

# Kapitel 2

## Simulationstechniken

### 2.1 Importance sampling: Metropolis-Algorithmus

Eine zentrale Fragestellung in der statistischen Physik ist die Bestimmung von Erwartungswerten einer Observablen  $O$  in einem wohldefinierten Ensemble. Da die Anzahl aller möglichen Konfigurationen mit steigender Teilchenzahl schnell sehr groß wird, führt eine einfache Mittelung über alle Zustände nicht zum Erfolg. Würde man z.B. eine Simulationsbox entlang jeder Achse in lediglich zehn Abschnitte unterteilen, erhält man bei zehn Teilchen in  $d = 3$  Dimensionen etwa  $10^{30}$  Möglichkeiten. Selbst wenn alle Teilchen gleichartig sind und man lediglich unterschiedliche Konfigurationen berücksichtigt, liegt deren Gesamtzahl immer noch bei  $2.6 \cdot 10^{23}$  und somit jenseits jeglicher heute zugänglicher Rechenleistung. Die Bestimmung eines Erwartungswertes muss folglich durch eine Mittelung über eine endliche Auswahl von Zuständen erfolgen. Da der mit Abstand größte Teil der Konfigurationen lediglich einen vernachlässigbaren Beitrag zum Mittelwert liefert, führt eine zufällige Auswahl („simple“ oder „random sampling“) nicht zum Erfolg. Vielmehr sollte das statistische Gewicht eines Zustand  $S_i$  berücksichtigt werden, welches in einem klassischen System durch die Boltzmannverteilung gegeben ist. Im kanonischen Ensemble ist:

$$P(S_i) = \frac{e^{-\beta E(S_i)}}{Z}. \quad (2.1)$$

$E$  entspricht der Energie der Konfiguration und  $Z$  der kanonischen Zustandssumme. Der kinetische Anteil spielt bei den folgenden Betrachtungen keine Rolle und kann z.B. in  $Z$  hineingezogen werden. Die Monte-Carlo-Simulation versucht nun Konfigurationen zu erzeugen, die einer vorgegebenen Wahrscheinlichkeitsverteilung (z.B. (??)) genügen. Der im folgenden Abschnitt vorgestellte Algorithmus wurde 1953 von *Metropolis et al.* veröffentlicht (N. Metropolis, et al., J. Chem. Phys. 21(6), 1087

(1953)). Die hierauf aufbauenden Monte–Carlo–Techniken bilden zusammen mit der Molekulardynamik–Methode (MD) die Grundlagen klassischer molekularer Simulation. Die Darstellung des Algorithmus folgt im Wesentlichen der Darstellung in Landau / Binder, *A Guide to Monte Carlo Simulations in Statistical Physics*.

Zustände, die einer gegebenen Verteilung genügen, können mit Hilfe einer Markov–Kette realisiert werden. Hierzu definiere man einen stochastischen Prozess mit diskreten Zeitschritten  $t_1, t_2, \dots$  für ein System mit einer endlichen Anzahl von Zuständen  $S_1, S_2, \dots$ .  $X_t$  sei der Zustand eines Systems zum Zeitpunkt  $t$ . Ferner betrachte man:

$$P(X_{t_n} = S_{i_n} | X_{t_{n-1}} = S_{i_{n-1}}, X_{t_{n-2}} = S_{i_{n-2}}, \dots, X_{t_1} = S_{i_1}), \quad (2.2)$$

also die bedingte Wahrscheinlichkeit, dass sich das momentane System ( $X_{t_n} = S_{i_n}$ ) zum Zeitpunkt  $t_{n-1}$  im Zustand  $S_{i_{n-1}}$ , zum Zeitpunkt  $t_{n-2}$  im Zustand  $S_{i_{n-2}}$  usw. befand. Eine Abfolge von Zuständen  $\{X_t\}$  bezeichnet man als Markov–Kette, falls die bedingte Wahrscheinlichkeit das System zum Zeitpunkt  $t_n$  im Zustand  $S_i$  anzutreffen lediglich vom direkten Vorgängerzustand  $X_{t_{n-1}}$  abhängt:

$$P = P(X_{t_n} = S_{i_n} | X_{t_{n-1}} = S_{i_{n-1}}) =: W_{i_{n-1}i_n} = W(S_{i_{n-1}} \rightarrow S_{i_n}). \quad (2.3)$$

Die „Vergangenheit“ vor  $t_{n-1}$  hat also keinen Einfluss auf die Eintrittswahrscheinlichkeit von  $X_{t_n}$  und die weitere Entwicklung der Kette. Wie in Gleichung (??) angedeutet, lässt sich  $P$  auch als Übergangswahrscheinlichkeit  $W_{ij}$  vom Zustand  $i$  nach  $j$  interpretieren. Ist diese unabhängig von  $t$ , bezeichnet man den Prozess als stationär. Als Sprungwahrscheinlichkeit ist  $W_{ij}$  immer größer Null. Die Wahrscheinlichkeit von  $i$  in einen beliebigen Zustand  $j$  zu springen und somit  $\sum_j W_{ij}$  ist gleich Eins.

