

Markov Chain Monte Carlo (MCMC)

Maximilian Rettinger

03. Februar 2020

1 Begriffsklärung

Monte Carlo bezeichnet ein stochastisches Verfahren, dessen Basis eine große Zahl gleichartiger Zufallsexperimente bilden. Es findet Anwendung in solchen mathematischen Fragestellungen, in denen eine analytische oder eine exakte numerische Lösung nur sehr aufwendig, häufig aber auch gar nicht erreicht werden kann. Mit Monte Carlo Methoden kann die Lösung oft dennoch zumindest numerisch approximiert werden.

Der Begriff der Markov Kette bezeichnet einen stochastischen Prozess, gegeben durch eine Folge X_1, X_2, \dots zufälliger Elemente einer Menge. Die bedingte Wahrscheinlichkeitsverteilung von X_{n+1} hängt einzig vom vorherigen Folgenglied X_n ab. Hängt sie des weiteren nicht von n ab, so liegen stationäre Übergangs- oder Transitionswahrscheinlichkeiten vor. Dieser Fall ist von besonderem Interesse in MCMC und wir für den Rest dieses Vortrages stets vorausgesetzt. Die Werte, welche von X_i mit $i \in \mathbb{N}$ angenommen werden können, liegen im sogenannten Zustandsraum.

Markov Chain Monte Carlo bezeichnet einen Spezialfall der Monte Carlo Methoden, in welchem die stochastische Abhängigkeit innerhalb der Markov Ketten eine große Rolle spielt [1]. Der große Vorteil dieser Methode tritt dann zutage, wenn ein mathematisches Problem gelöst werden soll, welches nur sehr schwer und häufig gar nicht analytisch sowie oft auch nur sehr aufwendig exakt numerisch gelöst werden kann. Rechenkapazität wird dadurch eingespart, dass immer nur der aktuelle Zustand der verwendeten Markov Kette gespeichert werden muss, niemals alle Zustände auf einmal berücksichtigt werden müssen. Eine solche Markov Kette wird künstlich passend zum betrachteten Problem konstruiert. Zu erklären wie das von statten geht sowie wie genau der erwähnte Vorteil erreicht wird soll das Ziel dieses Vortrages sein.

2 Grundidee von MCMC

Dieser Abschnitt beginnt mit einer kurzen Auffrischung der benötigten theoretischen Grundlagen, in diesem Fall spezielle Eigenschaften von Markov Ketten sowie darauf aufbauende Sätze [3]. Letztere werden allerdings nicht im Zuge dieses Vortrages bewiesen, da sie einerseits den zeitlichen Rahmen sprengen würden und andererseits bereits in den vorangegangenen Vorträgen des Seminars ausführlicher behandelt worden sein sollten. Zugleich wird eine recht einfacher Anwendungsmöglichkeit von MCMC vorbereitet und erklärt. So einfach zwar, dass das noch zu beschreibende Problem in der Praxis niemals auf diese Weise gelöst wird. Trotzdem eignet es sich gut, um die Grundidee von MCMC hervorzuheben.

Definition 1 *Irreduzibilität*

Sei (X_0, X_1, \dots) eine Markov Kette mit Zustandsraum $S = \{s_1, \dots, s_k\}$ mit $k \in \mathbb{N}$ und Transitionsmatrix P . Ein Zustand $s_i \in S$ *kommuniziert* mit einem anderen Zustand s_j , kurz $s_i \rightarrow s_j$, falls ein $n \in \mathbb{N}$ existiert, sodass

$$P(X_{m+n} = s_j | X_m = s_i) = (P^n)_{i,j} > 0.$$

Die Markov Kette heißt *irreduzibel* genau dann, wenn für alle Zustände $s_i, s_j \in S$ gilt $s_i \rightarrow s_j$ sowie $s_j \rightarrow s_i$. Ansonsten wird die Kette als *reduzibel* bezeichnet.

Anschaulich gesprochen ist eine Markov Kette irreduzibel, wenn jeder Zustand von jedem anderen Zustand aus erreicht werden kann. Im Falle von endlichen Markov Ketten ist dies oft bereits mit dem bloßen Auge erkennbar.

Abbildung 1 zeigt ein einfaches Beispiel einer irreduziblen Markov Kette.

