

# Monte Carlo Simulationen in der statistischen Physik

M. Hasenbusch\*

*Humboldt Universität zu Berlin, Institut für Physik  
Invalidenstr. 110, D-10115 Berlin, Germany  
Raum 417 tel.: (2093) 7664*

## Zusammenfassung

Die Vorlesung soll eine Einführung in die Methode der Monte Carlo Simulation in der statistischen Mechanik geben. Als Standardbeispiel wird häufig das Ising-Modell betrachtet.

Es wird der Unterschied zwischen “simple sampling” und “importance sampling” erklärt. Der letzte Fall ist für uns im Folgenden der wichtige. Hier werden in einer “Markovkette” Konfigurationen mit dem Boltzmannfaktor als Gewicht erzeugt.

Dazu werden verschiedene Simulationsalgorithmen verwendet: Metropolis-, Wärmebad-, Überrelaxations-, und Cluster-Algorithmen.

Die Daten, die aus der Simulation hervorgehen, müssen ausgewertet und interpretiert werden. Dazu werden Methoden der Datenanalyse besprochen. Stichworte dazu sind: Statistische Fehler, Varianz, Autokorrelationszeit, Binning, Jackknife.

Schließlich wollen wir aus den Zahlen etwas zur Physik lernen: Wie kann man die Phasenübergangstemperatur bestimmen? Wie erkennt man die Ordnung des Phasenübergangs? Wie bestimmt man kritische Exponenten?

---

\*e-mail: [hasenbus@physik.hu-berlin.de](mailto:hasenbus@physik.hu-berlin.de)

# 1 Einleitung

## 1.1 Physik

### 1.1.1 Ising-Modell

Als Beispiel werden wir oft das Ising-Modell betrachten. Dieses Modell ist als einfaches Modell eines Ferromagneten gedacht. Man stellt sich vor, daß auf den Punkten eines Gitters ein "Elementarmagnet" lebt, der sich entlang einer Achse ausrichten kann. Von einem solchen groben (klassischen) Modell kann man (zunächst) keine genauen quantitativen Ergebnisse erwarten. Es geht vielmehr um ein qualitatives Verständnis eines Systems mit vielen Freiheitsgraden. Von besonderem Interesse sind hier Phasenübergänge. Bei tiefen Temperaturen zeigt ein Ferromagnet eine spontane (d.h. ohne daß ein äußeres Feld anliegt) Magnetisierung. Wird er aufgeheizt, wird die Magnetisierung immer schwächer und ab einer bestimmten Temperatur, der Phasenübergangstemperatur, verschwindet sie vollständig.

Das Ising-Modell wurde von E.Ising in Z.Phys.31 (1925) 253 für ein eindimensionales Gitter gelöst. Daher rührt der Name des Modells. Es zeigt sich, daß in einer Dimension das Phänomen der spontanen Magnetisierung noch nicht auftritt.

1944 gelang Onsager die exakte Lösung des zweidimensionalen Modells (L. Onsager, Phys.Rev. 65 (1944) 117). Hier tritt in der Tat ein Phasenübergang bei endlicher Temperatur auf. In drei Dimensionen gibt es bis heute (trotz der Einfachheit des Modells) keine exakte Lösung. Hier ist man auf Approximationen und (wie in dieser Vorlesung besprochen werden soll) auf Monte-Carlo-Simulationen angewiesen. In vier und mehr Dimensionen werden wichtige Eigenschaften des Modells exakt durch die "Mean-field" Theorie wiedergegeben.

Nun zur genauen mathematischen Definition des Modells: Die Gitterpunkte  $x = (x_1, \dots, x_d)$  sind Elemente eines d-dimensionalen Gitters. Der Einfachheit halber betrachten wir hier einfach kubische Gitter

$$x_i \in \{1, 2, 3, \dots, L_i\} . \quad (1)$$

Die Gitterkonstante haben wir hier gleich 1 gesetzt. Meistens werden wir sogenannte periodische Randbedingungen wählen:  $(x_1, \dots, L_i, \dots, x_d)$  ist der nächste Nachbar von  $(x_1, \dots, 1, \dots, x_d)$ .

Auf den Gitterpunkten wird ein "Spin" definiert  $s_x \in \{-1, 1\}$ . Dem Wort "Spin" darf man an dieser Stelle nicht allzuviel Bedeutung beimessen.  $s_x$  ist lediglich eine Zahl die entweder den Wert  $-1$  oder  $+1$  annimmt. Um nun die statistische Mechanik des Modells zu definieren, muß man jeder Konfiguration, d.h. jeder möglichen Belegung der  $s_x$ , einen Energiewert zuweisen. Wir bemerken, daß es genau  $2^V$  Konfigurationen gibt. Dabei ist  $V$  die Anzahl der Gitterpunkte. Im Isingmodell ist die Hamiltonfunktion durch

$$H = -J \sum_{\langle xy \rangle} s_x s_y - h \sum_x s_x \quad (2)$$

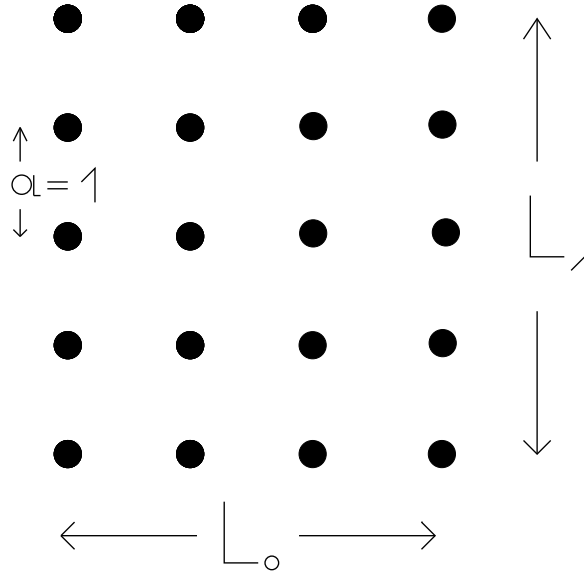


Abbildung 1: Abbildung eines zweidimensionalen Gitters.

gegeben. Dabei bezeichnet  $\langle xy \rangle$  ein Paar von nächsten Nachbarpunkten auf dem Gitter.  $J$  ist die Kopplungsstärke zwischen benachbarten Spins und  $h$  ein äußeres Feld. Meistens werden wir den Spezialfall  $h = 0$  betrachten.

Im kanonischen Ensemble tritt jede Konfiguration mit dem Boltzmannfaktor  $\exp(-\beta H)$  als Gewicht auf. Dabei ist

$$\beta = \frac{1}{k_b T} . \quad (3)$$

$T$  ist die absolute Temperatur und  $k_b$  die Boltzmannkonstante.

Die Zustandssumme ist durch

$$Z = \sum_{s_0, \dots, 0 = \pm 1} \dots \sum_{s_{L_1, \dots, L_d} = \pm 1} \exp(-\beta H) \quad (4)$$

gegeben.

Als Abkürzung verwenden wir im folgenden:

$$\sum_{conf} := \sum_{s_0, \dots, 0 = \pm 1} \dots \sum_{s_{L_1, \dots, L_d} = \pm 1} . \quad (5)$$

Dabei sind die Gitterpunkte durch ihren Ort  $(x_1, \dots, x_d)$  gekennzeichnet.

Erwartungswerte einer Observablen  $A$  sind durch

$$\langle A \rangle := \frac{\sum_{conf} \exp(-\beta H) A}{Z} \quad (6)$$

gegeben.

Beispiele für Observabeln sind:

Magnetisierung :

$$m := \frac{1}{V} \sum_x s_x \quad , \quad (7)$$

Korrelationsfunktion:

$$G(x, y) := s_x s_y \quad . \quad (8)$$

Weitere Beispiele werden im Lauf der Vorlesung gegeben.

Umfassende Diskussionen lassen sich auch in vielen Lehrbüchern zur statistischen Mechanik finden. z.B. Kapitel 14-18 in der zweiten Auflage von Kerson Huang, *Statistical Mechanics* , John Wiley & Sons.

Neben dem Magnetismus kann man das Ising-Modell auch als Modell für weitere Vielteilchensysteme verstehen. Am wichtigsten ist dabei die Gruppe der binären Gemische. Dabei wird die eine Teilchensorte mit den  $-1$  Spins identifiziert und die andere mit  $+1$ .

Durch geeignete Wahl der Randbedingungen und der Geometrie des Gitters lassen sich eine Vielzahl von Phänomenen untersuchen:

Dünne Filme, Kapillareffekte, Oberflächen, Trennschichten, usw.

Beispiele aus der Literatur:

Critical behaviour of three-dimensional Ising ferromagnets at imperfect surfaces

H. W. Diehl

cond-mat/9801254 , Eur. Phys. J. B 1, 401-404 (1998)

Computing the Roughening Transition of Ising and Solid-On-Solid Models by BCSOS

M. Hasenbusch, K. Pinn

cond-mat/9605019, J.Phys. A30 (1997) 63-80

Geht man davon ab die Kopplungskonstante  $J$  ortsunabhängig zu wählen, so lassen sich z.B. Phänomene untersuchen, die durch Verunreinigungen und Störstellen im Gitter hervorgerufen werden.

Beispiel aus der Literatur:

Critical exponents of the three dimensional diluted Ising model

H. G. Ballesteros, L. A. Fernandez, V. Martin-Mayor, A. Munoz Sudupe, G. Parisi,

J. J. Ruiz-Lorenzo

cond-mat/9802273

Wird die Kopplungskonstante  $J_{\langle xy \rangle}$  für jedes Paar von Gitterpunkten zufällig gewählt (z.B. mit gleicher Wahrscheinlichkeit  $+1$  oder  $-1$  oder aus einer Gaussverteilung um  $0$ .), hat man ein Modell für sogenannte "Spingläser". Bis heute sind

solche Systeme nicht gut verstanden. Das beste Werkzeug zur Untersuchung ist die Monte-Carlo-Simulation.

Beispiel aus der Literatur:

On the Phase Structure of the 3D Edwards Anderson Spin Glass

E. Marinari, G. Parisi, J. J. Ruiz-Lorenzo

cond-mat/9802211

Vor kurzem ist ein Artikel über einen Spezialcomputer zur Simulation von Spingläsern erschienen:

cond-mat/0004080

SUE: A Special Purpose Computer for Spin Glass Models

A. Cruz, J. Pech, A. Tarancon, P. Tellez, C. L. Ullod, C. Ungil

### 1.1.2 Andere Gittermodelle

In der Literatur wird ein große Zahl weiterer Modelle diskutiert. Wählt man den Spin als 3-komponentigen Einheitsvektor, erhält man das klassische Heisenberg-Modell. Mit zwei Komponenten erhält man das XY-Modell.

$$H = -J \sum_{\langle xy \rangle} \vec{s}_x \vec{s}_y - \vec{h} \sum \vec{s}_x \quad (9)$$

Eine andere Verallgemeinerung des Ising-Modells sind die q-Zustands-Potts-Modelle. Hier kann der Spin  $q$  verschiedene diskrete Zustände annehmen.  $s_x \in \{1, 2, \dots, q\}$

$$H = -J \sum_{\langle xy \rangle} \delta_{s_x, s_y} \quad (10)$$

Im Landau-Ginzburg-Modell nimmt der Spin Werte aus dem  $R^N$  an.

Diese Liste ist bei weitem nicht vollständig.

### 1.1.3 Gittereichtheorien

Ein Forschungsschwerpunkt des Lehrstuhls von Herrn Prof. Wolff ist die numerische Untersuchung (mittels Monte-Carlo-Simulationen) der QCD. Die theoretische Grundlage dazu bietet die Gittereichttheorie. Dabei betrachtet man Quantenfeldtheorien in der Pfadintegralformulierung. Nach Wickrotation (zu imaginärer Zeit) und Diskretisierung kommen die Quantenfeldtheorien in eine Form, die derjenigen der oben betrachteten Modelle der klassischen statistischen Physik sehr ähnlich ist.

Damit lassen sich die Rechenmethoden der statistischen Physik (und damit auch die Monte Carlo Simulation) auf diese Probleme anwenden. In der Vorlesung wollen wir nicht näher auf Gittereichttheorien eingehen. Der Interessierte ist auf Lehrbücher verwiesen:

M. Creutz, Quarks, Gluons and the Lattice, Cambridge University Press.

Heinz J. Rothe, Lattice Gauge Theories, An Introduction, World Scientific  
Münster und Montvay, Quantum Fields on the Lattice, Cambridge University Press

Für eine kurze Einführung siehe auch die Heräus lectures von Prof. Wolff:  
FTP://linde.physik.hu-berlin.de/pub/heraeus/

Dieser Formalismus lässt sich bereits auf die Nichtrelativistische Quantenmechanik anwenden. (Siehe Lehrbücher zur Quantenfeldtheorie z.B. Kerson Huang, Quarks, Leptons and Gaugefields, World Scientific, Kapitel 7.)

## 1.2 Mathematik

Numerische Integration einer Funktion  $f(x)$  von  $t_0$  bis  $t_1$ :

$$\int_{t_0}^{t_1} f(x)dx = \lim_{n \rightarrow \infty} \frac{t_1 - t_0}{n} \sum_{i=1}^n f(x_i) \quad (11)$$

mit

$$x_i = t_0 + \frac{i}{n}(t_1 - t_0) \quad (12)$$

Die Punkte  $x_i$  heißen Stützstellen

Fehler bei der Numerischen Integration :  $O(a)$

mit  $a = (t_1 - t_0)/n$

Verbesserte Verfahren sind Trapezregel, Simpsonregel ...

Fehler bei der Numerischen Integration :  $O(a^2)$ , ...

Für reelle Funktionen führen solche Verfahren in der Regel ohne großen Aufwand zu einem befriedigenden Resultat. (Siehe verschiedene Lehrbücher der Numerischen Mathematik oder Kapitel 4 von Press et. al. Numerical Recipies)

Problematisch wird es, wenn hochdimensionale Integrale berechnet werden sollen.

$$I = \int dx_1 \int dx_2 \dots \int dx_D f(x_1, x_2, \dots, x_D). \quad (13)$$

Nehmen wir der Einfachheit wegen an, daß die Integrationsgrenzen in jeder Dimension gleich sind, o.b.d.A. von 0 bis 1, und daß man in jeder Dimension den gleichen Abstand  $a$  zwischen den Stützstellen hat. Dann ist die Zahl der Stützstellen  $n$  mit  $n = (1/a)^D$ . Der Rechenaufwand ist proportional zur Zahl der Stützstellen. Nehmen wir als Beispiel  $a = 0.1$ , was sicherlich einer groben Einteilung des Intervalls  $[0, 1]$  entspricht. Dann ist für unser Beispiel die Zahl der Stützstellen  $n = 10^D$ .

Nehmen wir des weiteren an, daß die Auswertung der Funktion für eine Stützstelle  $10^{-7}s$  benötigt (realistische Größenordnung für moderne Workstations). Dann ergeben sich die in Tabelle 1 gegebenen Rechenzeiten als Funktion der Dimension.

Aus dieser groben Abschätzung sehen wir, daß die numerische Berechnung von Integralen einer Dimension größer als (ungefähr) 12 mit äquidistanten Stützstellen unmöglich wird.

Tabelle 1: CPU-zeit in Abhängigkeit der Dimension des Integrals. Details siehe Text.

D	CPU-zeit
1	$10^{-6}$ s
2	$10^{-5}$ s
3	$10^{-4}$ s
4	$10^{-3}$ s
...	...
10	16.7 min
11	2.8 h
12	1.16 Tage

Hochdimensionale Integrale treten in der Statistischen Mechanik und der Pfadintegralformulierung von Quantenfeldtheorien auf.

## 2 Monte-Carlo-Integration

Ein Ausweg aus dem Problem bietet die Monte-Carlo-Integration. Dabei geht man von regulär verteilten Stützstellen ab und nimmt statt dessen “zufällig” aus dem Integrationsgebiet gewählte Stützstellen. Was hierbei zufällig heißen soll, wird unten genauer erklärt.

Für ein 1-dimensionales Integral sieht das wie folgt aus:

$$\int_{t_0}^{t_1} f(x)dx \approx \frac{t_1 - t_0}{N} \sum_{i=1}^N f(x_i) \quad , \quad (14)$$

wobei nun die  $x_i$  gleichverteilte Zufallszahlen aus dem Intervall  $t_1 - t_0$  sind.

Der (statistische) Fehler der Methode ist proportional zu  $1/\sqrt{N}$ . Wichtig ist, daß auch in mehr als einer Dimension diese Abhängigkeit des Fehlers von der Zahl der Stützstellen gilt.

### 2.1 Zufallszahlen

Als erstes wollen wir diskutieren, wie man Zufallszahlen einer bestimmten Verteilung erhalten kann. Ich möchte im folgenden keinen Versuch unternehmen, den Begriff Zufall philosophisch auszuleuchten, sondern anhand von Beispielen deutlich machen was gemeint ist.

- Würfel. Mit gleicher Wahrscheinlichkeit bekommt man eine der Zahlen  $\{1, 2, 3, 4, 5, 6\}$ . Das Ergebnis eines einzelnen Wurfes ist nicht vorhersagbar.
- Lottozahlen.
- Radioaktiver Zerfall.

Eine Möglichkeit, Zufallszahlen in der Numerik zu verwenden, ist ein physikalisches System, das den Beispielen 1 bis 3 ähnlich ist, an den Rechner zu koppeln. Dies ist jedoch recht unpraktisch und für die meisten Zwecke zu langsam.

Deshalb verwendet man für die Monte-Carlo-Integration in der Regel sogenannte “Pseudo-Zufallszahlen”. Diese “Pseudo-Zufallszahlen” werden nach einer einfachen Vorschrift als eine Folge von Zahlen gebildet. Diese Zahlenfolge ist daher vollkommen deterministisch. Für einen Betrachter der Folge ist es jedoch schwer, das Bildungsgesetz der Folge zu erkennen.

Häufig verwendet man “linear congruential” Zufallszahlen-Generatoren. Ihr Bildungsgesetz ist

$$I_{j+1} = (aI_j + c) \bmod m \quad . \quad (15)$$

Das heißt, daß die alte Zahl  $I_j$  mit  $a$  multipliziert wird und  $c$  addiert wird. Danach wird der Modulus mit  $m$  genommen. Dadurch ist sichergestellt, daß  $0 \leq I_j < m$  für alle  $j$  gilt.



Eine wichtige Kenngröße eines Zufallszahlengenerators ist die Periode. Die Periode  $p$  gibt an, nach wie vielen Schritten sich eine Zahl in der Folge  $I_j$  wiederholt:  $I_j = I_{j+p}$ . Ziel bei der Festlegung von  $a$ ,  $c$  und  $m$  ist eine möglichst lange Periode des Generators.

Im günstigsten Fall ist die Periode  $p = m$ . Es können in der Tat auch solche  $a$  und  $c$  gefunden werden, so daß  $p = m$  gilt. Diese  $a$  und  $c$  sind vollständig klassifiziert:

(Hull and Dubell, SIAM Review 4 (1962) 230)

Ein linear kongruenter Zufallszahlengenerator mit Modulus  $m$  hat genau dann die Periode  $m$  wenn gilt:

- (i)  $c$  und  $m$  haben keine gemeinsamen Primfaktoren
- (ii)  $a - 1$  ist ein Vielfaches von  $d$  für jeden Primfaktor von  $m$ .
- (iii)  $a - 1$  ist ein Vielfaches von 4, falls  $m$  ein Vielfaches von 4 ist.

Linear kongruente Generatoren mit  $m = 2^n$  lassen sich besonders effizient implementieren. Deshalb werden häufig solche Generatoren verwendet. Die Bedingungen (i)-(iii) bedeuten für diesen Spezialfall  $a = 4k + 1$  mit einer natürlichen Zahl  $k$  und  $c$  ist ungerade.

Beispiele für die Wahl der Konstanten  $a$ ,  $c$  und  $m$  sind  
drand48 aus der C-library:

$a = 0x5DEECE66D$  (base 16) = 0273673163155 (base 8)

$c = 0xB$  (base 16) = 013 (base 8)

In Dezimalzahlen  $a =$ ,  $c = 11$ ,  $m = 2^{48}$ . Die Bedingungen (i) -(iii) lassen sich leicht überprüfen und man sieht, daß drand48() die Periode  $m$  hat.

G05CAF aus der NAG-library:

$a = 13^{13}$ ,  $c = 0$ ,  $m = 2^{59}$

Da hier  $c = 0$  ist, kann die volle Periode nicht erreicht werden. Im Skript von Andrea Pelissetto finden wir, daß für  $m = 2^w$  mit  $w > 3$  für  $a \bmod 8 = 3, 5$  und  $X_0$  ungerade die Periode gleich  $2^{w-2}$  ist. In der Tat gilt  $13^{13} \bmod 8 = 5$ . Damit ist die Periode von G05CAF  $p = 2^{57}$ .

Der Generator erzeugt also zunächst ganze Zahlen zwischen 0 und  $m - 1$ . Durch multiplizieren der  $I_j$  mit  $1/m$  werden reelle Zahlen erzeugt, die gleichförmig in dem Intervall  $[0, 1)$  verteilt sind.

Die "linear congruential" Zufallszahlengeneratoren sind nicht der einzigen Typ von Zufallszahlengeneratoren, die benutzt werden. Weitere Typen sind die "Fibonacci"- oder "shift register"-Generatoren. Für Infos zu diesen Generatoren siehe das CPII Skript oder das Skript von A. Pelissetto.

Eine lange Periode ist leider noch nicht das einzige Kriterium für einen guten Zufallszahlengenerator. Für eine weitergehende Diskussion möchte ich auf das CPII

Skript oder das Skript von A. Pelissetto verweisen.

Darüber hinaus gibt es auch jüngere Forschungsarbeiten zu diesem Thema. Als Beispiele für neuere Arbeiten, in der eine grössere Zahl von Generatoren getestet wurde, möchte ich

New tests of random numbers for simulations in physical systems, I. Vattulainen, cond-mat/9411062

The RANLUX generator: resonances in a random walk test, Lev N. Shchur, Paolo Butera, hep-lat/9805017

A new test for random number generators: Schwinger-Dyson equations for the Ising, H. G. Ballesteros, V. Martin-Mayor, cond-mat/9806059

nennen.

Die oben aufgeführten "G05CAF" und "drand48" erweisen sich für unsere Anwendungen als vollständig befriedigend. Für die kommenden Übungsaufgaben werden wir einen dieser beiden Generatoren, je nach Verfügbarkeit, verwenden.

## 2.2 Verteilungen von Zufallszahlen (Zufallsvariablen)

### a) Diskrete Zufallszahlen; Beispiel: Würfel; Lottozahlen

Beim Würfel sind die möglichen elementaren Ereignisse (Ergebnisse) die Zahlen  $X \in \{1, 2, 3, 4, 5, 6\}$ . ( $\{1, 2, 3, 4, 5, 6\}$  Gesamtereignis).

Jedes dieser Ereignisse hat eine Wahrscheinlichkeit  $W(X)$ :

$$0 \leq W(X) \leq 1 \quad . \quad (16)$$

Eine der möglichen Zahlen fällt mit Sicherheit:

$$\sum_X W(X) = 1 \quad (17)$$

Sind  $A$  und  $B$  Teilmengen des Gesamtereignisses und  $A \cap B = \emptyset$  dann gilt:

$$W(A \cup B) = W(A) + W(B) \quad (18)$$

Experimentell wird  $W(X)$  bestimmt, indem man häufig ( $N$ -mal) würfelt. Dann ist

$$W(X) \approx \frac{N_X}{N} \quad , \quad (19)$$

wobei  $N_X$ -mal die Zahl  $X$  gewürfelt worden ist.

