

Monte Carlo-Simulationen  
zum kritischen Verhalten  
dünner Ising-Filme

Dissertation

zur Erlangung des Grades  
*Doktor der Naturwissenschaften*  
am Fachbereich Physik der  
Johannes Gutenberg-Universität Mainz,  
vorgelegt von

OLIVER DILLMANN  
geboren in Limburg

Mainz, im Juni 2000

Datum der mündlichen Prüfung: 29. Juni 2000

## Zusammenfassung

Dünne Ising-Filme können als vereinfachtes Modell zur Beschreibung von Flüssigkeiten in schlitzartigen Kapillaren oder von binären Mischungen, die als Film auf ein Substrat aufgebracht sind, dienen. Dabei wird der Einfluß der Wände und eines chemischen Potentials auf das System durch die Einführung von Magnetfeldern modelliert. Infolge der eingeschränkten Geometrie zeigen diese Systeme interessante Eigenschaften. So unterscheidet sich das kritische Verhalten signifikant von dem eines Bulk-Systems, es kommt zu einem Crossover von zwei- zu dreidimensionalem kritischem Verhalten. Zusätzlich verschiebt sich der Phasenübergang in den ungesättigten Bereich, ein Effekt, der in der Literatur als “capillary condensation” bezeichnet wird.

In der vorliegenden Arbeit wurden die kritischen Eigenschaften von Ising-Filmen im Rahmen einer Monte Carlo-Simulation untersucht. Dabei variierten die Schichtdicken der Filme zwischen  $D = 4$  und  $D = 32$ . Zur Verbesserung der Effizienz wurde ein Cluster-Algorithmus verwendet, der, um einen Ghost-Spin-Term zur Behandlung der Felder erweitert, in der Umgebung der kritischen Punkte eine bis zu einem Faktor 6 gegenüber konventionellen Algorithmen reduzierte Autokorrelationszeit zeigte. Bei der Datenanalyse kamen moderne Multi-Histogramm-Techniken zur Anwendung, die zu einer weiteren Verbesserung der Genauigkeit beitrugen.

Für jede Schichtdicke konnten kritische Temperatur und Magnetfeld mit sehr guter Genauigkeit bestimmt werden. Die Skalenhypothese von Fisher und Nakanishi, die die Verschiebung des kritischen Punktes gegenüber seines Bulk-Wertes beschreibt, wurde sowohl für Systeme mit freien Oberflächen als auch für Systeme mit schwachem Oberflächenfeld bestätigt. Der Wert des Gap-Exponenten der Oberfläche wurde mit  $\Delta_1 = 0.459(13)$  abgeschätzt, was in Übereinstimmung mit den in der Literatur angegebenen Werten ist, welche jedoch durch Reihenentwicklungen ermittelt wurden.

Die Observablen Magnetisierung und magnetische Suszeptibilität sowie deren auf die Oberfläche bezogenen Entsprechungen zeigen kein reines zweidimensionales kritisches Verhalten. Zu ihrer Beschreibung in der Nähe des kritischen Punktes wurden effektive Exponenten für die einzelnen Schichtdicken bestimmt, die sich für kleine Schichtdicken den zweidimensionalen kritischen Exponenten annähern. Abschließend wurden Energie- und Magnetisierungsprofile vorgestellt und diskutiert.



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
<b>2</b>	<b>Das Modell und seine Eigenschaften</b>	<b>5</b>
2.1	Übergang vom Gittergas-Modell zum Ising-Spin-Modell . . . . .	5
2.2	Einige einfache Eigenschaften des Modells . . . . .	7
<b>3</b>	<b>Die Simulation</b>	<b>10</b>
3.1	Monte Carlo-Simulationen . . . . .	10
3.1.1	Cluster-Algorithmen . . . . .	12
3.1.2	Ghostspin-Algorithmus . . . . .	15
3.1.3	Leistungsvergleich zwischen Ghostspin und Metropolis . . . . .	17
3.2	Reweighting . . . . .	22
3.2.1	Single-Histogram-Reweighting . . . . .	22
3.2.2	Multi-Histogramm-Reweighting . . . . .	27
3.3	Bestimmung der kritischen Punkte . . . . .	29
3.3.1	Spezifische Wärme und magnetische Suszeptibilität . . . . .	30
3.3.2	Kumulantenschnittpunkte . . . . .	34
3.4	Details der Simulation . . . . .	39
<b>4</b>	<b>Scaling-Theorie von Fisher und Nakanishi</b>	<b>42</b>

<b>5</b>	<b>Systeme mit Oberflächenfeld</b>	<b>48</b>
5.1	Kumulanten . . . . .	48
5.2	Bestimmung der kritischen Temperaturen . . . . .	54
5.3	Untersuchung des Fisher-Nakanishi-Scalings . . . . .	62
5.4	Untersuchung des 2d-kritischen Verhaltens . . . . .	68
5.5	Profile . . . . .	76
<b>6</b>	<b>Systeme mit freien Oberflächen</b>	<b>80</b>
6.1	Kumulanten . . . . .	80
6.2	Bestimmung der kritischen Temperaturen . . . . .	85
6.3	Untersuchung des Fisher-Nakanishi-Scalings . . . . .	92
<b>7</b>	<b>Zusammenfassung der Ergebnisse und Ausblick</b>	<b>95</b>
<b>A</b>	<b>Die Programme</b>	<b>105</b>

# Abbildungsverzeichnis

1.1	Skizze des Systems . . . . .	3
2.1	Skizze der Phasenkoexistenzkurve . . . . .	9
3.1	Binning für $D = 16$ und $h_1 = 0$ . . . . .	19
3.2	Binning für $D = 32$ und $h_1 = 0$ . . . . .	19
3.3	Binning für $D = 16$ und $h_1 = -0.015$ . . . . .	20
3.4	Binning für $D = 32$ und $h_1 = -0.015$ . . . . .	20
3.5	Histogramm bezüglich $m$ und $m_1$ . . . . .	25
3.6	Maxima von $\chi$ für verschiedene $L$ . . . . .	33
3.7	Maxima von $C_V$ für verschiedene $L$ . . . . .	33
3.8	Finite-Size-Effekte bei Kumulantenschnittpunkten . . . . .	38
3.9	Koexistenzkurve für $D = 28$ . . . . .	40
5.1	Kumulantenschnittpunkte für $D = 4$ . . . . .	49
5.2	Kumulantenschnittpunkte für $D = 8$ . . . . .	49
5.3	Kumulantenschnittpunkte für $D = 12$ . . . . .	50
5.4	Kumulantenschnittpunkte für $D = 16$ . . . . .	50
5.5	Kumulantenschnittpunkte für $D = 24$ . . . . .	51
5.6	Kumulantenschnittpunkte für $D = 28$ . . . . .	51
5.7	Kumulantenschnittpunkte für $D = 32$ . . . . .	52
5.8	Bestimmung von $T_c$ für $D = 4$ . . . . .	54
5.9	Bestimmung von $T_c$ für $D = 8$ . . . . .	55

5.10	Bestimmung von $T_c$ für $D = 12$ . . . . .	55
5.11	Bestimmung von $T_c$ für $D = 16$ . . . . .	56
5.12	Bestimmung von $T_c$ für $D = 24$ . . . . .	56
5.13	Bestimmung von $T_c$ für $D = 28$ . . . . .	57
5.14	Bestimmung von $T_c$ für $D = 32$ . . . . .	57
5.15	Master-Kurve für $\chi$ . . . . .	59
5.16	Master-Kurve für $c_V$ . . . . .	59
5.17	Scaling der $\Delta T_c(D)$ . . . . .	63
5.18	Extrapolation zu $D \rightarrow \infty$ für $-1/\nu$ . . . . .	64
5.19	Scaling der $\Delta h_c(D)$ . . . . .	65
5.20	Extrapolation zu $D \rightarrow \infty$ für $-(\Delta - \Delta_1)/\nu$ . . . . .	67
5.21	FSS von $m$ für $D = 4$ . . . . .	70
5.22	FSS von $m$ für $D = 32$ . . . . .	70
5.23	FSS von $m_1$ für $D = 4$ . . . . .	71
5.24	FSS von $m_1$ für $D = 32$ . . . . .	71
5.25	FSS von $\chi$ für $D = 4$ . . . . .	72
5.26	FSS von $\chi$ für $D = 32$ . . . . .	72
5.27	FSS von $\chi_{11}$ für $D = 4$ . . . . .	73
5.28	FSS von $\chi_{11}$ für $D = 16$ . . . . .	73
5.29	Magnetisierungsprofil $m^+$ bei $T_c(D)$ . . . . .	77
5.30	Magnetisierungsprofil $m^-$ bei $T_c(D)$ . . . . .	77
5.31	Magnetisierungsprofil $m$ bei $T_c(D)$ . . . . .	78
5.32	Energieprofil $u$ bei $T_c(D)$ . . . . .	78
6.1	Kumulantenschnittpunkte für $D = 4$ bei freien Oberflächen . . . . .	81
6.2	Kumulantenschnittpunkte für $D = 8$ bei freien Oberflächen . . . . .	82
6.3	Kumulantenschnittpunkte für $D = 12$ bei freien Oberflächen . . . . .	82
6.4	Kumulantenschnittpunkte für $D = 16$ bei freien Oberflächen . . . . .	83

6.5	Kumulantenschnittpunkte für $D = 24$ bei freien Oberflächen . . . . .	83
6.6	Kumulantenschnittpunkte für $D = 28$ bei freien Oberflächen . . . . .	84
6.7	Kumulantenschnittpunkte für $D = 32$ bei freien Oberflächen . . . . .	84
6.8	Bestimmung von $T_c$ für $D = 4$ bei freien Oberflächen . . . . .	85
6.9	Bestimmung von $T_c$ für $D = 8$ bei freien Oberflächen . . . . .	86
6.10	Bestimmung von $T_c$ für $D = 12$ bei freien Oberflächen . . . . .	86
6.11	Bestimmung von $T_c$ für $D = 16$ bei freien Oberflächen . . . . .	87
6.12	Bestimmung von $T_c$ für $D = 24$ bei freien Oberflächen . . . . .	87
6.13	Bestimmung von $T_c$ für $D = 28$ bei freien Oberflächen . . . . .	88
6.14	Bestimmung von $T_c$ für $D = 32$ bei freien Oberflächen . . . . .	88
6.15	Master-Kurve für $\chi$ bei freien Oberflächen . . . . .	90
6.16	Master-Kurve für $c_V$ bei freien Oberflächen . . . . .	90
6.17	Scaling der $\Delta T_c(D)$ bei freien Oberflächen . . . . .	93
6.18	Extrapolation zu $D \rightarrow \infty$ für $-1/\nu$ bei freien Oberflächen . . . . .	94

# Tabellenverzeichnis

3.1	Dynamische kritische Exponenten $z$ für 2D- und 3D-Ising . . . . .	14
5.1	Werte für $T_c(D)$ und $h_c(D)$ . . . . .	60
5.2	Scaling von $\Delta T_c$ . . . . .	64
5.3	Scaling von $\Delta h_c$ . . . . .	67
5.4	Werte der effektiven Exponenten für $m$ und $\chi$ . . . . .	75
5.5	Werte der effektiven Exponenten für $m_1$ und $\chi_{11}$ . . . . .	75
6.1	Werte für $T_c(D)$ bei freien Oberflächen . . . . .	91
6.2	Scaling von $\Delta T_c$ bei freien Oberflächen . . . . .	93

# Kapitel 1

## Einleitung

Flüssigkeiten in dünnen schlitzartigen Kapillaren und binäre Mischungen in dünnen Filmen auf Oberflächen sind von sehr großer technologischer Bedeutung, was durch eine Fülle von Anwendungen in den Materialwissenschaften begründet ist. Stellvertretend seien hier nur die Bereiche Adhäsion, Schmierung und die Beschichtung von Substraten genannt. So ist es nicht erstaunlich, daß sie Gegenstand regen experimentellen Interesses sind [1–15].

Aber auch vom theoretischen Standpunkt aus gesehen zeigen diese Systeme ungewöhnliche Eigenschaften. So unterscheidet sich ihr kritisches Verhalten signifikant von dem eines Systems im Volumen (Bulk): Es kommt zu einem “Crossover” von einem 2- zu einem 3-dimensionalen Verhalten am kritischen Punkt und die Phasenumwandlung von der flüssigen Phase zur Gasphase verschiebt sich in den ungesättigten Bereich [16], ein Phänomen, welches als “capillary condensation” bezeichnet wird. Ursache dafür sind zum einen Effekte durch die eingeschränkte Geometrie, da sich in einer dünnen Schicht die Ausdehnung in einer Raumrichtung von der in den beiden anderen unterscheidet, zum anderen Effekte durch die Wechselwirkung mit den Wänden, die, sofern es sich nicht um “neutrale” Wände handelt, eine Präferenz für eine der beiden Phasen zeigen. Eine weitere mit dem Einfluß der Wände auf das System verknüpfte Erscheinung ist der bei attraktiven Wänden auftretende Benetzungs-Übergang (Wetting-Übergang) [17–19]. Bei diesem kondensiert an der Wand eine makroskopisch dicke flüssige Lage eines gesättigten Gases, während die Partikel im Bulk in der Gasphase bleiben. Dem Wetting-Übergang kann ein sogenanntes “Prewetting” vorausgehen, in dessen Verlauf sich bereits in der ungesättigten Gasphase eine dünne Lage kondensierter Teilchen an der Wand niederschlägt.

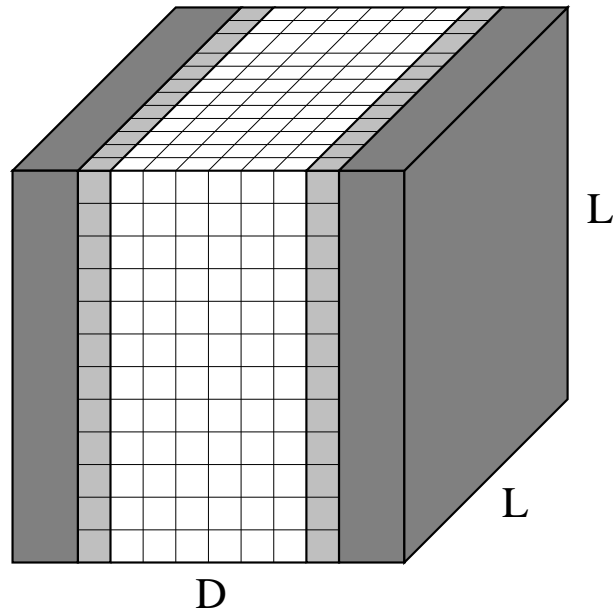
Die Beschreibung dieses reichhaltigen Spektrums an Phänomenen mit einer analytischen Theorie ist ein seit langer Zeit bestehendes [20] und aufgrund seiner Komplexität ungelöstes Problem [21–29]. Einen Beitrag zu dessen Lösung kann die Anwendung von Computersimulations-Techniken leisten. Die Simulation von kontinuierlichen Flüssigkeitsmodellen mit realistischen Potentialen hat jedoch mit den Schwierigkeiten zu kämpfen, daß nur verhältnismäßig kleine Systeme mit vertretbarem Aufwand untersucht werden können und schon alleine die ausreichend genaue Bestimmung der Phasenkoexistenzkurve im Volumen nur mit sehr großem numerischen Aufwand möglich ist [30–34]. Dies führt dazu, daß die mit dieser Technik erzielbaren Ergebnisse leider meist nur von qualitativem Charakter sind. Ein Weg, um dennoch zu quantitativen Aussagen zu gelangen, besteht in einer Vereinfachung des Modelles:

- Die Orte, die die Partikel einnehmen können werden vom Kontinuum auf die Kreuzungspunkte eines regelmäßigen 3-dimensionalen Gitters eingeschränkt.
- Die Kräfte zwischen den Teilchen werden nicht durch ein realistisches Potential, sondern durch eine attraktive Wechselwirkung zwischen auf dem Gitter benachbarten Flüssigkeitsteilchen bestimmt.

Das durch diese Einschränkungen entstandene Modell trägt den Namen “Gittergas-Modell”. Dem Nachteil, daß dieses auf *mikroskopischer* Ebene kein realistisches Abbild einer Flüssigkeit darstellt, stehen zwei bedeutende Vorteile gegenüber:

- Der Wert des chemischen Potentials  $\mu$ , bei dem im Bulk-System eine Phasenkoexistenz vorliegt, ist aufgrund von Symmetrieüberlegungen exakt bekannt.
- Die Gitterstruktur und das diskrete Spektrum der Energiezustände eines Teilchens erlauben die Implementierung hocheffizienter Algorithmen [35].

Eine dem Gittergas-Modell vollständig äquivalente Formulierung ist das Ising-Modell, in dem die Teilchen nicht durch eine Besetzungs-, sondern durch eine Ising-Spin-Variable, welche auch zwei Zustände annehmen kann, beschrieben werden. In dieser Beschreibung wird die Rolle des chemischen Potentials von einem Magnetfeld übernommen, während der Einfluß der Wände auf das System durch ein Randmagnetfeld vermittelt wird, dessen Reichweite auf die beiden Oberflächenschichten begrenzt ist. Die Wahl der Formulierung ist eine Frage der persönlichen Präferenz,



**Abb. 1.1:** Skizze eines Systems der Schichtdicke  $D$  und der lateralen Ausdehnung  $L$ . Die Wände sind dunkelgrau und die Schichten des Gitters, die dem Einfluß der Wände unterliegen, sind hellgrau unterlegt.

ob man lieber von Magnetisierung und Magnetfeldern oder von Dichten und chemischen Potentialen spricht.

Für die Untersuchung des kritischen Verhaltens von dünnen Filmen im Rahmen einer Monte Carlo-Simulation wurde für diese Arbeit die Beschreibung durch das Ising-Modell gewählt. Als Simulationsalgorithmus wurde ein Cluster-Algorithmus verwendet, um das an einem kritischen Punkt für lokale Algorithmen, deren prominentester Vertreter der Metropolis-Algorithmus [36] ist, auftretende “critical slowing down” zu mindern. Um den Magnetfeldern Rechnung zu tragen, wurde dieser Algorithmus um einen “Ghost-Spin”-Term erweitert.

Am Anfang der Untersuchung des kritischen Verhalten eines Systemes steht naturgemäß die Bestimmung des kritischen Punktes. Diese mußte für jede einzelne der verschiedenen Schichtdicken  $D$  durchgeführt werden. Dabei wurden zwei Konstellationen für den Einfluß der Wände vorgegeben:

Zum einen neutrale Wände, die keinen Einfluß auf die Oberflächenschichten des Systems ausüben. Oberflächeneffekte sind in diesem Fall auf die fehlenden Nachbarn der in den Oberflächen lokalisierten Teilchen zurückzuführen. Dieser Fall wird durch die Wahl eines verschwindenden Oberflächenmagnetfeldes modelliert. Zum anderen

Wände, die gering mit den sich in den oberen Schichten befindenden Teilchen wechselwirken. Diese werden durch ein schwaches Oberflächenfeld realisiert.

Ein Ziel dieser Arbeit ist es, die Gültigkeit der Skalen-Theorie von Fisher und Nakanishi [22, 23] für den Fall nicht neutraler Wände zu prüfen. Die Theorie beschreibt das Skalieren der Verschiebung des kritischen Punktes relativ zu seinem Bulk-Wert mit der untersuchten Schichtdicke. Ein weiterer Schwerpunkt ist die Untersuchung des “Crossovers” von einem 2- zu einem 3-dimensionalen kritischen Verhalten und die Form der Magnetisierungs- und Energieprofile in der Nähe des kritischen Punktes.

Die Arbeit gliedert sich wie folgt:

Im nächsten Kapitel wird das zu untersuchende Ising-Modell vorgestellt und auf seine Äquivalenz zum Gittergas-Modell eingegangen. In einem weiteren Unterkapitel wird die Scaling-Theorie von Fisher und Nakanishi dargelegt. Das dritte Kapitel befaßt sich ausführlich mit der Methodik der Simulation. Nach einer allgemeinen Einführung in die Grundlagen der Monte Carlo-Simulationen wird der verwendete Algorithmus erläutert und die bei der Analyse der Daten zur Anwendung kommende Technik des Multi-Histogramm-Reweichtings sowie die Extrapolation der kritischen Temperatur aus den Kumulantenschnittpunkten und den Maxima kritischer Observablen dargestellt. In den beiden folgenden Kapiteln werden dann die Ergebnisse dieser Bemühungen präsentiert, im ersten der beiden für ein schwaches Oberflächenfeld, im zweiten für den feldfreien Fall. Das letzte Kapitel ist der Zusammenfassung der Ergebnisse gewidmet.

# Kapitel 2

## Das Modell und seine Eigenschaften

Zur vereinfachten Beschreibung von Flüssigkeiten in einer dünnen Kapillare bzw. binärer Mischungen in dünnen Filmen im Rahmen einer Gittertheorie bietet sich als “natürliche” Wahl das Modell des Gittergases an. Dennoch wurde in dieser Arbeit für die Untersuchung der Phänomene der Rahmen eines Ising-Modells gewählt. Diese Vorgehensweise wird durch eine vollständige Äquivalenz dieser beiden Modelle ermöglicht, welche im folgenden Unterkapitel näher erläutert werden wird. Im darauffolgenden Abschnitt werden in einer ersten groben Analyse einige asymptotische Eigenschaften des Modells untersucht. Der letzte Abschnitt befaßt sich mit der Skalen-Theorie von Fisher und Nakanishi, deren Ziel es ist, die Verschiebung des kritischen Punktes eines Systems von endlicher Schichtdicke  $D$  sowie dessen Phasenkoexistenzkurve zu beschreiben.

### 2.1 Übergang vom Gittergas-Modell zum Ising-Spin-Modell

Um das Verhalten eines dünnen flüssigen Filmes zu modellieren, wird der folgende Ausdruck für die potentielle Energie zugrunde gelegt:

$$\mathcal{H}(\{c_i\}) = -\varphi \sum_{\langle i,j \rangle} c_i c_j - \epsilon_1 \sum_{i \in \text{Oberfläche}} c_i . \quad (2.1)$$

Der Zustand dieses Systems wird durch einen Satz von Zustandsvariablen  $c_i$ ,  $i = 1, \dots, N$  beschrieben, die die Werte  $c_i = 1$ , falls der Gitterplatz von einem Teilchen besetzt ist, oder  $c_i = 0$ , falls der Platz leer bleibt, annehmen können. Mit  $N$  wird die Anzahl der Gitterplätze bezeichnet, die sich bei einem Film der Schichtdicke  $D$  und der lateralen Ausdehnung  $L$  zu  $N = DL^2$  ergibt.

Der erste Summand des Ausdrucks für die potentielle Energie resultiert aus einer Wechselwirkung der Teilchen untereinander, deren Reichweite auf die direkt angrenzenden Nachbarn beschränkt ist. Die Summation erstreckt sich somit über alle Paare benachbarter Teilchen. Durch die Wahl von  $\varphi > 0$  wird eine attraktive Wechselwirkung festgelegt. In der zweiten Summe schlägt sich die Wechselwirkung der Teilchen mit der Wand nieder. Auch hier wird durch die Wahl eines positiven  $\epsilon_1$  eine attraktive Wechselwirkung zwischen Wand und Teilchen festgelegt. Die Summation erstreckt sich hier allerdings nur auf die  $2L^2$  Teilchen, die sich in der Oberfläche des Gitters befinden, entsprechend der begrenzten Reichweite des attraktiven Potentials der Wand.

Mit der Einführung eines chemischen Potentials  $\mu$ , welches ein Maß für die Fluktuationen der Teilchenanzahl ist, erweitert sich der Ausdruck für die potentielle Energie um den Term  $-\mu \sum_i c_i$ .

Ersetzt man die Besetzungsvariable  $c_i$  durch die Spinvariable  $S_i$ , für die

$$S_i = 1 - 2c_i \quad (2.2)$$

mit  $S_i \in \{-1, +1\}$  gilt, transformiert sich das ursprüngliche Gittergas-Modell in einen Ising-Ferromagneten:

$$\mathcal{H}(\{c_i\}) - \mu \sum_i c_i = \mathcal{H}_0 + \mathcal{H}_{\text{Ising}}(\{S_i\}) . \quad (2.3)$$

Hier bezeichnet  $\mathcal{H}_0$  einen physikalisch irrelevanten konstanten Anteil

$$\mathcal{H}_0 = -L^2 \{z\varphi(D-2)/8 + \varphi(z-1)/4 + \epsilon_1 + D\mu/2\} \quad (2.4)$$

und  $\mathcal{H}_{\text{Ising}}$  die Hamilton-Funktion des Ising-Ferromagneten

$$\mathcal{H}_{\text{Ising}}(\{S_i\}) = -J \sum_{\langle i,j \rangle} S_i S_j - h \sum_i S_i - h_1 \sum_{i \in \text{Oberflächen}} S_i . \quad (2.5)$$

In diesem Modell wird die Wechselwirkung zwischen benachbarten Spins nun durch die Konstante  $J = \varphi/4$  bestimmt. Für die hier nicht betrachtete Wahl  $\varphi < 0$ , die einer repulsiven Wechselwirkung entspricht, hätte man einen Antiferromagneten erhalten. Der Einfluß der Wand wird durch ein Oberflächenfeld

$$h_1 = -\epsilon_1/2 + J \tag{2.6}$$

vermittelt, und an die Stelle des chemischen Potentials tritt ein Bulk-Magnetfeld

$$h = -\mu/2 - zJ, \tag{2.7}$$

welches an jeden Spin des Gitters koppelt. Die Konstante  $z$  steht für die Koordinationszahl des Gitters, d. h. die Anzahl der nächsten Nachbarn eines Gitterplatzes im Bulk. Im hier behandelten Fall handelt es sich um eine einfach-kubische Gitterstruktur, deren Koordinationszahl  $z = 6$  ist.

Man beachte, daß die Transformation (2.3) bedeutet, daß das Gittergas im großkanonischen Ensemble (das chemische Potential  $\mu$  ist die vorgegebene unabhängige Variable) dem Ising-Modell mit dem Magnetfeld als unabhängige Variable entspricht. Möchte man ein Gittergas-Modell bei fester (vorgegebener) Dichte betrachten, entspräche dies einem Ising-Modell bei fester Magnetisierung.

## 2.2 Einige einfache Eigenschaften des Modells

Um einen ersten groben Eindruck vom Verlauf der Phasenkoexistenzlinie zu gewinnen, bietet sich eine Untersuchung des Grundzustandes des Systems an. Dieser wird bei einer Temperatur  $T = 0$  eingenommen. Betrachtet man den Grenzfall unendlicher Schichtdicke ( $D \rightarrow \infty$ ), so kann der Einfluß des Randmagnetfeldes, welcher sich nur auf die beiden Oberflächen auswirkt, vernachlässigt werden. Das Verhalten wird durch die Volumeneigenschaften geprägt, die die des 3-dimensionalen Ising-Modells sind. In diesem Modell tritt eine Koexistenz der beiden Phasen, die durch eine positive bzw. negative Orientierung aller Spins charakterisiert sind, bei einem Magnetfeld  $h = 0$  auf. Aufgrund der Symmetrieeigenschaften der Hamiltonfunktion des Ising-Modells hängt der Wert des Koexistenzfeldes nicht von der Temperatur ab, so daß sich im Grenzfall  $D \rightarrow \infty$  die Phasenkoexistenzkurve zu  $h_{\text{coex}}(T) = 0$  ergibt. Übersetzt in die Sprache des Gittergas-Modells läßt sich mit (2.7) das korrespondierende

chemische Koexistenzpotential zu

$$\mu_{\text{coex}}^{\infty} = -z \frac{\varphi}{2} \quad (2.8)$$

bestimmen. Die Phasenkoexistenzlinie läßt sich somit im Grenzfall  $D \rightarrow \infty$  auf die des reinen Ising-Modells zurückführen und exakt bestimmen.

Im Falle endlicher Schichtdicken gestaltet sich die Situation ungleich schwieriger. Zwar ist es noch möglich, einen Ausdruck für das Koexistenzfeld bei  $T = 0$  zu finden, aber das im vorherigen Falle verwendete Symmetrieargument läßt sich auf diesen Fall nicht anwenden, um die Gültigkeit des Ausdrucks auf Temperaturen  $T > 0$  auszuweiten. Zur Bestimmung des Koexistenzfeldes im Grundzustand  $T = 0$  setzt man die Energien der beiden Phasen, deren Spins eine gegensätzliche Orientierung aufweisen, gleich. Der Term der Spin-Spin-Wechselwirkung in (2.5) ist invariant unter einem Vorzeichenwechsel der Spinvariablen im Gegensatz zu den Wechselwirkungstermen der Magnetisierung mit den Feldern. Somit ergibt sich

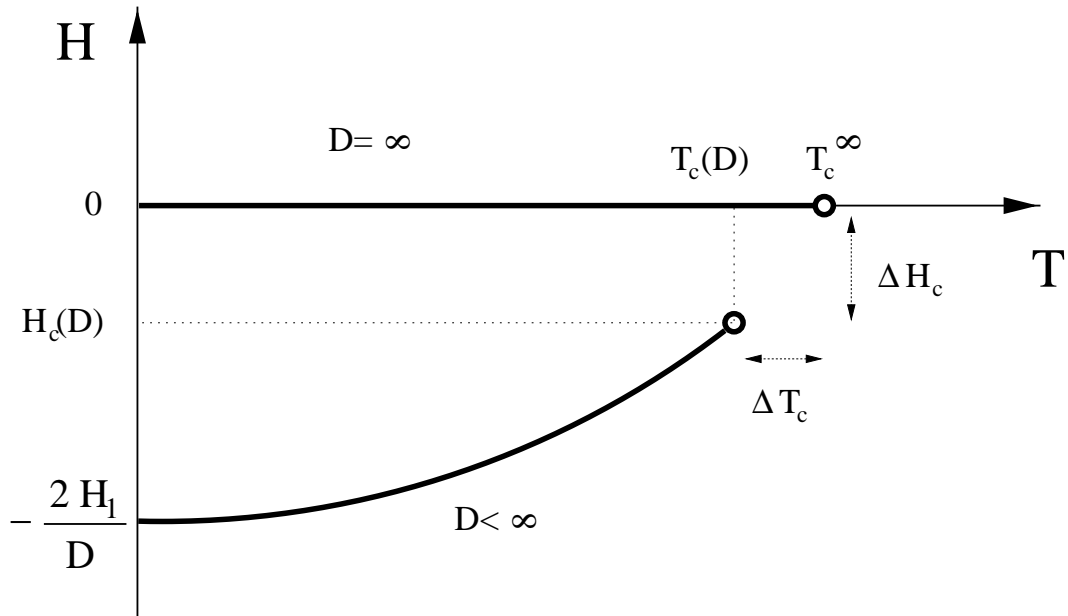
$$h_{\text{coex}}(T = 0) = -\frac{2h_1}{D}, \quad (2.9)$$

wobei verwendet wurde, daß der Anteil der Spins in den beiden Oberflächen an der Gesamtzahl der Spins  $2/D$  beträgt. Übertragen auf das Gittergas-Modell bedeutet dies eine Verschiebung des Koexistenzpotentials gegenüber seines Bulk-Wertes von

$$\begin{aligned} \mu_{\text{coex}}(T = 0) - \mu_{\text{coex}}^{\infty} &= -2h_{\text{coex}}(T = 0) \\ &= -\frac{1}{D} (2\epsilon_1 + \varphi), \end{aligned} \quad (2.10)$$

wobei (2.6), (2.7) und (2.9) Anwendung fanden.

Die Verschiebung der Phasenkoexistenz im Grundzustand in den Bereich eines negativen Magnetfeldes  $h < 0$  (eine positives Oberflächenmagnetfeld vorausgesetzt) läßt sich anschaulich dadurch erklären, daß bei verschwindendem Magnetfeld  $h = 0$  im Grundzustand die Phase energetisch günstiger wäre, die die gleiche Orientierung wie das Randmagnetfeld aufweist und somit die Entartung des Grundzustandes aufgehoben wäre. Erst durch die Wahl eines dem Randmagnetfeld entgegengesetzt orientierten Bulk-Magnetfeldes ist es möglich, daß im Grundzustand die andere Phase mit der gleichen Energie koexistieren kann. Auch für Temperaturen  $T > 0$  bleibt diese



**Abb. 2.1:** Skizzierter Verlauf der Phasenkoexistenzkurve für endliche und unendliche Schichtdicken  $D$  bei positivem Randmagnetfeld  $H_1 > 0$ . Die Verschiebung des kritischen Punktes bei endlicher Dicke ist mit  $\Delta T_c$  und  $\Delta H_c$  bezeichnet.

Verschiebung des Phasenkoexistenzfeldes gültig, so daß die gesamte Koexistenzkurve für positive Randfelder in den Bereich eines negativen Magnetfeldes verschoben ist [22]. In Abb. 2.1 ist der Verlauf der Kurve skizziert.

# Kapitel 3

## Die Simulation

### 3.1 Monte Carlo-Simulationen

Eines der Grundprobleme in der statistischen Mechanik ist die Berechnung der Zustandssumme eines Systems, die im kanonischen Ensemble als

$$Z(\beta) = \sum_{\{s_i\}} e^{-\beta H(\{s_i\})} \quad (3.1)$$

definiert ist. Ist die Zustandssumme bekannt, können mit Hilfe der Gleichungen der statistischen Mechanik die Erwartungswerte der Observablen berechnet werden. Die analytische Bestimmung von (3.1) ist nur für einige wenige recht einfach konstruierte Modelle möglich [37]. Der Versuch, (3.1) numerisch durch einfache Summation zu bestimmen, scheitert selbst für einfachste Systeme an der mit der Systemgröße rapide ansteigenden Zahl der Summanden, da die Summation über *jeden einzelnen* Zustand  $\{s_i\}$  des Systems zu erfolgen hat. An dieser Stelle setzt die Monte Carlo-Methode [36, 38–43] an, der die Idee zugrunde liegt, nicht die Zustandssumme, sondern Erwartungswerte beliebiger Observablen  $\mathcal{O}$ ,

$$\langle \mathcal{O} \rangle_\beta = \frac{1}{Z} \sum_{\{s_i\}} \mathcal{O}(\{s_i\}) e^{-\beta H(\{s_i\})}, \quad (3.2)$$

durch Summation über eine begrenzte Anzahl zufällig ausgewählter Zustände zu approximieren. Es empfiehlt sich meist nicht, die Zustände entsprechend einer konstanten Wahrscheinlichkeitsverteilung auszuwählen, weil die einen wesentlichen Beitrag zum Erwartungswert leistenden Konfigurationen im hochdimensionalen Phasenraum

in einem relativ schmalen Bereich konzentriert sind, der dann nur selten getroffen wird. Günstiger ist es, die Technik des *Importance Sampling* [38] zu verwenden, bei der die Zustände mit einer Wahrscheinlichkeit, die ihrem Boltzmann-Gewicht

$$P^B(\{s_i\}) \propto e^{-\beta H(\{s_i\})} \quad (3.3)$$

entspricht, generiert werden. Die Berechnung des Erwartungswertes (3.2) reduziert sich dann auf eine einfache Mittelwertbildung über den der Konfiguration  $\bar{s}_i$  entsprechenden Wert der Observablen  $\mathcal{O}(\bar{s}_i)$ , der “Meßwerte”.

Eine Kette von Zuständen, die gemäß einem vorgegebenen  $P(\{s_i\})$  verteilt sind, kann mit Hilfe eines Markov-Prozesses [44] realisiert werden. Dieser Prozeß wird durch eine Übergangsmatrix  $W(\{s_i\} \rightarrow \{s'_i\})$  charakterisiert, die die Wahrscheinlichkeit angibt, von Konfiguration  $\{s_i\}$  aus Zustand  $\{s'_i\}$  zu erreichen. Erfüllt die Matrix  $W$  die *Detailed-Balance*-Bedingung

$$P(\{s_i\}) W(\{s_i\} \rightarrow \{s'_i\}) = P(\{s'_i\}) W(\{s'_i\} \rightarrow \{s_i\}) , \quad (3.4)$$

so nähert sich die Verteilung der Markov-Kettenglieder der vorgegebenen Verteilung im Grenzfall unendlicher Kettenlänge an. Durch die Bedingung (3.4) ist die Markov-Kette *nicht eindeutig* bestimmt; es gibt verschiedene Markov-Prozesse, die der gleichen Gleichgewichtsverteilung entgegenstreben.

Das Generieren der Konfigurationen durch Realisierung einer Markov-Kette hat zwei Nachteile. Erstens kann man nur Meßwerte zur Mittelwertbildung heranziehen, die nach einer gewissen Zeitspanne gewonnen wurden, welche die Markov-Kette benötigt, um aus der Startkonfiguration ins Gleichgewicht zu relaxieren. Zweitens sind die Meßwerte der Kette zeitlich korreliert.

Ein Maß zur Beschreibung dieser Autokorrelationen ist die integrierte Autokorrelationszeit  $\tau_{\text{int}}^{\mathcal{O}}$ , die für einen Simulationslauf von  $N$  Messungen der Observablen  $\mathcal{O}$  durch

$$\tau_{\text{int}}^{\mathcal{O}} = \frac{1}{2} + \sum_{j=1}^N \Phi_{\mathcal{O}}(j) \left(1 + \frac{j}{N}\right) \quad (3.5)$$

definiert ist, unter Verwendung der Autokorrelationsfunktion

$$\Phi_{\mathcal{O}}(j) = \frac{\langle \mathcal{O}_i \mathcal{O}_{i+j} \rangle - \langle \mathcal{O}_i \rangle \langle \mathcal{O}_i \rangle}{\langle \mathcal{O}_i^2 \rangle - \langle \mathcal{O}_i \rangle \langle \mathcal{O}_i \rangle}. \quad (3.6)$$

Für einen ökonomischen Monte Carlo-Algorithmus ist es wichtig, eine Markov-Kette zu erzeugen, deren Autokorrelationszeit möglichst klein ist, da durch die Korrelationen der statistische Fehler des Mittelwertes der Observablen um einen Faktor  $\sqrt{2\tau_{\text{int}}^{\mathcal{O}}}$  erhöht wird [45, 46].

Ein häufig verwendeter Vertreter aus der Klasse Importance-Sampling-Algorithmen ist der Metropolis-Algorithmus [36]. Bei diesem lokalen Algorithmus unterscheiden sich die aufeinanderfolgenden Konfigurationen nur durch die Zustandsänderung eines einzelnen Spins bzw. Teilchens. Die Übergangsmatrix  $W$  ist durch

$$W(\{s_i\} \rightarrow \{s'_i\}) = \begin{cases} e^{-\beta(H(\{s'_i\}) - H(\{s_i\}))}, & \text{wenn } H(\{s'_i\}) > H(\{s_i\}), \\ 1, & \text{wenn } H(\{s'_i\}) \leq H(\{s_i\}), \end{cases} \quad (3.7)$$

gegeben. Ein Spinflip wird also immer akzeptiert, wenn die dadurch erreichte neue Konfiguration eine niedrigere Energie besitzt. Falls der neue Zustand jedoch energetisch ungünstiger ist, erfolgt der Spinflip mit einer nur von der Energiedifferenz der beiden Zustände abhängigen Wahrscheinlichkeit.

Die Stärke des Metropolis-Algorithmus ist die durch seine Schlichtheit ermöglichte große Flexibilität, die zu einer Anwendung bei einer Vielzahl von völlig unterschiedlichen physikalischen Systemen geführt hat. Sein Nachteil ist die mit der räumlichen Korrelationslänge  $\xi$  ansteigende Autokorrelationslänge, deren Zusammenhang als

$$\tau \propto \xi^z \quad (3.8)$$

ermittelt wurde, wobei  $z$  den durch (3.8) definierten dynamischen kritischen Exponenten bezeichnet, für den im Fall lokaler Algorithmen  $z \approx 2$  [47, 48] gilt. Dies führt dazu, daß in der Nähe eines kritischen Punktes, wo  $\xi$  divergiert bzw. in endlichen Systemen den Wert der linearen Gitterlänge  $L$  annimmt, die Autokorrelationszeit  $\tau$  stark ansteigt, es also zu einem *critical slowing down* kommt.

### 3.1.1 Cluster-Algorithmen

Die Suche nach Algorithmen, deren Effizienz nicht durch das dem Metropolis-Verfahren immanente Problem des *critical slowing down* beschränkt ist, führte zur

Entwicklung von Cluster-Algorithmen [49, 50]. Dieser Fortschritt mußte jedoch auf Kosten der vielseitigen Anwendbarkeit des Verfahrens erkauft werden. Im folgenden soll der Swendsen-Wang-Algorithmus für Ising-Modelle beschrieben werden. Dieses Verfahren läßt sich ohne weiteres auf  $q$ -Zustands-Potts-Modelle [51] verallgemeinern, eine Modellklasse, die auch das Ising-Modell als Spezialfall ( $q = 2$ ) umfaßt. Auch eine Erweiterung auf  $O(n)$ -Spin-Modelle mit  $n \geq 2$  (das Ising-Modell ist ein Vertreter dieser Modellklasse mit  $n = 1$ ) ist möglich [50, 52–55].

Ausgangspunkt ist die Zustandssumme des Ising-Modells

$$Z(\beta) = \sum_{\{s_k\}} \exp \left( \beta \sum_{\langle ij \rangle} s_i s_j \right), \quad \text{mit } \beta = \frac{J}{k_B T}. \quad (3.9)$$

Das Produkt  $s_i s_j$  kann im Ising-Modell nur die Werte  $s_i s_j = \pm 1$  annehmen. Führt man die Abkürzung  $p = 1 - e^{-2\beta}$  ein, läßt sich (3.9) umformen zu

$$Z(\beta) = \sum_{\{s_k\}} \prod_{\langle ij \rangle} e^\beta \left[ (1 - p) + p \delta_{s_i s_j} \right]. \quad (3.10)$$

Benutzt man die triviale Identität

$$A + B = \sum_{b=0}^1 (A \delta_{b,0} + B \delta_{b,1}), \quad (3.11)$$

und assoziiert zu jedem Nachbarpaar  $\langle ij \rangle$  eine *Bondvariable*  $b_{ij}$ , die die Werte 0 (“gebrochener” Bond) und 1 (“aktiver” Bond) annehmen kann, so mündet die Zustandssumme nach einer zusätzlichen Summation über alle zulässigen Bondkonfigurationen  $\{b_{ij}\}$  in die sogenannte Fortuin-Kasteleyn-Darstellung [56–59]:

$$Z_{\text{FK}}(\beta) = \sum_{\{s_k\}} \sum_{\{b_{ij}\}} \prod_{\langle ij \rangle} e^\beta \left[ (1 - p) \delta_{b_{ij},0} + p \delta_{s_i s_j} \delta_{b_{ij},1} \right]. \quad (3.12)$$

Der Übergang von einem Glied der Markov-Kette zum darauffolgenden, ein “Update” der Konfiguration, untergliedert sich nun in zwei Teilschritte. Es wird abwechselnd bei gegebener Spinkonfiguration eine neue Bondkonfiguration und danach bei festgehaltener Bondkonfiguration eine neue Spinkonfiguration ausgewählt. Dies läßt sich als Algorithmus direkt formulieren:

1. Weise jedem Nachbarpaar  $\langle ij \rangle$  auf dem Gitter einen Bond  $b_{ij}$  zu mit dem

Wert  $b_{ij} = 0$  falls  $s_i \neq s_j$ . Im Falle  $s_i = s_j$  bekommt  $b_{ij}$  den Wert 1 mit einer Wahrscheinlichkeit von  $p$ , den Wert 0 mit einer von  $1 - p$  zugewiesen.

2. Identifiziere die Cluster, d.h. die Spins, die mit einem aktiven Bond  $b_{ij} = 1$  miteinander verbunden sind, wobei ein Cluster auch aus einem einzelnen Spin bestehen kann.
3. Ziehe für jedes Cluster separat einen zufälligen Spinwert  $\pm 1$ , der jedem einzelnen Spin im entsprechenden Cluster zugewiesen wird.
4. Beginne wieder mit Schritt 1.

Dieses Verfahren wird als *Swendsen-Wang-Cluster-Algorithmus* bezeichnet. Charakteristisch für ihn ist, daß das gesamte Gitter in Cluster aufgeteilt wird und somit bei einem Durchlauf des Algorithmus ein Invertieren bei jedem Spin des Gitters versucht wird.

