mathrm{P}^{N-1}$ Model

Semester Thesis

Institute for Theoretical Physics (ITP)
Swiss Federal Institute of Technology (ETH) Zurich

Supervisor

Dr. Philippe de Forcrand

February 10, 2011

Abstract

The problem of critical slowing down is reviewed and a comprehensive description of how to measure the efficiency of Monte Carlo algorithms in practice is given. Special regard is paid to the O(N) and $\mathbb{C}\mathrm{P}^{N-1}$ σ models. The dynamic critical exponents of Wolff's single-cluster algorithm are determined on the two-dimensional O(3) and $\mathbb{C}\mathrm{P}^3$ models to confirm its efficiency on the former and inefficiency on the latter. The idea of worm algorithms is explained on the exemplary Ising model and the analogous formalism based on the high-temperature expansion by Chandrasekharan is derived for the $\mathbb{C}\mathrm{P}^{N-1}$ model with general N. An efficient computer implementation is provided in detail, and the algorithm is verified. Consistency with existing numerical results is reported for $\mathbb{C}\mathrm{P}^1$ in two dimensions to very high precision, together with a dynamic critical exponent of $z = 0.32(3)$ for winding numbers and $z \approx 0$ for energy and magnetic susceptibility. The algorithm is found to lack ergodicity for $N > 2$, and the problem is quantified numerically. An effort to include the disconnected piece of the $\mathbb{C}\mathrm{P}^{N-1}$ Green's function into the sampling scheme results in a sign problem.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Motivation

The lattice approach to quantum field theory (QFT) is accompanied by the inherent need for efficient Monte Carlo (MC) algorithms for near-continuum simulations. On the journey of finding such for various problems in QFT, the (1+1)-dimensional O(N) invariant non-linear σ models and $\mathbb{C}P^{N-1}$ models (chapter 2) serve as suitable toy models. They share a set of fundamental properties with quantum chromodynamics ((3+1)-dimensional Yang-Mills theory), such as asymptotic freedom (chapters 2 & 4), or the presence of instanton solutions ($\mathbb{C}P^{N-1}$ for all N) [1]. The $\mathbb{C}P^{N-1}$ model is subject to particular prominence also due to its equivalence to N-flavor quantum electrodynamics (QED) [2].

Cluster algorithms such as the one by Wolff efface the problem of critical slowing down (CSD, chapter 4) in the 2D O(N) model nicely [3–6], but unfortunately, it has been argued in a no-go theorem [7] that a generalization of Wolff-type embedding to models where an MC update has a fixed-point manifold with codimension higher than one, which would be a first step towards lattice gauge theories, won't be efficient. Indeed, Wolff's algorithm isn't on the $\mathbb{C}\mathrm{P}^{N-1}$ model (codimension 2).

In 1992, Hasenbusch and Meyer [8] proposed a multigrid MC algorithm for the two-dimensional $\mathbb{C}\mathrm{P}^3$ model which (almost) eliminates CSD. More recently, using the ideas of dimensional reduction from D-theory, Beard, Pepe, Riederer and Wiese achieved to construct a cluster algorithm for the $\mathbb{C}\mathrm{P}^{N-1}$ model that doesn't suffer from CSD at all [9,10], at the painful cost of one additional lattice dimension however. The (somewhat surprisingly) successful approach of worm algorithms by Prokof'ev and Svistunov [11] on ϕ^4 theory and a variety of universality classes during the last few years gives rise to hope that the worm algorithm might also accelerate the $\mathbb{C}\mathrm{P}^{N-1}$ model. Yet, the worm algorithm in Prokof'ev and Svistunov's formalism has never been outlined for the latter. It is the major purpose of the present thesis to fill this gap.

However, just during the writing of this work, Wolff published an adaption of his loop algorithm for the O(N) model [12], which is essentially a generalized worm algorithm with an alternative implementation approach, to the $\mathbb{C}\mathrm{P}^{N-1}$ model [13], showing that critical slowing down can indeed be completely defeated.

1.2 Preliminary Comments

We restrict ourselves to the simple square lattice Λ with periodic boundary conditions, i.e. in the following the lattice dimensionality is always $D = 2$. For convenience we use equal lattice extents L in all dimensions, hence the spacetime volume is given by $V = |\Lambda| = L^2$. Furthermore we drop nondimensionalizations like L/a (where a is the lattice spacing) and implicitly understand L and related quantities in dimensionless lattice units. As is customary in lattice theories, site indices $x \in \Lambda$ are implicitly understood as vectors $x \in \{0, 1, ..., L-1\}^D$, or *global* indices taking scalar integral values $x \in \{0, 1, ..., L^D - 1\}$, whichever makes sense in the context.

Where not stated differently, we will give the *standard error of the mean* $\sigma/\sqrt{n}$, where σ is the sample standard deviation from n measurements (i.e. from n independent simulations), as statistical error bounds in brackets for any numerical results. This corresponds to an approximate 68% confidence level (or slightly lower for small n, where the Student's t-distribution is to be considered).

In some cases we will provide the *reduced chi-square statistic* χ^2_{red} as a measure for the goodness of a fit, defined on a set of k data points $\{y_i\}_{1 \le i \le k}$, each with standard error Δy_i and a fitted value f_i, as

$$\chi^2_{\text{red}} = \frac{1}{k-p} \sum_{i=1}^{k} \left(\frac{y_i - f_i}{\Delta y_i} \right)^2 , \tag{1.1}$$

where p denotes the number of fit parameters (and hence $k - p$ is the number of degrees of freedom). $\chi^2_{\text{red}} > 1$ indicates a poor fitting model (or underestimated errors), whereas $\chi^2_{\text{red}} < 1$ is an indicator for over-fitting.

Be aware that a number of different quantities are identified by the same symbols in the following. This is simply for consistency and is considered to enhance readability. We denote them the way they are commonly denoted by in literature. Along many other examples, the topological charge is represented by Q, which also stands for the site integral in the worm algorithm. Such ambiguities are especially numerous with the two models at hand, where we denote both actions by S, the couplings by β, etc. These are of course not identical. Where we don't explicitly repeat the meaning of certain variables, it should be clear within the given context.

1.3 Program Implementation

All programs in this thesis were implemented in C++, with some postprocessing in MATLAB. We used the `lagged_fibonacci1279` pseudo-random number generator from the Boost C++ libraries, which provides a good balance between sufficient performance and high quality, for all our simulations. To avoid numerical trouble, our programs work with double IEEE 754 floating point precision (64-bit), and even quadruple precision (128-bit) for averaging and statistical evaluation.

2 The Spin Models

In this chapter we define the spin models and their statistical properties that were studied for this thesis.

2.1 The O(N) Model

The non-linear lattice O(N) σ model (often just called O(N) *model* or N*-vector model*) was first proposed by H. E. Stanley in 1968 [14, 15]. When discretized on a lattice, it consists of a classical vector-valued spin field z with N real-valued components and unit Euclidean length, i.e.

$$z(x) \in \mathbb{R}^N, \quad N \in \mathbb{N}_0, \quad \|z(x)\|_2 = 1 \quad \forall x \in \Lambda = \{0, 1, ..., L-1\}^D. \tag{2.1}$$

In the absence of a magnetic field, the lattice action usually reads

$$S = -\beta \sum_{\langle x,y \rangle} z(x) \cdot z(y) \tag{2.2}$$

with inverse temperature (or *coupling*) β, where the sum runs over all nearest-neighbor coordinate pairs x, y. The name of the model origins from its invariance under global O(N) transformations. It's also worth mentioning that the O(N) model has special names for the lowest values of N:

- $N = 0$: self-avoiding walk [16]
- $N = 1$: Ising model
- $N = 2$: XY model
- $N = 3$: Heisenberg model

In two spacetime dimensions, the Ising model has a second-order phase transition at $\beta_c = \log(1 + \sqrt{2})/2$ [17], while the XY model features a Kosterlitz-Thouless transition at $\beta_c \approx 1.1199$ [18, 19]. For $N \geq 3$ the O(N) model is asymptotically free [20–25], i.e. exhibits no phase transition (cf. section 4.3.1).

The Green's function (or *two-point function*, *spin-spin correlator*) is given by

$$G(x, y) = \langle z(x) \cdot z(y) \rangle. \tag{2.3}$$

Among common statistical quantities of interest are the energy

$$\langle E \rangle = -\sum_{\langle x,y \rangle} G(x, y) = -\sum_{\langle x,y \rangle} \langle z(x) \cdot z(y) \rangle, \tag{2.4}$$

and the absolute magnetization

$$\langle |M| \rangle = \Big\langle \Big\| \sum_x z(x) \Big\|_2 \Big\rangle, \tag{2.5}$$

and using the fluctuation dissipation theorem one finds the heat capacity

$$\langle C_V \rangle = \beta^2 \Big(\langle E^2 \rangle - \langle E \rangle^2 \Big), \tag{2.6}$$

the (disconnected) magnetic susceptibility

$$\langle \chi \rangle = \frac{1}{V} \langle M^2 \rangle = \frac{1}{V} \Big\langle \Big(\sum_x z(x) \Big)^2 \Big\rangle = \frac{1}{V} \sum_{x,y} G(x,y), \tag{2.7}$$

the connected magnetic susceptibility

$$\langle \chi_c \rangle = \frac{1}{V} \Big(\langle M^2 \rangle - \langle M \rangle^2 \Big) = \langle \chi \rangle \tag{2.8}$$

in a symmetric phase ($\langle M \rangle = 0$) and

$$\langle \chi_c \rangle = \frac{1}{V} \Big(\langle M^2 \rangle - \langle |M| \rangle^2 \Big) \tag{2.9}$$

in a broken phase ($\langle M \rangle \neq 0$).

The Heisenberg (O(3)) model is topologically non-trivial. One defines the topological susceptibility as [26, 27]

$$\langle \chi_t \rangle = \frac{1}{V} \langle Q^2 \rangle \tag{2.10}$$

where

$$Q = \sum_x q(x) \in \mathbb{Z} \tag{2.11}$$

is the topological charge on a periodic lattice with charge density q, that, for the Heisenberg model, efficiently calculates as [28, 29]

$$\begin{aligned} q(x) = {} & \delta_{\operatorname{sign}|z_1\ z_2\ z_3| = \operatorname{sign}|z_1\ z_2\ p| = \operatorname{sign}|z_2\ z_3\ p| = \operatorname{sign}|z_3\ z_1\ p|} \ \operatorname{sign}|z_1\ z_2\ z_3| \\ & + \delta_{\operatorname{sign}|z_1\ z_3\ z_4| = \operatorname{sign}|z_1\ z_3\ p| = \operatorname{sign}|z_3\ z_4\ p| = \operatorname{sign}|z_4\ z_1\ p|} \ \operatorname{sign}|z_1\ z_3\ z_4| \end{aligned} \tag{2.12}$$

for any arbitrary constant reference point p on the 2-sphere. $\operatorname{sign}|a\ b\ c| \in \{-1, 1\}$ stands for the signum of the determinant of the matrix defined column-wise by the vectors a, b, c. This amounts to counting the number of times p lies within one of the two oriented spherical triangles spanned by the four sites $z_1, ..., z_4$ that generate the plaquette at site x like depicted in fig. 2.1. An alternative, less efficient, but equivalent way of calculating $q(x)$ is summing the areas of those two spherical triangles, see refs. [26, 27]. Unfortunately, short-range fluctuations of Q dominate the field contribution on lattices, which makes numerical results for Q and χ_t inconsistent with perturbation theory [26, 30, 31].

Note that the above observables (2.4) to (2.11) are *extensive* properties. In order to obtain meaningful values, and in particular if varying spacetime volumes V or even the continuum limit are studied, they are usually divided by V to become *intensive* (or *specific*).

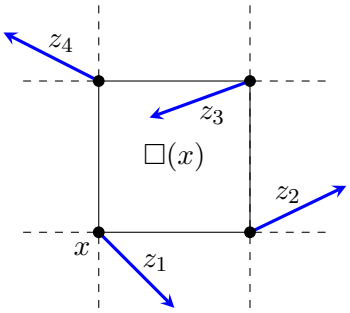


Figure 2.1: Illustration of the plaquette $\square(x)$ at site x and spin vectors $z_1 = z(x), ..., z_4$ at the corner sites.

2.2 The $\mathbb{C}\mathrm{P}^{N-1}$ Model

The non-linear $\mathbb{C}\mathrm{P}^{N-1}$ σ model ($\mathbb{C}\mathrm{P}^{N-1}$ *model* in short) is a direct generalization of the $\mathrm{O}(N)$ model to complex fields. It was first defined by H. Eichenherr in 1978 [1]. In close analogy to the $\mathrm{O}(N)$ model we formulate it on the lattice in terms of a vector field z with N complex-valued components and unit length, i.e. [32]

$$z(x) \in \mathbb{C}^N, \quad 2 \leq N \in \mathbb{N}, \quad |z(x)|^2 = \sum_{a=1}^{N} |z_a(x)|^2 = 1 \quad \forall x \in \Lambda = \{0, 1, ..., L-1\}^D. \tag{2.13}$$

There exist various lattice actions with different properties (such as gauge invariance for instance) that reproduce the $\mathbb{C}\mathrm{P}^{N-1}$ model in the continuum limit. In this work we use the standard *quartic action*

$$\begin{aligned} S &= -\beta \sum_{\langle x,y \rangle} |\bar{z}(x) \cdot z(y)|^2 = -\beta \sum_{\langle x,y \rangle} \Big| \sum_{a=1}^{N} \bar{z}_a(x)\, z_a(y) \Big|^2 \\ &= -\beta \sum_{\langle x,y \rangle} \sum_{a,b=1}^{N} \big(\bar{z}_a(x)\, z_b(x)\big)\big(\bar{z}_b(y)\, z_a(y)\big) \end{aligned} \tag{2.14}$$

with coupling β, where the sum is again taken over all nearest-neighbor coordinate pairs x, y, and the bar denotes complex conjugation. (Some literature denote the coupling by 2κ and define a different $\beta = 2\kappa/N$, see e.g. ref. [33].) This action is invariant under global $\mathrm{SU}(N)$ transformations and $\mathrm{U}(1)$ gauge transformations. It is well known that the $\mathbb{C}\mathrm{P}^1$ model is equivalent to the $\mathrm{O}(3)$ model. (We will numerically confirm this in chapters 4 and 5.) The convention to write $\mathbb{C}\mathrm{P}^{N-1}$ instead of, say, $\mathbb{C}\mathrm{P}(N)$ originates from the observation that one N-component spin vector z parametrizes a point in the complex projective space $\mathbb{C}\mathrm{P}^{N-1}$. All $\mathbb{C}\mathrm{P}^{N-1}$ models are asymptotically free, i.e. exhibit no phase transition (see however ref. [34]). More on this in section 4.4.1.

The connected two-point function takes the form [33, 35]

$$G_c(x,y) = \big\langle |\bar{z}(x)\cdot z(y)|^2 \big\rangle_c = \big\langle |\bar{z}(x)\cdot z(y)|^2 \big\rangle - \frac{1}{N} = G(x,y) - \frac{1}{N}. \tag{2.15}$$

Among common statistical quantities of interest are the energy

$$\langle E\rangle = -\sum_{\langle x,y\rangle} G(x,y) = -\sum_{\langle x,y\rangle} \big\langle \big|\bar{z}(x)\cdot z(y)\big|^2 \big\rangle, \tag{2.16}$$

the absolute magnetization

$$\langle |M|\rangle = \Big\langle \Big(\sum_{a=1}^{N} \Big| \sum_x z_a(x) \Big|^2 \Big)^{1/2} \Big\rangle, \tag{2.17}$$

the magnetic susceptibility

$$\langle \chi\rangle = \frac{1}{V}\sum_{x,y} G_c(x,y) = \frac{1}{V}\Big\langle \sum_{a,b=1}^{N} \Big| \sum_x \bar{z}_a(x)\, z_b(x) \Big|^2 \Big\rangle - \frac{V}{N}, \tag{2.18}$$

and the heat capacity C_V given by (2.6).

The $\mathbb{C}\mathrm{P}^{N-1}$ model is topologically non-trivial for all N [1, 32]. The topological susceptibility and charge are defined like in (2.10) and (2.11), respectively. The charge density $q(x)$, on the other hand, is determined from the U(1) parallel transporters [33]

$$U(x,\mu) = \frac{\bar{z}(x)\, z(x+\hat{\mu})}{|\bar{z}(x)\, z(x+\hat{\mu})|} = \exp\big(i\phi(x,\hat{\mu})\big) \in \mathrm{U}(1), \tag{2.19}$$

namely as the sum of the arguments ϕ of the four parallel transporters along the oriented boundary of the plaquette $\square(x)$ at site x, shifted by an integer multiple $k(x)$ of 2π to the standard $]-\pi,\pi]$ interval:

$$2\pi q(x) = \sum_{(y,\,\mu)\,\in\,\partial\square(x)} \phi(y,\hat{\mu}) + 2\pi k(x) \;\in\,]-\pi,\pi] \tag{2.20}$$

For a drawing of the plaquette $\square(x)$, see fig. 2.1. For the very same reason as for the O(3) model, lattice results for Q and χ_t at $N \leq 3$ are typically inconsistent with perturbation theory. Luckily, however, this doesn't hold for $\mathbb{C}\mathrm{P}^3$ and above [31].

3 Efficiency of Monte Carlo Algorithms

Beyond Onsager's solution [17] of the simple two-dimensional Ising model, no exact (non-perturbative) analytic solution for either of the above models is known. They are therefore studied numerically, using Monte Carlo simulations. The most elementary importance sampling MC algorithm for such systems is the popular single-spin-flip Metropolis algorithm [36].

The major disadvantage of the single-spin-flip Metropolis algorithm is the dramatic loss of efficiency at second-order phase transitions (or in the "critical" $\beta \to \infty$ limit for O(N), $N \geq 3$, and $\mathbb{C}\mathrm{P}^{N-1}$), known as *critical slowing down*, see section 3.3 below. This handicap limits its application to the uninteresting noncritical regimes of the phase diagrams, leaving it unsuitable for calculations near the continuum limit. Sections 3.1 to 3.3 give an introduction to the characteristics of this problem.

Much effort has been put into the development of more efficient algorithms to overcome or reduce critical slowing down. In chapter 4 we discuss Wolff's cluster algorithm as an example of non-local algorithms, and chapter 5 is on the more recently developed local, yet in many cases efficient classical worm algorithm.

3.1 Autocorrelation Times

The most common MC algorithms sample the configuration space by modifying the current configuration of the system to obtain a new one. Two subsequent samples are thus not statistically independent; they are *correlated*. The autocorrelation time τ_A of any state function A (energy, magnetization, etc.) is a measure of how many updates a MC algorithm typically needs to perform, until a configuration is created, such that the correlation between the current value for A and the last measurement has decayed significantly. It is important to stress that τ_A depends on the choice of A, the model, the MC algorithm and D.
Obviously, as we are interested in approximating $\langle A \rangle$ by the arithmetic mean of a series of (correctly sampled) measurements $\{A_s\}_{1 \leq s \leq T}$,

$$\langle A \rangle \approx \bar{A} = \frac{1}{T} \sum_{s=1}^{T} A_s, \tag{3.1}$$

we would like τ_A to be as small as possible, such that we obtain reasonably accurate results from decorrelated A_t within short simulation time.

The (unnormalized) *time-displaced autocorrelation function* is defined as [37,38]

$$C_{AA}(t) = \big\langle \big(A_s - \langle A\rangle\big)\big(A_{s+t} - \langle A\rangle\big)\big\rangle = \langle A_s\, A_{s+t}\rangle - \langle A\rangle^2 \tag{3.2}$$

where we are averaging over all times s. For convenience, it is usually normalized to

$$\rho_{AA}(t) = C_{AA}(t)/C_{AA}(0). \tag{3.3}$$

The typical observation is that ρ_{AA} decays exponentially at large time separation t;

$$\rho_{AA}(t) \approx \exp(-|t|/\tau_A), \quad |t| \gg 1. \tag{3.4}$$

(For a more rigorous discussion of the characteristics of ρ_{AA} involving Markov matrices, see ref. [38].) This defines the *exponential autocorrelation time* [37]

$$\tau_A^{\exp} = \limsup_{t\to\infty} \frac{-t}{\log|\rho_{AA}(t)|}. \tag{3.5}$$

One could then in principle fit the measured $\rho_{AA}(t)$ by an exponential and use its exponent as an estimate for τ_A. However, in the large-time regime where the exponential rate of decay is closest to τ_A, ρ_{AA} is usually completely dominated by noise, as depicted in fig. 3.1. This limitation on the fitting range renders the fitting approach unsuitable for an automated numerical implementation. Moreover (and most importantly), $\tau_A^{\exp}$ describes the relaxation of A towards equilibrium, which is in general not the same as the measure of autocorrelation *in* equilibrium we are looking for [37,39]. We therefore *integrate* rather than *fit* the autocorrelation function, exploiting the identity

$$\int_0^\infty \exp(-t/\tau)\,\mathrm{d}t = \tau. \tag{3.6}$$

Assuming that (3.4) is a good approximation for all times t, we define the *integrated autocorrelation time*

$$\tau_A^{\mathrm{int}} = \int_0^\infty \rho_{AA}(t)\,\mathrm{d}t \approx \tau_A. \tag{3.7}$$

In a practical implementation, τ_A^{int} is obtained as follows: From a series of measurements $\{A_s\}_{1\le s\le T}$ taken at regularly spaced MC times s, calculate the autocorrelation function as [38]

$$C_{AA}(t) = \frac{1}{T-t}\sum_{s=1}^{T-t}\big(A_s - \bar{A}\big)\big(A_{s+t} - \bar{A}\big), \quad 0 \le t \le T-1, \tag{3.8}$$

and normalize to obtain $\rho_{AA}(t)$. Then, integrate by summing the first M values, to obtain a trucated estimate:

$$\tilde{\tau}_A^{\mathrm{int}}(M) = \frac{1}{2} + \sum_{t=1}^{M}\rho_{AA}(t). \tag{3.9}$$

Finally, calculate and add the truncated tail ($t \in [M,\infty[$) using the estimator $\tilde{\tau}_A^{\mathrm{int}}$ by setting

$$\tau_A^{\mathrm{int}} = \tilde{\tau}_A^{\mathrm{int}}(M)\Big(1 + \exp\big(-M/\tilde{\tau}_A^{\mathrm{int}}(M)\big)\Big). \tag{3.10}$$

The only question remaining is how to decide at which M to truncate the summation. This is a tradeoff between bias and variance; the authors of ref. [37] have found that choosing M minimal under the contraint $M \geq c\tau_A^{\text{int}}(M)$ is convenient. We use $c = 3$ for all computations in the following.