### b) Kontinuierliche Zufallszahlen:

Die Wahrscheinlichkeitsverteilung  $w(x)$  einer kontinuierlichen Zufallszahl  $x$  lässt sich aufbauend auf der Wahrscheinlichkeit für diskrete Ereignisse definieren: Sei  $W(a, b)$  die Wahrscheinlichkeit, daß  $x$  im Intervall  $[a, b]$  liegt. Dann ist  $w(x)$  dadurch bestimmt, daß für jedes beliebige Intervall  $[a, b]$

$$W(a, b) = \int_a^b dx w(x) \quad (20)$$

gilt. Mit den oben besprochenen Zufallszahlen wird (zumindest in einer für unsere Zwecke hinreichend guten Näherung) eine gleichförmige Verteilung im Intervall  $[0, 1)$  erzeugt. D.h.  $w(x) = 1$  für  $x \in [0, 1)$  und  $w(x) = 0$  außerhalb dieses Intervalls.

Wichtige Eigenschaften von  $w(x)$  sind:

$$w(x) \geq 0 \quad (21)$$

für alle  $x$  und

$$\int dx w(x) = 1 \quad (22)$$

D.h.: "Es passiert immer etwas".

### Erwartungswert und Varianz einer Funktion f:

Erwartungswert

$$\langle f \rangle_w := \int dx w(x) f(x) \quad (23)$$

Varianz

$$\begin{aligned} \sigma_w^2(f) &:= \int dx w(x) (f(x) - \langle f \rangle_w)^2 \quad (24) \\ &= \int dx w(x) f(x)^2 - \langle f \rangle_w^2 \\ &= \langle f^2 \rangle_w - \langle f \rangle_w^2 \quad . \end{aligned}$$

## 2.3 Erzeugung verschiedener Wahrscheinlichkeitsverteilungen

Die oben besprochenen Zufallszahlengeneratoren liefern uns eine gleichförmige Verteilung in Intervall  $[0, 1)$ . Häufig möchte man aber Zufallszahlen mit einer anderen Verteilung verwenden.

Ein einfaches Verfahren, andere Verteilungen zu erhalten, ist es, die durch den Zufallszahlengenerator erzeugten Zahlen mit Hilfe einer Funktion  $y_j = f(x_j)$  abzubilden. Dabei transformiert sich die Verteilung gemäß

$$\tilde{w}(y) = w(x) \left| \frac{dx}{dy} \right| \quad (25)$$

Ein Beispiel ist die Erzeugung von Zufallszahlen, die in einem Intervall  $[a, b)$  gleich verteilt sind. Dies wird durch die Abbildung  $y = a + (b - a)x$  erreicht, wobei  $x$  im Intervall  $[0, 1)$  gleich verteilt ist.

Ein weiteres Beispiel, welches wir im weiteren Verlauf der Vorlesung verwenden werden, ist eine exponentiell zerfallende Verteilung in  $[0, \infty)$ :  $\tilde{w}(y) = \exp(-y)$ . Diese kann aus einer im Intervall  $(0, 1]$  gleich verteilten Zufallszahl  $x$  durch die Abbildung  $y = -\ln(x)$  erzeugt werden. Da  $x = \exp(-y) = -dx/dy$  gilt, führt diese Abbildung in der Tat zu dem gewünschten Ergebnis.

Nicht für alle gewünschten Verteilungen kommt man mit der oben erklärten Methode zum Ziel. Ein generell einsetzbares Verfahren ist das “rejection”-(Ablehnungs)-Verfahren.

Sei  $\tilde{w}(y)$  die gewünschte Verteilung. Nun sucht man eine Verteilung  $f(y)$ , die sich mit Hilfe der oben besprochenen Methode erzeugen läßt, so daß

$$cf(y) \geq \tilde{w}(y) \tag{26}$$

für alle  $y$ . Dann erzeugt man Zufallszahlen mit der Verteilung  $f(y)$ . Diese Zufallszahlen werden mit der Wahrscheinlichkeit

$$A(y) = \frac{w(y)}{cf(y)} \tag{27}$$

angenommen und ansonsten verworfen.

Implementierung der “rejection method” in C:

(Beispiel:

Ausgangsverteilung:  $w(x)$ : Gleichverteilung in  $[0, 1)$ . Neue Verteilung:  $\tilde{w}(x) = 2x$  in  $[0, 1)$ .

```
double reject()
{
    double vor;
    /* Erzeuge einen Vorschlag: */
    anfang:
    vor=g05caf();
    /* rejection-schritt; vor = \tilde{w}(vor)/ 2 w(vor): */
    if(vor < g05caf()) goto anfang;
    return vor;
}
```

Ohne das berüchtigte goto mit der do-while-Schleife:

```
double reject()
{
    double vor;
    /* Erzeuge einen Vorschlag: */
    do
    {
        vor=g05caf();
    }
    /* rejection-schritt; vor = \tilde{w}(vor)/ 2 w(vor): */
    while(vor < g05caf());
    return vor;
}
```

## 2.4 Monte-Carlo-Integration

Nun möchten wir die in Gleichung (14) in der Einleitung angegebene Integrationsmethode genauer betrachten.

$$I = \int_a^b dx g(x) = \int_a^b dx w(x) f(x) = \langle f \rangle_w \quad . \quad (28)$$

Wobei  $w(x) f(x) = g(x)$  im Intervall  $[a, b]$ . Den Erwartungswert  $\langle f \rangle_w$  können wir durch den Mittelwert über endlich viele ( $N$ ) Zufallszahlen approximieren:

$$\langle f \rangle_w \approx \bar{f}_N := \frac{1}{N} \sum_{i=1}^N f(x_i) \quad . \quad (29)$$

Dabei haben die Zufallszahlen  $x_i$  die Verteilung  $w(x)$ .

Wichtige Frage: Wie groß ist der Fehler; d.h. wie groß ist die Differenz  $\langle f \rangle - \bar{f}_N$ ?

## 2.5 Statistische Fehler

Die Frage nach dem statistischen Fehler der stochastischen Integration bedeutet genauer formuliert, welche Verteilung ein Ensemble von (unabhängigen) Mittelungen über  $N$  viele Zufallszahlen hat. Die Breite dieser Verteilung interpretieren wir als den statistischen Fehler der Monte-Carlo-Integration.

Die mittlere quadratische Abweichung (Varianz) des Ensembles von unabhängig voneinander bestimmten  $\bar{f}_N$  von  $\langle f \rangle$  ist ein Maß für die Breite der Verteilung:

$$\sigma^2(\bar{f}_N) := \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{j=1}^M (\langle f \rangle - \bar{f}_{N,j})^2 \quad (30)$$

Wenn wir nur eine Messreihe haben, können wir  $\sigma^2(\bar{f}_N)$  nicht direkt bestimmen. Wir müssen  $\sigma^2(\bar{f}_N)$  mit der Varianz  $\sigma^2(f)$  von  $f$  selbst in Beziehung bringen. Es ergibt sich:

$$\begin{aligned} & \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{j=1}^M (\langle f \rangle - \bar{f}_{N,j})^2 \\ &= \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{j=1}^M \left( \frac{1}{N} \sum_{i=1}^N (\langle f \rangle - f(x_{i,j})) \right)^2 \\ & \quad \text{Dabei bezeichnet } x_{i,j} \text{ die } i\text{-te Zufallszahl im } j\text{-ten Mittelwert.} \\ &= \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{j=1}^M \frac{1}{N^2} \left( \sum_{i=1}^N (\langle f \rangle - f(x_{i,j})) \right) \left( \sum_{l=1}^N (\langle f \rangle - f(x_{l,j})) \right) \\ &= \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{j=1}^M \frac{1}{N^2} \sum_{i=1}^N \sum_{l=1}^N (\langle f \rangle^2 - \langle f \rangle (f(x_{i,j}) + f(x_{l,j})) + f(x_{i,j})f(x_{l,j})) \end{aligned}$$

$$\begin{aligned}
&= \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{j=1}^M \frac{1}{N^2} \sum_{i=1}^N \sum_{l=1}^N (f(x_{i,j})f(x_{l,j}) - \langle f \rangle^2) \\
&= \frac{1}{N} \lim_{M \rightarrow \infty} \frac{1}{MN} \sum_{j=1}^M \sum_{i=1}^N (f(x_{i,j})^2 - \langle f \rangle^2) \\
&= \frac{1}{N} \sigma^2(f)
\end{aligned} \tag{31}$$

Man beachte, daß sich von der drittletzten zur vorletzten Zeile gilt:

$$\begin{aligned}
&\lim_{M \rightarrow \infty} \frac{1}{M} \sum_{j=1}^M \frac{1}{N^2} \sum_{i=1, l=1}^N f(x_{i,j})f(x_{l,j}) \\
&= \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{j=1}^M \frac{1}{N^2} \left( \sum_{i=1, l=1, i \neq l}^N f(x_{i,j})f(x_{l,j}) + \sum_{i=1}^N f(x_{i,j})^2 \right) \\
&= \frac{N-1}{N} \langle f \rangle^2 + \frac{1}{N} \langle f^2 \rangle .
\end{aligned} \tag{32}$$

$\sigma^2(f)$  können wir nun mit Hilfe unserer Messreihe bestimmen:

$$\sigma^2(f) \approx \frac{1}{N-1} \sum_{i=1}^N (\bar{f}_N - f(x_i))^2 \tag{33}$$

Als Fehler der stochastischen Integration erhält man

$$\sqrt{\sigma^2(\bar{f}_N)} = \sqrt{\sigma^2(f)/N} . \tag{34}$$

Der Zentrale Grenzwertsatz besagt, daß die Verteilung der  $\bar{f}_N$  im Limes großer  $N$  durch eine Gaußverteilung gegeben ist:

$$w(\bar{f}_N) \approx \frac{\sqrt{N}}{2\pi\sigma(f)} \exp\left(-N \frac{(\langle f \rangle - \bar{f}_N)^2}{2\sigma^2(f)}\right) . \tag{35}$$

Tabelle 2: Wahrscheinlichkeit  $W(n)$ , daß das Resultat  $\bar{f}$  der Monte-Carlo-Integration in dem Intervall  $[\langle f \rangle - n\sigma(\bar{f}), \langle f \rangle + n\sigma(\bar{f})]$  liegt.

n	W(n)
1	0.682689...
2	0.954499...
3	0.997300...
4	0.999936...
5	0.999999...

## 2.6 Importance sampling, Varianzreduzierung

Ziel bei der Monte-Carlo-Berechnung eines Integrals ist es natürlich, einen möglichst kleinen statistischen Fehler zu erhalten.

Dies kann man auf der einen Seite dadurch erreichen, daß mehr CPU-Zeit investiert wird und sich so der Faktor  $1/\sqrt{N}$  verkleinert. Auf der anderen Seite sollte man die Verteilungsfunktion  $w(x)$  möglichst so wählen, daß die Varianz von  $f$  möglichst klein ist.

Im Idealfall haben wir  $w(x) \propto g(x)$ . Dann ist  $f(x)$  eine Konstante und die Varianz ist gleich null. Allerdings ist in diesem Fall die Aufgabe auch schon exakt analytisch gelöst und wir brauchen den Rechner nicht mehr zu bemühen.

Zur Verdeutlichung der Idee der Varianzreduzierung möchten ich das einfache Beispiel

$$I = \int_0^t dx \exp(-x) x \quad (36)$$

betrachten. Den Wert dieses Integrals können wir natürlich exakt ausrechnen:

$$\int_0^t dx \exp(-x) x = -\exp(-x)x - \exp(-x)|_0^t = 1 - (1+t) \exp(-t) \quad (37)$$

Als erste mögliche Wahl von  $w(x)$  nehmen wir  $w_1(x) = \frac{1}{t}$  im Intervall  $[0, t]$ . Damit ist

$$f_1(x) = t \exp(-x) x \quad (38)$$

Als zweite Wahl betrachten wir

$$w_2(x) = \frac{1}{1 - \exp(-t)} \exp(-x) \quad (39)$$

im Intervall  $[0, t]$ . Damit ist

$$f_2(x) = (1 - \exp(-t)) x \quad (40)$$

Berechnen wir nun die Varianz für diese beiden Wahlen:

$$\begin{aligned} \sigma_{w_1}^2(f_1) &= \int_0^t dx \frac{1}{t} (t \exp(-x) x)^2 - \langle f \rangle^2 \\ &= t \left( \frac{1}{4} - \left( \frac{1}{2}t^2 + \frac{1}{2}t + \frac{1}{4} \right) \exp(-2t) \right) - \langle f \rangle^2 \\ &\approx \frac{1}{4}t \end{aligned} \quad (41)$$

für grosse  $t$ . Das bedeutet, daß die Varianz mit wachsendem Intervall immer grösser wird. Im Limes  $t \rightarrow \infty$  ist die Verteilung  $w_1$  auch gar nicht mehr definiert.

$$\sigma_{w_2}^2(f_2) = \int_0^t dx \frac{1}{1 - \exp(-t)} \exp(-x) ((1 - \exp(-t))x)^2 - \langle f \rangle^2$$



$$\begin{aligned}
&= (1 - \exp(-t)) \int_0^t dx \exp(-x)x^2 - \langle f \rangle^2 \\
&= (1 - \exp(-t)) (2 - (t^2 + 2t + 2) \exp(-t)) - \langle f \rangle^2 \\
&\approx 1
\end{aligned} \tag{42}$$

für grosse  $t$ . Das heißt, daß die Varianz etwa gleich bleibt bei grösser werdendem  $t$ . Insbesondere ist die Verteilung  $w_2(x)$  auch für den Limes  $t \rightarrow \infty$  verwendbar.

Faustregel:  $g(x)/w(x)$  soll im Integrationsgebiet möglichst wenig schwanken, damit man einen kleinen statistischen Fehler bei der Monte-Carlo-Integration erhält.

## 2.7 Einfaches Monte-Carlo-Experiment:

Als physikalisches Beispiel wollen wir den "Random Walk" betrachten (Brownsche Bewegung, Diffusion). Nehmen wir an, daß ein Teilchen bei der Position  $(0, 0)$  (in 2 Dimensionen) startet. Die Richtung sei zufällig gewählt, mit jeder Richtung gleich wahrscheinlich. Die Länge des Weges die das Teilchen zurücklegt bevor es ein anderes trifft, genüge einer Verteilung  $p(r) = \exp(-r)$ . Der Einfachheit halber wollen wir annehmen, daß alle Richtungen beim Weiterlaufen des Teilchen nach der Streuung an einem anderen Teilchen gleich wahrscheinlich sind. Damit ergibt sich:

$$w(\vec{r}) = w(r, \phi) = \frac{1}{2\pi} \frac{1}{r} \exp(-r) . \tag{43}$$

```

r=-ln(g05caf());
phi=2.*3.1415926535898*g05caf();
r1=r*cos(phi);
r2=r*sin(phi);

```

Das Problem läßt sich auch auf einem Gitter formulieren. Nehmen wir konkret ein 2-dimensionales quadratisches Gitter, so kann sich der Zufallsläufer in jedem Schritt aufs neue mit gleicher Wahrscheinlichkeit für eine der 4 möglichen Richtungen entscheiden.

Die Frage ist nun, wie weit sich das Teilchen im Mittel nach  $N$  Schritten von seiner Ausgangsposition entfernt hat.

Diese Frage ist in einfacher Weise mit dem Problem der oben diskutierten Fehlerrechnung zu verknüpfen. Wir müssen lediglich den gesamten zurückgelegten Weg durch die Zahl der Schritte dividieren. Sei  $\vec{R}(N)$  der Weg, der nach  $N$  Schritten zurückgelegt ist, dann identifizieren wir:

$$\vec{f}_N = \frac{\vec{R}(N)}{N} \tag{44}$$

und die Funktion  $\vec{f} = \vec{r}$  ist der in einem Schritt zurückgelegte Weg.

In beiden oben diskutierten Fällen ergibt sich  $\langle \vec{f} \rangle = (0, 0)$ . Die Varianz ist damit

$$\sigma^2(\vec{f}) = \int dx_1 \int dx_2 \vec{f}(\vec{x})^2 . \tag{45}$$

Für die betrachteten Spezialfälle:

$$\sigma^2(\vec{f}) = \frac{1}{4}((1, 0)^2 + (0, 1)^2 + (-1, 0)^2 + (0, -1)^2) = 1 \quad (46)$$

für den Gitterlauf und

$$\sigma^2(\vec{f}) = \int_0^{2\pi} d\phi \int_0^\infty dr \frac{1}{2\pi} \exp(-r) r^2 (\cos(\phi), \sin(\phi))^2 = \int_0^\infty dr r^2 \exp(-r) = 2 \quad (47)$$

für die kontinuierliche Version der random walks. Als mittlere quadratische Weglänge erhalten wir damit

$$\langle |\vec{R}(N)|^2 \rangle = N \quad (48)$$

für den Lauf auf dem Gitter und

$$\langle |\vec{R}(N)|^2 \rangle = 2N \quad (49)$$

für den kontinuierlichen Lauf.

### **Aufgabe:**

Bestimmen Sie numerisch  $\langle |\vec{R}(N)|^2 \rangle$  für alle  $0 < N < 1000$  für die beiden oben diskutierten Typen des "random walks". Dazu mitteln Sie über 10000 auf dem Rechner erzeugte Zufallswege. Plotten Sie das Ergebnis zusammen mit Fehlerbalken und der theoretischen Vorhersage.

Wichtiges Beispiel: Perkolation. Siehe das CPII-Skript, Kapitel 6.

## 3 Importance sampling, Markov-Kette und Metropolis

### 3.1 “Simple sampling”

Die konzeptionell einfachste Möglichkeit, Observablen im Ising-Modell mit Hilfe von Monte-Carlo-Methoden zu bestimmen ist das “simple sampling”.

Dabei wird eine Konfiguration dadurch erzeugt, daß an jedem Gitterpunkt  $x$ , unabhängig von den anderen Gitterpunkten, mit gleicher Wahrscheinlichkeit entweder  $s_x = -1$  oder  $s_x = +1$  gewählt wird.

Erzeugung einer Konfiguration in C:

```
for(i=0;i<V;i++)
{
  if(g05caf(>)>0.5)
  {
    spin[i]=1;
  }
  else
  {
    spin[i]=-1;
  }
}
```

Dabei ist  $V$  die Anzahl der Gitterpunkte. Alle Konfigurationen werden mit der gleichen Wahrscheinlichkeit erzeugt. Da die Zahl der Konfigurationen  $2^V$  ist, wird eine bestimmte Konfiguration mit der Wahrscheinlichkeit  $2^{-V}$  erzeugt.

Observablen sind definiert über

$$\langle A \rangle := \frac{\sum_X \exp(-\beta H(X)) A(X)}{\sum_X \exp(-\beta H(X))} , \quad (50)$$

wobei  $Z = \sum_X \exp(-\beta H(X))$  die Zustandssumme ist.  $X$  steht hier für eine Konfiguration  $(s_1, s_2, \dots, s_V)$ .

Noch einmal zur Erinnerung: Sobald die Zahl der Gitterpunkte etwas grösser als 40 wird, kann man die Summen in Gleichung (50) nicht mehr auf dem Computer ausführen. (Etwa  $10^6$  s für eine Konfiguration bedeutet 13 h für alle Konfigurationen auf 40 Gitterpunkten.)

Mit der Monte-Carlo-Methode lässt sich jedoch ein Schätzwert der Summen in Zähler und Nenner von Gleichung (50) bestimmen:

Erzeugt man  $N$  Konfigurationen mit dem oben beschriebenen einfachen Verfahren, so ergibt sich

$$Z \approx \frac{2^V}{N} \sum_{i=1}^N \exp(-\beta H(X_i)) \quad (51)$$

und

$$\sum_X \exp(-\beta H(X)) A(X) \approx \frac{2^V}{N} \sum_{i=1}^N \exp(-\beta H(X_i)) A(X_i) . \quad (52)$$

Das Problem von “simple sampling” ist, daß nur ein sehr kleiner Anteil aller Konfigurationen einen großen Boltzmannfaktor hat. Verwendet man “simple sampling” (D.h. man erzeugt alle Konfigurationen mit gleichem Gewicht), so erzeugt man sehr viele Konfigurationen fast nutzlos, da sie nur sehr wenig zu der Zustandssumme, und damit auch zu den Erwartungswerten von Observablen beitragen.

Im folgenden möchte ich das Problem konkreter am Beispiel des 2-dimensionalen Ising-Modells ohne äußeres Feld bei der kritischen Temperatur betrachten. Wir betrachten ein quadratisches Gitter mit  $L \times L$  Gitterpunkten und periodischen Randbedingungen. In Tabelle 3 ist der Anteil der Konfigurationen gegeben, der nötig ist, um bei der kritischen Temperatur  $T_c = 1/0.4406868\dots$  90% der Zustandssumme zu erhalten. Dazu wurden alle Konfigurationen für diese Gitter erzeugt. Man sieht, daß simple sampling mit wachsender Gittergröße weniger effizient wird.

Tabelle 3: Anteil der Konfigurationen die nötig sind um 90% der Zustandssumme im 2D Ising-Modell am Phasenübergang zu erhalten. Details siehe Text.

$L$	Anteil
2	0.875
3	0.133
4	0.0343
5	0.00283

### 3.2 Lösung des Problems: Importance sampling

Die Lösung ist, die Physik auf dem Rechner dahingehend nachzubilden, daß Konfigurationen genau mit ihrem Boltzmannfaktor als relativem Gewicht erzeugt werden. Das heißt, daß wir die Konfiguration  $X$  mit einer Wahrscheinlichkeit

$$W(X) = \frac{1}{Z} \exp(-\beta H(X)) \quad (53)$$

erzeugen wollen. Das Problem dabei ist nun, daß wir den Normierungsfaktor, die Zustandssumme  $Z$  nicht kennen. Daher ist es insbesondere nicht möglich, einzelne Konfigurationen mit der gewünschten Wahrscheinlichkeit unabhängig voneinander zu erzeugen.

Die Lösung ist ein stochastischer Prozess. Hierbei sind lediglich Verhältnisse der Wahrscheinlichkeit nötig:

$$\frac{W(X)}{W(Y)} = \frac{\exp(-\beta H(X))}{\exp(-\beta H(Y))} , \quad (54)$$

welche einfach zu berechnen sind.

In der (klassischen) Physik erfolgt die Erzeugung der Konfiguration mittels der Dynamik der Teilchen gemäß der Bewegungsgleichung. In der Monte-Carlo-Simulation werden wir auch eine Dynamik, d.h. eine zeitliche Abfolge von Konfigurationen einsetzen, um unser Ziel zu erreichen. Dabei werden wir jedoch nicht die Dynamik der Physik verwenden, sondern eine “erfundene”. Wesentliche Unterschiede zur Physik sind, daß die Zeitschritte endlich sind und daß der Prozess nicht deterministisch, sondern stochastisch ist.

Um den Zeitablauf in dieser “erfundene” Dynamik von einer physikalischen Zeitentwicklung zu unterscheiden, verwenden wir im folgenden den Begriff der Monte-Carlo-Zeit.

Damit diese erfundene Dynamik wirklich die Konfigurationen mit dem Boltzmannfaktor/ $Z$  als Wahrscheinlichkeit erzeugt, müssen bestimmte Bedingungen erfüllt sein. Diese sollen im folgenden diskutiert werden.

### 3.3 Markovkette

Der Typ des stochastischen Prozesses, den wir benutzen wollen, heißt Markovkette (erster Ordnung). Eine Markovkette erster Ordnung ist dadurch charakterisiert, daß die Wahrscheinlichkeit, daß einer Konfiguration  $X$  die Konfiguration  $Y$  in der Zeitreihe folgt, vollkommen unabhängig von den Konfigurationen, die zeitlich vor  $X$  liegen, ist.

Bemerkung: Nach dieser Klassifizierung ist der normale random-walk, bei dem wir den Ort des Läufers als Konfiguration auffassen, eine Markovkette, während der self-avoiding walk keine Markovkette ist.

