

IMPLEMENTIERUNG DER

STOCHASTIC SERIES EXPANSION

FÜR SPIN $-1/2$ HEISENBERG SYSTEME

29. Juli 2011



Lukas B. Lentner

Vorwort

*»Man versteht etwas nicht wirklich,
wenn man nicht versucht, es zu implementieren.«*

— von DONALD ERVIN KNUTH [Knu02]

Lukas B. Lentner
kontakt@lukaslentner.de

München, 29. Juli 2011

Inhaltsverzeichnis

| | |
|---|-----------|
| Vorwort | 3 |
| 1 Einleitung | 7 |
| 2 Theorie der Monte Carlo Simulation | 9 |
| 2.1 Geschichte | 9 |
| 2.2 Ziel | 9 |
| 2.3 Markov-Kette | 10 |
| 2.4 Metropolis Algorithmus | 11 |
| 2.5 Thermalisierung | 11 |
| 2.6 Autokorrelationsfunktion und Fehlerberechnung | 11 |
| 3 Klassische MCS am Beispiel des Ising Modells | 13 |
| 3.1 Methode | 13 |
| 3.1.1 Das Ising-Modell | 13 |
| 3.1.2 Kanonische Übergangswahrscheinlichkeiten | 14 |
| 3.1.3 Messgrößen | 14 |
| 3.2 Implementierung | 15 |
| 3.3 Ergebnisse | 15 |
| 4 Quantenmechanische MCS mit Hilfe der Stochastic Series Expansion | 17 |
| 5 Zusammenfassung | 19 |
| A Quellcode | 21 |
| A.1 Hauptprogramm | 21 |
| A.2 Modelle | 21 |
| A.2.1 Abstraktes Modell | 21 |
| A.2.2 1D Modell mit offenen Randbedingungen | 22 |
| A.2.3 1D Modell mit periodischen Randbedingungen | 22 |
| A.2.4 2D Modell mit offenen Randbedingungen | 22 |
| A.2.5 2D Modell mit periodischen Randbedingungen | 22 |
| A.3 Algorithmen | 22 |
| A.3.1 Abstrakter Algorithmus | 22 |
| A.3.2 SSE | 22 |
| A.3.3 ED | 22 |
| Abbildungsverzeichnis | 23 |

| | |
|-----------------------------|-----------|
| Tabellenverzeichnis | 25 |
| Quellcodeverzeichnis | 27 |
| Literaturverzeichnis | 29 |

Kapitel 1

Einleitung

Kapitel 2

Theorie der Monte Carlo Simulation

2.1 Geschichte

Basierend auf den Ideen von Enrico Fermi (um 1935) verwendete zum ersten Mal Stanislaw Ulam und John von Neumann um 1945 das Prinzip der Monte Carlo Methode während ihrer Arbeit am Los Alamos Scientific Laboratory.

Der von Nicholas Metropolis gewählte Name bezieht sich auf die Spielbank Monte Carlo, die im gleichnamigen Stadtteil des Stadtstaates Monaco liegt. Anlass hierfür soll Ulams Onkel gegeben haben, der sich mehrmals von Verwandten Geld zum spielen leihen wollte [Met87].

Heute findet die Methode zahlreiche Anwendungen in der Statistischen Physik, Numerik und Optimierung.

2.2 Ziel

Die Idee der Monte Carlo Simulation (MCS) lässt sich beschreiben als ein gewichteter Weg durch einen n -dimensionalen Zustandsraum Ω . Hierbei interessiert man sich speziell für den statistischen Mittelwert einer Größe A ,

$$\langle A \rangle = \sum_{\sigma \in \Omega} p_{\sigma} \cdot A(\sigma) . \quad (2.1)$$

p_{σ} steht hier für die Wahrscheinlichkeit des Zustandes σ und $A(\sigma)$ ist der Wert der Größe A bei diesem Zustand. Für kontinuierliche Fälle ersetzt man die Summe durch ein Integral.

2.3 Markov-Kette

Oft ist es nicht möglich, die oben angegebene Summe auszuwerten (z.B. wenn Ω sehr groß ist). In diesem Fall kann der Zustandsraum quasidicht durch eine Markov-Kette von M Zuständen $\sigma_0, \sigma_1, \dots, \sigma_{M-1}$ abgelaufen werden. Die Häufigkeit eines Zustandes σ in der Kette soll im Grenzfalle $M \rightarrow \infty$ genau der Wahrscheinlichkeit des Zustandes p_σ entsprechen (Importance Sampling). Der Mitterwert kann sodann erheblich leichter nach dem Gesetz für große Zahlen durch das arithmetische Mittel über die Kette, also

$$\langle A \rangle \approx \bar{A} = \frac{1}{M} \sum_{m=0}^{M-1} A(\sigma_m) , \quad (2.2)$$

genähert werden.