Obviously, to obtain a reliable estimate for the integrated autocorrelation time, $\rho_{AA}(t)$ needs to be resolved at time steps well below τ_A, and at the same time, many samples ($T \approx 1000\tau_A$ or even more [37]) are required to obtain a clear exponential. This sampling window evidently strongly depends on τ_A, the (a-priori unknown) quantity which we want to determine with it in the first place. Calculating τ_A^{int} can therefore be tedious in practice. A good initial guess typically proves quite worthy.

Note that these calculations give the integrated autocorrelation time *in units of measurements*. Usually one would like to know it on a meaningful MC time scale such as in units of lattice sweeps. That is, updating each lattice degree of freedom (each spin vector in our case) once on average. This is easy to achieve: simply multiply τ_A^{int} by the average number of spin flips between measurements and divide by the spacetime volume V.

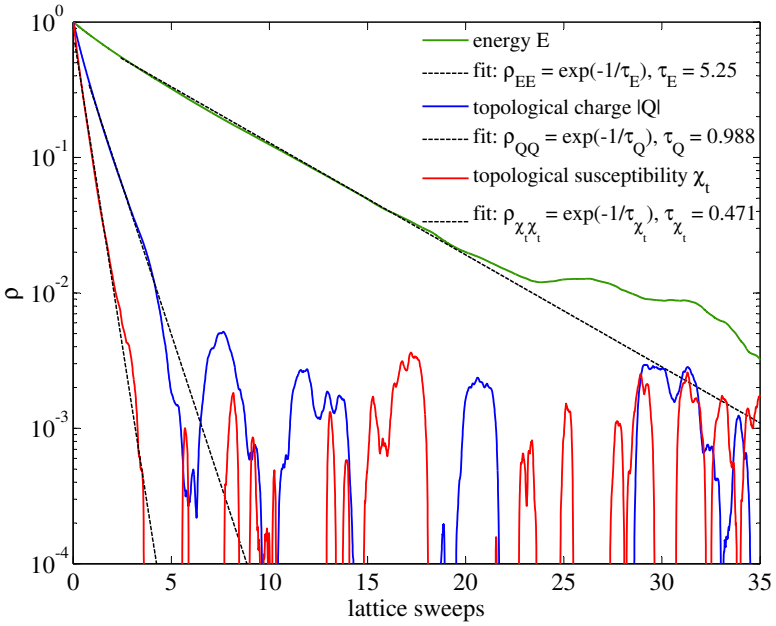


Figure 3.1: Examples of the normalized autocorrelation function on a semi-logarithmic plot, for the O(3) model simulated on a $L = 128$ square grid at $\beta = 1.5$ with Wolff's cluster algorithm. After each cluster update, a measurement of the drawn three modes was performed, with a total of 2×10^7 updates. Clearly, the data for the topological charge, for instance, is dominated by noise after roughly 4 sweeps. The slowest mode is the energy. The autocorrelation times are given in units of sweeps and were obtained by fitting a straight line in the appropriate region.

3.2 The Correlation Length

The typical distance on the lattice at which spins decorrelate is measured by the correlation length ξ. It is a state function describing the average size of clusters of spins with "similar" orientation. In that sense, it may be seen as a measure of the order of the system. As we will see in section 3.3, ξ is a key component of the definition of the efficiency of a MC algorithm. How can it be calculated?

3.2.1 The Classical Way

Let us consider the two-point function of the $\mathbb{C}\mathrm{P}^{N-1}$ model along lattice directions $\mu = 1, ..., D$,

$$G_c(r) = \left\langle |\bar{z}(0) \cdot z(r)|^2 \right\rangle_c = \frac{1}{D} \sum_{\mu=1}^{D} \left(\frac{1}{V} \sum_x \left\langle |\bar{z}(x) \cdot z(x + r\hat{\mu})|^2 \right\rangle \right) - \frac{1}{N}, \tag{3.11}$$

where we used the usual notation of $\hat{\mu}$ being the μth canonical unit vector in D dimensions. At large separations r ($1 \ll r \approx L/2$, $L \to \infty$) it decays as

$$G_c(r) \sim r^{-p} \exp(-r/\xi). \tag{3.12}$$

Here it becomes evident why ξ is called *correlation length*. It is the expected distance at which the two-point correlator decreases roughly by a factor of 1/e. On a periodic lattice, $G_c(r)$ is L-periodic and symmetric: $G_c(r) = G_c(L-r)$, $r = 0, 1, ..., L$. To obtain an estimate for $\xi = \xi(L)$, one can hence measure $G_c(r)$ using (3.11) at a complexity of $\mathcal{O}(DL^{D+1}N)$ for each sample, and then fit (3.12) from the left and right to the data. Figure 3.2 shows an example of such a fit. However, this is only precise at large $r \approx L/2$, and the power law correction doesn't help to extract a clear signal for ξ. Similarly, ξ may be extracted from the "zero momentum" correlator

$$G_{0,c}(r) = \left\langle |\bar{z}(0) \cdot z(r)|^2 \right\rangle_{0,c} = \frac{1}{D} \sum_{\mu=1}^{D} \left(\frac{L}{V^2} \sum_{x,y} \delta\big(|(y-x)_\mu| - r\big) \left\langle |\bar{z}(x) \cdot z(y)|^2 \right\rangle \right) - \frac{1}{N}, \tag{3.13}$$

which projects the D-dimensional system to only one dimension by averaging the spins in all lattice dimensions other than the μth. The indicator function δ holds $\bar{z}(x)$ and $z(y)$ at a distance r in direction μ. The observed decay is then purely exponential,

$$G_{0,c}(r) \sim \exp(-r/\xi), \tag{3.14}$$

and hence ξ can be determined from fitting the L-periodic MC estimator for $G_{0,c}(r)$ to $C \cosh\big((L/2 - r)/\xi\big)$ without any disruptive power law term. The price to pay is a higher computational cost: $\mathcal{O}(DL^{2D}N)$ per sample for $G_{0,c}$.

For systems with constant physical extent L/ξ (cf. also section 3.3) the Green's functions are expected to collapse after rescaling by a *field renormalization* prefactor $Z = Z(\beta)$ because they describe the same physical system, at different discretization coarseness (i.e. different lattice spacings a). Indeed, we observe an accurate collapse for the $\mathbb{C}\mathrm{P}^3$ model in fig. 3.3, where the data from table 4.2 is plotted. $Z(\beta)$ grows linearly with β in the shown range.

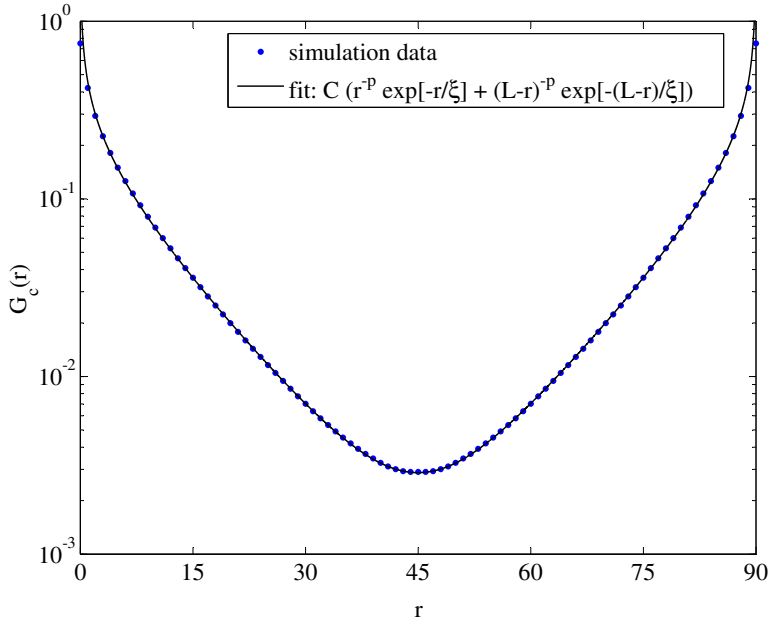
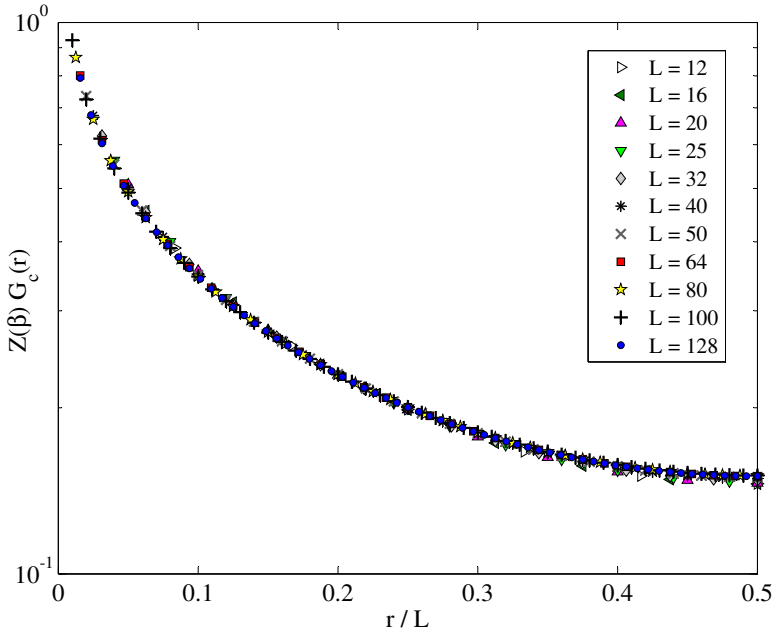
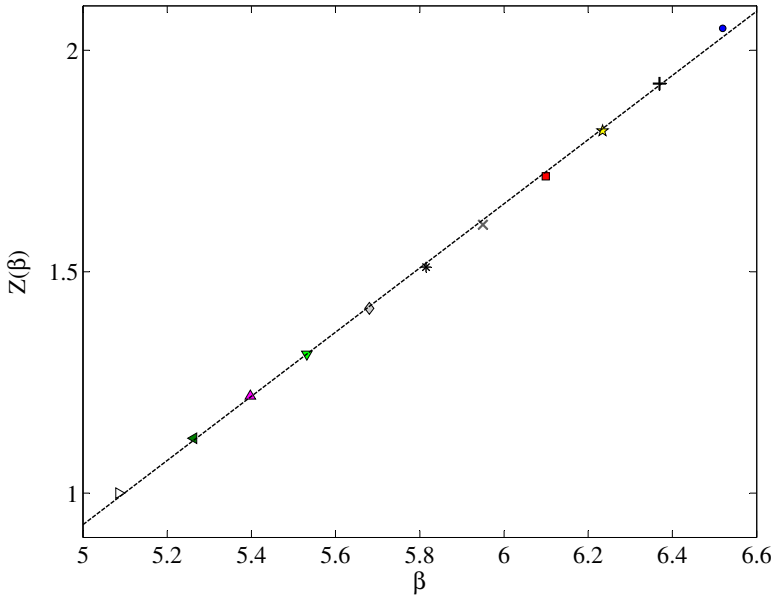


Figure 3.2: Semi-logarithmic plot of the Green's function $G_c(r)$ of the $\mathbb{C}\mathrm{P}^3$ model on a $L = 90$ square lattice at $\beta = 5.5$. The data was calculated from a single simulation using Wolff's cluster algorithm, at the same statistics as Jansen & Wiese (1992) [33]. The resulting correlation length $\xi = 11.12$ is consistent with their prediction of 10.3(9). Clearly, the fit to the form $C(r^{-p}\exp[-r/\xi] + (L-r)^{-p}\exp[-(L-r)/\xi])$ is poor at small separations $r \approx 0$.



(a) Renormalized Green's functions on a semi-log scale. Error bars are smaller than the symbols and were therefore omitted.



(b) The field renormalization Z that was used to collapse the averaged Green's functions in (a) is plotted against β, and a straight line is fitted to it.

Figure 3.3: Collapse of the Green's functions of the $\mathbb{C}\mathrm{P}^3$ model at different lattice spacings. The data from table 4.2 is used. Both vertical axes are given up to a constant prefactor only.

3.2.2 Fitting a Free Massive Boson

In lattice units, the inverse of the correlation length corresponds to the mass (the so-called *mass gap* $m = 1/\xi$) of a freely moving boson propagating the interaction among the sites. This identity can be used to determine ξ more precisely from the full two-point correlator $G(x, y)$. Consider the Euclidean action of a free massive bosonic field ϕ in one spatial dimension

$$S_E = \int \frac{1}{2}\big(\partial_\mu \phi(\tau, x)\big)^2 + \frac{1}{2} m^2 \phi(\tau, x)^2 \, \mathrm{d}\tau \mathrm{d}x, \tag{3.15}$$

with x being the spacial coordinate, and τ imaginary time. On a square lattice, this discretizes to the lattice action

$$S_L = \frac{1}{2} \phi^\dagger M(m) \phi, \tag{3.16}$$

where $\phi = \big(z(0), ..., z(V-1)\big)^T \in \mathbb{C}^V$ is the field vector, and

$$M(m) = (m + 2D)\mathbb{1}_V - \sum_x \hat{x}\, n(x)^T \in \mathbb{R}^{V \times V} \tag{3.17}$$

is the discretized Laplacian matrix with the reduced mass m on the diagonal. Here, $\hat{x}$ is the xth canonical unit vector in V dimensions, and $n(x)$ is the vector containing the nearest neighbors of the site with global index x,

$$n_y(x) = \begin{cases} 1 & \text{if } x,y \text{ are nearest neighbors} \\ 0 & \text{otherwise} \end{cases}, \quad y = 1, ..., V. \tag{3.18}$$

This setup then allows to fit the full translation-invariant two-point correlator $g \in \mathbb{R}^V$, given component-wise by

$$g_y = \langle G(0, y) \rangle = \frac{1}{V} \sum_x G(x, y), \quad y = 1, ..., V, \tag{3.19}$$

to the first column of the inverse of $M(m)$ by

$$g \propto M(m)^{-1} e_1, \tag{3.20}$$

and thereby extracting the correlation length $\xi(L) = 1/m(L)$ from the complete information g gathered during the simulation. Figure 3.4 shows an example of such a fit.

This method has a major drawback compared with the second-moment definition, which we will become acquainted with in the next subsection, for algorithms that directly operate on the spin configuration, such as cluster algorithms. Calculating g there is rather expensive, since one sample has complexity $\mathcal{O}(L^{2D}N)$. However, in chapter 5 we will study the worm algorithm, which directly samples the translation-invariant Green's function, such that g comes almost for free. It is thus most efficient to extract ξ using this method for the worm algorithm.

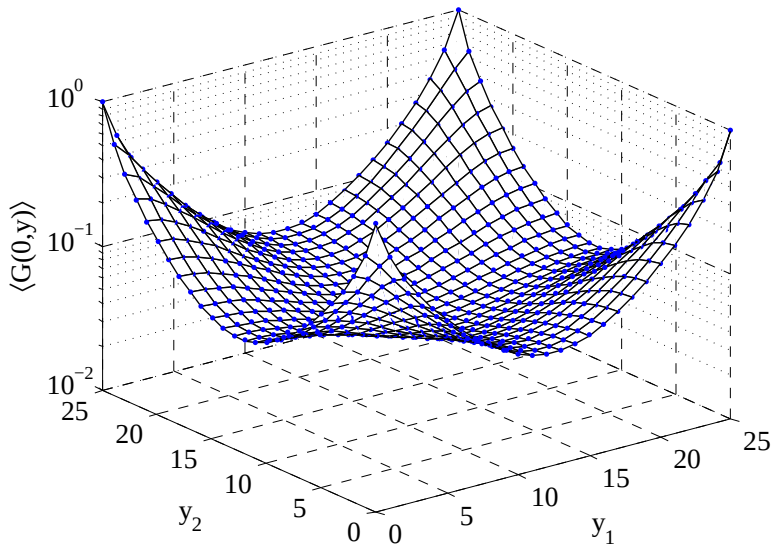


Figure 3.4: The propagator $c\,M(m)^{-1}e_1$ of a free boson (black grid) fitted to the Green's function $g \mathrel{\widehat{=}} \langle G(0,y)\rangle$, $y=(y_1,y_2)$, of the Ising model (blue dots) on a $L=25$ square lattice, calculated from a single simulation at $\beta=0.4$ using the worm algorithm. The fit yields a correlation length of $\xi=1/m=5.85$. The root mean square error of the fit is as small as RMSE $=6.26\times10^{-4}$, thus confirming the high precision of the data.

3.2.3 The Second-Moment Definition

There exists a somewhat more robust way to directly extract the correlation length from the simulated system. Let's identify the Fourier transform of the continuum Green's function at momentum k, $\tilde{G}(k)$, normalized by its value at zero momentum, $\tilde{G}(k=0)=\langle\chi\rangle$, by

$$\frac{\tilde{G}(k)}{\tilde{G}(0)}=\frac{m^2}{m^2+k^2}. \tag{3.21}$$

By inserting the minimal lattice momentum $k_{\min}=2\pi/L$, and discretizing it to the lattice via $a\hat{k}=2\sin(ak/2)$, one defines the *second-moment correlation length* [40–42]

$$\xi_{\text{2nd}}(L)=\frac{1}{2\sin(\pi/L)}\sqrt{\tilde{G}(0)/\tilde{G}(\hat{k}_{\min})-1}. \tag{3.22}$$

Note that sometimes it is defined with linearized sine for large L [43, 44]:

$$\xi^{(2)}_{\text{2nd}}(L)=\frac{L}{2\pi}\sqrt{\tilde{G}(0)/\tilde{G}(\hat{k}_{\min})-1}. \tag{3.23}$$

For yet another way of defining it, see e.g. ref. [45]. These three definitions coincide in the thermodynamic limit $L\to\infty$. In the case of the O(N) model, $\tilde{G}(\hat{k}_{\min})$ takes the form [42]

$$\tilde{G}(\hat{k}_{\min}) = \frac{1}{D}\sum_{\mu=1}^{D}\frac{1}{V}\left\langle \left| \sum_x \exp\left(i\, k_{\min}\, x_\mu\right) z(x) \right|^2 \right\rangle, \tag{3.24}$$

($\mathcal{O}(DL^DN)$ time per sample) while for the $\mathbb{C}\mathrm{P}^{N-1}$ model, it can be computed as [41]

$$\tilde{G}(\hat{k}_{\min}) = \frac{1}{D}\sum_{\mu=1}^{D}\sum_{a,b=1}^{N}\frac{1}{V}\left\langle \left| \sum_x \exp\left(i\, k_{\min}\, x_\mu\right) \bar{z}_a(x)\, z_b(x) \right|^2 \right\rangle \tag{3.25}$$

in $\mathcal{O}(DL^DN^2)$ time per sample.

A big advantage of the second-moment definition is that no data fitting is involved. The ratio of Fourier transforms implicitly isolates the exponential we are interested in. ξ_{2nd} is widely used in the literature as a good approximation of the real correlation length in a finite system.

3.2.4 Finite Size Scaling

If the correlation length is calculated on a finite lattice using one of the above methods, the obtained value can of course never exceed the lattice size L by definition. This is a finite size artifact; the system "sees its finite size" when it is simulated in regions of the parameter space where ξ is of the order of (or larger than) L. Many textbooks discuss these *finite size effects* in detail, see for example ref. [38]. The finite size scaling function

$$F(y) = \xi(2L)/\xi(L), \quad y = \xi(L)/L, \tag{3.26}$$

quantifies the severity of the finite size effects. It is closely related to the *step scaling function* first defined in ref. [46]. Numerical data for F on the two-dimensional O(3) model can be found in ref. [25] for instance. In refs. [9, 10] F is given for the $\mathbb{C}\mathrm{P}^2$ model and the second-moment definition of ξ. *Finite size scaling* then amounts to extrapolating to infinite system size by

$$\xi = \xi(\infty) = \xi(L)\prod_{k=0}^{\infty} F(y_0/2^k), \quad y_0 = \xi(L)/L. \tag{3.27}$$

Luckily, F is *universal* and hence independent of the coupling, which means that the error $|1 - \xi(L)/\xi(\infty)|$ made when measuring the correlation length in a finite system is constant as long as the physical system size y_0 remains the same. For $\mathbb{C}\mathrm{P}^3$ at $y_0 = 1/2$, $L = 16$, we measured $\xi(4L)/\xi(L) = 0.853(3)$ (ξ from eq. (3.12)), i.e. $\xi(L)$ actually overestimates the true correlation length. This should be kept in mind regarding our results in the following studies, in particular section 4.4.1.