Self-avoiding heißt, daß der “Läufer” die Gitterpunkte, die er schon einmal betreten hatte, kein zweites Mal betreten darf. Damit wird die Entscheidung für einen neuen Gitterpunkt von der gesamten Vorgeschichte abhängig.

Die statistischen Eigenschaften der Markovkette sind vollständig beschrieben durch die Festlegung der Übergangswahrscheinlichkeit  $W(Y, X)$  von einer Konfiguration  $X$  zu einer Konfiguration  $Y$ .

Damit die in der Markovkette erzeugten Konfigurationen gemäß der Boltzmannverteilung verteilt sind, müssen folgende Anforderungen an die Übergangswahrscheinlichkeit erfüllt sein:

**i) Normierung:**

$$\sum_Y W(Y, X) = 1 \quad (55)$$

für alle  $X$ . Dies bedeutet, daß es genau eine Nachfolgekongfiguration gibt.

**ii) Ergodizität:**

$$W(Y, X) > 0 \quad (56)$$

für alle  $X$  und  $Y$ . Das heißt, daß alle Konfiguration erreicht werden können.

**iii) “Stabilität”: Der Boltzmannfaktor ist Eigenvektor mit Eigenwert 1 von  $W(Y, X)$**

$$\sum_X W(Y, X)B(X) = B(Y) \quad (57)$$

für alle  $Y$ . Dabei ist  $B(Y) = \frac{\exp(-\beta H(Y))}{Z}$ .

Wenn die Forderungen i,ii,iii erfüllt sind, dann gilt:

Gegeben sei eine Startverteilung  $V_0(X)$ .

Die Kette wird fortgesetzt durch  $V_{t+1}(X) = \sum_Y W(X, Y)V_t(Y)$ .

Dann wird im Limes großer  $t$  die Boltzmannverteilung erreicht:

$$\lim_{t \rightarrow \infty} \|V_t - B\| = 0 \quad (58)$$

mit

$$\|V_t - B\| := \sum_X |V_t(X) - B(X)| \quad (59)$$

**Beweis:**

$$\begin{aligned} & \sum_X |V_{t+1}(X) - B(X)| \\ = & \sum_X \left| \sum_Y W(X, Y) V_t(Y) - B(X) \right| \\ = & \sum_X \left| \sum_Y W(X, Y) (V_t(Y) - B(Y)) \right| \\ & \text{Mit } \tilde{W}(X, Y) = W(X, Y) - W_{min} \geq 0 \text{ und } W_{min} > 0 \\ = & \sum_X \left| \sum_Y \tilde{W}(X, Y) (V_t(Y) - B(Y)) + W_{min} \sum_Y (V_t(Y) - B(Y)) \right| \\ = & \sum_X \left| \sum_Y \tilde{W}(X, Y) (V_t(Y) - B(Y)) \right| \\ & \text{Mit der Dreiecksungleichung:} \\ \leq & \sum_X \sum_Y |\tilde{W}(X, Y) (V_t(Y) - B(Y))| \\ & \text{Da } \tilde{W}(X, Y) \geq 0 : \\ = & \sum_X \sum_Y \tilde{W}(X, Y) |(V_t(Y) - B(Y))| \\ & \text{Mit der Eigenschaft i):} \\ = & (1 - N_{conf} W_{min}) \sum_Y |(V_t(Y) - B(Y))| \end{aligned} \quad (60)$$

Damit haben wir gezeigt, daß die Markovkette bei beliebiger Startverteilung gegen die Boltzmannverteilung konvergiert.

Es gilt

$$\|V_t - B\| \leq \exp\left(-\frac{t}{R}\right) \|V_0 - B\| \quad (61)$$

wobei  $R = -1/\ln(1 - N_{conf}P_{min})$ .

Eine hinreichende Bedingung für iii) ist das sogenannte detaillierte Gleichgewicht oder “**detailed balance**”:

$$\frac{W(X, Y)}{W(Y, X)} = \frac{B(X)}{B(Y)} \quad (62)$$

für alle  $X$  und  $Y$ . **Beweis:** (mit Benutzung von i) )

$$\sum_Y W(X, Y)B(Y) = \sum_Y W(Y, X)B(X) = B(X) \quad (63)$$

In der Praxis spielt “detailed balance” eine wichtige Rolle; alle hier diskutierten Algorithmen erfüllen “detailed balance”.

### 3.4 Metropolis-Algorithmus

Nun wollen wir beginnen, konkrete Algorithmen zu besprechen. Ein einfacher und auch sehr allgemein einsetzbarer Algorithmus ist der Metropolis-Algorithmus (N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller J.Chem.Phys. 21 (1953) 1087).

Dabei wird die Übergangswahrscheinlichkeit  $W(Y, X)$  in zwei Teile aufgespalten. Der eine Teil ist die Wahrscheinlichkeit  $T(Y, X)$ , ausgehend von der Konfiguration  $X$  eine Konfiguration  $Y$  vorzuschlagen. Der Vorschlag  $Y$  wird mit einer Wahrscheinlichkeit  $A(Y, X)$  akzeptiert. Ansonsten wird die alte Konfiguration beibehalten. Somit nimmt  $W(Y, X)$  die Form

$$W(Y, X) = A(Y, X) T(Y, X) + (1 - \sum_U A(U, X) T(U, X)) \delta(X, Y) \quad (64)$$

an. Dabei werden an  $T$  und  $A$  folgende Bedingungen gestellt:

$$\sum_Y T(Y, X) = 1 \quad (65)$$

$$T(X, Y) = T(Y, X) \quad (66)$$

und

$$A(Y, X) = \min(1, \exp(-\beta(H(Y) - H(X)))) \quad (67)$$

Man sieht sofort, daß die Normierungsbedingung

$$\sum_Y W(Y, X) = 1 \quad (68)$$

erfüllt ist.

Auch der Beweis von **detailed balance** ist recht einfach.

Sei  $X \neq Y$

(Der Fall  $X = Y$  ist trivial).

**Fall 1)**  $H(X) > H(Y)$

$$\begin{aligned}
 \frac{W(Y, X)}{W(X, Y)} &= \frac{T(Y, X)A(Y, X)}{T(X, Y)A(X, Y)} \\
 &= \frac{1}{\exp(-\beta(H(X) - H(Y)))} \\
 &= \frac{\exp(-\beta H(Y))}{\exp(-\beta H(X))}
 \end{aligned} \tag{69}$$

**Fall 2)**  $H(X) \leq H(Y)$

$$\begin{aligned}
 \frac{W(Y, X)}{W(X, Y)} &= \frac{T(Y, X)A(Y, X)}{T(X, Y)A(X, Y)} \\
 &= \frac{1}{\exp(-\beta(H(Y) - H(X)))} \\
 &= \frac{\exp(-\beta H(Y))}{\exp(-\beta H(X))}
 \end{aligned} \tag{70}$$

Um zu sehen, ob der Algorithmus ergodisch ist, muß  $T(Y, X)$  noch genauer spezifiziert werden.

Man beachte, daß Gleichung 67 nicht die einzige mögliche Wahl für die Akzeptanzwahrscheinlichkeit ist. Die einfachste Modifikation ist es, das oben gegebene  $A(Y, X)$  mit einer Konstanten zwischen 0 und 1 zu multiplizieren. (Was in der Regel aber zu einer Verschlechterung der Leistung des Algorithmus führt).

### 3.5 Metropolis für das Ising-Modell

Nun wollen wir den Metropolis-Algorithmus auf das Ising-Modell anwenden. In einem einzelnen Schritt wird vorgeschlagen, den Spin auf einem einzelnen Gitterpunkt  $x$  umzuklappen.

$$T_x(Y, X) = \delta([s_1, \dots, -s_x, \dots, s_V], [s_1, \dots, s_x, \dots, s_V]) \tag{71}$$

Um die Akzeptanzwahrscheinlichkeit zu bestimmen, müssen wir die Differenz  $(H(Y) - H(X))$  ausrechnen. Da sich der Spin nur an einem Gitterpunkt ändert, heben sich die meisten Terme weg, und es bleibt

$$H(Y) - H(X) = 2 s_x \sum_{y.nn.x} s_y , \tag{72}$$



wobei  $y.nn.x$  heißen soll:  $y$  ist nächster Nachbar von  $x$ .

Praktische Nebenbemerkung:

Da das Berechnen der Exponentialfunktion auf dem Rechner relativ teuer ist, kann das Programm beschleunigt werden, indem man die möglichen Werte von  $A(Y, X)$  vorab berechnet und in einer Tabelle ablegt. Dabei ist zu beachten, daß z.B. in 2 Dimensionen  $\sum_{y.nn.x} s_y$  nur die Werte  $-4, -2, 0, 2, 4$  annehmen kann, wenn das äußere Feld verschwindet.

### **Ergodizität:**

Da in einem elementaren Schritt lediglich der Spin eines einzigen Gitterpunktes abgeändert werden kann, reicht ein einzelner Schritt nicht aus, um Ergodizität zu erfüllen. Dies kann erst erreicht werden, wenn elementare Schritte für verschiedene Gitterpunkte hintereinander ausgeführt werden.

Dies kann auf verschiedene Weisen erreicht werden:

Eine Möglichkeit ist es, für jeden elementaren Schritt den Gitterpunkt zufällig auszuwählen.

Häufiger verwendet man einen geordneten Durchlauf durch das Gitter. Ein vollständiger Durchlauf durch das Gitter wird in der Literatur auch “**sweep**” genannt.

Unten ist eine function eines C-Programms abgedruckt, welches das 2D Ising-Modell ohne äußeres Feld mit dem Metropolis-Algorithmus simuliert.

Im Hauptprogramm (oder in anderen functions) wird die Konfiguration initialisiert, und es werden die Observablen “gemessen” (d.h. für die erzeugten Konfigurationen ausgerechnet.)

```
void metro()
{
  int i0,i1,i0p,i0m,i1p,i1m,neigh;

  /*sweep over the lattice*/
  for(i0=0;i0 < L0;i0++)
  {
  /*take periodic boundary conditions into account*/
  i0p=i0+1; if(i0p==L0) i0p=0; i0m=i0-1; if(i0==0) i0m=L0-1;
  for(i1=0;i1 < L1;i1++)
  {
  /*take periodic boundary conditions into account*/
  i1p=i1+1; if(i1p==L1) i1p=0; i1m=i1-1; if(i1==0) i1m=L1-1;

  /*take the sum over the nearest neighbours*/
  neigh=spin[i0][i1p]+spin[i0][i1m]+spin[i0p][i1]+spin[i0m][i1];
  neigh=neigh*spin[i0][i1];
```

```

/* decide whether to accept the proposal spin to -spin */
    if(p[4+neigh]>drand48()) spin[i0][i1]=-spin[i0][i1];
    }
}
}

```

Im Hauptprogramm ist bereits vorausgerechnet:

```

/* compute the acceptance probabilities */
p[0]=1.;
p[2]=1.;
p[4]=1.;
p[6]=exp(-4.*beta);
p[8]=exp(-8.*beta);

```

In dem oben abgedruckten Programm haben wir für die Spins ein 2-dimensionales array verwendet. Eine andere Möglichkeit ist es die Spins in einem ein-dimensionales array abzulegen, und die Nachbarschaftsbeziehung zwischen den Gitterpunkten in einem array abzuspeichern.

Im Hauptprogramm ist bereits vorausgerechnet:

```

/* define the neighbours of a lattice point */
for(i0=0;i0 < L0;i0++)
{
    /*take periodic boundary conditions into account*/
    i0p=i0+1; if(i0p==L0) i0p=0; i0m=i0-1; if(i0==0) i0m=L0-1;

    for(i1=0;i1 < L1;i1++)
    {
        /*take periodic boundary conditions into account*/
        i1p=i1+1; if(i1p==L1) i1p=0; i1m=i1-1; if(i1==0) i1m=L1-1;

        nlist[i0*L1+i1][0]=i0p*L1+i1;
        nlist[i0*L1+i1][1]=i0*L1+i1p;
        nlist[i0*L1+i1][2]=i0m*L1+i1;
        nlist[i0*L1+i1][3]=i0*L1+i1m;
    }
}

```

Damit sieht das Metropolis-Programm so aus:

```

void metro()
{
int i,d,neigh;

```

```

/*sweep over the lattice*/
for(i=0;i < V;i++)
{
  neigh=0;
  /*take the sum over the nearest neighbours*/
  for(d=0; d<2*D; d++) neigh+=spin[nlist[i][d]];
  neigh*=spin[i];
  /* decide whether to accept the proposal spin to -spin */
  if(p[4+neigh]>drand48()) spin[i]=-spin[i];
}
}

```

Als nächstes wollen wir das oben abgedruckte Programm für die Simulation des 2D Ising-Modells verwenden. Von der exakten Lösung des Modells wissen wir, daß es bei  $\beta_c = 0.4406868\dots$  einen Phasenübergang zweiter Ordnung gibt. Für Temperaturen, die größer sind als  $T_c = 1/\beta_c$  gibt es keine spontane Magnetisierung. Für tiefere Temperaturen setzt eine spontane Magnetisierung ein. Der exakte Ausdruck für die Magnetisierung (im unendlich großen Volumen) ist

$$\langle m \rangle = \left(1 - [\sinh(2\beta)]^{-4}\right)^{1/8} . \quad (73)$$

(C. N. Yang, Phys. Rev. 85 (1952) 808.)

Als erstes wollen wir die Frage der Thermalisierung betrachten. Für große Monte-Carlo-Zeiten  $t$  gilt

$$\|B - V_t\| \propto \exp\left(-\frac{t}{R}\right) . \quad (74)$$

Um korrekte Schätzwerte für die Observablen zu erhalten, darf man die ersten  $nR$  (mit  $n$  in der Praxis etwa 10 bis 100) erzeugten Konfigurationen nicht in die Berechnung der Observablen einbeziehen.

Da  $R$  in der Regel nicht exakt bekannt ist, muß man versuchen,  $R$  aus der Simulation selbst abzuschätzen.

Dazu ist es nützlich die (Monte-Carlo) Zeitentwicklung der gemessenen Observablen graphisch darzustellen. Insbesondere, wenn keine Erfahrungen mit dem verwendeten Algorithmus, angewendet auf das gegebene Problem, vorliegen, ist es auch sehr wichtig, Simulationen mit verschiedenen Startkonfigurationen zu vergleichen. (Etwa mit einer geordneten und einer vollkommen ungeordneten Konfiguration.)

Zur Illustration sind in Abbildung 2 die ersten 50 bzw. 100 Werte von  $-H$  und der Magnetisierung für den Metropolis Algorithmus angewandt auf das 2D Ising-Modell bei  $\beta = \beta_c$  auf einem  $32^2$  Gitter gegeben. Zwischen den einzelnen Messungen liegt jeweils ein sweep durch das Gitter.

Man sieht, daß die Werte für die Magnetisierung etwa ab  $t = 80$  zu überlappen beginnen. Um auf der sicheren Seite zu sein, wird man etwa die ersten 800 Konfigurationen bei der Mittelung weglassen. Hier findet man auch andere Faustregeln ("rules of thumb"). Von Alan Sokal z.B. stammt die Regel, daß man  $100\tau_{exp}$  oder

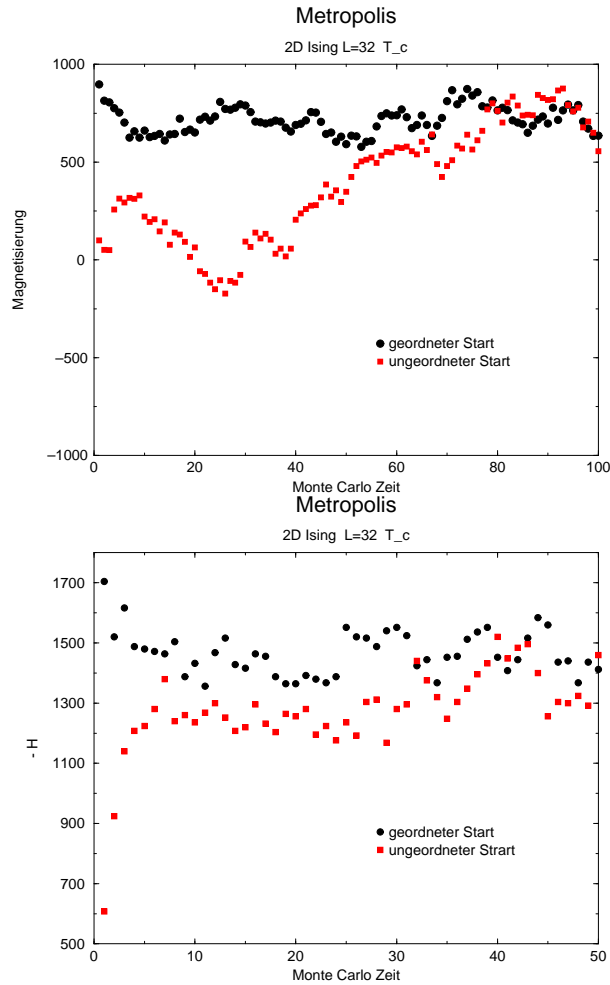


Abbildung 2:

noch besser  $1000\tau_{exp}$  wegwerfen soll. Zur Definition der Autokorrelationszeit  $\tau_{exp}$  siehe weiter unten.

In der Abbildung 3 sind 10000 Messungen der Magnetisierung für ein  $32^2$  Gitter gegeben. Zwischen den Messungen liegt jeweils ein Metropolis-sweep durch das Gitter. Die  $\beta$ -Werte sind 0.48,  $\beta_c = 0.4406868$  und 0.40. Ein Lauf mit 10000 sweeps auf dem  $32^2$  Gitter braucht 44.4 sec auf jerry.

Man erkennt deutlich den Unterschied zwischen  $\beta > \beta_c$  und  $\beta < \beta_c$ . Im ersten Fall hat sich das System spontan für ein Vorzeichen der Magnetisierung entschieden, und alle Messungen liefern eine Magnetisierung, die größer als Null ist (dies müssen wir allerdings noch einmal kritisch betrachten). Für  $\beta = 0.40$  fluktuiert die Magnetisierung um die Null herum.

In allen drei Beispielen sehen wir, daß die Magnetisierung bei kurz aufeinander folgenden Messungen ähnlich ist. Das rührt daher, daß die Konfigurationen

in der Markovkette nicht unabhängig voneinander erzeugt werden. Es gibt eine statistische Abhängigkeit zwischen den erzeugten Konfigurationen. Will man den statistischen Fehler des Simulationsergebnisses korrekt bestimmen, so muß diese Korrelation berücksichtigt werden.

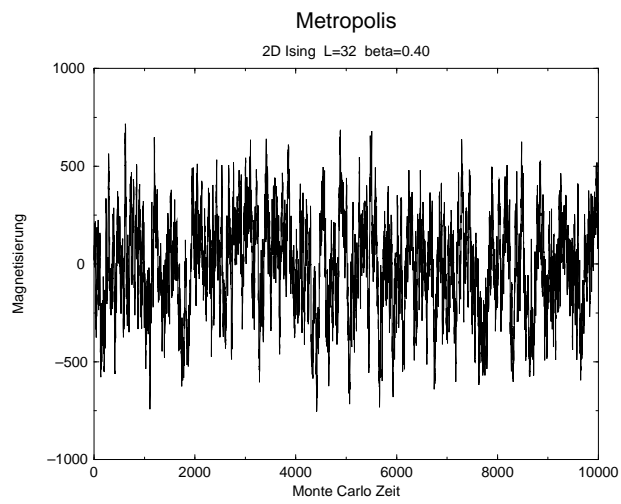
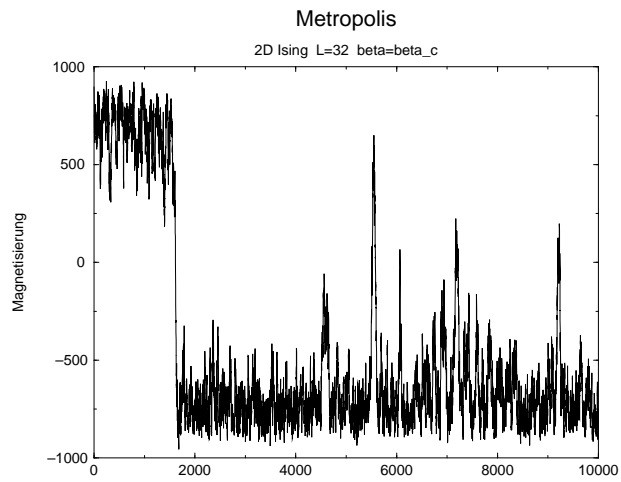
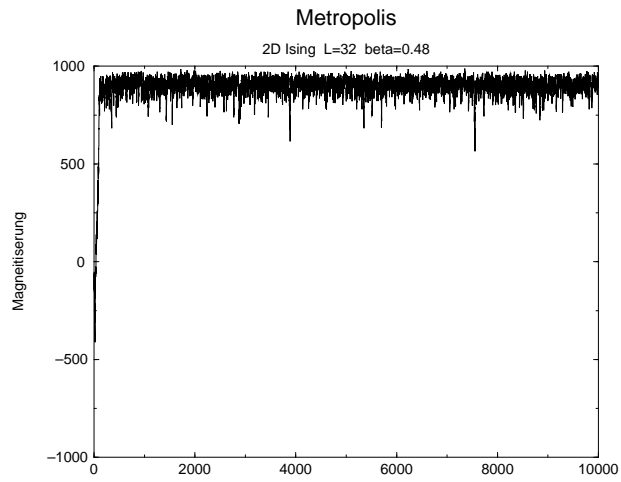


Abbildung 3:

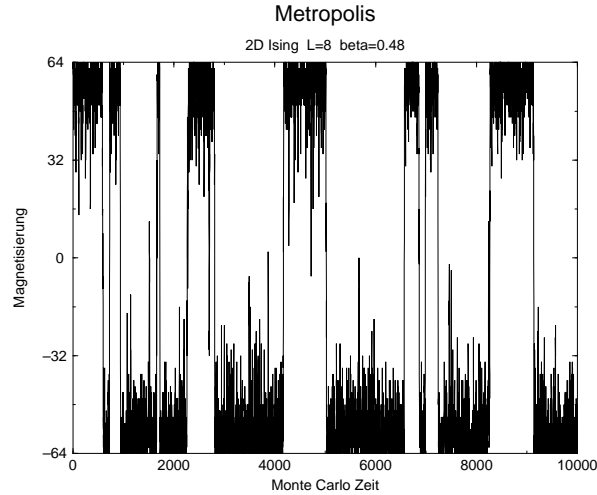


Abbildung 4:

## 4 Statistische Fehler und Autokorrelationszeit

Für Daten, die mit Hilfe einer Markovkette gewonnen worden sind, müssen wir unsere Analyse des statistischen Fehlers erweitern. Die einzelnen “Messungen”, d.h. die Werte der Observablen für die erzeugten Konfigurationen, sind statistisch korreliert.

Die einfachste Methode der Fehlerrechnung ist das sogenannte “binning”. Dazu werden die Daten zuerst in Teilmessreihen oder “bins” der Größe  $n_b$  zusammengefasst:

$$\hat{A}_j = \frac{1}{n_b} \sum_{i=(j-1)n_b+1}^{jn_b} A_i \quad (75)$$

Im Limes  $n_b \rightarrow \infty$  werden die  $\hat{A}_j$  voneinander statistisch unabhängig. Dadurch können wir auf die Fehlerrechnung für unkorrelierte Daten zurückgreifen:

$$\sigma^2(\bar{A}) = \frac{n_b}{N} \sigma^2(\hat{A}). \quad (76)$$

Dabei ist  $N/n_b$  die Anzahl der Bins. In der Praxis wird man den statistischen Fehler für eine Reihe von verschiedenen  $n_b$  berechnen, um zu überprüfen, ob sich der Fehler mit wachsendem  $n_b$  stabilisiert. Zu beachten ist dabei jedoch, daß mit wachsendem  $n_b$ , bei einer festen Zahl von Messungen, die Zahl der bins sinkt, und damit der statistische Fehler des bestimmten Fehlers steigt.