Eine Markov-Kette beginnt mit einem beliebigen Anfangszustand σ_0 . Von diesem aus werden mit einer Übergangswahrscheinlichkeit $W_{\sigma_0\sigma_1}$ Sprünge im Zustandsraum ausgeführt (MC-Schritte), welche die neuen Kettenglieder $\sigma_2 \dots$ definieren. Damit die Markov-Kette zur gewünschten Wahrscheinlichkeitsverteilung führt, muss bei der Bildung von \mathbf{W} auf die zwei folgenden Bedingungen geachtet werden:

- a) Die Bildung der Kette muss **ergodisch** sein. D.h. sie muss theoretisch alle Zustände enthalten können, was sie in der Praxis natürlich nicht tut, da wir $M \ll |\Omega|$ wählen.
- b) Die Übergangswahrscheinlichkeiten \mathbf{W} müssen insofern im **Gleichgewicht** sein, als dass

$$\sum_{\sigma \in \Omega} p_\sigma \cdot W_{\sigma\nu} = p_\nu . \quad (2.3)$$

Eine deutlich stärkere Bedingung als b) stellt *Detailed Balance* (dt. detailliertes Gleichgewicht) dar,

$$p_\sigma \cdot W_{\sigma\nu} = p_\nu \cdot W_{\nu\sigma} . \quad (2.4)$$

In Worten besagt sie, dass ein Sprung von einem Markov-Kettenglied zum Nachbar genauso wahrscheinlich ist, wie andersherum. Die Kette besitzt also keine ausgezeichnete Richtung.

Gleichung 2.4 erfüllt automatisch Gl. 2.3, da

$$\sum_{\sigma \in \Omega} p_\sigma \cdot W_{\sigma\nu} = \sum_{\sigma \in \Omega} p_\nu \cdot W_{\nu\sigma} = p_\nu \cdot \sum_{\sigma \in \Omega} W_{\nu\sigma} = p_\nu . \quad (2.5)$$

Hierbei verwendet man im letzten Schritt, dass der Zustand ν in jedem Fall in irgendeinen nächsten Zustand σ übergeht. Diese zunächst starke Einschränkung wird häufig verwendet, um der Bedingung b) zu genügen. Später werden wir sehen, dass sie in unserem Fall auch die Berechnung von \mathbf{W} deutlich vereinfacht.

2.4 Metropolis Algorithmus

Ein möglicher Algorithmus zur Bestimmung der Übergangswahrscheinlichkeiten \mathbf{W} wurde 1953 von Nicholas Metropolis et al. vorgestellt [MRR⁺53],

$$W_{\nu\sigma} = \begin{cases} p_{\sigma}/p_{\nu} & p_{\sigma} < p_{\nu} \\ 1 & p_{\sigma} \geq p_{\nu} \end{cases} . \quad (2.6)$$

Es kann leicht gezeigt werden, dass der Vorschlag die *Detailed Balance* (Gl. 2.4) erfüllt. Ein weiterer Algorithmus ist nach Roj J. Glauber benannt (*Glauber dynamics*) [LLP10].

2.5 Thermalisierung

Nachdem als Anfangszustand der Markov-Kette ein beliebig ausgewählter Zustand verwendet wird, ist es ziemlich unwahrscheinlich, dass dieser Zustand ein hohes Wahrscheinlichkeitsgewicht P_{σ} besitzt. Es wird sich also nicht um einen Zustand im Gleichgewicht handeln. Aus diesem Grund sollte vor der eigentlichen Messung eine genügend große Anzahl von Thermalisierungsschritten (MC-Schritte) durchgeführt werden.

In der Praxis werden entweder Erfahrungswerte verwendet, die eine konstante, meist zu große Schrittzahl erfordern oder die Daten werden vollständig gespeichert und in der Auswertung sortiert. Im Nachhinein kann über tatsächliche Wahrscheinlichkeitsverteilung auf die Thermalisierungsphase geschlossen werden. Diese Daten werden dann für die anschließende Analyse nicht verwendet.