3.3 Critical Slowing Down

We now have all means prepared to study critical slowing down (CSD). A MC algorithm applied to a specific model is said to suffer from *critical slowing down*, if the integrated autocorrelation time of the slowest mode (in units of sweeps!) diverges as a power law with the correlation length near a critical point, i.e. if

$$\tau^{\text{int}} \sim \xi^z, \quad z > 0, \quad \tau^{\text{int}} = \max_A \tau_A^{\text{int}}. \tag{3.28}$$

z is called *(integrated) dynamic critical exponent*. The same can in principle be defined for the *exponential* autocorrelation time [39], but the practical relevance is limited. In this work we focus on τ^{int} and $z = z^{\text{int}}$. Just like τ^{int}, z is a characteristic of the MC algorithm combined with the model it is applied to, and the lattice dimensionality D.

CSD is a pain. If z is large, the corresponding algorithm becomes very inefficient near phase transitions, or at large couplings for models with asymptotic freedom, where $\xi \gg 1$ (cf. fig. 4.2). The single-spin-flip Metropolis algorithm has a dynamic critical exponent $z = 2.1665(12)$ for the two-dimensional Ising model [47], where Wolff's cluster algorithm (cf. chapter 4) is much more efficient with $z = 0.25(1)$ [48]. Since power laws with small exponents are hard to distinguish from logarithmic divergences, small values of z, such as $z = 0.25(1)$, are very hard to determine, though. The data in ref. [48] is also consistent with a logarithmic divergence ($z = 0$), which has lead to conjecture that Wolff's cluster algorithm completely eliminates CSD for the O(N) model in two dimensions (see e.g. ref. [3]). The same observation (consistency with logarithmic divergence) holds for the worm algorithm applied to the 2D and 3D Ising and XY model [11] (see however ref. [49]). The efficiency of the latter two algorithms will be discussed in more detail in chapters 4 and 5.

3.3.1 Finite Volume Renormalization

Varying L and setting the β such that the Lüscher-Weisz-Wolff (LWW) coupling [46] $\bar{g}^2 = L/\xi(L) = m(L)L = \text{const.} \approx 2 - 3$ is common practice for the determination of the dynamic critical exponent, because the latter can then be extracted from the slope of τ vs. L in a log-log plot. This procedure of keeping $L/\xi(L)$ constant can be pictured as shortening the lattice spacing of a system with constant physical size and is called *finite volume renormalization*. It is applied in the following calculations.

To carry out finite volume renormalization, it is clear that reliable knowledge about the behavior (the so-called *running*) of the coupling as a function of the length scale ξ is desired. This can be found in sections 4.3.1 and 4.4.1, where we compare the scaling predictions from perturbation theory with numerical data.

4 Cluster Algorithms

In the 1980s R. H. Swendsen and J.-S. Wang proposed a completely new type of MC algorithm for statistical models [50]. Inspired by percolation theory (based on the work by Fortuin and Kasteleyn [51]), their idea was to build clusters of spins with ''similar'' orientation by introducing connecting bonds between them, and to reflect all spins belonging to the same cluster at once. The decisive point is that the activation probability of each bond is chosen such that the clusters have sizes of the order of ξ, rendering an elementary update much more effective than in a single-spin-flip algorithm near criticality, where $\xi \gg 1$. Each spin strongly interacts with its (widespread) neighborhood, and the algorithm is hence said to be *non-local*. It reduces the dynamic critical exponent quite substantially for a wide range of models, see e.g. refs. [48, 50, 52]. Furthermore, *improved estimators* can be used to reduce the variance of some observables.

U. Wolff published an improved version of the cluster algorithm in 1989 [3], formulated for the O(N) model. Instead of looking for all clusters in the system and reflecting each with probability 1/2, Wolff's algorithm identifies only one cluster per update and always reflects it. This simplifies the implementation, and in most cases even increases the efficiency. As mentioned above, Wolff found no sign for critical slowing down ($z \approx 0$ for τ_χ) when applying his algorithm to the O(N) model. In sections 4.3.2 and 4.4.2, we calculate the dynamic critical exponent for a larger number of observables on the O(3) model and the $\mathbb{C}\mathrm{P}^3$ model, respectively, showing that it is still inefficient on the latter.

A generic single-cluster update for our spin models consists of the following procedure [3]:

1. Draw a random spin r in the respective space ((2.1) or (2.13)).
2. Allocate an empty list of sites and add a random lattice site to it.
3. For each site x that was added in the previous step, add each nearest-neighboring site y with probability $P(x, y, r)$ (defined below) to the list, if it hasn't been added yet.
4. Repeat step 3 until no more sites were added. The list now represents a cluster $C \subset \Lambda$.
5. Reflect all spin vectors contained in the cluster on the hyperplane perpendicular to r.

Note that in each Wolff update, any of the DV bonds may be tested for activation merely once at most, but each site may be tested for addition from any direction, i.e. up to $2D$ times.

The bond activation probability reads [3]

$$P(x,y,r) = 1 - \exp\Big[\min\Big\{0, -2\beta\, R(x,y,r)\Big\}\Big] \tag{4.1}$$

(ref. [3] is missing a minus sign, while ref. [33] is mixing up $\max\{0,\cdot\}$ and $\max\{1,\cdot\}$ for both models!), where R is model-specific. For O(N) it is given by

$$R_{\mathrm{O}}(x,y,r) = \big(r\cdot z(x)\big)\big(z(y)\cdot r\big). \tag{4.2}$$

Reflecting the cluster in step 5 then amounts to setting

$$z(x) \to z(x) - 2\big(r\cdot z(x)\big)\, r \quad \forall x \in C. \tag{4.3}$$

Analogously, one finds for the $\mathbb{C}\mathrm{P}^{N-1}$ model [33]

$$R_{\mathbb{C}\mathrm{P}}(x,y,r) = 2\,\Re\Big\{\Big[\bar z(x)\cdot z(y) - \big(\bar z(x)\cdot r\big)\big(\bar r\cdot z(y)\big)\Big]\big(\bar r\cdot z(x)\big)\big(\bar z(y)\cdot r\big)\Big\} \tag{4.4}$$

and

$$z(x) \to z(x) - 2\big(\bar r\cdot z(x)\big)\, r \quad \forall x \in C. \tag{4.5}$$

In the introduction we briefly indicated the reason why Wolff-type embedding algorithms, such as the one above, are precluded from expunging critical slowing down for a particular class of σ models: The authors of ref. [7] argue that it may be efficient only if the manifold of fixed points of a particular update (the hyperplane perdendicular to r in our case) has codimension one in the space it is embedded in. Quite clearly, this is the case for the O(N) model, but not for $\mathbb{C}\mathrm{P}^{N-1}$. See sections 4.3.2 and 4.4.2 for numerical evidence.

4.1 Improved Estimators

The information encoded in the clusters may be used to reduce the variance of observable estimation [53]. Aside from the alleviation of critical slowing down (for O(N)), this is another major benefit coming along with cluster algorithms. Given a cluster C built during a Wolff update, and the associated reflection normal r, the Green's function of the O(N) model can be calculated from its *improved estimator* (IE) [5]

$$G(x,y) = NV\Big\langle \frac{1}{|C|}\delta_C(x)\delta_C(y)\, R_{\mathrm{O}}(x,y,r)\Big\rangle \tag{4.6}$$

where $|C|$ denotes the cluster size. δ_C is the cluster incidence function:

$$\delta_C(x) = \begin{cases} 1 & \text{if } x \in C \\ 0 & \text{otherwise} \end{cases}. \tag{4.7}$$

The IE for the (disconnected) magnetic susceptibility it is given by [5]

$$\langle\chi\rangle = N\Big\langle \frac{1}{|C|}\Big(\sum_{x\in C} z(x)\cdot r\Big)^2\Big\rangle. \tag{4.8}$$

This identity assigns the O(N) clusters a physical interpretation. Their size is proportional to the magnetic susceptibility, with a proportionality constant that exclusively depends on N [5]. In particular, it is unity for the Ising model.

For the $\mathbb{C}\mathrm{P}^{N-1}$ model we just mention the IE for the Green's function [33]:

$$G(x,y) = \frac{N+2}{2} V \Big\langle \frac{1}{|C|} \delta_C(x)\delta_C(y)\, R_{\mathbb{C}\mathrm{P}}(x,y,r) \Big\rangle \tag{4.9}$$

4.2 Parallelization

We place a brief note on parallelization here. The simple Metropolis spin flip algorithm is of course easily parallelizable thanks to its ultralocal elementary updates. This can go as far as multi-spin coding [54], a technique primarily popular in the 1980s for the Ising model. Cluster algorithms are by far less trivial to parallelize in general. Nevertheless, numerous attempts hereto have been made, many of them successful in some degree. A somewhat intuitive result of these is that the multi-cluster algorithm by Swendsen and Wang tends to be better suited for parallelization than Wolff's single-cluster adaption. A nice overview including a fairly well-scaling parallel implementation of Wolff's algorithm can be found in ref. [55].

What can be done with any MC algorithm, of course, is the trivial parallelization of letting multiple (n) simulations with the same setting (up to random seeds) run on different CPUs simultaneously. The thereby collected n statistically independent samples of any quantity of interest can then be used to reduce statistical errors by a factor of $1/\sqrt{n}$ by averaging. This is indeed a technique we have applied extensively for the present work.

4.3 Numerical Results for the O(N) Model

Table 4.1 lists some numerical results for the O(3) model simulated with Wolff's cluster algorithm. Topological modes ($\langle|Q|\rangle$, $\langle\chi_t\rangle$) are excluded, see section 2.1 for the reason why. We also simulated innumerable parameter settings (N,β,L) that are not tabulated, producing results invariably consistent with values in literature such as refs. [3–7, 12, 26, 27, 30, 44, 53, 56].

Table 4.1: Numerical results for the O(3) model obtained with Wolff's cluster algorithm at $L/\xi_{\mathrm{2nd}}(L) \approx 2$, averaged over n simulations for each setting. u is the number of updates after equilibration, s the number of updates between measurements, and $|C|$ the average cluster size. All autocorrelation times are in units of lattice sweeps and were computed from T_τ measurements with s_τ updates between each.

L	β	n	$u\times10^{-6}$	s	$T_\tau\times10^{-6}$	s_τ	$\lvert C\rvert/V$
16	1.55300	12	1000	15	15	1	0.243325(6)
23	1.62000	12	1000	20	15	1	0.220169(9)
32	1.67623	12	500	25	15	1	0.20162(2)
45	1.73215	12	300	30	15	1	0.18494(2)
64	1.78951	12	200	40	15	1	0.17028(3)
91	1.84646	12	100	50	15	1	0.15758(5)
128	1.90133	12	40	75	15	1	0.14670(8)
181	1.95674	12	30	100	15	2	0.13680(12)
256	2.01190	12	30	150	10	3	0.12755(8)
362	2.06763	12	10	250	2.5	4	0.1192(4)
512	2.12140	12	5	400	1	5	0.1105(5)

E/V	C_V/V	$\lvert M\rvert/V$	χ/V
-1.2544342(15)	1.57281(10)	0.562981(3)	0.322514(3)
-1.290678(2)	1.53586(10)	0.536435(2)	0.292540(2)
-1.319598(3)	1.49334(14)	0.513793(4)	0.268226(4)
-1.346588(3)	1.44736(19)	0.492426(6)	0.246282(5)
-1.372248(2)	1.4019(2)	0.472794(7)	0.226943(7)
-1.395712(3)	1.3617(4)	0.454981(11)	0.210096(9)
-1.4165642(15)	1.3273(5)	0.438955(18)	0.19552(2)
-1.436058(2)	1.3012(14)	0.42358(3)	0.18204(3)
-1.4540708(15)	1.2769(8)	0.40904(3)	0.16976(2)
-1.471034(4)	1.2566(19)	0.39559(6)	0.15878(5)
-1.486338(2)	1.245(4)	0.38186(11)	0.14798(8)

ξ_{2nd}	τ_E^{int}	τ_M^{int}	τ_χ^{int}
7.99740(12)	4.299(3)	1.8988(12)	1.9461(12)
11.49981(12)	4.619(4)	1.7381(6)	1.7928(7)
15.9796(3)	4.881(6)	1.6117(9)	1.6617(9)
22.4517(6)	5.156(5)	1.5112(11)	1.5487(16)
31.9638(15)	5.417(6)	1.4212(5)	1.4637(6)
45.523(3)	5.708(8)	1.3537(10)	1.3935(10)
64.111(8)	5.963(7)	1.3004(8)	1.3325(15)
90.649(18)	6.233(7)	1.254(2)	1.2863(6)
128.09(3)	6.499(6)	1.210(3)	1.2491(4)
180.97(6)	6.772(11)	1.182(4)	1.2181(7)
255.0(2)	7.018(12)	1.1382(9)	1.170(4)

4.3.1 Scaling and Asymptotic Freedom

As indicated earlier, the running of the coupling is a fundamental and highly substantial characteristic. For $N > 2$ perturbation theory predicts asymptotic freedom [24]:

$$1/\xi = m \approx C \exp\bigl(-2\pi\beta/(N-2)\bigr)\bigl(2\pi\beta/(N-2)\bigr)^{-1/(N-2)}. \tag{4.10}$$

The three-loop expansion can also be found in ref. [24]. Since this is of minor interest here we will not go into more detail. The scaling of the $\mathbb{C}\mathrm{P}^{N-1}$ model (section 4.4.1) will be more absorbing.

4.3.2 Dynamic Critical Exponents

In figure 4.1 we plot the integrated autocorrelation times τ_E, τ_M and τ_χ against the second-moment correlation length for Wolff's single-cluster algorithm on the two-dimensional Heisenberg model. The data set is the same as in table 4.1. The fits are of the form

$$\tau^{\mathrm{int}}(L) = c_1 \xi_{2\mathrm{nd}}(L)^z \bigl(1 + c_2/\log[\xi_{2\mathrm{nd}}(L)]\bigr). \tag{4.11}$$

Our results show no indication for critical slowing down, confirming that Wolff's clusters update the $\mathrm{O}(N)$ model efficiently. All data are also consistent with the purely logarithmic form $\tau^{\mathrm{int}}(L) = \tau_0 + c\log[\xi_{2\mathrm{nd}}(L)]$, i.e. with $z = 0$.

4.4 Numerical Results for the $\mathbb{C}\mathrm{P}^{N-1}$ Model

Table 4.2 lists some numerical results for the $\mathbb{C}\mathrm{P}^3$ model we simulated with Wolff's cluster algorithm. We have also consistently reproduced numerous results from literature, notably refs. [8, 13, 33, 57, 58], which we don't tabulate here.

We must stress that our numerical results for the correlation length at $L > 32$ are reproducibly inconsistent with the perturbative prediction, i.e. too low. These values are marked with an asterisk. The origin of this systematic discrepancy remains unclear. A poor fitting quality can be excluded from the range of possibilities, as it remains high also at large L (cf. fig. 3.3). Also, we don't expect τ_ξ^{int} to be large enough to cause such a significant bias. In any case, the values for β in the marked simulations were chosen such that $\xi(L) = L/2$ in theory according to eq. (4.12). This will allow us to use the tabulated data to examine the dynamic critical behavior without too many finite size effects in the next section.

Table 4.2: Numerical results for the $\mathbb{C}\mathrm{P}^3$ model obtained with Wolff's cluster algorithm at $L/\xi(L) \approx 2$, averaged over n simulations. ξ is extracted from the exponential decay of G_c, eq. (3.12). See the caption of table 4.1 for an explanation of the listed parameters. Asterisks mark inconsistencies with perturbative predictions (see text).

L	β	n	$u\times10^{-6}$	s	$T_\tau\times10^{-6}$	s_τ	$\lvert C\rvert/V$
12	5.08540	23	2000	100	15	1	0.755398(17)
16	5.26290	20	1710	180	15	1	0.76657(3)
20	5.39750	16	1100	275	15	2	0.77541(3)
25	5.53172	15	700	350	15	2	0.78405(4)
32	5.68002	20	420	350	15	2	0.79296(7)
40	5.81504	17	268	400	15	3	0.80061(8)
50	5.94980	24	170	500	15	4	0.80747(13)
64	6.09912	17	100	500	15	5	0.81464(14)
80	6.23418	23	64	640	12.8	5	0.8203(2)
100	6.36944	20	40	800	5	8	0.8262(2)
128	6.51914	24	24	1000	2	12	0.8313(4)

E/V	C_V/V	$\lvert M\rvert/V$	χ/V	$\lvert Q\rvert$	$\chi_t\times10^3$
-1.279536(3)	6.2890(5)	0.079488(2)	0.174667(3)	0.26104(3)	2.1169(3)
-1.310039(3)	6.0573(7)	0.059714(3)	0.155744(4)	0.31342(4)	1.4699(3)
-1.333137(4)	5.7846(13)	0.047818(2)	0.144011(6)	0.34244(9)	1.0433(3)
-1.355279(5)	5.4882(13)	0.038284(2)	0.134223(8)	0.36010(11)	0.7078(3)
-1.378294(3)	5.1683(13)	0.029934(2)	0.125040(8)	0.36671(14)	0.4407(2)
-1.397753(3)	4.921(3)	0.023964(3)	0.11781(2)	0.3634(3)	0.2783(3)
-1.415800(3)	4.710(2)	0.019183(3)	0.111153(17)	0.3549(4)	0.1729(3)
-1.434333(4)	4.526(4)	0.015000(4)	0.10427(4)	0.3425(5)	0.10099(18)
-1.449922(4)	4.396(4)	0.012001(3)	0.09848(5)	0.3291(7)	0.06160(16)
-1.464553(3)	4.284(8)	0.009606(4)	0.09306(7)	0.3152(10)	0.03748(14)
-1.479704(7)	4.188(9)	0.007503(4)	0.08720(12)	0.3023(18)	0.02170(16)

ξ	τ_E^{int}	τ_M^{int}	τ_χ^{int}	τ_Q^{int}	$\tau_{\chi_t}^{\mathrm{int}}$
6.0028(3)	65.68(7)	105.0(3)	92.61(13)	5.138(14)	4.123(3)
8.0019(6)	71.47(15)	127.1(4)	142.7(5)	5.928(18)	4.772(4)
9.9998(10)	72.15(11)	146.9(4)	201.2(9)	6.72(4)	5.378(5)
12.4940(19)	70.12(14)	168.5(5)	286.8(1.1)	7.490(5)	6.05(3)
15.994(3)	66.73(9)	196.5(6)	426(3)	8.214(11)	6.89(4)
19.980(8)*	63.68(8)	224.0(7)	620(3)	9.069(10)	7.476(13)
24.895(9)*	61.51(4)	253.4(5)	909(4)	9.920(16)	8.38(8)
31.61(3)*	59.88(7)	288.5(6)	1439(9)	10.88(9)	9.17(11)
39.21(4)*	59.18(5)	320.1(7)	2198(29)	11.20(6)	9.58(6)
48.60(9)*	58.77(4)	354.5(19)	3366(42)	12.73(20)	10.78(14)
61.16(19)*	58.63(5)	395.4(19)	5832(136)	15.3(4)	12.9(3)

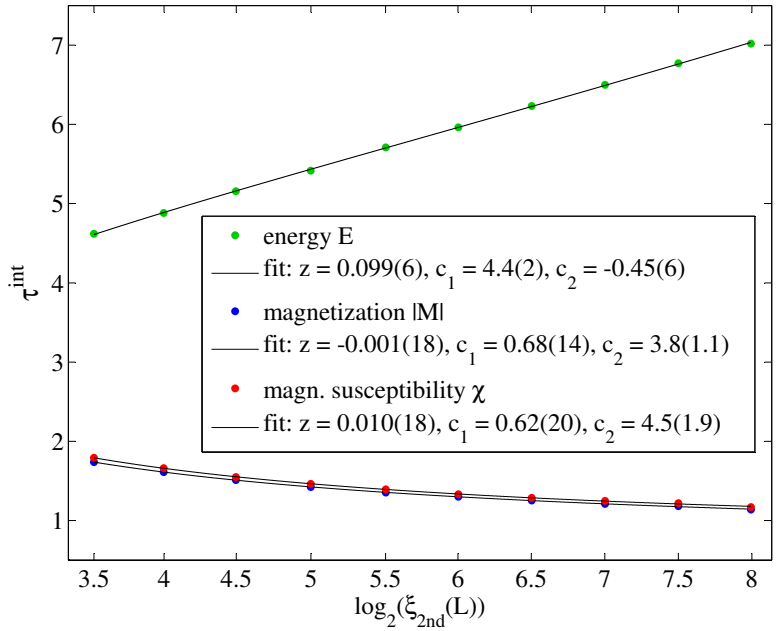


Figure 4.1: Semi-log plot of integrated autocorrelation times for Wolff's cluster algorithm applied to the 2D O(3) model at constant physical size $L/\xi_{2\mathrm{nd}}(L) \approx 2$. Horizontal and vertical error bars are smaller than the dots and were therefore omitted. In the legend we give 95% confidence bounds for the fit parameters in brackets. χ^2_{red} is 2.9, 14.1 and 38.6, respectively.