Eine genauere Bestimmung des statistischen Fehlers erlaubt die Analyse der Autokorrelationszeiten.

Beginnen wir unsere Betrachtung mit der Varianz der bins  $\hat{A}$ . Die Zahl der bins

sei  $M = N/n_b$ .

$$\begin{aligned}
\sigma^2(\hat{A}) &= \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{j=1}^M (\bar{A} - \hat{A}_j)^2 \\
&= \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{j=1}^M (\hat{A}_j^2 - \bar{A}^2) \\
&\quad \text{Nun setzen wir gemäß der Definition von } \hat{A} \text{ ein und multiplizieren das Quadrat aus:} \\
&= \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{j=1}^M \left( \frac{1}{n_b^2} \sum_{l=(j-1)n_b+1}^{jn_b} \sum_{k=(j-1)n_b+1}^{jn_b} A_l A_k - \bar{A}^2 \right) \\
&= \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{j=1}^M \left( \frac{1}{n_b^2} \sum_{l=(j-1)n_b+1}^{jn_b} \sum_{t=(j-1)n_b+1-l}^{jn_b-l} A_l A_{l+t} - \bar{A}^2 \right) \\
&\approx \frac{1}{n_b} \sum_{t=-\infty}^{\infty} (\langle A_i A_{i+t} \rangle - \langle A \rangle^2) \tag{77}
\end{aligned}$$

Dabei verwenden wir, daß zum einen die Wahrscheinlichkeitsverteilung der  $A_i$  unabhängig von  $i$  ist, und daß  $\langle A_i A_{i+t} \rangle - \langle A \rangle^2$  exponentiell schnell in  $t$  zerfällt.

Man führt die normierte **Autokorrelationsfunktion** ein

$$\rho_A(t) := \frac{\langle A_i A_{i+t} \rangle - \langle A \rangle^2}{\langle A^2 \rangle - \langle A \rangle^2} \tag{78}$$

Die **“integrierte” Autokorrelationszeit** ist durch

$$\tau_A := \frac{1}{2} \sum_{t=-\infty}^{\infty} \rho_A(t) = \frac{1}{2} + \sum_{t=1}^{\infty} \rho_A(t) \tag{79}$$

gegeben.

Damit schreibt sich der statistische Fehler wie

$$\sigma^2(\bar{A}) = \frac{2\tau_A}{N} \sigma^2(A) \tag{80}$$

Dies können wir so interpretieren, daß sich die Zahl der effektiv unabhängigen Messungen von  $N$  auf  $N/(2\tau_A)$  reduziert.

In der Praxis wird neben  $\sigma^2(A)$  auch  $\tau_A$  numerisch aus den Daten bestimmt. Dabei muß die Summation über  $t$  eingeschränkt werden. Es hat sich als sinnvoll erwiesen,  $t_{max} \approx 6\tau$  zu wählen.

Neben der integrierten Autokorrelationszeit findet man in der Literatur auch die sogenannte exponentielle Autokorrelationszeit, die durch

$$\rho_A(t) \sim \exp\left(-\frac{t}{\tau_{exp}}\right) \tag{81}$$

für große  $t$  gegeben ist.



## 5 Physik des Phasenübergangs im Ising-Modell

$$\langle m \rangle \sim t^\beta \quad (82)$$

Kollision in der Notation :  $\beta$  wird sowohl für den kritischen Exponenten als auch für  $1/T$  verwendet.

Die magnetische Suszeptibilität ist die Ableitung der Magnetisierung nach dem äußeren Feld.

Wenn wir den Boltzmannfaktor als  $\exp(\beta \sum_{\langle x,y \rangle} s_x s_y + h \sum_x s_x)$  schreiben, dann erhalten wir

$$\chi = \frac{\partial \langle m \rangle}{\partial h} = V \left( \langle m^2 \rangle - \langle m \rangle^2 \right) . \quad (83)$$

Die Herleitung ist einfach:

$$\begin{aligned} \frac{\partial \langle m \rangle}{\partial h} &= \frac{(Z^{-1} \sum_s \exp(\beta \sum_{\langle x,y \rangle} s_x s_y + h \sum_x s_x) \sum_x s_x / V)}{\partial h} \\ &= Z^{-1} \sum_s \exp(\beta \sum_{\langle x,y \rangle} s_x s_y + h \sum_x s_x) (\sum_x s_x)^2 / V \\ &\quad + \left( Z^{-1} \sum_s \exp(\beta \sum_{\langle x,y \rangle} s_x s_y + h \sum_x s_x) \sum_x s_x / V \right)^2 \\ &= V \left( \langle m^2 \rangle - \langle m \rangle^2 \right) . \end{aligned} \quad (84)$$

Für  $h = 0$  hat man die Schwierigkeit, daß es auf dem endlichen Gitter keine spontane Magnetisierung gibt. Deshalb setzt man (zumindest für  $\beta > \beta_c$ ) wieder Beträge bei der Magnetisierung ein.

Eine weitere interessante Größe ist die spezifische Wärme. Sie gibt die Ableitung der inneren Energiedichte nach der Temperatur.

$$C = \frac{\partial \langle E \rangle}{\partial T} = \frac{1}{T^2} V \left( \langle E^2 \rangle - \langle E \rangle^2 \right) . \quad (85)$$

Eine wichtige Rolle in der Theorie der Phasenübergänge spielt die Korrelationslänge  $\xi$ . Sie ist die Zerfallskonstante der Korrelationsfunktion

$$G(x, y) = \langle s_x s_y \rangle - \langle s_x \rangle \langle s_y \rangle \quad (86)$$

mit

$$G(x, y) \sim \exp\left(-\frac{|x-y|}{\xi}\right) . \quad (87)$$

In der Praxis verwendet man die sogenannte "time-slice" (Zeitschichten) Korrelationsfunktion zur Bestimmung der Korrelationslänge.

Man definiert den Mittelwert des Spins über eine  $D - 1$  Linie, Ebene, ... des Gitters:

$$S_{x_0} = L^{D-1} \sum_{x_1, \dots, x_D} s_x . \quad (88)$$

Dann erhält man die dazugehörige Korrelationsfunktion durch

$$G(y) = \langle S_x S_{x+y} \rangle - \langle S_x \rangle \langle S_{x+y} \rangle \quad (89)$$

Für diese Art von Korrelationsfunktion kann man mit dem Transfermatrixformalismus zeigen, daß

$$G(y) = \sum_i c_i \exp\left(-\frac{y}{\xi_i}\right) \quad (90)$$

gilt.

Sei  $t = |\beta - \beta_c|$ . Am Phasenübergang verhalten sich die eingeführten Größen wie

$$\chi \sim t^{-\gamma}, \quad (91)$$

$$C \sim t^{-\alpha} \quad (92)$$

und

$$\xi \sim t^{-\nu} \quad (93)$$

Aus der exakten Lösung des 2D Ising-Modells erhält man:

$$\beta = 0.125, \gamma = 1.75, \nu = 1, \alpha = 0.$$

In 3 Dimensionen:

$$\beta = 0.32648(18), \gamma = 1.2371(4), \nu = 0.63002(23), \alpha = 0.1099(7).$$

(Massimo Campostrini, Andrea Pelissetto, Paolo Rossi, Ettore Vicari, cond-mat/9905078, Phys.Rev. E60 (1999) 3526-3563, Hochtemperaturentwicklung mit Information aus Monte-Carlo-Simulationen)

In 4 Dimensionen und größer:

$$\beta = 0.5, \gamma = 1, \nu = 0.5, \alpha = 0. \text{ (Meanfield Exponenten)}$$

## 5.1 Spontane Symmetriebrechung

Für ein endliches System ist die Magnetisierung bei verschwindendem äusseren Feld bei jeder Temperatur 0. Dies sieht man recht einfach daran, daß beim Multiplizieren aller Spins mit  $-1$  der Wert der Energie und damit der Boltzmannfaktor erhalten bleibt, aber die Magnetisierung das Vorzeichen wechselt. Daher heben sich bei der Summe über alle Konfigurationen von  $\exp(-H) \sum_x s_x$  die Summanden paarweise weg.

Einen von Null verschiedenen Wert der Magnetisierung erhält man nur bei der Grenzwertbildung

$$\langle m \rangle |_{h=0} := \lim_{h \rightarrow 0} \lim_{L \rightarrow \infty} \frac{1}{V} \frac{1}{Z} \sum_{conf} \exp(-H) \sum_x s_x \quad (94)$$

Um das Problem besser zu verstehen, betrachten wir das Histogramm der Magnetisierung:

$$P(m) := \frac{1}{Z} \sum_{conf} \exp(-\beta H) \delta\left(m, \frac{1}{V} \sum_x s_x\right) \quad (95)$$

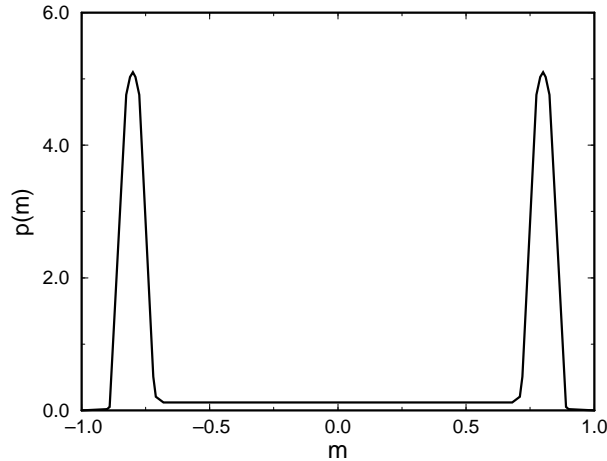


Abbildung 5: Schematische Darstellung des Histogramms der Magnetisierung bei tiefen Temperaturen  $\beta > \beta_c$ . Zur Vereinfachung der Darstellung sind kontinuierliche Werte für die Magnetisierung angenommen, und damit eine Wahrscheinlichkeitsdichte angegeben.

Für  $\beta < \beta_c$  und  $h = 0$  hat  $P(m)$  sein Maximum bei  $m = 0$ .

Auf der anderen Seite, für  $\beta > \beta_c$ , bekommt das Histogramm eine Doppelspitzenstruktur. Die Beiträge zwischen den Spitzen gehen exponentiell mit dem Volumen nach unten. Schematisch ist dies in Abbildung 5 dargestellt.

Mit diesem Histogramm lässt sich auch die zeitliche Entwicklung der Magnetisierung bei  $\beta = 0.48$  in den Abbildungen 3 und 4 verstehen. In einem elementaren Schritt des Metropolis-Algorithmus ändert sich  $\sum_x s_x$  nur um  $\pm 2$ . Auf dem Weg von positiver zu negativer Magnetisierung muß  $m = 0$  durchlaufen werden. Die Wahrscheinlichkeit, daß  $m = 0$  angenommen wird, ist sehr klein. Daher wechselt das Vorzeichen erst nach langer Monte-Carlo-Zeit. Insbesondere wächst die mittlere Zeit für den Vorzeichenwechsel exponentiell mit dem Volumen des Gitters. In Abbildung 4 mit  $L = 8$  wechselt das Vorzeichen noch recht häufig, wogegen für  $L = 32$  in Abbildung 3 sich kein Vorzeichenwechsel ereignet.

In der Monte-Carlo-Simulation wäre es sehr aufwendig, die Limesbildung in Gleichung (94) nachzubilden. Als beste Möglichkeit erweist sich

$$\langle m \rangle |_{h=0, L=\infty} := \lim_{L \rightarrow \infty} \frac{1}{V} \frac{1}{Z} \sum_{conf} \exp(-H) \left| \sum_x s_x \right| \quad (96)$$

zu bestimmen. Für die oben diskutierte Form des Histogramms der Magnetisierung führen die Definitionen (94) und (96) zum gleichen Resultat.

In Tabelle 4 ist der Erwartungswert des Betrages der Magnetisierung für quadratische Gitter mit periodischen Randbedingungen für  $L = 3, 4$  und  $5$  für verschiedene

Werte von  $\beta$  gegeben. Zum Vergleich ist der exakte Wert für den thermodynamischen Limes in der letzten Zeile aufgeführt.

Wir sehen, daß die Abweichung des Resultats der endlichen Gitter von dem thermodynamischen Limes umso grösser wird, je näher wir an den Phasenübergang heranrücken.

Tabelle 4: Erwartungswert des Betrages der Magnetisierung für  $3^2, 4^2$  und  $5^2$  Gitter bei  $h = 0$ . Zum Vergleich wird die exakte Lösung für den thermodynamischen Limes in der letzten Zeile gegeben.

$L$	0.48	0.50	0.55	0.60	0.70
3	0.910397	0.925985	0.954502	0.972103	0.989301
4	0.898543	0.918943	0.953589	0.972867	0.989969
5	0.891143	0.915209	0.953511	0.973291	0.990123
$\infty$	0.877523	0.911319	0.953945	0.973609	0.990163

## 5.2 Endliche Systeme

Potenzgesetze gelten für den **thermodynamischen Limes** (d.h. für das unendlich große Volumen).

Da wir auf dem Rechner nur endliche Systeme betrachten können, und auch die Gittergröße nicht über sehr große Bereiche variieren können, ist ein gutes Verständnis der Volumen- (Gittergrößen)-Abhängigkeit nötig.

Die Zustandssumme ist für endliche Systeme eine analytische Funktion der Temperatur und des äußeren Feldes. Analytisch heißt, daß die Funktion stetig und beliebig oft differenzierbar ist. Die Zustandssumme im endlichen System ist eine endliche Summe von Exponentialfunktionen. Da die Exponentialfunktion analytisch ist, muß auch die Zustandssumme analytisch sein. Entsprechend sind auch Observablenwartungswerte im endlichen Gitter analytisch. Die Singularitäten beim Phasenübergang, so wie sie in den Potenzgesetzen zum Ausdruck kommen, können also nur durch den Grenzübergang  $Volumen \rightarrow \infty$  zustande kommen.

Auf einem endlichen Gitter kann die magnetische Suszeptibilität nicht unendlich werden. Jedoch wird die magnetische Suszeptibilität in der Nähe des Phasenübergangs auf dem endlichen System ein Maximum annehmen. Der Wert des Maximums wächst mit wachsender Gittergröße.

Hier ist es nützlich, eine weitere Methode der Datenanalyse einzuführen. Bisher hatten wir das “simple sampling” und das “importance sampling” besprochen. Es ist auch möglich, Mischformen beider Methoden zu verwenden:

$$\langle A \rangle (\beta) \approx \frac{\sum_i \exp(-(\beta - \beta_s)H_i) A_i}{\sum_i \exp(-(\beta - \beta_s)H_i)}, \quad (97)$$

wobei  $\beta_s$  das bei der Simulation verwendete  $\beta$  ist. Diese Methode (“Umwichten”) erlaubt es auch für  $\beta$  in der Nähe von  $\beta_s$ , die Erwartungswerte von Observablen

## 2D Ising Model

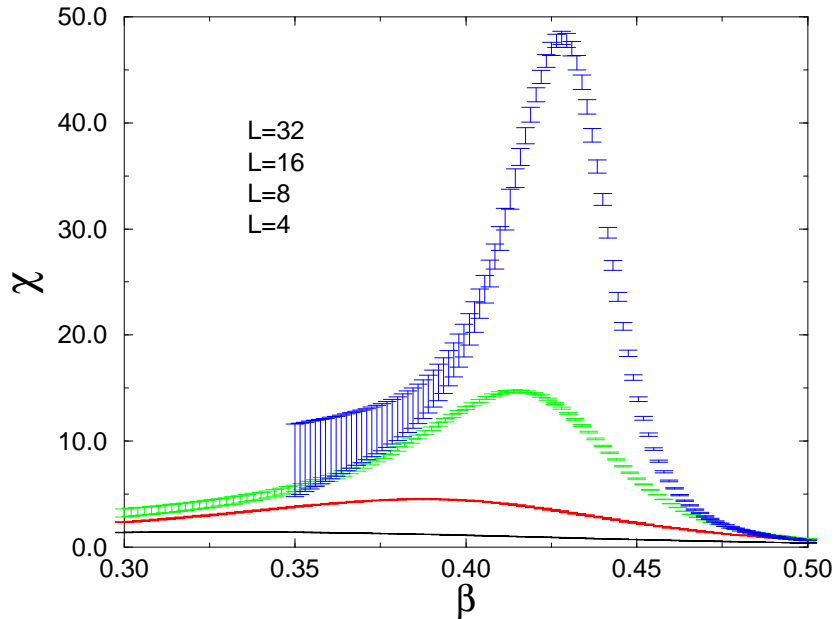


Abbildung 6:

zu bestimmen. Mit wachsendem  $|\beta_s - \beta|$  steigt jedoch der statistische Fehler. Diese Methode ist recht einfach, wurde aber erst mit der Arbeit von Swendsen und Ferrenberg, Phys.Rev.Lett. 61 (1988) 2635 populär.

Ein Beispiel ist in Abbildung 6 gegeben. Hier wird das 2-dimensionale Ising-Modell bei  $\beta = 0.4406868$  für die Gittergrößen  $4^2$ ,  $8^2$ ,  $16^2$  und  $32^2$  simuliert. Es werden jeweils 100000 Messungen von  $-H$  und  $M$  durchgeführt.

Dann wird mit Hilfe des Umwichtverfahrens für 100  $\beta$ -Werte in der Nähe von  $\beta_c$  die magnetische Suszeptibilität ausgerechnet. Dazu wird separat  $\langle |m| \rangle$  und  $\langle m^2 \rangle$  für die einzelnen  $\beta$ -Werte ausgerechnet.

Je größer  $|\beta - \beta_s|$ , desto größer wird der statistische Fehler. Die Ursache dafür ist ähnlich wie bei dem Problem beim simple sampling. Das Phänomen ist recht deutlich bei  $L = 32$  zu sehen.

Trotz dieser Limitierung ist die Methode der Umwichtung sehr nützlich. Ausgehend von einer einzigen Simulation können wir das Maximum der magnetischen Suszeptibilität bestimmen.

Für die Fehlerrechnung wurde eine Variante des Binnings verwendet: Das Jackknife-Verfahren. Diese Methode ist für einfache Observablen dem Binning äquivalent. Ein Vorteil des Jackknife-Verfahrens ergibt sich bei zusammengesetzten Größen wie z.B. der magnetischen Suszeptibilität oder auch bei dem Umwichtverfahren.

Im Jackknife-Verfahren werden die Daten betrachtet, die zu einzelnen Bins kom-

plementär sind.

Für den Erwartungswert einer Observablen  $A$  sieht das wie folgt aus:

$$\tilde{A}_j = \frac{1}{N - n_b} \left( \sum_{i=1}^N A_i - \sum_{i=(j-1)n_b+1}^{jn_b} A_i \right) . \quad (98)$$

Der statistische Fehler (d.h. die Wurzel der Varianz ) des Mittelwertes ergibt sich als

$$\sigma_{\tilde{A}} = \sqrt{m \left[ \frac{1}{m} \sum_{j=1}^m \tilde{A}_j^2 - \left( \frac{1}{m} \sum_{j=1}^m \tilde{A}_j \right)^2 \right]} . \quad (99)$$

Man beachte, daß im Gegensatz zu dem Binning hier  $\sqrt{m}$  im Zähler auftaucht.  $m = N/n_b$  ist hier die Zahl der Bins.

Möchten wir nun den statistischen Fehler einer Größe betrachten, die eine nicht-lineare Funktion  $F(\langle A \rangle, \langle B \rangle, \dots) \approx F(\bar{A}, \bar{B}, \dots)$  mehrerer Observablen ist, so geht man wie folgt vor:

Zuerst berechnet man

$$\tilde{F}_j = F(\tilde{A}_j, \tilde{B}_j, \dots) . \quad (100)$$

Dabei sind die  $\tilde{A}_j$  usw. wie in Gleichung (98) bestimmt. Nun ergibt sich der statistische Fehler von  $F$  als

$$\sigma_{\tilde{F}} = \sqrt{m \left[ \frac{1}{m} \sum_{j=1}^m \tilde{F}_j^2 - \left( \frac{1}{m} \sum_{j=1}^m \tilde{F}_j \right)^2 \right]} . \quad (101)$$

Man beachte, daß der statistische Fehler von  $F$  **nicht** mit der einfachen Fehlerfortpflanzungsregel

$$\sigma_{\tilde{F}} = \sqrt{\left( \frac{\partial F}{\partial \langle A \rangle} \sigma_{\tilde{A}} \right)^2 + \left( \frac{\partial F}{\partial \langle B \rangle} \sigma_{\tilde{B}} \right)^2 + \dots} . \quad (102)$$

Da die Observablen-Messungen von  $A$ ,  $B$  ... miteinander korreliert sind:

$$\langle AB \rangle - \langle A \rangle \langle B \rangle \neq 0 . \quad (103)$$

Ähnlich wie beim binning müssen wir auch hier die Fehleranalyse für eine Reihe von  $n_b$  durchführen und uns von der Stabilisierung dies Fehlers bei wachsendem  $n_b$  überzeugen.

## 6 Finite Size Scaling

Für endliche Systeme kann man mittels der Renormierungsgruppe sogenannte "Finite Size Scaling"-Gesetze herleiten. Für die magnetische Suszeptibilität zum Beispiel ergibt sich

$$\chi(L, \beta) = g(L/\xi) L^{\gamma/\nu} . \quad (104)$$

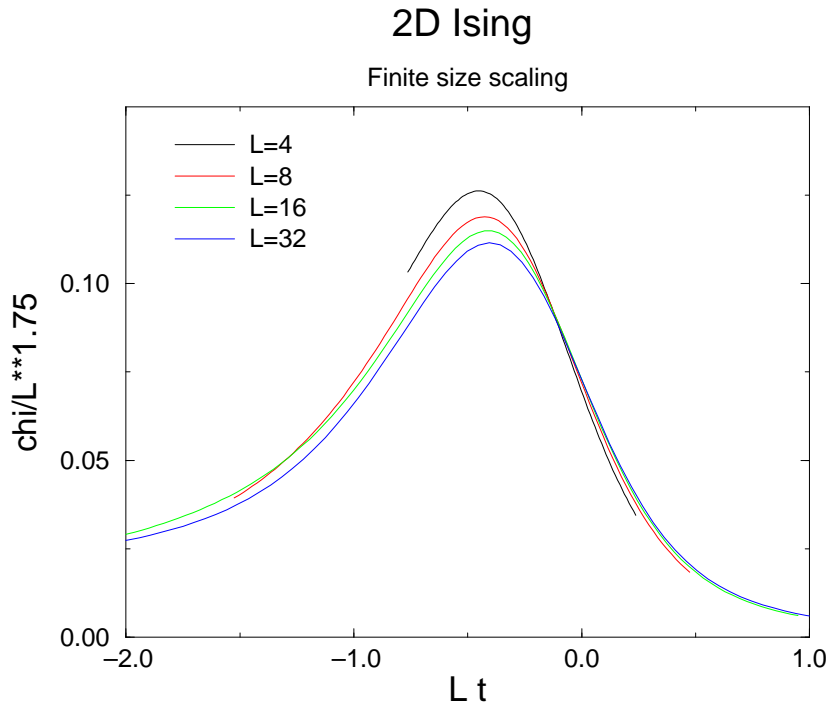


Abbildung 7:

Dahinter steckt, daß in der Nähe des Phasenübergangs die mikroskopischen Details des Modells keine Rolle mehr spielen. Wichtige Längenskalen sind nur noch  $L$  und  $\xi$ .

Am kritischen Punkt selbst ergeben sich wieder Potenzgesetze

$$\chi(L, \beta_c) \propto L^{\gamma/\nu} \quad . \quad (105)$$

Als Regel kann man sich merken, daß im Vergleich zu den Potenzgesetzen für den thermodynamischen Limes  $t$  durch  $L$  ersetzt ist, und der Exponent mit  $-1/\nu$  geteilt wird.