Eine Variation des *Swendsen-Wang-Cluster-Algorithmus* ist der *Wolff-Cluster-Algorithmus* [50] oder auch *Single-Cluster-Algorithmus*. Er basiert auf der Konstruktion eines einzelnen Clusters, die von einem zufällig ausgewählten Gitterplatz startet. Das so erzeugte einzelne Cluster ändert *immer* seine Orientierung.

Algorithmus	$D = 2$	$D = 3$	Observable
Metropolis	2.125	2.03	
Swendsen-Wang	0.27(2)	0.50(3)	$z_E$
	0.20(2)	0.50(3)	$z_\chi$
Wolff	0.26(2)	0.28(2)	$z_E$
	0.13(2)	0.14(2)	$z_\chi$

**Tab. 3.1:** Dynamische kritische Exponenten  $z$  für das 2D- und 3D-Ising-Modell (nach [60] und [61]).

Die Effizienz der Cluster-Algorithmen in der Nähe eines kritischen Punktes ist wesentlich größer als die des Metropolis-Algorithmus. Dies liegt darin begründet, daß Cluster-Algorithmen nicht nur einzelne, sondern ganze Domänen von Spins umklappen. Diese Domänen haben die Größenordnung der mittleren Clustergröße, was in etwa der Skala der räumlichen Korrelationslänge entspricht.

Um einen Eindruck von der Effizienzsteigerung zu geben, sind in Tab. 3.1 die dynamischen kritischen Exponenten für das 2- und 3-dimensionale Ising-Modell aufgeführt (nach [60] und [61]).

### 3.1.2 Ghostspin-Algorithmus

Die im Kap. 3.1.1 vorgestellten Cluster-Algorithmen sind in ihrem Anwendungsbereich auf Spin-Modelle *ohne* Wechselwirkung mit einem Magnetfeld beschränkt. Um sie auch für das in dieser Arbeit untersuchte Modell (vgl. (2.5)) nutzbar zu machen, ist eine Verallgemeinerung des Konzeptes, die einem äußeren Feld Rechnung trägt, notwendig [62–64].

Die Zustandssumme für ein Ising-Modell im magnetischen Feld ist von folgender Form (vgl. (3.9)):

$$Z(\beta, h) = \sum_{\{s_k\}} \exp \left( \beta \sum_{\langle ij \rangle} s_i s_j + \beta h \sum_n s_n \right). \quad (3.13)$$

Durch künstliche Einführung eines *Ghostspins*  $s_G$ , der nur die Werte der Ising-Spins annehmen kann,  $s_G = \pm 1$ , und immer in Richtung des Magnetfeldes ausgerichtet bleibt,  $h = |h|s_G$ , läßt sich (3.13) zu

$$Z(\beta, h) = \sum_{\{s_k\}} \exp \left( \beta \sum_{\langle ij \rangle} s_i s_j + \beta |h| \sum_n s_n s_G \right) \quad (3.14)$$

umformen. Der Term, welcher die Wechselwirkung mit dem Magnetfeld beschreibt, ist nun formal von der gleichen Gestalt wie der der Spin-Spin-Wechselwirkung.

Somit läßt sich mit Hilfe einer zweiten Bondvariablen, dem *Ghostbond*  $g_n$ , eine der Fortuin-Kasteleyn-Form (3.12) analoge Darstellung der Zustandssumme finden:

$$\begin{aligned} Z(\beta, h) &= \sum_{\{s_k\}} \prod_{\langle ij \rangle} e^{\beta [(1-p) + p \delta_{s_i s_j}]} \prod_n e^{\beta |h| [(1-p_G) + p_G \delta_{s_n s_G}]} \\ &= \sum_{\{s_k\}} \sum_{\{b_{ij}\}} \prod_{\langle ij \rangle} \left( e^{\beta [(1-p) \delta_{b_{ij},0} + p \delta_{s_i s_j} \delta_{b_{ij},1}]} \right) \times \\ &\quad \times \sum_{\{g_k\}} \prod_n \left( e^{\beta |h| [(1-p_G) \delta_{g_n,0} + p_G \delta_{s_n s_G} \delta_{g_n,1}]} \right), \end{aligned} \quad (3.15)$$

wobei die Variable  $p_G$  durch

$$p_G \equiv 1 - e^{-2\beta|h|} \quad (3.16)$$

definiert wird.

Entsprechend dem feldfreien Cluster-Algorithmus zerfällt auch hier der Übergang zum nächsten Glied der Markov-Kette in zwei Teilschritte. Es wird bei festgehaltener Spinkonfiguration eine neue Konfiguration sowohl der Bonds, als auch der Ghostbonds gewählt. Danach wird bei invarianter Konfiguration beider Bond-Arten ein Update der Spinzustände durchgeführt.

Der zugehörige Algorithmus hat dann folgende Gestalt:

1. Weise jedem Nachbarpaar  $\langle ij \rangle$  auf dem Gitter einen Bond  $b_{ij}$  wie im feldfreien Cluster-Algorithmus zu.
2. Verbinde jeden Spin des Gitters mit einer Wahrscheinlichkeit von  $p_G$  durch einen Ghostbond  $g_n$  mit dem Ghostspin, falls beide Spins die *gleiche* Orientierung haben.
3. Identifiziere die Cluster, d.h. die Spins, die mit einem aktiven Bond  $b_{ij} = 1$  oder einem aktiven Ghostbond  $g_n = 1$  miteinander verbunden sind, wobei ein Cluster auch aus einem einzelnen Spin bestehen kann.
4. Ziehe für jedes Cluster, welches *keine* Verbindung zum Ghostspin hat, separat einen zufälligen Spinwert  $\pm 1$ , der jedem einzelnen Spin im entsprechenden Cluster zugewiesen wird.
5. Beginne wieder mit Schritt 1.

Für die Implementierung dieses Algorithmus bietet sich noch eine kleine Variation an, die zu einer Vereinfachung und Beschleunigung führt. Diese Variation besteht darin, daß es für den Update eines Clusters nicht von belang ist, mit *wievielen* Ghostbonds es mit dem Ghostspin verbunden ist. Von Interesse ist nur die Tatsache, *ob* ein Cluster verbunden ist. Deshalb sollte bei der Identifizierung der feldfreien Cluster auch die Größe  $C$  des Clusters bestimmt werden, weil mit ihrer Hilfe die Wahrscheinlichkeit, daß das gesamte Cluster *nicht* durch eine Verbindung mit dem Ghostspin blockiert ist, durch

$$p^f(C) = e^{-2\beta|h|C} \quad (3.17)$$

berechnet werden kann. Somit kann die Wirkung des Magnetfeldes auf den Cluster unter Verwendung einer einzigen Zufallszahl bestimmt werden.

Bei Modellen mit inhomogenen Feldern muß eine entsprechende Anzahl von Ghostspins eingeführt werden. Bei dem in dieser Arbeit untersuchten Modell genügte zwei, da das Feld nur in den beiden Oberflächenschichten einen vom Bulk-Feld abweichenden Wert hatte. In solchen Fällen muß durch eine geeignete “Buchhaltung” sichergestellt werden, daß die einzelnen Spins eines Clusters dem richtigen Ghostspin zugeordnet werden können.

### 3.1.3 Leistungsvergleich zwischen Ghostspin und Metropolis

Bei der Auswahl des Simulationsalgorithmus war seine Effizienz in der Nähe des kritischen Punktes von entscheidender Bedeutung. In diesem Abschnitt sollen nun die Autokorrelationszeiten  $\tau_{\text{int}}$  des verwendeten Clusteralgorithmus mit denen eines Metropolis-Algorithmus direkt verglichen werden, um seine Eignung für die untersuchte Problemstellung beurteilen zu können. Zusätzlich ist es für die sorgfältige Analyse der Daten einer Monte Carlo-Simulation notwendig, sich ein Bild von der Größenordnung der Autokorrelationen zu verschaffen. Dieses Ziel kann durch die Analyse der Zeitreihen einer Observablen mittels der Methode des *binning* erreicht werden, die an dieser Stelle kurz erläutert werden soll.

Für das Fehlerquadrat korrelierter Daten einer Observablen  $\mathcal{O}$  gilt

$$\epsilon^2 = 2 \tau_{\text{int}}^{\mathcal{O}} \frac{\sigma_{\mathcal{O}_i}^2}{N} \quad . \quad (3.18)$$

Mit  $\tau_{\text{int}}^{\mathcal{O}}$  ist die integrierte Autokorrelationslänge bezeichnet,  $\sigma_{\mathcal{O}_i}^2/N$  ist das Quadrat des Fehlers, welcher sich bei naiver, d. h., die Autokorrelationen nicht berücksichtigender Fehlerrechnung, ergibt und Monte Carlo-Simulationen leicht zugänglich ist.

Das Problem ist nun,  $\epsilon$  mittels Analyse der Daten einer Zeitreihe zu bestimmen. Es läßt sich durch Anwendung des *binning*s lösen, welches auf folgender Idee basiert:

Die  $N$  Messungen der Zeitreihe werden zu  $N/b$  Blöcken zusammengefaßt, wobei  $b$  die Größe dieser Blöcke angibt. Man kann nun für jeden einzelnen der  $N/b$  Blöcke einen Mittelwert  $\tilde{\mathcal{O}}_i$  bilden. Auf diese Mittelwerte wird die Standard-Formel für die Fehlerrechnung

$$\Delta\mathcal{O}(b) = \sqrt{\frac{1}{\frac{N}{b} \left(\frac{N}{b} - 1\right)} \sum_{i=1}^{N/b} (\tilde{\mathcal{O}}_i - \bar{\mathcal{O}})^2} \quad , \quad \bar{\mathcal{O}} = \frac{b}{N} \sum_{i=1}^{N/b} \tilde{\mathcal{O}}_i \quad , \quad (3.19)$$

angewendet. Der Mittelwert über die einzelnen Blockmittelwerte,  $\bar{\mathcal{O}}$ , ist für Größen wie Energie oder Magnetisierung unabhängig von der Blockgröße. Die Mittelwerte der verschiedenen Blöcke,  $\tilde{\mathcal{O}}_i$ , sind für verschiedene  $i$  nicht miteinander korreliert, falls  $b \gg 2\tau_{\text{int}}^{\mathcal{O}}$  gilt. Der gemäß (3.19) berechnete Fehler wird dann den wahren Fehler wiedergeben. Im Fall  $b \leq 2\tau_{\text{int}}^{\mathcal{O}}$  gilt dies nicht, die einzelnen Blockmittelwerte sind korreliert und der berechnete Fehler wird den wirklichen Fehler unterschätzen.

Variiert man nun die Blockgröße  $b$ , so wird der Wert von  $\Delta\mathcal{O}(b)$  mit größerwerdendem  $b$  ausgehend von seinem Minimum bei  $\Delta\mathcal{O}(1)$ , dem naiven Fehler, ansteigen, um sich dann für  $b \gg 2\tau_{\text{int}}^{\mathcal{O}}$  asymptotisch einem Plateau-Wert zu nähern.

Dieser Plateau-Wert ergibt sich entsprechend der Annahme aus unkorrelierten Daten und kann deswegen mit dem wirklichen Fehler identifiziert werden:

$$\epsilon = \Delta\mathcal{O} \left( b \gg 2\tau_{\text{int}}^{\mathcal{O}} \right) \quad . \quad (3.20)$$

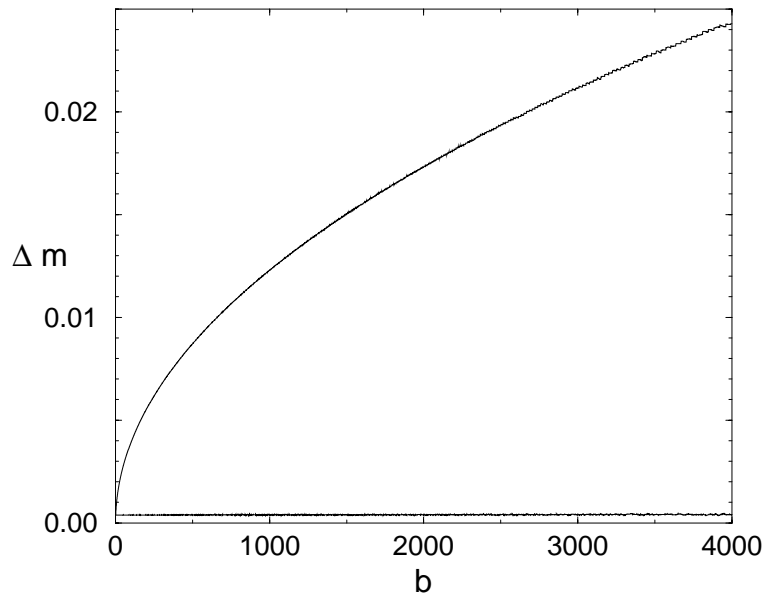
Die integrierte Autokorrelationszeit ergibt sich dann zu:

$$\tau_{\text{int}}^{\mathcal{O}} = \frac{1}{2} \left( \frac{\Delta\mathcal{O}(b \gg \tau_{\text{int}}^{\mathcal{O}})}{\Delta\mathcal{O}(b=1)} \right)^2 \quad . \quad (3.21)$$

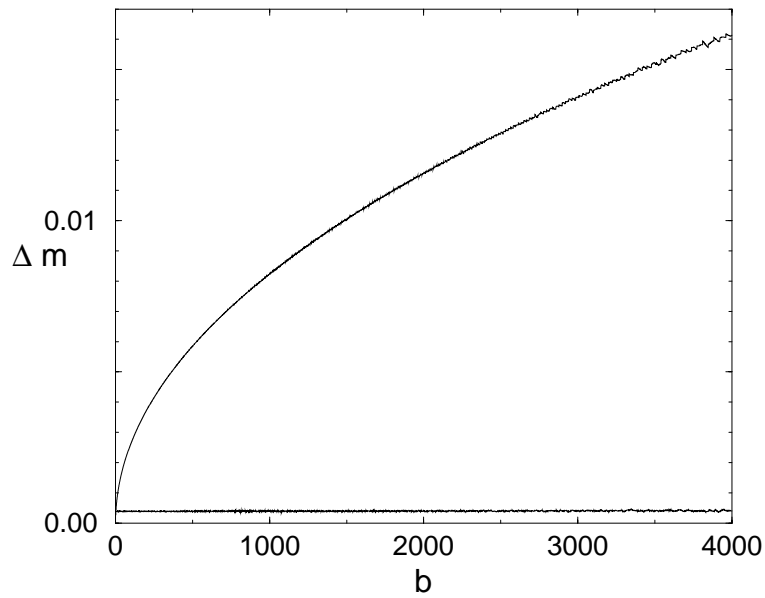
Aus Gründen der Selbstkonsistenz muß die Zeitreihe eine ausreichende Länge  $M \gg \tau_{\text{int}}^{\mathcal{O}}$  aufweisen, welche a priori nicht bekannt ist. Sie ist erreicht, wenn sich im  $\Delta\mathcal{O}(b)$ - $b$ -Diagramm ein deutlich sichtbares Plateau zeigt.

Wendet man die erläuterte Methodik auf die mit den beiden Algorithmen erzeugten Zeitreihen der Magnetisierung pro Spin  $m$  an, wird die Überlegenheit des Cluster-Algorithmus deutlich. In den Abb.3.1–3.4 sind die  $\Delta\mathcal{O}(b)$ - $b$ -Diagramme für verschiedene Schichtdicken und Oberflächenfelder am kritischen Punkt zusammengestellt.

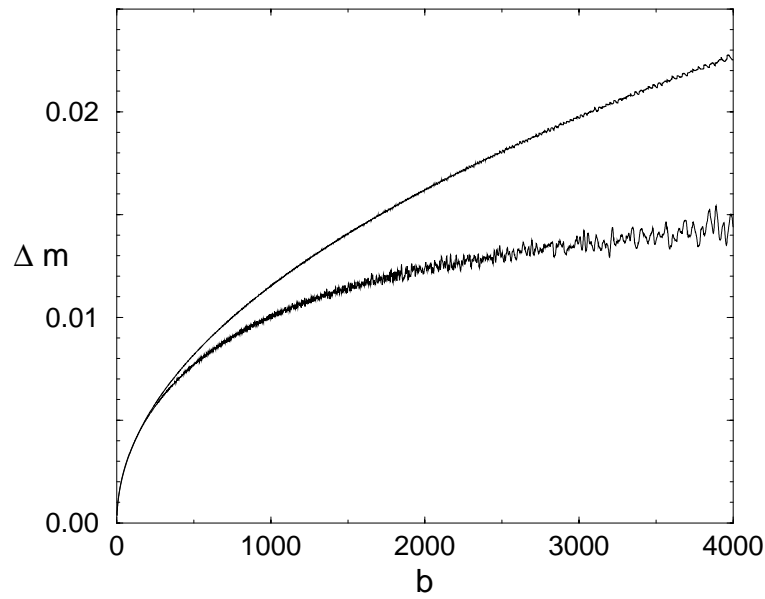
Im Fall der feldfreien Oberflächen wird sowohl für die Dicke  $D = 16$  (Abb. 3.1), als auch für  $D = 32$  (Abb. 3.2) im Falle des Clusters schon bei geringer Blockgröße  $b$  ein sehr flaches, deutlich ausgeprägtes Plateau erreicht. Damit läßt sich in beiden Fällen die Größenordnung der integrierten Autokorrelationszeit  $\tau_{\text{int}}^m$  mit 1 angeben. Weitaus ungünstiger stellt sich die Situation für den Metropolis-Algorithmus dar. Für beide Schichtdicken kann das Erreichen eines Plateaus nicht festgestellt werden.



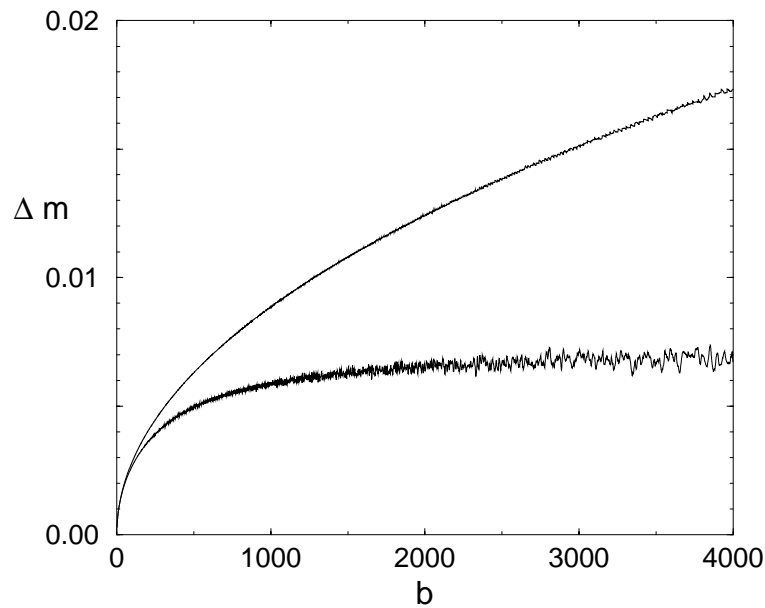
**Abb. 3.1:** Binning der Magnetisierung  $m$  bei freien Oberflächen für ein System der Größe  $D = 16$  und  $L = 128$  am kritischen Punkt. Die obere Kurve gilt für den Metropolis-, die untere für den Cluster-Algorithmus.



**Abb. 3.2:** Binning der Magnetisierung  $m$  bei freien Oberflächen für ein System der Größe  $D = 32$  und  $L = 128$  am kritischen Punkt. Die obere Kurve gilt für den Metropolis-, die untere für den Cluster-Algorithmus.



**Abb. 3.3:** Binning der Magnetisierung  $m$  bei Oberflächenfeld  $h_1 = -0.015$  eines Systems der Größe  $D = 16$  und  $L = 128$  am kritischen Punkt. Die obere Kurve gilt für den Metropolis-, die untere für den Cluster-Algorithmus.



**Abb. 3.4:** Binning der Magnetisierung  $m$  bei Oberflächenfeld  $h_1 = -0.015$  eines Systems der Größe  $D = 32$  und  $L = 128$  am kritischen Punkt. Die obere Kurve gilt für den Metropolis-, die untere für den Cluster-Algorithmus.

Für die Autokorrelationszeit kann somit nur eine untere Schranke von  $\tau_{\text{int}}^m > 2000$  angegeben werden.

Auch bei Systemen, die einem Randmagnetfeld ausgesetzt sind, zeigt sich ein deutlicher Vorsprung des nicht-lokalen Algorithmus (Abb. 3.3–3.4). Während auch in diesem Fall die Kurven des lokalen Algorithmus kein Plateau ausbilden, zeigen die zum Cluster-Algorithmus korrespondierenden Kurven ein deutlich günstigeres Verhalten. Aus ihren Plateauwerten läßt sich die Autokorrelationszeit zu  $\tau_{\text{int}}^m = 660$  für  $D = 16$  und  $\tau_{\text{int}}^m = 280$  für  $D = 32$  bestimmen. Mit ansteigender Schichtdicke nehmen die Korrelationen ab, was auf die gleichfalls mit steigender Schichtdicke abnehmende Stärke des Koexistenzmagnetfeldes zurückzuführen ist. Für den Metropolis-Algorithmus kann nur die untere Schranke von  $\tau_{\text{int}}^m > 1900$  ( $\tau_{\text{int}}^m > 1700$ ) für die kleinere (größere) der beiden Dicken angegeben werden. Ein Vergleich der beiden Simulationstechniken zeigt, daß mit dem nicht-lokalen Cluster-Algorithmus für eine Schichtdicke  $D = 16$  ein um einen Faktor 3 kleinere Autokorrelationszeit erzielt werden kann. Dieser Wert verbessert sich noch mit steigender Dicke. Es wurde für  $D = 32$  sogar eine Reduzierung der Korrelationen um den Faktor 6 erreicht. Somit bietet die Verwendung eines Cluster-Algorithmus einen deutlichen Vorteil bei der Untersuchung von Systemen in der Nähe ihres kritischen Punktes, der sich in einer verbesserten Statistik bei vergleichbarem Einsatz von Rechenzeit niederschlägt.

## 3.2 Reweighting

Im allgemeinen erhält man durch eine einzelne Monte Carlo-Simulation nur Ergebnisse für Observablen bei einer einzigen Parameter-Konstellation; man ist also gezwungen, für jede interessierende Kombination der Parameter einen separaten Simulationslauf durchzuführen. Dies kann mühsam werden, wenn man z.B. eine Untersuchung eines Modells bei der kritischen Temperatur durchführen will, den Wert dieses Parameters aber a priori nicht kennt. Falls der Parameterraum sogar mehrdimensional ist, ist die Suche nach einem kritischen Punkt ohne eine Methode zur Extrapolation fast aussichtslos. Im folgenden wird auf die Extrapolation mittels “Reweighting” eingegangen. Grundzüge dieser Methode wurden schon früh in der Literatur erwähnt [65, 66] und in jüngerer Zeit u.a. für Spinsysteme [67] und Polymermischungen [68] formuliert. Voraussetzung für die Anwendung dieser Extrapolationstechnik ist die Aufnahme ganzer Histogramme während des Simulationslaufes. Diese detailliertere Information über den Systemzustand kann zu einer Parameterkonstellation, die zur ursprünglichen eng benachbart im Parameterraum liegt, extrapoliert werden. Falls als Basis hierfür nur ein einzelnes Histogramm dient, spricht man auch von “Single-Histogram-Reweighting”; dienen mehrere als Basis, die für verschiedene Parameterpunkte aufgenommen wurden, spricht man von “Multi-Histogram-Reweighting”. Diese beiden Techniken werden in den beiden nächsten Unterkapiteln beschrieben.

### 3.2.1 Single-Histogram-Reweighting

Die prinzipielle Vorgehensweise beim Reweighting gliedert sich in zwei Schritte. Zuerst wird basierend auf der in einem Histogramm gespeicherten detaillierteren Information über das untersuchte System eine Schätzung für die mikrokanonische Zustandssumme durchgeführt. Aus dieser Zustandssumme können dann mit Hilfe der statistischen Mechanik die Werte für die interessierenden Observablen berechnet werden.

Das in dieser Arbeit untersuchte Modell wird durch den Hamilton-Operator (vgl. (2.5))

$$\mathcal{H}(\{S_i\}) = -J \sum_{\langle i,j \rangle} S_i S_j - h \sum_i S_i - h_1 \sum_{i \in \text{Oberflächen}} S_i \quad (3.22)$$

beschrieben. Führt man die Wechselwirkungsenergie  $W = J \sum_{\langle i,j \rangle} S_i S_j$ , die Gesamtmagnetisierung  $M$  und die Magnetisierung der beiden Oberflächen  $M_1$  ein, dann gilt für die Gesamtenergie

$$E(\{S_i\}) = -W(\{S_i\}) - h M(\{S_i\}) - h_1 M_1(\{S_i\}) . \quad (3.23)$$

Analog zur mikrokanonischen Zustandssumme  $\Omega(E)$ , die die Anzahl der Zustände mit Energie  $E$  beschreibt, wird ein statistisches Gewicht  $\Gamma(W, M, M_1)$  definiert:

$$\Gamma(W, M, M_1) = \sum_{\{S_i\}} \delta(W(\{S_i\}) - W) \delta(M(\{S_i\}) - M) \delta(M_1(\{S_i\}) - M_1) . \quad (3.24)$$

Dieses statistische Gewicht ist im Gegensatz zur mikrokanonischen Zustandssumme unabhängig von den beiden Feldern  $h$  und  $h_1$ . Dieser Vorteil wird jedoch dadurch erkauft, daß die Gesamtenergie nicht mehr als Ganzes das Argument des Gewichtes bildet, sondern ihre drei Teilbeiträge, repräsentiert durch die Wechselwirkungsenergie  $W$ , die Magnetisierung  $M$  und die Oberflächenmagnetisierung  $M_1$ . Wie noch erläutert werden wird, hat dies Auswirkungen auf die Komplexität der durch die Simulation zu gewinnenden Histogramme.

Der Übergang von der mikrokanonischen zur kanonischen Zustandssumme  $Z$  erfolgt durch eine Laplace-Transformation:

$$Z(\beta, h, h_1) = \int dW \int dM \int dM_1 e^{-\beta E_{h,h_1}(W,M,M_1)} \Gamma(W, M, M_1) , \quad (3.25)$$

wobei mit  $\beta$  die inverse Temperatur bezeichnet wird.

Zu diesem Ausdruck für die Zustandssumme korrespondiert die Verteilung  $P(W, M, M_1)$  mit

$$P_{\beta,h,h_1}(W, M, M_1) = \frac{e^{-\beta E_{h,h_1}(W,M,M_1)}}{Z(\beta, h, h_1)} \Gamma(W, M, M_1) . \quad (3.26)$$

Der Zusammenhang zwischen dem in der Monte Carlo-Simulation gemessenen Histogramm  $H(W, M, M_1)$  und dem der mikrokanonischen Zustandssumme entsprechenden statistischen Gewicht  $\Gamma(W, M, M_1)$  wird durch die Forderung hergestellt, daß im Grenzfall einer unendlichen Anzahl von unabhängigen Messungen  $N$  das normierte Histogramm die Verteilung  $P(W, M, M_1)$  approximieren soll:

$$P_{\beta,h,h_1}(W, M, M_1) \approx \frac{H_{\beta,h,h_1}(W, M, M_1)}{N}. \quad (3.27)$$

Somit gilt mit (3.26), daß

$$\Gamma(W, M, M_1) \approx \frac{Z(\beta, h, h_1)}{N} e^{\beta E_{h,h_1}(W, M, M_1)} H_{\beta,h,h_1}(W, M, M_1). \quad (3.28)$$

Für die Zustandssumme bei den Parameterwerten  $\beta'$ ,  $h'$  und  $h'_1$ , zu denen extrapoliert werden soll, gilt nach (3.25) und der Approximierung (3.28) ( $E' = E_{h',h'_1}(W, M, M_1)$  und  $E = E_{h,h_1}(W, M, M_1)$ ):

$$\begin{aligned} Z(\beta', h', h'_1) &= \int dW \int dM \int dM_1 e^{-\beta' E'} \Gamma(W, M, M_1) \\ &\approx \frac{Z(\beta, h, h_1)}{N} \int dW \int dM \int dM_1 e^{-(\beta' E' - \beta E)} H_{\beta,h,h_1}(W, M, M_1). \end{aligned} \quad (3.29)$$

Für die Wahrscheinlichkeitsverteilung  $P'(W, M, M_1) = P_{\beta',h',h'_1}(W, M, M_1)$  bei den neuen Werten gilt nun nach (3.26) unter Verwendung von (3.28) und (3.29):

$$\begin{aligned} P'(W, M, M_1) &= \frac{e^{-\beta' E'}}{Z(\beta', h', h'_1)} \Gamma(W, M, M_1) \\ &\approx \frac{Z(\beta, h, h_1)}{N Z(\beta', h', h'_1)} e^{-(\beta' E' - \beta E)} H_{\beta,h,h_1}(W, M, M_1) \\ &\approx \frac{e^{-(\beta' E' - \beta E)} H_{\beta,h,h_1}(W, M, M_1)}{\int dW \int dM \int dM_1 e^{-(\beta' E' - \beta E)} H_{\beta,h,h_1}(W, M, M_1)} \\ &= \frac{e^{(\beta' - \beta)W + (\beta' h' - \beta h)M + (\beta' h'_1 - \beta h_1)M_1} H_{\beta,h,h_1}(W, M, M_1)}{\int dW \int dM \int dM_1 e^{(\beta' - \beta)W + (\beta' h' - \beta h)M + (\beta' h'_1 - \beta h_1)M_1} H_{\beta,h,h_1}(W, M, M_1)}, \end{aligned} \quad (3.30)$$

wobei im letzten Schritt

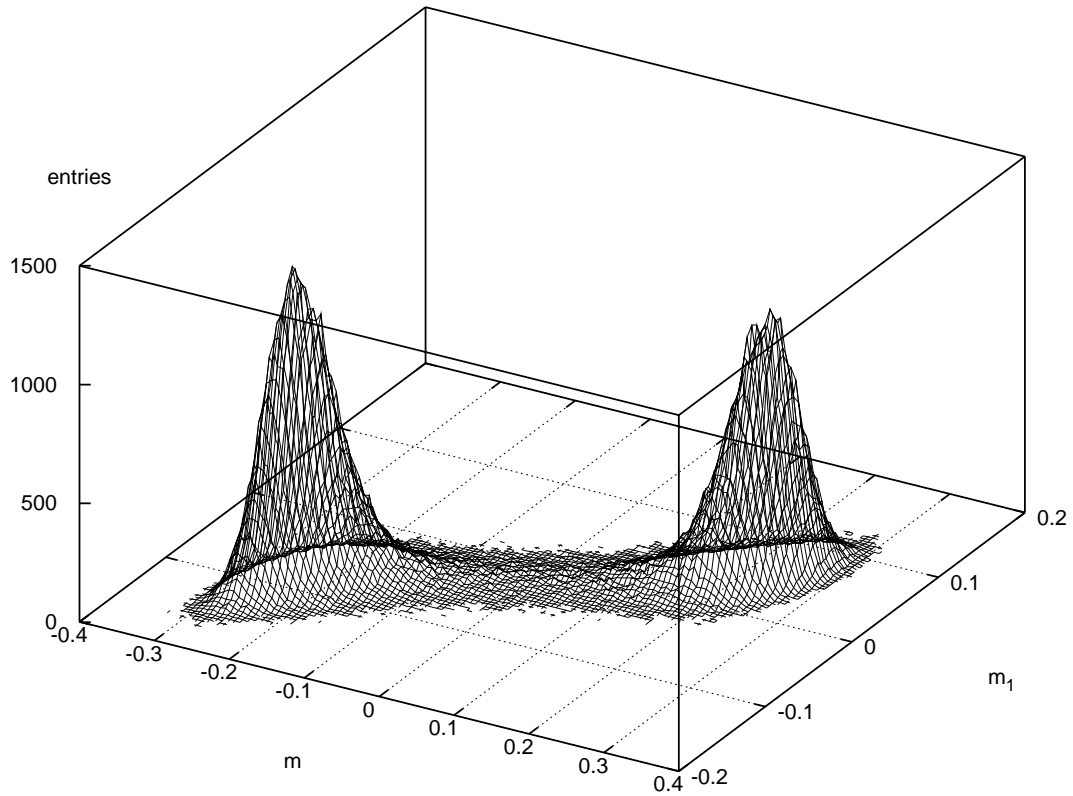
$$\beta' E' - \beta E = -(\beta' - \beta)W - (\beta' h' - \beta h)M - (\beta' h'_1 - \beta h_1)M_1 \quad (3.31)$$

verwendet wurde. In die rechte Seite von (3.31) gehen nur noch die Histogramme und die Parameterwerte ein. Die kanonische Zustandssumme, die nur bis auf einen konstanten Faktor definiert und einer Simulation nicht direkt zugänglich ist, kürzt sich heraus.

Der Erwartungswert einer Observablen, die eine Funktion von Wechselwirkungsenergie, Magnetisierung und Oberflächenmagnetisierung ist, lässt sich bei den neuen Parameterwerten zu

$$\langle f(W, M, M_1) \rangle_{\beta', h', h'_1} = \int dW \int dM \int dM_1 f(W, M, M_1) P_{\beta', h', h'_1}(W, M, M_1) \quad (3.32)$$

extrapolieren.



**Abb. 3.5:** Histogramm bezüglich der Magnetisierung pro Spin  $m$  und der Oberflächenmagnetisierung pro Spin  $m_1$  für ein System mit  $D = 32$ ,  $L = 96$  und Oberflächenfeld  $h_1 = -0.015$  in der Nähe der Phasenkoexistenzlinie ( $h = 0.000280$ ) und in der Nähe der kritischen Temperatur  $T_c(D)$  ( $T = 4.471$ ).

Da die Güte eines jeden gemessenen Histogramms aufgrund der endlichen Anzahl von Einträgen von statistischen Fehlern beeinträchtigt ist, ist es nicht möglich, durch

Reweighting verläßliche Informationen über einen beliebig großen Parameterbereich zu gewinnen. Der Grund dafür ist, daß der Exponentialfaktor in (3.31) die Werte in den Ausläufern des Histogramms besonders stark gewichtet, obwohl dort nur noch wenige und somit mit besonders großen relativen Fehlern behaftete Einträge vorhanden sind. Dies schränkt die Reichweite einer verläßlichen Extrapolation sehr stark ein. Es bietet sich deshalb an, Histogramme in der Nähe des kritischen Punktes zu erzeugen, da dort die Suszeptibilität und die spezifische Wärme divergieren und somit das Histogramm stark verbreitert ist, was der Reichweite zugute kommt.

Ein Problem bei der praktischen Durchführung des Reweichtings ist der Speicheraufwand für das Histogramm. Da in dem hier untersuchten Modell nicht nur die Energie- und die Magnetisierungswerte in dem Histogramm gespeichert werden müssen, sondern auch die Oberflächenmagnetisierung von Bedeutung ist, wäre ein Datenfeld, welches einen Eintrag für jede mögliche Kombination der drei Werte anlegt, sogar 3-dimensional. Dieses Feld hätte schon für das kleinste in dieser Arbeit untersuchte System mit Schichtdicke  $D = 4$  und lateraler Ausdehnung  $L = 16$  ( $z$  bezeichnet die Koordinationszahl des Gitters)

$$(DL^2 + 1) \times (zD/2 - 1)L^2 \times (2L^2 + 1) \approx 2.7 \times 10^8 \quad (3.33)$$

Einträge, die allerdings fast alle den Wert Null hätten. Eine grobere Rasterung der Meßwerte in *bins* auf der anderen Seite könnte jedoch zu numerischen Ungenauigkeiten führen. Weil die Anzahl der Messungen während eines Simulationslaufes jedoch um einige Größenordnungen niedriger liegt als die Anzahl der möglichen Kombinationen, bietet es sich nun an, für jede Messung ein Feldelement anzulegen, welches das Tripel der Meßwerte aufnimmt. Mit dieser Vorgehensweise ergibt sich bei  $10^6$  Messungen und einer Integergröße von 4 Byte ein Speicherbedarf von ca. 12 MByte, unabhängig von der Systemgröße.

### 3.2.2 Multi-Histogramm-Reweighting

Die Genauigkeit der im vorigen Kapitel beschriebenen Extrapolationsmethode läßt sich steigern, wenn man der Abschätzung der mikrokanonischen Zustandssumme gemäß (3.28) *mehrere* Histogramme zugrunde legt, die durch Simulationsläufe bei variierenden Parameterwerten gewonnen wurden. Unter der Annahme, daß  $n$  verschiedene Histogramme bei Parameterwerten  $(\beta^k, h^k, h_1^k)$  mit  $k = 1, \dots, n$  aufgenommen wurden, verallgemeinert sich die Schätzung für die Zustandssumme (3.28) zu

$$\Gamma(W, M, M_1) \approx \sum_{k=1}^n \omega_k(W, M, M_1) \frac{Z(\beta^k, h^k, h_1^k)}{N_k} e^{\beta^k E^k} H_{\beta^k, h^k, h_1^k}(W, M, M_1), \quad (3.34)$$

wobei  $E^k = -JW - h^k M - h_1^k M_1$  und  $N_k$  die Anzahl der Einträge im Histogramm  $k$  ist. Mit  $\omega_k(W, M, M_1)$  wurde ein allgemeiner Gewichtungsfaktor eingeführt, da es unsinnig wäre, alle Histogramme mit gleichem Gewicht in die Schätzung eingehen zu lassen. So muß z.B. ein Histogramm, welches bei den Argumentwerten  $(W, M, M_1)$  von  $\Gamma$  viele Einträge besitzt, stärker gewichtet werden als eines, bei dem diese Argumentwerte in den Ausläufern liegen und somit die Anzahl der Einträge starken statistischen Fluktuationen unterliegt. Für den statistischen Fehler, mit dem die Histogramm-Einträge behaftet sind, gilt

$$\Delta H_{\beta, h, h_1}(W, M, M_1) = \sqrt{H_{\beta, h, h_1}(W, M, M_1)}. \quad (3.35)$$

Für die Bestimmung der optimalen Gewichte führt man eine Variationsrechnung durch [69], bei der der aus  $\Delta H$  resultierende Fehler  $\Delta \Gamma$  der Abschätzung (3.34) als Funktion der  $\omega_k$  minimiert wird. Daraus ergibt sich folgender Ausdruck für die Gewichte [69]:

$$\omega_k(W, M, M_1) = \frac{N_k e^{-\beta^k E^k}}{Z(\beta^k, h^k, h_1^k)} \left( \sum_{j=1}^n \frac{N_j e^{-\beta^j E^j}}{Z(\beta^j, h^j, h_1^j)} \right)^{-1}. \quad (3.36)$$

Führt man die Abkürzungen

$$X(W, M, M_1) := \sum_{j=1}^n H_{\beta^j, h^j, h_1^j}(W, M, M_1), \quad (3.37)$$

$$\begin{aligned}
Y(W, M, M_1) &:= \sum_{j=1}^n \frac{N_j e^{-\beta^j E^j}}{Z(\beta^j, h^j, h_1^j)} \\
&= \sum_{j=1}^n \frac{N_j e^{\beta^j W + \beta^j h^j M + \beta^j h_1^j M_1}}{Z(\beta^j, h^j, h_1^j)}
\end{aligned} \tag{3.38}$$

ein, läßt sich (3.34) unter Verwendung von (3.36) schreiben als

$$\Gamma(W, M, M_1) \approx \frac{X(W, M, M_1)}{Y(W, M, M_1)}. \tag{3.39}$$

Mit dieser Schätzung  $\Gamma(W, M, M_1)$  kann man wie im Falle des Single-Histogram-Reweighting alle interessierenden Größen der statistischen Mechanik ableiten (vgl. Kap. 3.2.1).

Insbesondere ergibt sich für die Zustandssumme

$$\begin{aligned}
Z(\beta', h', h_1') &= \int dW \int dM \int dM_1 e^{-\beta' E'} \Gamma(W, M, M_1) \\
&\approx \int dW \int dM \int dM_1 e^{-\beta' E'} \frac{X(W, M, M_1)}{Y(W, M, M_1)}
\end{aligned} \tag{3.40}$$

und für die Verteilung

$$\begin{aligned}
P_{\beta', h', h_1'}(W, M, M_1) &= \frac{e^{-\beta' E'}}{Z(\beta', h', h_1')} \Gamma(W, M, M_1) \\
&\approx \frac{e^{-\beta' E'}}{Z(\beta', h', h_1')} \frac{X(W, M, M_1)}{Y(W, M, M_1)} \\
&\approx \frac{e^{\beta' W + \beta' h' M + \beta' h_1' M_1} \frac{X(W, M, M_1)}{Y(W, M, M_1)}}{\int d\tilde{W} \int d\tilde{M} \int d\tilde{M}_1 e^{\beta' \tilde{W} + \beta' h' \tilde{M} + \beta' h_1' \tilde{M}_1} \frac{X(\tilde{W}, \tilde{M}, \tilde{M}_1)}{Y(\tilde{W}, \tilde{M}, \tilde{M}_1)}}, \tag{3.41}
\end{aligned}$$

aus der sich wiederum nach (3.32) die Erwartungswerte der einzelnen Observablen berechnen lassen.

Im Gegensatz zum Single-Histogram-Reweighting treten in dem Ausdruck für die Verteilung (3.41) auch die  $n$  Zustandssummen  $Z(\beta^j, h^j, h_1^j)$  auf, deren Werte der Simulation nicht direkt zugänglich sind. Sie gehen über  $Y(W, M, M_1)$  in die Verteilung ein und kürzen sich nicht wie im Falle einzelner Histogramme heraus (vgl. (3.31)). Es ist deshalb notwendig, zu einer Abschätzung der  $n$  Zustandssummen zu

gelangen. Eine wichtige Forderung an diese Abschätzung ist die Selbstkonsistenz, indem sie den Ausdruck (3.40), der in ausführlicher Form

$$Z_k = \sum_{j=1}^n \int dW \int dM \int dM_1 H_{\beta^j, h^j, h_1^j}(W, M, M_1) e^{\beta^k W + \beta^k h^k M + \beta^k h_1^k M_1} \times \left( \sum_{i=1}^n \frac{N_i e^{\beta^i W + \beta^i h^i M + \beta^i h_1^i M_1}}{Z_i} \right)^{-1}, \quad \forall k = 1, \dots, n \quad (3.42)$$

lautet, für jedes einzelne  $Z_k \equiv Z(\beta^k, h^k, h_1^k)$  erfüllt. Diese Selbstkonsistenzforderung läßt sich nutzen, um mit Hilfe eines Iterationsverfahrens zu einem Satz selbstkonsistenter Zustandssummen zu gelangen: Man beginnt mit einem Vektor von Startwerten  $\vec{Z}^0 = (Z_1^0, \dots, Z_n^0)$  für die  $n$  gesuchten Zustandssummen, setzt diese auf der rechten Seite von (3.42) ein und erhält in diesem ersten Iterationsschritt  $n$  neue Werte  $\vec{Z}^1 = (Z_1^1, \dots, Z_n^1)$ , die wiederum im Zuge der nächsten Iteration auf der rechten Seite in der Gleichung eingesetzt werden können, usw. Konvergiert dieses Verfahren, kann es beim Erreichen der angestrebten relativen Genauigkeit  $\epsilon$  abgebrochen werden, wenn nach  $t$  Iterationen gilt:

$$\left| \frac{Z_k^t}{Z_k^{t-1}} - 1 \right| < \epsilon, \quad \forall k = 1, \dots, n. \quad (3.43)$$

Das Konvergenzverhalten der  $\vec{Z}^t$  hängt von der Wahl des Startvektors  $\vec{Z}^0$  ab. In der hier vorliegenden Arbeit wurde mit der Wahl  $\vec{Z}^0 = (N_1, \dots, N_n)$  eine relative Präzision von  $\epsilon = 10^{-8}$  schon nach 10 – 50 Iterationen erreicht. Diese Schwankung rührt vor allem von der Dimension des Zustandssummenvektors her: Der kleinere Wert repräsentiert den Fall eines nur 3-dimensionalen Vektors, d.h. die Basis der Extrapolation wird von drei Histogrammen gebildet, während der größere einer Extrapolation aus 12 Histogrammen entstammt.