4.4.1 Scaling and Asymptotic Freedom

The perturbative two-loop prediction for asymptotic scaling in the $\mathbb{C}\mathrm{P}^{N-1}$ models is [33, 41, 59]

$$1/\xi = m \approx C_m^p \exp(-2\pi\beta/N)(2\pi\beta/N)^{-2/N}, \tag{4.12}$$

where C_m^p needs to be determined non-perturbatively. (Note that refs. [33, 41] use β/N as the coupling, while the β in ref. [59] coincides with our $\beta/2$.) For $\mathbb{C}\mathrm{P}^3$ we find $C_m^p = 1400.4(3)$, in contrast to Jansen & Wiese's erroneous result of 190(6). To obtain the true thermodynamic constant, our value should be finite-size scaled (cf. section 3.2.4). Figure 4.2 shows a comparison of the data. Clearly, Jansen & Wiese also underestimated their errors. The second-loop power law correction significantly increases the goodness of fit compared to the pure exponential from one-loop expansion ($\chi^2_{\mathrm{red}} = 0.53$ vs. $\chi^2_{\mathrm{red}} = 2344$). In the inset, m is reduced by its first-loop exponential and then log-log plotted against the coupling. The black straight line represents the pure second-loop power law. Our data is precise enough to show a clear systematic deviation from the two-loop formula for $\beta < 5.3$ supposably arising from higher-order corrections. Note that a few blue dots carry a slight bias in cases where $L/\xi(L)$ didn't precisely hit two. Both fitting and goodness-of-fit analysis were restricted to data points no. 10 to 15 ($L = 17$ to 32), where the higher-order corrections disappear and yet our numerical correlation lengths don't veer away from the expected first-order exponential.

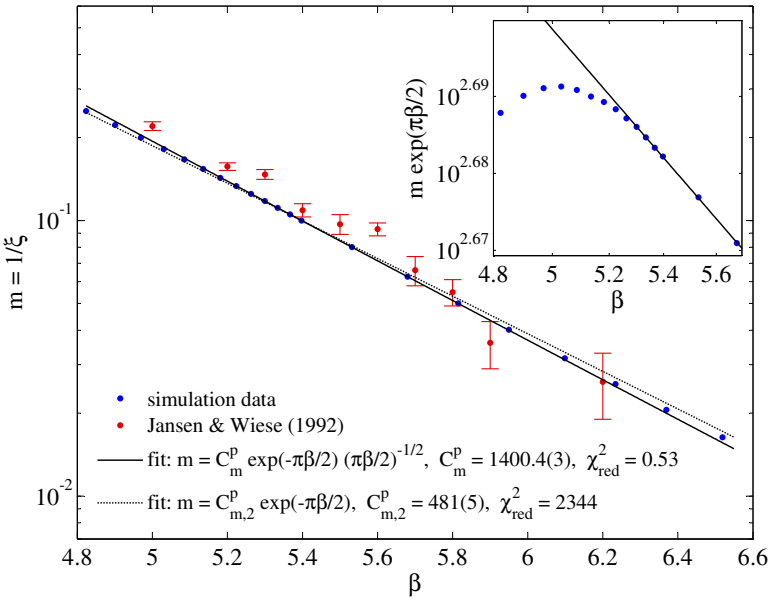


Figure 4.2: Scaling and asymptotic freedom of the $\mathbb{C}\mathrm{P}^3$ model. Our data (blue) combine the results from table 4.2 with additional simulations to refine the range of couplings. 95% confidence intervals for the fitted prefactors are given in brackets. Error bars are smaller than the symbols.

4.4.2 Dynamic Critical Exponents

The integrated autocorrelation times from table 4.2 are plotted against the correlation length in figure 4.3. The horizontal axis displays the *expected* correlation length $\xi(L) = L/2$ according to the perturbative prediction for $\xi \geq 20$, where our numerical results for $\xi(L)$ veer away from $L/2$. The magnetic susceptibility is by far the slowest mode, while the topological modes decorrelate the fastest in the systems examined here. A pure power law of the form $\tau^{\mathrm{int}}(L) = c\xi(L)^z$ is fitted to each of the modes in the range where the power law is dominant, revealing a dynamic critical exponent of about two for Wolff's single-cluster algorithm on the $\mathbb{C}\mathrm{P}^3$ model. Jansen & Wiese's early crude estimate of $z \approx 2$ [33] is confirmed. This concludes our studies of the cluster algorithm, which is evidently unable to accelerate the $\mathbb{C}\mathrm{P}^{N-1}$ model. The next chapter will be dedicated to the worm algorithm.

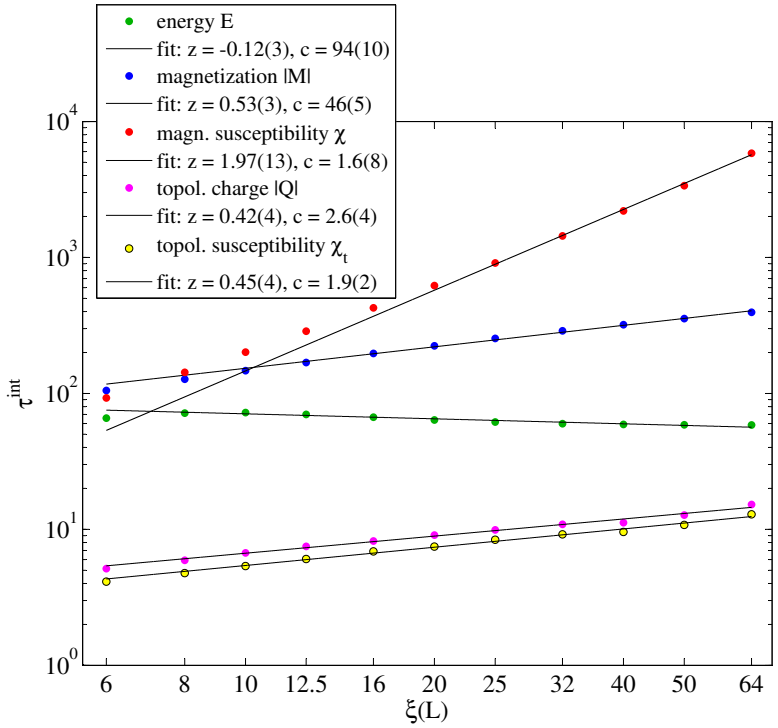


Figure 4.3: Log-log plot of integrated autocorrelation times for Wolff's cluster algorithm applied to the 2D $\mathbb{C}\mathrm{P}^3$ model at constant physical size $L/\xi(L) \approx 2$. Horizontal and vertical error bars are smaller than the dots and were therefore omitted. In the legend we give 95% confidence bounds for the fit parameters in brackets.

5 The Worm Algorithm

In the late 1990s, N. V. Prokof'ev, B. V. Svistunov and I. S. Tupitsyn developed a novel type of local MC algorithms for quantum systems and called it the worm algorithm (WA) [60]. The basic underlying idea is that instead of sampling the trajectories between two fixed world-line discontinuities, the trajectory itself is held still and the end points are moved around on the lattice. The resulting ergodic movement of the discontinuities (then called "head" and "tail") looks like a worm, hence the name. The WA directly samples the Green's function, as we will see later.

Prokof'ev and Svistunov later reformulated the worm algorithm for classical statistical systems [11]. The key ingredient of the classical WA is a duality transformation [61] of the partition function from field vectors to bond variables forming closed loops, which can be achieved from a high-temperature expansion of the Boltzmann weights. The findings concerning the efficiency is rather astonishing for a MC algorithm that operates locally: Critical slowing down is dramatically reduced on the tested models and even completely absent on the two- and tree-dimensional Ising and XY model [11, 49].

In the following section we sketch the construction of the worm algorithm for the Ising model to provide some intuition of its main principles. Then, in section 5.2, the ideas of the worm algorithm are deployed to the $\mathbb{C}\mathrm{P}^{N-1}$ model. In both cases our focus will be restricted to the prominent high-temperature expansion as the technique for reformulating the configuration space and arriving at a loop representation. There exist other ways however, for instance domain boundaries for lattice models with discrete spins variables [11].

5.1 Worm Algorithm for the Ising Model

We start off with the simple Ising model, for which the worm formulation is particularly plain and many technicalities cancel, to demonstrate the basic principles informing the worm algorithm.

5.1.1 From Spins to Bonds

The lattice partition function in terms of spin configurations $\{z\}$ reads

$$Z = \sum_{\{z\}} \exp\bigl(-S(\{z\})\bigr) = \sum_{\{z\}} \exp\Bigl(\beta \sum_{\langle x,y\rangle} z(x)\, z(y)\Bigr) = \sum_{\{z\}} \prod_{\langle x,y\rangle} \exp\bigl(\beta\, z(x)\, z(y)\bigr), \quad (5.1)$$

i.e. it is the sum of Boltzmann weights over all spin configurations. Now, we perform a *high-temperature expansion* (or *strong coupling expansion* in the terminology of QFT),

which is a mere Maclaurin series expansion in β (i.e. for strong couplings), on each of these weights [11]:

$$\exp(\,\cdot\,) = \sum_{n=0}^{\infty} \frac{(\,\cdot\,)^n}{n!} \quad \Longrightarrow \quad Z = \sum_{\{z\}} \prod_{\langle x,y\rangle} \sum_{n_{xy}=0}^{\infty} \frac{\beta^{n_{xy}}}{n_{xy}!} \big(z(x)\, z(y)\big)^{n_{xy}} \tag{5.2}$$

The term *high-temperature expansion* is due to the Taylor expansion being a good approximation at the lower order the smaller the inverse temperature β, i.e. the higher the temperature. Each pair of nearest-neighbor sites $\langle x, y\rangle$, which will be called *bond* from now on, is assigned a *bond variable* (or *bond state*, *bond occupation number*) $n_{xy} \in \mathbb{N}_0$ constituting the order of expansion of its Boltzmann weight. For each configuration $\{n\}$ of bond states the summation over spins factorizes and we can rewrite the partition function as [11]

$$Z = \sum_{\{n\}} \left(\prod_{\langle x,y\rangle} \frac{\beta^{n_{xy}}}{n_{xy}!} \right) \left(\prod_{x} Q\big(k(x)\big) \right), \tag{5.3}$$

where

$$Q\big(k(x)\big) = \sum_{z(x)=\pm 1} z(x)^{k(x)} = \begin{cases} 2 & \text{if } k(x) \text{ is even} \\ 0 & \text{if } k(x) \text{ is odd} \end{cases} \tag{5.4}$$

is the sum of all possible spin values at site x with *site charge*

$$k(x) = \sum_{\mu=1}^{D} \big(n_{x(x+\hat{\mu})} + n_{(x-\hat{\mu})x}\big). \tag{5.5}$$

In other words, by summing over bond configurations rather than spin configurations, we can explicitly sum out the spin variables z (eq. (5.4)), and solely describe our system in terms of bond variables! The charge $k(x)$ is nothing but the sum of all bond variables incident on site x. It counts the number of $z(x)$ factors in an expansion term.

Now, obviously we have $Z = 0$ if there exists any site x with an odd charge $k(x)$, which means that any such configuration doesn't contribute to the partition function. In figure 5.1 we see what that means. The bond variables may be visualized by drawing lines between sites, as many (or as thick) as their values specify. The evenness condition on the charges then corresponds to these lines forming *closed loops* (or *closed paths*, CP, not to be confused with $\mathbb{C}$P). Any configuration with open paths has zero weight and can therefore be excluded from the summation. This finding is most vital to the WA, as we will soon see. We henceforth denote the set of closed-path configurations by CP. By combining (5.3) and (5.4) we find the final expression for the Ising partition function in the worm formalism:

$$Z = 2^V \sum_{\{n\}\in CP} \left(\prod_{\langle x,y\rangle} \frac{\beta^{n_{xy}}}{n_{xy}!} \right). \tag{5.6}$$

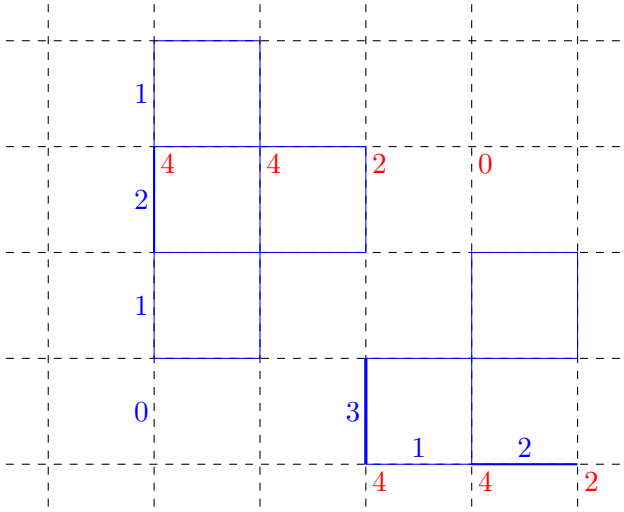


Figure 5.1: Illustration of the worm algorithm I: Sample closed path configuration contributing to Z. Each bond $\langle x, y\rangle$ is populated with a bond variable n_{xy} (blue lines with a width proportional to their values, or black dashed lines if zero). The bond variables incident on site x add up to the charge $k(x)$ (red).

5.1.2 The Algorithm

Above, we transformed the Ising field to a dual loop/graph representation, often called *loop gas*. Simulating the fluctuations of this loop gas is the task of a Monte Carlo algorithm. The worm algorithm is a simple local Metropolis scheme that acts on the bond states by updating them one after the other, transforming one CP configuration into another and thereby sampling Z. This is achieved by relaxing the CP constraint temporarily: during an update, two sites x_h, x_t carry a "field insertion" $z(x_\mathrm{h})$ and $z(x_\mathrm{t})$, respectively, and are hence enabled to have an odd sum of incident bond variables contributing to the (still even) charge

$$k(x) = \sum_{\mu=1}^{D} \left(n_{x(x+\hat{\mu})} + n_{(x-\hat{\mu})x}\right) + \delta_{x,x_\mathrm{h}} + \delta_{x,x_\mathrm{t}}, \tag{5.7}$$

i.e. they form the endpoints of an open path, called "worm". x_h and x_t represent the "head" and "tail" of the worm. We have

$$\sum_{\mu=1}^{D} \left(n_{x(x+\hat{\mu})} + n_{(x-\hat{\mu})x}\right) = \begin{cases} \text{odd} & \text{for } x = x_\mathrm{h}, x_\mathrm{t} \\ \text{even} & \text{elsewhere} \end{cases} \tag{5.8}$$

as long as the worm forms an open path ($x_\mathrm{h} \neq x_\mathrm{t}$). The reason why it is precisely two sites falls into place when the translation invariant two-point function, that we would like to calculate with the algorithm, is considered:

$$G(x_\mathrm{h} - x_\mathrm{t}) = \langle z(x_\mathrm{t})\, z(x_\mathrm{h})\rangle = \frac{1}{Z}\sum_{\{z\}} z(x_\mathrm{t})\, z(x_\mathrm{h}) \exp\bigl(-S(\{z\})\bigr) = \frac{1}{Z} g(x_\mathrm{h} - x_\mathrm{t}) \tag{5.9}$$

By performing the same high-temperature expansion to g as to Z above, we immediately realize that g and Z only differ in the exponents of $z(x_\mathrm{t})$ and $z(x_\mathrm{h})$, which have increased by one and are odd for g:

$$g(x_\mathrm{h} - x_\mathrm{t}) = \sum_{\{z\}} \prod_{\langle x,y\rangle} \sum_{n_{xy}=0}^{\infty} \frac{\beta^{n_{xy}}}{n_{xy}!} \big(z(x)\, z(y)\big)^{n_{xy}} z(x_\mathrm{t})\, z(x_\mathrm{h}) \tag{5.10}$$

Thus, while relaxing the CP constraint on exactly two sites for a transition between valid CP configurations, we can directly sample g. In the WA, the tail x_t will always remain seated during an update, while the head x_h moves around. Each bond it traverses is either increased or decreased in value, a process one may picture by drawing or erasing lines. The locomotion of the head is stopped only when it coincides with the tail again, i.e. when the open path closes and a (new) CP configuration is found. All we need to take care of is moving the head with the correct Metropolis probabilities according to (5.6).

The final worm algorithm for the Ising model goes as follows [11]:

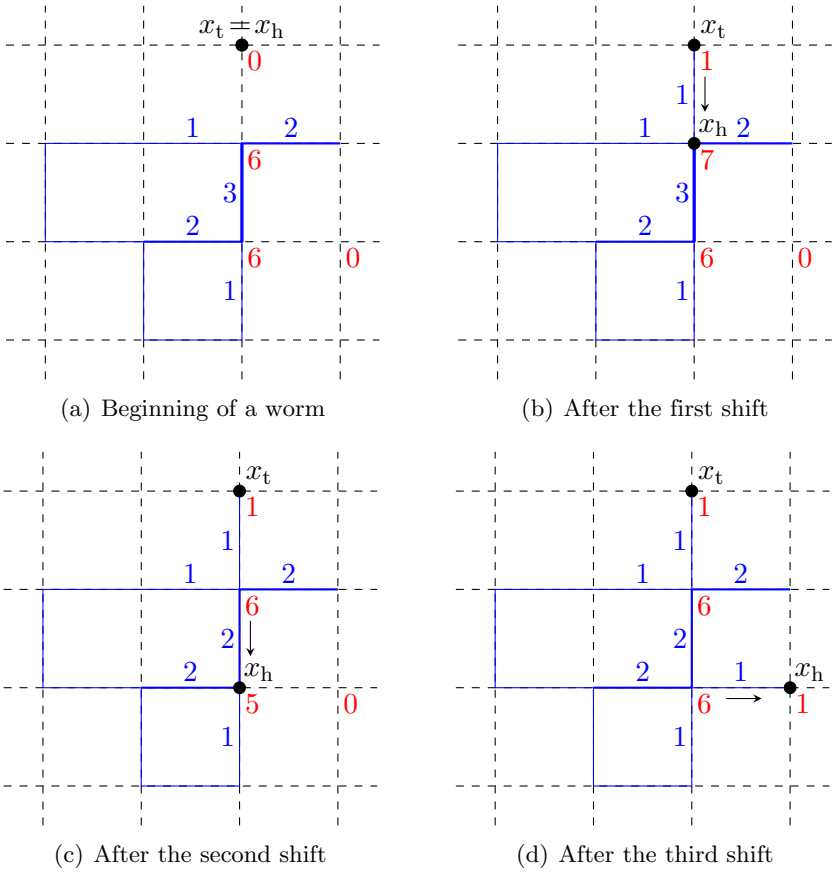
1. Allocate an array the size of the lattice for g and zero it. Set $Z = 0$.
2. Assign each of the DV bonds an initial non-negative integer value such that they form a CP configuration, e.g. all zero.
3. Select a random site x_m and set $x_\mathrm{h} = x_\mathrm{t} = x_\mathrm{m}$.
4. Pick one of the $2D$ bonds incident on x_h at random and call its other end x_c (c for candidate).
5. Propose to increase or decrease the bond variable $n_{x_\mathrm{h}x_\mathrm{c}}$ by 1 with probability $\frac{1}{2}$.
6. Perform the change with probability $P_\mathrm{sh} = \min\{1, R_\mathrm{sh}\}$ (defined below). If accepted, ‘‘shift’’ the head to the candidate site, i.e. set $x_\mathrm{h} = x_\mathrm{c}$.
7. Increment $g(x_\mathrm{h} - x_\mathrm{t})$ by 1, regardless of whether the shift was accepted or not.
8. If $x_\mathrm{h} = x_\mathrm{t}$, increment Z by 1, measure any desired quantity on the current configuration (see below) and go to step 3, otherwise continue with step 4.

The Metropolis acceptance ratios are straightforwardly derived from (5.3, 5.7) [11]:

$$\begin{aligned} R_\mathrm{sh}(x_\mathrm{h} \to x_\mathrm{c},\, n_{x_\mathrm{h}x_\mathrm{c}} \to n_{x_\mathrm{h}x_\mathrm{c}} + 1) &= \frac{\beta}{n_{x_\mathrm{h}x_\mathrm{c}} + 1} \frac{Q\big(k(x_\mathrm{c}) + 2\big)}{Q\big(k(x_\mathrm{c})\big)} = \frac{\beta}{n_{x_\mathrm{h}x_\mathrm{c}} + 1} \\ R_\mathrm{sh}(x_\mathrm{h} \to x_\mathrm{c},\, n_{x_\mathrm{h}x_\mathrm{c}} \to n_{x_\mathrm{h}x_\mathrm{c}} - 1) &= \frac{n_{x_\mathrm{h}x_\mathrm{c}}}{\beta} \frac{Q\big(k(x_\mathrm{h}) - 2\big)}{Q\big(k(x_\mathrm{h})\big)} = \frac{n_{x_\mathrm{h}x_\mathrm{c}}}{\beta} \end{aligned} \tag{5.11}$$

Note that their remarkable simplicity is thanks to the 2-periodicity of Q for the Ising model. The ratio of Q functions doesn't cancel in more elaborate models, such as $\mathbb{C}\mathrm{P}^{N-1}$, which will be considered later on. Also, the relocation of head and tail to a randomly selected site in step 3, called ‘‘move’’, which we can always accept here, is in general subject to an acceptance ratio R_mv of Q functions.

Each worm (steps 3 to 8, until $x_\mathrm{h} = x_\mathrm{t}$) transforms the system from one CP configuration into another (not necessarily different) one. The intermediate local head shifts might look inefficient at first glance, but they aren't. Each one of them, whether accepted or not, increases the statistics for the Green's function g by 1. Figure 5.2 shows a series of consecutive head shifts, each of the configurations contributing to $g(x_\mathrm{h} - x_\mathrm{t})$ but not to Z (except the first), because the path has not yet closed ($x_\mathrm{h} \neq x_\mathrm{t}$). Along its way, the worm may build new loops by increasing bond variables, erase ("backtrack") others by decreasing them, change their path, combine or even break up some of them, all during the same journey of head shifts. In Prokof'ev & Svistunov's picture, it is hence wrong to imagine a worm as a mere (purely positive or negative) path record of the diffusive head. A worm might for instance "eat itself up" and leave a part behind, forming a stand-alone closed loop. From two consecutive CP configurations created by the WA, it is in general impossible to reconstruct the head trajectory, because the worm is empowered to switch from drawing to erasing lines and vice versa at each head shift independently.