Für die folgende Form des Finite-Size-Scaling-Gesetzes braucht man die Korrelationslänge  $\xi$  nicht zu kennen:

$$\chi(L, \beta) = g(L/\xi) L^{\gamma/\nu} = \tilde{g}(Lt^\nu) L^{\gamma/\nu} \quad (106)$$

Bei der Umformung haben wir das Potenzgesetz  $\xi \propto t^{-\nu}$  verwendet.

Zur Überprüfung dieses Verhaltens sind in der Abbildung 7 die Daten aus der Abbildung 6 durch  $L^{\gamma/\nu}$  geteilt und als Funktion von  $L^{1/\nu}t$  aufgetragen. Wenn das Finite-Size-Scaling-Gesetz erfüllt ist, so müssen die Kurven für die verschiedenen Gittergrößen aufeinanderfallen. Wir erkennen, daß dies bis auf kleine Abweichungen in der Tat erfüllt ist.

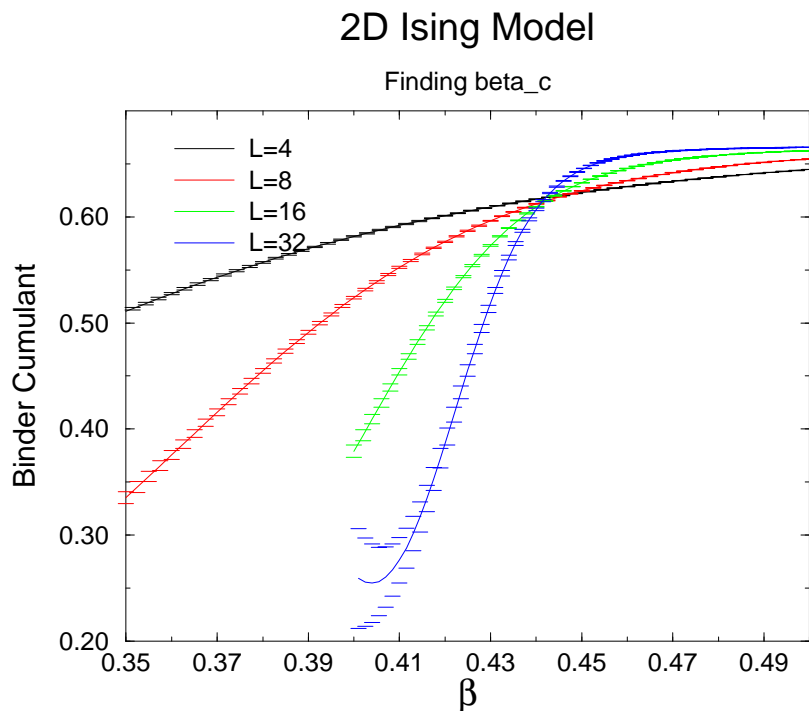


Abbildung 8:

Eine gute Möglichkeit, die kritische Temperatur zu bestimmen, bietet die sogenannte Binder Kumulante.

$$U(L, \beta) = 1 - \frac{1}{3} \frac{\langle m^4 \rangle}{\langle m^2 \rangle^2} . \quad (107)$$

Für große Gitter und große  $\xi$  gilt

$$U(L, \beta) = \tilde{U}(L/\xi) . \quad (108)$$

Das bedeutet, daß die Längenskala der Gitterkonstante unwichtig wird. Da am Phasenübergang  $\xi = \infty$  ist, nimmt die Binderkumulante für  $\beta_c$  einen von  $L$  unabhängigen Wert an. Darauf beruht nun die Methode zur Bestimmung von  $\beta_c$ : Man bestimmt für eine Reihe von Gittergrößen die Binderkumulante als Funktion von  $\beta$ . Dazu eignet sich die Umwichtmethode sehr gut. Der Schnittpunkt dieser Kurven gibt nun einen Schätzwert für  $\beta_c$ . Dies ist am Beispiel des 2D Ising-Modells in Abbildung 8 gezeigt.

## 7 Critical slowing down

Wir erinnern uns, daß die Zahl der effektiv unabhängigen Messungen durch  $N_{eff} = N/(2\tau)$  gegeben ist. Je größer  $\tau$ , desto kleiner ist die Zahl der effektiv unabhängigen



Messungen. Dies wiederum bedingt einen größeren statistischen Fehler. Ein guter Algorithmus ist daher durch kleine  $\tau$  charakterisiert. (Natürlich muß auch die CPU-Zeit berücksichtigt werden, welche für einen sweep benötigt wird.)

Für lokale (lokal heißt hier, daß in einem Elementarschritt nur an einem Gitterpunkt etwas geändert wird) Algorithmen wie den Metropolis-Algorithmus wächst die Autokorrelationszeit an, wenn man sich dem kritischen Punkt nähert. Für  $\xi \ll L$  gilt

$$\tau \sim \xi^z \sim t^{-\nu z} . \quad (109)$$

Allgemeiner gilt die Finite-Size-Scaling-Form:

$$\tau \sim f(L/\xi)L^z . \quad (110)$$

Mit dem Argument, daß sich Veränderungen in einer Konfiguration diffusionsartig ausbreiten, erhält man  $z \approx 2$  für den lokalen Metropolis-Algorithmus.

In Abbildung 9 sind die Autokorrelationsfunktionen für die Energie und das Quadrat der Magnetisierung für das 2D Ising-Modell, simuliert am kritischen Punkt, abgebildet. Man erkennt, daß die Autokorrelationen mit wachsendem  $L$  immer größer werden. Die integrierten  $\tau$ s, die sich aus diesen Kurven ergeben, sind in Tabelle 3 zusammengefasst.

Man sieht, daß zwischen  $L = 16$  und  $L = 32$  sich die  $\tau$  s fast vervierfachen. In der Literatur kann man hierzu weit genauere Zahlen finden, z.B. cond-mat/9601059 ( $z = 2.1665(12)$ ).

Tabelle 5: Integrierte Autokorrelationszeiten für die Energie und das Quadrat der Magnetisierung für den Metropolis-Algorithmus angewendet auf das 2D Ising-Modell am Phasenübergang; jeweils 100000 Messungen.

$L$	$\tau_E$	$\tau_\chi$
4	1.24(2)	1.23(2)
8	2.45(7)	2.95(9)
16	6.1(3)	9.5(5)
32	16(2)	34(6)

Eine Idee zur Bekämpfung des critical slowing downs ist, in einem größeren Teil des Gitters die Spins gleichzeitig, d.h. in einem elementaren Schritt, zu verändern.

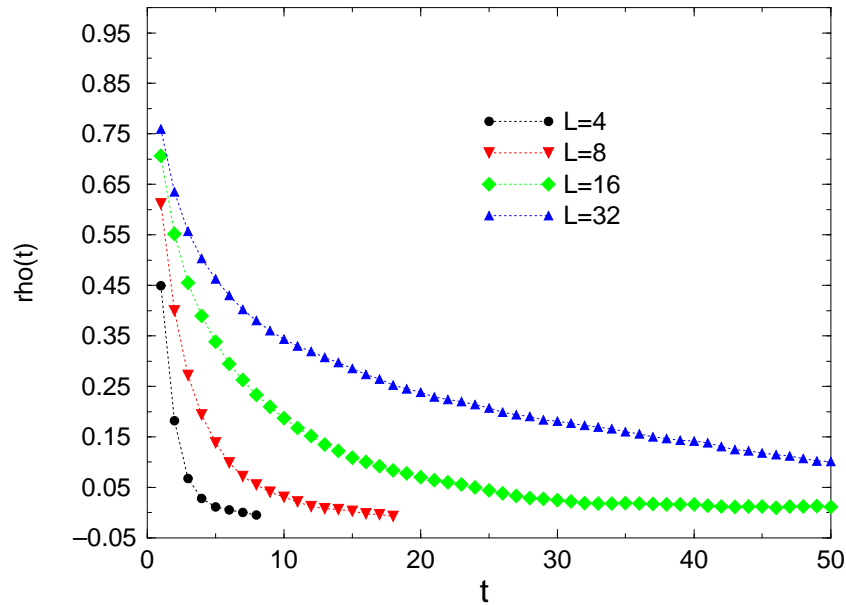
Als einfachstes könnte man sich überlegen, daß in einem Vorschlag alle Spins eines  $l^D$  Teilgitters das Vorzeichen wechseln.

Das Problem dabei ist jedoch, daß im Mittel mehr benachbarte Spins parallel zueinander stehen als antiparallel. Daher ergibt sich bei dem gerade beschriebenen Vorschlag meistens eine Energievergrößerung proportional zu der Oberfläche des Teilgitters in dem die Spins geflippt werden sollen. Das heißt, wenn man sich die Akzeptanzbedingung des Metropolis nocheinmal anschaut, daß die Akzeptanzwahrscheinlichkeiten exponentiell mit der Oberfläche des Teilgitters sinken.

Wir müssen also nach einer besseren Implementierung unserer Idee Ausschau halten.

# Autokorrelationsfunktion

2D Ising , Energie



# Autokorrelationsfunktion

2D Ising , chi

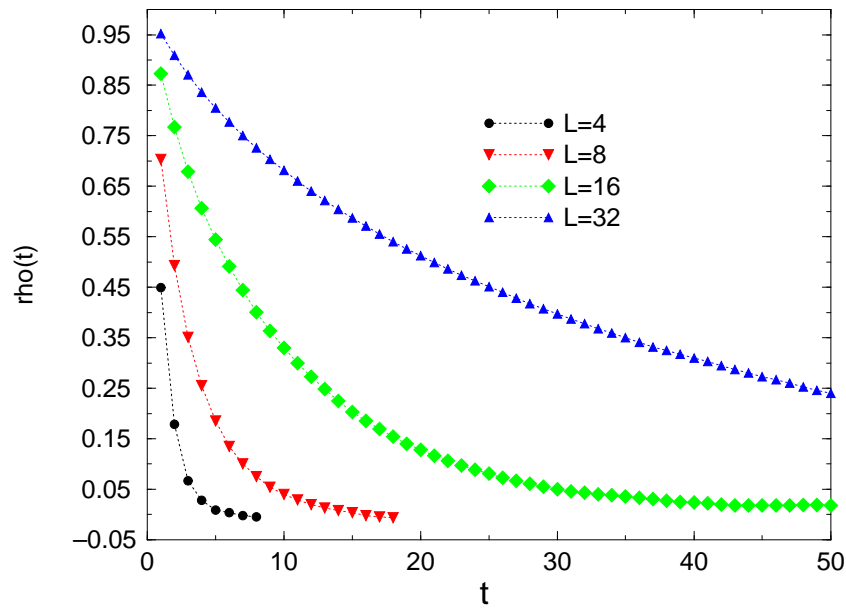


Abbildung 9:

## 8 Cluster-Algorithmus

Die Lösung des oben beschriebenen Problems wurde mit Hilfe des sogenannten Cluster-Algorithmus erreicht. Die erste Arbeit dazu ist: R.H. Swendsen and J-Sh. Wang, Phys.Rev.Lett. 58 (1987) 86.

Der Cluster-Algorithmus beruht auf der Darstellung des Ising-Modells (und q-Zustands-Potts-Modellen) als gewichtete Perkolations-Modelle. Diese Darstellung geht auf C.M. Fortuin and P.W. Kasteleyn, Physica 57 (1972) 536 zurück.

$$\begin{aligned}
 Z &= \sum_{s=\pm 1} \exp(\beta \sum_{\langle xy \rangle} s_x s_y) \\
 &= \sum_{s=\pm 1} \prod_{\langle xy \rangle} \exp(\beta s_x s_y) \\
 &= \exp(-DV\beta) \sum_{s=\pm 1} \prod_{\langle xy \rangle} (1 + \delta_{s_x, s_y} (\exp(+2\beta) - 1)) \\
 &= \exp(-DV\beta) \sum_{s=\pm 1} \sum_{d=0,1} \prod_{\langle xy \rangle} ((1 - d_{\langle xy \rangle}) + d_{\langle xy \rangle} \delta_{s_x, s_y} (\exp(+2\beta) - 1))
 \end{aligned} \tag{111}$$

Hier geht es in der Rechnung von Fortuin und Kasteleyn weiter, indem die Summation über die Spins ausgeführt wird und man bei einem Modell anlangt, das lediglich die  $d$  als Freiheitsgrade hat. (Gewichtetes Perkolations-Modell).

Um den Cluster-Algorithmus zu beschreiben, ist es nötig, bei der Version zu bleiben, die sowohl die ursprünglichen Spins  $s$  als auch die neu eingeführten  $d$  als Freiheitsgrade hat.

Die grobe Struktur des Algorithmus ist es nun, zunächst den  $d$  bei festgehaltenen  $s$  neue Werte zu geben, und in einem zweiten Teilschritt die Spins  $s$  bei festgehaltenen  $d$  neu zu besetzen.

Als erstes möchte ich die Auffrischung der  $d$  besprechen. Zunächst sehen wir, daß das relative Gewicht von  $d_{\langle xy \rangle} = 0$  und  $d_{\langle xy \rangle} = 1$  unabhängig von den Werten der  $d$  auf anderen Kanten ist. Wir können also die  $d_{\langle xy \rangle}$  alle unabhängig von einander neu setzen. Eine Abhängigkeit besteht lediglich von den Spins  $s_x$  und  $s_y$  an den Enden der Kante  $\langle xy \rangle$ .

**Fall 1:**  $s_x s_y = -1$

Für das Gewicht von  $d_{\langle xy \rangle}$  ergibt sich:

$$1 - d_{\langle xy \rangle} + d_{\langle xy \rangle} \delta_{s_x, s_y} (\exp(+2\beta) - 1) = 1 - d_{\langle xy \rangle} \tag{112}$$

Die Sache ist also sehr einfach.  $d_{\langle xy \rangle} = 0$  hat das Gewicht 1 und  $d_{\langle xy \rangle} = 1$  das Gewicht 0.

**Fall 2:**  $s_x s_y = 1$

Für das Gewicht von  $d_{\langle xy \rangle}$  ergibt sich:

$$1 - d_{\langle xy \rangle} + d_{\langle xy \rangle} \delta_{s_x, s_y} (\exp(+2\beta) - 1) = 1 - d_{\langle xy \rangle} + d_{\langle xy \rangle} (\exp(+2\beta) - 1) \quad (113)$$

Also hat  $d_{\langle xy \rangle} = 0$  die Wahrscheinlichkeit

$$p(d_{\langle xy \rangle} = 0) = \frac{1}{\exp(+2\beta) - 1 + 1} = \exp(-2\beta) \quad (114)$$

und

$$p(d_{\langle xy \rangle} = 1) = 1 - \exp(-2\beta) \quad (115)$$

Der Update der  $d$ 's sieht nun so aus, daß man auf allen Kanten die  $d$ 's neu, entsprechend den oben gegebenen Wahrscheinlichkeiten, setzt. Zusammengefasst ist diese Wahrscheinlichkeit mit

$$p(d_{\langle xy \rangle} = 0) = \exp(-\beta(1 + s_x s_y)) \quad (116)$$

und

$$p(d_{\langle xy \rangle} = 1) = 1 - p(d_{\langle xy \rangle} = 0) \quad (117)$$

gegeben.

Im **zweiten Schritt** werden bei festem  $d$  **neue Spinvariablen** erzeugt.

Dazu betrachten wir jetzt die beiden Fälle  $d_{\langle xy \rangle} = 0$  und  $d_{\langle xy \rangle} = 1$ .

**Fall 1:**  $d_{\langle xy \rangle} = 1$

Es ergibt sich :

$$1 - d_{\langle xy \rangle} + d_{\langle xy \rangle} \delta_{s_x, s_y} (\exp(+2\beta) - 1) = \delta_{s_x, s_y} (\exp(+2\beta) - 1) \quad (118)$$

Wegen  $\delta_{s_x, s_y}$  muß  $s_x s_y = 1$  gelten.

**Fall 2:**  $d_{\langle xy \rangle} = 0$

$$1 - d_{\langle xy \rangle} + d_{\langle xy \rangle} \delta_{s_x, s_y} (\exp(+2\beta) - 1) = 1 = \delta_{s_x, s_y} + \delta_{s_x, -s_y} \quad (119)$$

Das heißt, daß  $s_x s_y = 1$  und  $s_x s_y = -1$  das gleiche Gewicht haben.

Nun müssen wir diskutieren, welche Konfigurationen unter der Forderung die sich aus dem Fall  $d_{\langle xy \rangle} = 1$  ergibt, erlaubt sind.

Dazu führen wir den Begriff **Cluster** ein. Zwei Punkte des Gitters gehören zum gleichen Cluster, wenn es eine Verbindungslinie zwischen den beiden Punkten gibt, so daß für alle Kanten  $\langle xy \rangle$ , die auf dieser Verbindungslinie liegen,  $d_{\langle xy \rangle} = 1$  gilt.

Man erkennt, daß alle Spins die zu einem Cluster gehören gleich sein müssen. Die verbleibenden Freiheitsgrade sind also die Vorzeichen der Spins der einzelnen Cluster. Dabei haben beide Vorzeichen das gleiche Gewicht.

Im Algorithmus von Swendsen und Wang wird nun mit gleicher Wahrscheinlichkeit für jedes Cluster der Wert  $-1$  oder  $+1$  neu gewählt.

## 8.1 Bestimmung der Cluster

Die effiziente Bestimmung der Cluster ist ein nichttriviales Problem. Diese Aufgabe ergibt sich auch bei dem Perkolations-Modell. Außerhalb der Physik spielt die Berechnung von Clustern z.B. eine Rolle bei Bilderkennungsverfahren.

In der Literatur werden hauptsächlich zwei Verfahren diskutiert:

- Hoshen-Kopelman, J.Hoshen and R.Kopelman, Phys.Rev.B 14 (1976) 3438
- Depth-First Search ("Ants in the labyrinth").

Wir werden uns hier auf die Diskussion des zweiten Verfahrens beschränken. (Weiterführend siehe CP II Vorlesung.)

Zuerst wird ein Gitterpunkt  $x$  ausgewählt, der noch keinem Cluster zugeordnet ist. Als nächstes betrachtet man die Nachbarn  $y$  dieses Gitterpunktes. Falls  $d_{<xy>} = 1$ , wird der Gitterpunkt als dem Cluster zugehörig eingeordnet. Die neuen Mitglieder des Clusters werden in einer Liste vermerkt. Nun geht die Suche mit den in dieser Liste stehenden Gitterpunkten weiter. Wenn kein Gitterpunkt mehr in der Liste steht, von dem aus man nach neuen Clusterpunkten gesucht hat, ist man mit der Konstruktion des Clusters fertig.

Pseudo-C code für die Bestimmung eines Clusters.

```
ineu=1;
ialt=0;
position[ineu]=V*g05caf();
while(ialt < ineu)
{
  ialt++;
  for(alles Nachbarn von position[ialt])
  {
    if(d_{Nachbar,position[ialt]} = 0) && Nachbar noch nicht im Cluster)
    {
      ineu++;
      position[ineu]=Nachbar;
      Kennzeichne Nachbar als Cluster-mitglied;
    }
  }
}
```

Neben dem Swendsen-Wang-Algorithmus gibt es weitere Varianten des Cluster-Algorithmus:

Die wichtigste Variante ist der Single-Cluster-Algorithmus, der 1989 von U.Wolff vorgeschlagen wurde.

COLLECTIVE MONTE CARLO UPDATING FOR SPIN SYSTEMS  
Phys.Rev.Lett.62:361,1989.

Hierbei wird ein Gitterpunkt zufällig ausgewählt, und nur das Vorzeichen aller Spins gewechselt, welche zu dem Cluster gehören, das zu dem ausgewählten Gitterpunkt gehört.

Unten finden Sie ein Programmbeispiel für das 3D Ising-Modell mit dem Single-Cluster. Man beachte, daß die neuen  $d_{\langle xy \rangle}$  unabhängig vom alten Wert der  $d_{\langle xy \rangle}$  bestimmt werden. Deshalb müssen die  $d_{\langle xy \rangle}$  nicht gespeichert werden und können genau dann bestimmt werden, wenn der Wert von  $d_{\langle xy \rangle}$  beim Clusteraufbau benötigt wird.

```

void cluster()
{
int i0,i1,i2,is0,is1,is2,iq0,iq1,iq2;
int iz1,iz2,ipp;

/* bestimme den Startpunkt des Clusteraufbaus */
i0=L0*g05caf();
i1=L1*g05caf();
i2=L2*g05caf();

iz1=-1 ; iz2=0 ;
spin[i0][i1][i2]=-spin[i0][i1][i2];

/*Gehe so lange weiter bis die Liste cc[] abgearbeitet ist */
while(iz1<iz2)
{
  ipp=spin[i0][i1][i2];

/* alle Nachbarn werden abgearbeitet */
  is0=i0+1; if(is0==L0) is0=0; /* periodische Randbedingungen*/
  FREEZE_DELETE(is0,i1,i2)
  iq0=i0-1; if(i0==0) iq0=L0-1; /* periodische Randbedingungen*/
  FREEZE_DELETE(iq0,i1,i2)
  is1=i1+1; if(is1==L1) is1=0; /* periodische Randbedingungen*/
  FREEZE_DELETE(i0,is1,i2)
  iq1=i1-1; if(i1==0) iq1=L1-1; /* periodische Randbedingungen*/
  FREEZE_DELETE(i0,iq1,i2)
  is2=i2+1; if(is2==L2) is2=0; /* periodische Randbedingungen*/
  FREEZE_DELETE(i0,i1,is2)
  iq2=i2-1; if(i2==0) iq2=L2-1; /* periodische Randbedingungen*/
  FREEZE_DELETE(i0,i1,iq2)

  i0=cc[++iz1];

```

```

/* die drei koordinaten i0,i1,i2 werden zum Platz sparen in
   ein Wort geschoben. Dazu werden Befehle verwendet, die
   auf dem level der bits arbeiten.
*/
i1=i0>>10;
i2=i0>>20;
i0=i0&1023;
i1=i1&1023;
}
}

```

Mit dem C-Präprozessor wird eingefügt:

```

#define FREEZE_DELETE(A,B,C)      \
if(spin[A] [B] [C] !=ipp)        \
{                                  \
G05;                               \
if(pd<xx)                          \
{                                  \
spin[A] [B] [C]=ipp;              \
cc[iz2++]=A | (B<<10) | (C<<20);  \
}                                  \
}
}

```

## 9 Anwendung des Cluster-Algorithmus auf weiter Modelle

Seit der Einführung des Cluster-Algorithmus in 1987 gelang es, den Algorithmus auf eine Reihe weiterer Modelle mit Erfolg anzuwenden. Der Trick, der allen Anwendungen zugrunde liegt, ist es, sich für jeden Gitterpunkt einen Freiheitsgrad aus  $-1, 1$  herauszugreifen. Während eines Schrittes des Algorithmus werden nur diese Freiheitsgrade verändert, während alle anderen festgehalten werden.

Wie dies genau gemacht wird, werden wir für eine Reihe von Modellen besprechen: Landau-Ginzburg-Modell auf dem Gitter,  $O(N)$ -invariante nichtlineare  $\sigma$ -Modelle und Oberflächenmodelle.

### 9.1 Landau-Ginzburg-Modell auf dem Gitter

Zunächst wollen wir das Landau-Ginzburg-Modell mit einer Feldkomponente betrachten. Die Hamiltonfunktion ist durch

$$\beta H = \sum_x \left( -2\kappa \sum_{\mu} \phi_x \phi_{x+\hat{\mu}} + \phi_x^2 + \lambda(\phi_x^2 - 1)^2 \right) . \quad (120)$$

In Limes  $\lambda \rightarrow \infty$  wird diese Modell exakt zum Ising-Modell. Für andere Werte von  $\lambda > 0$  liegt das Modell noch in der gleichen Universalitätsklasse wie das Ising-Modell. Das heißt unter anderem, daß die kritischen Exponenten nicht von  $\lambda$  abhängen, sondern immer die Werte des Ising-Modells annehmen.