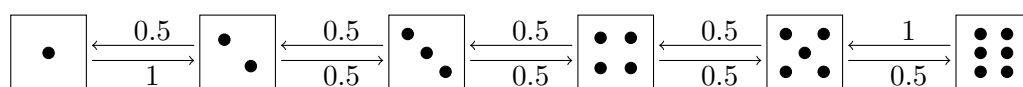


Abbildung 1: Der Transitionsgraph einer irreduziblen Markov Kette mit sechs Zuständen. Zum strengen Nachweis der Irreduzibilität anhand der Definition muss für jedes Paar $i, j = 1, \dots, 6$ ein $n \in \mathbb{N}$ gefunden werden, sodass $(P^n)_{i,j} > 0$. Dass dies auch tatsächlich der Fall ist, kann leicht anhand des Graphen gesehen werden. Jeder Zustand kommuniziert mit einem beliebigen anderen Zustand, da letzterer stets durch eine geeignete Hintereinandereiung der einzelnen Übergänge mit strikt positiver Wahrscheinlichkeit erreicht werden kann.

Definition 2 *Aperiodizität*

Sei (X_0, X_1, \dots) eine Markov Kette mit Zustandsraum $S = \{s_1, \dots, s_k\}$ mit $k \in \mathbb{N}$ und Transitionsmatrix P . Die *Periode* d eines Zustandes $s_i \in S$ ist wie folgt definiert durch den größten gemeinsamen Teiler (ggT):

$$d(s_i) := \text{ggT}(\{n \geq 1 : (P^n)_{i,i} > 0\}).$$

Der Zustand s_i heißt *aperiodisch* genau dann, wenn $d(s_i) = 1$. Die Markov Kette kann ebenfalls als aperiodisch bezeichnet werden, insofern all ihre Zustände aperiodisch sind. Ansonsten handelt es sich um eine *periodische* Markov Kette.

Die bereits erwähnte Abbildung 1 zeigt eine periodische Markov Kette. Um das zu erkennen, muss nicht einmal die Transitionsmatrix P aufgestellt werden. Es reicht aus einen Zustand s_i zu finden, welcher eine Periode $d(s_i) > 1$ besitzt. Tatsächlich erfüllen dies alle Zustände der gegebenen Markov Kette. Von einem beliebigen Zustand aus wird stets eine gerade Anzahl von Transitionen benötigt, bevor derselbe Zustand wieder erreicht werden kann. Die kleinste Zahl $n \in \mathbb{N}$, sodass $(P^n)_{i,i} > 0$, ist für jeden Zustand die 2, nämlich genau dann, wenn von dem betrachteten Zustand ein benachbarter Zustand (d.h. einer, der durch einen einzelnen Übergang erreichbar ist) besucht und danach direkt wieder der Ausgangszustand aufgesucht wird.

Durch eine kleine Anpassung kann eine aperiodische Markov Kette erzeugt werden, dargestellt in Abbildung 2.

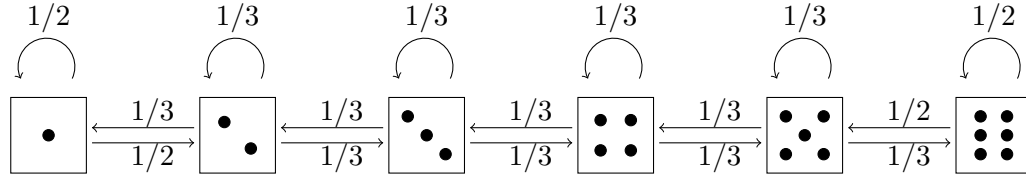


Abbildung 2: Eine irreduzible sowie aperiodische Markov Kette. Jeder Zustand kann innerhalb nur eines einzigen Überganges wieder erreicht werden, deshalb ist die Periode stets gleich 1.

Definition 3 *Stationäre Verteilung*

Sei (X_0, X_1, \dots) eine Markov Kette mit Zustandsraum $\{s_1, \dots, s_k\}$ mit $k \in \mathbb{N}$ und Transitonsmatrix P . Ein Zeilenvektor $\pi = (\pi_1, \dots, \pi_k)$ heißt *stationäre Verteilung* der Markov Kette, falls gilt

- (i) $\pi_i \geq 0$ für $i = 1, \dots, k$ und $\sum_{i=1}^k \pi_i = 1$ und
- (ii) $\pi P = \pi$, d.h. $\sum_{i=1}^k \pi_i P_{i,j} = \pi_j$ für $j = 1, \dots, k$.