### 3.3 Bestimmung der kritischen Punkte

Ein wichtiges Ziel dieser Arbeit und Grundlage für die weitere Analyse der Daten ist die Bestimmung der einzelnen kritischen Temperaturen  $T_c(D)$  für die verschiedenen Schichtdicken  $D$ . Sind diese Temperaturen  $T_c(D)$  bekannt, kann das entsprechende kritische Bulk-Feld  $h_c(D)$  als letzte noch fehlende Koordinate des kritischen Punktes an der Phasenkoexistenzlinie abgelesen werden.

Zur Bestimmung der kritischen Temperatur gibt es eine große Anzahl von verschiedenen Techniken [39–42, 70]. In den beiden folgenden Unterkapiteln sollen die beiden Techniken beschrieben werden, die in dieser Arbeit zur Lösung dieses Problems herangezogen wurden:

Die Extrapolation aus den Temperaturwerten der Maxima der spezifischen Wärme bzw. der magnetischen Suszeptibilität und die Extrapolation aus den Kumulanten-schnittpunkten.

### 3.3.1 Spezifische Wärme und magnetische Suszeptibilität

An einem kritischen Punkt ist die Zustandssumme eines Systems im thermodynamischen Limes nicht mehr analytisch. Dies hat zur Konsequenz, daß die aus der freien Energie durch Differentiation gewonnenen Observablen ein singuläres Verhalten zeigen. Ein solches Verhalten kann mit Hilfe von sogenannten *kritischen Exponenten* beschrieben werden, welche modellabhängig sind. Mit Hilfe der Universalitätshypothese [71] lassen sich jedoch die Modelle in einzelne Klassen unterteilen, in denen jeweils die gleichen kritischen Exponenten gültig sind. Kriterien für diese Klasseneinteilung sind die Dimension des Raumes, die Dimension des Ordnungsparameters und die Art der Randbedingungen.

Für die Magnetisierung pro Spin  $m$ , die magnetische Suszeptibilität  $\chi$ , die spezifische Wärme  $C_V$  und die Korrelationslänge  $\xi$  läßt sich das singuläre Verhalten durch folgende, nur in der Nähe von  $T_c$  gültigen Beziehungen beschreiben:

$$m(T) \propto \left|1 - \frac{T}{T_c}\right|^\beta, \quad T < T_c, \quad (3.44)$$

$$\chi(T) \propto \left|1 - \frac{T}{T_c}\right|^{-\gamma}, \quad (3.45)$$

$$C_V(T) \propto \left|1 - \frac{T}{T_c}\right|^{-\alpha}, \quad (3.46)$$

$$\xi(T) \propto \left|1 - \frac{T}{T_c}\right|^{-\nu}. \quad (3.47)$$

In der Nähe des kritischen Punktes  $T = T_c$  spiegeln diese Relationen, positive Werte für die kritischen Exponenten vorausgesetzt, das erwartete Verhalten der Observablen wider: Die Suszeptibilität, die spezifische Wärme und die Korrelationslänge divergieren am kritischen Punkt, der Ordnungsparameter  $m$  verschwindet. Es gibt

auch Modelle, für die  $\alpha = 0$  gilt, wie z.B. das 2-dimensionale Ising-Modell und das Mean-Field Ising-Modell. Im Falle des 2d-Ising-Modells spricht man von einer logarithmischen Divergenz und beim Mean-Field-Modell handelt es sich um eine Sprung-Singularität.

Bei endlicher Systemgröße bleibt die Zustandssumme auch am kritischen Punkt analytisch. Somit verschwindet auch das singuläre Verhalten der Observablen: Die im thermodynamischen Limes divergierende spezifische Wärme und die Suszeptibilität nehmen Maxima an, deren Orte jedoch gegenüber der kritischen Temperatur verschoben sind, der Anstieg der Korrelationslänge  $\xi$  in der Nähe von  $T_c$  wird auf die endliche Systemlänge  $L$  begrenzt und es gilt für den Ordnungsparameter  $m(T_c, L) \neq 0$ .

Die Abhängigkeit dieser Phänomene von der Länge  $L$  wird in der Nähe von  $T_c$  durch die *Finite-Size-Scaling-Theorie* [70, 72–74] beschrieben. Diese Theorie basiert auf der Hypothese, daß es nur eine einzige relevante Längenskala im System gibt, welche durch die Korrelationslänge vorgegeben wird. Somit gehen alle Längen nur im Verhältnis zu  $\xi$  ein, und alle Größen lassen sich in geeigneten Potenzen dieser Skala messen. So läßt sich mit einem Finite-Size-Scaling-Ansatz die Suszeptibilität als

$$\chi(T, L) = L^\kappa \tilde{f}_1(L/\xi(T)) \quad (3.48)$$

darstellen, wobei  $\kappa$  ein noch näher zu bestimmender Exponent ist. Mit (3.47) und Einführung von  $t = |1 - T/T_c|$  sowie der Scaling-Variablen  $x = t L^{1/\nu}$  erhält man für das Argument der Funktion  $\tilde{f}_1$

$$L/\xi(T) = L/t^{-\nu} = (t L^{1/\nu})^\nu = x^\nu, \quad (3.49)$$

und (3.48) geht über in

$$\chi(T, L) = L^\kappa \tilde{f}_1(x^\nu) = L^\kappa f_1(x). \quad (3.50)$$

Im Limes  $L \rightarrow \infty$  muß die Suszeptibilität in der Nähe von  $T_c$  wieder das aus (3.45) bekannte Verhalten zeigen. Möglich ist dies jedoch nur, wenn für die Skalen-Funktion  $f_1(x)$

$$f_1(x) \rightarrow x^{-\gamma} \text{ für } L \rightarrow \infty \quad (3.51)$$

gilt. Damit tritt zwangsläufig auch ein Term  $L^{-\gamma/\nu}$  auf, welcher, um mit (3.45) konsistent zu sein, durch die Wahl von  $\kappa = \gamma/\nu$  kompensiert werden muß. Somit ergibt sich für die magnetische Suszeptibilität bei endlichen Systemgrößen

$$\chi(T, L) = L^{\gamma/\nu} f_1(t L^{1/\nu}) , \quad (3.52)$$

und analog dazu für die spezifische Wärme

$$C_V(T, L) = L^{\alpha/\nu} f_2(t L^{1/\nu}) . \quad (3.53)$$

Diese beiden Aussagen sind natürlich nur asymptotisch für große  $L$  gültig (vgl. (3.51)).

Aus (3.52) und (3.53) läßt sich nun die Beziehung herleiten, die beschreibt, wie die Temperaturen  $T_{\max}$ , bei denen  $\chi$  bzw.  $C_V$  für verschiedene  $L$  maximal ist, gegenüber der kritischen Temperatur  $T_c$  verschoben sind. Sei nun  $\chi(T, L)$  bzw.  $C_V(T, L)$  für ein beliebiges  $x_{\max}$  maximal. Dann gilt

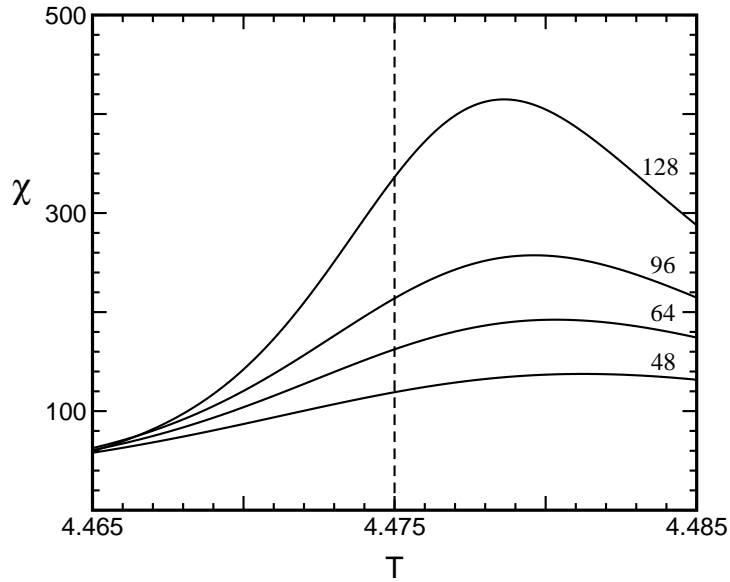
$$\begin{aligned} x_{\max} &= |1 - T_{\max}/T_c| L^{1/\nu} \\ \Rightarrow T_{\max} &= T_c - T_c \tilde{x}_{\max} L^{-1/\nu} , \end{aligned}$$

und nach Definition von  $a = -T_c \tilde{x}_{\max}$

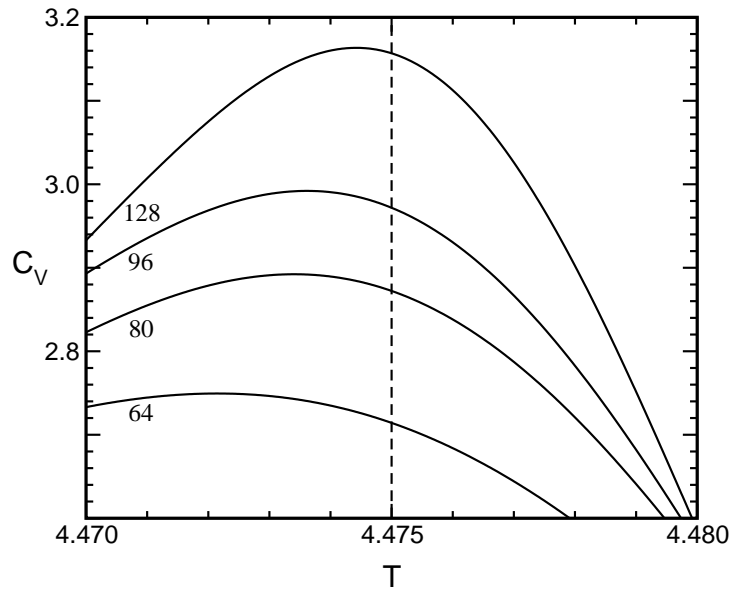
$$T_{\max} = T_c + a L^{-1/\nu} . \quad (3.54)$$

In den Abb. (3.6) und (3.7) sind exemplarisch Kurven für die Suszeptibilität und die spezifische Wärme in der Nähe der kritischen Temperatur bei fester Schichtdicke  $D$  für verschiedene laterale Längen  $L$  dargestellt. In beiden Bildern ist deutlich die mit steigendem  $L$  abnehmende Verschiebung der Orte der Maxima gegenüber dem Wert der kritischen Temperatur erkennbar.

Ist der Exponent  $\nu$  für das zu untersuchende Modell bekannt, kann mittels einer Auftragung von  $T_{\max}$  gegen  $L^{-1/\nu}$  und Anpassen einer Geraden an die Daten die kritische Temperatur bestimmt werden. Ihr Wert kann als Ordinatenabschnitt abgelesen werden. Wie bereits angesprochen, gilt (3.54) nur asymptotisch für genügend große  $L$ . Ist dies nicht erfüllt, treten Korrekturterme auf, von denen später noch die Rede sein wird.



**Abb. 3.6:** Magnetische Suszeptibilität  $\chi$  in der Nähe von  $T_c$  für ein System der Schichtdicke  $D = 32$  mit feldfreien Oberflächen. In die Abbildung sind die Werte von  $L$  aufgenommen. Die senkrechte durchbrochene Linie markiert den Wert der kritischen Temperatur  $T_c(D = 32)$  (vgl. Tab. 6.1).



**Abb. 3.7:** Spezifische Wärme  $C_V$  in der Nähe von  $T_c$  für ein System der Schichtdicke  $D = 32$  mit feldfreien Oberflächen. In die Abbildung sind die Werte von  $L$  aufgenommen. Die senkrechte durchbrochene Linie markiert den Wert der kritischen Temperatur  $T_c(D = 32)$  (vgl. Tab. 6.1).

### 3.3.2 Kumulantenschnittpunkte

Aus der im Kap. 3.3.1 angesprochenen Finite-Size-Scaling-Hypothese lassen sich nicht nur Aussagen über das Verhalten von *Observablen* bezüglich einer endlichen Systemgröße, sondern auch über ganze Ordnungsparameterverteilungen ableiten.

Betrachtet man beispielsweise die Verteilung der Magnetisierung

$$P_L(m) = P(m, \xi, L) \quad (3.55)$$

für einen Bereich nahe  $T_c$ , der die Bedingung erfüllt, daß für die Korrelationslänge  $\xi(T) \approx L$  gilt, dann hängt  $P(m)$  nicht mehr separat von den drei Variablen  $m$ ,  $\xi$  und  $L$  ab, sondern entsprechend der Skalenhypothese nur von den beiden skalierten Kombinationen  $L/\xi$  und  $m/\xi^{-\beta/\nu}$ . Die letzte der beiden Kombinationen ergibt sich aus der Wahl von  $\xi^{-\beta/\nu}$  als “natürliche” Einheit der Magnetisierung, welche durch (vgl. (3.44) und (3.47))

$$m \propto \left(1 - \frac{T}{T_c}\right)^\beta \propto \xi^{-\beta/\nu} \quad (3.56)$$

motiviert wird. Um die Normierung

$$\int_{-\infty}^{\infty} dm P_L(m) = 1 \quad (3.57)$$

zu gewährleisten, muß die Verteilungsfunktion mit einem Vorfaktor  $\xi^{\beta/\nu}$  versehen werden, so daß gilt:

$$P_L(m) = \xi^{\beta/\nu} \tilde{P}(L/\xi, m/\xi^{-\beta/\nu}) = L^{\beta/\nu} \bar{P}(L/\xi, x), \quad (3.58)$$

wobei im letzten Schritt eine Variablentransformation auf die Scaling-Variable  $x$  durch

$$x \equiv m L^{\beta/\nu} = (m/\xi^{-\beta/\nu}) (L/\xi)^{\beta/\nu} \quad (3.59)$$

vorgenommen wurde, ebenfalls unter Erhaltung der Normierung (3.57). Mit Hilfe der Verteilung (3.58) lassen sich nun die Erwartungswerte der einzelnen Momente der Magnetisierung berechnen,

$$\begin{aligned}
\langle |m|^k \rangle_L &= \int_{-\infty}^{\infty} dm |m|^k P_L(m) \\
&= \int_{-\infty}^{\infty} dm |m|^k L^{\beta/\nu} \bar{P}(L/\xi, x) \\
&= L^{-k\beta/\nu} \int_{-\infty}^{\infty} dx |x|^k \bar{P}(L/\xi, x) \\
&= L^{-k\beta/\nu} \tilde{M}^{(k)}(L/\xi) \\
&= L^{-k\beta/\nu} M^{(k)}(tL^{1/\nu}), \tag{3.60}
\end{aligned}$$

unter Verwendung von (3.49) im letzten Schritt. Die Struktur von (3.60) macht deutlich, daß die nach dem Schema

$$f_{ij}^{kl}(tL^{1/\nu}) = \frac{(\langle |m|^k \rangle_L)^l}{(\langle |m|^i \rangle_L)^j} = \frac{(M^{(k)}(tL^{1/\nu}))^l}{(M^{(i)}(tL^{1/\nu}))^j} \tag{3.61}$$

aus den Verhältnissen der Momente gebildete Funktion  $f_{ij}^{kl}$  für die Wahl  $ij = kl$  keine *explizite* Größenabhängigkeit  $L$  mehr zeigt, da sich die Vorfaktoren kürzen. Die Abhängigkeit von  $L$  ist nur noch *implizit* über die Argumente der Skalenfunktionen  $M^{(k)}$  gegeben. Diese Tatsache verleiht der Funktion die interessante Eigenschaft, daß sie am kritischen Punkt, wo das Argument der Skalenfunktionen verschwindet, ihre Systemgrößenabhängigkeit einbüßt. Dies läßt sich in Simulationen ausnutzen, um die kritische Temperatur durch den gemeinsamen Schnittpunkt, den die Graphen von  $f_{ij}^{kl}(T)$  für verschiedene  $L$  haben, zu bestimmen. Der Vorteil dieses Verfahrens ist, daß die Bestimmung der kritischen Temperatur unabhängig von der Kenntnis der kritischen Exponenten möglich ist. Besonders geeignet dazu sind Funktionen  $f_{ij}^{kl}$  mit niedrigen Werten für die Indizes  $k$  und  $i$ , da die bei dieser Wahl involvierten Momente von niedriger Ordnung sich im allgemeinen durch eine bessere Statistik auszeichnen.

In dieser Arbeit wurde die Kumulante 4. Ordnung [75]

$$U_L = 1 - \frac{1}{3} f_{22}^{41} = 1 - \frac{\langle m^4 \rangle_L}{3 (\langle m^2 \rangle_L)^2} \tag{3.62}$$

verwendet. Durch die Wahl dieser speziellen Normierung verschwindet  $U_L$  im Limes  $T \rightarrow \infty$  für Modelle mit eindimensionalem Ordnungsparameter wie z.B. das Ising-

Modell, wie im Folgenden gezeigt werden soll:

Im Hochtemperatur-Limes sind die einzelnen Spins voneinander unabhängige Zufallsvariablen, so daß die Magnetisierung nach dem Zentralen Grenzwertsatz [76] für  $L \rightarrow \infty$  Gaußverteilt ist. Für die Momente einer Gaußverteilten skalaren Variable  $x$  gilt

$$\langle x^m \rangle = \frac{\int_{-\infty}^{\infty} dx x^m e^{-\alpha x^2}}{\int_{-\infty}^{\infty} dx e^{-\alpha x^2}} . \quad (3.63)$$

Unter Verwendung von

$$\int_{-\infty}^{\infty} dx x^{2m} e^{-\alpha x^2} = \left( -\frac{\partial}{\partial \alpha} \right)^k \sqrt{\frac{\pi}{\alpha}} , \quad k \in N^0 , \quad (3.64)$$

läßt sich für das Verhältnis  $\langle x^4 \rangle / \langle x^2 \rangle^2$

$$\frac{\langle x^4 \rangle}{\langle x^2 \rangle^2} = \frac{\frac{\partial^2}{\partial \alpha^2} \sqrt{\frac{\pi}{\alpha}}}{\left( \frac{\partial}{\partial \alpha} \sqrt{\frac{\pi}{\alpha}} \right)^2} = 3 \quad (3.65)$$

herleiten. Dies entspricht genau der Normierungskonstanten, die in (3.62) auftritt.

Im Grenzfall  $T \rightarrow 0$  verschwinden die Fluktuationen der Magnetisierung, die Verteilungsfunktion zeigt einen scharfen Peak bei  $m_0$  und für die Momente gilt  $\langle m^k \rangle \approx m_0^k$ . Die Kumulante nähert sich somit einem Wert von  $U_L = 2/3$ .

Für  $T = T_c$  nimmt die Kumulante den schwach universellen Wert

$$U^* \equiv U_L(T = T_c) \quad (3.66)$$

an, der jedoch von der Art der Randbedingungen des Modells abhängt [77]. Für das Ising-Modell auf dem Quadratgitter mit periodischen Randbedingungen finden sich mehrere Angaben zu  $U^*$  in der Literatur [77–80]. Die genaueste Schätzung findet sich bei Kamieniarz und Blöte mit  $U^* = 0.610\,690(1)$  [80]. Diese beiden Autoren bestimmen auch  $U^*$  für einige Seitenlängenverhältnisse beim allgemeineren Fall des Rechteckgitters.

Da die Kumulante für endliche  $L$  eine analytische Funktion ist, läßt sie sich in der Nähe von  $T_c$  in der Scaling-Variablen  $x = (1 - T/T_c) L^{1/\nu}$  entwickeln:

$$\begin{aligned}
U_L(x) &= U_L(0) + \left. \frac{\partial U_L}{\partial x} \right|_{x=0} x + \dots \\
&= U^* + U'(0) \left( 1 - \frac{T}{T_c} \right) L^{1/\nu} + \dots .
\end{aligned} \tag{3.67}$$

Daraus läßt sich für das Verhältnis der Steigungen der Kumulanten bei der kritischen Temperatur  $T_c$  und linearer Ausdehnung  $L$  und  $L' = bL$  folgende Beziehung ableiten:

$$\left. \frac{\partial U_{bL}}{\partial U_L} \right|_{U^*} = b^{1/\nu} . \tag{3.68}$$

Gemäß dieser Relation läßt sich durch Auftragung des Logarithmus des Längenverhältnisses,  $\ln(b)$ , gegen den Logarithmus des Verhältnisses der Steigungen der Kumulanten,  $\ln(\partial U_{bL}/\partial U_L)$ , der Wert von  $1/\nu$  aus den Simulationsdaten gewinnen: Der inverse Wert dieses Exponenten ist die Steigung der Ausgleichsgerade.

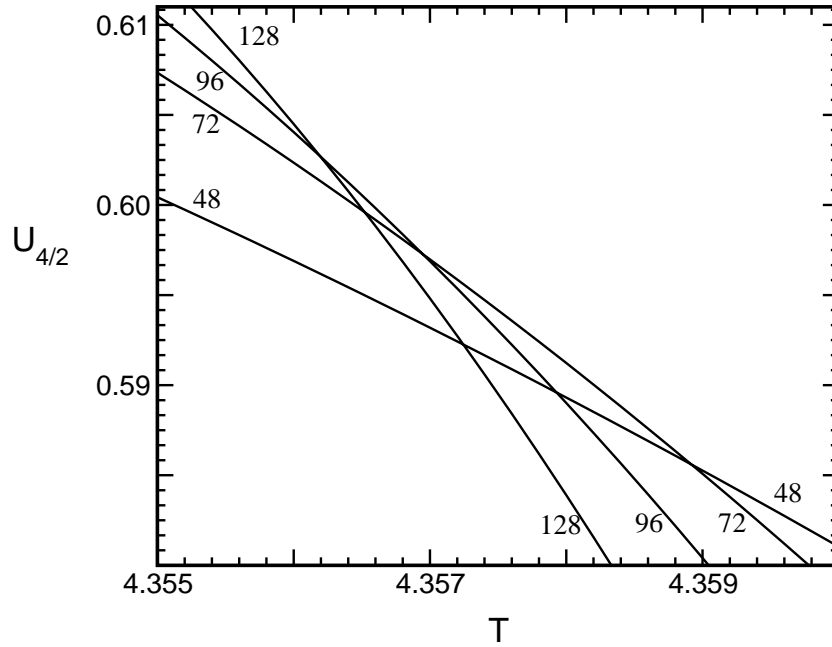
Die bisher in diesem Kapitel gemachten Aussagen gelten nur insoweit, daß die betrachteten Systeme so groß sind, daß Korrekturen zum Scaling vernachlässigbar sind. Dies ist jedoch bei Computersimulationen zugänglichen Systemgrößen nicht immer der Fall. So zeigen die Kumulanten für die hier in dieser Arbeit untersuchten Systemgrößen eine systematische Abweichung von einem gemeinsamen Schnittpunkt, die bei entsprechend feiner Skalierung der Achsen deutlich zutage tritt (siehe Abb.3.8). Notwendigerweise muß die Gleichung (3.67) durch die Aufnahme eines Korrekturtermes erweitert werden. Die Gleichung nimmt dann die Form [75]

$$U_L(x) = U^* + U'(0) \left( 1 - \frac{T}{T_c} \right) L^{1/\nu} + L^{-w} f_u(x) + \dots \tag{3.69}$$

an. Mit  $w$  wird der Exponent bezeichnet, welcher die Korrekturen zum Scaling beschreiben soll und  $f_u(x)$  ist die zugehörige Scaling-Funktion. Für die weitere Analyse wird die Scaling-Funktion in der Umgebung von  $T_c$  durch eine Konstante genähert,  $f_u(x) = \widehat{f}_u$ . Durch die Erweiterung schneiden sich die zu verschiedenen Systemgrößen gehörigen Kumulanten nicht bei  $T_c$ , sondern bei einem

$$T_L^x(b) = T_c \left( 1 + \frac{\widehat{f}_u}{U'(0)} L^{-w-1/\nu} \frac{1 - b^{-w}}{b^{1/\nu} - 1} \right) . \tag{3.70}$$

Aus (3.70) wird deutlich, daß es zwei verschiedene Möglichkeiten gibt, den Limes  $T^x \rightarrow T_c$  zu erreichen, entweder durch den Übergang  $L \rightarrow \infty$  oder durch  $b \rightarrow \infty$ .



**Abb. 3.8:** Kumulanten für das System mit  $D = 12$  und Oberflächenfeld  $h_1 = -0.015$ . Die Abweichungen von einem gemeinsamen Schnittpunkt treten in dieser Auflösung besonders deutlich zutage. Die zur jeweiligen Kumulante korrespondierende Gitterlänge  $L$  ist auch in die Abbildung aufgenommen.

Um aus den paarweisen Schnittpunkten der Kumulanten einen Schätzwert für  $T_c$  zu extrapolieren, wird man sich des letztgenannten Grenzübergangs bedienen. Bei Auftragen von  $T^x$  für festes  $L$  gegen  $(b^{1/\nu} - 1)^{-1}$  für verschiedene Werte von  $b$  läßt sich erwarten, daß die Werte ein lineares Verhalten zeigen und auf einer Geraden liegen. Deren Ordinatenabschnitt ist dann der Wert für die kritische Temperatur. Diese Aussagen gelten allerdings nur für den Fall, daß die Korrekturen zum Scaling gering sind, d.h., daß der Korrektur-Exponent  $w$  deutlich kleiner als  $1/\nu$  ist und somit die Approximation

$$\frac{1 - b^{-w}}{b^{1/\nu} - 1} \approx \frac{1}{b^{1/\nu} - 1} \quad (3.71)$$

gültig ist.

## 3.4 Details der Simulation

Die Generierung der Daten erfolgte mit einem Swendsen-Wang Cluster-Algorithmus, der um einen Ghost-Spin-Term erweitert wurde zur Berücksichtigung der Magnetfelder. Um Problemen mit korrelierten Zufallszahlen aus dem Wege zu gehen, die bei Verwendung des weit verbreiteten Lagged-Fibonacci-Generators R250 [81, 82] in Cluster-Algorithmen in Erscheinung traten [83–85], wurde eine verbesserte Version des R250 eingesetzt. Diese basiert auf der Verknüpfung der Ausgabe zweier unabhängiger Zufallszahlengeneratoren mit der logischen Operation “XOR”. Die beiden verwendeten Zufallszahlengeneratoren sind ein R250 mit dem “Magischen” Zahlenpaar [81, 82] (250, 103) und ein R521 mit dem Paar (521, 168).

Die untersuchten Filme hatten die Schichtdicke  $D = 4, 8, 12, 16, 24, 28$  und  $32$ . Für jeden einzelnen Wert von  $D$  wurden Simulationen zu mehreren verschiedenen lateralen Ausdehnungen  $L$  durchgeführt, um eine Extrapolation zum Limes  $L \rightarrow \infty$  zu ermöglichen. Die Werte für  $L$  variierten zwischen  $L = 16$  und  $L = 128$  und es wurde bei ihrer Auswahl darauf geachtet, daß das Verhältnis von lateraler Ausdehnung zu Schichtdicke mindestens  $L/D \geq 2$  beträgt, damit die Bezeichnung “dünner Film” zumindest annähernd gerechtfertigt ist.

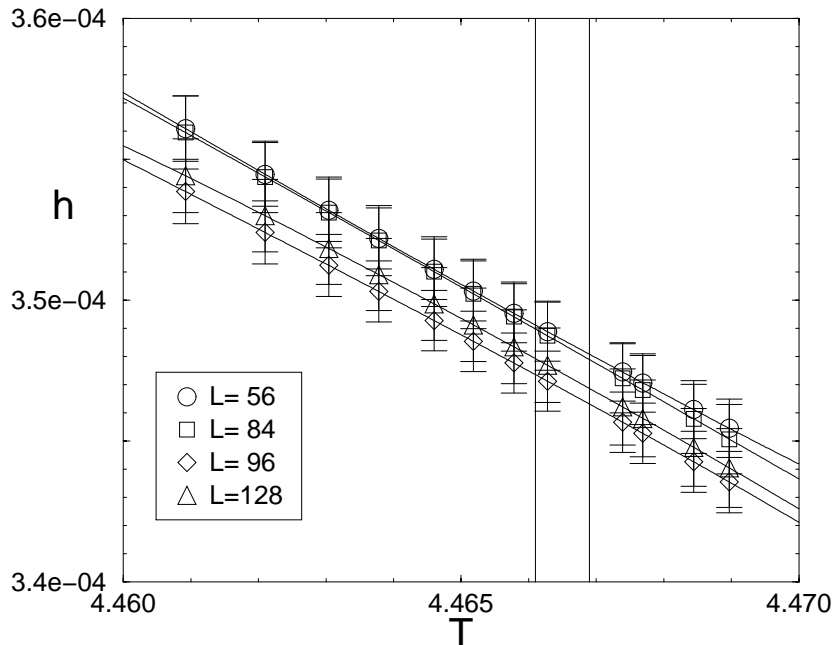
Zur Untersuchung des Einflusses der Wände auf das System wurden zwei Oberflächenmagnetfelder gewählt: Das verschwindende Oberflächenfeld  $h_1 = 0$  korrespondierend zum Fall neutraler Wände und das schwache Feld  $h_1 = -0.015$ .

Die Länge eines Simulationslaufes schwankte zwischen 200 000 und 1 200 000 Monte Carlo-Steps (MCS), wobei die Konfigurationen des ersten Zehntels des Laufes verworfen wurden, um sicherzustellen, daß nur Gleichgewichtskonfigurationen in die Analyse eingehen.

Die Werte der Magnetisierung und der Energie pro Spin wurden nach jedem Monte Carlo-Schritt in Ausgabedateien geschrieben. Diese Informationen sind zwar redundant aufgrund der Korrelation der Daten, aber durch die höhere Zahl von Einträgen in den Histogrammen waren diese “glatter” und es ließ sich eine bessere Eignung für die Anwendung der Reweighting-Techniken erhoffen. Die Werte der beiden Observablen wurden nach Schichten aufgelöst gespeichert, um eine spätere Analyse der Magnetisierungs- und Energieprofile zu ermöglichen.

Zum Auffinden der kritischen Punkte im von der Temperatur  $T$  und dem Bulk-Magnetfeld  $h$  aufgespannten 2-dimensionalen Parameterraum wurden als erster

Schritt Simulationen mit einigen zufällig bestimmten Parameterpaaren durchgeführt. Diese dienten für eine Einschätzung, auf welcher Seite der Phasenkoexistenzlinie man sich befindet und gaben somit die Richtung vor, in die der Wert  $h$  variiert werden mußte. Sobald man sich der Koexistenzlinie so weit angenähert hatte, daß die Histogramme des Ordnungsparameters Konfigurationen beider Phasen zeigten, konnte durch Reweighting entlang der Isotherme der Wert für das Koexistenz-Bulk-Feld bestimmt werden. Das System wurde als im Zustand der Phasenkoexistenz befindlich angesehen, wenn das statistische Gewicht der Ordnungsparameterverteilung für Werte unterhalb des Erwartungswertes gleich dem für Werte größer des Erwartungswertes war.



**Abb. 3.9:** Koexistenzkurve für  $D = 28$  in der Nähe des kritischen Punktes. Das Fehlerintervall der kritischen Temperatur wird durch die beiden senkrechten Linien dargestellt.

Um zu einer Einschätzung zu gelangen, ob die dieser Isothermen entsprechende Temperatur unter- oder oberhalb der kritischen Temperatur liegt, wurde die Kumulante  $U$  bestimmt und mit ihrem Fixpunktwert  $U^*$  für das 2-dimensionale Ising-Modell verglichen. Wenn durch diese Vorgehensweise der Bereich des kritischen Punktes grob eingegrenzt war, wurden in diesem zwischen 3 und 12 Simulationsläufe gestartet, auf deren Basis mit Multi-Histogramm-Reweighting die Koexistenzkurve im Bereich des kritischen Punktes (siehe als Beispiel Abb. 3.9), sowie die Werte der

interessierenden Observablen bestimmt wurden. Zu einer Einschätzung der Fehlergröße gelangte man durch die Anwendung von “Jack-knife”-Techniken [86].

# Kapitel 4

## Scaling-Theorie von Fisher und Nakanishi

Das wohl interessanteste Phänomen, das beim Studium des kritischen Verhaltens von Schichten endlicher Dicke, auf die ein Oberflächenmagnetfeld  $h_1$  wirkt, auftritt, ist die Verschiebung des kritischen Punktes  $(T_c(D), h_c(D))$  relativ zum kritischen Punkt  $(T_c^\infty, 0)$  des Volumens. Man beachte dabei, daß es sich auch bei einer dünnen Schicht um einen “echten” kritischen Punkt handelt, da nur die Ausdehnung in einer Raumdimension beschränkt ist und sich der thermodynamische Limes durch einen Grenzübergang  $L \rightarrow \infty$  in der Raumrichtung parallel zu den Oberflächen erreichen läßt. Eine Untersuchung dieser Verschiebung sowie der Phasenkoexistenzkurve wurde von Fisher und Nakanishi durchgeführt, die in eine Scaling-Theorie mündete [22]. Diese Theorie wurde in einer nachfolgenden Veröffentlichung [23] weiter konkretisiert und im Rahmen einer klassischen Mean-Field-Näherung erfolgte die numerische Bestimmung von Skalenfunktionen, Ordnungsparameterprofilen und Koexistenzkurven. Im folgenden soll diese Theorie kurz zusammengefaßt und erläutert werden.

Ausgangspunkt ist die Skalenhypothese für den singulären Anteil der freien Energie eines Filmes der Dicke  $D$ , der einem Magnetfeld  $h$  und einem Oberflächenfeld  $h_1$  ausgesetzt ist [87–91]

$$f_{\text{sing}}(D; T, h, h_1) \approx |t|^{2-\alpha} W_{\pm} \left( D|t|^{\nu} ; h|t|^{-\Delta} , h_1|t|^{-\Delta_1} \right) . \quad (4.1)$$

Diese Aussage ist asymptotisch gültig für den Grenzübergang  $t, h, h_1 \rightarrow 0$  und  $D \rightarrow \infty$ , wobei die Skalenkombinationen  $D|t|^{\nu}$ ,  $h|t|^{-\Delta}$  und  $h_1|t|^{-\Delta_1}$  endlich bleiben.

Mit  $t$  ist die reduzierte Abweichung der Temperatur von der kritischen Temperatur im Volumen,

$$t = \frac{(T - T_c^\infty)}{T_c^\infty}, \quad (4.2)$$

bezeichnet. Die auftretenden Exponenten  $\alpha$ ,  $\nu$  und  $\Delta = \beta + \gamma$  sind diejenigen, die auch das 3-dimensionale kritische Verhalten des Ising-Modells im Volumen bestimmen (siehe auch Kap. 3.3.1), während  $\Delta_1$  für den Exponenten steht, der das kritische Verhalten in der Oberflächenschicht eines halbbunendlichen Systems beschreibt [92]. Mit  $W_\pm$  ist die Skalenfunktion bezeichnet, wobei sich der Index auf das Vorzeichen von  $t$  bezieht. Führt man die Skalierungsvariablen  $x = D |t|^\nu$  und  $w = h_1 D^{\Delta_1/\nu}$  ein, folgt aus (4.1)

$$f_{\text{sing}} \approx |t|^{2-\alpha} W_\pm \left( D |t|^\nu; \frac{h D^{\Delta/\nu}}{x^{\Delta/\nu}}, \frac{w}{x^{\Delta_1/\nu}} \right). \quad (4.3)$$

Ein kritischer Punkt korrespondiert mit einer Singularität der Skalenfunktion  $W_\pm(x; y, y_1)$ . Diese zeigt eine Linie von Singularitäten, welche durch die Variable  $w = y_1 x^{\Delta_1/\nu}$  parametrisiert wird. Durch Wahl eines festen  $\tilde{w}$  kann ein einzelner kritischer Punkt  $(x_c(\tilde{w}), y_c(\tilde{w}))$  isoliert werden. Da die Skalenfunktion  $W_\pm(x; y, y_1)$  immer bei gleichen Werten ihrer Argumente singulär wird und das singuläre Verhalten allein durch diese bestimmt wird (die Funktion selbst ist im allgemeinen analytisch), lassen sich aus (4.3) direkt die folgenden Beziehungen für die Verschiebung der kritischen Temperatur und des kritischen Feldes ableiten:

$$\Delta T_c = -B_T D^{-1/\nu} X_c \left( C_T h_1 D^{\Delta_1/\nu} \right), \quad (4.4)$$

$$\Delta h_c = -B_h D^{-\Delta/\nu} Y_c \left( C_h h_1 D^{\Delta_1/\nu} \right). \quad (4.5)$$

Die neu eingeführten Skalenfunktionen  $X_c(w)$  und  $Y_c(w)$  sind bei Normierung ihrer metrischen Faktoren auf  $C_T = C_h = C_0$  universell, im Gegensatz zu den Amplituden  $B_T$  und  $B_h$  [22, 23].

Zur Untersuchung der Funktion  $X_c(w)$  betrachte man den Spezialfall eines verschwindenden Randfeldes  $h_1 = 0$ . In diesem Fall ergibt sich das einfache Skalenverhalten  $\Delta T_c \propto D^{-1/\nu}$  [87] einer dünnen Schicht mit freien Oberflächen, woraus

$$X_c(0) = 1 \quad (4.6)$$

resultiert. Da die Verschiebung der kritischen Temperatur aus Symmetriegründen nicht von der Orientierung des Randfeldes abhängig sein kann, können in einer Entwicklung von  $X_c$  nur gerade Potenzen von  $w$  auftreten:

$$X_c(w) = 1 + w^2 + \dots, \text{ für } w \rightarrow 0. \quad (4.7)$$

Für die Skalenfunktion  $Y_c(w)$  ist eine ähnliche Argumentation möglich. Aufgrund der Symmetrie verschwindet die Skalenfunktion im Grenzfall freier Oberflächen und es gilt somit

$$Y_c(0) = 0. \quad (4.8)$$

Im Gegensatz zur Verschiebung der kritischen Temperatur ist die des kritischen Feldes abhängig vom Vorzeichen des Oberflächenfeldes, weshalb in der Entwicklung der Funktion in diesem Fall nur ungerade Potenzen von  $w$  auftreten:

$$Y_c(w) = w [1 \pm w^2 + \dots], \text{ für } w \rightarrow 0. \quad (4.9)$$

Berücksichtigt man in der Entwicklung der Skalenfunktionen Terme bis zur ersten Ordnung, so gilt für die Verschiebungen

$$\Delta T_c \propto D^{-1/\nu}, \quad (4.10)$$

$$\Delta h_c \propto D^{-(\Delta - \Delta_1)/\nu}. \quad (4.11)$$

Eine Möglichkeit, die Linearität der Skalenfunktion  $Y_c(w)$  gemäß (4.9) zu überprüfen, ergibt sich im Rahmen der Analyse der Phasenkoexistenzlinie. Für deren Untersuchung bei  $T > 0$  bietet es sich an, mit verschwindenden Magnetfeldern  $h_1 = h = 0$  zu beginnen. Unter diesen Bedingungen koexistieren zwei Phasen, die durch die beiden verschiedenen Magnetisierungsprofile  $m^+(z)$  und  $m^-(z)$  beschrieben werden und die gleiche freie Energie besitzen. Es kann angenommen werden, daß beide Gleichgewichtsprofile stetig von den Feldern  $h$  und  $h_1$  abhängen, so daß sie in führender Ordnung als konstant bei kleinen Variationen in  $h_1$  angesehen werden können. Für die entsprechenden Änderungen der freien Energien gilt somit näherungsweise

$$-\Delta F^+ \approx \overline{m^+} h D L^2 + 2 m_s^+ h_1 L^2, \quad (4.12)$$

$$-\Delta F^- \approx -\overline{m^+} h D L^2 - 2 m_s^+ h_1 L^2. \quad (4.13)$$

Das Symbol  $\overline{m^+}$  steht hier für den Mittelwert über das Magnetisierungsprofil  $m^+(z)$  und mit  $m_s$  ist die Magnetisierung in den Oberflächenschichten bezeichnet. Es gilt  $\overline{m^+} = -\overline{m^-}$  und  $m_s^+ = -m_s^-$ . Auf der Koexistenzkurve muß  $\Delta F^+ = \Delta F^-$  gelten, womit aus (4.12) und (4.13) für das Koexistenzmagnetfeld

$$h_{\text{coex}}(D; T, h_1) \approx -\frac{2 h_1}{D} \frac{m_s^+(D; T)}{\overline{m^+}(D; T)} \quad (4.14)$$

folgt. Bei  $T = 0$  reproduziert (4.14) die Aussage (2.9), da im Grundzustand alle Spins die gleiche Orientierung haben und somit  $\overline{m^+} = m_s^+ = 1$  ist.

Ist die Bedingung  $D|t|^\nu \gg 1$  erfüllt, d.h.,  $D$  ist im Vergleich zur Korrelationslänge  $\xi$  groß, nehmen die Magnetisierungsprofile die Form an, die sie in einem halbbunendlichen System hätten: Für große  $z$  nähert sich  $\overline{m^+}(D; T)$  dem Wert der spontanen Magnetisierung  $m_0(T)$  im 3-dimensionalen Ising-Modell an. Entsprechend approximiert  $m_s^+(D; T)$  die spontane Oberflächenmagnetisierung  $m_1(T)$  im halbbunendlichen System. Die Gleichung (4.14) vereinfacht sich dann zu

$$h_{\text{coex}}(D; T, h_1) \approx -\frac{2 h_1}{D} \frac{m_1(D; T)}{m_0(D; T)}. \quad (4.15)$$

Nimmt man an, daß die eingeführten Näherungen auch im Bereich des kritischen Punktes ihre Gültigkeit behalten, so kann mit Hilfe der beiden folgenden Definitionen der kritischen Exponenten  $\beta$  und  $\beta_1$  [87, 89, 93]

$$m_0 \approx B_0 |t|^\beta, \quad (4.16)$$

$$m_1 \approx B_1 |t|^{\beta_1}, \quad (4.17)$$

die Phasenkoexistenzkurve zu

$$h_{\text{coex}}(D; T, h_1) \approx -\frac{2 h_1}{D} \frac{B_1}{B_0} |t|^{\beta_1 - \beta} \quad (4.18)$$

bestimmt werden, wobei die Bedingung eines kleinen Randfeldes und einer relativ zur Korrelationslänge großen Schichtdicke erfüllt sein muß.

Die Verschiebung des Magnetfeldes am kritischen Punkt erhält man einerseits durch Einsetzen der kritischen Temperatur, die man aus (4.4) im Grenzfall  $h_1 D^{\Delta/\nu} \rightarrow 0$  gewinnt:

$$h_c(D; h_1) = h_{\text{coex}}(D; T_c, h_1) \approx -\tilde{B} h_1 D^{-(1+(\beta_1-\beta)/\nu)} . \quad (4.19)$$

Zur Vereinfachung wurde die Abkürzung

$$\tilde{B} = 2 \frac{B_1}{B_0} \left( \frac{B_T}{T_c^\infty} \right)^{\beta_1-\beta} \quad (4.20)$$

eingeführt. Andererseits läßt sich die Verschiebung des Feldes auch direkt aus (4.5) unter Einbeziehung der linearen Approximierung der Skalenfunktion  $Y_c(w)$  (4.9) bestimmen:

$$h_c(D; h_1) = -\tilde{B} h_1 D^{-(\Delta-\Delta_1)/\nu} , \quad (4.21)$$

wobei in diesem Falle  $\tilde{B} = B_h C_h$  gelte. Die Konsistenz dieser beiden Ergebnisse und somit die Rechtfertigung des linearen Ansatzes in (4.9) läßt sich leicht durch Anwendung der Skalenrelationen [92]

$$\beta = 2 - \alpha - \Delta , \quad (4.22)$$

$$\beta_1 = 2 - \alpha - \Delta_1 - \nu \quad (4.23)$$

verifizieren.