Wie man nun den Cluster-Algorithmus auf dieses Modell anwenden kann, ist in EMBEDDED DYNAMICS FOR  $\phi^4$  THEORY, R.C. Brower, P. Tamayo, PHYS. REV. LETT. 62 (1989) 1087-1090. beschrieben.

Als Freiheitsgrad für den Cluster-Algorithmus wählt man sich das Vorzeichen des Feldes  $\phi$  aus. Man erkennt, daß sich beim Wechseln des Vorzeichens nur der Term

$$-2\kappa \sum_x \sum_{\mu} \phi_x \phi_{x+\hat{\mu}} \quad (121)$$

ändert. Schreiben wir  $\phi_x = s_x |\phi_x|$  so ergibt sich als effektives  $\beta_{\langle xy \rangle} = 2\kappa |\phi_x| |\phi_y|$  für die  $s$ . Wir können dieses effektive  $\beta_{\langle xy \rangle}$  in Gleichung (116) für  $p(d)$  einsetzen:

$$p(d_{\langle xy \rangle} = 0) = \exp(-\beta_{\langle xy \rangle} (1 + s_x s_y)) = \exp(-2\kappa |\phi_x| |\phi_y| (1 + s_x s_y)) = \min(1, \exp(-4\kappa \phi_x \phi_y)) \quad (122)$$

Damit der update auch ergodisch wird, muß man alternierend mit den Clusterupdates Metropolis-sweeps ausführen.

Ein Vorschlag für einen neuen Wert von  $\phi_x$  kann man erzeugen, indem man  $\phi'_x = \phi_x + s(dr\text{and}48() - 0.5)$  wählt.  $s$  ist nun ein freier Parameter des Algorithmus. Man wählt ihn so, daß sich Akzeptanzraten von etwa 50% einstellen. (Faustregel)

## 9.2 O(N)-invariante nichtlineare $\sigma$ -Modelle

COLLECTIVE MONTE CARLO UPDATING FOR SPIN SYSTEMS, Ulli Wolff, Phys.Rev.Lett.62:361,1989.

Diese Modelle haben die Hamiltonfunktion (bei verschwindendem äusseren Feld)

$$H = -J \sum_{\langle xy \rangle} \vec{s}_x \vec{s}_y \quad (123)$$

Dabei sind  $\vec{s}_x$  Einheitsvektoren in  $R^N$ . Im folgenden setzen wir  $J = 1$ .

Wieder besteht der Trick darin, daß man einen  $-1, 1$  Freiheitsgrad aussucht. Hier geschieht dies, indem man zunächst zufällig eine Richtung im  $R^N$  auswählt. Dann erlaubt man in einem Schritt des Cluster-Algorithmus lediglich Vorzeichenwechsel der Komponente parallel zu dieser Richtung. Es ergibt sich

$$p(d_{\langle xy \rangle} = 0) = \min\left(1, \exp(-2\beta s_x^{(p)} s_y^{(p)})\right) \quad (124)$$

Wählt man für jeden Schritt des Cluster-Algorithmus die Richtung, in der dieser Vorzeichenwechsel geschieht, neu aus, so ist der Algorithmus auch ergodisch.



### 9.3 Oberflächenmodelle

STOCHASTIC CLUSTER ALGORITHMS FOR DISCRETE GAUSSIAN (SOS) MODELS. By H.G. Evertz (Florida State U., SCRI), M. Hasenbusch (Kaiserslautern U.), M. Marcu (Hamburg U. and Tel Aviv U.), K. Pinn (Hamburg U. and Münster U., ITP), S. Solomon (Hebrew U.), Phys.Lett.B254:185-191,1991.

Hier wollen wir kurz sogenannte “Solid on Solid”-(SOS)-Modelle diskutieren.

Diese Modelle sind dadurch gekennzeichnet, daß die Feldvariable  $h_x$  ganzzahlige Werte annimmt. Man stellt sich vor, daß  $h_x$  die Zahl der Atomlagen über einer 2D-Referenzfläche abzählt.

In der Literatur werden verschiedene Hamiltonfunktionen diskutiert. z.B. (Discrete Gaussian SOS)

$$H = \sum_{\langle xy \rangle} (h_x - h_y)^2 \quad (125)$$

oder (Absolute Value SOS)

$$H = \sum_{\langle xy \rangle} |h_x - h_y| \quad (126)$$

Gemeinsam ist all diesen  $H$ 's, daß Rauigkeit, d.h. Schwankungen der  $h$ , zu grösseren  $H$ 's führt. Bei tiefen Temperaturen werden also fast alle  $h_x$  gleich sein; d.h. die Oberfläche glatt sein.

Wird die Temperatur höher, so wird die Oberfläche rauher. Zunächst bleibt diese Rauigkeit jedoch auf kleine Längenskalen beschränkt.

Wird aber eine bestimmte Temperatur überschritten, wird die Oberfläche auf allen Skalen rau. Diese ist die sogenannte “roughening transition”.

Für den Cluster-Algorithmus brauchen wir wieder  $-1, 1$  Freiheitsgrade. Die erhält man hier, indem man Spiegelungen der  $h_x$  an einer Referenzhöhe  $m$  betrachtet.

$$h_x = m + s_x |h_x - m| \quad (127)$$

Damit die  $h_x$  bei der Spiegelung ganzzahlig bleiben, muß  $m$  entweder ganzzahlig oder ganzzahlig  $+1/2$  sein.

Wie wird  $m$  bestimmt?

Ein Gitterpunkt  $x$  wird zufällig herausgegriffen. Man wählt  $m = h_x \pm 0.5$  mit gleicher Wahrscheinlichkeit für  $+1/2$  und  $-1/2$ . Für ganzzahlige  $m$  wählt man  $m = h_x \pm 1$ .

Man beachte, daß bei dieser Wahl von  $m$  die Wahrscheinlichkeit, daß man nach dem Cluster-update das gleiche  $m$  wählt, genauso groß ist, wie vor dem Cluster-update.

### 9.4 Wo scheitert der Cluster-Algorithmus?

Leider haben sich die großen Erwartungen die sich mit dem Cluster-Algorithmus um 1990 verbunden hatten, nicht alle erfüllt. Für eine Reihe wichtiger Modelle wurde keine befriedigende Implementierung des Cluster-Algorithmus gefunden. Insbesondere gelang es nicht, einen guten Cluster-Algorithmus für die QCD auf dem Gitter

zu finden. Ebenso können Spin-Gläser nicht sinnvoll mit dem Cluster-Algorithmus simuliert werden. Man ist zur Einsicht gelangt, daß der Cluster-Algorithmus immer dann scheitert, wenn Frustration eintritt. Frustration bedeutet, daß das Produkt aller Kopplungen  $J_{\langle xy \rangle}$  auf einer geschlossenen Schleife auf dem Gitter negativ ist. Dies ist offensichtlich bei Spin-Gläsern der Fall.

## 10 Metropolis, Wärmebad und Überrelaxation für das klassische Heisenberg-Modell.

Das klassische Heisenberg-Modell ohne äußeres Feld ist durch

$$H = -J \sum_{\langle xy \rangle} \vec{s}_x \vec{s}_y \quad (128)$$

gegeben. (Wir setzen  $J = 1$ ). Dabei ist  $\vec{s}_x$  ein Einheitsvektor aus dem  $R^3$ . Wie oben besprochen kann dieses Modell sehr effizient mit dem Cluster-Algorithmus simuliert werden. Hier wollen wir dieses einfache Modell verwenden, um neben dem Metropolis-Algorithmus weitere lokale Algorithmen kennenzulernen. Diese Algorithmen werden zum Beispiel bei der Simulation der QCD auf dem Gitter verwendet.

Doch zunächst kurz zur Physik:

In 3 Dimensionen hat das Heisenberg-Modell wie das Ising-Modell einen Phasenübergang zweiter Ordnung. Die letzte und genaueste Monte-Carlo-Simulation dieses Modells findet sich in:

Finite size effects on measures of critical exponents in d=3 O(N) models

H. G. Ballesteros, L. A. Fernandez, V. Martin-Mayor, A. Munoz Sudupe (Universidad Complutense de Madrid), cond-mat/9606203, Phys.Lett. B387 (1996) 125-131

Es ergibt sich für das Inverse der kritischen Temperatur:  $\beta_c = 0.693002(12)$ .

Critical Exponents of the Classical Heisenberg Ferromagnet

Christian Holm, Wolfhard Janke

(FU-Berlin and JGU Mainz), hep-lat/9605024, Phys. Rev. Lett. 17 (1997) 2265

Die Exponenten nehmen andere Werte als im Ising-Modell an:

$$\nu = 0.705(3)$$

$$\gamma = 1.386(4)$$

$$\beta = 0.3645(25)$$

$$\alpha = -0.115(9)$$

In 2 Dimensionen hat das Heisenberg-Modell keinen Phasenübergang. Bei allen endlich Temperaturen verschwindet die Magnetisierung. Das Modell ist asymptotisch frei. Diese Eigenschaft will ich hier nicht diskutieren, sondern nur anmerken, daß auch QCD in 4 Dimensionen diese Eigenschaft hat, und deshalb das 2D

Heisenberg-Modell von Hochenergie-Theoretikern häufig als “Spielmodell” benutzt wird. (z.B. Diplomarbeit H. Molke am Lehrstuhl COM)

## 10.1 Metropolis

Als erstes wollen wir uns überlegen, wie der Metropolis-Algorithmus für dieses Modell realisiert werden kann. In der Tat gibt es viele verschiedene Möglichkeiten. Die vielleicht einfachste Variante:

In einem Elementarschritt des Algorithmus werden lediglich zwei der drei Komponenten des Spin abgeändert. Der Vorschlag wird durch eine Drehung realisiert. Für den Fall, daß wir die 1 und 2 Komponente herausgreifen, ergibt sich

$$\begin{aligned} s'_1 &= \cos(\alpha) s_1 + \sin(\alpha) s_2 \\ s'_2 &= -\sin(\alpha) s_1 + \cos(\alpha) s_2 \end{aligned} \quad (129)$$

Dabei ist  $\alpha$  eine gleichverteilte Zufallszahl aus dem Intervall  $[-\alpha_{max}, \alpha_{max}]$ . Hier könnte man auch andere Verteilungen wählen. Wichtig für detailed balance ist nur, daß  $p(-\alpha) = p(\alpha)$  gilt. Die Akzeptanzwahrscheinlichkeit sieht wieder genauso aus wie im Ising-Modell

$$A(s', s) = \min(1, \exp(-\beta(H(s') - H(s)))) \quad (130)$$

Damit der Algorithmus ergodisch wird, müssen wir alternierend auch Drehungen zwischen der 1 und 3 Komponente zulassen. Am besten ist ein Zyklus wie  $\{1, 2\}$ ,  $\{1, 3\}$ ,  $\{2, 3\}$ .

Im Gegensatz zum Ising-Modell haben wir hier mit der “Schrittweite”  $\alpha_{max}$  einen freien Parameter im Algorithmus. Die  $\tau$  werden von der Wahl von  $\alpha_{max}$  abhängen. Wählt man  $\alpha_{max}$  sehr klein, hat man zwar eine große mittlere Akzeptanzwahrscheinlichkeit, dafür ist aber die Änderung von  $s$  nach  $s'$  klein. Auf der anderen Seite wird die Akzeptanzwahrscheinlichkeit klein, wenn das  $\alpha_{max}$  zu groß wird. Das Optimum liegt dazwischen. Es gilt die Faustregel, daß  $\alpha_{max}$  so eingestellt werden soll, daß die mittlere Akzeptanzwahrscheinlichkeit etwa 50% ist.

## 10.2 Wärmebad

Die Idee des Wärmebad-Algorithmus ist es, den Spin an einem Gitterpunkt, bei festgehaltenen restlichen Spins, gemäß der Boltzmannverteilung neu zu bestimmen. Im Gegensatz zu dem Metropolis-Algorithmus gibt es damit keine statistische Korrelation zu dem alten Wert des Spins auf diesem Gitterpunkt. Die Wahrscheinlichkeit für einen bestimmten Wert des Spins ist durch

$$P(\vec{s}_x) \propto \delta(\vec{s}_x^2 - 1) \exp(\beta \vec{S}_N \vec{s}_x) \quad (131)$$

gegeben, wobei  $S_N = \sum_{y.nn.x} \vec{s}_x$ .

Wie können wir nun  $\vec{s}_x$  mit dieser Wahrscheinlichkeitsverteilung erzeugen?

Wir zerlegen den Spin  $\vec{s}_x$  in eine Komponente  $s_{p,x}$  parallel zu  $\vec{S}_N$  und in die beiden senkrechten Komponenten  $s_{s1,x}$ ,  $s_{s2,x}$ .

Die Verteilung für die  $s_{p,x}$  Komponente ergibt sich durch Integration über die restlichen Freiheitsgrade:

$$\begin{aligned}
\tilde{P}(s_p) &= \int_{-\infty}^{+\infty} ds_{s1} \int_{-\infty}^{+\infty} ds_{s2} P(\vec{s}_x) \\
&\propto \int_{-\infty}^{+\infty} ds_{s1} \int_{-\infty}^{+\infty} ds_{s2} \delta(\vec{s}_x^2 - 1) \exp(\beta \vec{S}_N \vec{s}_x) \\
&= \exp(\beta |S_N| s_p) \int_{-\infty}^{+\infty} ds_{s1} \int_{-\infty}^{+\infty} ds_{s2} \delta(\vec{s}_x^2 - 1) . \quad (132)
\end{aligned}$$

Für  $c > 0$  gilt:

$$\int_{-\infty}^{+\infty} dx \delta(x^2 - c) = 2 \int_0^{+\infty} dy \frac{1}{2\sqrt{y}} \delta(y - c) = \frac{1}{\sqrt{c}} . \quad (133)$$

Dabei haben wir  $y = x^2$  substituiert. Die 2 vor dem Integral kommt von den zwei Wurzeln von  $x^2$ .

Dieses Ergebnis können wir nun verwenden:

$$\begin{aligned}
&\int_{-\infty}^{+\infty} ds_{s1} \int_{-\infty}^{+\infty} ds_{s2} \delta(s_{s1}^2 + s_{s2}^2 - (1 - s_p^2)) \\
&= \int_{-\sqrt{1-s_p^2}}^{+\sqrt{1-s_p^2}} ds_{s1} \frac{1}{\sqrt{1-s_p^2 - s_{s1}^2}} . \quad (134)
\end{aligned}$$

Der Wert dieses Integrals hängt nicht von  $s_p$  ab, wie man mit der Variablensubstitution

$$y = \frac{s_{s1}}{\sqrt{1-s_p^2}} \quad (135)$$

sieht.

Allgemein ergibt sich für  $N > 1$ :

$$\begin{aligned}
&\int dx_1 \dots \int dx_{N-1} \delta(x_1^2 + \dots + x_{N-1}^2 + x_N^2 - 1) \\
&\propto \int \text{Winkel} \int dr r^{-N+2} \delta(r^2 + x_N^2 - 1) \\
&\propto \int dr r^{N-2} \delta(r^2 + x_N^2 - 1) \\
&\propto \int dy y^{(N-2)/2-1/2} \delta(y + x_N^2 - 1) \\
&= (1 - x_N^2)^{(N-3)/2} . \quad (136)
\end{aligned}$$

$N = 3$  ist also der einfachste Fall. Wir können nun  $s_p$  mit

$$s_p = \ln(z + (z^{-1} - z) r) / \tilde{\beta} . \quad (137)$$

erzeugen. Dabei ist  $z = \exp(-\tilde{\beta})$  und  $\tilde{\beta} = \beta|\vec{S}_N|$ .  $r$  ist eine Zufallszahl, die mit gleichförmiger Verteilung im Intervall  $(0, 1]$  erzeugt wird.

Überzeugen wir uns, daß dadurch die richtige Verteilung generiert wird:

$$\begin{aligned} \frac{ds_p}{dr} &= \frac{1}{\tilde{\beta}} \frac{z^{-1} - z}{z + (z^{-1} - z)r} \\ &= \frac{z^{-1} - z}{\tilde{\beta}} \frac{1}{z + (z^{-1} - z)r} \\ &= \frac{z^{-1} - z}{\tilde{\beta}} \exp(-\tilde{\beta}s_p) \end{aligned} \quad (138)$$

Der letzte Schritt folgt aus Gleichung (137), da

$$\exp(\tilde{\beta}s_p) = z + (z^{-1} - z)r \quad (139)$$

Wir erinnern uns:

$$\tilde{P}(s_p) = \left| \frac{dr}{ds_p} \right| P(x) \quad (140)$$

Also ist

$$\tilde{P}(s_p) = \frac{\tilde{\beta}}{z^{-1} - z} \exp(\tilde{\beta}s_p) \quad (141)$$

wie gewünscht.

Die Konstruktion der verbleibenden beiden Komponenten ist recht einfach. Es wird gleichverteilt ein Winkel  $\alpha$  aus  $(0, 2\pi]$  ausgewählt. Dann sind die senkrechten Komponenten durch

$$\begin{aligned} s_{s1} &= \sqrt{1 - s_p^2} \cos(\alpha) \\ s_{s2} &= \sqrt{1 - s_p^2} \sin(\alpha) \end{aligned} \quad (142)$$

gegeben.

Zur Illustration der entsprechende Teil aus einem Fortran77 Programm:

```
xx=neigh(1)**2+neigh(2)**2+neigh(3)**2
  xx=beta*sqrt(xx)
yy=1D0/xx

u(1)=beta*neigh(1)*yy
u(2)=beta*neigh(2)*yy
u(3)=beta*neigh(3)*yy

z=exp(-xx)
t=log(z+(1D0/z-z)*xjames())*yy
```

```

if(abs(u(1)).gt.0.25D0) then
  a=sqrt((1D0-t**2)/(1D0-u(3)**2))
  v1(1)= u(2)*a
  v1(2)=-u(1)*a
  v1(3)=0D0
else
  a=sqrt((1D0-t**2)/(1D0-u(1)**2))
  v1(1)=0D0
  v1(2)= u(3)*a
  v1(3)=-u(2)*a
endif

v2(1)=u(2)*v1(3)-u(3)*v1(2)
v2(2)=u(3)*v1(1)-u(1)*v1(3)
v2(3)=u(1)*v1(2)-u(2)*v1(1)

ph=2D0*3.1415926535897932D0*xjames()
  co=cos(ph)
  si=sin(ph)

  sp(1,i1,i2,i3)=t*u(1)+co*v1(1)+si*v2(1)
  sp(2,i1,i2,i3)=t*u(2)+co*v1(2)+si*v2(2)
  sp(3,i1,i2,i3)=t*u(3)+co*v1(3)+si*v2(3)

```

### 10.3 Überrelaxation

Die Idee der Überrelaxation stammt aus der iterativen Lösung linearer Gleichungssysteme mit dünn besetzten Matrizen. Solche Gleichungssysteme kommen zustande, wenn z.B. partielle Differentialgleichungen diskretisiert werden. Traditionelle Verfahren zur Lösung solcher Probleme sind das Jacobi- und das Gauss-Seidel-Verfahren. (siehe Lehrbücher der Numerischen Mathematik, oder auch Press et al).

Eine Verbesserung dieser Methoden ist die Überrelaxation. Die Idee dabei ist, über den Gauss-Seidel-Schritt hinweg zu schießen.

Die einfachste Realisierung für  $O(N)$ -invariante nichtlineare  $\sigma$ -Modelle ist es, den neuen Spin so zu wählen, daß zum einen der Wert der Hamiltonfunktion exakt gleich bleibt, und zum anderen die Entfernung von dem alten Spin am größten ist. Dies wird erreicht, indem man den Spin an der Summe seiner Nachbarn spiegelt.

$$\vec{s}_x' = 2 \frac{(\vec{S}_N \vec{s}_x)}{S_N^2} \vec{S}_N - \vec{s}_x \quad (143)$$

Um zu sehen, daß dieser Algorithmus **detailed balance** erfüllt, ist es am ein-

fachsten, ihn als Spezialfall des Metropolis aufzufassen. Offensichtlich ist die Wahrscheinlichkeit, von  $s_x$  nach  $s'_x$  zu kommen, die gleiche wie von  $s'_x$  nach  $s_x$ . Da sich dabei  $H$  nicht ändert, ist die Akzeptanzwahrscheinlichkeit gleich 1.

Da  $H$  durch diese updates nicht verändert werden kann, kann der Algorithmus für sich genommen nicht ergodisch sein. Deshalb müssen sweeps mit dem Überrelaxations-Algorithmus mit Metropolis- oder Wärmebad-sweeps ergänzt werden.

Ein vollständiger update-Zyklus hat die Form  
 1 Wärmebad-sweep gefolgt von  $M$  Überrelaxations-sweeps  
 Dabei ist nun  $M$  ein Parameter, der so eingestellt werden soll, daß  $\tau \times$  (CPU-Zeit pro Zyklus) minimal wird.

Analytische Rechnungen für das Gaussche Modell und numerische Experimente haben gezeigt, daß

$$M_{opt} \propto \xi \quad (144)$$

für  $L \gg \xi$  bzw.

$$M_{opt} \propto L \quad (145)$$

für  $T \approx T_c$ .

Bei dieser optimalen Einstellung des Parameters kann  $z \approx 1$  erreicht werden.

## 10.4 Dämonen

Die Verwendung von "demons" in Monte-Carlo-Simulationen wurde von M.Creutz vorgeschlagen.

M. Creutz, Phys. Rev. Lett. 50 (1983) 1411.

Der Name läßt Geheimnisvolles vermuten. Es handelt sich aber um einen recht einfachen Trick. Man fügt dem System zusätzliche Freiheitsgrade hinzu, die jedoch nicht an die physikalischen Freiheitsgrade koppeln. In der Hamiltonfunktion sieht das so aus:

$$H_{comb}(s, d) = H_{spin}(s) + H_{demon}(d) \quad . \quad (146)$$

Dabei wählt man  $H_{demon}(d)$  im einfachsten Fall als

$$H_{demon}(d) = \sum_{i=1}^{N_d} d_i \quad . \quad (147)$$

Dabei ist  $d_i$  eine positive reelle Zahl. Die Zahl der Dämonen ist dabei zunächst beliebig. Zunächst zeigen wir, daß man für Observablen, die nur von den Spins abhängen mit  $H_{comb}(s, d)$  die gleichen Erwartungswerte bekommt, wie mit  $H_{spin}(s)$ .

$$\frac{\int Ds \int Dd \exp(-\beta H_{spin}(s) - \beta H_{demon}(d)) A(s)}{Z_{comb}} = \frac{\int Ds \exp(-\beta H_{spin}(s)) A(s)}{Z_{spin}} \frac{\int Dd \exp(-\beta H_{demon}(d))}{Z_{demon}} =$$

$$\frac{\int Ds \exp(-\beta H_{spin}(s)) A(s) Z_{demon}}{Z_{spin} Z_{demon}} = \langle A \rangle . \quad (148)$$

Die Dämonen ermöglichen nun neue Möglichkeiten des Updates. Nehmen wir an, wir hätten genau einen Dämonen. Ein elementarer Metropolissschritt mit Verwendung des Dämons sieht nun so aus: Wie bisher bestimmen wir einen Vorschlag  $s'_x$  für einen neuen Wert des Spins am Gitterpunkt  $x$ . Wir berechnen die dazugehörige Änderung  $\Delta H_{spin} = H'_{spin} - H_{spin}$  der Hamiltonfunktion des Spin-Modells. Beim "normalen" Metropolis berechnen wir daraus die Akzeptanzwahrscheinlichkeit. Beim Verfahren mit dem Dämon prüfen wir, ob

$$d' = d - \Delta H_{spin} \geq 0 . \quad (149)$$

Falls diese Bedingung erfüllt ist, so wird  $s_x$  durch  $s'_x$  und  $d$  durch  $d'$  ersetzt. Im anderen Fall bleiben die alten Werte.