(a) Beginning of a worm (b) After the first shift

(c) After the second shift (d) After the third shift

Figure 5.2: Illustration of the worm algorithm II: Series of head shifts producing configurations contributing to g, in consecutive order (a to d). The marking is the same as in fig. 5.1, with the red numbers indicating the sums of incident bond states (5.8). Head and tail are additionally marked by black dots. Arrows identify head shift directions.

As soon as enough statistics are collected, we can abort the scheme and obtain measurables from the collected data according to (2.4), (2.7) and (5.9) by

$$\langle E\rangle = -\frac{V}{DZ}\sum_{|x|=1} g(x), \tag{5.12}$$

$$\langle \chi\rangle = \frac{1}{Z}\sum_{x} g(x), \tag{5.13}$$

The correlation length can be computed using a fitting method described in section 3.2. The Green's function $G(x) = g(x)/Z$ comes at particularly low cost in the worm algorithm, which, aside from the low dynamic critical exponent, is another major advantage of the WA. Computing autocorrelation times of measurables calculated from the Green's function, on the other hand, is a bit less convenient with the WA, because no MC time series of measurements is naturally on hand, for G being available at full precision no earlier than at the end of MC dynamics. Therefore, for computing autocorrelation times of such modes, one resorts to binning the accumulation of statistics for G, calculating the measurables on each bin separately and then performing the usual autocorrelation analysis on the resulting series [13]. The price to pay is a high variance, of course. In principle, one could set the bin size to a single head shift proposal only, with the binned Green's function then being $g_{\mathrm{bin}}(x) = \delta_{x,x_\mathrm{h}-x_\mathrm{t}}$, but the induced variance is of course preposterous. Rather, bins of size $\sum_x g_{\mathrm{bin}}(x) = \mathcal{O}(DV)$ should be used, i.e. proportional to the number of lattice degrees of freedom in the system.

It is worth stressing that the above transformation recipe and sampling algorithm are both directly applicable to any hypercubic lattice dimensionality D without modification. This generality property accentuates the beauty of the WA and facilitates a generic implementation.

5.1.3 Direct Estimators

As usually, there exists a second way (other than the Green's function) to determine the energy, also for the WA, namely from the logarithmic derivative of the partition function w.r.t. β [11]:

$$\begin{aligned}\langle E\rangle &= -\frac{\mathrm{d}\ln Z}{\mathrm{d}\beta} = -\frac{1}{Z}\frac{\mathrm{d}Z}{\mathrm{d}\beta} = -\frac{1}{Z}\left[2^V\sum_{\{n\}\in CP}\left(\frac{\mathrm{d}}{\mathrm{d}\beta}\prod_{\langle x,y\rangle}\frac{\beta^{n_{xy}}}{n_{xy}!}\right)\right]\\ &= -\frac{1}{Z}\left[2^V\sum_{\{n\}\in CP}\left(\frac{1}{\beta}\sum_{\langle x,y\rangle} n_{xy}\right)\left(\prod_{\langle x,y\rangle}\frac{\beta^{n_{xy}}}{n_{xy}!}\right)\right] = -\frac{1}{\beta}\Big\langle\sum_{\langle x,y\rangle} n_{xy}\Big\rangle_{CP},\end{aligned} \tag{5.14}$$

which is perfectly intuitive. $\langle\cdot\rangle_{CP}$ denotes the thermal average over closed-path configurations. To obtain a direct MC estimator for the energy from a CP configuration we can hence just sum up all bond variables in step 8 of the WA, when Z is non-zero, or better even keep track of the change ±1 during the head shifts, record in step 8, and divide by β after averaging. Analogously we find the heat capacity

$$
\begin{aligned}
\langle C_V \rangle &= \left(\frac{\mathrm{d}\langle E \rangle}{\mathrm{d}\beta^{-1}} \right)_V = \beta^2 \frac{\mathrm{d}}{\mathrm{d}\beta} \left(\frac{1}{Z} \frac{\mathrm{d}Z}{\mathrm{d}\beta} \right) \\
&= \frac{\beta^2}{Z^2} \left(\frac{\mathrm{d}^2 Z}{\mathrm{d}\beta^2} Z - \left(\frac{\mathrm{d}Z}{\mathrm{d}\beta} \right)^2 \right) = \frac{\beta^2}{Z} \frac{\mathrm{d}^2 Z}{\mathrm{d}\beta^2} - \Big\langle \sum_{\langle x,y \rangle} n_{xy} \Big\rangle_{CP}^2 \\
&= \frac{2^V}{Z} \sum_{\{n\} \in CP} \Big(\sum_{\langle x,y \rangle} n_{xy} \Big) \Big(\sum_{\langle x,y \rangle} n_{xy} - 1 \Big) \Big(\prod_{\langle x,y \rangle} \frac{\beta^{n_{xy}}}{n_{xy}!} \Big) - \Big\langle \sum_{\langle x,y \rangle} n_{xy} \Big\rangle_{CP}^2 \\
&= \Big\langle \Big(\sum_{\langle x,y \rangle} n_{xy} \Big) \Big(\sum_{\langle x,y \rangle} n_{xy} - 1 \Big) \Big\rangle_{CP} - \Big\langle \sum_{\langle x,y \rangle} n_{xy} \Big\rangle_{CP}^2 \\
&= \Big\langle \Big(\sum_{\langle x,y \rangle} n_{xy} \Big)^2 \Big\rangle_{CP} - \Big\langle \sum_{\langle x,y \rangle} n_{xy} \Big\rangle_{CP} - \Big\langle \sum_{\langle x,y \rangle} n_{xy} \Big\rangle_{CP}^2 \\
&= \beta^2 \Big(\langle E^2 \rangle - \langle E \rangle^2 \Big) + \beta \langle E \rangle,
\end{aligned}
\tag{5.15}
$$

which differs from the common formula in spin notation (2.6)! Autocorrelation times for E and C_V can be computed in the usual way as described in section 3.1.

Spin-like observables like the magnetization or derivative modes thereof, on the other hand, are incomparably less manifest. Since the spins no longer exist in our system, they can at best be retrieved from backward mapping the loop configuration to spin fields, which is only possible for a fraction of configurations, even for the unpretentious Ising model [62]. We expect this portion to shrink with increasing spin dimension N and to be insufficient for as low as $N \geq 2$ in O(N), but are unaware of any rigorous investigation in this regard.

5.1.4 Variant Implementations

Note that there exists a substantial performance enhancement trick in the simple case of the Ising model [11]: One may replace the full Maclaurin series expansion in (5.2) by the identity

$$
\exp\big(\beta\, z(x)\, z(y)\big) = \cosh(\beta) \sum_{n_{xy}=0,1} \big[\tanh(\beta)\, z(x)\, z(y) \big]^{n_{xy}} \tag{5.16}
$$

to truncate the configuration space of bond variables to binary values $n_{xy} \in \{0, 1\}$. As this is of no use in this illustrative example of the WA fundamentals, we omit further details, which are straightforward.

Also, the relocation of head and tail every time the worm closes may in principle be performed at any probability $p < 1$ or even completely skipped without compromising the correctness of the results [11, 49]. However, equilibration will take a lot longer in the latter case, of course, particularly for large physical system sizes L/ξ.

Moreover, keeping the tail fixed while shifting the head is a simplification enabled by translation invariance. Using the exact same rules as for the head, also the tail could be shifted around on the lattice, although at no additional revenue, which is why this is usually not implemented.

5.2 Worm Algorithm for the $\mathbb{C}\mathrm{P}^{N-1}$ Model

Prokof'ev and Svistunov successfully applied their worm algorithm to the complex scalar ϕ^4 theory [11]. The essential difference to real-valued models is that the lines drawn along bonds become oriented. Both directions can (and do) coexist, one of them belonging to $\bar{z}(x)\, z(y)$ and the other to the complex conjugate. A similar generalization will occur in our deployment to the $\mathbb{C}\mathrm{P}^{N-1}$ model.

5.2.1 From Spins to Bonds

A compact derivation of the high-temperature expansion for the $\mathbb{C}\mathrm{P}^{N-1}$ partition function has been given by S. Chandrasekharan in 2008 [63]. We replicate it here in some more detail for completeness and to introduce the terms and nomenclature. Again starting from the partition function in terms of spin configurations (2.13),

$$\begin{aligned} Z &= \int \exp\bigl(-S(\{z\})\bigr) \prod_x \mathrm{D}z(x) \\ &= \int \prod_{\langle x,y\rangle} \prod_{a,b=1}^{N} \exp\Bigl[\beta\bigl(\bar{z}_a(x)\, z_b(x)\bigr)\bigl(\bar{z}_b(y)\, z_a(y)\bigr)\Bigr] \prod_x \mathrm{D}z(x) \end{aligned} \tag{5.17}$$

with infinitesimal normalized invariant $\mathbb{C}\mathrm{P}^{N-1}$ measure

$$\mathrm{D}z = \delta(|z|^2 - 1)\, \mathrm{d}^N\bar{z}\, \mathrm{d}^N z, \tag{5.18}$$

we Maclaurin-expand in β to obtain

$$Z = \int \prod_{\langle x,y\rangle} \prod_{a,b=1}^{N} \sum_{n_{xy}^{ab}=0}^{\infty} \Bigl[\frac{\beta^{n_{xy}^{ab}}}{n_{xy}^{ab}!}\bigl(\bar{z}_a(x)\, z_b(x)\bigr)^{n_{xy}^{ab}}\bigl(\bar{z}_b(y)\, z_a(y)\bigr)^{n_{xy}^{ab}}\Bigr] \prod_x \mathrm{D}z(x). \tag{5.19}$$

While this expression looks rather lengthy, it is actually the same as in the aforementioned Ising model, except that each bond $\langle x, y\rangle$ now carries a bond state for each "flavor" pair ab, such that we need to assign each bond an $N{\times}N$ *bond matrix* $n_{xy} \in \mathbb{N}_0^{N\times N}$. Its abth element n_{xy}^{ab} gives the power of the $\bigl(\bar{z}_a(x)\, z_b(x)\bigr)$ factor in the expansion, and at the same time the power of $\bigl(\bar{z}_b(y)\, z_a(y)\bigr)$. In a way, the bond matrix seen from site x is n_{xy}, and n_{xy}^T from the perspective of site y.

As usually, we can now integrate over all possible spin fields. Let $z_a = r_a \exp(i\phi_a)$, $r = (r_1, ..., r_N)$, $\phi = (\phi_1, ..., \phi_N)$, and $k = (k_1, ..., k_N)$, $l = (l_1, ..., l_N) \in \mathbb{N}_0^N$. Then

$$\begin{aligned} Q(k,l) &= \int \prod_{a=1}^{N} (\bar{z}_a)^{k_a} (z_a)^{l_a} \, \mathrm{D}z(x) \\ &= \int \delta(|r|-1)\, \delta_{[0,2\pi]^N}(\phi) \prod_{a=1}^{N} r_a^{k_a+l_a+1} \exp\left[i\phi_a(l_a-k_a)\right] \mathrm{d}^N\phi \, \mathrm{d}^N r \\ &= \prod_{a=1}^{N} (\delta_{k_a,l_a})\ \pi^N \int \delta\Big(\sum_{a=1}^{N} u_a - 1\Big) \prod_{a=1}^{N} u_a^{k_a} \mathrm{d}u_a \\ &= \delta_{k,l}\, \frac{2\pi^N \prod\limits_{a=1}^{N} k_a!}{\Big(\sum\limits_{a=1}^{N} k_a \ + N - 1\Big)!}, \end{aligned} \tag{5.20}$$

where we substituted $u_a = r_a^2$. We observe that the integral is non-zero only if the power vectors k and l are equal in each component. This is the equivalent to the evenness condition (5.4) on the charge that we encountered in the Ising model. Let's now rewrite the partition function using this integral:

$$Z = \sum_{\{n\}} \left(\prod_{\langle x,y \rangle} \prod_{a,b=1}^{N} \frac{\beta^{n_{xy}^{ab}}}{n_{xy}^{ab}!} \right) \left(\prod_{x} Q\big(q(x), p(x)\big) \right), \tag{5.21}$$

with auxiliary ‘‘outflow’’ and ‘‘inflow’’ vector fields q and p, respectively, given component-wise at each site by

$$q_a(x) = \sum_{\mu=1}^{D} \sum_{b=1}^{N} \big(n_{x(x+\hat{\mu})}^{ab} + n_{(x-\hat{\mu})x}^{ba} \big), \quad p_a(x) = \sum_{\mu=1}^{D} \sum_{b=1}^{N} \big(n_{x(x+\hat{\mu})}^{ba} + n_{(x-\hat{\mu})x}^{ab} \big). \tag{5.22}$$

Note the nifty similarities to eqs. (5.3–5.5)! The outflow $q(x)$ is the combination of the row sum of the bond matrix in positive lattice direction plus the column sum in negative direction, and the inflow $p(x)$ is just the transposed analogon. Of course their naming is purely conventional, we could just as well call q the inflow.

Z is non-zero if and only if inflow and outflow are balanced on each site. This is the closed path constraint with oriented loops for the $\mathbb{C}\mathrm{P}^{N-1}$ model. Especially in the context of *oriented* loops, the CP constraint is often also interpreted as ‘‘zero divergence’’, ‘‘flux conservation’’ or ‘‘current conservation’’. On each lattice site, any incoming line has an outgoing counterpart.

5.2.2 The Algorithm

For the derivation of the algorithm, we proceed analogously to the Ising model above. Using two sites at which the CP constraint $q \equiv p$ is broken, we will transform the system from one CP configuration into another and thereby sample Z, using standard Metropolis rules. In the process, the movement of the worm will sample g.

Let's first consider the translation-invariant connected Green's function

$$\begin{aligned} G_c(x_\mathrm{h} - x_\mathrm{t}) &= \left\langle |\bar{z}(x_\mathrm{h}) \cdot z(y_t)|^2 \right\rangle - \frac{1}{N} \\ &= \frac{1}{Z} \int |\bar{z}(x_\mathrm{h}) \cdot z(x_\mathrm{t})|^2 \exp(-S) \prod_x \mathrm{D}z(x) - \frac{1}{N} \\ &=: \frac{1}{Z} g(x_\mathrm{h} - x_\mathrm{t}) - \frac{1}{N}. \end{aligned} \tag{5.23}$$

We now high-temperature-expand g and observe that it can be decomposed into a sum of N^2 components g^{ab} that are labeled by a flavor pair ab:

$$\begin{aligned} g(x_\mathrm{h} - x_\mathrm{t}) &= \int \Big[\sum_{a,b=1}^{N} \big(\bar{z}_a(x_\mathrm{h})\, z_b(x_\mathrm{h})\big)\big(\bar{z}_b(x_\mathrm{t})\, z_a(x_\mathrm{t})\big) \Big] \prod_{\langle x,y \rangle} \prod_{c,d=1}^{N} \sum_{n_{xy}^{cd}=0}^{\infty} \Big[\ldots \Big] \prod_x \mathrm{D}z(x) \\ &= \sum_{a,b=1}^{N} \int \big(\bar{z}_a(x_\mathrm{h})\, z_b(x_\mathrm{h})\big)\big(\bar{z}_b(x_\mathrm{t})\, z_a(x_\mathrm{t})\big) \prod_{\langle x,y \rangle} \prod_{c,d=1}^{N} \sum_{n_{xy}^{cd}=0}^{\infty} \Big[\ldots \Big] \prod_x \mathrm{D}z(x) \\ &=: \sum_{a,b=1}^{N} g^{ab}(x_\mathrm{h} - x_\mathrm{t}), \end{aligned} \tag{5.24}$$

where [...] is just the same expansion expression as in eq. (5.19). The core statement is the same as for the simple Ising model. We have found an expression for the Green's function that essentially differs from Z at only two sites: the head x_h and the tail x_t. For each pair ab, g^{ab} carries an additional "source" factor $\bar{z}_a\, z_b$ at the head and a "sink" factor $\bar{z}_b\, z_a$ at the tail. They increase the power of the corresponding expansion terms by one each. Their presence allows us to violate the zero-flux contraint $q(x) = p(x)$ at those two sites during the updates. Again, the terms "head" and "tail", or "source" and "sink", are merely conventional and interchangeable, as long as they are used consistently.

Eventually, a worm will operate in a single matrix element ab only, sampling g^{ab} by shifting $\bar{z}_a\, z_b$ and $\bar{z}_b\, z_a$ factors around. A new index pair cd will be proposed uniformly and accepted with correct Metropolis probability whenever a new worm is generated, i.e. in the "move" step of the WA. That newly created worm will then sample g_{cd}, and so on. In the end, we will sum up $g = \sum_{ab} g^{ab}$ to obtain the full correlation.

Let's define the weight integral

$$I(k) = \frac{2\pi^N \prod_{a=1}^{N} k_a!}{\Big(\sum_{a=1}^{N} k_a + N - 1 \Big)!}, \quad k \in \mathbb{N}_0^N, \tag{5.25}$$

such that $Q(q,p) = \delta_{q,p} I(q)$. Let's further assume that we have chosen a worm label $ab \in \{1,2,...,N\}^2$, i.e. consider the case where we sample a single component g^{ab} of the Green's function with a worm whose head and tail are $\bar{z}_a(x_\mathrm{h})\, z_b(x_\mathrm{h})$ and $\bar{z}_b(x_\mathrm{t})\, z_a(x_\mathrm{t})$, respectively. For g^{ab}, the zero-flux contraint $q \equiv p$ is modified into $k \equiv l$ with

$$\begin{aligned} k(x) &= q(x) + \delta_{x,x_\mathrm{h}} \hat{a} + \delta_{x,x_\mathrm{t}} \hat{b}, \\ l(x) &= p(x) + \delta_{x,x_\mathrm{h}} \hat{b} + \delta_{x,x_\mathrm{t}} \hat{a}. \end{aligned} \tag{5.26}$$

Here, $\hat{a}$ is the ath canonical unit vector in N dimensions. Accordingly, $\prod_x I\big(q(x)\big)$ in eq. (5.21) becomes $\prod_x I\big(k(x)\big)$ for g^{ab}:

$$g^{ab}(x_\mathrm{h} - x_\mathrm{t}) = \sum_{\{n\}} \left(\prod_{\langle x,y \rangle} \prod_{c,d=1}^{N} \frac{\beta^{n_{xy}^{cd}}}{n_{xy}^{cd}!} \right) \left(\prod_x \delta_{k(x),l(x)} I\big(k(x)\big) \right) \tag{5.27}$$

This expression readily provides us with the ratios of Boltzmann weights as acceptance probability for the Metropolis updates. Suppose the head is located at site x and will be shifted in lattice dimension μ to site $(x \pm \hat{\mu})$. We need to distinguish four different possible changes in bond configuration with their respective impacts on k and l:

- **Increase $n^{ab}_{x(x+\hat{\mu})}$:**
  - $q(x) \to q(x) + \hat{a}$, which replaces the leaving head $(-\delta_{x,x_\mathrm{h}} \hat{a})$ in k
  - $p(x) \to p(x) + \hat{b}$, which replaces the leaving head $(-\delta_{x,x_\mathrm{h}} \hat{b})$ in l
  - $q(x+\hat{\mu}) \to q(x+\hat{\mu}) + \hat{b}$, which compensates the new head $(+\delta_{x+\hat{\mu},x_\mathrm{h}} \hat{b})$ in l
  - $p(x+\hat{\mu}) \to p(x+\hat{\mu}) + \hat{a}$, which compensates the new head $(+\delta_{x+\hat{\mu},x_\mathrm{h}} \hat{a})$ in k
- **Increase $n^{ba}_{(x-\hat{\mu})x}$:**
  - $q(x) \to q(x) + \hat{a}$, which replaces the leaving head $(-\delta_{x,x_\mathrm{h}} \hat{a})$ in k
  - $p(x) \to p(x) + \hat{b}$, which replaces the leaving head $(-\delta_{x,x_\mathrm{h}} \hat{b})$ in l
  - $q(x-\hat{\mu}) \to q(x-\hat{\mu}) + \hat{b}$, which compensates the new head $(+\delta_{x-\hat{\mu},x_\mathrm{h}} \hat{b})$ in l
  - $p(x-\hat{\mu}) \to p(x-\hat{\mu}) + \hat{a}$, which compensates the new head $(+\delta_{x-\hat{\mu},x_\mathrm{h}} \hat{a})$ in k
- **Decrease $n^{ba}_{x(x+\hat{\mu})}$:**
  - $q(x) \to q(x) - \hat{b}$, which compensates the leaving head $(-\delta_{x,x_\mathrm{h}} \hat{b})$ in l
  - $p(x) \to p(x) - \hat{a}$, which compensates the leaving head $(-\delta_{x,x_\mathrm{h}} \hat{a})$ in k
  - $q(x+\hat{\mu}) \to q(x+\hat{\mu}) - \hat{a}$, which is replaced by the new head $(+\delta_{x+\hat{\mu},x_\mathrm{h}} \hat{a})$ in k
  - $p(x+\hat{\mu}) \to p(x+\hat{\mu}) - \hat{b}$, which is replaced by the new head $(+\delta_{x+\hat{\mu},x_\mathrm{h}} \hat{b})$ in l
- **Decrease $n^{ab}_{x(x-\hat{\mu})}$:**
  - $q(x) \to q(x) - \hat{b}$, which compensates the leaving head $(-\delta_{x,x_\mathrm{h}} \hat{b})$ in l
  - $p(x) \to p(x) - \hat{a}$, which compensates the leaving head $(-\delta_{x,x_\mathrm{h}} \hat{a})$ in k
  - $q(x-\hat{\mu}) \to q(x-\hat{\mu}) - \hat{a}$, which is replaced by the new head $(+\delta_{x-\hat{\mu},x_\mathrm{h}} \hat{a})$ in k
  - $p(x-\hat{\mu}) \to p(x-\hat{\mu}) - \hat{b}$, which is replaced by the new head $(+\delta_{x-\hat{\mu},x_\mathrm{h}} \hat{b})$ in l