Eine der wesentlichen Voraussetzungen dafür, dass mit MCMC brauchbare Ergebnisse berechnet werden können, ist die Existenz einer stationären Verteilung. Diese ist, unter gegebenen Annahmen, durch den folgenden Satz sichergestellt.

Satz 1 *Existenz stationärer Verteilungen*

Ist eine Markov Kette irreduzibel sowie aperiodisch, so besitzt sie mindestens eine stationäre Verteilung.

Die Markov Kette in Abbildung 2 besitzt eine stationäre Verteilung

$$\pi = (1/8, 3/16, 3/16, 3/16, 3/16, 1/8).$$

Die Bedingungen der Definition lassen sich leicht überprüfen. Es gilt offensichtlich $\pi_i \geq 0$ für $i = 1, \dots, 6$ sowie $\sum_{i=1}^6 \pi_i = 2/16 + 3/16 + 3/16 + 3/16 + 3/16 + 2/16 = 1$. Die etwas kompliziertere Rechnung ist ebenfalls recht schnell erledigt:

$$\pi P = \left(\frac{1}{8}, \frac{3}{16}, \frac{3}{16}, \frac{3}{16}, \frac{3}{16}, \frac{1}{8}\right) \cdot \begin{pmatrix} 1/2 & 1/2 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 1/3 & 0 & 0 & 0 \\ 0 & 1/3 & 1/3 & 1/3 & 0 & 0 \\ 0 & 0 & 1/3 & 1/3 & 1/3 & 0 \\ 0 & 0 & 0 & 1/3 & 1/3 & 1/3 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \end{pmatrix} = \left(\frac{1}{8}, \frac{3}{16}, \frac{3}{16}, \frac{3}{16}, \frac{3}{16}, \frac{1}{8}\right) = \pi.$$

Wie sich mit dem folgenden Satz herausstellen wird, handelt es sich bei π sogar um die einzige stationäre Verteilung der Markov Kette aus Abbildung 2. Zuerst wird die Definition der verwendeten Metrik sowie des Konvergenzbegriffes der Vollständigkeit halber aufgeführt.

Definition 4 *Konvergenz nach dem Variationsabstand*

Gegeben sei der Zustandsraum $S = \{s_1, \dots, s_k\}$ für ein $k \in \mathbb{N}$ sowie zwei Wahrscheinlichkeitsverteilungen $\nu^{(1)} = (\nu_1^{(1)}, \dots, \nu_k^{(1)})$ und $\nu^{(2)} = (\nu_1^{(2)}, \dots, \nu_k^{(2)})$ auf S . Der Variationsabstand d_V zwischen $\nu^{(1)}$ und $\nu^{(2)}$ sei definiert als:

$$d_V(\nu^{(1)}, \nu^{(2)}) := \frac{1}{2} \sum_{i=1}^k \left| \nu_i^{(1)} - \nu_i^{(2)} \right|.$$

Seien nun durch $\nu^{(1)}, \nu^{(2)}, \dots, \nu$ Wahrscheinlichkeitsverteilungen auf S gegeben. Die Folge $\nu^{(n)}$ konvergiert gegen ν nach dem Variationsabstand für $n \rightarrow \infty$, kurz $\nu^{(n)} \xrightarrow{V} \nu$, falls

$$\lim_{n \rightarrow \infty} d_V(\nu^{(n)}, \nu) = 0.$$

Damit sind alle Voraussetzungen beisammen, um den folgenden Konvergenzsatz formulieren zu können.

Satz 2 *Konvergenztheorem*

Gegeben sei eine irreduzible und aperiodische Markov Kette (X_0, X_1, \dots) mit Zustandsraum $S = \{s_1, \dots, s_k\}$, Transitionsmatrix P , einer beliebigen Anfangsverteilung μ_0 sowie einer stationären Verteilung π . Dann handelt es sich bei π um die einzige stationäre Verteilung der Markov Kette und für die rekursiv definierte Folge von Wahrscheinlichkeitsverteilungen $\mu^{(n+1)} = \mu^{(n)} \cdot P$ für $n \in \mathbb{N}_0$ gilt