Aus dem Ausdruck für die Phasenkoexistenzkurve (4.18) läßt sich auch eine Aussage über ihre Steigung am kritischen Punkt ableiten. Für den Steigungswinkel  $\Theta$  gilt

$$\tan(\Theta) = \frac{\partial h}{\partial T} \propto -\frac{h_1}{D} |t|^{\beta_1-\beta-1} . \quad (4.24)$$

Unter Verwendung des approximierten Skalenverhaltens der Temperatur  $|t| \propto D^{-1/\nu}$  in der Region des kritischen Punktes führt (4.24) zu

$$\begin{aligned} \frac{\partial h}{\partial T} &\propto -h_1 D^{-(\beta_1-\beta-1)/\nu-1} \\ &= -h_1 D^{-(\Delta-\Delta_1-1)/\nu} , \end{aligned} \quad (4.25)$$

wobei im letzten Schritt die Ausdrücke für die Skalenrelationen (4.22) und (4.23) zur Anwendung kamen. Im Limes  $D \rightarrow \infty$  verschwindet somit die Steigung der Koexistenzkurve am kritischen Punkt, falls die Relation  $\Delta - \Delta_1 > 1$  gilt. Bei Werten von  $\Delta = 1.564$  [94] und  $\Delta_1 = 0.464$  [95] ist dies in unserem Fall erfüllt. Somit sind nur geringe *field-mixing*-Effekte zu erwarten [96]

# Kapitel 5

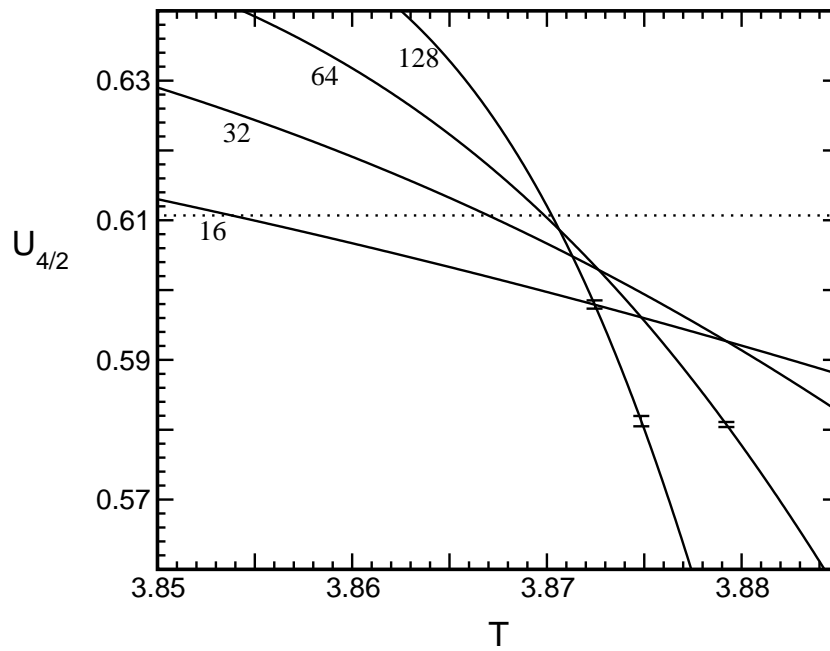
## Systeme mit Oberflächenfeld

Den Ergebnisteil möchte ich mit der Darstellung der Resultate für die dünnen Schichten, die einem schwachen Oberflächenfeld ausgesetzt sind, beginnen. Für alle untersuchten Schichtdicken  $D$  wurde ein fester Wert für dieses Magnetfeld von  $h_1 = -0.015$  gewählt. Die in diesem Kapitel angegebenen Werte der Observablen beziehen sich immer auf ein durch die Phasenkoexistenzforderung fixiertes Bulk-Magnetfeld  $h = h_{\text{coex}}$ . Somit ist die in den Abbildungen gezeigte Variation der Observablen in der Temperatur als Variation entlang der Koexistenzlinie  $h_{\text{coex}}(T)$  aufzufassen.

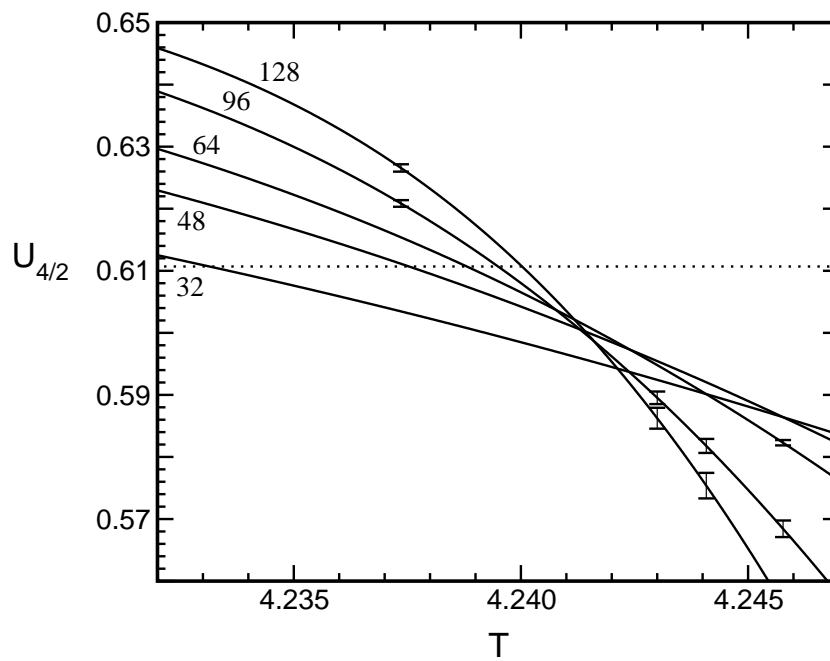
### 5.1 Kumulanten

Ein erster Ansatzpunkt zur Bestimmung der kritischen Temperatur  $T_c(D)$  ergibt sich durch die Untersuchung der Kumulanten 4. Ordnung  $U_{4/2}$  (vgl. Kap. 3.3.2). In den folgenden Abbildungen 5.1–5.7 sind die Kumulanten für die einzelnen Schichtdicken gegen die Temperatur aufgetragen. Die für das jeweilige  $D$  untersuchten Systemgrößen sind durch Angabe ihrer lateralen Ausdehnung  $L$  charakterisiert.

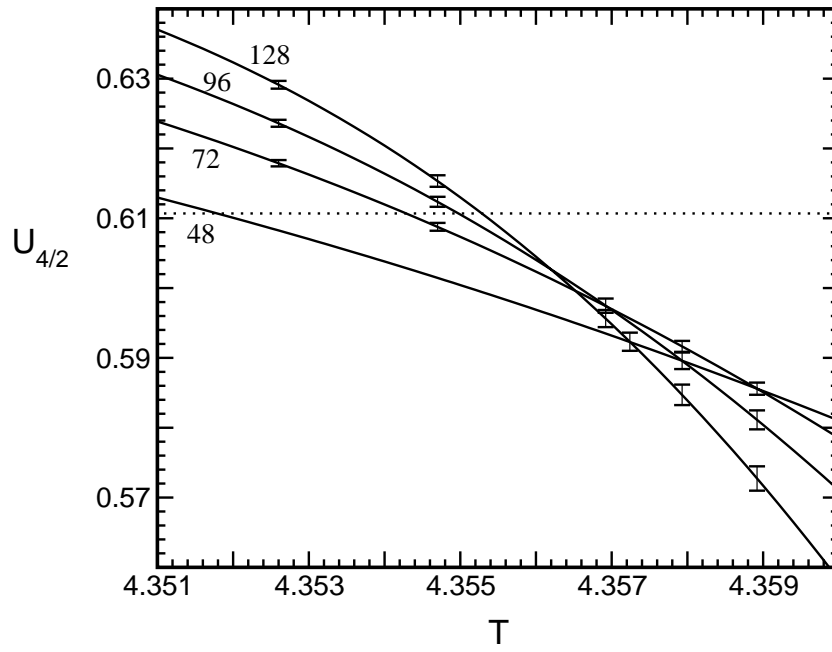
Um ein Gefühl für die erreichte Genauigkeit zu vermitteln, sind in den Bildern Fehlerbalken eingetragen, deren Angabe jedoch aus Gründen der Übersichtlichkeit auf einzelne Datenpunkte beschränkt bleiben muß, die repräsentativ für ganze Temperaturbereiche der Kumulante sind. Bei Fehlerbalken, deren Ausdehnung die Linienbreite der Kurven nicht überschreitet, wird auf eine Darstellung ganz verzichtet. Dies ist in der Regel zumindest für die jeweils kleinste, manchmal auch für die beiden



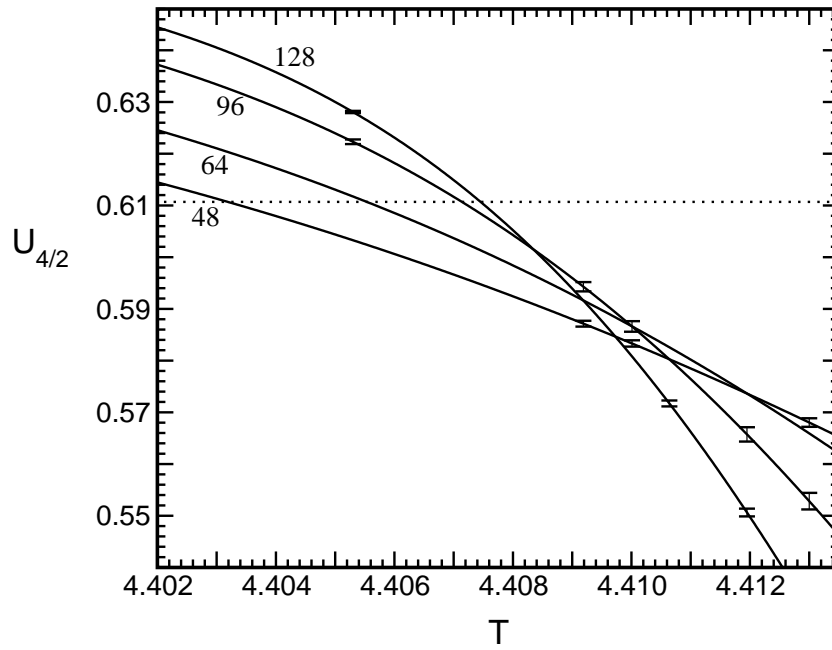
**Abb. 5.1:** Kumulantenschnittpunkte für  $D = 4$ . Die gepunktete Linie gibt den Wert von  $U^*$  an.



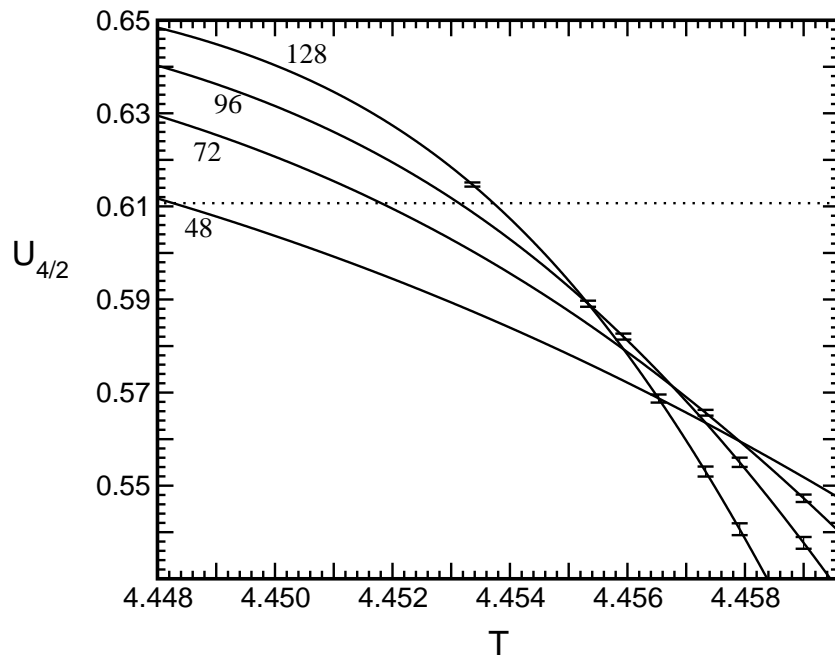
**Abb. 5.2:** Kumulantenschnittpunkte für  $D = 8$ . Die gepunktete Linie gibt den Wert von  $U^*$  an.



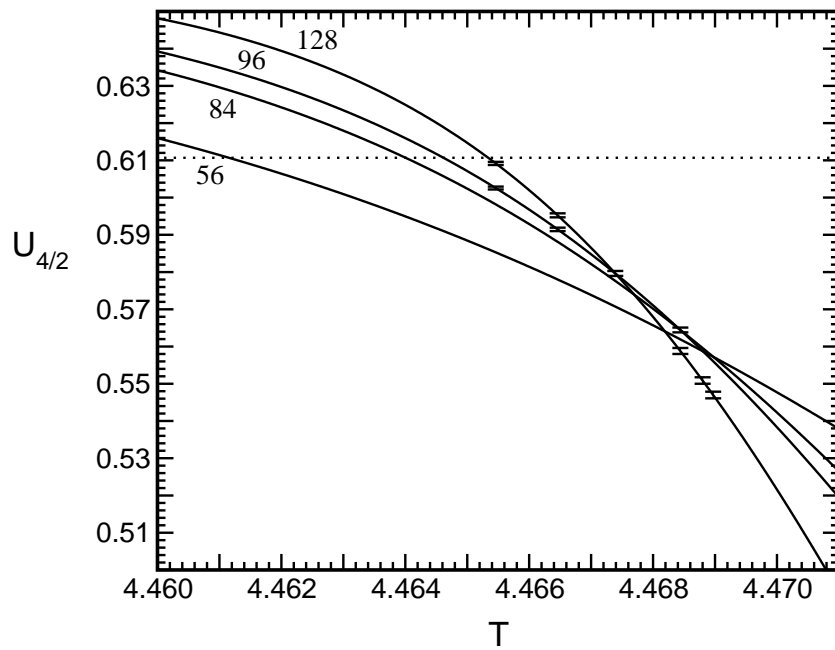
**Abb. 5.3:** Kumulantenschnittpunkte für  $D = 12$ . Die gepunktete Linie gibt den Wert von  $U^*$  an.



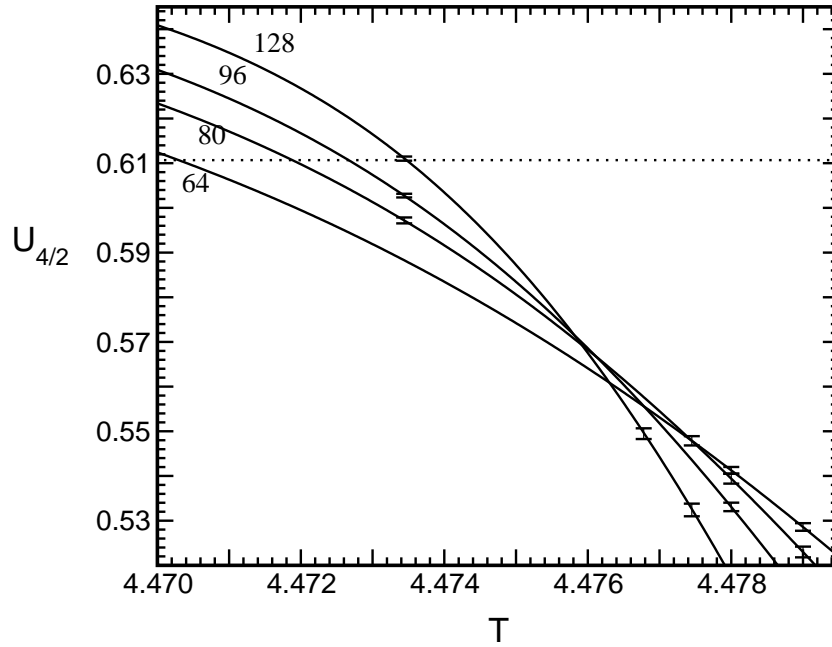
**Abb. 5.4:** Kumulantenschnittpunkte für  $D = 16$ . Die gepunktete Linie gibt den Wert von  $U^*$  an.



**Abb. 5.5:** Kumulantenschnittpunkte für  $D = 24$ . Die gepunktete Linie gibt den Wert von  $U^*$  an.



**Abb. 5.6:** Kumulantenschnittpunkte für  $D = 28$ . Die gepunktete Linie gibt den Wert von  $U^*$  an.



**Abb. 5.7:** Kumulantenschnittpunkte für  $D = 32$ . Die gepunktete Linie gibt den Wert von  $U^*$  an.

kleinsten Systemgrößen der Fall. Eine Ausnahme stellt die Dicke  $D = 16$  dar, bei der die Fehlerbalken für die kleinste Systemgröße die der nächst größeren übertreffen. Dies ist auf eine bessere Statistik durch einen größeren numerischen Aufwand für das zweitkleinste System zurückzuführen.

Die Schnittpunkte der Kumulanten miteinander zeigen eine deutliche Abweichung von einem gemeinsamen Kreuzungspunkt. Dies weist darauf hin, daß bei den untersuchten Systemen deutliche Korrekturen zum Skalenverhalten zum Tragen kommen. Die Abweichungen sind im Einklang mit der Systematik, die in den Gleichungen (3.69) und (3.70) zum Ausdruck kommt:

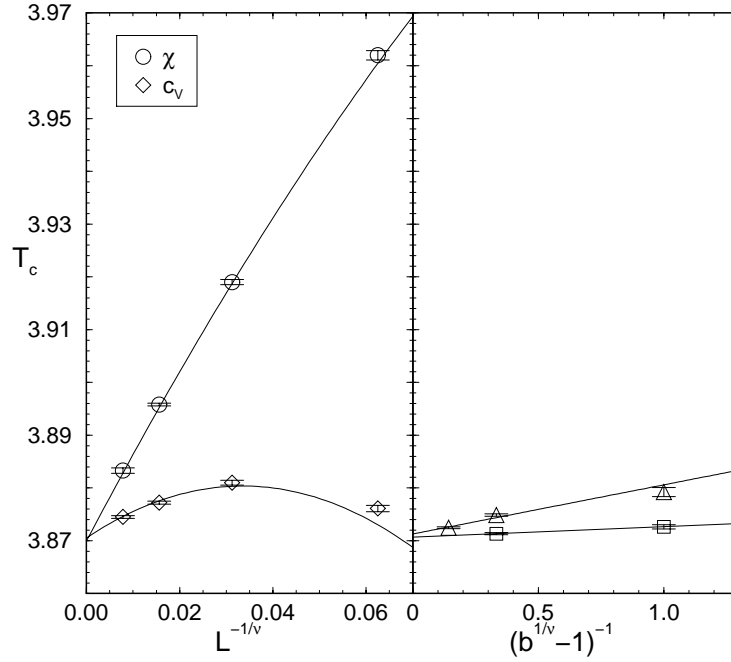
Eine Kumulante schneidet mit ansteigender Temperatur die der anderen Systeme in absteigender Reihenfolge ihrer Systemgrößen. Dieses Prinzip wird dem Augenschein nach für die Dicken  $D = 8$  und  $D = 28$  durchbrochen. Beachtet man jedoch die Größenordnung der Fehler, wird deutlich, daß in diesen Fällen aufgrund der statistischen Qualität der Daten, insbesondere der größeren Systeme, keine verlässliche Einschätzung der Systematik möglich ist.

Eine erste grobe Abschätzung der kritischen Temperaturen  $T_c(D)$  kann mit dem Temperaturbereich, der durch die Schnittpunkte der Kumulanten untereinander eingegrenzt ist, getroffen werden. Dieses Verfahren ist allerdings zur Untersuchung des

Fisher-Nakanishi-Scalings (vgl. Kap. 4) ungeeignet, da man aufgrund der fehlenden Extrapolation zu Gittern mit unendlicher lateraler Ausdehnung die kritische Temperatur  $T_c(D)$  unterschätzt.

In alle Abbildungen ist zur Orientierung der Wert des Fixpunktes  $U^*$  im 2-dimensionalen Ising-Modell als gepunktete Linie eingetragen. Tendenziell läßt sich erkennen, daß die Werte von  $U_{4/2}$  der Schnittpunkte sich mit zunehmender Schichtdicke immer weiter von diesem Wert entfernen. Die Ursache dafür ist das, aus der Perspektive der Asymptotik betrachtet, sich verschlechternde Verhältnis von Seitenlänge zu Schichtdicke mit steigendem  $D$ , weil die größte untersuchte Länge auf  $L = 128$  beschränkt war, um den numerischen Aufwand in vertretbarem Rahmen zu halten. Während man somit für die kleinste Schichtdicke noch ein Verhältnis von  $L/D = 32$  erreicht, sinkt dieses für die größte untersuchte Dicke auf  $L/D = 4$ , ein Umstand, der ein reines 2-dimensionales Verhalten nicht erwarten läßt. Der schwach universelle Wert des Fixpunktes  $U^*$  im 2-dimensionalen Ising-Modell wird hier nicht erreicht.

## 5.2 Bestimmung der kritischen Temperaturen



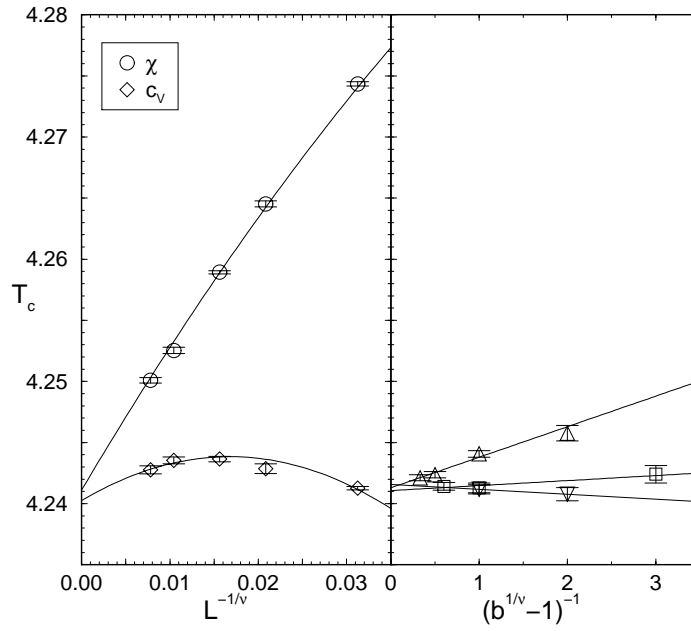
**Abb. 5.8:** Bestimmung von  $T_c$  für  $D = 4$ . Im linken Bild mit dem Maximum von spezifischer Wärme bzw. magnetischer Suszeptibilität, im rechten Bild mittels der Kumulantenschnittpunkte.

Zu Beginn dieses Unterkapitels möchte ich die einzelnen Schritte dargelegen, die zur Bestimmung der kritischen Punkte  $(T_c(D), h_c(D))$  ausgeführt wurden.

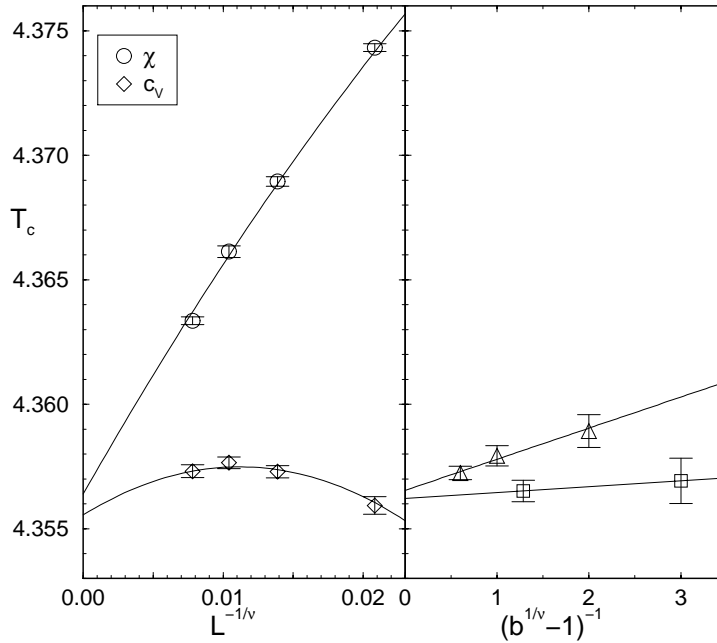
Die Grundlage der Untersuchung der kritischen Punkte bilden die Scaling-Eigenschaften dreier Observablen (vgl. Kap. 3.3.1 und 3.3.2):

- das Skalieren der Maxima der spezifischen Wärme mit der Seitenlänge  $L$ ,  $T_{c_v}^{max} - T_c(D) \propto L^{-1/\nu}$ ,
- das Skalieren der Maxima der magnetischen Suszeptibilität mit der Seitenlänge,  $T_{\chi}^{max} - T_c(D) \propto L^{-1/\nu}$ ,
- und das Skalieren der Schnittpunkte der Kumulanten  $U_{4/2}$  mit dem Verhältnis ihrer Seitenlängen  $b$ ,  $T_x \propto (b^{-1/\nu} - 1)^{-1}$ .

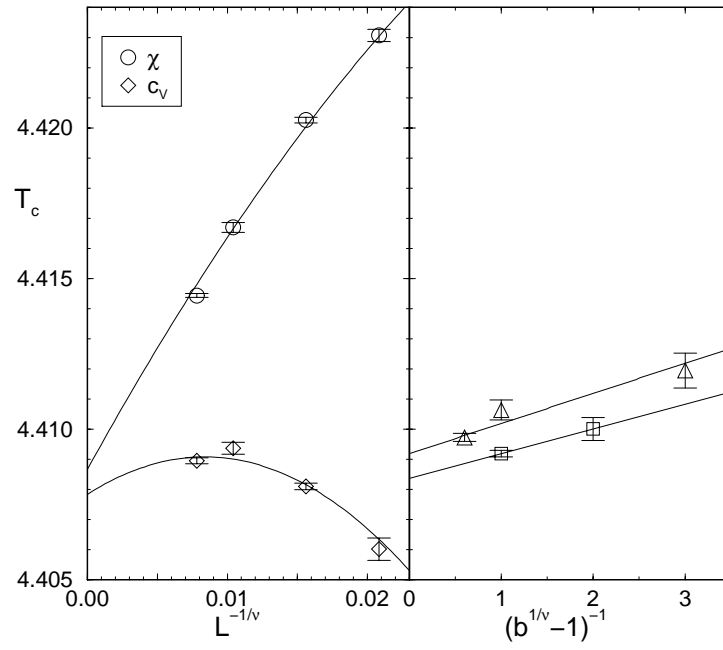
Die skalierten Observablen sind für die einzelnen Dicken  $D$  in Abb. 5.8–5.14 dargestellt.



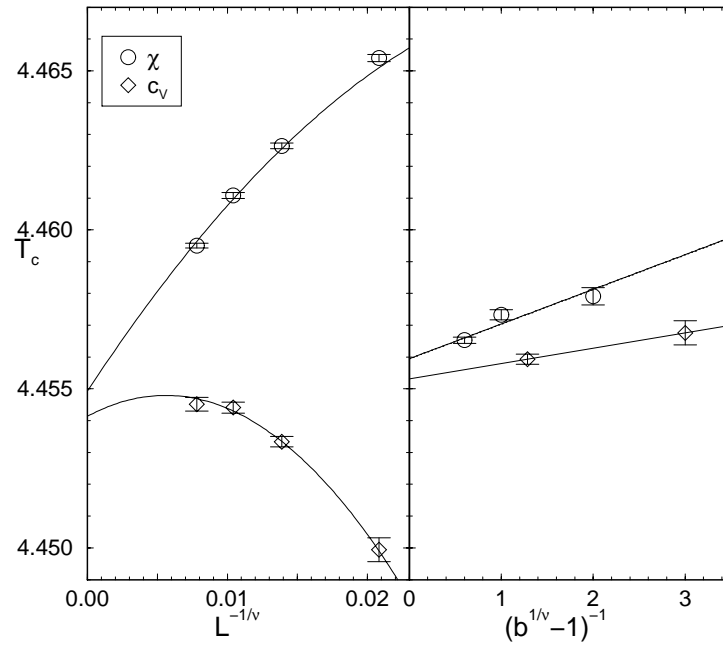
**Abb. 5.9:** Bestimmung von  $T_c$  für  $D = 8$ . Im linken Bild mit dem Maximum von spezifischer Wärme bzw. magnetischer Suszeptibilität, im rechten Bild mittels der Kumulantenschnittpunkte.



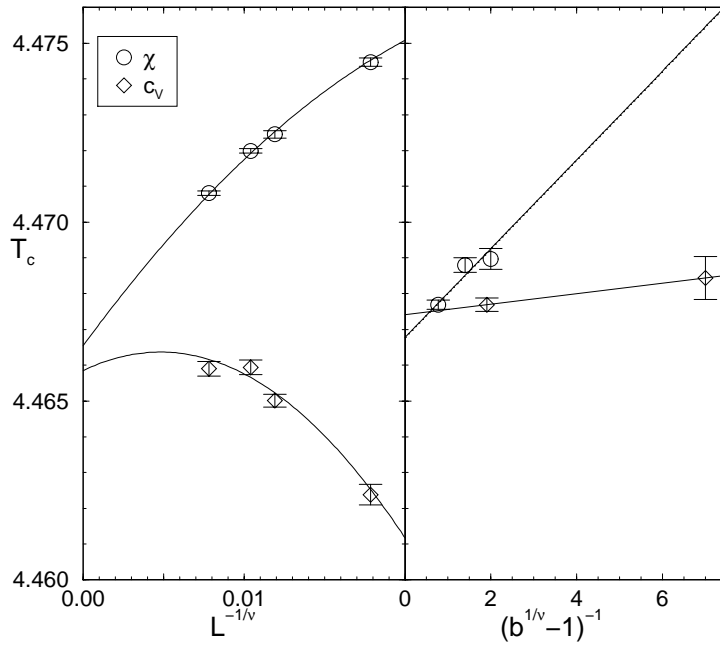
**Abb. 5.10:** Bestimmung von  $T_c$  für  $D = 12$ . Im linken Bild mit dem Maximum von spezifischer Wärme bzw. magnetischer Suszeptibilität, im rechten Bild mittels der Kumulantenschnittpunkte.



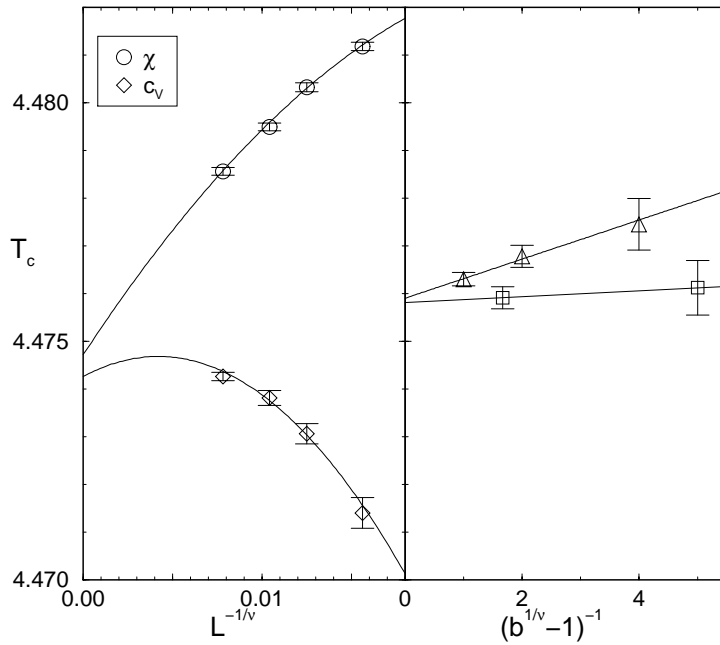
**Abb. 5.11:** Bestimmung von  $T_c$  für  $D = 16$ . Im linken Bild mit dem Maximum von spezifischer Wärme bzw. magnetischer Suszeptibilität, im rechten Bild mittels der Kumulanten-schnittpunkte.



**Abb. 5.12:** Bestimmung von  $T_c$  für  $D = 24$ . Im linken Bild mit dem Maximum von spezifischer Wärme bzw. magnetischer Suszeptibilität, im rechten Bild mittels der Kumulanten-schnittpunkte.



**Abb. 5.13:** Bestimmung von  $T_c$  für  $D = 28$ . Im linken Bild mit dem Maximum von spezifischer Wärme bzw. magnetischer Suszeptibilität, im rechten Bild mittels der Kumulanten-schnittpunkte.



**Abb. 5.14:** Bestimmung von  $T_c$  für  $D = 32$ . Im linken Bild mit dem Maximum von spezifischer Wärme bzw. magnetischer Suszeptibilität, im rechten Bild mittels der Kumulanten-schnittpunkte.

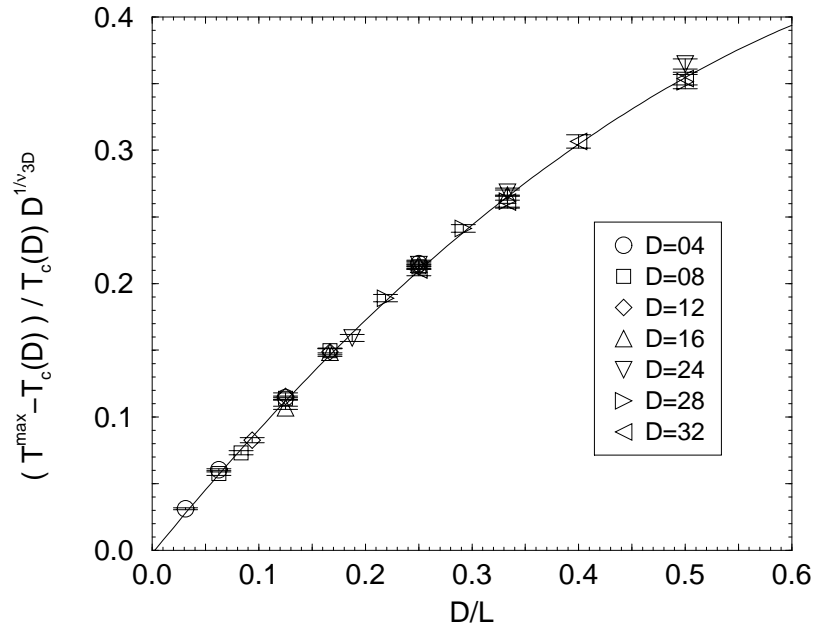
In allen drei Skalenrelationen wurden der zweidimensionale kritische Ising-Exponent  $\nu = 1$  verwendet.

An die in der rechten Teilabbildung eingezeichneten Werte der Kumulantenschnittpunkte lassen sich problemlos Geraden anpassen und durch Extrapolation  $b \rightarrow \infty$  können an der Ordinate zwei, bzw. im Fall  $D = 8$  drei Schätzwerte für die kritische Temperatur abgelesen werden. Die Konvergenz dieser Schätzwerte ist allerdings deutlich besser als die Größe der Fehlerbalken erwarten läßt. Dies ist ein Hinweis darauf, daß diese Werte für  $T_c(D)$  nicht als statistisch unkorreliert angesehen werden können. Dieser Punkt wird bei der späteren Diskussion der Ergebnisse nochmals aufgegriffen.

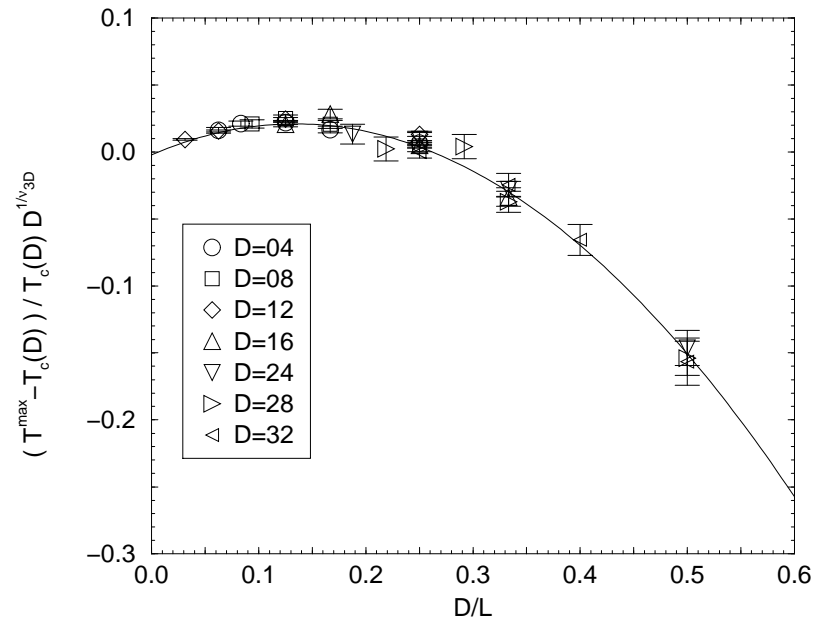
Die in der linken Teilabbildung dargestellten Werte der Maxima von spezifischer Wärme und magnetischer Suszeptibilität zeigen im Gegensatz zu den Kumulantenschnittpunkten eine deutliche Abweichung von einem linearen Verhalten. Diese Krümmung ist besonders bei der spezifischen Wärme ausgeprägt. Wegen der geringen Anzahl der Datenpunkte ist es nicht ratsam, ein Polynom 2. Grades an die Punkte anzupassen. Deshalb wurde an dieser Stelle ein anderer Weg beschritten.

Der Ansatz dafür ist, durch eine geeignete Skalierung der Werte ein Kollabieren der Maxima für verschiedene Systemgrößen auf eine Master-Kurve zu erzwingen. An diese läßt sich dann aufgrund der höheren Anzahl von Datenpunkten, in unserem Fall sind es 29, ein Polynom höherer Ordnung anpassen. Der gewünschte Datenkollaps läßt sich durch eine Auftragung von  $D^{1/\nu} (T - T_c(D))/T_c(D)$  gegenüber  $D/L$  erzielen. Dabei ist zu beachten, daß mit  $\nu$  der Wert des kritischen Exponenten des 3-dimensionalen Ising-Modells bezeichnet ist. Da in dem Ausdruck für die Skalierung das gesuchte  $T_c(D)$  selbst vorkommt, kann man auf diese Art die kritische Temperatur nur im Zuge eines Iterationsverfahrens bestimmen. Es besteht jedoch die Hoffnung, durch die Verwendung eines guten Schätzwertes für  $T_c(D)$  mit nur wenigen, bestenfalls sogar mit einer Iteration eine befriedigende Genauigkeit zu erreichen.

Als Iterationsstartwert bietet sich die aus den Kumulantenschnittpunkten gewonnene Schätzung der kritischen Temperatur an. Die skalierten Datensätze der Maxima für die verschiedenen Dicken wurden in einen Graphen eingetragen und diese Datensätze als ganzes bei Bedarf entlang der y-Achse verschoben, bis optisch ein Datenkollaps zu erkennen war. Diese Verschiebung entlang der y-Achse entspricht einer Anpassung des Ausgangsschätzwertes für  $T_c(D)$ . An den so justierten Gesamtdatensatz wurde ein Polynom zweiten Grades gefittet, da man ein analytisches Verhalten



**Abb. 5.15:** Master-Kurve für die magnetische Suszeptibilität  $\chi$ . Die durchgezogene Linie gibt die für die Analyse verwendete Kurve wieder.



**Abb. 5.16:** Master-Kurve für die spezifische Wärme  $c_V$ . Die durchgezogene Linie gibt die für die Analyse verwendete Kurve wieder.

der durch die oben angegebene Auftragung definierten Skalenfunktion  $f(D/L)$  erwartet. Die Parameter für die Parabel der Form  $y = a + bx + cx^2$  ergaben sich zu  $a = -0.000(2)$ ,  $b = 0.981(10)$  und  $c = -0.534(25)$  im Falle der Suszeptibilität,  $a = -0.000(2)$ ,  $b = 0.344(16)$  und  $c = -1.282(50)$  im Falle der spezifischen Wärme. Diese Kurven wurden dann in die Abb. 5.8–5.14 aufgenommen. Zur Optimierung des Ergebnisses wurde in den einzelnen Graphen ebenfalls eine Justierung der Fitkurven entlang der Temperaturachse vorgenommen. Nachdem eine ausreichende Übereinstimmung erreicht war, wurde an der Ordinate der jeweilige Wert für die kritische Temperatur abgelesen. Zur Überprüfung der Konsistenz der “von Hand” vorgenommenen Justierungen wurden die Rohdaten nochmals unter Verwendung der neu gewonnenen Schätzwerte für  $T_c(D)$  skaliert. Die so entstandenen Graphen zeigen einen deutlichen Datenkollaps (siehe Abb. 5.15 und 5.16). Die eingezeichneten Kurven entsprechen den oben angegebenen Parametern und zeigen eine gute Übereinstimmung mit den Datenpunkten. Die schon durch diese einzige Iteration erreichte Konsistenz ist für die weitere Analyse der Daten ausreichend und es wurde somit auf weitere Iterationen verzichtet.

$D$	$T_c(D)$	$h_c(D)$
4	3.8705(3)	0.006644(32)
8	4.2409(3)	0.002528(14)
12	4.3561(3)	0.001367(10)
16	4.4084(3)	0.000867(6)
24	4.4549(5)	0.000448(5)
28	4.4665(4)	0.000348(3)
32	4.4749(5)	0.000279(3)

**Tab. 5.1:** Werte für die kritischen Punkte ( $T_c(D)$ ,  $h_c(D)$ ) für verschiedene  $D$  bei vorhandenem Oberflächenfeld.

Aus den durch diese Methodik gewonnenen vier Werte für jedes  $T_c(D)$  muß ein geeigneter Mittelwert gebildet werden. Wie schon erwähnt, sind diese Schätzwerte als nicht vollständig statistisch unabhängig anzusehen. Diese Korrelationen kommen am stärksten bei den aus den Kumulantenschnittpunkten gewonnenen Temperaturen zum Tragen. Aus diesem Grund werden diese vorab zu einem Mittelwert zusammengezogen, so daß es nur noch zu einer Mittelwertbildung über drei Werte für  $T_c(D)$  kommt. In der Tabelle 5.1 sind diese für die jeweiligen Schichtdicken zusammengestellt.

Zur vollständigen Beschreibung der kritischen Punkte ist auch die Bestimmung der zugehörigen kritischen Bulk-Magnetfelder  $h_c(D)$  notwendig. Deren Werte können aus der Koexistenzkurve abgelesen werden. In Abbildung 3.9 ist eine solche Kurve exemplarisch für die Schichtdicke  $D = 28$  dargestellt. Die Werte des kritischen Feldes sind ebenfalls in Tabelle 5.1 aufgenommen. Die für  $h_c(D)$  angegebenen Fehler ergeben sich aus zwei verschiedenen Fehlerquellen. Dies ist zum einen die Ungenauigkeit bei der Ermittlung von  $T_c$ , zum anderen die bei der Berechnung der Koexistenzkurve  $h_{\text{coex}}(T)$ .

Durch die Bestimmung der kritischen Punkte für die einzelnen Schichtdicken sind nun die Voraussetzungen zur Untersuchung des Fisher-Nakanishi-Scalings geschaffen.