Note that the index pair of the bond matrix element which is changed depends on the direction the head is shifted (positive or negative) as well as the change itself (± 1)! $k(x)$ remains unchanged in the first two cases where the bond matrix element is incremented, while $k(x \pm \hat{\mu})$ remains unchanged in the latter two. The Metropolis acceptance ratios for the four changes are hence given by

$$\begin{aligned}
R_{\mathrm{sh}}^{\mathrm{inter}}(x \to x+\hat{\mu},\, n^{ab}_{x(x+\hat{\mu})} \to n^{ab}_{x(x+\hat{\mu})}+1) &= \frac{\beta}{n^{ab}_{x(x+\hat{\mu})}+1}\frac{I\big(q(x+\hat{\mu})+\hat{a}+\hat{b}\big)}{I\big(q(x+\hat{\mu})\big)},\\
R_{\mathrm{sh}}^{\mathrm{inter}}(x \to x-\hat{\mu},\, n^{ba}_{(x-\hat{\mu})x} \to n^{ba}_{(x-\hat{\mu})x}+1) &= \frac{\beta}{n^{ba}_{(x-\hat{\mu})x}+1}\frac{I\big(q(x-\hat{\mu})+\hat{a}+\hat{b}\big)}{I\big(q(x-\hat{\mu})\big)},\\
R_{\mathrm{sh}}^{\mathrm{inter}}(x \to x+\hat{\mu},\, n^{ba}_{x(x+\hat{\mu})} \to n^{ba}_{x(x+\hat{\mu})}-1) &= \frac{n^{ba}_{x(x+\hat{\mu})}}{\beta}\frac{I\big(q(x)-\hat{b}\big)}{I\big(q(x)+\hat{a}\big)},\\
R_{\mathrm{sh}}^{\mathrm{inter}}(x \to x-\hat{\mu},\, n^{ab}_{(x-\hat{\mu})x} \to n^{ab}_{(x-\hat{\mu})x}-1) &= \frac{n^{ab}_{(x-\hat{\mu})x}}{\beta}\frac{I\big(q(x)-\hat{b}\big)}{I\big(q(x)+\hat{a}\big)},
\end{aligned} \tag{5.28}$$

if, and this is important to see, the head shift is neither the worm's first ($x \neq x_{\mathrm{t}}$) nor last ($x \pm \hat{\mu} \neq x_{\mathrm{t}}$). $R_{\mathrm{sh}}^{\mathrm{inter}}$ is only valid for *intermediate* head shifts, hence the superscript. For the *first* shift, the tail contributes an additional $\delta_{x,x_{\mathrm{t}}}\hat{b}$ to $k(x = x_{\mathrm{t}})$, thus the ratios become

$$\begin{aligned}
R_{\mathrm{sh}}^{\mathrm{first}}(x \to x+\hat{\mu},\, n^{ab}_{x(x+\hat{\mu})} \to n^{ab}_{x(x+\hat{\mu})}+1) &= R_{\mathrm{sh}}^{\mathrm{inter}}(...),\\
R_{\mathrm{sh}}^{\mathrm{first}}(x \to x-\hat{\mu},\, n^{ba}_{(x-\hat{\mu})x} \to n^{ba}_{(x-\hat{\mu})x}+1) &= R_{\mathrm{sh}}^{\mathrm{inter}}(...),\\
R_{\mathrm{sh}}^{\mathrm{first}}(x \to x+\hat{\mu},\, n^{ba}_{x(x+\hat{\mu})} \to n^{ba}_{x(x+\hat{\mu})}-1) &= \frac{n^{ba}_{x(x+\hat{\mu})}}{\beta}\frac{I\big(q(x)\big)}{I\big(q(x)+\hat{a}+\hat{b}\big)},\\
R_{\mathrm{sh}}^{\mathrm{first}}(x \to x-\hat{\mu},\, n^{ab}_{(x-\hat{\mu})x} \to n^{ab}_{(x-\hat{\mu})x}-1) &= \frac{n^{ab}_{(x-\hat{\mu})x}}{\beta}\frac{I\big(q(x)\big)}{I\big(q(x)+\hat{a}+\hat{b}\big)}.
\end{aligned} \tag{5.29}$$

During the *last* shift, the tail contributes an additional $\delta_{x,x_{\mathrm{t}}}\hat{b}$ to $k(x \pm \hat{\mu} = x_{\mathrm{t}})$:

$$\begin{aligned}
R_{\mathrm{sh}}^{\mathrm{last}}(x \to x+\hat{\mu},\, n^{ab}_{x(x+\hat{\mu})} \to n^{ab}_{x(x+\hat{\mu})}+1) &= \frac{\beta}{n^{ab}_{x(x+\hat{\mu})}+1}\frac{I\big(q(x+\hat{\mu})+\hat{a}+2\hat{b}\big)}{I\big(q(x+\hat{\mu})+\hat{b}\big)},\\
R_{\mathrm{sh}}^{\mathrm{last}}(x \to x-\hat{\mu},\, n^{ba}_{(x-\hat{\mu})x} \to n^{ba}_{(x-\hat{\mu})x}+1) &= \frac{\beta}{n^{ba}_{(x-\hat{\mu})x}+1}\frac{I\big(q(x-\hat{\mu})+\hat{a}+2\hat{b}\big)}{I\big(q(x-\hat{\mu})+\hat{b}\big)},\\
R_{\mathrm{sh}}^{\mathrm{last}}(x \to x+\hat{\mu},\, n^{ba}_{x(x+\hat{\mu})} \to n^{ba}_{x(x+\hat{\mu})}-1) &= R_{\mathrm{sh}}^{\mathrm{inter}}(...),\\
R_{\mathrm{sh}}^{\mathrm{last}}(x \to x-\hat{\mu},\, n^{ab}_{(x-\hat{\mu})x} \to n^{ab}_{(x-\hat{\mu})x}-1) &= R_{\mathrm{sh}}^{\mathrm{inter}}(...).
\end{aligned} \tag{5.30}$$

But this is not the whole story. In the case where $a = b$, i.e. if the worm is "diagonal", the Kronecker deltas for head and tail in the definition of k and l (eq. (5.26)) need no compensation by a change in bond state when shifting the head around. The constraint $k \equiv l$ is automatically fulfilled if $q \equiv p$. An ergodic sampling of the diagonal entries of g must therefore also include head shifts which don't modify the diagonal bond matrix elements n^{aa}_{xy}. The ratios of new over old weight for such "no-change" shifts are easily

found to be

$$\begin{aligned} R^{\text{first}}_{\text{ncsh}}(x \to x \pm \hat{\mu},\, a = b) &= \frac{I\big(q(x) + \hat{a}\big)}{I\big(q(x) + 2\hat{a}\big)} \frac{I\big(q(x \pm \hat{\mu}) + \hat{a}\big)}{I\big(q(x \pm \hat{\mu})\big)}, \\ R^{\text{inter}}_{\text{ncsh}}(x \to x \pm \hat{\mu},\, a = b) &= \frac{I\big(q(x)\big)}{I\big(q(x) + \hat{a}\big)} \frac{I\big(q(x \pm \hat{\mu}) + \hat{a}\big)}{I\big(q(x \pm \hat{\mu})\big)}, \\ R^{\text{last}}_{\text{ncsh}}(x \to x \pm \hat{\mu},\, a = b) &= \frac{I\big(q(x)\big)}{I\big(q(x) + \hat{a}\big)} \frac{I\big(q(x \pm \hat{\mu}) + 2\hat{a}\big)}{I\big(q(x \pm \hat{\mu}) + \hat{a}\big)}. \end{aligned} \tag{5.31}$$

When a worm closes ($x_\text{h} = x_\text{t}$), a new CP configuration contributing to Z is reached. For the Ising model, where the g sector was the same as the Z sector due to the 2-periodicity of the site integrals Q, the weight $w(\{n\})$ of a CP configuration resulting from the removal of the source and sink terms was exactly unity. This no longer holds here. k and l are reduced to q and p again, respectively, which modifies the argument where the site integral is evaluated, yielding the configuration weight

$$w(\{n\}) = \frac{1}{N^2} \frac{Q\big(q(x), p(x)\big)}{Q\big(k(x), l(x)\big)} = \frac{1}{N^2} \frac{I\big(q(x)\big)}{I\big(q(x) + \hat{a} + \hat{b}\big)}, \quad x = x_\text{h} = x_\text{t}. \tag{5.32}$$

The prefactor $1/N^2$ accounts for the fact that N^2 worms are needed to sample all components ab. This weight serves as the increment for the statistics of the partition function as well as a prefactor for any observable A (cf. section 5.2.4) measured while sampling Z, via

$$\langle A \rangle_{CP} = \frac{1}{Z} \sum_{\{n\} \in CP}^{\text{MC}} A(\{n\})\, w(\{n\}), \quad Z = \sum_{\{n\} \in CP}^{\text{MC}} w(\{n\}), \tag{5.33}$$

where $\sum^{\text{MC}}$ is the Monte Carlo sum of sampled configurations.

What remains to be specified is the Metropolis ratio for the ‘‘move’’ step in the WA, i.e. for the transition of one worm into another one. Here, both the change of location and the change of the flavor label ab need to be taken into account. The ratio of weights for moving a closed worm from site $x = x_\text{h} = x_\text{t}$ to site x_m and at the same time reflavoring it from ab to cd is

$$\begin{aligned} R_\text{mv}(x = x_\text{h} = x_\text{t} \to x_\text{m},\, ab \to cd) &= \frac{Q\big(q(x), p(x)\big)}{Q\big(k(x), l(x)\big)} \frac{Q\big(k(x_\text{m}), l(x_\text{m})\big)}{Q\big(q(x_\text{m}), p(x_\text{m})\big)} \\ &= \frac{I\big(q(x)\big)}{I\big(q(x) + \hat{a} + \hat{b}\big)} \frac{I\big(q(x_\text{m}) + \hat{c} + \hat{d}\big)}{I\big(q(x_\text{m})\big)}. \end{aligned} \tag{5.34}$$

This corresponds to the removal of the source $\bar{z}_a\, z_b$ and the sink $\bar{z}_b\, z_a$ from site x and placing a new source $\bar{z}_c\, z_d$ and sink $\bar{z}_d\, z_c$ at site x_m. R_mv cancels to unity for the Ising model, which is why the move is always accepted there.

With the above considerations, we devise the worm algorithm as follows:

1. Allocate a $V \times N \times N$ array for the components g^{ab} and zero it. Set $Z = 0$.
2. Assign each of the DV bonds an $N \times N$ matrix of non-negative integers and initialize them such that $q \equiv p$, e.g. all elements zero.

3. Select any site x_m and set $x_\mathrm{h} = x_\mathrm{t} = x_\mathrm{m}$. Start with any $a, b \in \{1, 2, ..., N\}$.
4. Pick one of the $2D$ lattice directions $\pm\hat{\mu}$ at random.
5. If $a = b$, then with probability P_nc (see below) propose a head shift without change of bond state. In this case, set $x_\mathrm{h} = x_\mathrm{h} \pm \hat{\mu}$ with probability $P_\mathrm{ncsh} = \min\{1, R_\mathrm{ncsh}\}$ and continue with step 8.
6. Propose to increase or decrease the bond state by 1 with probability 1/2. Together with the decision $+\hat{\mu}$ or $-\hat{\mu}$ this identifies the four possible bond modifications.
7. Perform the change with probability $P_\mathrm{sh} = \min\{1, R_\mathrm{sh}\}$. If accepted, set $x_\mathrm{h} = x_\mathrm{h} \pm \hat{\mu}$.
8. Increment $g^{ab}(x_\mathrm{h} - x_\mathrm{t})$ by 1, regardless of whether the shift was accepted or not.
9. If $x_\mathrm{h} \neq x_\mathrm{t}$, continue with step 4.
10. Increment Z by $w(\{n\})$ and measure any desired quantity $A(\{n\})\, w(\{n\})$.
11. With probability 1/2 (cf. ref. [11]), draw a new random flavor pair and a new location x_m for head and tail, and accept with probability $P_\mathrm{mv} = \min\{1, R_\mathrm{mv}\}$. Go to step 4.

When enough statistics have been collected, the simulation cycle is terminated. The full connected Green's function is retrieved using eqs. (5.23) and (5.24), and the energy and magnetic susceptibility are derived from it in the usual way as described in section 2.2.

It's worth mentioning that the WA operates with reals and integers only, as can be seen without difficulty. No complex numbers are involved, even though the original lattice model is completely complex-valued. This astonishing property stands in sharp contrast to cluster algorithms.

The "no-change" probability P_nc with which we decide to propose a shift that leaves the bonds untouched if the worm is diagonal, can be chosen freely in]0,1[. We observed no impact on the simulation results if P_nc was varied. The decorrelation power is expected to vary significantly, though. Furthermore, ergodicity and detailed balance are also maintained if this special shift is modified to a "heatbath", where a randomly drawn site x_c replaces the role of the nearest neighbor $(x \pm \hat{\mu})$ as the candidate site for the shift (or "jump" in that case). This heatbath version of the algorithm has been checked to yield the same results as well, and is also expected to lead to different autocorrelation times. A rigorous study in these regards is still pending.

An interesting feature of the $\mathbb{C}\mathrm{P}^{N-1}$ worm is that head and tail cannot be located from the bond configuration alone if $a = b$, which is rather peculiar. Z also includes "open" paths labeled aa, which are broken open by the no-change shifts. The "CP" constraint $q \equiv p$ is fulfilled for arbitrary values in the diagonal elements of the bond matrices, i.e. they do not necessarily need to form closed lines. In other words, when g^{aa} is sampled with a diagonal worm, any open-line configuration (whether $x_\mathrm{h} = x_\mathrm{t}$ or not) carries a non-zero weight in Z and may be used to measure observables from. In principle, one could even consider interrupting the head shifting before $x_\mathrm{h} = x_\mathrm{t}$ and start another worm elsewhere. Neither of these measures is required for ergodicity, though. In any case, the set of configurations contributing to Z, which we continue denoting CP here, is strictly speaking not exclusive to closed paths in the original sense.

5.2.3 Efficient Implementation

A predominant portion of the simulation code comprises the evaluation of site integrals ratios $I(...)/I(...)$ with only small differences in the argument between numerator and denominator. While all ratios nicely cancel for the Ising model, the site integrals can be tabulated for many other models prior to the simulation [11]. This is rather useless in our case. Even for moderate N, the dimensionality of such a lookup table is presumably large enough to make the costs for finding the memory location of the table entry more expensive than recalculating the ratio of integrals in the first place, if the table even fits into memory at all. Instead, we can exploit the identities

$$\frac{I\big(q(x)+\hat{a}\big)}{I\big(q(x)\big)} = \frac{q_a(x)+1}{\tilde{q}(x)} \quad \text{and} \quad \frac{I\big(q(x)-\hat{a}\big)}{I\big(q(x)\big)} = \frac{\tilde{q}(x)-1}{q_a(x)}, \tag{5.35}$$

where

$$\tilde{q}(x) = \sum_{c=1}^{N} q_c(x) + N. \tag{5.36}$$

Using trivial expansions of the form

$$\frac{I\big(q(x)+\hat{a}+\hat{b}\big)}{I\big(q(x)\big)} = \frac{I\big(q(x)+\hat{a}\big)}{I\big(q(x)\big)} \frac{I\big(\breve{q}(x)+\hat{b}\big)}{I\big(\breve{q}(x)\big)}, \quad \breve{q}(x) = q(x)+\hat{a}, \tag{5.37}$$

all ratios (5.28–5.34) occurring in the algorithm can be decomposed into products of (5.35) or their reciprocals, which are very cheap to compute. In practice, the integral formula (5.25) doesn't ever need to be actually evaluated.

In principle, neither of q and p vectors need to be kept track of. They can be reconstructed from the bond matrices via eq. (5.22) at any time. In a practical implementation, however, one is of course advised to avoid this recalculation by storing q and $\tilde{q}$ at each site anyway. The update to q and $\tilde{q}$ corresponding to the bond matrix modification in step 7 of the algorithm can be readily seen from the four allowed bond changes disclosed further above. The evaluation of any integral ratio is then possible in $\mathcal{O}(1)$ time, i.e. independent of N. The p vectors are not needed at all during the simulation.

5.2.4 Direct Estimators

In the exact same manner as for the Ising model, the energy of the constrained $\mathbb{C}\mathrm{P}^{N-1}$ loop gas can be written as the sum of all bond states:

$$\langle E \rangle = -\frac{1}{\beta} \Big\langle \sum_{\langle x,y \rangle} \sum_{a,b=1}^{N} n_{xy}^{ab} \Big\rangle_{CP} \tag{5.38}$$

The fluctuative-dissipative formula (5.15) for the heat capacity also holds for the $\mathbb{C}\mathrm{P}^{N-1}$ model, with $\langle E \rangle$ from eq. (5.38).

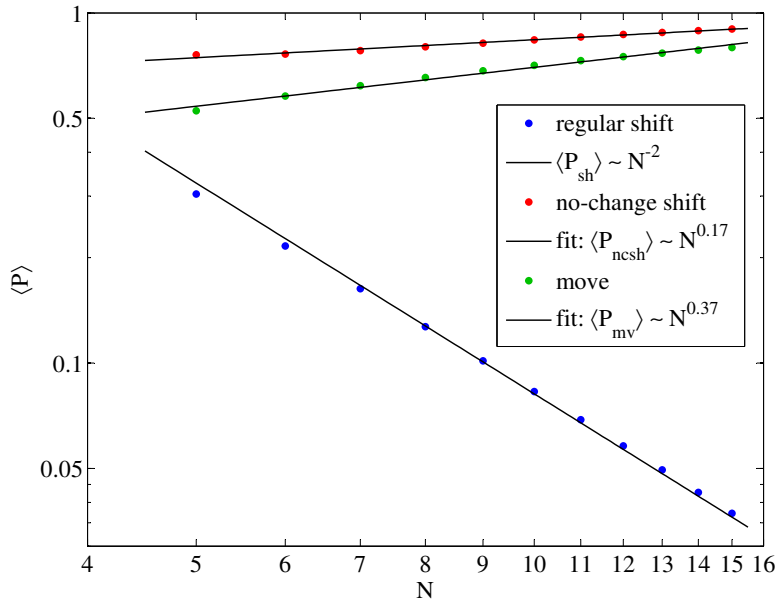
The topological charge has recently been identified in the loop gas formulation by Wolff [13]. It is rather non-trivial to extract and reportedly involves the infamous sign problem. Numerical studies are still pending. However, the presence of *oriented* loops (together with periodic boundaries) grants easy direct access to another topological mode. From the difference of outflow and inflow per flavor along each bond, we define the *winding number* $W \in \mathbb{Z}^{D \times N}$ of a CP configuration component-wise by

$$W_{\mu,a} = \frac{1}{L} \sum_{x} \sum_{b=1}^{N} \big(n_{x(x+\hat{\mu})}^{ab} - n_{x(x+\hat{\mu})}^{ba} \big). \tag{5.39}$$

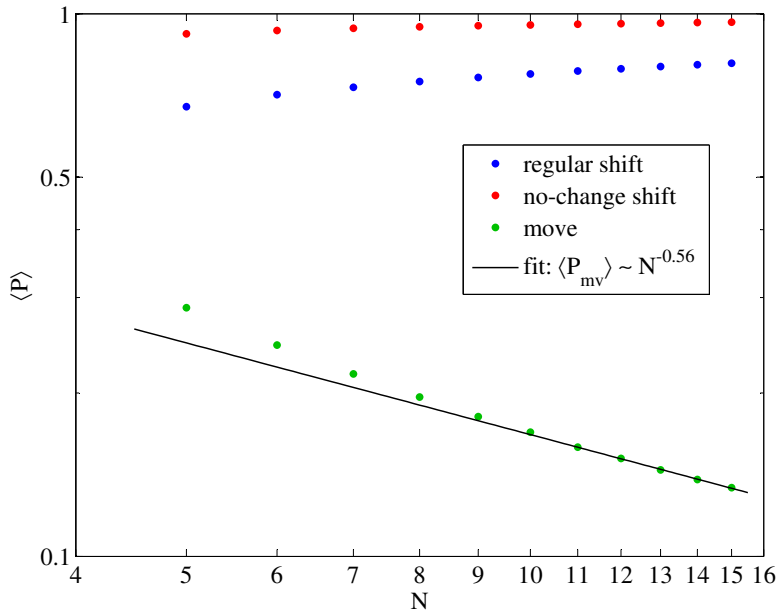
Pictorially speaking, it is the net number of loops spanning the D-torus, for each component and lattice dimension. The worm algorithm holds the pleasant power of modifying these winding numbers.