$$\mu^{(n)} \xrightarrow{V} \pi.$$

Dank dieses nützlichen Theorems ist es möglich, die gesuchte Wahrscheinlichkeitsverteilung einer Größe in einem beliebigen Problem mithilfe eines Algorithmus der Klasse MCMC anzunähern, solange man nur eine geeignete Markov Kette dazu modellieren kann. Insbesondere werden dadurch Probleme lösbar, welche zuvor weder analytisch noch numerisch exakt beantwortet werden konnten. Ein solches Beispiel folgt im weiteren Verlauf des Vortrages. Zunächst wird aus Gründen der Anschaulichkeit auf ein sehr einfaches Problem eingegangen, dessen analytische Lösung nicht nur existent, sondern auch bereits sehr lange bekannt ist. Es soll daher vielmehr als Spielwiese dienen, um zumindest einen Teil des Prinzips der MCMC-Methodik in einem möglichen Anwendungsfall sichtbar zu machen.

Beispiel: Wahrscheinlichkeitsverteilung des Würfels

Betrachtet wird ein herkömmlicher Würfel, wie er aus unterschiedlichen Spielen bekannt ist. Gesucht sei nun die Wahrscheinlichkeitsverteilung der einzelnen Seiten eines solchen Würfels. Um dies nach dem Schema MCMC zu erreichen, muss zunächst eine irreduzible sowie aperiodische Markov Kette für die Fragestellung entworfen werden. Klar ist, dass der Raum der möglichen Zustände genau sechs Elemente enthält. Die Erzeugung einer irreduziblen Markov Kette wie in Abbildung 1 ist ein weiterer einfacher Schritt. Zur Erfüllung der Forderung nach Aperiodizität ist auch die Anpassung wie in Abbildung 2 schnell getan. Im weiteren Verlauf dieses Beispiels war das Wissen, dass die gesuchte Wahrscheinlichkeitsverteilung eine Gleichverteilung ist, ausschlaggebend für den endgültigen Entwurf. Abbildung 3 zeigt den Kandidaten für die Markov Kette, deren stationäre Verteilung die Gleichverteilung ist. Dies lässt sich ganz analog zur Verifizierung der stationären Verteilung im Fall der Kette aus Abbildung 2 beweisen. Es folgt ein Code Beispiel, in der Programmiersprache Python geschrieben. Das Vorgehen besteht aus einer Initialisierung des Anfangszustandes mit einer willkürlichen Verteilung. Im Anschluss daran wird N mal die gleiche Update Funktion auf den jeweils aktuellen Zustand angewendet. In diesem Fall sind die Wahrscheinlichkeiten für die drei möglichen Änderungen pro Zustand stets dieselben, im Allgemeinen ist das jedoch nicht der Fall. Im Speicher wird stets nur der aktuelle Zustand gehalten, zusammen mit einem Array, welcher die absoluten

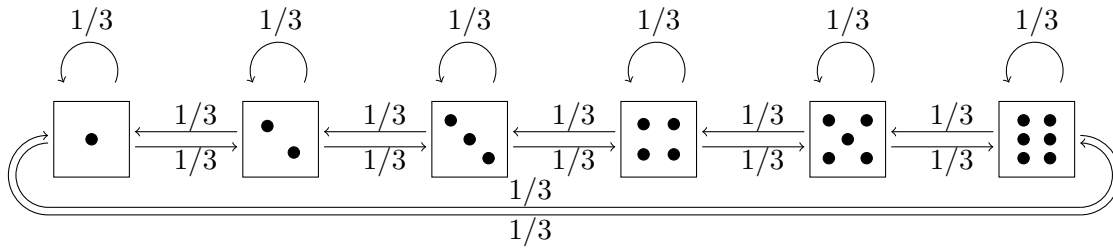


Abbildung 3: Eine irreduzible und aperiodische Markov Kette, welche sich als Modell für die Berechnung der Wahrscheinlichkeitsverteilung eines Würfels mittels MCMC eignet. In tatsächlichen Anwendungen wird es oft leider aufgrund der hohen Anzahl an Zuständen und deren Verbindungslinien nicht praktikabel sein, die Markov Kette graphisch darzustellen.