Verwendet man ausschließlich solche Updates, bleibt der Wert von  $H_{comb}$  konstant. Man simuliert das mikrokanonische Ensemble des zusammengesetzten Systems. Im Limes  $N_d \rightarrow \infty$  erhält man aber für das Spinsystem wieder ein kanonisches Ensemble.

Eine einfache Möglichkeit, das kanonische Ensemble für das zusammengesetzte System zu bekommen, ist es mit Updates zu alternieren, die den Wert der Hamiltonfunktion ändern können. Eine Möglichkeit ist ein Wärmebad für die Dämonen:

$$d' = -\ln(\text{drand48}())/\beta \quad (150)$$

Dämonen lassen sich mit vielen Algorithmen kombinieren. So hat M.Creutz auch eine "dämonische" Variante des Cluster-Algorithmus gefunden: MICROCANONICAL CLUSTER MONTE CARLO, Phys.Rev.Lett.69:1002-1005,1992, hep-lat-9204005

## 11 Phasenübergänge erster Ordnung

Viele Phasenübergänge in der Natur sind erster Ordnung. Erste Ordnung heißt, daß die ersten Ableitungen der freien Energiedichte unstetig sind (im thermodynamischen Limes, im endlichen System bleibt natürlich alles stetig). Bei einem temperaturgetriebenen Phasenübergang erster Ordnung macht die Energiedichte einen Sprung beim Phasenübergang. Die Differenz ist die latente Wärme.

Phasenübergänge erster Ordnung treten in vielen Bereichen auf. Uns allen ist das Schmelzen von Eis zu Wasser und das Verdampfen von Wasser bekannt. Phasenübergänge erster Ordnung spielen aber zum Beispiel auch eine Rolle in der Kosmologie.



Als einfache Modelle mit einem Phasenübergang erster Ordnung möchte ich die  $q$ -Zustands-Potts-Modelle in 3 Dimensionen diskutieren. Die Hamiltonfunktion ist

$$H = -J \sum_{\langle xy \rangle} \delta_{\sigma_x, \sigma_y} \quad , \quad (151)$$

wobei  $\sigma \in \{1, 2, \dots, q\}$ . Für  $q = 2$  erhält man das Ising-Modell. In drei Dimensionen haben die Modelle für  $q \geq 3$  einen Phasenübergang erster Ordnung. In zwei Dimensionen für  $q \geq 5$ .

Die letzte große Monte-Carlo-Studie des 3-Zustands-Potts-Modells ist

Three-Dimensional 3-State Potts Model Revisited With New Techniques  
 Wolfhard Janke (JGU Mainz), Ramon Villanova (UPF Barcelona)

hep-lat/9612008

Nucl.Phys. B489 (1997) 679-696

Hier findet sich für das 3-Zustands-Potts-Modell  $\beta_c = 0.550565(10)$

Interessante Größen, die einen Phasenübergang erster Ordnung charakterisieren sind die latente Wärme

$$\Delta E = \lim_{T \searrow T_c} E - \lim_{T \nearrow T_c} E \quad , \quad (152)$$

wobei  $E$  die Energiedichte im thermodynamischen Limes ist, die Korrelationslängen  $\xi$  in den beiden Phasen und die Oberflächenspannung zwischen den Phasen. Die Oberflächenspannung eines dreidimensionalen Systems ist mikroskopisch durch

$$\sigma := \lim_{D \rightarrow \infty} \lim_{L \rightarrow \infty} \frac{1}{L^2} \left( F_{ug} - \frac{1}{2} F_{uu} - \frac{1}{2} F_{gg} \right) \quad (153)$$

definiert. Dabei betrachten wir ein System der Größe  $D \times L^2$ . In  $x_2$  und  $x_3$  Richtung haben wir periodische Randbedingungen, in  $x_1$  Richtung sind feste Ränder vorgegeben. Diese Ränder sind entweder ungeordnet ( $u$ ) oder geordnet ( $g$ ). Hat man auf der einen Seite einen geordneten Rand und auf der anderen einen ungeordneten, so wird eine Phasengrenzfläche in das System gezwungen. Die Temperatur ist  $T_c$ .  $F$  bezeichnet die freie Energie.

Ein "schwacher" Phasenübergang erster Ordnung zeichnet sich durch eine kleine latente Wärme, eine kleine Oberflächenspannung und große Korrelationslängen aus. Verschwinden latente Wärme und Oberflächenspannung, und divergiert die Korrelationslänge, dann haben wir einen Phasenübergang zweiter Ordnung.

Auf einem endlichen Gitter können wir natürlich keine Unstetigkeit der Energiedichte erhalten. Wir müssen uns überlegen, wie wir am besten die interessanten Größen aus Simulationen endlicher Systeme bestimmen können.

Um einen ersten Eindruck zu gewinnen, was in der Simulation geschieht, ist in Abbildung 10 die Monte-Carlo-Zeitentwicklung der Energie eines  $16^3$  Systems für das 3-Zustands-Potts-Modell bei  $\beta_c$  gegeben. Simuliert wurde mit Hilfe des Metropolis-Algorithmus.

Wir sehen erste Anzeichen für Metastabilitäten. Das System verharrt für große Zeiten in einer Phase, bis es in eine andere wechselt.

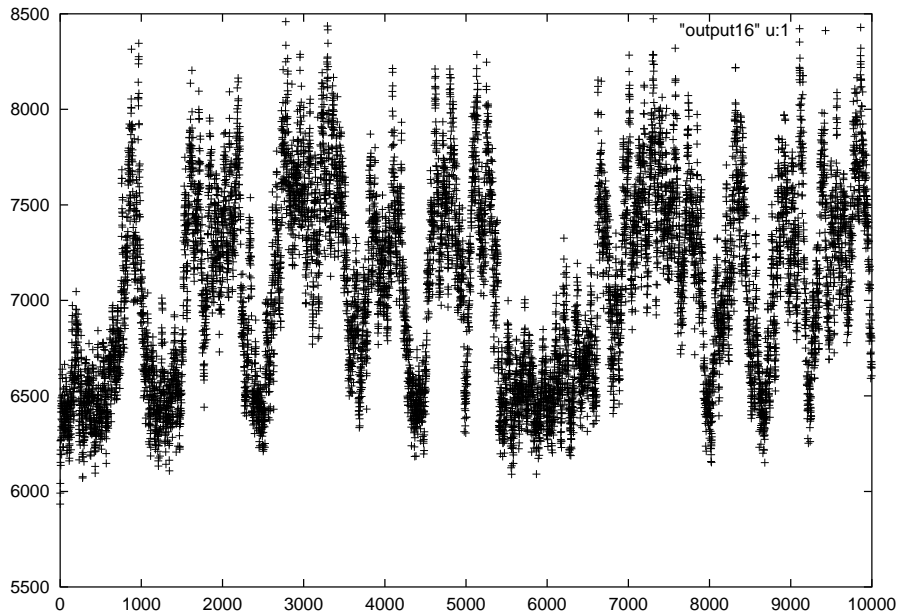


Abbildung 10:

Die Separation der Phasen läßt sich noch besser an einem Histogramm der Energiedichte erkennen. In Abbildung 11 ist das Histogramm der Energiedichte für  $8^3$ ,  $12^3$  und  $16^3$  bei  $\beta_c$  gegeben.

Man erkennt, wie sich mit zunehmender Systemgröße eine “double-peak”- (Doppelspitzen)-Struktur herausbildet. Dies ist ein markantes Zeichen für einen Phasenübergang erster Ordnung. Wichtig dabei ist, daß die Ausprägung dieser Struktur mit zunehmender Systemgröße stärker wird.

Phänomenologisch läßt sich dies so erklären, daß die peaks zu Konfigurationen gehören, bei denen eine Phase das ganze Gitter erfaßt. Dazwischen liegen Konfigurationen, bei denen ein Teil des Gitters von einer Phase und ein anderer Teil von einer anderen Phase erfaßt wird. Bei einem kubischen Gitter mit periodischen Randbedingungen sind dabei Konfigurationen von der freien Energie bevorzugt, bei denen die Phasen durch zwei  $L^2$  große Trennflächen voneinander getrennt sind. In Abbildung 15 ist diese Situation skizziert.

Mit diesem Bild können wir verstehen, in welcher Weise sich die Doppelspitzenstruktur mit wachsendem  $L$  ausprägt: Das Verhältnis des Talniveaus zu den Spitzen verhält sich wie  $\exp(-2L^2\sigma)$ .

Zu diesem Verhalten gibt es Korrekturen. Diese möchte ich hier nicht diskutieren, sondern auf die Literatur verweisen, z.B. der Band J.Mod.Phys.C 3 (1992).

## 3D 3-state Potts model

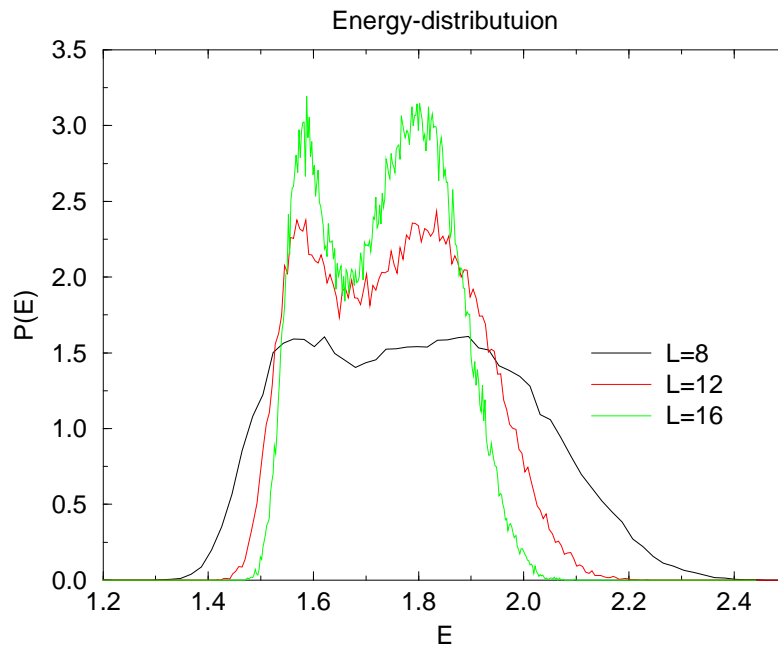


Abbildung 11:

### 11.1 Zur Bestimmung der Phasenübergangstemperatur:

Die Phasenübergangstemperatur zeichnet sich dadurch aus, daß die freien Energiedichten der auftretenden Phasen gleich sind.

Bei endlichen Gittern heißt das, daß man nach dem  $\beta_c$  sucht, bei dem alle Phasen mit dem gleichen Gewicht auftreten. Dabei muß man bei den Potts-Modellen beachten, daß es  $q$  verschiedene geordnete Phasen gibt.

Nun muß man natürlich noch ein Kriterium angeben, um zu entscheiden, zu welcher Phase eine gegebene Konfiguration gehört. Ein einfaches Kriterium ist die Energie einer Konfiguration. Ist  $E < E_0$ , dann ist man in einer der geordneten Phasen und für  $E > E_0$  in der ungeordneten Phase. Nun bleibt noch  $E_0$  festzulegen.

Ein Vorschrift dazu ist, daß man das  $E$  zwischen den peaks nimmt, bei dem  $P(E)$  am kleinsten ist.

```
program betac
  integer ene(200000)
  open(10,file="output")
  read(10,*) L0,L1,L2,betas
  write(*,*) "geben sie die zahl der Messungen"
  read*, imax
  write(*,*) "geben sie die Zahl der thermalisierungsschritte"
```

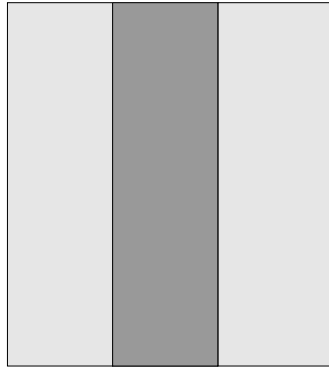


Abbildung 12:  
Schematische Skizze von Konfigurationen, in denen zwei Phasen koexistieren.

```

read*, idis
imax=imax-idis
do i=1, idis
  read(10,*) ii
enddo
do i=1, imax
  read(10,*) ene(i)
enddo
write(*,*) "geben sie die trennenergie"
read*, itrenn

```

C

```

bestimme betac

```

```

db1=0.
gew1=gewicht(ene, imax, db1, itrenn)
db2=0.001
gew2=gewicht(ene, imax, db2, itrenn)

do i=1, 1000
  dsave=db2
  db2=db1+(db2-db1)*(4.-gew1)/(gew2-gew1)
  gew1=gew2
  gew2=gewicht(ene, imax, db2, itrenn)
  db1=dsave
  write(*,*) db2, gew2
  if(abs(db1-db2).lt.0.0000001) goto 134
enddo
continue

```

134

```

write(*,*) "betac= ", betas+db2

```

```

end

```

```

real function gewicht(ene, imax, db, itrenn)
integer ene(200000)
  g1=0.
  g2=0.
  do i=1, imax
    if(ene(i).gt.itrenn) then
      g2=g2+exp(db*ene(i))
    else
      g1=g1+exp(db*ene(i))
    endif
  enddo

```

```

        enddo
        gewicht=g2/g1
    end

```

Für das 4-Zustands-Potts-Modell finde ich damit  $\beta_c \approx 0.6285$ .

## 11.2 Modifikation des Gewichts, mit dem die Konfigurationen erzeugt werden.

In der letzten Vorlesung haben wir gelernt, daß man mit Hilfe des relativen Gewichts der Phasen die Phasenübergangstemperatur sehr gut bestimmen kann. Die statistischen Fehler des relativen Gewichts sind umso kleiner, je häufiger man von der einen in die andere Phase bei der Simulation wechselt.

Wir haben auch gelernt, daß die Wahrscheinlichkeit für Konfigurationen, bei denen zwei Phasen koexistieren, mit einem Faktor  $\exp(-2L^2\sigma)$  unterdrückt ist. Dadurch wird der Wechsel der Phasen in der Simulation mit wachsendem  $L$  immer seltener.

Abhilfe: B. Berg and T. Neuhaus

Multicanonical Ensemble: A New Approach to Simulate First-order Phase Transitions,

hep-lat/9202004, Phys. Rev. Lett. 68 (1992) 9.

Anstatt Konfigurationen mit dem Boltzmannfaktor als Gewicht zu erzeugen, wählt man ein modifiziertes Gewicht:

$$P(X) \propto \exp(-\beta H(X)) \times P_M(H(X)) \quad . \quad (154)$$

Dabei wird  $P_M(H(X))$  gerade so gewählt, daß das Tal zwischen den beiden Spitzen in der Energieverteilung aufgefüllt wird.  $P_M(H(X))$  muß in der Regel numerisch bestimmt werden.

Der Metropolis-Algorithmus muß für das modifizierte Gewicht angepaßt werden. Dabei sieht der Vorschlag aus wie bisher. Geändert wird die Akzeptanzwahrscheinlichkeit:

$$A(X', X) = \min \left( 1, \exp(-\beta(H(X') - H(X))) \frac{P_M(H(X'))}{P_M(H(X))} \right) \quad . \quad (155)$$

Entsprechendes Programmteil aus einem C-Programm für den Metropolis-Algorithmus:

```

/* decide whether to accept the proposal */
enerp=ener+dif;
if((p[dif+6]*pm(enerp)/pm(ener))>drand48())
{
    ener=enerp;
    spin[i0][i1][i2]=pro;
}

```

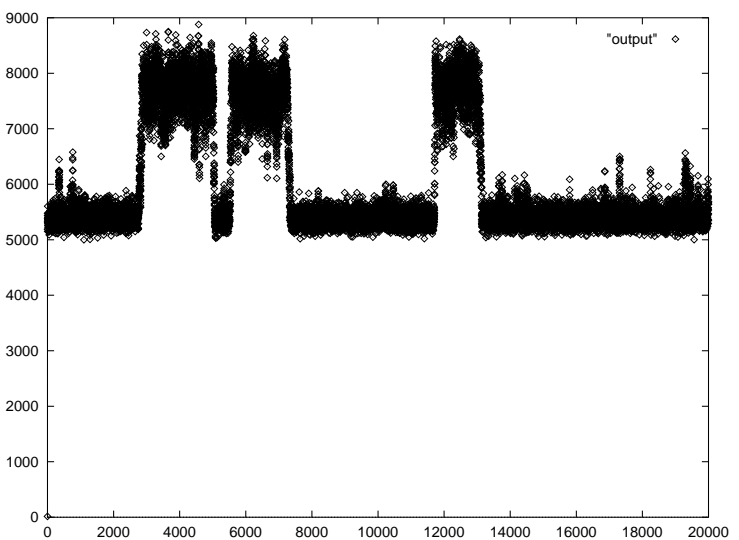


Abbildung 13: Z  
 eitentwicklung der Energie. Metropolis-Algorithmus,  $16^3$  Gitter 4-  
 Zustands-Potts-Modell bei  $\beta = 0.6380$ .

Abbildung 14:

Abbildung 15:

Um  $\text{pm}(\text{ener})$  zu bestimmen muß eine entsprechende function bereitgestellt werden.

```
double pm(int ene)
{
double xx;
xx=1.;
if((ene>5600)&&(ene<=5900))
{
xx=1.+19.*(ene-5600)/300.;
}
if((ene>5900)&&(ene<=7000))
{
xx=20.;
}
if((ene>7000)&&(ene<=7300))
{
xx=1.+19.*(7300-ene)/300.;
}
return xx;
}
```

Um nun Observabeln auszuwerten, müssen wir wieder zu dem Boltzmannfaktor zurück kommen. Dies sieht ähnlich aus wie bei dem Umwichten in  $\beta$ . Werden die Konfigurationen mit  $\exp(-\beta H(X)) \times P_M(H(X))$  als relativem Gewicht erzeugt, so ergibt sich der Erwartungswert als

$$\langle A \rangle \approx \frac{\sum_i P_M(H_i)^{-1} A_i}{\sum_i P_M(H_i)^{-1}} . \quad (156)$$

A. Billoire, T. Neuhaus and B. Berg:  
A DETERMINATION OF INTERFACE FREE ENERGIES,  
hep-lat/9307017, Nucl. Phys. B413 (1994) 795

Verwandte Methoden:

Simulated Tempering: A New Monte Carlo Scheme,  
by Enzo Marinari and Giorgio Parisi,  
hep-lat/9205018, Europhys. Lett. 19 (1992) 451-458



Hier wird die inverse Temperatur als Variable behandelt. Neben den “normalen” update-Schritten gibt es solche, die das  $\beta$  verändern.

Parallel Tempering:

Optimized Monte Carlo Methods

Enzo Marinari

cond-mat/9612010, Lectures given at the 1996 Budapest Summer School on Monte Carlo Methods.

### 11.3 Mikrokanonisches Ensemble

Eine andere Herangehensweise ist die mikrokanonische Simulation mit einem Dämon. Man simuliert

$$H_{comb} = H_{potts} + H_{demon} \quad (157)$$

mit

$$H_{demon} = d \quad , \quad (158)$$

wobei  $d$  aus  $N_0$  den natürlichen Zahlen einschliesslich der Null ist. Im kanonischen Ensemble lässt sich der Erwartungswert des Dämons leicht ausrechnen:

$$\langle d \rangle_{kanonisch}(\beta) = \frac{\sum_{d=0}^{\infty} \exp(-\beta d) d}{\sum_{d=0}^{\infty} \exp(-\beta d)} \quad . \quad (159)$$

Je grösser das Gitter ist, desto besser spielt das Spin-System die Rolle eines Wärmebads für den Dämon. Damit ist die Verteilung immer besser durch die Boltzmannverteilung des kanonischen Ensembles beschrieben. Wir können Gleichung (159) benutzen, um den Dämon als Thermometer zu eichen.

So können wir die Temperatur als Funktion der Energie bestimmen:

a) Starte das kombinierte System aus Spin-Modell und Dämon mit einer festgelegten Energie. Dies kann man z.B. einfach realisieren, indem man mit allen Spins gleich startet, und den Rest der Energie in den Dämon steckt.

b) Simuliere das System mit dem modifizierten Metropolis-Algorithmus aus dem Kapitel 10. Nach der Thermalisierung messe den Dämon. Aus dem so bestimmten Erwartungswert des Dämons bestimmt man durch Invertierung von Gl. (159) die Temperatur.

Der Phasenübergang erster Ordnung zeichnet sich nun dadurch aus, daß  $\beta(E)$  zwischen  $E_+$  und  $E_-$  etwa waagrecht verläuft.

## 12 Renormierungsgruppe

Die Renormierungsgruppe (RG) ist die moderne Grundlage unseres Verständnisses kontinuierlicher Phasenübergänge. Mit Hilfe der Renormierungsgruppe lässt sich verstehen, wie es zu den Potenzgesetzen bei Phasenübergängen zweiter Ordnung

kommt. Man kann Relationen zwischen kritischen Exponenten herleiten. Die Renormierungsgruppe ist auch die theoretische Grundlage für das finite size scaling. Einen neueren Übersichtsartikel findet man in:

M.E. Fisher, Renormalization group: Its basis and formulation in statistical physics, *Reviews of Modern Physics* 70 (1998) 653.

Pionierarbeit in diesem Feld wurde von K.G. Wilson geleistet. (Siehe z.B. K.G. Wilson and J. Kogut *Phys. Rep.* 12 (1974) 75).

Bücher zum Thema:

Kerson Huang, *Statistical Mechanics*, second edition

Giorgio Parisi, *Statistical Field Theory*, Addison Wesley.

Claude Itzykson and Jean-Michel Drouffe, *Statistical Field Theory*, Volume I, Cambridge University Press.

John Cardy, *Scaling and Renormalization in Statistical Physics*

C. Domb: *The Critical Point*, Taylor&Francis

R.J. Creswick, H.A. Farach, C.P. Poole: *Introduction to Renormalization Group Methods in Physics*, John Wiley and sons

J.J. Binney, N.J. Dowrick, A.J. Fisher, and M.E.J. Newman, *The theory of critical phenomena, an introduction to the Renormalization group*, Oxford Science publishing.

Nigel Goldenfeld, *Lectures on Phase Transitions and the Renormalization Group*, Addison-Wesley.

In den drei letzten Vorlesungen möchte ich in die Theorie der Renormierungsgruppe im "Ortsraum" einführen und zeigen, wie sich RG-Transformationen mit Hilfe von Monte-Carlo-Simulationen ausführen lassen.

Als Beispiel greifen wir wieder auf das Ising-Modell zurück. Eine konkrete Realisierung der RG-Transformationen im Ortsraum ist die "Blockspin-Transformation". Die Idee dazu stammt von L.P. Kadanoff, *Physics* 2 (1966) 263.

Die Idee dabei ist es, die Zahl der Freiheitsgrade zu reduzieren, dabei aber die Physik auf großen Skalen zu erhalten. Bei der Blocktransformation ordnet man  $b^d$  großen Blöcken einen Punkt im geblockten Gitter zu. Auf diesem Punkt lebt ein "Blockspin", der aus den einzelnen Spins des dazugehörigen Blocks berechnet wird.

Beim Ising-Modell ist eine häufig diskutierte Blockungsregel die sogenannte Majoritätsregel. Wenn die Mehrheit der Spins im Block gleich 1 ist, wird der Blockspin gleich 1 gesetzt. Ist die Mehrheit  $-1$ , so wird der Blockspin gleich  $-1$  gesetzt. Ist die Zahl der Punkte im Block gerade, so kann es auch dazu kommen, daß die Summe der Spins im Block gleich 0 ist. Dann wird mit Wahrscheinlichkeit  $1/2$  für Blockspin 1 oder  $-1$  entschieden.