### 5.3 Untersuchung des Fisher-Nakanishi-Scalings

In einem ersten Schritt zum Test der Fisher-Nakanishi-Skalenhypothese soll für Systeme mit einem schwachen Oberflächenfeld  $h_1 = -0.015$  untersucht werden, ob die Beziehung, die die Abweichung der kritischen Temperatur eines Systems mit endlicher Schichtdicke  $D$  von der kritischen Temperatur des Volumens  $T_c^\infty$  beschreibt, gültig ist. Diese Beziehung ergibt sich aus (4.4) unter Verwendung der Näherung (4.6) zu

$$\Delta T_c \propto D^{-1/\nu} . \quad (5.1)$$

In diesem Abschnitt soll das negative Vorzeichen der Temperaturverschiebung in der Proportionalitätskonstante absorbiert werden, es gelte also

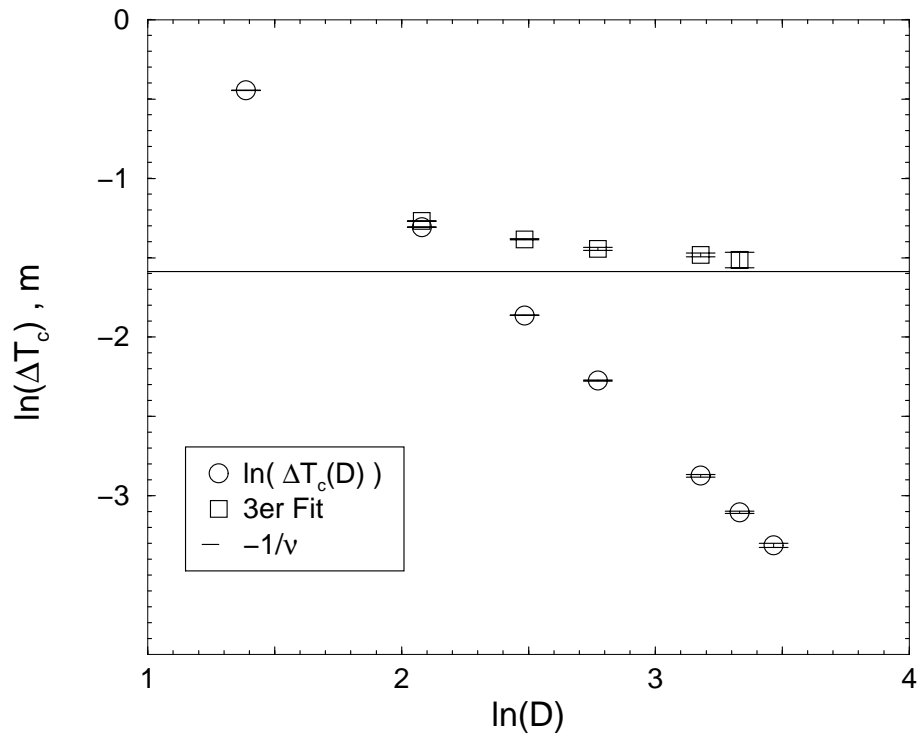
$$\Delta T_c = |T_c(D) - T_c^\infty| . \quad (5.2)$$

Die kritische Temperatur des Ising-Modells im Volumen ist mittels Monte Carlo-Untersuchungen mit einer sehr großen Präzision bestimmt worden. Die in verschiedenen Arbeiten [94,97–103] der letzten Jahre ermittelten Werte übertreffen in ihrer Genauigkeit deutlich die in dieser Arbeit erreichte Auflösung für  $T_c(D)$ . Für die weitere Diskussion der Ergebnisse wird ein Wert von  $T_c^\infty = 4.5115$  aus [94] verwendet.

In Abb. 5.17 ist die Verschiebung  $\Delta T_c$  gegen die Schichtdicke doppelt-logarithmisch aufgetragen. Der Verlauf der Linie zeigt eine leichte Krümmung, die darauf zurückzuführen ist, daß womöglich der asymptotische Bereich großer Schichtdicken für die untersuchten Systeme noch nicht erreicht ist. Aus diesem Grunde wurde an Gruppen von jeweils drei aufeinanderfolgenden Dicken eine Gerade gefittet. Die sich ergebenden Werte für die Geradensteigung sind in der Tabelle 5.2 zusammen mit den paarweisen Steigungen aufgenommen. Beide Serien zeigen eine deutliche Drift der Werte, die sich aber mit steigender Dicke dem verwendeten Literaturwert [94] annähern. Die mit dem gewichteten Fit erzielten Werte für die Gruppe mit den drei größten Dicken ist innerhalb der doppelten Standardabweichung mit dem Literaturwert verträglich. Für die paarweisen Steigungen ist die Aussagekraft der Ergebnisse für die größten Systeme aufgrund der immensen Fehlerintervalle sehr stark eingeschränkt.

Da die Ergebnisse vermuten lassen, daß für die untersuchten Schichtdicken der asymptotische Bereich noch nicht erreicht ist, wurde eine Extrapolation zu größeren Dicken hin versucht. Dazu wurden die Werte der Steigungen gegen  $D^{-1}$  aufgetragen

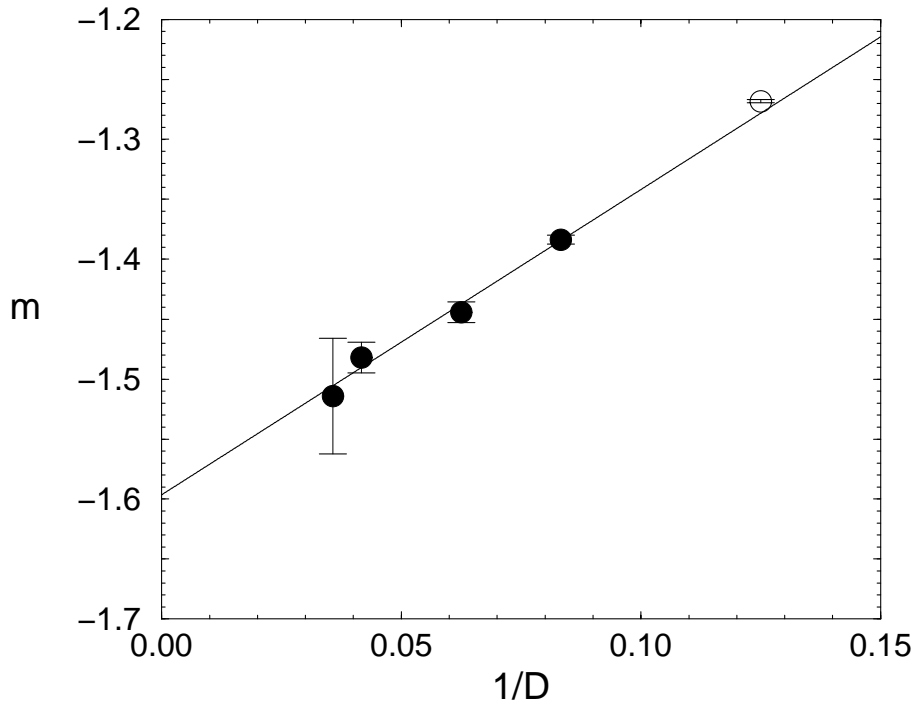
(siehe Abb. 5.18) und an die Daten eine Gerade der Form  $y = a + b x$  angepaßt. Dabei blieb in diesem Fall der Wert, in den das kleinste System  $D = 4$  eingeht, unberücksichtigt. Als Parameter der Kurve ergaben sich  $a = -1.596(21)$  und  $b = 2.55(27)$ . Der Wert des Ordinatenabschnitts  $a$  entspricht dem Wert des Exponenten  $-1/\nu$  bei Extrapolation zu  $D \rightarrow \infty$ . Er ist innerhalb seines Fehlerintervalles sehr gut mit dem Literaturwert verträglich. Die Fisher-Nakanishi-Skalenhypothese kann also im Rahmen dieser Extrapolation bezüglich der die Temperaturverschiebung behandelnden Teilhypothese bestätigt werden.



**Abb. 5.17:** Doppelt-logarithmischer Plot von  $\Delta T_c(D)$  gegen  $D$  (Kreise). Die Steigung der Fits an jeweils drei benachbarte Schichtdicken,  $m$  (Quadrate), nähern sich mit steigendem  $D$  dem Literaturwert für  $-1/\nu$  (durchgezogene Linie).

Fit		Paarweise Steigungen	
$D$	$m$	$D$	$m$
4, 8,12	-1.268(1)	4, 8	-1.244(3)
8,12,16	-1.384(4)	8,12	-1.369(7)
12,16,24	-1.444(9)	12,16	-1.428(16)
16,24,28	-1.482(13)	16,24	-1.480(26)
24,28,32	-1.514(48)	24,28	-1.490(100)
		28,32	-1.561(157)
$-1/\nu \approx -1.587$ [94]			

**Tab. 5.2:** Test des Fisher-Nakanishi-Scalings für  $\Delta T_c$ .



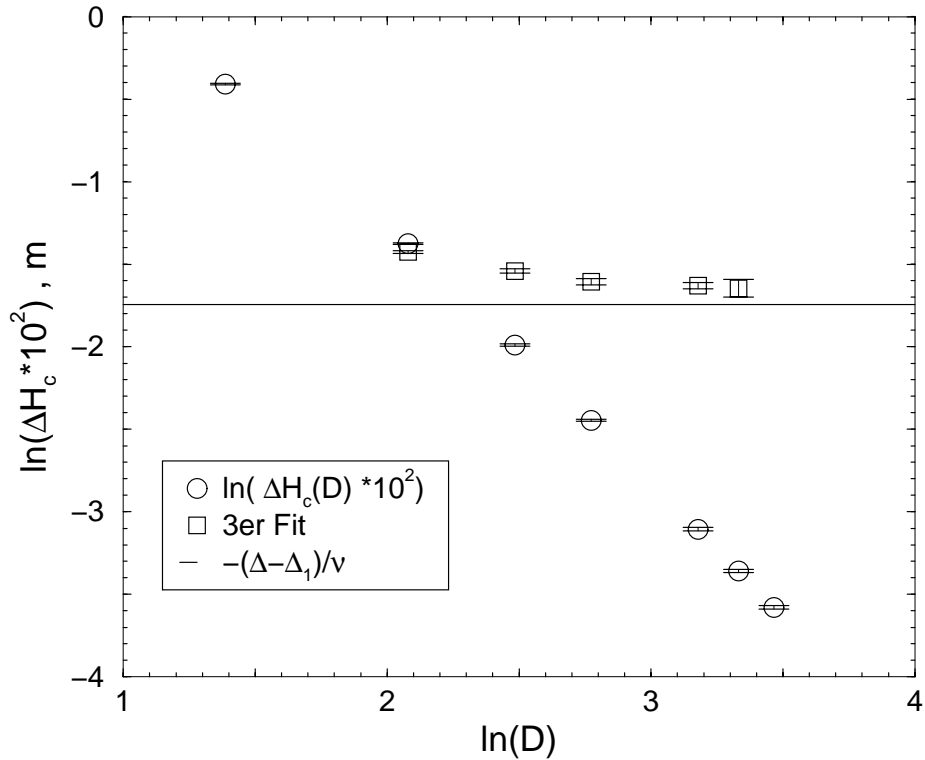
**Abb. 5.18:** Auftragung der Werte für  $-1/\nu$  aus den 3er Fits gegen die inverse Schichtdicke  $D$ . Die Gerade ist ein linearer Fit an die durch gefüllte Symbole dargestellten Werte.

Zu untersuchen bleibt der Teil der Hypothese, der die Verschiebung des kritischen Magnetfeldes  $\Delta h_c$  beschreibt. Diese ist identisch mit dem kritischen Magnetfeld  $h_c(D)$ , da im Volumen der Wert für Koexistenzmagnetfeld verschwindet. Aus der Fisher-Nakanishi-Theorie ergibt sich für die Verschiebung (vgl. (4.5))

$$\Delta h_c \propto D^{-(\Delta - \Delta_1)/\nu} , \quad (5.3)$$

wenn in der Entwicklung der Skalenfunktion (4.9) nur Terme bis erster Ordnung betrachtet werden. Durch Einsetzen von Literaturwerten für  $\Delta = \beta + \gamma = 1.564(3)$  und  $\nu = 0.6301$  (entnommen aus [94]) und  $\Delta_1 = 0.464$  (aus [95]) ergibt sich für den Exponenten

$$-(\Delta - \Delta_1)/\nu = -1.746 . \quad (5.4)$$



**Abb. 5.19:** Doppelt-logarithmischer Plot von  $\Delta h_c(D)$  gegen  $D$  (Kreise). Die paarweisen Steigungen  $m$  (Quadrate) nähern sich mit steigendem  $D$  dem Wert  $-(\Delta - \Delta_1)/\nu$ .

In Abb. 5.19 ist die Verschiebung  $\Delta h_c$  gegen die Schichtdicke doppelt-logarithmisch

aufgetragen. Wie im Falle der Temperaturverschiebung zeigt der Verlauf der Linie eine leichte Krümmung, die darauf zurückzuführen ist, daß womöglich auch in diesem Fall der asymptotische Bereich großer Schichtdicken für die untersuchten Systeme noch nicht erreicht ist. Aus diesem Grunde wurde analog zum vorherigen Vorgehen an Gruppen von jeweils drei aufeinanderfolgenden Dicken eine Gerade gefittet. Die sich ergebenden Werte für die Geradensteigung sind in der Tabelle 5.3 zusammen mit den paarweisen Steigungen aufgenommen. Eine deutliche Drift der Werte ist bei beiden Serien gegeben, die sich aber mit steigender Dicke dem verwendeten Literaturwert (5.4) annähern. Die mit dem gewichteten Fit erzielten Werte für die Gruppe mit den drei größten Dicken ist innerhalb der doppelten Standardabweichung mit dem Wert (5.4) verträglich. Für die paarweisen Steigungen ist auch hier die Aussagekraft der Ergebnisse für die größten Systeme aufgrund großer Fehlerintervalle sehr stark eingeschränkt.

Da auch hier der asymptotische Bereich noch nicht erreicht ist, wurde eine Extrapolation zu größeren Dicken hin versucht. Dazu wurden die Werte der Steigungen wiederum gegen  $D^{-1}$  aufgetragen (siehe Abb. 5.20) und an die Daten eine Gerade der Form  $y = a + b x$  angepaßt. Als Parameter der Kurve ergaben sich  $a = -1.754(20)$  und  $b = 2.61(19)$ . Der Wert des Ordinatenabschnitts  $a$  entspricht dem Wert des Exponenten  $-(\Delta - \Delta_1)/\nu$  bei Extrapolation zu  $D \rightarrow \infty$ . Er ist innerhalb seines Fehlerintervalles sehr gut mit dem Literaturwert verträglich. Auch der Teil der Fisher-Nakanishi-Skalenhypothese, der die Verschiebung des kritischen Magnetfeldes beschreibt, kann im Rahmen dieser Extrapolation bestätigt werden.

Im allgemeinen ist die Bestimmung der kritischen Exponenten, die das Verhalten eines Systems im Volumen beschreiben, technisch einfacher durchzuführen als die der Oberflächenexponenten. Deswegen ist die Genauigkeit, mit der Exponenten wie z.B.  $\Delta_1$  bekannt sind, geringer. Ein Schätzwert für  $\Delta_1$  aus dem Extrapolationsergebnis  $a$  kann durch

$$\Delta_1 = \Delta + a \nu \quad (5.5)$$

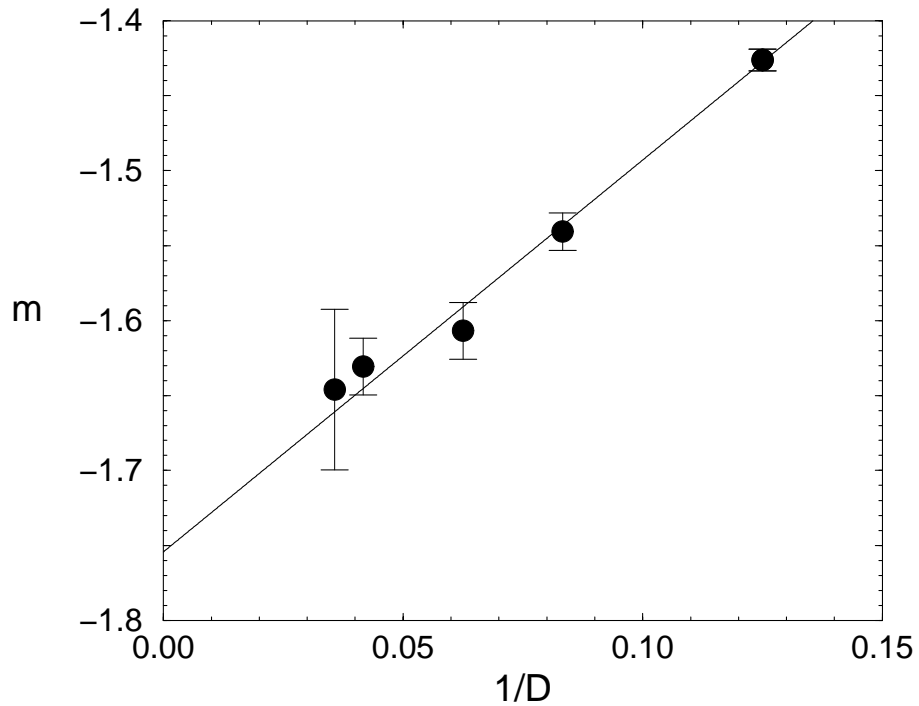
zu

$$\Delta_1 = 0.459(13) \quad (5.6)$$

ermittelt werden. Dieser stimmt sehr gut mit dem oben angeführten Wert aus [95] überein.

Fit		Paarweise Steigungen	
$D$	$m$	$D$	$m$
4, 8,12	-1.426(8)	4, 8	-1.394(15)
8,12,16	-1.541(12)	8,12	-1.516(32)
12,16,24	-1.607(19)	12,16	-1.583(50)
16,24,28	-1.631(19)	16,24	-1.628(44)
24,28,32	-1.646(54)	24,28	-1.639(128)
		28,32	-1.655(145)
$-(\Delta - \Delta_1)/\nu \approx -1.746$			

**Tab. 5.3:** Test des Fisher-Nakanishi-Scalings für  $\Delta h_c$ .



**Abb. 5.20:** Auftragung der Werte für  $-(\Delta - \Delta_1)/\nu$  aus den 3er Fits gegen die inverse Schichtdicke  $D$ . Die Gerade ist ein linearer Fit an die durch gefüllte Symbole dargestellten Werte.

## 5.4 Untersuchung des 2d-kritischen Verhaltens

In diesem Abschnitt soll das Skalen-Verhalten in der Nähe des kritischen Punktes untersucht werden. Dafür wurden die Observablen Magnetisierung  $m$ , Oberflächenmagnetisierung  $m_1$ , magnetische Suszeptibilität  $\chi$  und magnetische Suszeptibilität der Oberflächen  $\chi_{11}$  ausgewählt, deren Definitionen hier kurz zusammengestellt sind:

$$m = -\frac{\partial f}{\partial H} = \frac{1}{L^2 D} \sum_i S_i, \quad (5.7)$$

$$m_1 = -\frac{\partial f}{\partial H_1} = \frac{1}{2L^2} \sum_{i \in \text{Oberflächen}} S_i, \quad (5.8)$$

$$\chi = -\frac{\partial^2 f}{\partial H^2} = \beta L^2 D [\langle m^2 \rangle - \langle m \rangle^2], \quad (5.9)$$

$$\chi_{11} = -\frac{\partial^2 f}{\partial H_1^2} = \beta L^2 [\langle m_1^2 \rangle - \langle m_1 \rangle^2], \quad (5.10)$$

wobei es sich bei  $f$  um die freie Energie handelt. Bei der Untersuchung dieser Observablen in einer Computersimulation ist darauf zu achten, daß die in unendlichen bzw. sehr großen Systemen auftretende spontane Symmetriebrechung in Simulationen aufgrund der verhältnismäßig kleinen Systemgrößen nicht erreicht wird, da die Wahrscheinlichkeit des Übergangs in die andere Phase auch unterhalb der kritischen Temperatur deutlich von null verschieden ist. Somit würde das System bei ausreichend langer Simulationsdauer mehrmals zwischen den beiden Phasen wechseln, und der Erwartungswert des Ordnungsparameters würde bei allen Temperaturen verschwinden. Dieses Problem wird in einer Simulation dadurch umgangen, daß man die Beträge der Größen betrachtet, also in unserem Fall  $\langle |m| \rangle$  und  $\langle |m_1| \rangle$  anstelle von  $\langle m \rangle$  und  $\langle m_1 \rangle$ .

Nähert man sich entlang der Koexistenzlinie dem kritischen Punkt  $T_c(D)$ , so steigt die Korrelationslänge  $\xi$  an und erreicht in etwa die Größenordnung der Schichtdicke  $D$ . Damit wird die eingeschränkte Ausdehnung des Systems entlang der Wandnormalen relevant. Das System ändert sein kritisches Verhalten von dem eines drei- zu dem eines zweidimensionalen Ising-Modells, es tritt ein sogenanntes *Crossover* auf. Somit erwartet man für die hier betrachteten Observablen folgendes kritische Verhalten:

$$|m - \langle m \rangle| \propto |m_1 - \langle m_1 \rangle| \propto |t|^{\beta_2}, \quad (5.11)$$

und für die Suszeptibilitäten

$$\chi \propto \chi_{11} \propto |t|^{\gamma_2} . \quad (5.12)$$

Sowohl die beiden Magnetisierungen als auch die beiden Suszeptibilitäten haben jeweils die gleiche temperaturartige Singularität, die sich nur in ihren Amplituden unterscheidet.

Wie der Index anzeigt, soll es sich um die Exponenten des zweidimensionalen Ising-Modells handeln, und es gilt  $t = 1 - T/T_c(D)$ . Durch die Oberflächenfelder zeigt das hier untersuchte System eine Asymmetrie (siehe auch Abb. 3.5), die zur Folge hat, daß der Erwartungswert der Magnetisierung am kritischen Punkt nicht verschwindet. Deswegen muß in der Skalenrelation (5.11) die Änderung der Magnetisierungen relativ zu ihrem Erwartungswert betrachtet werden.

Läßt man die Temperatur konstant und erhöht die Anzahl der Schichten, so erwartet man ein Crossover zu einem dreidimensionalen Verhalten, welches von den Exponenten  $\beta$  und  $\beta_1$ , bzw.  $\gamma$  und  $\gamma_{11}$  bestimmt wird.

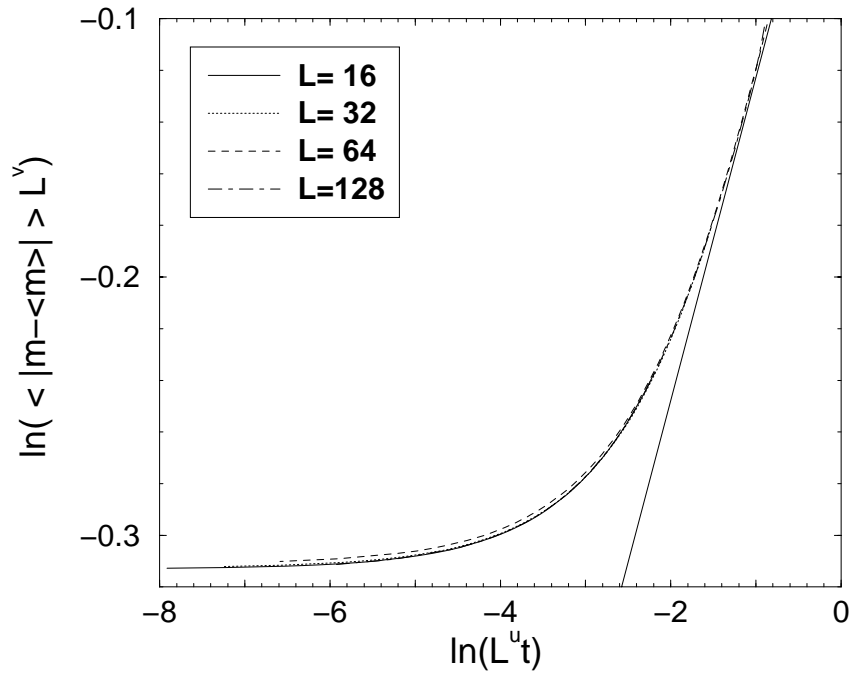
Da bei den hier untersuchten Systemgrößen, besonders für die dickeren Schichten, das Verhältnis  $L/D$  nicht ausreichend war, um ein reines zweidimensionales Verhalten beobachten zu können, muß man sich zur Beschreibung des kritischen Verhaltens mit sogenannten *effektiven* Exponenten  $u$ ,  $v$  und  $w$  behelfen, deren Werte im Idealfall zwischen den Exponenten des zwei- und dreidimensionalen Ising-Modells interpolieren. Mit Hilfe dieser effektiven Exponenten kann versucht werden, die Daten an Skalenformen wie

$$\langle |m - \langle m \rangle| \rangle L^v = \tilde{M}(L^u t) , \quad (5.13)$$

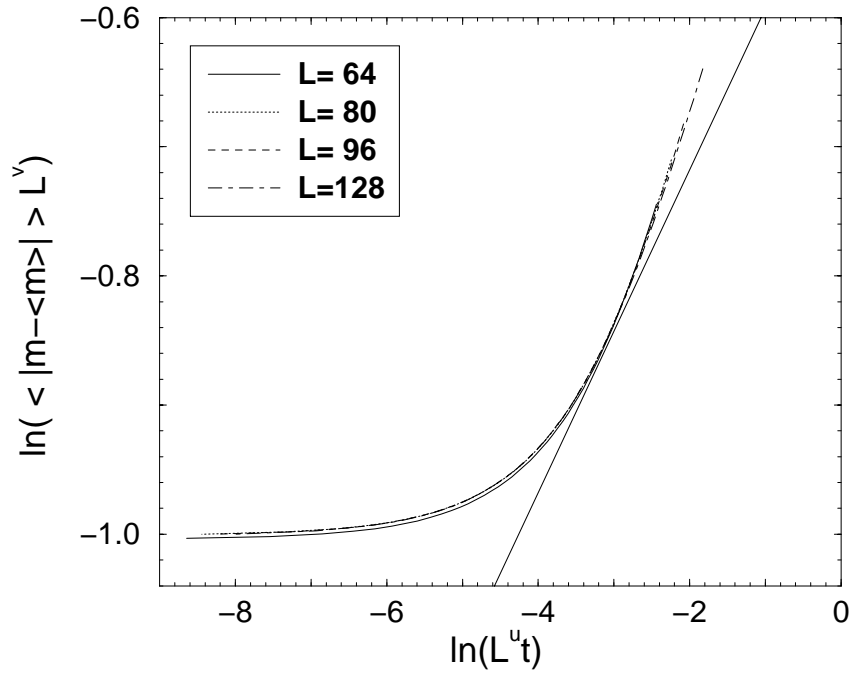
$$\chi L^{-w} = \tilde{\chi}(L^u t) , \quad (5.14)$$

so anzupassen, daß sie für verschiedene laterale Ausdehnungen  $L$  auf eine einzelne Master-Kurve  $\tilde{M}(z)$  kollabieren. Im Falle eines reinen zweidimensionalen Verhaltens gilt  $u = 1/\nu_2$ ,  $v = \beta_2/\nu_2$  und  $w = \gamma_2/\nu_2$  (vgl. Kap. 3.3.1). Damit lassen sich die Exponenten zu  $\beta_{\text{eff}} = v/u$ ,  $\gamma_{\text{eff}} = w/u$  und  $\nu_{\text{eff}} = 1/u$  berechnen.

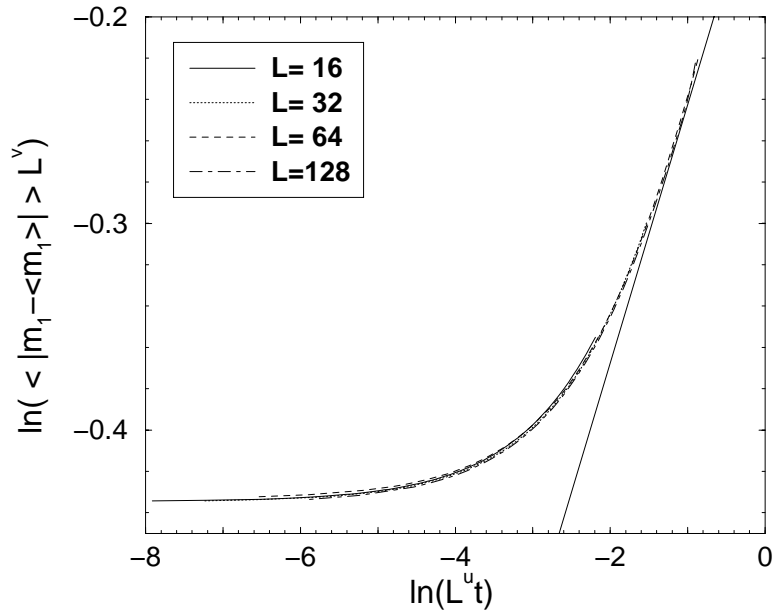
Zur Bestimmung der effektiven Exponenten wurde als eine erste Abschätzung von  $u$  für jede Schichtdicke die Steigung der Kumulanten am kritischen Punkt  $T_c(D)$  untersucht. Trägt man diese doppelt-logarithmisch gegen  $L$  auf, so erwartet man, daß die Daten auf einer Geraden mit der Steigung  $1/\nu = u$  liegen.



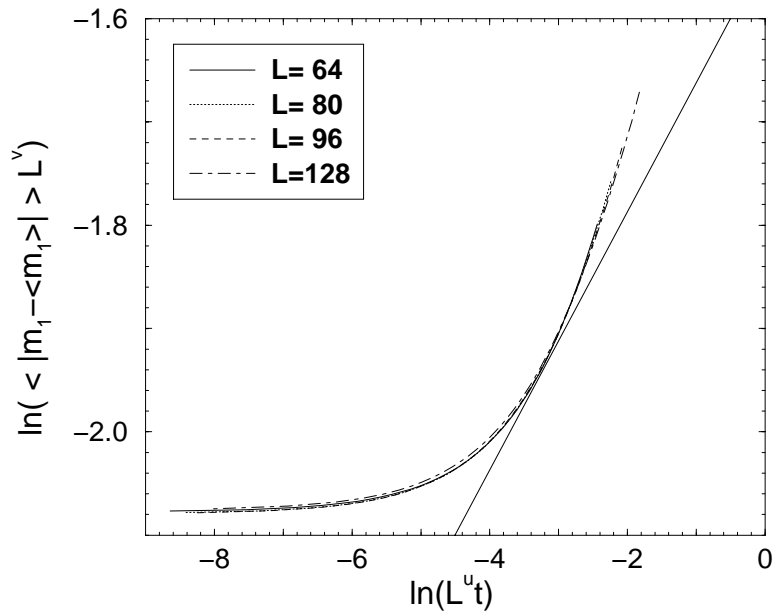
**Abb. 5.21:** Doppelt-logarithmischer Plot des Scalings der Magnetisierung  $m$  für  $D = 4$ . Die eingezeichnete Gerade hat eine Steigung, die einem Exponenten  $\beta = 1/8$  entspricht.



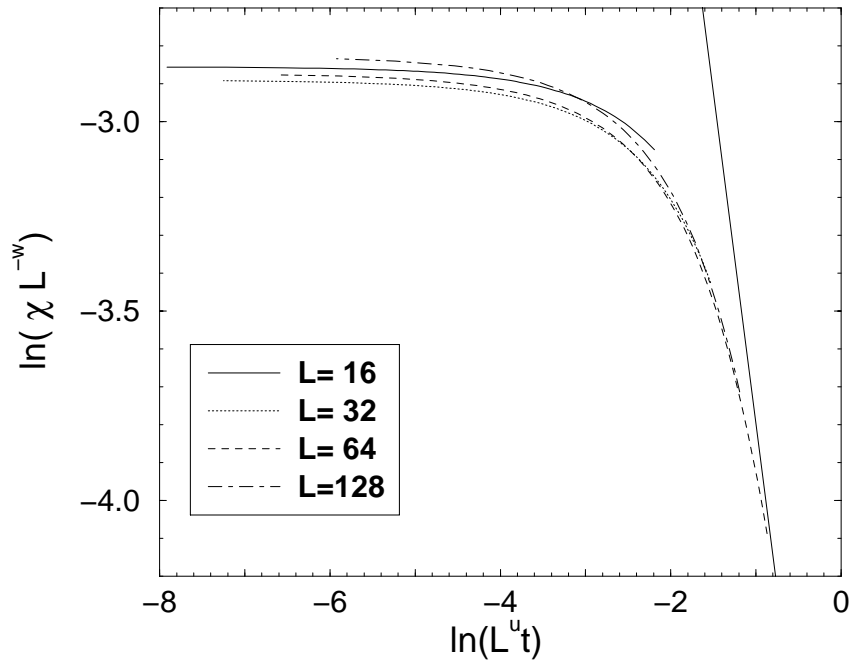
**Abb. 5.22:** Doppelt-logarithmischer Plot des Scalings der Magnetisierung  $m$  für  $D = 32$ . Die eingezeichnete Gerade hat eine Steigung, die einem Exponenten  $\beta = 1/8$  entspricht.



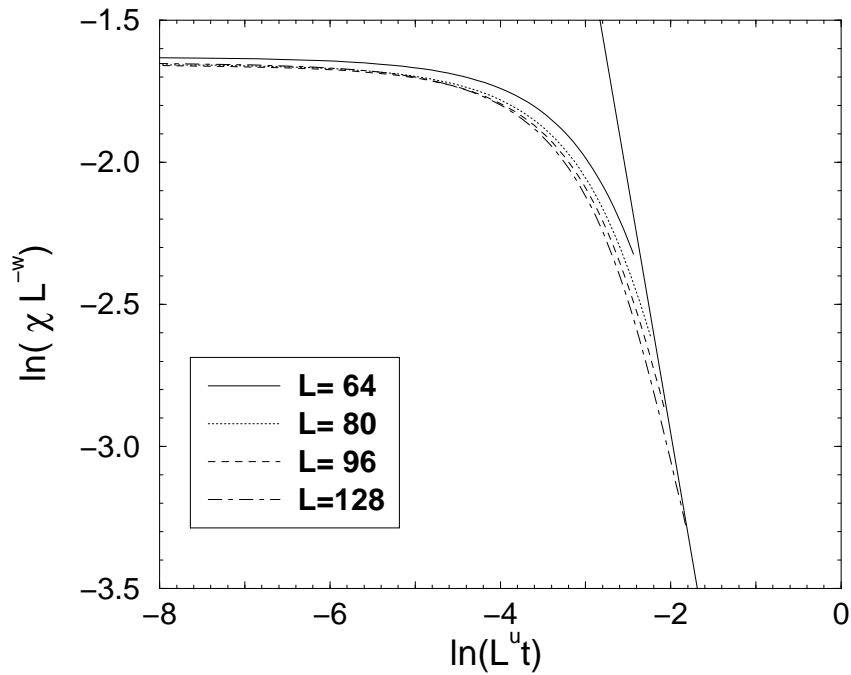
**Abb. 5.23:** Doppelt-logarithmischer Plot des Scalings der Oberflächenmagnetisierung  $m_1$  für  $D = 4$ . Die eingezeichnete Gerade hat eine Steigung, die einem Exponenten  $\beta = 1/8$  entspricht.



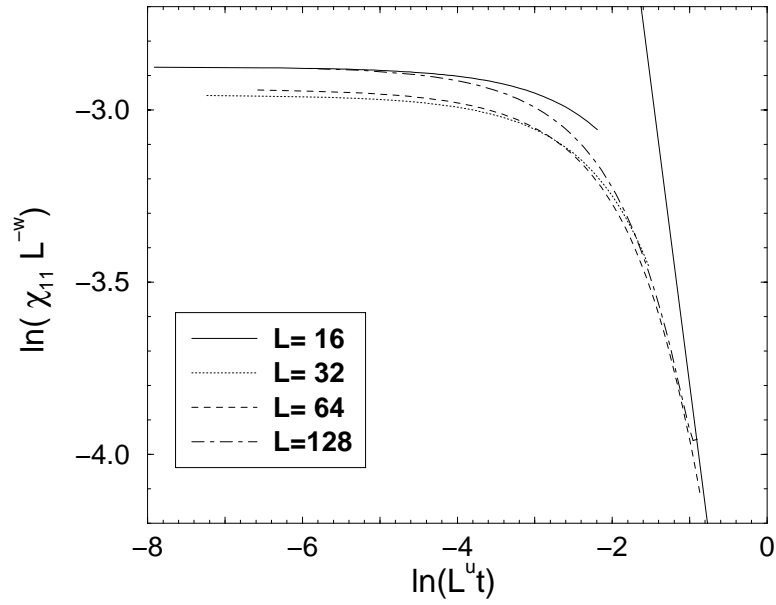
**Abb. 5.24:** Doppelt-logarithmischer Plot des Scalings der Oberflächenmagnetisierung  $m$  für  $D = 32$ . Die eingezeichnete Gerade hat eine Steigung, die einem Exponenten  $\beta = 1/8$  entspricht.



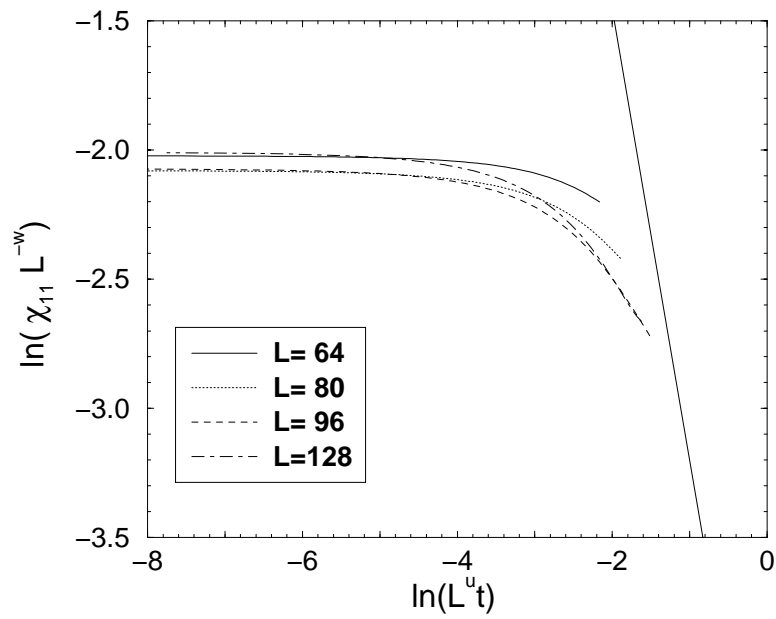
**Abb. 5.25:** Doppelt-logarithmischer Plot des Scalings der Suszeptibilität  $\chi$  für  $D = 4$ . Die eingezeichnete Gerade hat eine Steigung, die einem Exponenten  $\gamma = 7/4$  entspricht.



**Abb. 5.26:** Doppelt-logarithmischer Plot des Scalings der Suszeptibilität  $\chi$  für  $D = 32$ . Die eingezeichnete Gerade hat eine Steigung, die einem Exponenten  $\gamma = 7/4$  entspricht.



**Abb. 5.27:** Doppelt-logarithmischer Plot des Scalings der Oberflächenssuszeptibilität  $\chi_{11}$  für  $D = 4$ . Die eingezeichnete Gerade hat eine Steigung, die einem Exponenten  $\gamma = 7/4$  entspricht.



**Abb. 5.28:** Doppelt-logarithmischer Plot des Scalings der Oberflächenssuszeptibilität  $\chi_{11}$  für  $D = 16$ . Die eingezeichnete Gerade hat eine Steigung, die einem Exponenten  $\gamma = 7/4$  entspricht.

In der Nähe der kritischen Temperatur ist infolge von Finite-Size-Effekten ( $L < \xi$ ) der Wert der Magnetisierung annähernd konstant. Er nimmt für alle Längen  $L$  Plateau-Werte an. Eine Abschätzung für den Wert von  $\nu$  erhält man dadurch, daß man diese Plateau-Werte, die die Magnetisierungen in der Nähe des kritischen Punktes annehmen, so skaliert, daß die Plateaus für alle  $L$  den gleichen Wert haben. Mit diesen Abschätzungen wird ein sehr guter Datenkollaps für die beiden Magnetisierungen erzielt. In den Abb. 5.21–5.24 sind die Master-Kurven für die jeweils kleinste und größte Schichtdicke für  $m$  und  $m_1$  dargestellt. Die numerischen Werte der ermittelten Exponenten sind in Tab. 5.4 und Tab. 5.5 zusammengetragen.

Für die Suszeptibilitäten ist die Qualität der Rohdaten geringer. Auch sind die Plateaus, die für eine erste Abschätzung von  $w$  herangezogen wurden, nicht so deutlich ausgeprägt. Somit wurde bei der Bestimmung von  $w$  eine deutlich geringere Genauigkeit erreicht, die auch einen wesentlich schlechteren Datenkollaps zur Folge hat. Die Master-Kurven für die kleinste Dicke und für eine größere sind sowohl für  $\chi$  als auch  $\chi_{11}$  in den Abb. 5.25–5.28 wiedergegeben. Die Exponenten sind in Tab. 5.4 und Tab. 5.5 zusammengestellt.

Die in Tab.5.4 zusammengestellten, auf der Basis der Observablen  $m$  und  $\chi$  bestimmten effektiven Exponenten  $\nu_{\text{eff}}$ ,  $\beta_{\text{eff}}$  und  $\gamma_{\text{eff}}$  nähern sich für die kleinste untersuchte Schichtdicke  $D = 4$  dem Wert des zweidimensionalen Ising-Modells an. Mit steigender Dicke nimmt der Wert von  $\beta_{\text{eff}}$  zu, bleibt jedoch auch für die größte untersuchte Schichtdicke  $D = 32$  deutlich kleiner als der Wert im dreidimensionalen Ising-Modell. Im Gegensatz dazu bleibt der effektive Exponent  $\gamma_{\text{eff}}$  nahezu konstant in der Nähe des zweidimensionalen Wertes  $\gamma_2$ , wobei die geringere Genauigkeit der Daten beachtet werden muß. Der dritte effektive Exponent,  $\nu_{\text{eff}}$ , zeigt ein überraschendes nicht-monotones Verhalten, welches auch bei Systemen mit freien Oberflächen beobachtet wurde [104].

Die Bestimmung der effektiven Exponenten mittels der Observablen  $m_1$  und  $\chi_{11}$  führte im Falle von  $\beta_{\text{eff}}$  zu einem gleichen Verhalten. Jedoch ist der Anstieg der Werte mit steigender Schichtdicke ausgeprägter. Auch  $\gamma_{\text{eff}}$  zeigte eine deutliche Drift hin zu kleineren Werten bei ansteigender Dicke.

$D$	$u$	$v$	$w$	$\beta_{\text{eff}}$	$\gamma_{\text{eff}}$	$\nu_{\text{eff}}$
2d	1	1/8	7/4	1/8	7/4	1
4	0.956	0.126	1.67	0.132	1.75	1.064
8	1.018	0.136	1.72	0.133	1.69	0.982
12	0.944	0.138	1.67	0.146	1.77	1.059
16	0.938	0.139	1.61	0.148	1.72	1.066
24	0.898	0.145	1.53	0.161	1.70	1.114
28	0.853	0.141	1.48	0.165	1.74	1.172
32	0.884	0.155	1.54	0.175	1.74	1.131
$\infty$	1.587	0.518	1.96	0.327	1.24	0.630

**Tab. 5.4:** Werte der effektiven Exponenten für die Gesamtmagnetisierung  $m$  und die magnetische Suszeptibilität des Volumens  $\chi$  für die einzelnen Schichtdicken. Zum Vergleich sind die exakten Exponenten des 2d-Ising-Modells und die Literaturwerte für den dreidimensionalen Fall angegeben.

$D$	$u$	$v$	$w$	$\beta_{\text{eff}}$	$\gamma_{\text{eff}}$	$\nu_{\text{eff}}$
2d	1	1/8	7/4	1/8	7/4	1
4	0.956	0.126	1.63	0.131	1.71	1.064
8	1.018	0.138	1.74	0.136	1.61	0.982
12	0.944	0.137	1.45	0.145	1.54	1.059
16	0.938	0.142	1.31	0.151	1.40	1.066
24	0.898	0.163	1.21	0.181	1.35	1.114
28	0.853	0.165	1.23	0.158	1.44	1.172
32	0.884	0.183	1.27	0.208	1.44	1.131
$\infty$	1.587	1.263	-0.52	0.796	-0.33	0.630

**Tab. 5.5:** Werte der effektiven Exponenten für die Oberflächenmagnetisierung  $m_1$  und die magnetische Suszeptibilität  $\chi_{11}$  für die einzelnen Schichtdicken. Zum Vergleich sind die exakten Exponenten des 2d-Ising-Modells und die Literaturwerte für  $\beta_1$  bzw.  $\gamma_{11}$  im dreidimensionalen Fall angegeben [95].