2.6 Autokorrelationsfunktion und Fehlerberechnung

Alle Messwerte der Größe A müssen nach der Thermalisierung in der Auswertung statistisch interpretiert werden. Dabei ist zu beachten, dass die Daten von aufeinanderfolgenden Zuständen statistisch abhängig sind. Wie viele MC-Schritte zwischen zwei Messungen notwendig sind, um unabhängige Werte zu erhalten, gibt die Autokorrelationszeit τ_A an (Im weiteren ist mit „Zeit“ immer die Simulationszeit gemessen in MC-Schritten gemeint). Zur Berechnung derselben wird die Autokorrelationsfunktion

$$\Theta_A(t) = \frac{\langle A(\sigma_{i+t}) \cdot A(\sigma_i) \rangle - \langle A \rangle^2}{\langle A^2 \rangle - \langle A \rangle^2} \quad (2.7)$$

betrachtet. Hierbei läuft die Mittelwertbildung mit der Variable i über die gesamte ausgewertete Simulationszeit \bar{M} (Die gesamte Simulationszeit beträgt M). Sie ist in solch einer Weise normiert, dass $\Theta_A(\sigma_0) = 1$ und $\Theta_A(\sigma_{t \rightarrow \infty}) = 0$. Die Autokorrelationsfunktion hängt sodann mit der Autokorrelationszeit negativ exponentiell zusammen,

$$\Theta_A(t) \sim e^{-t/\tau_A} . \quad (2.8)$$

Nachdem τ_A auf diese Art ermittelt wurde, können die Messwerte in Gruppen mit der Länge $3 \cdot \tau_A$ gebündelt und unabhängige Gruppenmittelwerte

$$\bar{A}_b = \sum_{i=0}^{3\tau_A} A(\sigma_{b \cdot \tau_A + i}) \quad (2.9)$$

berechnet werden, wobei b hier der null-basierte Gruppen-Index ist und die Anzahl der Gruppen

$$B = \left\lfloor \frac{\widetilde{M}}{3\tau_A} \right\rfloor. \quad (2.10)$$

Nach dem zentralen Grenzwert Satz folgen diese Gruppenmittelwerte sodann einer Gauß-Verteilung, welche den exakten Wert

$$\bar{A} = \frac{1}{B} \sum_{b=0}^{B-1} \bar{A}_b \quad (2.11)$$

in der Mitte hält. Als Fehler kann eine Standardabweichung, also

$$\sigma_A = \sqrt{\frac{1}{B(B-1)} \sum_{b=0}^{B-1} (\bar{A}_b - \bar{A})^2} \quad (2.12)$$

angegeben werden.

Kapitel 3

Klassische MCS am Beispiel des Ising Modells

Um die Grundlagen einer Monte Carlo Simulation (MCS) kennenzulernen, wurde im Vorfeld eine Anwendung zur Simulation des 2-dimensionalen Ising-Modells mit periodischer Randbedingung erstellt. Die gewonnene Erfahrung erwies sich für das Studium der quantenmechanischen MCS als äußerst hilfreich.

3.1 Methode

3.1.1 Das Ising-Modell

Für das klassische, ferromagnetische Ising-Modell ist der Hamiltonian

$$H_{\text{Ising}} = - \sum_{\langle i,j \rangle} J_{ij} \cdot S_i^z S_j^z - B\mu \sum_{i=0}^{N-1} S_i^z \quad (3.1)$$

zusammengesetzt aus einer z -Koppelung benachbarter Spins $\langle i,j \rangle$ (wird später über das Gittermodell definiert), die durch die Bindungsmatrix \mathbf{J} gewichtet wird, und einer magnetischen Wechselwirkung, in die das externe Magnetfeld $\mathbf{B} = (0, 0, B)^T$ und das Magnetische Moment $\boldsymbol{\mu} = (0, 0, \mu)^T$ eingeht. Für unser Beispiel setzten wir jedes $J_{ij} = 1$ und betrachten die Anordnung ohne Magnetfeld – da uns nur die z -Richtung interessiert setzen wir $S = S^z$. Der Hamiltonian erhält dann die vereinfachte Struktur,

$$H = - \sum_{\langle i,j \rangle} S_i S_j . \quad (3.2)$$

3.1.2 Kanonische Übergangswahrscheinlichkeiten

Wegen der vorgegebenen Teilchenanzahl N und Temperatur T setzen wir für eine beliebige Größe A den Mittelwert