Eine andere Möglichkeit besteht darin, daß der Blockspin nicht mehr aus  $\{-1, 1\}$  ist, sondern reell wird, z.B.

$$S_X = \frac{1}{b^y} \sum_{x \in X} s_x . \quad (160)$$

Dabei bezeichnet  $X$  einen Punkt auf dem Blockgitter. Den Exponenten  $y$  werden

wir später erklären. Diese Art der Transformation heißt lineare Blockung, da  $S_X$  linear von den  $s_x$  abhängt.

Die Blockspintransformation induziert ein Boltzmanngewicht für die Blockspin-konfigurationen:

$$B(S) := \sum_s T(S, s) \exp(-H(s)) . \quad (161)$$

Dabei beschreibt  $T(S, s)$  die Blockspintransformation.  $S$  bezeichnet hier die Gesamtheit aller Blockspins und  $s$  alle Spins auf dem ursprünglichen Gitter. Hierbei haben wir die Konvention bei der Hamiltonfunktion gewechselt:  $\beta$  ist in  $H$  absorbiert worden.

$T(S, s)$  soll dabei folgenden Bedingungen genügen:

$$\sum_s T(S, s) = 1 \quad (162)$$

und

$$T(S, s) \geq 0 . \quad (163)$$

Dies ist notwendig, damit wir  $T$  als eine Wahrscheinlichkeit interpretieren können.

Eine wichtige Eigenschaft der Transformation ist, daß die Zustandssumme erhalten bleibt:

$$\begin{aligned} Z' &= \sum_S B(S) \\ &= \sum_S \sum_s T(S, s) \exp(-H(s)) \\ &= \sum_s \left( \sum_S T(S, s) \right) \exp(-H(s)) \\ &= \sum_s \exp(-H(s)) = Z . \end{aligned} \quad (164)$$

Die Korrelationslänge transformiert sich wie

$$\xi' = \xi/b . \quad (165)$$

Wir erinnern uns, daß die Korrelationslänge durch das Abfallverhalten der Korrelationsfunktion bestimmt ist. Für die Korrelationsfunktion gilt

$$G_{block}(b x) \propto G(x) . \quad (166)$$

An dieser Stelle bemerken wir auch, daß die Binderkumulante (zumindest unter der "linearen" Blockung) konstant bleibt.

Schreiben wir den Boltzmannfaktor des geblockten Systems in der Form

$$B(S) = \exp(-H'(S)) , \quad (167)$$

so sehen wir, daß die Blockspintransformation eine Abbildung

$$H \rightarrow H' \quad (168)$$

bewirkt.

Einfaches Beispiel: 1-dimensionales Ising-Modell:

Eine einfache RG-Transformation ist die sogenannte Dezimation. Hierbei wird über jeden zweiten Spin auf der Kette summiert. Betrachten wir  $h = 0$ :

$$H = -K \sum_{x=1}^L s_x s_{x+1} . \quad (169)$$

Wir verwenden periodische Randbedingungen, d.h. wir identifizieren  $L + 1$  mit 1. Bei der Dezimation reduziert sich die Zahl der Gitterpunkte um die Hälfte:  $L' = L/2$ . Die Zuordnung von Punkten  $x$  des ursprünglichen Gitters und Punkten  $X$  des "geblockten" Gitters ist durch  $X = x/2$  gegeben. Der Transformationskern  $T$  ist durch

$$T(S, s) = \prod_X \delta_{S_X, s_{2X}} \quad (170)$$

gegeben.

Setzt man dies in Gleichung (161) ein, so erhält man

$$\begin{aligned} B(S) &= \sum_s T(S, s) \exp(-H(s)) \\ &= \sum_s \prod_X \delta_{S_X, s_{2X}} \exp\left(K \sum_{x=1}^L s_x s_{x+1}\right) \\ &= \sum_{s, X \text{ ungerade}} \prod_X \exp\left(K s_{2X+1} (S_X + S_{X+1})\right) \\ &= \prod_X \sum_{s_{2X+1}=\pm 1} \exp\left(K s_{2X+1} (S_X + S_{X+1})\right) \\ &= \prod_X [(\exp(K (S_X + S_{X+1})) + \exp(-K (S_X + S_{X+1})))] \\ &= C^{L'} \exp\left(K' \sum_X S_X S_{X+1}\right) . \end{aligned} \quad (171)$$

Mit  $C = 2 \exp(1/2) \cosh(2K)$  und  $K' = \frac{1}{2} \ln(\cosh(2K))$   $C$  und  $K'$  erhält man durch Lösen von

$$\begin{aligned} C \exp(+K') &= \exp(2K) + \exp(-2K) \\ C \exp(-K') &= 2 . \end{aligned} \quad (172)$$

Für grosse  $K$  erhält man  $K' \approx K - \frac{1}{2} \ln(2)$ .  $K$  wird also unter der Transformation kleiner.

Eine Besonderheit des eindimensionalen Modells ist, daß die Hamiltonfunktion unter der RG-Transformation ihre einfache Form behält. Es kommt lediglich die Konstante  $C$  hinzu, die aber für die Wahrscheinlichkeitsverteilung der Spins keine Rolle spielt.

Schon in zwei Dimensionen ist die Sache nicht mehr so einfach. Man kann ausgehend von dem Hamiltonian mit nächster Nachbarwechselwirkung die Dezimationstransformation ausführen. Hierbei zerlegt man das Gitter im Schachbrettmuster. Aufsummiert wird über die Spins auf den schwarzen Feldern.

Die Hamiltonfunktion des dezimierten Systems enthält aber nicht nur den Term

$$\sum_{\langle XY \rangle} S_X S_Y \quad , \quad (173)$$

sondern auch

$$\sum_{[XY]} S_X S_Y \quad , \quad (174)$$

wobei  $[XY]$  Paare von übernächsten Nachbarn auf dem Gitter bezeichnet, und

$$\sum_{\langle XYUV \rangle} S_X S_Y S_U S_V \quad , \quad (175)$$

wobei  $\langle XYUV \rangle$  vier Punkte auf einem Einheitsquadrat sind.

Für diese kompliziertere Hamiltonfunktion kann man einen weiteren Dezimationsschritt nicht mehr explizit ausrechnen.

Im allgemeinen (auch für andere Transformationsregeln) muß man feststellen, daß  $H'$  nicht mehr die einfache Form der ursprünglichen Ising-Hamiltonfunktion haben wird. Im Fall eines Blockspins aus  $\{-1, 1\}$  werden weitere Terme der Form

$$\sum_{[x,y]} s_x s_y \quad (176)$$

auftreten, wobei jetzt die Punkte  $x$  und  $y$  größeren Abstand als 1 haben können. Es werden aber auch Terme mit vier Spins

$$\sum_{[x,y,u,v]} s_x s_y s_u s_v \quad (177)$$

6, 8 usw. auftreten. Betrachten wir allgemein den Fall eines nicht verschwindenden äußeren Feldes, so werden auch Terme mit einem, drei, fünf usw. Spins auftreten.

Zu jedem dieser Terme gehört eine Kopplungskonstante  $K_\alpha$ :

$$H = - \sum_{\alpha} K_{\alpha} S_{\alpha} \quad . \quad (178)$$

Hier bezeichnet  $S_{\alpha}$  einen Term wie (176) oder (177) in der Hamiltonfunktion. Damit können wir eine Hamiltonfunktion durch einen Vektor im Kopplungsraum charakterisieren.

Die RG-Transformation wird so zu einer Abbildung Raum der Kopplungen  $K_{\alpha}$ :

$$\vec{K} \rightarrow \vec{K}' \quad . \quad (179)$$

Ein Fluß, der von einem kritischen Punkt aus startet, läuft in einen Fixpunkt  $\vec{K}^*$ . D.h., das System ist skaleninvariant.

Um die Skalengesetze zu verstehen reicht es aus, die erste Ordnung der Taylorreihe der RG-Transformation bei  $\vec{K}^*$  zu betrachten:

$$\vec{K}' - \vec{K}^* = T (\vec{K} - \vec{K}^*) + \dots \quad (180)$$

Dies erfordert, daß die RG-Transformation auch bei  $\vec{K}^*$  eine analytische Funktion ist. Dies kann man zwar in der Regel nicht beweisen, aber es gibt sehr gute Argumente dafür.

,Dabei ist es nützlich die Eigenwerte und Eigenvektoren von  $T$  zu betrachten:

$$\lambda_i \Phi_i = \Phi_i T \quad . \quad (181)$$

Man definiert "Skalenfelder" (scaling fields)

$$u_i = \Phi_i (\vec{K} - \vec{K}^*) \quad (182)$$

und RG-Exponenten:

$$b^{y_i} = \lambda_i \quad . \quad (183)$$

Die RG-Exponenten werden als relevant, marginal oder irrelevant klassifiziert:

Relevant:

$$y_i > 0 \quad . \quad (184)$$

Das heißt, daß das dazugehörige Skalenfeld unter RG-Transformationen größer wird.

Marginal:

$$y_i = 0 \quad . \quad (185)$$

Irrelevant:

$$y_i < 0 \quad . \quad (186)$$

Das heißt, daß das dazugehörige Skalenfeld unter RG-Transformationen kleiner wird.

Im Ising-Modell in 2 und 3 Dimensionen oder den  $O(N)$ -invarianten nicht-linearen  $\sigma$ -Modellen ( $N > 1$ ) in drei Dimensionen gibt es zwei relevante Skalenfelder: Ein temperaturartiges Skalenfeld  $u_t$  und ein Skalenfeld  $u_h$  das mit dem äußeren Feld zusammenhängt. Alle anderen Skalenfelder sind irrelevant. Die Exponenten  $\nu$ ,  $\gamma$ ,  $\alpha$ ,  $\beta$ ,  $\delta$ ,  $\eta$  die wir bisher eingeführt hatten lassen sich auf die beiden RG-Exponenten  $y_t$  und  $y_h$  zurückführen.

Wir starten mit der RG-Transformation nahe bei  $\beta_c$  und  $h = 0$  und führen  $n_0$  RG-Transformationen durch.  $n_0$  ist so gewählt, daß die irrelevanten Skalenfelder vernachlässigbar klein werden, aber  $u_t$  und  $u_h$  noch klein genug sind, so daß wir die RG-Transformation linear approximieren können.

Da wir nur endlich viele RG-Transformationen durchgeführt haben, können wir  $u_t$  und  $u_h$  in  $t = (\beta - \beta_c)/\beta_c$  und  $h$  Taylor-entwickeln:

$$u_t^{(n_0)} = c_t t + O(t^2, h^2) \quad (187)$$

und

$$u_h^{(n_0)} = c_h h + O(ht) . \quad (188)$$

Der obere Index  $n_0$  deutet an, daß es sich um den Wert von  $u_t$  bzw.  $u_h$  nach  $n_0$  RG-Transformationen handelt. Der Grund dafür, daß nur Terme  $O(t^2, h^2)$  und  $O(ht)$  in der zweiten Ordnung der Entwicklung auftauchen, ist die Symmetrie des Problems.

Um die Potenzgesetze herzuleiten, müssen wir das Transformationsverhalten der Observablen und der Skalenfelder zusammenbringen. Betrachten wir als erstes die Korrelationslänge im thermodynamischen Limes für  $h = 0$ :

Nach  $n_0$  (und mehr) Transformationen hängt  $\xi$  nur noch (bis auf kleine Korrekturen durch irrelevante Skalenfelder) von  $u_t$  ab. Transformieren wir nach  $n_0$  weitere  $n$ -mal, ergibt sich:

$$b^{-n} \xi(b^{ny_t} u_t^{(n_0)}) = \xi(u_t^{(n_0)}) . \quad (189)$$

Wählen wir nun einen festen Wert

$$\tilde{u}_t = b^{ny_t} u_t^{(n_0)} , \quad (190)$$

so ist auch  $\xi(b^{ny_t} u_t^{(n_0)})$  eine feste Zahl. Lösen wir Gleichung (190) nach  $b^n$  auf und setzen in Gleichung (189) ein, so ergibt sich

$$\xi(u_t^{(n_0)}) = \text{const} (u_t^{(n_0)})^{-1/y_t} . \quad (191)$$

Mit Gleichung (187) kommen wir nun zu dem versprochenen Resultat:

$$\xi(t) = f t^{-1/y_t} (1 + \dots + ct \dots) . \quad (192)$$

Damit haben wir gezeigt, daß  $\nu = 1/y_t$ . Der Korrekturterm  $ct$  stammt von  $O(t^2)$  in Gleichung (187). Berücksichtigt man das Skalenfeld  $u_3$  in der Rechnung, wobei  $y_3$  der größte irrelevante RG-Exponent ist, so ergibt sich

$$\xi(t) = f t^{-1/y_t} (1 + at^\theta + ct \dots) . \quad (193)$$

mit  $\theta = -y_3/y_t = \omega\nu$ .

Weitere Potenzgesetze können wir mittels der freien Energiedichte ableiten:

$$f(t, h) = \frac{1}{V} \ln Z(t, h) \quad (194)$$

Z.B.:

$$\langle m \rangle (t, h) = \frac{\partial f(t, h)}{\partial h} \quad (195)$$

oder

$$\chi(t, h) = \frac{\partial^2 f(t, h)}{\partial h^2} . \quad (196)$$

Zunächst schauen wir uns an, wie sich freie Energiedichte transformiert:

$$f' = b^d f . \quad (197)$$

Nun müssen wir  $f$  noch in einen sigulären und einen nicht-singulären Teil aufspalten:

$$f = f_{ns} + f_s . \quad (198)$$

Der Teil  $f_{ns}$  wird aufgrund der multiplikativen Konstanten der Zustandssumme unter der RG-transformation benötigt. Analog zu Gleichung (189) erhalten wir

$$b^{-nd} f_s(b^{ny_t} u_t^{(n_0)}, b^{ny_h} u_h^{(n_0)}) = f_s(u_t^{(n_0)}, u_h^{(n_0)}) . \quad (199)$$

Wieder wählen wir  $\tilde{u}_t = b^{ny_t} u_t^{(n_0)}$  fest und lösen nach  $b^n$  auf.

Es ergibt sich:

$$f_s(u_t, u_h) = \text{const } u_t^{d/y_t} \tilde{\Phi}(u_h u_t^{-y_h/y_t}) . \quad (200)$$

Einsetzen von  $h$  und  $t$ :

$$f_s(t, h) \propto t^{d/y_t} \Phi(h t^{-y_h/y_t}) \quad (201)$$

Durch Ableiten nach  $h$  an der Stelle  $h = 0$  erhalten wir die Potenzgesetze für die Magnetisierung und die magnetische Suszeptibilität.

$$\beta = \frac{d - y_h}{y_t} \quad (202)$$

und

$$\gamma = \frac{2y_h - d}{y_t} . \quad (203)$$

Als einfaches Beispiel für Finite Size Scaling können wir diskutieren, wie sich die Binderkumulante verhält. Zunächst einmal ist die Binderkumulante eine Funktion der Gittergröße  $L$  und  $u_t$ .

$$U(b^{-n} L, b^{ny_t} u_t) = U(L, u_t) . \quad (204)$$

Als erstes sehen wir, daß am kritischen Punkt ( $u_t = 0$ )  $U$  nicht von  $L$  abhängt. Als nächstes interessiert uns die Ableitung von  $U$  nach  $t$  und damit nach  $u_t$  am kritischen Punkt. Wir erhalten:

$$b^{ny_t} U'(L, 0) = U'(b^n L, 0) \quad (205)$$

Die Ableitung wächst also mit der Systemgröße wie

$$U'(L, 0) \propto L^{y_t} . \quad (206)$$

Für weitere Zusammenhänge zwischen den Exponenten siehe die Literatur, z.B: J.Cardy, Scaling and Renormalization in Statistical Physics Seite 43 bis 46.



## 12.1 RG-Transformationen mittels Monte-Carlo

Man kann auf verschiedene Arten versuchen, RG-Transformationen zu berechnen. Die meisten analytischen Rechnungen basieren auf der Impulsraumdarstellung. Es wird dabei meistens Störungstheorie um das Gaußsche Modell betrieben.

Im Ortsraum lassen sich RG-Transformationen für eindimensionale Modelle sehr einfach exakt durchführen. Aber schon bei zweidimensionalen Modellen (insbesondere auch beim Ising-Modell) ist man auf Näherungen und eine Beschränkung der Kopplungskonstanten auf endlich viele (praktisch wenige) angewiesen.

Insbesondere für das zweidimensionale Ising-Modell gibt es eine Reihe von Arbeiten, in denen versucht wird RG-Transformationen zu rechnen. Dabei werden entweder exakte Rechnungen für endliche (kleine) Systeme oder die Kumulantenentwicklungen verwendet. Eine gute Einführung in solche Rechnungen bekommt man durch Kapitel 5.9 in

J.J. Binney et al. :

The Theory of Critical Phenomena, An Introduction to the Renormalization Group. Oxford Science publications.

Auch die Monte-Carlo-Simulation kann als Rechenmethode für die RG-Transformation verwendet werden. In der Monte-Carlo-Simulation kann man nun im wesentlichen zwei Dinge versuchen:

- Lediglich die Berechnung der Matrix  $T$
- Berechnung der gesamten RG-Transformation

### 12.1.1 Berechnung der Matrix $T$

Merkwürdigerweise heißt dies in der Literatur “Monte Carlo Renormalization Group”  
Referenzen dazu sind:

S.-K. Ma, Phys. Rev. Lett. 37 (1976) 461.

Neuere Arbeiten zu MCRG sind:

C.F. Baillie, R. Gupta, K.A. Hawick, and G.S. Pawley, Phys. Rev. B 45 (1992) 10438, and references

Rajan Gupta, Pablo Tamayo  
Critical Exponents of the 3-D Ising Model  
cond-mat/9601048

H. W. J. Blöte, J. R. Heringa, A. Hoogland, E. W. Meyer, T. S. Smit  
Monte Carlo Renormalization of the 3-D Ising model: Analyticity and Convergence  
cond-mat/9602020

#### **Beispiel: 3D Ising-Modell**

Man simuliert das Modell bei  $\beta_c$  mit dem nächsten-Nachbar-Hamiltonian (bis auf

Blöte et al., welche auch übernächste und überübernächste Nachbarkopplungen betrachten) auf einem Gitter der Größe  $L^3$ . Die erzeugten Konfigurationen werden nun sukzessive geblockt auf Gitter der Größe  $L/2, L/4, \dots$  usw. bis eine Mindestgittergröße  $L_{min}$  erreicht ist. D.h., man kann umso mehr Blockungen durchführen (und damit  $\vec{K}^*$  umso näher kommen), je größer das simulierte Gitter ist.

Als Blockungsregel wird in den oben angegebenen Beispielen meistens die Majoritätsregel (auch hier Ausnahme: Blöte et al.) verwendet.

Nehmen wir an, daß sich die Hamiltonfunktion in der Form

$$H = - \sum_{\alpha} K_{\alpha} S_{\alpha} \quad (207)$$

schreiben läßt. In der Praxis (Gupta, Tamayo) werden in den Ansätzen bis zu 53 gerade und 46 ungerade Kopplungen verwendet.

Versehen wir die  $K_{\alpha}$  noch mit einem weiteren Index, um anzudeuten, zu welchem Blocklevel die Kopplung gehört :  $K_{\alpha}^{(i)}$ .

Um  $T$  zu bestimmen, müssen die partiellen Ableitungen

$$T_{\alpha,\beta}^{(i)} = \frac{\partial K_{\alpha}^{(i+1)}}{\partial K_{\beta}^{(i)}} \quad (208)$$

berechnet werden. Wie man dies mit Hilfe von Erwartungswerten ausdrücken kann, hat S.-K. Ma herausgefunden. Wir definieren

$$A_{\alpha,\beta}^{(i)} := \frac{\partial \langle S_{\alpha}^{(i+1)} \rangle}{\partial K_{\beta}^{(i)}} = \langle S_{\alpha}^{(i+1)} S_{\beta}^{(i)} \rangle - \langle S_{\alpha}^{(i+1)} \rangle \langle S_{\beta}^{(i)} \rangle \quad (209)$$

und

$$B_{\alpha,\beta}^{(i)} := \frac{\partial \langle S_{\alpha}^{(i+1)} \rangle}{\partial K_{\beta}^{(i+1)}} = \langle S_{\alpha}^{(i+1)} S_{\beta}^{(i+1)} \rangle - \langle S_{\alpha}^{(i+1)} \rangle \langle S_{\beta}^{(i+1)} \rangle \quad (210)$$

Damit ergibt sich  $T$  aus der Matrixgleichung

$$A = T B \quad (211)$$

Die Eigenwerte  $y_t$  und  $y_h$  können nun durch Diagonalisieren von  $T$  berechnet werden. Das wichtigste Problem bei diesem Vorgehen ist, daß der Raum der Kopplungen  $K$  trunkiert ist. Es ist auch nicht geklärt, wie man den Raum der Kopplungen systematisch vergrößert, so daß die berechneten  $y_t$  und  $y_h$  gegen die wahren Werte konvergieren. (Vorschlag von Blöte et al.: Importance index, der die Zahl der Spins und den Abstand der Spins voneinander berücksichtigt.)

### 12.1.2 Berechnung der RG-Transformation selbst

Es gibt verschiedene Möglichkeiten, die RG-Transformation  $\vec{K} \rightarrow \vec{K}'$  in der Monte-Carlo-Simulation auszuführen. Hier möchte ich mich auf eine konzeptionell einfache Methode beschränken. Bei dieser Methode kommen Dämonen zum Einsatz. Dämonen hatten wir bereits im Kapitel zum klassischen Heisenberg-Modell eingeführt.

Referenzen:

M. Creutz, A. Gocksch and M. Ogilvie, Phys.Rev.Lett. 53 (1984) 875.

M. Hasenbusch, K. Pinn, and C. Wierczkowski  
Canonical Demon Monte Carlo Renormalization Group  
Phys. Lett. B338 (1994) 308, hep-lat/9406019.

Die Dämonen, die wir hier verwenden, leben in einem Intervall  $[0, d_{max}]$ . Den Erwartungswert des Dämons können wir als Funktion der Kopplung leicht ausrechnen:

$$\langle d \rangle = \int_0^{d_{max}} dd \exp(-Kd) = \frac{\exp(-d_{max}K)(-d_{max}K - 1) + 1}{K^2} . \quad (212)$$

Kennen wir nun  $\langle d \rangle$  aus der Simulation, so können wir die obige Gleichung numerisch nach  $K$  auflösen. Der Dämon kann damit als "Thermometer" dienen.

Unsere Thermometer (Dämonen) müssen wir mit dem geblockten System in Kontakt bringen. Auf dem Niveau der Hamiltonfunktion sieht das so aus:

$$H_{comb} = - \sum_{\alpha} K'_{\alpha} (S_{\alpha} - d_{\alpha}) . \quad (213)$$

Damit unser Thermometer arbeiten kann, muß es zu einem Energieaustausch zwischen dem geblockten Spinsystem und den Dämonen kommen. Dazu verwenden wir einen update wie bereits in dem Kapitel zum Heisenberg-Modell besprochen.

- 1) Zuerst wird ein neuer Spin  $s'$  vorgeschlagen.
- 2) Es wird die dadurch hervorgerufene Änderung  $\Delta S_{\alpha}$  für alle Terme der Hamiltonfunktion berechnet. Daraus ergeben sich die  $d'_{\alpha} = d_{\alpha} + \Delta S_{\alpha}$ .
- 3) Falls für alle  $\alpha$  gilt  $0 < d'_{\alpha} < d_{\alpha, max}$ , wird der Vorschlag für die  $s'$  und  $d'_{\alpha}$  angenommen.

## 13 Danksagung

Mein Dank gilt meiner Ehefrau Almuth Haensch und den Herrn Tibor Török und Heiko Molke, Studenten am Lehrstuhl COM, für das Korrekturlesen des Skripts.