## 5.5 Profile

In diesem Abschnitt sollen Profile für die Systeme mit Randmagnetfeld  $h_1 = -0.015$  bei ihrer kritischen Temperatur  $T_c(D)$  diskutiert werden. Der Begriff “Profil” soll hier als Bezeichnung für eine Funktion dienen, die eine Observable in Abhängigkeit von ihrem Abstand zur Wand,  $z$ , beschreibt. Aus Symmetriegründen reicht es zur vollständigen Beschreibung aus, die Funktion auf dem Intervall von  $z = 1$  bis zur halben Schichtdicke  $z = D/2$  zu betrachten. Für die an die Wand grenzende Schicht wurde die Abstands-Koordinate  $z = 1$  festgelegt. Gegenstand der Untersuchung in diesem Kapitel sind das Profil der Energie  $u$  und der Magnetisierung  $m$  pro Teilchen. Dabei wurden für die Magnetisierung drei verschiedene Observablen betrachtet. Die Magnetisierung der positiven Phase  $m^+$ , in die nur Konfigurationen eingingen, welche in allen Schichten eine positive Magnetisierung aufwiesen, die Magnetisierung der negativen Phase  $m^-$ , in die entsprechend alle negativ magnetisierten Konfigurationen eingingen und Magnetisierung  $m$ , die als Mittelwert aus  $m^+$  und  $m^-$  berechnet wird.

In Abb. 5.29 ist das Magnetisierungsprofil der positiven Phase für verschiedene Dicken  $D$  dargestellt. Um den Vergleich zwischen den einzelnen Dicken zu erleichtern, wurde hier, und ebenso in den restlichen Abbildungen, der Abstand von der Wand  $z$  mit  $D$  normiert. Bei diesem Profil sind die Werte der Magnetisierung im Bereich der Wand deutlich gegenüber den Werten im Bulk abgesenkt. Dies ist darauf zurückzuführen, daß die Teilchen in der Oberfläche nur fünf anstatt sechs benachbarte Teilchen haben. Das ist vergleichbar mit einer reduzierten effektiven Wechselwirkungskonstante  $J$  bei vollständiger Anzahl von Nachbarn. Eine reduzierte Wechselwirkungskonstante ist aber äquivalent zu einer höheren Temperatur, was zu einer niedrigeren Magnetisierung führt. Dieser Mechanismus wird auch als *missing neighbor effect* bezeichnet.

Der Betrag der Magnetisierung nimmt mit steigender Dicke ab, was auf den damit verbundenen Anstieg der Teilchenzahl zurückzuführen ist.

Analog gilt für das Magnetisierungsprofil der negativen Phase (siehe Abb. 5.30), daß die Magnetisierung im Wandbereich gegenüber den Bulk-Werten erhöht ist, was einer Absenkung des Betrages entspricht. Der Betrag der Magnetisierung zeigt im wesentlichen die gleiche Systematik wie beim Profil  $m^+$ .

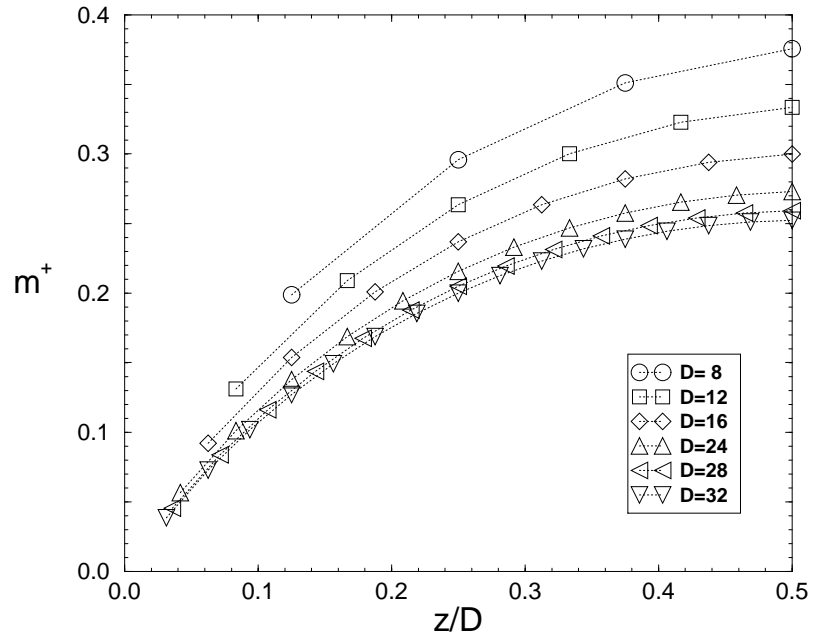


Abb. 5.29: Magnetisierungsprofile der positiven Phase bei  $T_c(D)$  für verschiedene  $D$ .

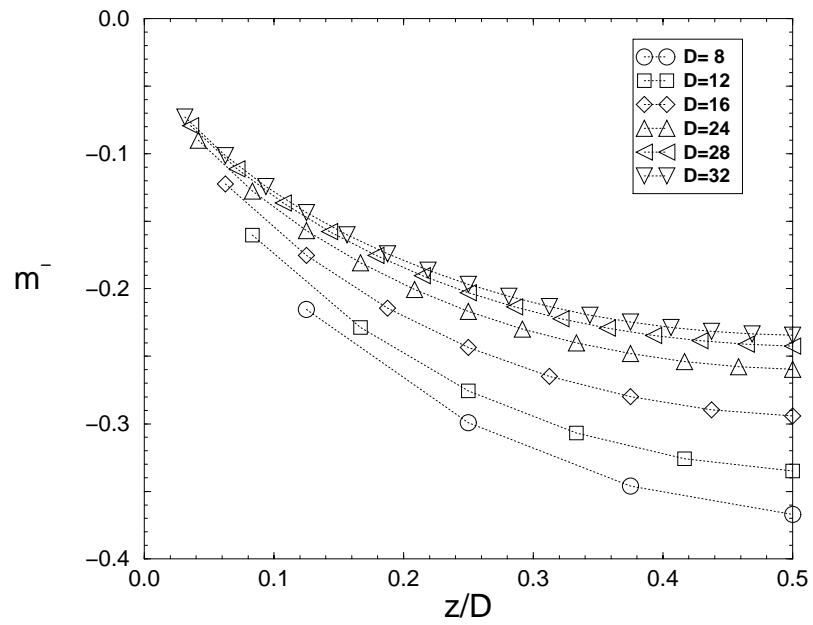


Abb. 5.30: Magnetisierungsprofile der negativen Phase bei  $T_c(D)$  für verschiedene  $D$ .

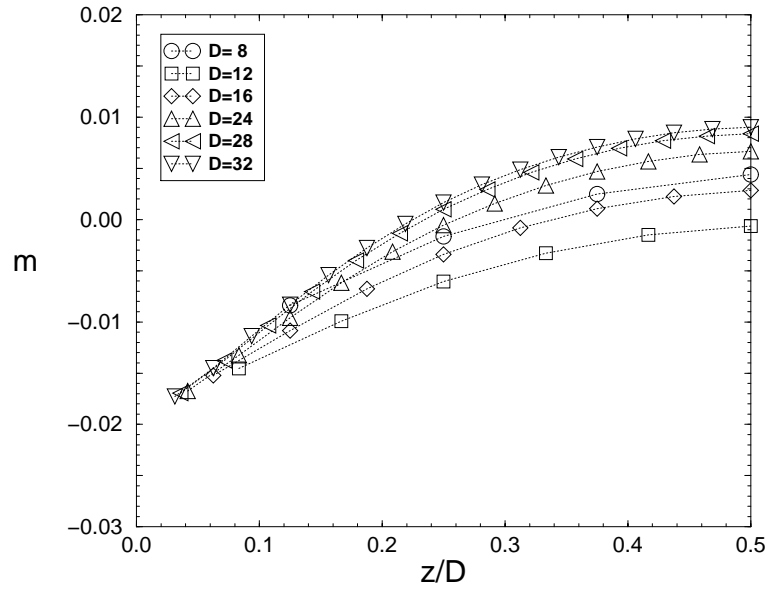


Abb. 5.31: Magnetisierungsprofile des Mittelwertes aus positiver und negativer Phase.

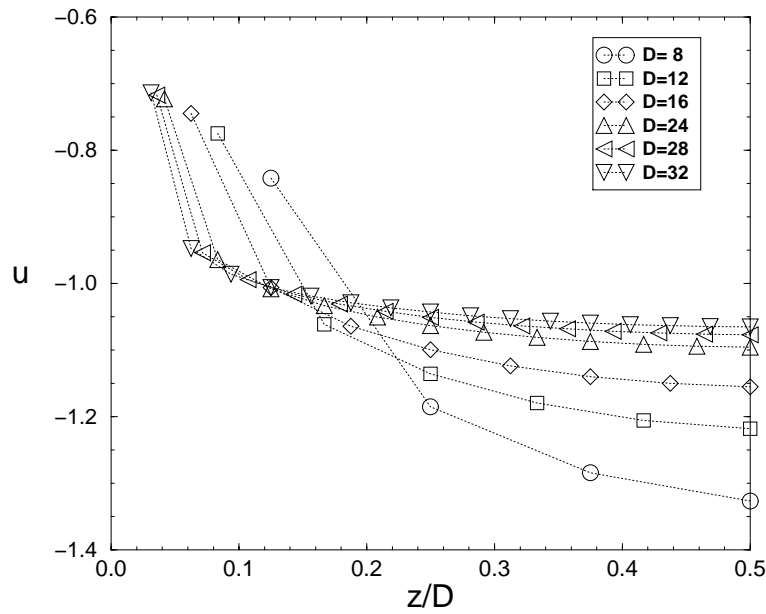


Abb. 5.32: Energieprofile bei  $T_c(D)$  für verschiedene  $D$ .

Der quantitative Unterschied zwischen diesen Profilen wird erst durch die Betrachtung des Mittelwertes  $m$  über die Kurven  $m^+$  und  $m^-$  deutlich (siehe Abb. 5.31). An den Wänden bleibt eine negative Magnetisierung über, die einer positiven des Bulks gegenüber steht. Dieses Verhalten korrespondiert zu den Vorzeichen der anliegenden Magnetfelder. Die Abbildung offenbart die Asymmetrie der Profile, die besonders im Bereich der Oberflächen ausgeprägt ist, wo das anliegende Magnetfeld viel größer als im Volumen ist.

Bei allen betrachteten Magnetisierungsprofilen tritt ihre Inhomogenität deutlich zutage.

Diese Inhomogenität läßt sich auch bei den Energieprofilen beobachten. In Abb. 5.32 ist die Gesamtenergie für verschiedene Schichtdicken dargestellt. Bei allen Dicken ist ein rapider Anstieg der Energie beim Übergang von der Schicht  $z = 2$  auf die der Wand benachbarten Schicht  $z = 1$  erkennbar.

# Kapitel 6

## Systeme mit freien Oberflächen

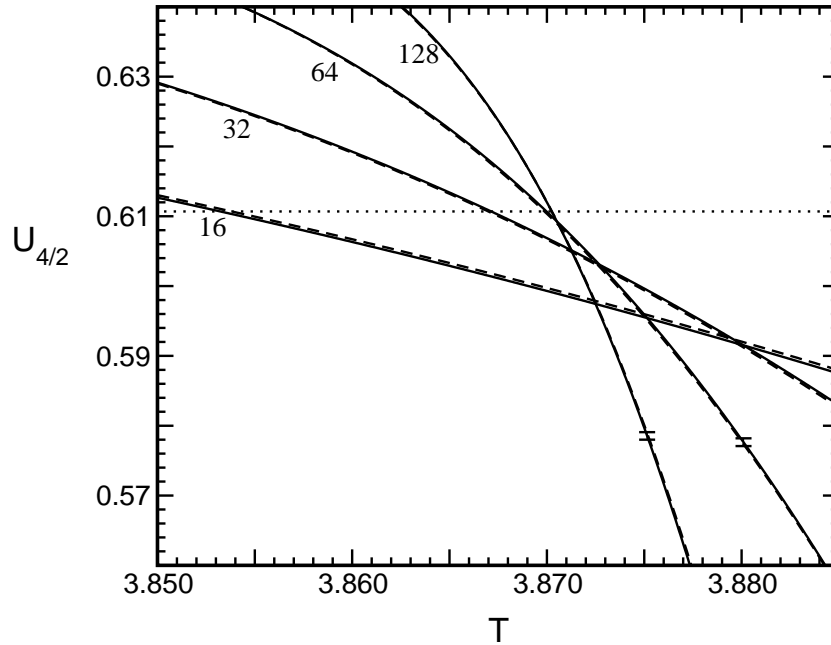
In diesem Kapitel werden die Ergebnisse für die Systeme mit freien Oberflächen dargestellt; es gilt  $h_1 = 0$ . Das Erzeugen von Daten im Bereich der kritischen Punkte wurde stark erleichtert durch die Tatsache, daß die Koexistenzkurve aus Symmetriegründen mit der Temperaturachse zusammenfällt und der Parameterraum, in dem diese Punkte bestimmt werden mußten, somit de facto eindimensional ist. Auch die Analyse der Daten wurde durch die entfallende Bestimmung der Koexistenzkurve vereinfacht.

### 6.1 Kumulanten

Analog zum Kap. 5 beginne ich mit der Diskussion der Kumulanten für die untersuchten Schichtdicken. In den Abb. 6.1–6.7 sind die Ergebnisse für die Kumulanten der verschiedenen Systemgrößen zusammengestellt. In den Abbildungen können die einzelnen Systemgrößen anhand der angegebenen lateralen Ausdehnung  $L$  leicht identifiziert werden.

Um einen direkten Vergleich zu den Systemen mit vorhandenem Oberflächenfeld zu ermöglichen, sind deren Kumulanten ebenfalls in die Abbildungen aufgenommen. Sie werden mit einer unterbrochenen Linie dargestellt.

In bezug auf die Fehlerbalken wurde die gleiche Vorgehensweise wie im Kap. 5.1 gewählt. Auf eine Darstellung der Fehler der Vergleichskurven wurde aus Gründen der Übersichtlichkeit verzichtet. Ihre Größenordnung kann den Abb. 5.1–5.7 entnommen werden. Aufgrund der geringeren Autokorrelationen im feldfreien Fall sind die

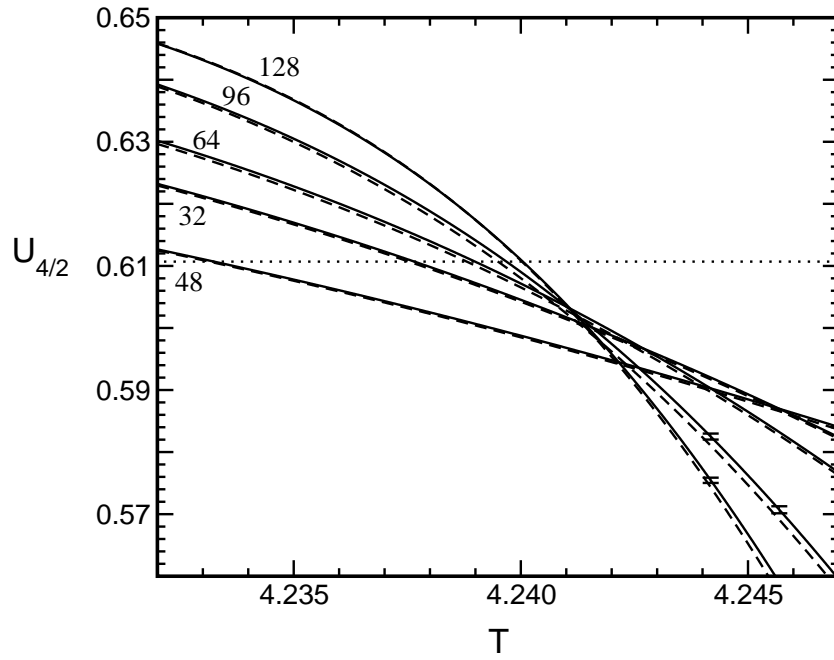


**Abb. 6.1:** Kumulanten für  $D = 4$  bei freien Oberflächen. Die gepunktete Linie gibt den Wert von  $U^*$  an, die unterbrochenen Linien die Kumulanten bei vorhandenem Oberflächenmagnetfeld.

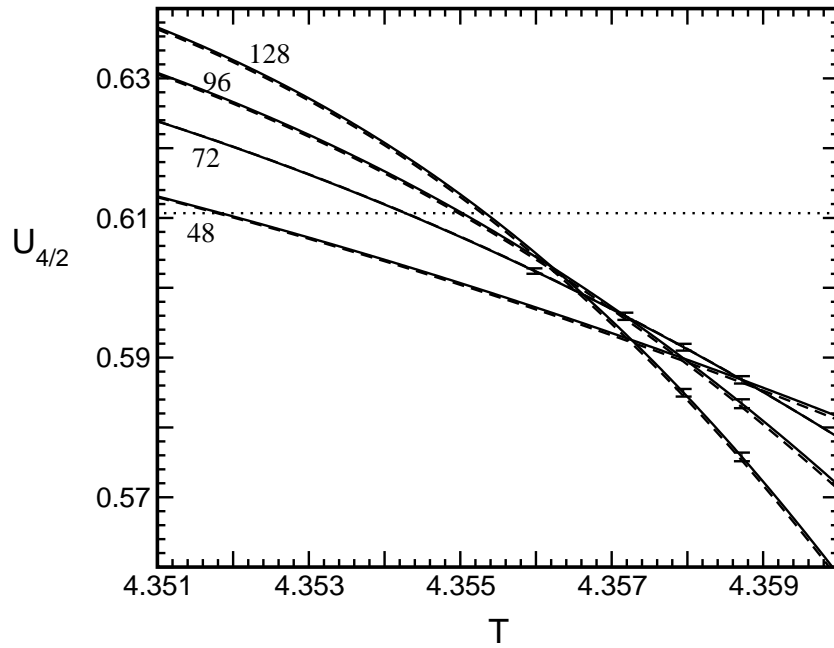
Fehlerbalken deutlich kleiner als für die Systeme mit Oberflächenfeld, da die Länge des Simulationslaufes in etwa gleich war.

Beim Betrachten der Abbildungen fällt auf, daß sich die Kumulanten im unteren Teil des abgebildeten Temperaturbereiches von den Vergleichskurven der Systeme mit Oberflächenfeld kaum unterscheiden. Erst bei höheren Temperaturen wird eine Differenzierung der Kurven deutlich. Diese wird mit steigender Schichtdicke ausgeprägter. Die Kurven für den feldfreien Fall scheinen dann systematisch zu höheren Temperaturen hin verschoben. Dieser Effekt übersteigt allerdings kaum die durch die Fehler der Kumulanten vorgegebenen Intervalle. Dieses Verhalten belegt, daß die Bezeichnung *schwaches* Feld, die im Kap. 5 ad hoc für das Oberflächenfeld der Stärke  $h_1 = -0.015$  eingeführt wurde, ihre Berechtigung hat.

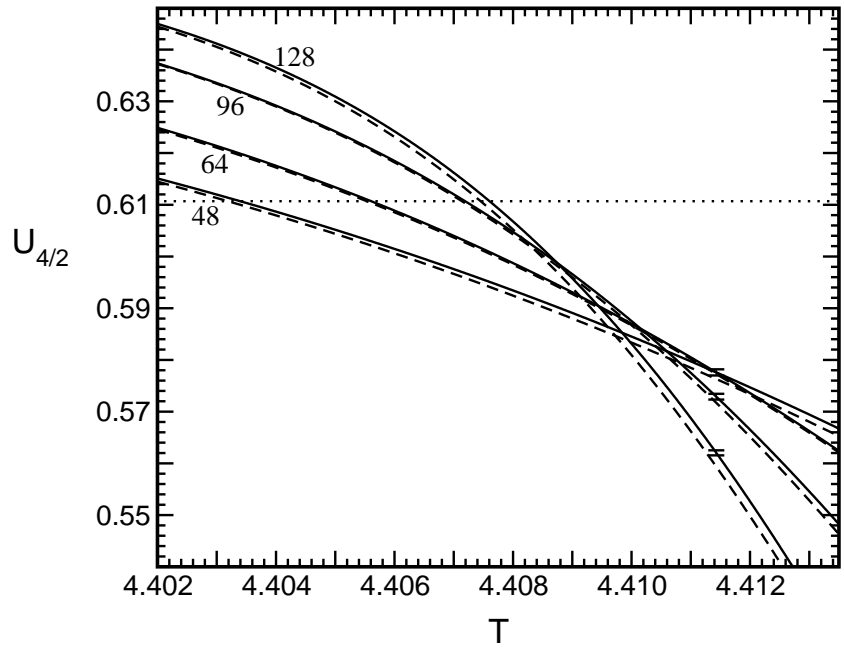
Die Abfolge der Schnittpunkte der Kumulanten untereinander zeigt die schon in Kap. 5.1 beschriebene Systematik. Auch die Abstände der Kumulantenwerte am Überkreuzungspunkt vom Fixpunktwert  $U^*$ , welcher in den Abbildungen als gepunktete Linie dargestellt ist, zeigen das gleiche Verhalten wie im Fall der nicht verschwindenden Oberflächenfelder, was durch dieselbe Argumentation begründet werden kann.



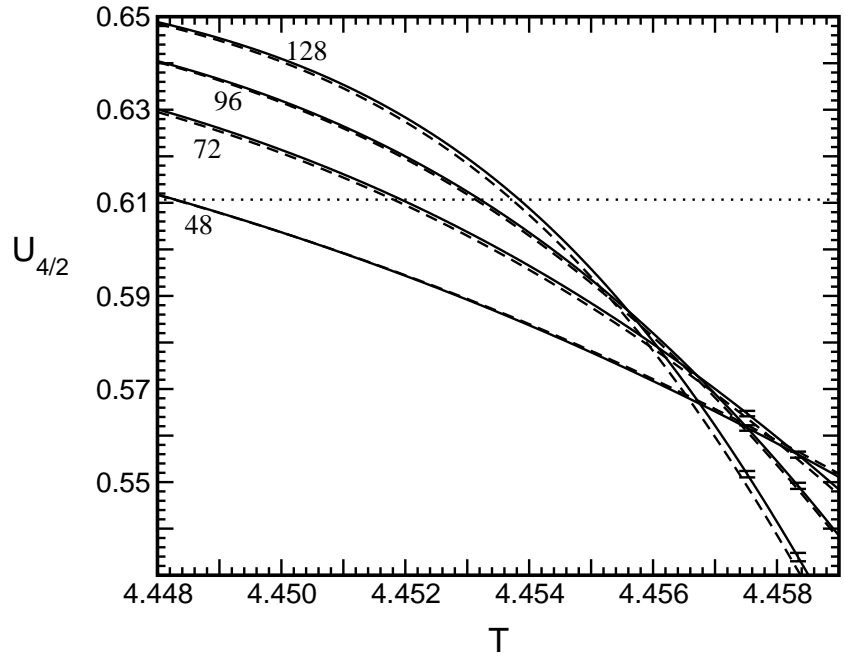
**Abb. 6.2:** Kumulanten für  $D = 8$  bei freien Oberflächen. Die gepunktete Linie gibt den Wert von  $U^*$  an, die unterbrochenen Linien die Kumulanten bei vorhandenem Oberflächenmagnetfeld.



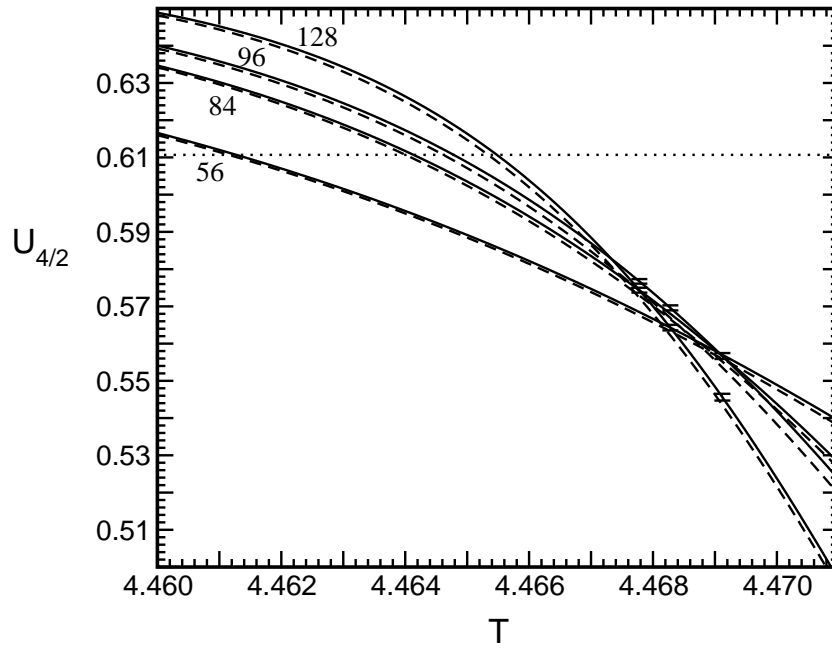
**Abb. 6.3:** Kumulanten für  $D = 12$  bei freien Oberflächen. Die gepunktete Linie gibt den Wert von  $U^*$  an, die unterbrochenen Linien die Kumulanten bei vorhandenem Oberflächenmagnetfeld.



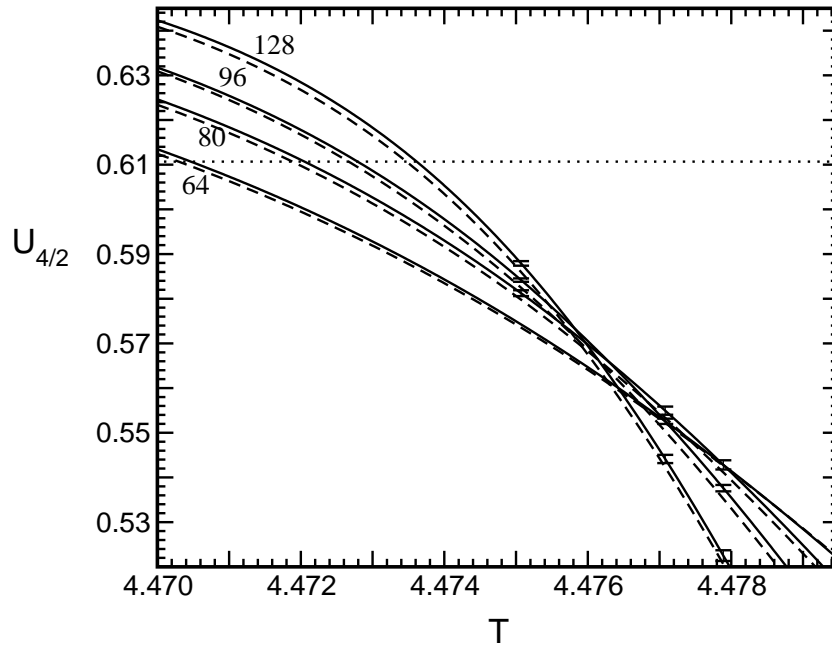
**Abb. 6.4:** Kumulanten für  $D = 16$  bei freien Oberflächen. Die gepunktete Linie gibt den Wert von  $U^*$  an, die unterbrochenen Linien die Kumulanten bei vorhandenem Oberflächenmagnetfeld.



**Abb. 6.5:** Kumulanten für  $D = 24$  bei freien Oberflächen. Die gepunktete Linie gibt den Wert von  $U^*$  an, die unterbrochenen Linien die Kumulanten bei vorhandenem Oberflächenmagnetfeld.

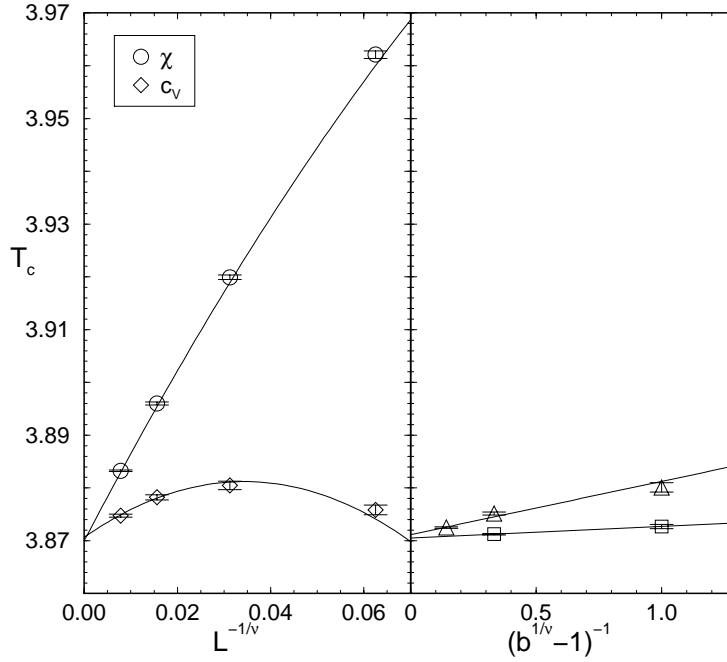


**Abb. 6.6:** Kumulanten für  $D = 28$  bei freien Oberflächen. Die gepunktete Linie gibt den Wert von  $U^*$  an, die unterbrochenen Linien die Kumulanten bei vorhandenem Oberflächenmagnetfeld.



**Abb. 6.7:** Kumulanten für  $D = 32$  bei freien Oberflächen. Die gepunktete Linie gibt den Wert von  $U^*$  an, die unterbrochenen Linien die Kumulanten bei vorhandenem Oberflächenmagnetfeld.

## 6.2 Bestimmung der kritischen Temperaturen



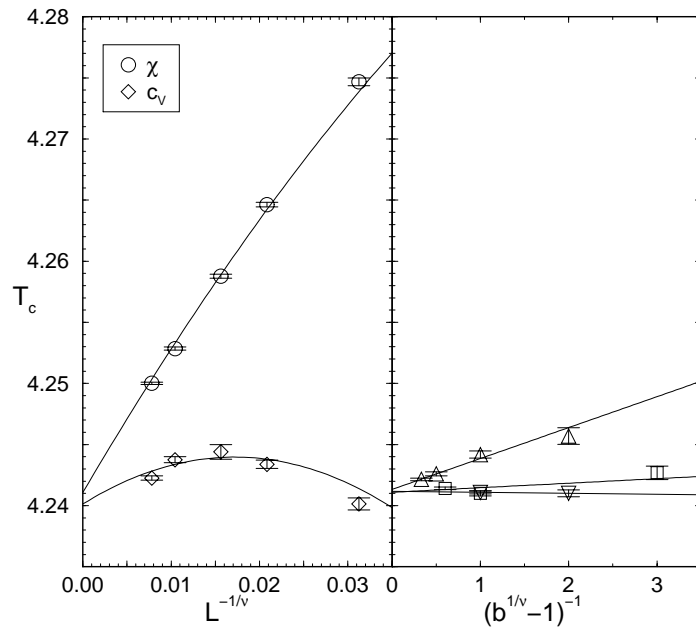
**Abb. 6.8:** Bestimmung von  $T_c$  für  $D = 4$  bei freien Oberflächen. Im linken Bild mit dem Maximum von spezifischer Wärme bzw. magnetischer Suszeptibilität, im rechten Bild mittels der Kumulantenschnittpunkte.

Die Bestimmung der kritischen Punkte  $(T_c(D), h_c(D))$  im Falle verschwindender Oberflächenfelder wurde analog zur Vorgehensweise in Kap. 5.2 durchgeführt.

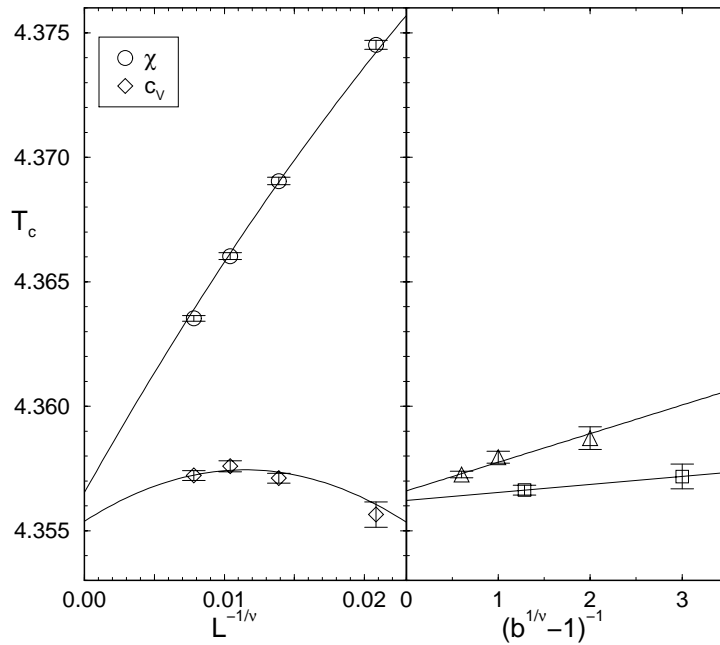
Die Grundlage der Untersuchung der kritischen Punkte bilden die Skaleneigenschaften dreier Observablen (vgl. Kap. 3.3.1 und 3.3.2):

- das Skalieren der Maxima der spezifischen Wärme mit der Seitenlänge  $L$ ,  $T_{c_v}^{max} - T_c(D) \propto L^{-1/\nu}$ ,
- das Skalieren der Maxima der magnetischen Suszeptibilität mit der Seitenlänge,  $T_{\chi}^{max} - T_c(D) \propto L^{-1/\nu}$ ,
- und das Skalieren der Schnittpunkte der Kumulanten  $U_{4/2}$  mit dem Verhältnis ihrer Seitenlängen  $b$ ,  $T_x \propto (b^{-1/\nu} - 1)^{-1}$ .

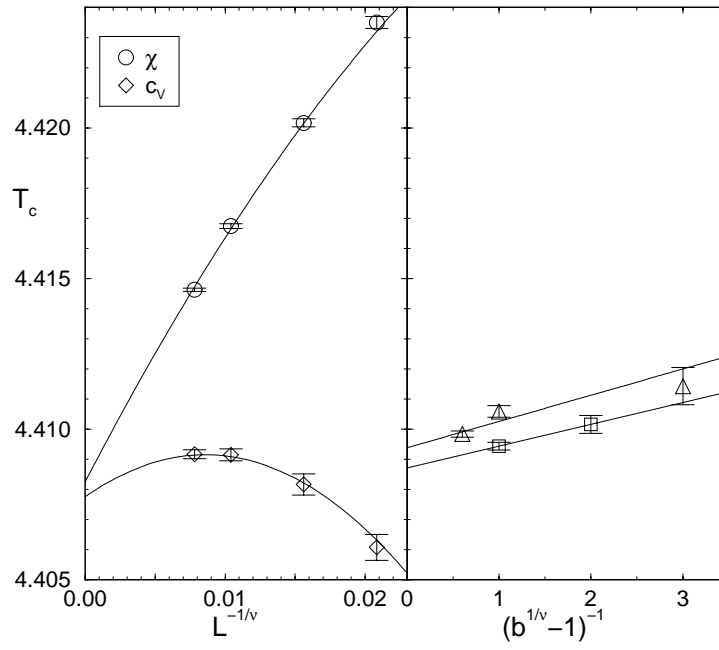
Die skalierten Observablen sind für die einzelnen Dicken  $D$  in Abb. 6.8–6.14 dargestellt.



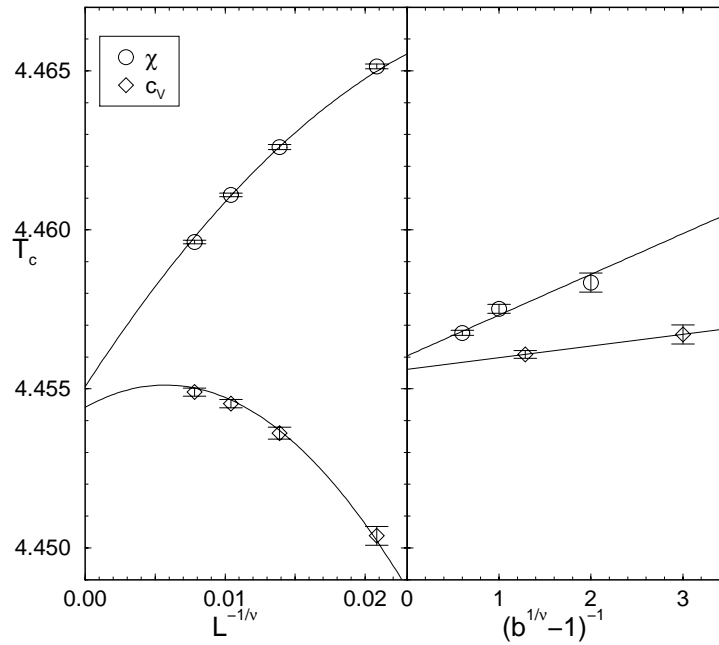
**Abb. 6.9:** Bestimmung von  $T_c$  für  $D = 8$  bei freien Oberflächen. Im linken Bild mit dem Maximum von spezifischer Wärme bzw. magnetischer Suszeptibilität, im rechten Bild mittels der Kumulantenschnittpunkte.



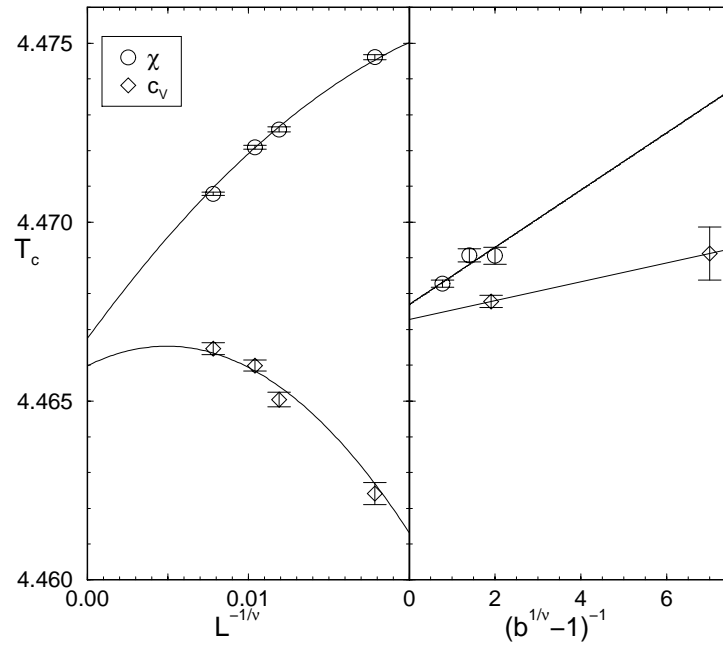
**Abb. 6.10:** Bestimmung von  $T_c$  für  $D = 12$  bei freien Oberflächen. Im linken Bild mit dem Maximum von spezifischer Wärme bzw. magnetischer Suszeptibilität, im rechten Bild mittels der Kumulantenschnittpunkte.



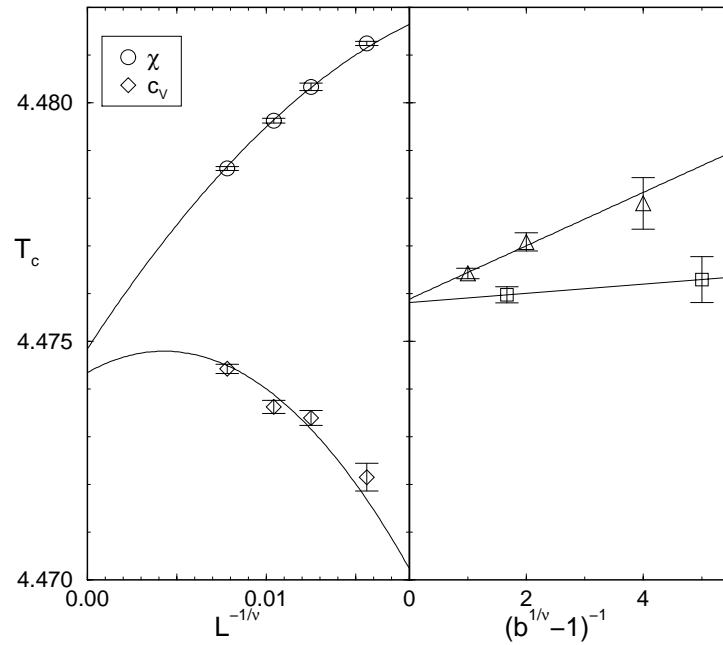
**Abb. 6.11:** Bestimmung von  $T_c$  für  $D = 16$  bei freien Oberflächen. Im linken Bild mit dem Maximum von spezifischer Wärme bzw. magnetischer Suszeptibilität, im rechten Bild mittels der Kumulantenschnittpunkte.



**Abb. 6.12:** Bestimmung von  $T_c$  für  $D = 24$  bei freien Oberflächen. Im linken Bild mit dem Maximum von spezifischer Wärme bzw. magnetischer Suszeptibilität, im rechten Bild mittels der Kumulantenschnittpunkte.



**Abb. 6.13:** Bestimmung von  $T_c$  für  $D = 28$  bei freien Oberflächen. Im linken Bild mit dem Maximum von spezifischer Wärme bzw. magnetischer Suszeptibilität, im rechten Bild mittels der Kumulantenschnittpunkte.



**Abb. 6.14:** Bestimmung von  $T_c$  für  $D = 32$  bei freien Oberflächen. Im linken Bild mit dem Maximum von spezifischer Wärme bzw. magnetischer Suszeptibilität, im rechten Bild mittels der Kumulantenschnittpunkte.

In allen drei Skalenrelationen wurden der zweidimensionale kritische Ising-Exponent  $\nu = 1$  verwendet.

An die in der rechten Teilabbildung eingezeichneten Werte der Kumulantenschnittpunkte wurden Geraden angepaßt und durch Extrapolation  $b \rightarrow \infty$  drei Schätzwerte für die kritische Temperatur bestimmt.

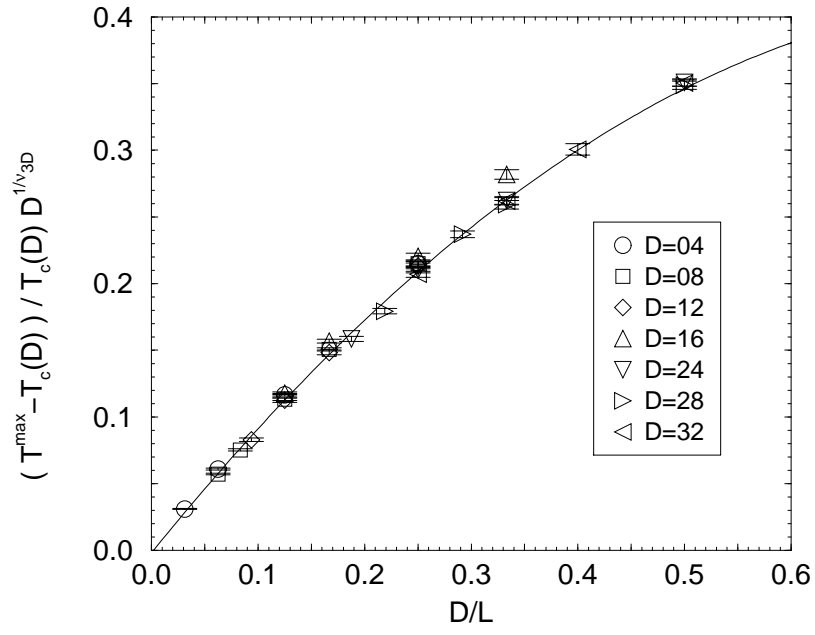
Das Verhalten der in der linken Teilabbildung dargestellten Werte der Maxima von spezifischer Wärme und magnetischer Suszeptibilität ist im Gegensatz dazu nicht linear. Besonders die spezifische Wärme zeigt eine deutlich ausgeprägte Krümmung. Wegen der geringen Anzahl der Datenpunkte muß jedoch von der Anpassung eines Polynoms höherer Ordnung Abstand genommen werden.

Entsprechend der Vorgehensweise in Kap. 5.2 wurde wieder durch eine geeignete Skalierung der Werte versucht, einen Kollaps der Maxima für die verschiedenen Systemgrößen auf eine Master-Kurve zu erreichen, an die dann ein Polynom höhere Ordnung angepaßt werden kann. Der gewünschte Datenkollaps läßt sich durch eine Auftragung von  $D^{1/\nu} (T - T_c(D))/T_c(D)$  gegenüber  $D/L$  erzielen. Da in dem für die Skalierung verwendeten Ausdruck die gesuchte kritische Temperatur selbst auftritt, kann sie nur im Zuge des Iterationsverfahrens, welches im Kap. 5.2 ausführlich beschrieben wurde, bestimmt werden.