Die so genannte „master equation“ betrachtet die Änderung der bedingten Wahrscheinlichkeit als Funktion der Zeit. Der Fluss von einem Zustand  $i$  in einen Zustand  $j$  ist gegeben durch das Produkt aus der Wahrscheinlichkeit  $P_i := P(X_{t_n} = S_i)$ , dass sich das System zum Zeitpunkt  $t$  im Zustand  $i$  befindet, mit der Sprungwahrscheinlichkeit  $W_{ij}$  von  $i$  nach  $j$ . Summiert man über alle Zustände  $i$ , erhält man den gesamten Zufluss nach  $j$ . Der Abfluss aus  $j$  ist entsprechend gegeben durch die Summe über alle Produkte  $P_j W_{ji}$ . Insgesamt erhält man also

$$\frac{dP_j(t)}{dt} = \sum_i P_i(t) W_{ij} - \sum_j P_j(t) W_{ji}. \quad (2.4)$$

Gleichung ?? kann als „Kontinuitätsgleichung“ aufgefasst werden. Die Gesamtwahrscheinlichkeit ( $\sum_j P_j$ ) bleibt zu jedem Zeitpunkt erhalten. Im Gleichgewichtszustand ist der Zufluss in einen Zustand gleich dem Abfluss aus dem Zustand, d.h.  $dP_j/dt = 0$ . Diese Bedingung ist auf jeden Fall erfüllt, falls gilt:

$$P_i(t) W_{ij} = P_j(t) W_{ji}. \quad (2.5)$$

Die strengere Forderung, dass zwei beliebige Zustände zueinander im Gleichgewicht stehen, bezeichnet man als „detailed balance“.

Die folgende Diskussion bezieht sich auf das kanonische Ensemble. Obwohl die Wahrscheinlichkeit in Gleichung (??) wegen der Zustandssumme im Nenner in der Regel nicht genau bestimmbar ist, kann man dennoch das Verhältnis zweier Wahrscheinlichkeiten angeben.  $Z$  kürzt sich dann in Gleichung (??) weg und es wird lediglich die Differenz in der Energie der beiden Zustände  $\Delta E = E_j - E_i$  benötigt. Jede Übergangsrate, die die Detailed-Balance-Bedingung erfüllt, ist prinzipiell akzeptabel. Das erste solche Kriterium, welches in der statistischen Physik bei der Simulation harter Scheiben Anwendung fand, stammt von Metropolis et al.:

$$W_{ij} = \begin{cases} \exp(-\beta\Delta E) & , \text{ falls } \Delta E > 0 \\ 1 & , \text{ falls } \Delta E < 0. \end{cases} \quad (2.6)$$

oder kürzer:

$$W_{ij} = \min(1, \exp(-\beta\Delta E)) =: \text{metrop}(\exp(-\beta\Delta E)). \quad (2.7)$$

Ein Sprung auf ein energetisch tieferes Niveau wird demnach immer akzeptiert. Ein Sprung auf ein höheres Niveau jedoch nur mit einer exponentiellen Wahrscheinlichkeit. Diese wird umso geringer, je größer die Energiezunahme ausfällt. Die Überprüfung der Detailed-Balance-Bedingung erfolgt durch das Einsetzen von Bedingung (??) in (??). Der Metropolis-Algorithmus kann auf einfache Weise implementiert werden. Das System befinde sich hierzu im Ausgangszustand  $i$ :

- |: 1. Wähle zufällig einen neuen Zustand  $j$  aus.
2. Berechne die Energiedifferenz  $\Delta E$  zwischen  $j$  und dem alten Zustand  $i$ .
3. Ziehe eine Zufallszahl  $z$  aus  $[0,1]$ .
4. Falls  $z < \exp(-\beta\Delta E)$ , akzeptiere den Schritt und führe das System in den neuen Zustand  $j$  über. :|

In einer typischen Simulation wird die obere Abfolge sehr oft wiederholt. Nach einer kurzen Relaxationsphase in der sich das System ins Gleichgewicht bewegt, können Messungen einer Observablen vorgenommen werden. Da die erzeugten Zustände bereits der Boltzmannverteilung genügen, berechnet sich der Erwartungswert  $\langle O \rangle = \sum_i P_i O_i$ , einfach aus dem arithmetischen Mittel.