$$\langle A \rangle = \sum_{\sigma \in \Omega} \frac{e^{-\beta E_\sigma}}{Z} \cdot A(\sigma) \quad (3.3)$$

kanonisch an, wobei

$$Z = \sum_{\sigma \in \Omega} e^{-\beta E_\sigma} \quad (3.4)$$

die kanonische Zustandssumme, β die reduzierte Temperatur $1/T$ und E_σ die Energie eines gewissen mikroskopischen Zustandes σ (Konfiguration) darstellt. Im Vergleich zur Gl. 2.1 sieht man, dass die Wahrscheinlichkeit eines Zustandes σ

$$p_\sigma = \frac{e^{-\beta E_\sigma}}{Z} \quad (3.5)$$

boltzmannverteilt ist. Diese Gewichte sind schwer zu berechnen, da der Zustandsraum in solch einem Spin-System exponentiell mit der Spinanzahl anwächst ($|\Omega| \sim 2^N$). Machen wir allerdings vom Metropolis Algorithmus Gebrauch (siehe Gl. 2.6), benötigen wir die einzelnen Gewichte gar nicht, sondern können uns mit deren Verhältnissen, die dann die Übergangswahrscheinlichkeiten darstellen, genügen,

$$W_{\nu\sigma} = \begin{cases} e^{-\beta(E_\sigma - E_\nu)} & E_\sigma > E_\nu \\ 1 & E_\sigma \leq E_\nu \end{cases} \quad (3.6)$$

3.1.3 Messgrößen

Folgende typischen, thermodynamischen Größen wollen wir in unserer Beispielanwendung messen:

$$\text{Energie : } E = \langle H \rangle , \quad (3.7)$$

$$\text{Magnetisierung pro Spin : } M = \langle S_i \rangle , \quad (3.8)$$

$$\text{Spezifische Wärme : } C_V = \frac{1}{NkT^2} \left(\langle H^2 \rangle - \langle H \rangle^2 \right) , \quad (3.9)$$

$$\text{Magnetische Suszeptibilität : } \chi_m = \frac{N}{kT} \left(\langle S_i^2 \rangle - \langle S_i \rangle^2 \right) . \quad (3.10)$$

3.2 Implementierung

Hallo

3.3 Ergebnisse

relaxation time goes up for transition

aussicht cluster alg

finite size scaling critical exp

Vergleich mit Meanfield

Kapitel 4

Quantenmechanische MCS

mit Hilfe der Stochastic Series Expansion

Kapitel 5

Zusammenfassung

Anhang A

Quellcode

A.1 Hauptprogramm

A.2 Modelle

A.2.1 Abstraktes Modell

```
1  #ifndef CLASS_ABSTRACTMODEL
2  #define CLASS_ABSTRACTMODEL
3
4  class AbstractModel {
5
6      protected:
7
8          int n;
9
10     public:
11
12         AbstractModel(int n_parameter) {
13
14             n = n_parameter;
15
16         };
17
18         int getN() {
19
20             return n;
21
22         };
23
24         virtual int getNb() = 0;
25         virtual int getI1(int b) = 0;
26         virtual int getI2(int b) = 0;
27
28     };
29
30 #endif
```

A.2.2 1D Modell mit offenen Randbedingungen

A.2.3 1D Modell mit periodischen Randbedingungen

A.2.4 2D Modell mit offenen Randbedingungen

A.2.5 2D Modell mit periodischen Randbedingungen

A.3 Algorithmen

A.3.1 Abstrakter Algorithmus

A.3.2 SSE

A.3.3 ED

Abbildungsverzeichnis

Tabellenverzeichnis

Quellcodeverzeichnis

| | | |
|-----|---|----|
| A.1 | Classes/Model/AbstractModel.cpp | 21 |
|-----|---|----|

Literaturverzeichnis

- [Knu02] Donald E. Knuth. Der perfektionist. *heise.de - c't*, page 190, May 2002.
- [LLP10] David Levin, Malwina Luczak, and Yuval Peres. Glauber dynamics for the mean-field ising model: cut-off, critical power law, and metastability. *Probability Theory and Related Fields*, 146:223–265, 2010.
- [Met87] Nicholas Metropolis. The beginning of the monte carlo method. *Los Alamos Science*, 15:125–130, 1987.
- [MRR⁺53] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equations of state calculations by fast computing machines. *The Journal Of Chemical Physics*, 21(6):1087–1092, 1953.