5.2.5 Scaling with the Flavor Number

The number of degrees of freedom (bond states) in the loop gas representation of the $\mathbb{C}\mathrm{P}^{N-1}$ model is DVN^2. Consequently, the CPU time needed by the worm algorithm grows at least like the square of N. But there is also an N-dependency hidden in the Metropolis probabilities. In particular, the acceptance probability of a regular head shift, P_{sh}, is critical, as it is the predominant one. To find the overall scaling behavior of the algorithm, we measured the average probabilities on a $L = 8$ square grid (fig. 5.3). When β is fixed, the average acceptance probability for a head shift diminishes like $1/N^2$ (fig. 5.3(a)), making the overall CPU time required scale like $\mathcal{O}(N^4)$. However, the case where ξ is fixed instead is more important. There, we observe a stable shift probability (fig. 5.3(b)). The overall scaling will still be slightly worse than $\mathcal{O}(N^2)$ though, due to the moderate decline of the move probability P_{mv}.



(a) At constant $\beta = 4.8483$ (i.e. varying ξ), P_{sh} diminishes like $1/N^2$. The other exponents serve only as a rough approximation in the shown range; clearly the probabilities approach 1 for large N.



(b) At constant ξ, i.e. β chosen such that $L/\xi = 2$ according to the two-loop scaling prediction (eq. (4.12)), no clear power law is observed. The shift acceptance probabilities slowly approach 1 with increasing N, while P_{mv} declines moderately.

Figure 5.3: Log-log plots of average Metropolis acceptance probabilities after equilibration, for a $L = 8$ sample grid. The key probability is that of a regular head shift (blue). Errors are smaller than the symbols.

5.2.6 Ergodicity Problem

With the above worm algorithm, we obtain results that are consistent with those from the single-flip and single-cluster algorithms for $N = 2$, see section 5.2.7. Surprisingly, though, we haven't been able to reproduce the correct results for larger N, not even for small lattices and rather long runs with, say, $\mathcal{O}(10^9)$ worms. To illustrate this, lets consider the (averaged) results of $n = 12$ independent simulations at $N = 4$, $L = 12$, $\beta = 5.0854$, with $u = 2\times10^9$ worms after equilibration. The number of worms between measurements was $s = 100$. The expected (correct) outcome is that of the first row in table 4.2. But instead, we obtain the following:

- from the average bond occupation: $\langle E\rangle/V = -1.295352(11)$
- from the nearest neighbor correlation: $\langle E\rangle/V = -1.29541(9)$
- $\langle\chi\rangle/V = 0.25624(4)$

These values are obviously wrong. Yet the two energy measurements match, which indicates that our worm updates sample only part of the full phase space. A possible cause of ergodicity failure becomes apparent when we look at the winding numbers from one of these simulations:

$$\langle|W|\rangle = 10^{-3}\begin{bmatrix}3.967 & 3.899 & 3.943 & 11.789\\ 3.993 & 3.963 & 3.968 & 11.904\end{bmatrix} \tag{5.40}$$

They are the same within statistical errors in both lattice directions (rows), but not for all components (columns). For symmetry, the true equilibrium values are of course the same in all components. When histogramming the component label ab the worms adopt in step 11 of the algorithm, we observed that for $N > 2$, the worms easily get "trapped" in the sense that there exists a flavor c such that the worms keep their labels within ac or cb (a row or column in the $N\times N$ matrix of label histograms) for a very long time, resulting in an imbalance between the winding numbers in eq. (5.40). Also, the statistics for the different components of g are affected in the same way, i.e. they remain perseveringly imbalanced until very long. With increasing N, this lack of ergodicity seems to grow even.

To get an idea of the potential severity of this problem, we investigated the simulation time required to overcome the mentioned imbalance. More precisely, we measured the average maximum value for β/N we could reach in $L = 16$ square lattice simulations at different N, subject to the constraint that the histograms for aa get balanced within a fixed number of lattice sweeps. Let's denote the number of times a worm carrying the component label ab is chosen in step 11 of the algorithm by $h(a, b)$. For a given value of N, we varied β and determined, at which value (on average) we would find that

$$\frac{\max_a h(a,a)}{\min_a h(a,a)} = 1.01 \tag{5.41}$$

after precisely T lattice sweeps, without thermalizing beforehand, i.e. starting from the zeroed configuration. (One lattice sweep is defined as DVN^2 head shift proposals, or increments to the statistics of g.) In other words, we determined the maximum values for β/N, for which the histogram imbalance is overcome up to a percent within T lattice sweeps. The results for $T = 10^4, 10^5, 10^6$ are shown in figure 5.4. These results quantify

the depth of the row/column ergodicity trap, providing an immediate prediction for the physical size L/ξ of the system one can simulate to a certain precision within a given amount of time, via the scaling law (4.12). The main message from figure 5.4 is actually the range of values rather than the behavior with N. For instance, a worm simulation of $\mathbb{C}\mathrm{P}^3$ at $\beta/4 = 0.5214(9)$ on a lattice as small as $L = 16$ will balance to 1% precision only after $T = 10^4$ lattice sweeps (green triangle at $N = 4$). The correlation length reached by such a simulation is as small as $\xi \approx 0.034$, a value far away from the desired regime. Allowing for a runtime of $T = 10^6$ sweeps extends the accessible length scale to no more than $\xi \approx 1.5$ ($\beta/4 = 1.06585(15)$, blue circle) in that example.

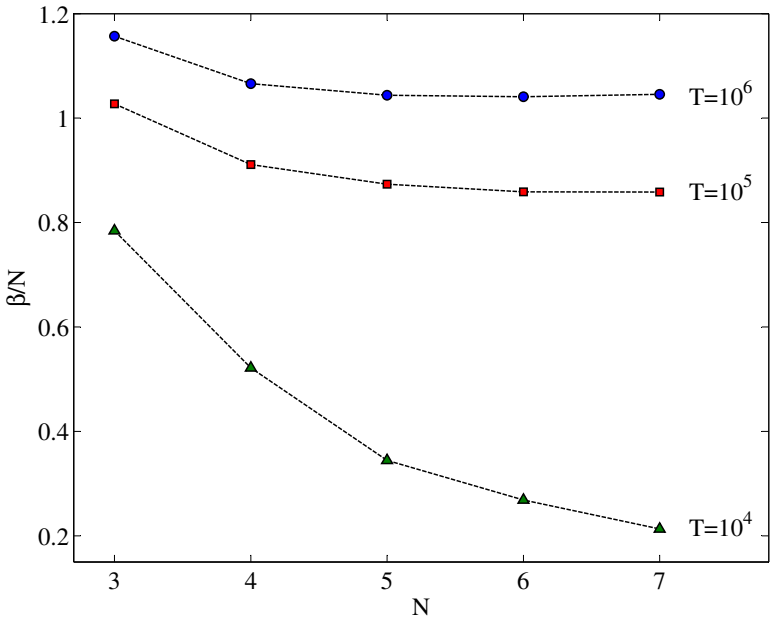


Figure 5.4: Accessible values for β/N at various N for a $L = 16$ lattice. The dashed lines are isolines for the number of lattice sweeps T required to sample the full flavor space evenly to 1% precision. A detailed description is given in the text. Errors are smaller than the symbols.

Non-ergodicity for large N has been reported also for the well-established spin algorithms by Vicari [64], thus it is not a new phenomenon. It has also been realized long ago that the high-temperature expansion and high-N expansion don't commute [65, 66]. However, the issues that are encountered with the worm algorithm certainly are of a different level of severity, as it fails already at $N = 3$. Using the *Simulated Tempering* method proposed by Marinari and Parisi [67], Vicari has been able to go to quite large N. In this method, the coupling β is varied while the system is kept at equilibrium at the same time. It has not yet been examined whether a similar approach could be successful also for the $\mathbb{C}\mathrm{P}^{N-1}$ WA. Aside from Simulated Tempering, there are other methods known to deal with very rough free energy surfaces. Umbrella Sampling [68] is certainly worth considering, as are the numerous histogram methods [69–71]. Or, just like in past simulations of the $\mathbb{C}\mathrm{P}^{N-1}$ model [8, 41, 57, 64, 72], remedy could also be found in overrelaxation, which may allow to tunnel between the separated vacua.

5.2.7 Numerical Results and Dynamic Behavior for $N=2$

In this section we report simulation results for the $\mathbb{C}\mathrm{P}^1$ model, which is known to be equivalent to the classical Heisenberg model. Table 5.1 lists high-precision numerical data for a series of simulations on square lattices up to $L = 128$. $P_{\mathrm{nc}} = 1/2$ and regular no-change head shifts were used (i.e. no heatbath). For benchmarking, we chose to simulate the exact same systems (β, L) as with the single-cluster algorithm on the O(3) model. The obtained values for the appropriately rescaled energy, specific heat and magnetic susceptibility are thus directly comparable to table 4.1. The tabulated parameters are explained there. Consistency within statistical errors is found in all cases. Note that the correlation length is measured with the zero-momentum definition (3.14) here, which is not identical to $\xi_{2\mathrm{nd}}$ from table 4.1. Additional unlisted simulation runs have also been carried out to assert consistency between the worm and single-cluster algorithms on $\mathbb{C}\mathrm{P}^1$, which is confirmed.

Table 5.1 also contains results for the winding numbers and their square. In favor of a compact notation, we write

$$|W| = \frac{1}{DN} \sum_{\mu=1}^{D} \sum_{a=1}^{N} |W_{\mu,a}| \tag{5.42}$$

and

$$W^2 = \frac{1}{DN} \sum_{\mu=1}^{D} \sum_{a=1}^{N} W_{\mu,a}^2 \tag{5.43}$$

with $W_{\mu,a}$ from (5.39). W^2 should be directly proportional to the spin stiffness ρ_s [73]. Furthermore, we use the notation

$$\langle E_G \rangle = -\sum_{\langle x,y \rangle} G(x,y) = -\frac{1}{Z} \sum_{\mu} g(\hat{\mu}) \tag{5.44}$$

to clearly distinguish the energy obtained from the Green's function, E_G, from the one obtained from the bond occupation (5.38), E without subscript.

The integrated autocorrelation times from table 5.1 are visualized in figure 5.5, where they are logarithmically drawn against the zero-momentum correlation lengths to reveal the dynamic critical behavior of the worm algorithm. This includes the modes derived from the Green's function, E_G and χ, whose autocorrelation times $\tau_{E_G}^{\mathrm{int}}$ and $\tau_{\chi}^{\mathrm{int}}$ were determined using the binning strategy mentioned towards the end of section 5.1.2. These two are well shorter than a single lattice sweep (1 sweep $= DVN^2$ head shift proposals) and may hence be subject to some inaccuracy induced by the very short integration domain ($M = 2$ mostly, cf. section 3.1).

Table 5.1: Numerical results for the $\mathbb{C}\mathrm{P}^1$ model obtained with the WA. The systems are chosen to be identical to our single-cluster simulations of the equivalent O(3) model (table 4.1). For direct comparability with the latter, some values are rescaled as indicated in the header.

L	$\beta/2$	n	$u\times10^{-9}$	s	$T_\tau\times10^{-6}$	s_τ	$2E/V+2$
12	1.49950	30	15	150	8	1	-1.225599(5)
16	1.55300	30	10	150	8	1	-1.254432(4)
23	1.62000	30	4	150	8	1	-1.290660(5)
32	1.67623	30	2	200	8	1	-1.319601(7)
45	1.73215	30	1.2	250	8	1	-1.346585(6)
64	1.78951	32	0.6	300	8	1	-1.372244(5)
91	1.84646	32	0.3	350	8	1	-1.395697(7)
128	1.90133	32	0.15	400	8	1	-1.416567(6)

$2E_G/V+2$	C_V/V	$2\chi/V$	$\lvert W\rvert$	W^2
-1.225629(14)	1.5791(3)	0.352203(8)	0.29049(3)	0.32250(3)
-1.25441(2)	1.5727(3)	0.322514(13)	0.28700(3)	0.31732(4)
-1.29067(4)	1.5357(4)	0.29249(3)	0.28721(7)	0.31685(8)
-1.31961(5)	1.4927(11)	0.26823(4)	0.28640(16)	0.31539(19)
-1.34664(7)	1.4479(13)	0.24629(6)	0.2848(3)	0.3132(4)
-1.37228(8)	1.403(2)	0.22707(9)	0.2863(5)	0.3147(6)
-1.39582(15)	1.368(5)	0.20984(17)	0.2900(5)	0.3189(14)
-1.41683(18)	1.331(5)	0.1954(2)	0.290(2)	0.319(3)

ξ	τ_E^{int}	$\tau_{E_G}^{\mathrm{int}}$	τ_χ^{int}	τ_W^{int}	$\tau_{W^2}^{\mathrm{int}}$
5.99919(16)	0.3861(4)	0.11078(5)	0.09597(4)	14.77(7)	12.33(6)
7.9559(3)	0.3720(4)	0.10859(5)	0.09271(5)	16.67(10)	13.92(7)
11.4486(13)	0.3572(5)	0.10637(6)	0.08939(5)	19.02(16)	15.92(10)
15.944(3)	0.3461(5)	0.10460(6)	0.08687(6)	21.48(2)	18.02(15)
22.463(8)	0.3348(12)	0.10320(6)	0.08483(6)	24.4(3)	20.41(20)
32.069(16)	0.3216(12)	0.10166(9)	0.08276(9)	26.2(6)	22.2(4)
45.63(6)	0.3151(11)	0.10052(12)	0.08121(13)	30.2(1.1)	25.4(7)
64.40(12)	0.3079(11)	0.09945(14)	0.07979(16)	33(2)	28.0(1.2)

Prokof'ev & Svistunov reported shorter autocorrelation times for χ than for E already in their original paper [11]. Wolff observed the same for O(N) at various N [12] and $\mathbb{C}\mathrm{P}^3$ [13]. Our finding for $\mathbb{C}\mathrm{P}^1$ is identical. The slowest modes are $|W|$ and W^2, which are also the only ones not even speeding up with growing system size. We have fitted the pure power law $\tau^{\mathrm{int}}(L) = c\,\xi(L)^z$ to each mode (excluding the two smallest lattices) and found excellent agreement for $|W|$ and W^2 with χ^2_{red} close to one. The fit parameters are shown in the legend of figure 5.5. The most essential result is $z = 0.32(3)$ for the winding numbers and $z \approx 0$ for the standard observables E, χ. All relations are also consistent with logarithmic divergence, i.e. $\tau^{\mathrm{int}}(L) = \tau_0 + c\log[\xi(L)]$ with zero dynamic critical exponent, within two standard errors. We can safely summarize that the worm algorithm accelerates the $\mathbb{C}\mathrm{P}^1$ model, completely eliminating critical slowing down for the standard modes.

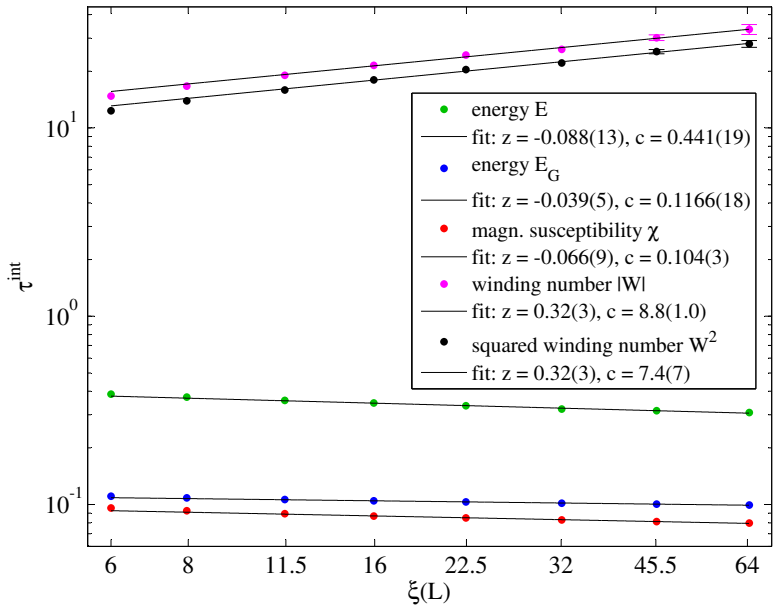


Figure 5.5: Log-log plot of integrated autocorrelation times for the WA applied to the 2D $\mathbb{C}\mathrm{P}^1$ model at constant physical size $L/\xi(L) \approx 2$. Where omitted, error bars (horizontal and vertical) are smaller than the dots. In the legend we give 95% confidence bounds for the fit parameters in brackets. The smallest two systems aren't used for fitting. χ^2_{red} takes the remarkably good values 1.3 and 2.2 for τ^{int}_W and $\tau^{\mathrm{int}}_{W^2}$, respectively.

5.2.8 Sampling the Connected Correlation

The disconnected piece $1/N$ of the connected Green's function precludes $G^{ab} = g^{ab}/Z$ from approaching zero at large separations on the diagonal ($a = b$). Instead, we have

$$\lim_{L\to\infty} G^{ab}(L/2) = \delta_{a,b}\frac{1}{N^2}, \tag{5.45}$$

i.e. the sampled diagonal correlation $\langle |z_a(x)\, z_a(y)|^2 \rangle$ is bounded by $1/N^2$ from below. A diagonal worm sampling this correlation therefore holds the power to shift its head

arbitrarily around the lattice before closing again. In order to circumvent this potential hindrance to efficiency, and hoping that also the row/column ergodicity problem described in section 5.2.6 would be alleviated, we considered modifying the probabilities in the worm algorithm such that it directly samples the *connected* correlation

$$G_c(x_\mathrm{h} - x_\mathrm{t}) = \frac{1}{Z}\sum_{a,b=1}^{N} g_c^{ab}(x_\mathrm{h} - x_\mathrm{t}), \quad g_c^{ab}(x_\mathrm{h} - x_\mathrm{t}) = g^{ab}(x_\mathrm{h} - x_\mathrm{t}) - \delta_{a,b}\frac{Z}{N^2}. \tag{5.46}$$

This modifies the site integral contribution to the Boltzmann weight:

$$I\big(k(x)\big) \to I\big(k(x)\big) - \delta_{a,b}\frac{1}{N^2} I\big(q(x)\big). \tag{5.47}$$

All Metropolis ratios need to be modified accordingly in the case where $a = b$. This can be done (and in fact we have done it), but the implied acceptance probabilities turn out to suffer from the sign problem, i.e. they can be negative, which makes the corresponding Metropolis scheme invalid. We won't give all weight ratios for this illegal algorithm here, as they become quite lengthy. Nevertheless, we note that they can all still be expressed in terms of eq. (5.35), such that I never requires evaluation. Furthermore, a bond state increment (decrement) now also affects the argument for $I(...)$ at site x ($x \pm \hat{\mu}$), because the change in bond state is compensated by the dislocation of the head in the first term of (5.47) only, but not in the second, inhibiting cancellation of new over old weight. To give an idea of what the modified acceptance ratios $\tilde{R}$ look like, we provide an assortment below.

$$\begin{aligned}
&\tilde{R}^{\mathrm{inter}}_{\mathrm{sh}}(x \to x+\hat{\mu},\, n^{ab}_{x(x+\hat{\mu})} \to n^{ab}_{x(x+\hat{\mu})} + 1) \\
&\quad= \frac{\beta}{n^{ab}_{x(x+\hat{\mu})}+1}\,\frac{I\big(q(x)+\hat{a}\big) - \delta_{a,b}\frac{1}{N^2} I\big(q(x)+\hat{a}\big)}{I\big(q(x)+\hat{a}\big) - \delta_{a,b}\frac{1}{N^2} I\big(q(x)\big)} \\
&\qquad\qquad \frac{I\big(q(x+\hat{\mu})+\hat{a}+\hat{b}\big) - \delta_{a,b}\frac{1}{N^2} I\big(q(x+\hat{\mu})+\hat{b}\big)}{I\big(q(x+\hat{\mu})\big) - \delta_{a,b}\frac{1}{N^2} I\big(q(x+\hat{\mu})\big)} \\
&\quad= \frac{\beta}{n^{ab}_{x(x+\hat{\mu})}+1}\left(1 - \frac{\delta_{a,b}}{N^2}\frac{I\big(q(x)\big)}{I\big(q(x)+\hat{a}\big)}\right)^{-1} \\
&\qquad\qquad \left(\frac{I\big(q(x+\hat{\mu})+\hat{a}+\hat{b}\big)}{I\big(q(x+\hat{\mu})+\hat{b}\big)} - \frac{\delta_{a,b}}{N^2}\right)\frac{I\big(q(x+\hat{\mu})+\hat{b}\big)}{I\big(q(x+\hat{\mu})\big)}
\end{aligned} \tag{5.48}$$

$$\begin{aligned}
&\tilde{R}^{\mathrm{inter}}_{\mathrm{ncsh}}(x \to x \pm \hat{\mu},\, a = b) \\
&\quad= \frac{I\big(q(x)\big) - \frac{1}{N^2} I\big(q(x)\big)}{I\big(q(x)+\hat{a}\big) - \frac{1}{N^2} I\big(q(x)\big)}\,\frac{I\big(q(x\pm\hat{\mu})+\hat{a}\big) - \frac{1}{N^2} I\big(q(x\pm\hat{\mu})\big)}{I\big(q(x\pm\hat{\mu})\big) - \frac{1}{N^2} I\big(q(x\pm\hat{\mu})\big)} \\
&\quad= \left(\frac{I\big(q(x)+\hat{a}\big)}{I\big(q(x)\big)} - \frac{1}{N^2}\right)^{-1}\left(\frac{I\big(q(x\pm\hat{\mu})+\hat{a}\big)}{I\big(q(x\pm\hat{\mu})\big)} - \frac{1}{N^2}\right)
\end{aligned} \tag{5.49}$$

$$\begin{aligned}
&\tilde{R}_{\mathrm{mv}}(x = x_\mathrm{h} = x_\mathrm{t} \to x_\mathrm{m},\, ab \to cd) \\
&\quad= \left(1 - \frac{\delta_{a,b}}{N^2}\right)\left(\frac{I\big(q(x)+\hat{a}+\hat{b}\big)}{I\big(q(x)\big)} - \frac{\delta_{a,b}}{N^2}\right)^{-1} \\
&\qquad \left(1 - \frac{\delta_{c,d}}{N^2}\right)^{-1}\left(\frac{I\big(q(x_\mathrm{m})+\hat{c}+\hat{d}\big)}{I\big(q(x_\mathrm{m})\big)} - \frac{\delta_{c,d}}{N^2}\right)
\end{aligned} \tag{5.50}$$

5.2.9 Wolff's Alternative Worm Algorithm

Starting with the Ising model, Wolff generalized the WA in a series of publications [12, 13, 74, 75]. The fourth of these papers is concerned with the $\mathbb{C}\mathrm{P}^{N-1}$ model and was published just during the writing of the present thesis. In that series he proposes a twofold generalization design: First off, the number of "flavors" N is allowed to be non-integral, an enhancement which is of rather limited use. Second, an additional reweighting field $\rho(x)$ (e.g. an initial guess for the Green's function) is introduced to the partition function biasing the dynamics, which turns out to improve on the signal to noise ratio of the correlation at large distances. These generalizations have lead to a terminology which to some extent diverges from the original worm formalism introduced by Prokof'ev and Svistunov. The more so as Wolff's implementation approach is very distinct and involves an alternative data representation. Instead of storing the bond occupation n_{xy} as integer values on a fixed set of bonds (i.e. the overlapping paths encoded in the dual lattice), the configuration is represented by a list containing the varying number of disentangled loops. This requires regrouping the distinction between different update types in the algorithm, which entails quite a radical and complicated change in implementation, including a need to traverse entire loops for reflagging purposes, which is to be seen as a computational overhead purely characteristic of his MC scheme. While this overhead reportedly remains unproblematic for the lattice sizes under study, things look worse when the continuum limit (increasing β and L) is actually taken. There, autocorrelation times still diverge, i.e. Wolff's algorithm, which at its core is nothing else but the worm algorithm with an unusual data organization, is subject to some critical slowing down [12]. This applies in particular to the $\mathbb{C}\mathrm{P}^{N-1}$ case for non-integer N [13]. Since the length of loops that need reflagging diverges roughly like ξ^2, a dynamic critical exponent of about two is faced in the thermodynamic limit. By limiting his scheme to integral N and treating a prefactor in his representation of the partition function stochastically rather than exactly, he finally observes no more systematic increase in autocorrelation times. It remains to be investigated whether an analogical procedure is applicable in the worm formalism derived in this thesis.