Im Vergleich mit einer Molekulardynamik-Simulation hat Monte Carlo den großen Vorteil, dass Sprünge von einem Zustand in den nächsten beliebig „unphysikalisch“ ausfallen können. Statistisch unabhängige Konfigurationen können so auf Kosten einer unrealistischen Dynamik sehr viel schneller erzeugt werden. Es folgen einige Beispiele.

### 2.1.1 Lokale Verrückungen

Eine der einfachsten und ältesten Übergänge ist die lokale Verrückung eines Polymer- oder Lösungsmittelteilchens. Hierbei wählt man zufällig ein Teilchen im System aus und verschiebt es um eine bestimmte Länge und Orientierung (Abb. ??). Im Anschluss

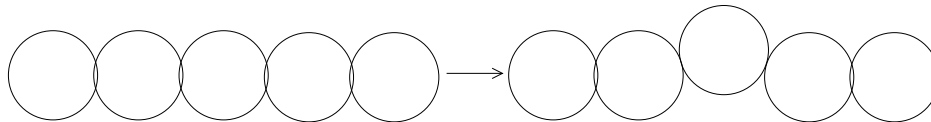


Abbildung 2.1: Schematische Darstellung eines lokalen Monte-Carlo-Schrittes.

wird die durch die Verrückung verursachte Energiedifferenz berechnet und der Zug gemäß des Metropolis-Kriteriums akzeptiert oder abgelehnt. Der lokale Übergang ist zur Äquilibration einer kanonischen Konfiguration (konstante Teilchenzahl in der Box) geringer oder mittlerer Dichte nur bedingt geeignet. In einer dichten Phase ist er hingegen einer der wenigen Schritte, die überhaupt akzeptiert werden.

### 2.1.2 Reptation

Beim Reptationsalgorithmus, der manchmal auch als Slithering-Snake-Algorithmus bezeichnet wird, schneidet man ein Endmonomer einer zufällig ausgewählten Kette ab und fügt es an der gegenüberliegenden Seite wieder an (Abb. ??). Dies bewirkt eine

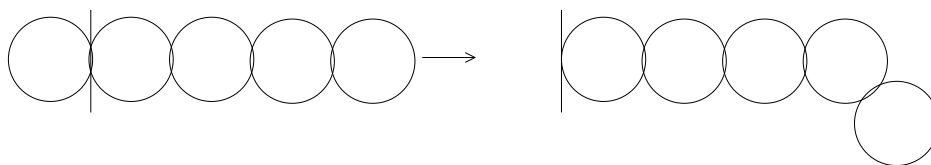


Abbildung 2.2: Schematische Darstellung eines Reptationsschrittes.

„schlangenartige“ Kriechbewegung des Polymers. Die Bindungslänge bleibt dabei in der einfachsten Ausführung erhalten. Die Implementierung des Algorithmus erfolgt wie bei der lokalen Verrückung über das Metropolis-Kriterium. Der Slithering-Snake-Algorithmus ist ein gutes Beispiel für einen „unphysikalischen“ Monte-Carlo-Schritt. Er ermöglicht jedoch bei geringen und mittleren Dichten eine wesentlich effizientere Äquilibration eines Kettenmoleküls als die lokale Verrückung.

### 2.1.3 Pivot

Bei der Simulation von nichtkollabierten Einzelketten leistet der Pivot-Algorithmus hervorragende Dienste. In der einfachsten Implementierung wählt man zufällig ein Monomer als Drehzentrum aus und dreht einen Teil der Kette angefangen von diesem Monomer bis zu einem Ende der Kette um einen zufällig ausgewählten Winkel. Auf diese Weise kann man in wenigen Zügen eine statistisch unkorrelierte Konfiguration erzeugen. Eine besonders effiziente Implementierung findet sich in T. Kennedy, *J.Statist.Phys.* 106, 407 (2002). In dichten Systemen, wie z.B. bei kollabierten Polymeren versagt der Algorithmus.