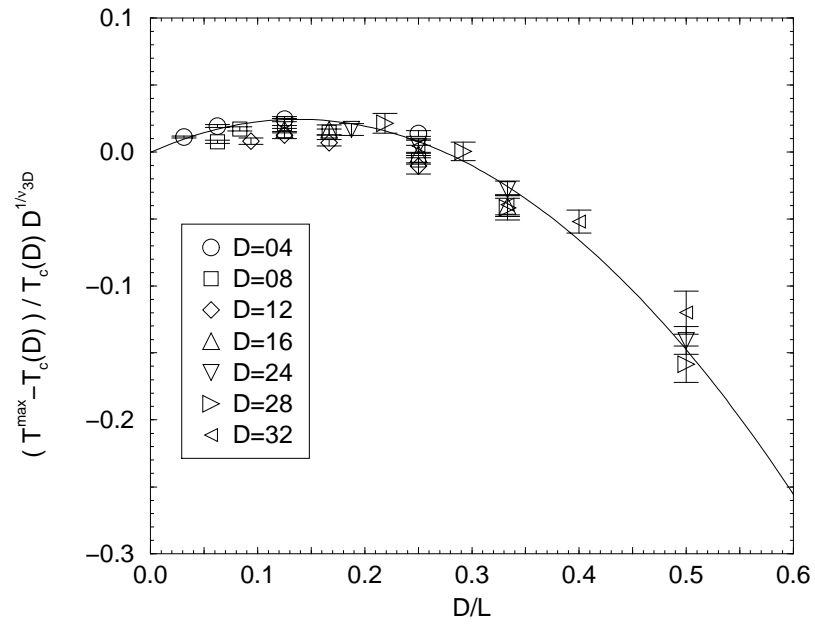
Als Zwischenergebnis des Iterationsverfahrens ergaben sich zwei Polynome zweiten Grades der Form  $y = a + bx + cx^2$  mit den Parametern  $a = -0.000(2)$ ,  $b = 0.990(7)$  und  $c = -0.557(15)$  im Falle der Suszeptibilität,  $a = -0.000(1)$ ,  $b = 0.359(16)$  und  $c = -1.307(47)$  im Falle der spezifischen Wärme. Im Rahmen der Genauigkeit entsprechen diese Skalenkurven denen der im Kap. 3.3.1 untersuchten Systeme mit Oberflächenmagnetfeld. Diese Kurven wurden dann in die Abb. 6.8–6.14 aufgenommen und der Ordinatenabschnitt der Parabeln noch “von Hand” justiert. Die mit Hilfe der so entstandenen Graphen gewonnenen Schätzwerte für  $T_c(D)$  wurden nochmals zur Skalierung der Rohdaten herangezogen. Die so entstandenen Graphen (siehe Abb. 6.15 und 6.16) zeigen einen deutlichen Datenkollaps und rechtfertigen somit die Vorgehensweise. Auch hier wurde schon nach einer einzigen Iteration eine gute Konsistenz erreicht, so daß auf weitere Iterationen verzichtet werden konnte

Aus den durch diese Methodik gewonnenen vier Werte für jedes  $T_c(D)$  wurde in analoger Vorgehensweise zum vorherigen Kapitel der Mittelwert gebildet. In der Tabelle 6.1 sind die kritischen Temperaturen für die einzelnen Dicken zusammengestellt.

Anders als für den Fall der Schichten mit Randmagnetfeld reicht die Kenntnis der



**Abb. 6.15:** Master-Kurve für die magnetische Suszeptibilität  $\chi$ . Die durchgezogene Linie gibt die für die Analyse verwendete Kurve wieder.



**Abb. 6.16:** Master-Kurve für die spezifische Wärme. Die durchgezogene Linie gibt die für die Analyse verwendete Kurve wieder.

$D$	$T_c(D)$
4	3.8705(3)
8	4.2408(4)
12	4.3561(4)
16	4.4084(4)
24	4.4551(5)
28	4.4667(5)
32	4.4750(5)

**Tab. 6.1:** Werte für die kritischen Temperatur  $T_c(D)$  für verschiedene  $D$  bei freien Oberflächen.

kritischen Temperaturen zur vollständigen Spezifizierung der kritischen Punkte aus, da aus Symmetriegründen das Magnetfeld im Falle der Phasenkoexistenz verschwinden muß.

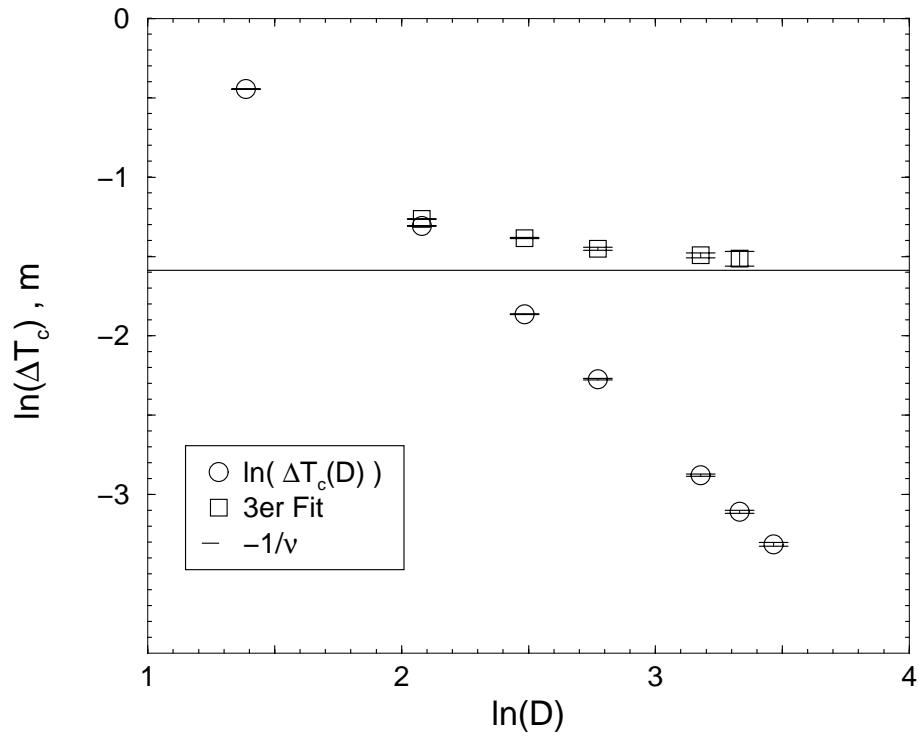
Durch die Bestimmung der kritischen Temperaturen für die einzelnen Schichtdicken sind nun auch für freie Oberflächen die Voraussetzungen geschaffen, die zur Untersuchung des Fisher-Nakanishi-Scalings erfüllt sein müssen.

### 6.3 Untersuchung des Fisher-Nakanishi-Scalings

In diesem Abschnitt soll die Gültigkeit der Fisher-Nakanishi-Skalenhypothese für die Systeme mit freien Oberflächen untersucht werden. Für diesen Spezialfall vereinfacht sich die Vorgehensweise gegenüber Kap. 5.3 in der Beziehung, daß eine Verschiebung des kritischen Magnetfeldes nicht auftritt. Somit reduziert sich die Untersuchung alleine auf die der Verschiebung der kritischen Temperatur gegenüber ihrem Wert im Volumen  $T_c^\infty$ . Analog zu Kap. 5.3 gilt  $\Delta T_c \propto D^{-1/\nu}$  und das negative Vorzeichen der Temperaturverschiebung ist in der Proportionalitätskonstante absorbiert,  $\Delta T_c = |T_c(D) - T_c^\infty|$ . Für  $T_c$  und  $-1/\nu$  werden die Werte verwandt, die im gleichen Kapitel erläutert wurden.

In Abb. 6.17 ist die Verschiebung  $\Delta T_c$  gegen die Schichtdicke doppelt-logarithmisch aufgetragen. Der Verlauf der Linie zeigt wie für die Systeme mit Randmagnetfeld eine leichte Krümmung, die darauf zurückzuführen ist, daß womöglich der asymptotische Bereich großer Schichtdicken für die untersuchten Systeme noch nicht erreicht ist. Aus diesem Grunde wurde an Gruppen von jeweils drei aufeinanderfolgenden Dicken eine Gerade gefittet. Die sich ergebenden Werte für die Geradensteigung sind in der Tabelle 6.2 zusammen mit den paarweisen Steigungen dargestellt. Beide Serien zeigen eine deutliche Drift der Werte, die sich aber mit steigender Dicke dem verwendeten Literaturwert  $-1/\nu = -1.587$  [94] annähern. Die mit dem gewichteten Fit erzielten Werte für die Gruppe mit den drei größten Dicken ist innerhalb der doppelten Standardabweichung mit dem Literaturwert verträglich. Für die paarweisen Steigungen ist die Aussagekraft der Ergebnisse für die größten Systeme wegen der ausgeprägten Fehlerintervalle beeinträchtigt.

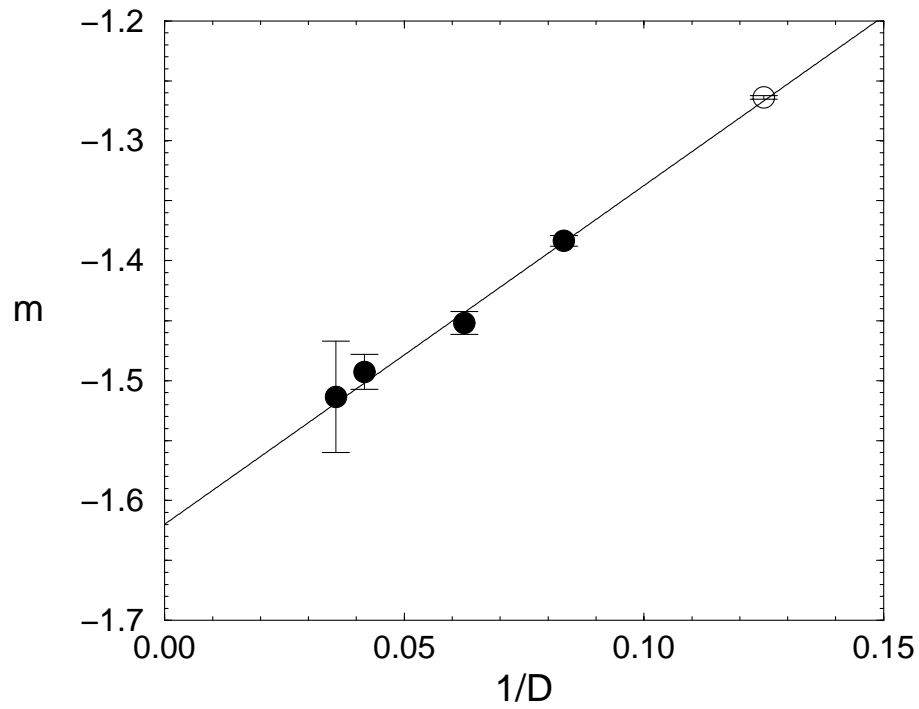
Auch hier legen die Ergebnisse die Vermutung nahe, daß für die untersuchten Schichtdicken der asymptotische Bereich noch nicht erreicht ist. Dies machte eine Extrapolation zu größeren Dicken hin notwendig. Dazu erfolgte eine Auftragung der Steigungen gegen  $D^{-1}$ , wie in Abb. 6.18 dargestellt. An die Daten wurde eine Gerade der Form  $y = a + bx$  angepaßt. Der Wert, in den das kleinste System  $D = 4$  eingeht, blieb dabei unberücksichtigt. Für die Parameter der Kurve ergaben sich  $a = -1.620(23)$  und  $b = 2.82(30)$ . Der Wert des Ordinatenabschnitts  $a$  entspricht dem Wert des Exponenten  $-1/\nu$  bei Extrapolation zu  $D \rightarrow \infty$ . Er ist innerhalb der doppelten Standardabweichung mit dem Literaturwert vereinbar. Die Fisher-Nakanishi-Skalenhypothese kann also im Rahmen dieser Extrapolation auch für den Spezialfall freier Oberflächen bestätigt werden.



**Abb. 6.17:** Doppelt-logarithmischer Plot von  $\Delta T_c(D)$  gegen  $D$  (Kreise). Die paarweisen Steigungen  $m$  (Quadrate) nähern sich mit steigendem  $D$  dem Wert  $-1/\nu$ .

Fit		Paarweise Steigungen	
$D$	$m$	$D$	$m$
4, 8,12	-1.264(2)	4, 8	-1.244(3)
8,12,16	-1.383(5)	8,12	-1.370(9)
12,16,24	-1.452(10)	12,16	-1.426(21)
16,24,28	-1.493(15)	16,24	-1.491(27)
24,28,32	-1.514(46)	24,28	-1.502(110)
		28,32	-1.533(164)
$-1/\nu \approx -1.587$ [94]			

**Tab. 6.2:** Drift des Exponenten in Abhängigkeit der Schichtdicke  $D$  im Fall freier Oberflächen.



**Abb. 6.18:** Auftragung der Werte für  $-1/\nu$  aus den 3er Fits gegen die inverse Schichtdicke  $D$ . Die Gerade ist ein linearer Fit an die durch gefüllte Symbole dargestellten Werte.

# Kapitel 7

## Zusammenfassung der Ergebnisse und Ausblick

In der hier vorliegenden Arbeit wurde das kritische Verhalten von dünnen Ising-Filmen mittels Monte Carlo-Simulationen untersucht. Die Ising-Filme sind einem Magnetfeld und einem Oberflächenfeld ausgesetzt, wobei letzteres den Einfluß der Wände auf das System modelliert. Besonderes Augenmerk wurde auf die Untersuchung der Fisher-Nakanishi-Skalenhypothese gerichtet, welche die Verschiebung der kritischen Punkte sowohl in der Temperatur als auch in der Magnetfeldstärke, bei der Phasenkoexistenz besteht, beschreibt. Diese Verschiebung erfolgt relativ zum kritischen Punkt eines Systems im Volumen. Um das in der Nähe eines kritischen Punktes auftretende und die Effizienz einer Simulation beeinträchtigende *critical slowing down* zu vermeiden, wurde als Simulationsalgorithmus ein Swendsen-Wang-Cluster-Algorithmus eingesetzt. Dieser wurde durch die Erweiterung um einen Ghostspin-Term in seiner Anwendbarkeit auf Systeme mit äußeren Feldern verallgemeinert. Dies ermöglichte erstmals einen genauen quantitativen Test der Vorhersage von Fisher und Nakanishi. Bei der Analyse der Daten wurden Multi-Histogramm-Methoden eingesetzt, die die statistischen Fehler reduzierten. Die Methodik wurde auf zwei Szenarien mit verschiedenen Werten für das Randmagnetfeld angewendet, zum einen für ein System mit feldfreien Oberflächen, d.h.,  $h_1 = 0$ , zum anderen für ein schwaches negatives Oberflächenmagnetfeld  $h_1 = -0.015$ .

## Zusammenfassung der Ergebnisse

Die Ergebnisse dieser Arbeit lassen sich in den folgenden Punkten zusammenfassen:

- Die Kumulante 4. Ordnung zeigt bei fester Schichtdicke  $D$  für verschiedene Gitterlängen  $L$  keinen gemeinsamen Schnittpunkt. Diese Abweichungen sind systematischer Natur. Es treten deutliche Korrekturen zum Skalenverhalten auf. Der Wert der Kumulanten im Bereich der einzelnen Schnittpunkte ist niedriger als der schwach universelle und somit von den Randbedingungen abhängige Wert für den Fixpunkt  $U^*$  im zweidimensionalen Ising-Modell. Die Kumulante zeigt somit kein reines 2d-Ising-Verhalten im Bereich des kritischen Punktes. Dieses Ergebnis hat für beide Werte des Oberflächenmagnetfeldes Gültigkeit.
- Zur Bestimmung der kritischen Temperatur  $T_c(D)$  wurden mit der Extrapolation aus den Maxima der Suszeptibilität und der spezifischen Wärme, sowie der Extrapolation aus den Kumulantenschnittpunkten drei unabhängige Verfahren angewendet. Da insbesondere bei Extrapolation der Maxima der spezifischen Wärme die angepaßte Kurve eine deutliche Krümmung aufwies, wurde hier ein neuer Weg der Datenanalyse beschritten. Eine Auftragung von  $D/L$  gegen den Ausdruck  $(T - T_c)/T_c D^{1/\nu}$  führte zu einem Kollaps der Maxima für verschieden  $D$  auf eine gemeinsame Master-Kurve, die durch einen aussagekräftigen nichtlinearen Fit mit guter Genauigkeit bestimmt werden konnte. Somit war eine zuverlässige Extrapolation auch bei den mit Korrekturen behafteten Maxima-Daten gewährleistet. Die kritischen Temperaturen konnten infolge dieser Vorgehensweise für die untersuchten Schichtdicken mit einer Genauigkeit unter 1 Promille bestimmt werden. Diese Präzision wurde für beide Werte des Randmagnetfeldes  $h_1$  erreicht.
- Der Verlauf der Phasenkoexistenzkurve im Bereich der kritischen Temperatur wurde unter Verwendung von Multi-Histogramm-Techniken für den nicht-trivialen Fall des Randmagnetfeldes  $h_1 = -0.015$  lokalisiert und daraus mit Hilfe der kritischen Temperaturen die Werte des Koexistenzmagnetfeldes  $h_c(D)$  am kritischen Punkt bestimmt. Die dabei erreichte Auflösung liegt im Prozentbereich.
- Es wurde die Skalenhypothese von Fisher und Nakanishi untersucht, die die Verschiebung des kritischen Punktes eines dünnen Ising-Films gegenüber dem

eines Bulk-Systems in Abhängigkeit von der Schichtdicke beschreibt. Die exakte Gültigkeit der Hypothese ist eingeschränkt auf den Bereich der Systemgeometrien, in dem der Limes  $L/D \rightarrow \infty$  erreicht wird. In diesen Bereich konnte im Rahmen dieser Untersuchung nicht vorgedrungen werden. Dies machte eine Extrapolation hin zu größeren Dicken notwendig, da für die größten in dieser Arbeit untersuchten Systeme mit  $D = 28$  und  $D = 32$  ein asymptotisches Verhalten noch nicht erreicht wurde. Für die Verschiebung der kritischen Temperatur konnte das asymptotische Verhalten  $\Delta T_c \propto D^{-1/\nu}$  für beide Randmagnetfelder bestätigt werden. Auch für die Verschiebung des Koexistenzfeldes konnte für  $h_1 = -0.015$  die Hypothese  $\Delta h_c \propto D^{-(\Delta-\Delta_1)/\nu}$  verifiziert werden. Unter der Annahme, daß das Fisher-Nakanishi-Scaling gültig ist, ergab sich mit  $\Delta_1 = 0.459(13)$  ein neuer Schätzwert des Gap-Exponenten der Oberfläche, der sehr gut mit den Literaturwerten [95] aus Reihenentwicklungen übereinstimmt.

- In der Umgebung des kritischen Punktes wurde das Verhalten der Observablen Magnetisierung  $m$ , Oberflächenmagnetisierung  $m_1$ , magnetische Suszeptibilität und Suszeptibilität der Oberflächen untersucht. Bei Variation in der Temperatur entlang der Phasenkoexistenzlinie zeigten die Observablen weder ein reines 2d- noch ein reines 3d-Ising-Verhalten. Deshalb wurde durch die Anpassung von effektiven Exponenten versucht, zu einer Beschreibung dieses Crossover-Verhaltens zu gelangen. Für große Verhältnisse  $L/D$  nähern sich diese effektiven Exponenten stark an die Werte der 2d-Ising-Exponenten an.
- Es wurden Magnetisierungs- und Energieprofile in der Nähe des kritischen Punktes für alle untersuchten Schichtdicken aufgenommen, die als Basis einer zukünftigen Analyse zur Verfügung stehen.

## Ausblick

Die in dieser Arbeit entwickelten Techniken und Methoden bieten das Rüstzeug, um die Arbeit in viele Richtungen fortzuführen:

- Durch Anwendung stärkerer Oberflächenmagnetfelder wäre eine Ausdehnung der Untersuchung auf Wetting-Phänomene möglich.
- Durch Einführung langreichweitiger Oberflächenkräfte könnte deren Einfluß auf die Benetzungspänomene untersucht werden.

- Durch die Wahl zweier entgegengesetzt orientierten Oberflächenfelder könnten Grenzflächen-Lokalisierungs-Delokalisierungsübergänge studiert werden.

# Literaturverzeichnis

- [1] D.T.Jacobs, R.C.Mockler und W.J.O'Sullivan: Phys.Rev.Lett. **37** (1976) 1471
- [2] B.A.Scheibner, M.R.Meadows, R.C.Mockler und W.J.O'Sullivan: Phys.Rev.Lett. **43** (1979) 590
- [3] M.R.Meadows, B.A.Scheibner, R.C.Mockler und W.J.O'Sullivan: Phys.Rev.Lett. **43** (1979) 592
- [4] C.G.V.Burgess, D.H.Everett, S.Nuttal: Pure Appl.Chem. **61** (1989) 1845
- [5] W.D.Machin: Langmuir **10** (1994) 1235; Langmuir **15** (1999) 169
- [6] J.H.Page, J.Liu, B.Abeles, H.W.Deckman und D.A.Weitz: Phys.Rev.Lett. **71** (1996) 1216
- [7] S.Gross und G.H.Findenegg: Ber.Bunsenges.Phys.Chem. **101** (1997) 1726
- [8] M.Thommes, G.H.Findenegg und M.Schoen: Langmuir **11** (1995) 2137
- [9] P.J. Branton, P.G. Hall und K.S.W.Sing: J.Chem.Soc., Chem.Comm. (1993) 1257
- [10] P.J.Branton, K.S.W.Sing und J.W.White: J.Chem.Soc. Faraday Trans. **93** (1997) 2337
- [11] M.Kruk, M.Jaroniec und A.Sayari: Langmuir **13** 6267
- [12] Y.Long, T.Xu, Y.Sun und W.Dong: Langmuir **14** (1998) 6173
- [13] K.Morishige und M.Shikimi: J.Chem.Phys. **108** (1998) 7821
- [14] J.Rathousky, A.Zukal, O.Franke und G.Schulz-Ekloff: J.Chem.Soc. Faraday Trans. **91** (1995) 937

- [15] P.I.Ravikovitch, S.C.O.Domhnaill, A.V.Neimark F.Schüth und K.K.Unger: Langmuir **11** (1995) 4765
- [16] S.J.Gregg und K.S.W.Sing *Adsorption, Surface Area and Porosity* (Academic Press, New York, 1982)
- [17] K.Binder und D.P.Landau: Phys.Rev.B **37** (1988) 1745
- [18] K.Binder, D.P.Landau und S.Wansleben: Phys.Rev.B **40** (1989) 6971
- [19] E.V.Albano, K.Binder, D.W.Heermann und W.Paul: J.Stat.Phys. **61** (1990) 161
- [20] T.L.Hill: J.Chem.Phys. **15** (1947) 767
- [21] M.Schick, in *Liquids at Interfaces*, Hrsg. J.Charvolin, J.F.Joanny und J.Zinn-Justin (Elsevier, Amsterdam 1990)
- [22] M.E.Fisher und H.Nakanishi: J.Chem.Phys. **75** (1981) 5857
- [23] H.Nakanishi und M.E.Fisher: J.Chem.Phys. **78** (1983) 3279
- [24] R.Evans und P.Tarazona: Phys.Rev.Lett. **52** (1984) 557
- [25] R.Evans und U.M.B.Marconi: Chem.Phys.Lett **114** (1985) 415; Phys.Rev.A **32** (1985) 3817
- [26] R.Evans, U.M.B.Marconi und P.Tarazona: J.Chem.Phys. **84** (1986) 2376; J.Chem.Soc. Faraday Trans. 2 **82** (1986) 1763
- [27] P.Tarazona, U.M.B.Marconi und R.Evans: Mol.Phys. **60** (1987) 573
- [28] E.Bruno, U.M.B.Marconi und R.Evans: Physica A **141** (1987) 187
- [29] A.O.Parry und R.Evans: Phys.Rev.Lett. **64** (1990) 439
- [30] J.-P.Hansen und L.Verlet: Phys.Rev. **184** (1969) 151
- [31] J.J.Nicolas, K.E.Gubbins, W.B.Street und D.J.Tildesley: Mol.Phys. **37** (1979) 1429
- [32] A.Z.Panagiotopoulos: Mol.Phys. **61** (1987) 813
- [33] B.Smit und D.Frenkel: Mol.Phys. **68** (1989) 932,951

- [34] M.Rovere, D.W.Heermann und K.Binder: J.Phys. Condensed Matter **2** (1990) 7009
- [35] S.Wansleben: Comput.Phys.Commun. **43** (1987) 9
- [36] N.Metropolis, A.W.Rosenbluth, M.N.Rosenbluth, A.W.Teller und E.Teller: J.Chem.Phys. **21** (1953) 1087
- [37] R.J.Baxter: *Exactly Solved Models in Statistical Mechanics* (Academic Press, London 1982)
- [38] J.M.Hammersley und D.C.Handscomb: *Monte Carlo Methods* (London 1965)
- [39] *Monte Carlo Methods in Statistical Physics*, Hrsg. K.Binder (Springer, Berlin 1986)
- [40] D.W.Heermann: *Computer Simulation Methods in Theoretical Physics* (Springer, Berlin 1986)
- [41] *Applications of the Monte Carlo Method in Statistical Physics* Hrsg. K.Binder (Springer, Berlin 1987)
- [42] *The Monte Carlo Method in Condensed Matter Physics* Hrsg. K.Binder (Springer, Berlin 1992)
- [43] *Monte Carlo and Molecular Dynamics Simulations in Polymer Science* Hrsg. K.Binder (Oxford University Press, New York/Oxford 1995)
- [44] W.Feller: *An Introduction to Probability Theory and its Applications* (J.Wiley & Sons, New York 1970)
- [45] H.Müller-Krumbhaar und K.Binder: J.Stat.Phys. **8** (1973) 1
- [46] H.Flyvbjerg und H.G.Peterson: J.Chem.Phys. **91** (1989) 461
- [47] S.Wansleben und D.P.Landau: Phys.Rev.B **43** (1991) 6006
- [48] S.Wansleben und D.P.Landau: J.Appl.Phys. **61** (1987) 3968
- [49] R.H.Swendsen und J.-S.Wang: Phys.Rev.Lett. **58** (1987) 86
- [50] U.Wolff: Phys.Rev.Lett. **62** (1989) 361
- [51] R.B.Potts: Proc.Camb.Phil.Soc. **48** (1952) 106

- [52] U.Wolff: Nucl.Phys.B **322** (1989) 759
- [53] U.Wolff: Nucl.Phys.B **334** (1990) 581
- [54] M.Hasenbusch: Nucl.Phys.B **333** (1990) 581
- [55] J.-S.Wang und R.H.Swendsen: Physica A **167** (1990) 565
- [56] P.W.Kasteleyn und C.M.Fortuin: J.Phys.Soc.Japan **26** (Suppl.) (1969) 11
- [57] C.M.Fortuin und P.W.Kasteleyn: Physica **57** (1972) 536
- [58] C.M.Fortuin: Physica **58** (1972) 393
- [59] C.M.Fortuin: Physica **59** (1972) 545
- [60] W.Janke, in *Computational Physics: Selected Methods, Simple Exercises, Serious Applications*, Hrsg. K.H.Hoffmann und M.Schreiber (Springer, Berlin 1996) S. 10
- [61] U.Wolff: Phys.Lett.A **228** (1989) 379
- [62] J.-S.Wang: Physica A **161** (1989) 249
- [63] P.G.Lauwers und V.Rittenberg: Phys.Lett.B **233** (1989) 197
- [64] V.S.Dotsenko, W.Selke und A.L.Talapov: Physica A **170** (1990) 278
- [65] Z.W.Salsburg, J.D.Jacobsen, W.Fickett, W.W.Wood: J.Chem.Phys. **30** (1959) 64
- [66] D.A.Chesnut, Z.W.Salsburg: J.Chem.Phys. **38** (1963) 2861
- [67] A.M.Ferrenberg, R.H.Swendsen: Phys.Rev.Lett. **61** (1988) 2635; Phys.Rev.Lett. **63** (1989) 1658(E)
- [68] H.P.Deutsch: J.Stat.Phys. **67** (1992) 1039
- [69] C.H.Bennett: J.Comp.Phys. **22** (1976) 245
- [70] K.Binder, in *Computational Methods in Field Theory*, Schladming Lecture Notes, Hrsg. H.Gausterer und C.B.Lang (Springer, Berlin 1992), S.59
- [71] L.P.Kadanoff: Physics **2** (1966) 263; in *Critical Phenomena*, Hrsg. M.S.Green (Academic Press, London 1971)

- [72] K.Binder, in *Monte Carlo Methods in Statistical Physics*, Hrsg. K.Binder (Springer, Berlin 1979), S.1
- [73] M.N.Barber, in *Phase Transitions and Critical Phenomena*, Vol. 8, Hrsg. C.Domb und J.L. Lebowitz (Academic Press, New York 1983), S.146
- [74] V.Privman, Hrsg., *Finite-Size Scaling and Numerical Simulations of Statistical Systems* (World Scientific, Singapore 1990)
- [75] K.Binder: Z.Phys.B **43** (1981) 119; Phys.Rev.Lett. **47** (1981) 693
- [76] K.Huang: *Statistical Mechanics (2nd Edition)* (J.Wiley & Sons, New York 1987)
- [77] T.W.Burkhardt, B.Derrida: Phys.Rev.B **32** (1985) 7273
- [78] A.D.Bruce: J.Phys.A **18** (1985) L873
- [79] D.Nicolaidis, A.D.Bruce: J.Phys.A **21** (1988) 223
- [80] G.Kamieniarz, H.W.J.Blöte: J.Phys.A **26** (1993) 201
- [81] S.Kirkpatrick und E.P.Stoll: J.Comp.Phys. **40** (1981) 517
- [82] N.Zierler in *Information and Control 15* (Academic Press, New York 1969)
- [83] A.M.Ferrenberg, D.P.Landau und Y.J.Wong: Phys.Rev.Lett. **69** (1992) 3382
- [84] P.Grassberger: Phys.Lett.A **181** (1993) 43
- [85] F.Schmid und N.Wilding: Int.J.Mod.Phys.C **6** (1995) 781
- [86] B.Efron: *The Jackknife, the Bootstrap and Other Resampling Plans* (SIAM, Philadelphia 1982)
- [87] K.Binder und P.C.Hohenberg: Phys.Rev.B **6** (1972) 3461
- [88] K.Binder und P.C.Hohenberg: Phys.Rev.B **9** (1974) 2194
- [89] M.E.Fisher: J.Vac.Sci.Technol. **10** (1973) 665
- [90] M.E.Fisher und P.-G. de Gennes: C.R.Acad.Sci.Paris **287** (1978) 207
- [91] M.E.Fisher und H.Au-Yang: Physica A **101** (1980) 255

- [92] K.Binder, in *Phase Transitions and Critical Phenomena*, Vol. 8, Hrsg. C.Domb und J.L.Lebowitz (Academic Press, New York 1983), S.1
- [93] M.E.Fisher, in *Proceedings of the 1970 Enrico Fermi School of Physics, Course No. 51, Varenna, Italy*, Hrsg. M.S.Green (Academic Press, New York 1971)
- [94] H.W.J.Blöte, E.Luijten und J.R.Heringa: *J.Phys.A* **28** (1995) 6289
- [95] H.W.Diehl und M.Shpot: *Nucl.Phys.B* **528** (1998) 595
- [96] A.D.Bruce und N.B.Wilding: *Phys.Rev.Lett.* **68** (1992) 193
- [97] D.P.Landau: *Physica A* **205** (1994) 41
- [98] H.W.J.Blöte und G.Kamieniarz: *Physica A* **196** (1993) 455
- [99] C.F.Baillie, R.Gupta, K.A.Hawick und G.S.Pawley: *Phys.Rev.B* **45** (1992) 10438
- [100] F.Livet: *Europhys.Lett.* **16** (1991) 139
- [101] A.M.Ferrenberg und D.P.Landau: *Phys.Rev.B* **44** (1991) 5081
- [102] N.Ito und M.Suzuki: *J.Phys.Soc.Japan* **60** (1991) 1978
- [103] H.W.J.Blöte, J.A.de Bruin, A.Compagner, J.H.Croockewit, Y.T.J.C.Fonk, R.Heringa, A.Hoogland und A.L.van Willigen: *Europhys.Lett.* **10** (1989) 105
- [104] Y.Rouault, J.Baschnagel und K.Binder: *J.Stat.Phys.* **80** (1995) 1009

# Anhang A

## Die Programme

Nachfolgend ist der Quellcode der Programme aufgelistet, mit denen die Monte Carlo-Daten erzeugt bzw. analysiert wurden.

Diese Programme sind

- **surf-sw-v01.c** : Dies ist das Monte Carlo-Programm mit dem die Daten erzeugt wurden. Der zugrunde liegende Algorithmus ist ein Swendsen-Wang-Cluster mit Ghostspin-Term.
- **r250\_new\_myv.c** : Dies ist der verwendete Zufallszahlengenerator, der auf dem R250 und dem R521 basiert.
- **multi-rew-v07.c** : Dieses Programm führt ein Multi-Histogram-Reweighting aus.

## surf-sw-v01.c

```
/* ~ /promo/critwet/sw_ghost/surf_sw_ghost.c 06.11.1997 */
/* ~ /***** */
/* ~ /***** */
#include <stdio.h>
#include <math.h>
#include <string.h>
#include <stdlib.h>
#include "/home/plato/dillmann/r250/r250_new_myv.h"
#define J 1
#define NN 6 /* Anzahl der naechsten Nachbarn */
#define BASIS_SEED 1568759
#define SEED 78931169
#define UPDATE mc_step_multicluster()
#define UPDATE mc_step_metropolis()
#define NR_G_BINS 32
#define DIG_L 4
#define DIG_D 2
#define F_NAME_MAX 65 /* Maximale Laenge des Out-File-Names */
#define BUFFER_LINES 1000 /* Anzahl der Ausgabe-Puffer Zeilen */
int *Lattice;
int D, L, A;
long N;
/* Fuer Swendsen-Wang */
long *Nn;
long *Stack;
char *Flag;
double *P_no_gh_bond_bulk;
double *P_no_gh_bond_surf;
double P;
int Gs_surf, Gs_bulk;
/* Fuer Metropolis */
double Prob[2*NN + 1];
double P_mag_surf[3];
double P_mag_bulk[3];
long Binsize;
double Temp, Beta;;
double Field, Field_h1, Surf_field;
double Epsilon, Mu;
char Series_flag;
int Start_config;
```

```
long Equi;
long Corr_g;
long Samples;
char Restart_flag;
long *Eng_histo;
double *Rnd_array;
long Rnd_per_call;
long Rnd_c;
double *E_array, *M_array; /* Ausgabe-Puffer der Laenge BUFFER_LINES */
int
make_global(int d, int l, double temp, double field, double field_h1,
            char series_flag, int start_config, long equi, long corr, long samples,
            int restart)
{
    double field_abs, surf_field_abs;
    int i, field_sign, surf_field_sign;
    long nr_bulk_sites, nr_surf_sites;
    long j;
    L = l;
    D = d;
    N = L * L * D;
    A = L * L;
    nr_bulk_sites = L*L*(D-2);
    nr_surf_sites = L*L* 2;
    Temp = temp;
    Beta = 1 / Temp;
    Field = field;
    Field_h1 = field_h1;
    Mu = -2 * (Field + NN * J);
    Epsilon = 2 * (J - Field_h1);
    field_sign = (Field < 0 ? -1 : 1);
    field_abs = (Field < 0 ? -Field : Field);
    Gs_bulk = field_sign;
    Surf_field = Field + Field_h1;
    surf_field_sign = (Surf_field < 0 ? -1 : 1);
    surf_field_abs = (Surf_field < 0 ? -Surf_field : Surf_field);
    Gs_surf = surf_field_sign;
    P = 1-exp(-2*Beta*J);
    Series_flag = series_flag;
    Start_config = start_config;
    Equi = equi;
    Corr_g = corr;
    Samples = samples;
    Binsize = Samples / (NR_G_BINS * Corr_g);
    Rnd_per_call = 5*N;
```

```

    Rnd_array = calloc(Rnd_per_call, sizeof(double));
    Lattice = calloc(N+1, sizeof(int)); /* +1 for ghost-site N (NN for surface-site) */
    Stack = calloc(N, sizeof(long));
    Mn = calloc(NN * N, sizeof(long));
    Flag = calloc(N+1, sizeof(char)); /* +1 for ghost-site N */
    Flag[N] = 1;
    Bond = calloc(NN * (N+1), sizeof(char)); /* +1 for ghost-site N */
    for(j=0; j < NN*(N+1); j++)
        Bond[j] = 0;
    P_no_gh_bond_bulk = calloc(nr_bulk_sites + 1, sizeof(double));
    P_no_gh_bond_surf = calloc(nr_surf_sites + 1, sizeof(double));
    for(i=0; i < nr_bulk_sites; i++)
        P_no_gh_bond_bulk[i] = exp(-2*Beta*field_abs*i);
    for(i=0; i < nr_surf_sites; i++)
        P_no_gh_bond_surf[i] = exp(-2*Beta*surf_field_abs*i);
    for(i=NN; i <= NN; i++)
        Prob[i+NN] = exp(-2 * Beta * J * i);
    for(i=-1; i <= 1; i++){
        P_mag_surf[i+1] = exp(-2 * Beta * Surf_field * i);
        P_mag_bulk[i+1] = exp(-2 * Beta * Field * i);
    }
    Restart_flag = restart;
    E_array = calloc(D*BUFFER_LINES, sizeof(double));
    M_array = calloc(D*BUFFER_LINES, sizeof(double));
    return 0;
}
/*****
int
create_fname_tail(char *fname_h, char *fname_t, char *fname_o)
{
    char file_name[60] = "";
    sprintf(file_name, "-DL%02dx%03d-t%.8lf-h%.8lf-hs%.8lf",
            D, L, Temp, Field, Field_hl);
    strcat(fname_t, file_name);
    strcat(fname_h, file_name);
    strcat(fname_o, file_name);
    strcat(fname_h, "-prof");
    strcat(fname_t, "-time");
    strcat(fname_o, "-out-");
    switch(Start_config){
    case -1: strcat(fname_h, "-lqd");
            strcat(fname_t, "-lqd");
            break;
    case 1: strcat(fname_h, "-gas");
            strcat(fname_t, "-gas");
            strcat(fname_o, "-gas");
            break;
    }
    return 0;
}
/*****
int
open_out_files(FILE **fpm_t, FILE **fpe_t,
               FILE **fpm_h, FILE **fpe_h,
               FILE **fpcf, FILE **fprd)
{
    char e_fname_h[F_NAME_MAX] = "e-", e_fname_t[F_NAME_MAX] = "e-";
    char m_fname_h[F_NAME_MAX] = "m-", m_fname_t[F_NAME_MAX] = "m-";
    char cf_fname[F_NAME_MAX] = "cf";
    char rd_fname_h[F_NAME_MAX] = "rd";
    char fname_h[F_NAME_MAX] = "", fname_t[F_NAME_MAX] = "";
    char fname_o[F_NAME_MAX] = "";
    create_fname_tail(fname_h, fname_t, fname_o);
    *fpcf = fopen(strcat(cf_fname, fname_o), "w");
    *fprd = fopen(strcat(rd_fname, fname_o), "w");
    if(Restart_flag){
        if(Series_flag){
            *fpe_t = fopen(strcat(e_fname_t, fname_t), "a");
            *fpm_t = fopen(strcat(m_fname_t, fname_t), "a");
        }
        return 0;
    }
    *fpe_h = fopen(strcat(e_fname_h, fname_h), "w");
    *fpm_h = fopen(strcat(m_fname_h, fname_h), "w");
    if(Series_flag){
        *fpe_t = fopen(strcat(e_fname_t, fname_t), "w");
        *fpm_t = fopen(strcat(m_fname_t, fname_t), "w");
    }
    return 0;
}
/*****
int
close_out_files(FILE **fpm_t, FILE **fpe_t,
                FILE **fpm_h, FILE **fpe_h,
                FILE **fpcf, FILE **fprd)
{
}

```

```

int
dealloc_dyn_locals(long **eng_base, long **mag, double **e1,
double **e2, double **c_v, double **m1,
double **m2, double **chi, double **err_eng, double **err_mag)
{
    free(*eng_base);
    free(*mag);
    free(*eng);
    free(*e1);
    free(*e2);
    free(*c_v);
    free(*m1);
    free(*m2);
    free(*chi);
    free(*err_eng);
    free(*err_mag);
    return 0;
}
/*****
int
init_spins_and_rnd()
{
    long i;
    int spin;
    FILE *fpcf, *fprd;
    char fname_o[60] = "", dummy1[60] = "", dummy2[60] = "";
    char cf_fname[F_NAME_MAX] = "cf", rd_fname[F_NAME_MAX] = "rd";
    if(Restart_flag){
        create_fname_tail(dummy1, dummy2, fname_o);
        fpcf = fopen( strcat(cf_fname, fname_o), "r");
        for(i=0; i < N; i++)
            fscanf(fpcf, "%d", &Lattice[i]);
        Lattice[N] = 0; /* spin of ghost-site N */
        fclose(fpcf);
        fprd = fopen( strcat(rd_fname, fname_o), "r");
        init_vector_random_generator(SEED, Rnd_per_call);
        vector_random_generator(Rnd_per_call, Rnd_array);
        read_random_generator(fprd);
        Rnd_c = 0;
        fclose(fprd);
    }
    return 0;
}
if (Start_config > 1){
    exit(0);
}
else
    for( i=0; i < N ; i++)
        Lattice[i] = Start_config;
}
/*****

```

```

fclose(*fpcf);
fclose(*fprd);
if(Restart_flag){
    if(Series_flag){
        fclose(*fpe_t);
        fclose(*fpm_t);
    }
    return 0;
}
fclose(*fpe_h);
fclose(*fpm_h);
if(Series_flag){
    fclose(*fpe_t);
    fclose(*fpm_t);
}
return 0;
}
/*****
int
alloc_dyn_locals(long **eng_base, long **mag, double **e1,
double **e2, double **c_v, double **m1,
double **m2, double **chi, double **err_eng, double **err_mag)
{
    int d, i;
    *eng_base = calloc(D, sizeof(long));
    *mag = calloc(D, sizeof(long));
    *eng = calloc(D, sizeof(double));
    *e1 = calloc(D, sizeof(double));
    *e2 = calloc(D, sizeof(double));
    *c_v = calloc(D, sizeof(double));
    *m1 = calloc(D, sizeof(double));
    *m2 = calloc(D, sizeof(double));
    *chi = calloc(D, sizeof(double));
    *err_eng = calloc(D, sizeof(double));
    *err_mag = calloc(D, sizeof(double));
    for(d=0; d < D; d++){
        (*eng)[d] = 0.0;
        (*e1)[d] = 0.0;
        (*e2)[d] = 0.0;
        (*c_v)[d] = 0.0;
        (*m1)[d] = 0.0;
        (*m2)[d] = 0.0;
        (*chi)[d] = 0.0;
        (*err_eng)[d] = 0.0;
        (*err_mag)[d] = 0.0;
    }
    return 0;
}
/*****

```



```

    nspins += Lattice[Mn[i+j*N]];
    delta = nspins * Lattice[i];
    if (Rnd_array[rnd_c++] < Prob[delta + MN] * P_mag_surf[Lattice[i] + 1])
        Lattice[i] = -Lattice[i];
    }
    return 0;
}
/*****
int
e_m_measurement(long *eng_base, long *mag)
{
    long i, n, d;
    long nspins;
    for (d=0; d < D; d++){
        eng_base[d] = 0;
        mag[d] = 0;
        for (i=0; i < A; i++){
            mag[d] += Lattice[d*A+i];
            for (n=0, nspins=0; n < MN; n++)
                nspins += Lattice[Mn[n*N+d*A+i]];
            eng_base[d] += nspins * Lattice[d*A+i];
        }
    }
    return 0;
}
/*****
int
free_globals()
{
    free(Mn);
    free(Stack);
    free(Lattice);
    free(Flag);
    free(Bond);
    free(P_no_gh_bond_bulk);
    free(P_no_gh_bond_surf);
    return 0;
}
/*****
int
dump_configuration(FILE *fp)
{
    long i;
    for (i=0; i < N; i++)
        fprintf(fp, "%d\n", Lattice[i]);
    return 0;
}
/*****
/****
/****
/****
MAIN
/****
/****
/****