5.3 Parallelization

Although being purely local, the worm algorithm presumably doesn't lend itself to a very easy and efficient parallelization. To the present day, the author is unaware of a successful parallel implementation of the WA for any model. A first approach could comprize updating multiple worms at a time and discarding them when a collision is detected. The larger the lattice and the shorter the range of propagation (i.e. the smaller β in asymptotically free theories), the more efficient we expect this ansatz to be. Also, models with a higher degree of complexity in terms of computational costs per head shift could possibly profit more from such a parallelization scheme than abecedarian theories.

6 Conclusion & Outlook

We have given a brief introduction to the problem of critical slowing down, along with a description of how to quantify spatial correlation and autocorrelation of measurements. With these recipies, we numerically confirmed that the single-cluster algorithm eliminates critical slowing down on the two-dimensional O(N) model, but completely fails to do so with the complex-valued $\mathbb{C}\mathrm{P}^{N-1}$ model, where a dynamic critical exponent of at least $z \approx 2$ is faced. Nevertheless, we have obtained high-precision results for moderately large lattices with it.

In chapter 5 we reviewed the worm algorithm for the Ising model and derived the equivalent formalism for the $\mathbb{C}\mathrm{P}^{N-1}$ model, based on the high-temperature expansion previously provided by Chandrasekharan in 2008 [63]. This is the first explicit formulation and implementation of the WA for $\mathbb{C}\mathrm{P}^{N-1}$ (known to the author) in Prokof'ev & Svistunov's original spirit. The set CP of configurations contributing to the partition function needed to be extended to include not only closed paths, but also arbitrarily discontinuous paths labeled aa (i.e. connecting $\bar{z}_a(x)z_a(x)$ with $\bar{z}_a(y)z_a(y)$). With this algorithm, we simulated the two-dimensional $\mathbb{C}\mathrm{P}^1$ model, reproducing exact results to up to six significant digits. The dynamic critical exponent for the slowest modes measured ($|W|$ and W^2) has been determined to be $z = 0.32(3)$, consistent with logarithmic divergence ($z = 0$) within 2σ, indicating that the WA (almost) defeats critical slowing down for $N = 2$. Integrated autocorrelation times for the energy and magnetic susceptibility were even found to slowly decrease with increasing correlation lengths.

For $N > 2$ however, the algorithm suffers from severe lack of ergodicity and is evidently unable to deliver correct results. Possible remedy in the form of a combination of the presented algorithm with ergodicity-enhancing techniques has been proposed. Such a hybrid algorithm may also help in further reducing the dynamic critical exponent for $N = 2$. We need to stress, though, that it is uncertain if the observed incorrectness is purely due to the ergodicity problem, since even very high statistics on tiny systems don't seem to overbear it. On the other hand, from the derivation alone, which holds for general N, we have no reason to believe that the algorithm should be limited to $N = 2$. This issue certainly requires further investigation.

The correlation sampled by our algorithm does not decay to zero at large distances, as the disconnected piece of the Green's function is not included in the configuration weights. For $N = 2$, this doesn't seem to pose a problem, but we may not fully preclude it from being a potential hindrance to efficiency for $N > 2$, although the dynamic critical exponents observed for $\mathbb{C}\mathrm{P}^1$ are quite promising. The disconnected piece might further play a role in the ergodicity deficit. A straightforward attempt to incorporate it into the sampling probabilities ended with the occurrence of the sign problem.

In this work, we derived the WA for the standard quartic action only. Other actions, such as the Symanzik improved version with next-nearest neighbor interaction [41]

$$S_{\mathrm{Sym}} = -\beta \sum_{x,\mu} \left(\frac{4}{3} |\bar{z}(x) \cdot z(x+\hat{\mu})|^2 - \frac{1}{12} |\bar{z}(x) \cdot z(x+2\hat{\mu})|^2 \right) \tag{6.1}$$

or actions with a gauge field $U(x,\mu) \in \mathrm{U}(1)$ like

$$S_{\mathrm{g}} = -\beta \sum_{x,\mu} \left(U(x,\mu)\, \bar{z}(x) \cdot z(x+\hat{\mu}) + \bar{U}(x,\mu)\, \bar{z}(x) \cdot z(x+\hat{\mu}) \right) \tag{6.2}$$

as used by Wolff [13] for example, could be interesting, particularly with regard to potential future applications of strong coupling expansion Monte Carlo to other gauge theories like lattice QCD. Adding a chemical potential is yet another option.

Two more things remain pending: Firstly, an explicit formula for the spin stiffness in the $\mathbb{C}\mathrm{P}^{N-1}$ model as a function of the squared winding numbers (5.39) is desirable and should be rather immediate. And secondly, the decorrelation behavior of our WA as a function of P_{nc} and the no-change update type (neighbor shift or heatbath) could be worth studying.

We conclude by remarking that the conventional WA for the general (in terms of N) $\mathrm{O}(N)$ model, which is still absent from literature, should be easily deducible using the general derivation recipe given in section 5.2, and should be accompanied with less trouble than for $\mathbb{C}\mathrm{P}^{N-1}$. Given that cluster algorithms are CSD-free there, this may be dispensable, though. Moreover, Wolff's alternative WA is available for $\mathrm{O}(N)$ already [12].

Acknowledgements

I would like to thank Dr. Philippe de Forcrand for the provision of such a marvellous concept for this thesis, and his great commitment which allowed me to learn a lot in the process. I would like to thank further Dr. Michael Fromm, Dr. Marco Panero, Dr. Aleksi Kurkela, and Dr. Wolfgang Unger (all at the Institute for Theoretical Physics at ETH Zurich), who have kindly offered numerous valuable discussions and whose help is also much appreciated.

The vast majority of our numerical simulations was carried out on the heterogenous HPC cluster *Brutus* at ETH Zurich.

Bibliography

[1] H. Eichenherr, SU(N) invariant non-linear σ models, Nucl. Phys. B **146** (1978) 215–223.

[2] S. Hikami, Renormalization Group Functions of CP^{N-1} Non-Linear σ-Model and N-Component Scalar QED Model, Prog. Theor. Phys. **62** (1979) 226–233.

[3] U. Wolff, Collective Monte Carlo Updating for Spin Systems, Phys. Rev. Lett. **62** (1989) 361–364.

[4] U. Wolff, Collective Monte Carlo updating in a high precision study of the x-y model, Nucl. Phys. B **322** (1989) 759–774.

[5] U. Wolff, Asymptotic freedom and mass generation in the O(3) nonlinear σ-model, Nucl. Phys. B **334** (1990) 581–610.

[6] R. G. Edwards and A. D. Sokal, Dynamic critical behavior of Wolff's collective-mode Monte Carlo algorithm for the two-dimensional O(n) nonlinear σ model, Phys. Rev. D **40** (1989) 1374–1377.

[7] S. Caracciolo, R. G. Edwards, A. Pelissetto, and A. D. Sokal, Wolff-type embedding algorithms for general nonlinear σ-models, Nucl. Phys. B **403** (1993) 475–541.

[8] M. Hasenbusch and S. Meyer, Testing accelerated algorithms in the lattice CP^3 model, Phys. Rev. D **45** (1992) R4376–R4380.

[9] B. B. Beard, M. Pepe, S. Riederer, and U.-J. Wiese, Study of CP($N-1$) θ-Vacua by Cluster Simulation of SU(N) Quantum Spin Ladders, Phys. Rev. Lett. **94** (2005) 010603.

[10] B. B. Beard, M. Pepe, S. Riederer, and U.-J. Wiese, Efficient cluster algorithm for $CP(N-1)$ models, Comput. Phys. Comm. **175** (2006) 629–634.

B. B. Beard, M. Pepe, S. Riederer, and U.-J. Wiese, An Efficient Cluster Algorithm for CP(N-1) Models, PoS **LAT2005** (2005) 113.

[11] N. Prokof'ev and B. Svistunov, Worm Algorithms for Classical Statistical Models, Phys. Rev. Lett. **87** (2001) 160601.

N. Prokof'ev and B. Svistunov, Worm algorithms for classical statistical models, `arXiv:cond-mat/0103146v1`.

[12] U. Wolff, Simulating the all-order strong coupling expansion III: O(N) sigma/loop models, Nucl. Phys. B **824** (2010) 254–272.

[13] U. Wolff, Simulating the All-Order Strong Coupling Expansion IV: CP($N-1$) as a loop model, Nucl. Phys. B **832** (2010) 520–537.

[14] H. E. Stanley, Dependence of Critical Properties on Dimensionality of Spins, Phys. Rev. Lett. **20** (1968) 589–592.

[15] H. E. Stanley, Spherical Model as the Limit of Infinite Spin Dimensionality, Phys. Rev. **176** (1968) 718–722.

[16] P. G. de Gennes, Exponents for the excluded volume problem as derived by the Wilson method, Phys. Lett. A **38** (1972) 339–340.

[17] L. Onsager, Crystal Statistics. I. A Two-Dimensional Model with an Order-Disorder Transition, Phys. Rev. **65** (1944) 117–149.

[18] J. M. Kosterlitz and D. J. Thouless, Ordering, metastability and phase transitions in two-dimensional systems, J. Phys. C: Solid State Phys. **6** (1973) 1181–1203.

[19] M. Hasenbusch and K. Pinn, Computing the roughening transition of Ising and solid-on-solid models by BCSOS model matching, J. Phys. A: Math. Gen. **30** (1997) 63–80.

[20] A. M. Polyakov, Interaction of Goldstone particles in two dimensions. Applications to ferromagnets and massive Yang-Mills fields, Phys. Lett. B **59** (1975) 79–81.

[21] E. Brézin and J. Zinn-Justin, Renormalization of the Nonlinear σ Model in $2 + \epsilon$ Dimensions-Application to the Heisenberg Ferromagnets, Phys. Rev. Lett. **36** (1976) 691–694.

[22] E. Brézin and J. Zinn-Justin, Spontaneous breakdown of continuous symmetries near two dimensions, Phys. Rev. B **14** (1976) 3110–3120.

[23] V. F. Müller, T. Raddatz, and W. Rühl, The scaling behaviour of the two-dimensional O(N) symmetric Heisenberg model to order N^{-1}, Nucl. Phys. B **251** (1985) 212–226.

[24] M. Falcioni and A. Treves, Corrections to universal scaling for the 2D non-linear sigma model on the lattice, Phys. Lett. B **159** (1985) 140–142.

[25] S. Caracciolo, R. G. Edwards, A. Pelissetto, and A. D. Sokal, Asymptotic Scaling in the Two-Dimensional O(3) σ Model at Correlation Length 10^5, Phys. Rev. Lett. **75** (1995) 1891–1894.

[26] B. Berg and M. Lüscher, Definition and statistical distributions of a topological number in the lattice O(3) σ-model, Nucl. Phys. B **190** (1981) 412–424.

[27] B. Berg and C. Panagiotakopoulos, Definition and statistical properties of a universal topological charge, Nucl. Phys. B **251** (1985) 353–373.

[28] P. Woit, Topological Charge in Lattice Gauge Theory, Phys. Rev. Lett. **51** (1983) 638–641.

[29] P. Woit, Topology and lattice gauge fields, Nucl. Phys. B **262** (1985) 284–298.

[30] B. Berg, Dislocations and topological background in the lattice O(3) σ-model, Phys. Lett. B **104** (1981) 475–480.

[31] M. Lüscher, Does the topological susceptibility in lattice σ models scale according to the perturbative renormalization group?, Nucl. Phys. B **200** (1982) 61–70.

[32] A. D'Adda, M. Lüscher, and P. D. Vecchia, A $1/n$ expandable series of non-linear σ models with instantons, Nucl. Phys. B **146** (1978) 63–76.

[33] K. Jansen and U.-J. Wiese, Cluster algorithms and scaling in $\mathbb{C}$P(3) and $\mathbb{C}$P(4) models, Nucl. Phys. B **370** (1992) 762–772.

[34] M. Imachi, S. Kanou, and H. Yoneyama, Phase structure of CP^{N-1} model with topological term, Nucl. Phys. B (Proc. Suppl.) **73** (1999) 644–646.

[35] D. Petcher and M. Lüscher, Topology and universality in the lattice CP^2 model, Nucl. Phys. B **225** (1983) 53–76.

[36] 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** (1953) 1087–1092.

[37] N. Madras and A. D. Sokal, The pivot algorithm: A highly efficient Monte Carlo method for the self-avoiding walk, J. Stat. Phys. **50** (1988) 109–186.

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

[39] A. D. Sokal, How to beat critical slowing-down: 1990 update, Nucl. Phys. B (Proc. Suppl.) **20** (1991) 55–67.

[40] M. E. Fisher and R. J. Burford, Theory of Critical-Point Scattering and Correlations. I. The Ising Model, Phys. Rev. **156** (1967) 583–622.

[41] M. Campostrini, P. Rossi, and E. Vicari, Monte Carlo simulation of CP^{N-1} models, Phys. Rev. D **46** (1992) 2647–2662.

[42] M. Hasenbusch and T. Török, High-precision Monte Carlo study of the 3D XY-universality class, J. Phys. A: Math. Gen. **32** (1999) 6361–6371.

[43] F. Cooper, B. Freedman, and D. Preston, Solving $\phi^4_{1,2}$ field theory with Monte Carlo, Nucl. Phys. B **210** (1982) 210–228.

[44] J.-K. Kim, Application of finite size scaling to Monte Carlo simulations, Phys. Rev. Lett. **70** (1993) 1735–1738.

[45] H. Arisue, Large-q expansion for the second moment correlation length in the two-dimensional q-state Potts model, Nucl. Phys. B (Proc. Suppl.) **83-84** (2000) 679–681.

[46] M. Lüscher, P. Weisz, and U. Wolff, A numerical method to compute the running coupling in asymptotically free theories, Nucl. Phys. B **359** (1991) 221–243.

[47] M. P. Nightingale and H. W. J. Blöte, Dynamic Exponent of the Two-Dimensional Ising Model and Monte Carlo Computation of the Subdominant Eigenvalue of the Stochastic Matrix, Phys. Rev. Lett. **76** (1996) 4548–4551.

[48] P. D. Coddington and C. F. Baillie, Empirical relations between static and dynamic exponents for Ising model cluster algorithms, Phys. Rev. Lett. **68** (1992) 962–965.

[49] Y. Deng, T. M. Garoni, and A. D. Sokal, Dynamic Critical Behavior of the Worm Algorithm for the Ising Model, Phys. Rev. Lett. **99** (2007) 110601.

[50] R. H. Swendsen and J.-S. Wang, Nonuniversal critical dynamics in Monte Carlo simulations, Phys. Rev. Lett. **58** (1987) 86–88.

[51] C. M. Fortuin and P. W. Kasteleyn, On the random-cluster model : I. Introduction and relation to other models, Physica **57** (1972) 536–564.

[52] U. Wolff, Critical slowing down, Nucl. Phys. B (Proc. Suppl.) **17** (1990) 93–102.

[53] U. Wolff, Monte Carlo simulation of a lattice field theory as correlated percolation, Nucl. Phys. B **300** (1988) 501–516.

[54] R. Zorn, H. J. Herrmann, and C. Rebbi, Tests of the multi-spin-coding technique in Monte Carlo simulations of statistical systems, Comp. Phys. Comm. **23** (1981) 337–342.

[55] S. Bae, S. H. Ko, and P. D. Coddington, Parallel Wolff Cluster Algorithms, Int. J. Mod. Phys. C **6** (1995) 197–210.

[56] M. Hasenbusch, Improved estimators for a cluster updating of O(n) spin models, Nuclear Physics B **333** (1990) 581–592.

[57] U. Wolff, Scaling topological charge in the CP^3 spin model, Phys. Lett. B **284** (1992) 94–98.

[58] M. Hasenbusch and S. Meyer, Does the topological charge scale in the lattice CP^3-model?, Phys. Lett. B **299** (1993) 293–298.

[59] N. Schultka and M. Müller-Preussker, The topological susceptibility of the lattice CP^{n-1} model on the torus and the sphere, Nucl. Phys. B **386** (1992) 193–214.

[60] N. V. Prokof'ev, B. V. Svistunov, and I. S. Tupitsyn, "Worm" algorithm in quantum Monte Carlo simulations, Phys. Lett. A **238** (1998) 253–257.

[61] F. J. Wegner, Duality in Generalized Ising Models and Phase Transitions without Local Order Parameters, J. Math. Phys. **12** (1971) 2259–2272.

[62] W. Janke, T. Neuhaus, and A. M. J. Schakel, Critical loop gases and the worm algorithm, Nucl. Phys. B **829** (2010) 573–599.

[63] S. Chandrasekharan, A new computational approach to lattice quantum field theories, PoS **LATTICE 2008** (2008) 003.

[64] E. Vicari, Monte Carlo simulation of lattice CP^{N-1} models at large N, Phys. Lett. B **309** (1993) 139–144.

[65] E. Rabinovici and S. Samuel, The CP^{N-1} model: A strong coupling lattice approach, Phys. Lett. B **101** (1981) 323–326.

[66] I. K. Affleck and H. Levine, Strong coupling versus large N in σ-models, Nucl. Phys. B **195** (1982) 493–502.

[67] E. Marinari and G. Parisi, Simulated Tempering: A New Monte Carlo Scheme, Europhys. Lett. **19** (1992) 451–458.

[68] G. M. Torrie and J. P. Valleau, Nonphysical Sampling Distributions in Monte Carlo Free-Energy Estimation: Umbrella Sampling, J. Comput. Phys. **23** (1977) 187–199.

[69] A. M. Ferrenberg and R. H. Swendsen, Optimized Monte Carlo data analysis, Phys. Rev. Lett. **63** (1989) 1195–1198.

[70] P. M. C. de Oliveira, T. J. P. Penna, and H. J. Herrmann, Broad histogram Monte Carlo, Eur. Phys. J. B **1** (1998) 205–208.

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

[72] M. Campostrini, P. Rossi, and E. Vicari, Topological susceptibility and string tension in the lattice CP^{N-1} models, Phys. Rev. D **46** (1992) 4643–4656.

[73] R. K. Kaul and R. G. Melko, Large-N estimates of universal amplitudes of the $\mathbb{CP}^{N-1}$ theory and comparison with a $S = \frac{1}{2}$ square-lattice model with competing four-spin interactions, Phys. Rev. B **78** (2008) 014417.

[74] U. Wolff, Simulating the all-order strong coupling expansion I: Ising model demo, Nucl. Phys. B **810** (2009) 491–502.

[75] U. Wolff, Simulating the all-order hopping expansion II: Wilson fermions, Nucl. Phys. B **814** (2009) 549–572.