```

```

if (Rnd_c >= Rnd_per_call){
    vector_random_generator(Rnd_per_call, Rnd_array);
    Rnd_c = 0;
}
if ( Rnd_array[Rnd_c++] < 0.5)
    continue; /* Falls Spin gleichbleibt => naechstes Cluster */
if (direction == Gs_bulk) /* Vssg.: Gs_bulk != Gs_surf */
    spins_in_surf = 0;
else
    spins_in_bulk = 0;
P_all = P_no_gh_bond_bulk[spins_in_bulk] *
        P_no_gh_bond_surf[spins_in_surf];
if (Rnd_c >= Rnd_per_call){
    vector_random_generator(Rnd_per_call, Rnd_array);
    Rnd_c = 0;
}
if ( Rnd_array[Rnd_c++] < p_all)
    for (i=0; i < cluster_size; i++)
        Lattice[ Stack[i] ] = -Lattice[ Stack[i] ];
}
return 0;
}
/*****
int
mc_step_metropolis()
{
    long i;
    int j, nspins;
    int delta;
    long rnd_c = 0;
    vector_random_generator(Rnd_per_call, Rnd_array);
    for (i=0; i < A; i++){
        for (j=0, nspins=0; j < MN; j++){
            nspins += Lattice[Mn[i+j*N]];
            delta = nspins * Lattice[i];
            if (Rnd_array[rnd_c++] < Prob[delta + MN] * P_mag_surf[Lattice[i] + 1])
                Lattice[i] = -Lattice[i];
        }
    }
    for ( ; i < N-A; i++){
        for (j=0, nspins=0; j < MN; j++){
            nspins += Lattice[Mn[i+j*N]];
            delta = nspins * Lattice[i];
            if (Rnd_array[rnd_c++] < Prob[delta + MN] * P_mag_bulk[Lattice[i] + 1])
                Lattice[i] = -Lattice[i];
        }
    }
    for ( ; i < N; i++){
        for (j=0, nspins=0; j < MN; j++){

```

```

/*****
main( int argc, char *argv[] )
{
    long i, j, k;
    int d, r;
    double *eng;
    double eng_all, eng_2_all;
    double *e1, *e2, *c_v;
    double mag_all, mag_2_all;
    double *m1, *m2, *chi;
    double *g1s, *g2s, *g1c, *g2c;
    double eng_tot = 0.0, eng_2_tot = 0.0;
    double mag_tot = 0.0, mag_2_tot = 0.0;
    double *err_eng, *err_mag;
    double err_eng_tot, err_mag_tot;
    double *m1_bin;
    long *eng_base, *mag;
    long samples_em = 0, samples_g = 0, factor;
    double fac_samp, fac_g_samp, fac_nr, fac_a, fac_l, fac_l3, fac_bin;
    FILE *fpm_t, *fpe_t;
    FILE *fpm_h, *fpe_h;
    FILE *fpcf, *fprd;
    int bin_sample = 0;

    make_global(atoi(argv[1]), atoi(argv[2]), atof(argv[3]), atof(argv[4]),
               atof(argv[5]), atoi(argv[6]), atoi(argv[7]), atoi(argv[8]),
               atoi(argv[9]), atoi(argv[10]), atoi(argv[11]));

    alloc_dyn_locals(&eng_base, &mag, &e1, &e2, &c_v, &m1,
                   &m2, &chi, &err_eng, &err_mag);

    init_spins_and_rnd();

    open_out_files(&fpm_t, &fpe_t, &fpm_h, &fpe_h, &fpcf, &fprd);

    init_pbc();

    fac_nr = 1.0 / N;
    fac_a = 1.0 / A;
    fac_l = 1.0 / L;
    fac_l3 = 1.0 / (L*L*L);
    fac_bin = 1.0 / Binsize;

    if(!Restart_flag)
        for(i=1; i <= Equi; i++)
            UPDATE;

    for(i=1; i <= Samples; i++){
        for(j=1; j <= Corr_g/BUFFER_LINES; j++){
            for(k=0; k < BUFFER_LINES; k++){
                UPDATE;

                e_m_measurement(eng_base, mag);
            }
        }
    }
}
/*****/

samples_em++;
for(d=0; d < D; d++)
    eng[d] = -0.5 * eng_base[d] * J - Field * mag[d];

eng[0] -= Field_h1 * mag[0];
eng[D-1] -= Field_h1 * mag[D-1];

eng_all = 0.0;
mag_all = 0.0;

for(d=0; d < D; d++){
    eng_all += eng[d];
    mag_all += mag[d];
    e1[d] += eng[d];
    e2[d] += eng[d] * eng[d];
    m1[d] += mag[d];
    m2[d] += mag[d] * mag[d];
    E_array[k*D+d] = eng[d] * fac_a; /* in den Puffer schreiben */
    M_array[k*D+d] = mag[d] * fac_a; /* in den Puffer schreiben */
}

eng_tot += eng_all;
eng_2_tot += eng_all * eng_all;
mag_tot += mag_all;
mag_2_tot += mag_all * mag_all;
}

if(Series_flag){
    for(k=0; k < BUFFER_LINES; k++){
        for(d=0; d < D; d++){
            fprintf(fpe_t, "%lf ", E_array[k*D+d]);
            fprintf(fpm_t, "%lf ", M_array[k*D+d]);
        }
        fprintf(fpe_t, "\n");
        fprintf(fpm_t, "\n");
    }
}

fac_samp = 1.0 / samples_em;

for(d=0; d < D; d++){
    e1[d] *= fac_samp * fac_a;
    e2[d] *= fac_samp * fac_a * fac_a;
    m1[d] *= fac_samp * fac_a;
    m2[d] *= fac_samp * fac_a * fac_a;
}

eng_tot *= fac_samp * fac_nr;
eng_2_tot *= fac_samp * fac_nr;
mag_tot *= fac_samp * fac_nr;
}

```

```

mag_2_tot *= fac_samp * fac_nr;
err_eng_tot = sqrt( fac_samp * (eng_2_tot * fac_nr - eng_tot * eng_tot) );
err_mag_tot = sqrt( fac_samp * (mag_2_tot * fac_nr - mag_tot * mag_tot) );
for(d=0; d < D; d++){
    err_eng[d] = sqrt( fac_samp * (e2[d] - e1[d] * e1[d]) );
    err_mag[d] = sqrt( fac_samp * (m2[d] - m1[d] * m1[d]) );
}
if(!Restart_flag){
    for(d=0; d < D; d++){
        fprintf(fpe_h,"%2d %lf %lf\n", d+1, e1[d], err_eng[d], e2[d]);
        fprintf(fpm_h,"%2d %lf %lf\n", d+1, m1[d], err_mag[d], m2[d]);
    }
    fprintf(fpe_h,"\n %lf %lf\n", eng_tot, err_eng_tot, eng_2_tot);
    fprintf(fpm_h,"\n %lf %lf\n", mag_tot, err_mag_tot, mag_2_tot);
}
dump_configuration(fpcf);
write_random_generator(fprd);
close_out_files(&fpm_t, &fpe_t, &fpm_h, &fpe_h, &fpcf, &fprd);
dealloc_dyn_locals(&eng_base, &mag, &eng, &e1, &e2, &c_v, &m1,
&m2, &chi, &err_eng, &err_mag);
free_globals();
return 0;
}

```

## r250\_new\_myv.c

```

/* ***** */
/* write_random_generator() and read_random_generator() adapted
/* adjusted to my needs... OD 29.01.1998
/* ***** */
/* Vectorized version of a (hopefully) improved shift-register
/* pseudorandom number generator.
/* Based on two independent generators with different pairs
/* of "magic numbers", here chosen as (250,103) and (521,168).
/* The output of those is XORed together and yields the final
/* random number. For theoretical reasons, this generator should
/* be significantly less hampered by correlations than the simple
/* good old R250.
/* Disclaimer of warranty:
/* I will not be held responsible for any problems whatsoever
/* which this program may cause.
/* B. Duenweg, July 9, 1996.
/* Some (not all!) other possible magic umbers are (see also
/* Kirkpatrick and Stoll, Journ. Comp. Phys. 40, 517 (1981), and
/* N. Zierler, Information and Control 15, p. 67 (1969)):
/* (98, 27)
/* (521, 32)
/* (521, 48)
/* (521, 158)
/* (607, 105)
/* (607, 147)
/* (607, 273)
/* (1279, 216)
/* (1279, 418)
/* (2281, 715)
/* (2281, 915)
/* (2281, 1029)
/* (9689, 4187)
/* The program can be changed very easily to other magic numbers
/* by just changing the parameters
/* BIGMAGIC1, BIGMAGIC2, SMALLMAGIC1 and SMALLMAGIC2.
/* Since it is explicitly written for 31 bit integers, it
/* should produce the SAME sequence on any architecture.
/* This is how it works:
/* - Specify a "seed" value iseed.
/* - Specify how many random numbers should be generated
/* on ONE call of the generator. Call this number, say,
/* nrand. A typical value is, say, 100000.
/* ***** */
/* - Invoke the function
/* init_vector_random_generator(iseed,nrand).
/* This does the following:
/* - A simple congruential generator is run for NWARM
/* (in our case 10000) times.
/* - Two integer working arrays of size BIGMAGIC1 + nrand,
/* BIGMAGIC2 + nrand, named rand_w_array1 and rand_w_array2
/* are created (i.e. memory is allocated).
/* - The first BIGMAGIC array elements are
/* filled with (congruential) random numbers.
/* - The bit columns are treated for linear independence.
/* - The shift-register generators are run for nrand times
/* (in order to also warm them up) and are then ready to go.
/* - Invoke the function
/* vector_random_generator(nrand, random_numbers).
/* Result: Normalized double precision random numbers in [0:1]
/* are written on the array random_numbers.
/* The current "state" of the generator is "coded" in the first
/* BIGMAGICX elements of rand_w_arrayX.
/* - There are also routines provided to save this status to a
/* file, and read it from there. Have care: In case you read
/* the status, make sure to first run the function
/* init_vector_random_generator in order to assure proper
/* memory allocation!
/* ***** */
#include <stdio.h>
#include <stdlib.h>
#define BIGMAGIC1 250 /* magic numbers for the
#define SMALLMAGIC1 103 /* first generator
#define BIGMAGIC2 521 /* magic numbers for the
#define SMALLMAGIC2 168 /* second generator
#define NBIT 32 /* use only (NBIT - 1) bits
#define BIGINTEGER 2147483647 /* = largest integer
#define BIGFLOAT 2147483647. /* same in float
#define FACTOR 4.6566128752457969e-10 /* = 1. / (largest integer)
#define MULTIPLY 16807. /* for congruential generator
#define NWARM 10000 /* number of empty runs, c.g.
#define WORKFILE "ran250.dat" /* to store the status
/* The working arrays are declared as static so they need not be passed */
/* to the main program */
static int *rand_w_array1 = NULL;
static int *rand_w_array2 = NULL;
void init_vector_random_generator(int iseed,int nrand)
{
extern int *rand_w_array1;

```

```

extern int *rand_w_array2;
double rmod;
int i, ihlp, imask1, imask2;
int icyc, ncy, nrest, ibas1, ibas2, ibas3;
if(iseed <=0 || iseed >= BIGINTEGER)
{
    printf("Message from random number initialization:\n");
    printf("Please specify a seed smaller than %d\n",BIGINTEGER);
    exit(0);
}
if(nrand <=0)
{
    printf("Message from random number initialization:\n");
    printf("Please specify a positive number of random numbers\n");
    exit(0);
}
rmod = (double) (iseed);
/* Warm up the congruential generator */
for(i = 0; i < NWARM; ++i)
{
    rmod = MULTIPLY * rmod;
    rmod = rmod - ( (double) ( (int) (rmod * FACTOR) ) ) * BIGFLOAT;
    ihlp = (int) (rmod + 0.1); /* This is done to get rid of */
    rmod = (double) (ihlp); /* possible roundoff errors */
}
/* Allocate memory for the working arrays */
rand_w_array1 = (int *) calloc(BIGMAGIC1 + nrand, sizeof(int));
rand_w_array2 = (int *) calloc(BIGMAGIC2 + nrand, sizeof(int));
/* Put congruential random numbers onto the working arrays */
for(i = 0; i < BIGMAGIC1; ++i)
{
    rmod = MULTIPLY * rmod;
    rmod = rmod - ( (double) ( (int) (rmod * FACTOR) ) ) * BIGFLOAT;
    ihlp = (int) (rmod + 0.1);
    rmod = (double) (ihlp);
    rand_w_array1[i] = ihlp;
}
for(i = 0; i < BIGMAGIC2; ++i)
{
    rmod = MULTIPLY * rmod;
    rmod = rmod - ( (double) ( (int) (rmod * FACTOR) ) ) * BIGFLOAT;
    ihlp = (int) (rmod + 0.1);
    rmod = (double) (ihlp);
    rand_w_array2[i] = ihlp;
}
/* Linear independence of the bit columns for both generators. */
/* Put ones on the main diagonal, and zeroes above. */

```

```

/* & is the bitwise AND
/* | is the bitwise OR
/* ^ is the bitwise XOR
imask1 = 1;
imask2 = BIGINTEGER;
for(i = NBIT - 2; i > 0; --i)
{
    rand_w_array1[i] = ( rand_w_array1[i] | imask1 ) & imask2;
    rand_w_array2[i] = ( rand_w_array2[i] | imask1 ) & imask2;
    imask2 ^= imask1;
    imask1 = imask1 * 2;
}
rand_w_array1[0] = imask1; /* This last element is treated separately */
rand_w_array2[0] = imask1; /* in order to avoid overflow in imask1 */
/* Warm up. Same structure as in vector_random_generator. */
/* Double loop structure to enable vectorization of inner loop */
/* First, generator one */
ncy = nrand / SMALLMAGIC1;
nrest = nrand - SMALLMAGIC1 * ncy;
ibas3 = BIGMAGIC1; /* position of first new random number */
ibas2 = BIGMAGIC1 - SMALLMAGIC1; /* position of first input for this */
ibas1 = 0; /* position of second input for this */
for(icyc = 0; icyc < ncy; ++icyc)
{
    #pragma ivdep
    for(i = 0; i < SMALLMAGIC1; ++i)
    {
        rand_w_array1[ibas3 + i] = rand_w_array1[ibas1 + i]
        ^ rand_w_array1[ibas2 + i];
    }
    ibas1 = ibas1 + SMALLMAGIC1;
    ibas2 = ibas2 + SMALLMAGIC1;
    ibas3 = ibas3 + SMALLMAGIC1;
}
if(nrest > 0)
{
    #pragma ivdep
    for(i = 0; i < nrest; ++i)
    {
        rand_w_array1[ibas3 + i] = rand_w_array1[ibas1 + i]
        ^ rand_w_array1[ibas2 + i];
    }
}
/* Put last elements to the beginning */
#pragma ivdep
for(i = 0; i < BIGMAGIC1; ++i)
{
    rand_w_array1[i] = rand_w_array1[nrand + i];
}
}
}

```

```

/* Now the same for the second generator */
ncyc = nrand / SMALLMAGIC2;
nrest = nrand - SMALLMAGIC2 * ncyc;
ibas3 = BIGMAGIC2; /* position of first new random number */
ibas2 = BIGMAGIC2 - SMALLMAGIC2; /* position of first input for this */
ibas1 = 0; /* position of second input for this */
for(icyc = 0; icyc < ncyc; ++icyc)
{
    rand_w_array2[ibas3 + i] = rand_w_array2[ibas1 + i]
    ^ rand_w_array2[ibas2 + i];
}
#pragma ivdep
for(i = 0; i < SMALLMAGIC2; ++i)
{
    rand_w_array2[ibas3 + i] = rand_w_array2[ibas1 + i]
    ^ rand_w_array2[ibas2 + i];
}
ibas1 = ibas1 + SMALLMAGIC2;
ibas2 = ibas2 + SMALLMAGIC2;
ibas3 = ibas3 + SMALLMAGIC2;
}
if(nrest > 0)
{
    #pragma ivdep
    for(i = 0; i < nrest; ++i)
    {
        rand_w_array2[ibas3 + i] = rand_w_array2[ibas1 + i]
        ^ rand_w_array2[ibas2 + i];
    }
}
/* Put last elements to the beginning */
#pragma ivdep
for(i = 0; i < nrest; ++i)
{
    rand_w_array2[ibas3 + i] = rand_w_array2[ibas1 + i]
    ^ rand_w_array2[ibas2 + i];
}
}
/* Now the same for the second generator */
ncyc = nrand / SMALLMAGIC2;
nrest = nrand - SMALLMAGIC2 * ncyc;
ibas3 = BIGMAGIC2; /* position of first new random number */
ibas2 = BIGMAGIC2 - SMALLMAGIC2; /* position of first input for this */
ibas1 = 0; /* position of second input for this */
for(icyc = 0; icyc < ncyc; ++icyc)
{
    rand_w_array2[ibas3 + i] = rand_w_array2[ibas1 + i]
    ^ rand_w_array2[ibas2 + i];
}
#pragma ivdep
for(i = 0; i < SMALLMAGIC2; ++i)
{
    rand_w_array2[ibas3 + i] = rand_w_array2[ibas1 + i]
    ^ rand_w_array2[ibas2 + i];
}
ibas1 = ibas1 + SMALLMAGIC2;
ibas2 = ibas2 + SMALLMAGIC2;
ibas3 = ibas3 + SMALLMAGIC2;
}
if(nrest > 0)
{
    #pragma ivdep
    for(i = 0; i < nrest; ++i)
    {
        rand_w_array2[ibas3 + i] = rand_w_array2[ibas1 + i]
        ^ rand_w_array2[ibas2 + i];
    }
}
}
/* Put last elements to the beginning */
#pragma ivdep
for(i = 0; i < nrest; ++i)
{
    rand_w_array2[ibas3 + i] = rand_w_array2[ibas1 + i]
    ^ rand_w_array2[ibas2 + i];
}
}
/* Initialization complete */
return;
}
void vector_random_generator(int nrand, double *random_numbers)
{
    extern int *rand_w_array1;
    extern int *rand_w_array2;
    int i, icyc, ncyc, nrest, ibas1, ibas2, ibas3;
    /* First, run generator one */
    ncyc = nrand / SMALLMAGIC1;
    nrest = nrand - SMALLMAGIC1 * ncyc;
    ibas3 = BIGMAGIC1; /* position of first new random number */
}

```

```

#pragma ivdep
for(i = 0; i < nrest; ++i)
{
    rand_w_array2[ibas3 + i] = rand_w_array2[ibas1 + i]
    ^ rand_w_array2[ibas2 + i];
}

/* Put last elements to the beginning */
#pragma ivdep
for(i = 0; i < BIGMAGIC2; ++i)
{
    rand_w_array2[i] = rand_w_array2[nrand + i];
}

/* Generate normalized random numbers:
/* Take output from generator one and combine it with
/* that from generator two, via a simple XOR
*/
#pragma ivdep
for(i = 0; i < nrand; ++i)
{
    random_numbers[i] = FACTOR *
    (rand_w_array1[i + BIGMAGIC1] ^ rand_w_array2[i + BIGMAGIC2]);
}
return;
}

void write_random_generator(FILE *fp)
{
    extern int *rand_w_array1;
    extern int *rand_w_array2;

    fwrite(rand_w_array1, sizeof(int), BIGMAGIC1, fp);
    fwrite(rand_w_array2, sizeof(int), BIGMAGIC2, fp);
}
return;
}

void read_random_generator(FILE *fp)
{
    extern int *rand_w_array1;
    extern int *rand_w_array2;

    fread(rand_w_array1, sizeof(int), BIGMAGIC1, fp);
    fread(rand_w_array2, sizeof(int), BIGMAGIC2, fp);
}
return;
}

```

## multi-rew-v07.c

```

*****
/*
**
**          07.12.1999
** Programm liest "init.dat" (enthalt die Dateinamen der Histogramme, die in
** die Berechnung eingehen sollen und deren
** Autokorrelationslaengen /tau
** und "para.dat" (spezifiziert den Parameterbereich der zu berech-
** nenden Daten.)
** ein.
** Falls Vektor der Zustandsummen noch nicht berechnet (Datei "part.dat"
** existiert nicht in diesem Verzeichnis) wird dieser iteriert und der Fort-
** gang in die Datei "iter.dat" ausgegeben.
**
** 1. Command-line parameter ist das Format der "para.dat" Datei, d. h.:
**      0 -> Area Format
**      1 -> single-point Format
**      -1 -> coex-line Format (\equiv Area Format)
**      -2 -> coex-single Format (\equiv single-point Format
**          mit H_min, H_max in 1. Zeile
**
** 2. Command-line parameter ist die Daten-Ausgabe, d. h. :
**      0 -> Out-Data
**      1 -> Out-Data + Out-Histo
**
*****
#include <stdio.h>
#include <math.h>
#include <string.h>
#include <stdlib.h>

#define F_IN_NAME_LENGTH 80
#define F_OUT_NAME_LENGTH 80
#define INIT_FILE_NAME "init.dat"
#define ITER_FILE_NAME "iter.dat"
#define PARA_FILE_NAME "para.dat"
#define PART_FILE_NAME "part.dat"
#define EDATA_FILE_NAME "e-data.dat"
#define MDATA_FILE_NAME "m-data.dat"
#define PDATA_FILE_NAME "p-data.dat"
#define COEX_FILE_NAME "coex-data.dat"
#define COEX_INIT_FILE_LENGTH 100
#define PREC 1.0e-10
#define OFFSET 1.0e-7
#define NR_ROWS 3
#define NR_PARAS 3
#define MAX_ITER 100
#define AREA_FORMAT 0
#define SINGLE_FORMAT 1
#define COEX_FORMAT -1
#define COEX_SINGLE_FORMAT -2
#define OUT_DATA 0
#define OUT_DATA_AND_HISTO 1
#define TOL 1.0e-8

int Nr_of_points;
FILE *Fp_init;

char Eng_in_file[F_IN_NAME_LENGTH];
char Mag_in_file[F_IN_NAME_LENGTH];
char Out_file[F_IN_NAME_LENGTH];
int D_L;
long N, L2, S;
double T_act, H_min, H_max, Hs_act;
double *Histo;
long *N_j;
double *Beta, *B_h, *B_hs, *Fac_tau, *Z;

int
open_for_read(char *file_name, FILE **fp_read)
{
    *fp_read = fopen(file_name, "r");
    return 0;
}

int
open_for_write(char *file_name, FILE **fp_write)
{
    *fp_write = fopen(file_name, "w");
    return 0;
}

int
close_file(FILE **fp)
{
    fclose(*fp);
    return 0;
}

int
get_nr_of_points()
{
    int lines = 0;
    char s[INIT_FILE_LENGTH];
    open_for_read(INIT_FILE_NAME, &Fp_init);
    while( fgets(s, sizeof(s), Fp_init) )
        lines++;
    close_file(&Fp_init);
    return lines;
}

int
get_params_and_open_in_files(FILE **fp_in_stack, double *tau, long *n,
                             double *beta, double *h, double *hs)
{
    int i;

```

```

double temp;
char file_name[Nr_of_points][INIT_FILE_LENGTH];
char s[INIT_FILE_LENGTH];
open_for_read(INIT_FILE_NAME, &Fp_init);
for(i=0; i < Nr_of_points; i++){
    fscanf(Fp_init, "%s", file_name[i]);
    fscanf(Fp_init, "%lf", &tau[i]);
    tau[i] = 1.0 / (1.0 + 2.0 * tau[i]);
}
for(i=0; i < Nr_of_points; i++){
    sscanf(file_name[i], "%d-%d%03d-%t%lf-h%lf-hs%lf-hist-gas.rw",
    &D, &L, &temp, &h[i], &hs[i]);
    beta[i] = 1.0 / temp;
}
N = D*L*L;
L2 = L*L;
S = 2*L2;
close_file(&Fp_init);
for(i=0; i < Nr_of_points; i++){
    fp_in_stack[i] = fopen(file_name[i], "r");
    n[i] = 0;
    while( fgets(s, sizeof(s), fp_in_stack[i]) )
        n[i]++;
    rewind(fp_in_stack[i]);
}
return 0;
}
int
get_histo(FILE **fp_in_stack, long *n)
{
    int file;
    long i, ww, rho, rho_s;
    double m, m_s;
    long index=0;
    for(file=0; file < Nr_of_points; file++){
        for(i=0; i < n[file]; i++){
            m = N-2*rho;
            m_s = 2*(L2-rho_s);
            Histo[index] = (double)ww;
            Histo[index+1] = (double)m;
            Histo[index+2] = (double)m_s;
            index += NR_ROWS;
        }
    }
    return 0;
}
int
compute_part_func(double *z_0, long *n, double *fac_tau,
double *beta, double *b_h, double *b_hs)
{
    int i, j, k, w, file;
    long line, factor;
    double ww, m, ms;
    double common_fac, zaehler, nenner;
    double *z, sum;
    z = calloc(Nr_of_points, sizeof(double));
    for(i=0; i < Nr_of_points; i++)
        z[i] = 0.0;
    for(w=0; w < Nr_of_points; w++){
        line = 0;
        for(file=0; file < Nr_of_points; file++){
            sum = 0.0;
            for(j=0; j < n[file]; j++){
                factor = NR_ROWS*line;
                ww = Histo[factor];
                m = Histo[factor + 1];
                ms = Histo[factor + 2];
                for(k=0, common_fac=0.0; k < Nr_of_points; k++){
                    common_fac += beta[k]*ww + b_h[k]*m + b_hs[k]*ms;
                }
                zaehler = exp( beta[w]*ww + b_h[w]*m + b_hs[w]*ms - common_fac);
                for(k=0, nenner=0.0; k < Nr_of_points; k++){
                    nenner += (double)n[k]/z_0[k] * fac_tau[k] *
                        exp( beta[k]*ww + b_h[k]*m + b_hs[k]*ms - common_fac );
                }
                sum += zaehler / nenner;
                line++;
            }
        }
        sum *= fac_tau[file];
        z[w] += sum;
    }
}
for(i=0; i < Nr_of_points; i++)
    z_0[i] = z[i];
free(z);
return 0;
}
int
part_data_exists(double *z)
{
    int i, yes=1, no=0, enough_points = 1;
    FILE *fp;
    open_for_read(PART_FILE_NAME, &fp);
    if(!fp)
        return no;
}

```

```

free(z_prev);
return 0;
}

int
init_para_list(double **list, char param_format)
{
    double temp, temp_min, temp_max, temp_step;
    double h, h_min, h_max, h_step;
    double hs, hs_min, hs_max, hs_step;
    int i, list_length;
    FILE *fp;
    char s[INIT_FILE_LENGTH];
    if ( (param_format == AREA_FORMAT) ){
        get_rev_para_area(&temp_min, &temp_max, &temp_step,
            &h_min, &h_max, &h_step,
            &hs_min, &hs_max, &hs_step);
        list_length = 0;
        for(temp=temp_min; temp < temp_max+OFFSET; temp+=temp_step)
            for(h=h_min; h < h_max+OFFSET; h+=h_step)
                for(hs=hs_min; hs < hs_max+OFFSET; hs+=hs_step)
                    list_length++;
        *list = calloc(NR_PARAS*list_length, sizeof(double));
        i = 0;
        for(temp=temp_min; temp < temp_max+OFFSET; temp+=temp_step)
            for(h=h_min; h < h_max+OFFSET; h+=h_step)
                for(hs=hs_min; hs < hs_max+OFFSET; hs+=hs_step){
                    (*list)[i] = temp;
                    (*list)[i+1] = h;
                    (*list)[i+2] = hs;
                    i += NR_PARAS;
                }
            return list_length;
        }
    if ( (param_format == COEX_FORMAT) ){
        get_rev_para_area(&temp_min, &temp_max, &temp_step,
            &h_min, &h_max, &h_step,
            &hs_min, &hs_max, &hs_step);
        H_min = h_min;
        H_max = h_max;
        list_length = 0;
        for(temp=temp_min; temp < temp_max+OFFSET; temp+=temp_step)
            list_length++;
        *list = calloc(NR_PARAS*list_length, sizeof(double));
        i = 0;
        for(temp=temp_min; temp < temp_max+OFFSET; temp+=temp_step){
            (*list)[i] = temp;
}
}

for(i=0; i < Nr_of_points; i++)
    fscanf(fp, "%lg", &z[i]);
close_file(&fp);
return yes;
}

int
iterate_part_func(double *z, long *n, double *fac_tau,
    double *beta, double *b_h, double *b_hs)
{
    FILE *fp_iter, *fp_part;
    int i, k, iter;
    double *z_prev;
    z_prev = calloc(Nr_of_points, sizeof(double));
    open_for_write(ITER_FILE_NAME, &fp_iter);
    for(i=0; i < Nr_of_points; i++)
        z[i] = (double)n[i];
    fprintf(fp_iter, "%d: ", iter=0);
    for(k=0; k < Nr_of_points; k++)
        fprintf(fp_iter, "%10lg ", z[k]);
    fflush(fp_iter);
    for(iter=1; iter <= MAX_ITER; iter++){
        for(k=0; k < Nr_of_points; k++)
            z_prev[k] = z[k];
        compute_part_func(z, n, fac_tau, beta, b_h, b_hs);
        fprintf(fp_iter, "%d: ", iter);
        for(k=0; k < Nr_of_points; k++)
            fprintf(fp_iter, "%10lg ", z[k]);
        fflush(fp_iter);
        if ( stop_iteration(z, z_prev) )
            break;
    }
    fprintf(fp_iter, "\n");
    close_file(&fp_iter);
    open_for_write(PART_FILE_NAME, &fp_part);
    for(k=0; k < Nr_of_points; k++)
        fprintf(fp_part, "%10lg ", z[k]);
    fprintf(fp_part, "\n");
    close_file(&fp_iter);
}

```

```

        double *hs_min, double *hs_max, double *hs_step)
    {
        int i;
        FILE *fp;

        open_for_read(PARA_FILE_NAME, &fp);

        fscanf(fp, "%lf", temp_min);
        fscanf(fp, "%lf", temp_max);
        fscanf(fp, "%lf", temp_step);
        fscanf(fp, "%lf", h_min);
        fscanf(fp, "%lf", h_max);
        fscanf(fp, "%lf", h_step);
        fscanf(fp, "%lf", hs_min);
        fscanf(fp, "%lf", hs_max);
        fscanf(fp, "%lf", hs_step);

        close_file(&fp);

        return 0;
    }

    int
    stop_iteration(double *z, double *z_prev)
    {
        int i, yes=1, no=0;
        double quality, prec;

        for(i=0; i < Nr_of_points; i++){
            prec = z[i]/z_prev[i] - 1.0;
            quality = (prec > 0.0 ? prec : -prec);
            if(quality > PREC)
                return no;
        }

        return yes;
    }

    int
    close_in_files(FILE **fp_in_stack)
    {
        int i;

        for(i=0; i < Nr_of_points; i++)
            fclose(fp_in_stack[i]);

        return 0;
    }

    double m_reweight(double param)
    {
        double mag, part, prob;
        double temp, h, hs;
        double b_new, bh_new, bhs_new;
        double ww, m, ms;
        long i, j, line, factor;
        long rho;

```

```

        (*list)[i+1] = h_min;
        (*list)[i+2] = hs_min;
        i += NR_PARAS;
    }

    return list_length;
}

if( (param_format == SINGLE_FORMAT) ){
    open_for_read(PARA_FILE_NAME, &fp);

    list_length = 0;
    while( fgets(s, sizeof(s), fp) )
        list_length++;

    rewind(fp);

    *list = calloc(NR_PARAS*list_length, sizeof(double));
    for(i=0; i < NR_PARAS*list_length; i++)
        fscanf(fp, "%lf ", &((*list)[i]));

    close_file(&fp);

    return list_length;
}

if( (param_format == COEX_SINGLE_FORMAT) ){
    open_for_read(PARA_FILE_NAME, &fp);

    list_length = 0;
    while( fgets(s, sizeof(s), fp) )
        list_length++;

    rewind(fp);

    list_length--; /* in 1.Zeile steht H_min, H_max*/
    fscanf(fp, "%lf %lf ", &H_min, &H_max);
    /* Einlesen von H_min, H_max */

    *list = calloc(NR_PARAS*list_length, sizeof(double));
    for(i=0; i < NR_PARAS*list_length; i++)
        fscanf(fp, "%lf ", &((*list)[i]));

    close_file(&fp);

    return list_length;
}
}

int
get_rev_para_area(double *temp_min, double *temp_max, double *temp_step,
                  double *h_min, double *h_max, double *h_step,

```

```

int k, file;
double *p, p_less, value;
double common_fac, zaehler, nenner;
p = calloc(N+1, sizeof(double));
for(i=0; i < N+1; i++)
    p[i] = 0.0;
temp = T_act;
h = param;
hs = Hs_act;
b_new = 1.0/temp;
bh_new = b_new * h;
bhs_new = b_new * hs;
mag = 0.0;
part = 0.0;
line = 0;
for(file=0; file < Nr_of_points; file++)
    for(j=0; j < N_j[file]; j++){
        factor = Nr_ROWS*line;
        ww = Histo[factor];
        m = Histo[factor + 1];
        ms = Histo[factor + 2];
        rho = (N-(long)m) / 2;
        for(k=0, common_fac=0.0; k < Nr_of_points; k++)
            common_fac += Beta[k]*ww + B_h[k]*m + B_hs[k]*ms;
        common_fac /= (double)k;
        zaehler = exp( b_new*ww + bh_new*m + bhs_new*ms - common_fac);
        for(k=0, nenner=0.0; k < Nr_of_points; k++)
            nenner += (double)N_j[k]/Z[k] * Fac_tau[k] *
                exp( Beta[k]*ww + B_h[k]*m + B_hs[k]*ms - common_fac );
        prob = zaehler / nenner * Fac_tau[file];
        part += prob;
        p[rho] += prob;
        m /= N;
        mag += m * prob;
        line++;
    }
mag /= part;
p_less = 0.0;
for(rho=0; rho < N+1; rho++){
    p[rho] /= part;
    m = 1.0 - 2.0/N * rho;
    if(m <= mag)
        p_less += p[rho];
    }
}

free(p);
value = p_less - 0.5;
return value;
}
#define ITMAX 100
#define EPS 3.0e-8
double
zbrent(double (*func)(double), double x1, double x2, double tol)
{
    int iter;
    double a=x1, b=x2, c, d, e, mini, min2;
    double fa=(*func)(a), fb=(*func)(b), fc, p, q, r, s, toll, xm;
    void nerror();
    if (fb*fa > 0.0) nerror("Root must be bracketed in ZBRENT");
    fc=fb;
    for (iter=1; iter<=ITMAX; iter++) {
        if (fb*fc > 0.0) {
            c=a;
            fc=fa;
            e=q=b-a;
        }
        if (fabs(fc) < fabs(fb)) {
            a=b;
            b=c;
            c=e;
            fa=fb;
            fb=fc;
            fc=fa;
        }
        toll=2.0*EPS*fabs(b)+0.5*tol;
        xm=0.5*(c-b);
        if (fabs(xm) <= toll || fb == 0.0) return b;
        if (fabs(e) >= toll && fabs(fa) > fabs(fb)) {
            s=fb/fa;
            if (a == c) {
                p=2.0*xm*s;
                q=1.0-s;
            } else {
                q=fa/fc;
                r=fb/fc;
                p=s*(2.0*xm*q*(q-r)-(b-a)*(r-1.0));
                q=(q-1.0)*(r-1.0)*(s-1.0);
            }
            if (p > 0.0) q = -q;
            p=fabs(p);
            mini=3.0*xm*q-fabs(toll*q);
            min2=fabs(e*q);
            if (2.0*p < (mini < min2 ? mini : min2)) {
                e=d;
                d=p/q;
            } else {

```

```

field_s = calloc(Nr_of_points, sizeof(double));
Fac_tau = calloc(Nr_of_points, sizeof(double));
N_j = calloc(Nr_of_points, sizeof(long));
Beta = calloc(Nr_of_points, sizeof(double));
B_h = calloc(Nr_of_points, sizeof(double));
B_hs = calloc(Nr_of_points, sizeof(double));
Z = calloc(Nr_of_points, sizeof(double));

get_params_and_open_in_files(fp_in_file_array, Fac_tau,
                             N_j, Beta, field, field_s);

for(i=0; i < Nr_of_points; i++){
    B_h[i] = Beta[i] * field[i];
    B_hs[i] = Beta[i] * field_s[i];
    printf("%lf %ld %lf %lf %lf\n", Fac_tau[i], N_j[i], Beta[i],
        B_h[i], field[i], B_hs[i], field_s[i]);
}
for(i=0; i < Nr_of_points; i++)
    entries += N_j[i];
printf("entries: %ld\n", entries);

Histo = calloc(entries * NR_ROWS, sizeof(double));
get_histo(fp_in_file_array, N_j);
close_in_files(fp_in_file_array);
if(part_data_exists(Z))
;
else
    iterate_part_func(Z, N_j, Fac_tau, Beta, B_h, B_hs);

for(i=0; i < Nr_of_points; i++)
    printf("%.10lg ", Z[i]);
printf("\n");

list_length = init_para_list(&para_list, para_format);
printf("list_length %d\n", list_length);

if( (para_format == COEX_FORMAT)
    || (para_format == COEX_SINGLE_FORMAT)){
    printf("\nDetermine Coexistence Line\n");
    fflush(NULL);
    open_for_write(COEX_FILE_NAME, &fp_coex);
    for(set=0; set < list_length; set++){
        T_act = para_list[set*NR_PARAS];
        Hs_act = para_list[set*NR_PARAS + 2];
        h_coex = zbrent( (double *) (double)(m_reweight), H_min, H_max, TOL);
        fprintf(fp_coex, "%lf %f %f\n", T_act, h_coex, Hs_act);
        fflush(NULL);
    }
}

```

```

d=xm;
e=d;
} else {
d=xm;
e=d;
}
a=b;
fa=fb;
if (fabs(d) > toll)
b += d;
else
b += (xm > 0.0 ? fabs(toll) : -fabs(toll));
fb=(*func)(b);
}
nrerror("Maximum number of iterations exceeded in ZBRENT");
}
}
#undef ITMAX
#undef EPS
/*****
*/
*/
*/
MAIN
/*****
*/
*/
*/
main( int argc, char *argv[] )
{
char para_format=atoi(argv[1]), out_format=atoi(argv[2]);
FILE *fp_in_file_array;
FILE *fp_e_data, *fp_m_data, *fp_m1_data, *fp_p_data;
FILE *fp_m_histo, *fp_m1_histo, *fp_coex;
char *file_name;
double *field, *fields;
int i;
long entries = 0;
double b_new, bh_new, bhs_new;
double temp, h, hs;
int j, k, file;
long line, factor;
double ww, m, ms;
long rho, rho1;
double common_fac;
double prob, part;
double eng, eng_1, eng_2, eng_3, eng_4;
double mag_1, mag_a, mag_2, mag_3, mag_4, cum;
double m1_1, m1_a, m1_2, m1_3, m1_4;
double *p, *p1, *p_less;
double *para_list, *coex_para_list;
int list_length, set;
char m_histo_file_name[F_OUT_NAME_LENGTH];
char m1_histo_file_name[F_OUT_NAME_LENGTH];
double h_coex;

Nr_of_points = get_nr_of_points(); /* Anzahl der Datenpunkte */
fp_in_file_array = calloc(Nr_of_points, sizeof(FILE *));
field = calloc(Nr_of_points, sizeof(double));

```



```

cum = 1.0 - mag_4 / (3.0 * mag_2 * mag_2);
if(out_format == OUT_DATA_AND_HISTO){
    sprintf(m_histo_file_name, "%w--DL%02dx%03d-t%lf-h%lf-hs%lf-hist-gas.rw",
            D, L, temp, h, hs);
    sprintf(m1_histo_file_name, "%M-DL%02dx%03d-t%lf-h%lf-hs%lf-hist-gas.rw",
            D, L, temp, h, hs);
    /* Rausschreiben des Magnetisierungshistogramms */
    open_for_write(m_histo_file_name, &fp_m_histo);
    p_less = 0.0;
    rho=0;
    p[rho] /= part;
    m = 1.0 - 2.0/N * rho;
    fprintf(fp_m_histo, "%ld %lf %lf %lf\n", rho, m, p[rho]);
    if(m <= mag_1)
        p_less += p[rho];
    for(rho=1; rho < N; rho++){
        p[rho] /= part;
        m = 1.0 - 2.0/N * rho;
        if(p[rho] > 0.0 || p[rho-1] > 0.0 || p[rho+1] > 0.0)
            fprintf(fp_m_histo, "%ld %lf %lf %lf\n", rho, m, p[rho]);
        if(m <= mag_1)
            p_less += p[rho];
    }
    p[rho] /= part;
    m = 1.0 - 2.0/N * rho;
    fprintf(fp_m_histo, "%ld %lf %lf %lf\n", rho, m, p[rho]);
    if(m <= mag_1)
        p_less += p[rho];
    close_file(&fp_m_histo);
    /* Rausschreiben des Oberflaechenmagnetisierungshistogramms */
    open_for_write(m1_histo_file_name, &fp_m1_histo);
    rho=0;
    p1[rho] /= part;
    m = 1.0 - 2.0/S * rho;
    fprintf(fp_m1_histo, "%ld %lf %lf %lf\n", rho, m, p1[rho]);
    for(rho=1; rho < S; rho++){
        p1[rho] /= part;
        m = 1.0 - 2.0/S * rho;
        if(p1[rho] > 0.0 || p1[rho-1] > 0.0 || p1[rho+1] > 0.0)
            fprintf(fp_m1_histo, "%ld %lf %lf %lf\n", rho, m, p1[rho]);
    }
    p1[rho] /= part;
    m = 1.0 - 2.0/S * rho;
    fprintf(fp_m1_histo, "%ld %lf %lf %lf\n", rho, m, p1[rho]);
    close_file(&fp_m1_histo);
}

if(out_format == OUT_DATA){
    line = 0;
    p_less = 0;
    for(file=0; file < Nr_of_points; file++){
        for(j=0; j < N_j[file]; j++){
            factor = NR_ROWS*line;
            ww = Histo[factor];
            m = Histo[factor + 1];
            ms = Histo[factor + 2];
            for(k=0, common_fac=0.0; k < Nr_of_points; k++){
                common_fac += Beta[k]*ww + B_h[k]*m + B_hs[k]*ms;
                zaehler = exp( b_new*ww + bh_new*m + bhs_new*ms - common_fac);
                for(k=0, nenner=0.0; k < Nr_of_points; k++){
                    nenner += (double)N_j[k]/Z[k] * Fac_tau[k] *
                        exp( Beta[k]*ww + B_h[k]*m + B_hs[k]*ms - common_fac );
                }
                prob = zaehler / nenner * Fac_tau[file];
                if(m/N <= mag_1)
                    p_less += prob/part;
            }
            line++;
        }
        fprintf(fp_e_data, "%10lf %10lf %10lf %10lf %10lf %10lf %10lf %10lf\n",
                temp, h, hs, eng_1, eng_2, eng_3, eng_4);
        fprintf(fp_m_data, "%10lf %10lf %10lf %10lf %10lf %10lf %10lf %10lf\n",
                temp, h, hs, mag_1, mag_2, mag_3, mag_4, cum, mag_a);
        fprintf(fp_m1_data, "%10lf %10lf %10lf %10lf %10lf %10lf %10lf %10lf\n",
                temp, h, hs, m1_1, m1_2, m1_3, m1_4, m1_a);
        fprintf(fp_p_data, "%10lf %10lf %10lf %10lf %10lf %10lf\n",
                temp, h, hs, p_less);
        fflush(NULL);
    }
    close_file(&fp_e_data);
    close_file(&fp_m_data);
    close_file(&fp_m1_data);
    close_file(&fp_p_data);
    free(para_list);
    free(fp_in_file_array);
    free(Fac_tau);
    free(N_j);
    free(Beta);
    free(B_h);
    free(field);
    free(B_hs);
    free(field_s);
    free(Z);
    free(Histo);
    free(p);
}

```

```
    free(p1);  
    return 0;  
}
```