Häufigkeiten des Auftretens der möglichen Zustände dokumentiert. Nach N Durchläufen wird daraus die relative Häufigkeit berechnet. Der Anfangszustand wird im Code nicht mit einbezogen, was jedoch rein willkürlich ist und keine Rolle spielt.

```
import random
import math
```

```
def initialise(mu0):
    u = random.uniform(0, 1)
    if ( u <= mu0[0] ):
        return 1
    elif ( u <= mu0[0] + mu0[1] ):
        return 2
    elif ( u <= mu0[0] + mu0[1] + mu0[2] ):
        return 3
    elif ( u <= mu0[0] + mu0[1] + mu0[2] + mu0[3] ):
        return 4
    elif ( u <= mu0[0] + mu0[1] + mu0[2] + mu0[3] + mu0[4] ):
        return 5
    else:
        return 6
```

```
def update(x):
    u = random.uniform(0, 1)
    delta = math.floor( 3 * u ) - 1
    return (( x + delta ) % 6 ) + 1
```

```
if __name__ == "__main__":
    mu0 = (1/6, 1/6, 1/6, 2/6, 1/6, 0)
    x = initialise(mu0)
    counts = [0, 0, 0, 0, 0, 0]
    N = 100000
    for i in range(0, N):
        x = update(x)
        counts[x-1] += 1
    print("Probabilities for N=", N, ":")
    for i in range(0, 6):
        print( (i+1), ":", counts[i] / N )
```

Output für $N = 100$: (0.21, 0.19, 0.14, 0.18, 0.11, 0.17)
Output für $N = 1000$: (0.16, 0.169, 0.153, 0.161, 0.18, 0.177)
Output für $N = 10000$: (0.1668, 0.1717, 0.1635, 0.17, 0.1664, 0.1616)
Output für $N = 100000$: (0.16714, 0.16765, 0.16671, 0.165282, 0.16725, 0.16543)

3 Hill climb Algorithmus

Gesucht seien die Maxima einer Funktion $f : S \rightarrow \mathbb{R}$, mit gegebenem Zustandsraum S . Vorausgesetzt wird eine Metrik auf S , sodass es Sinn ergibt, von einer „Nachbarschaft“ eines Elementes in S zu sprechen. Je nach Komplexität der Funktion fallen eine analytische oder numerisch exakte Lösung weg und die MCMC Methode kann zur Anwendung gebracht werden. Dazu wird eine Markov Kette konstruiert, deren stationäre Wahrscheinlichkeitsverteilung gerade die Maxima der Funktion widerspiegelt. Ausgehend vom Zustand $s_n \in S$ für $n \in \mathbb{N}$ wird der nächste Zustand $s_{n+1} \in S$ wie folgt bestimmt. Ein Kandidat s für s_{n+1} wird zufällig aus einer Nachbarschaft von s_n gezogen. Ist nun $f(s) > f(s_n)$, so setze $s_{n+1} = s$. Wenn dahingegen $f(s) \leq f(s_n)$, wird $s_{n+1} = s$ nur mit einer bestimmten Wahrscheinlichkeit gesetzt. Ansonsten bleibt der Zustand unverändert, d.h. $s_{n+1} = s_n$. Diese sogenannte „Akzeptanz“-Wahrscheinlichkeit könnte etwa die Wahrscheinlichkeit einer gezinkten Münze sein, dass eine bestimmte Seite oben liegen bleibt. Durch diese zweite Regel, welche auch Schritte entlang der Funktion f zu niedrigeren Werten erlaubt, wird das Erreichen globaler Maxima ermöglicht. Ohne diese Regel könnte es leicht passieren, dass ein lokales Maximum erreicht wird und kein Weg mehr zu einem globalen Maximum möglich ist, vgl. Abbildung 4.

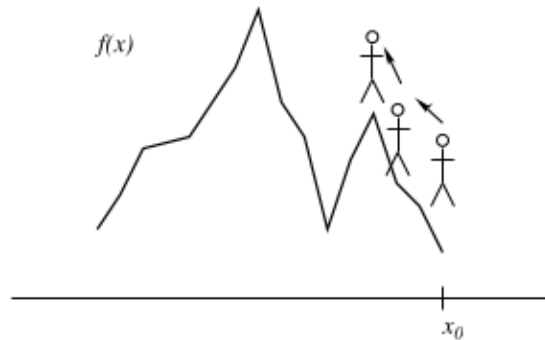


Abbildung 4: Ein Hill Climb Algorithmus kann an einem lokalen Maximum der Funktion f hängen bleiben, wenn keine Schritte nach unten erlaubt sind [2].

Es folgt der Beweis, dass ein solcher Algorithmus tatsächlich die globalen Maxima der Funktion f liefert [2].

Der Einfachheit halber wird eine symmetrische Transitionsmatrix der konstruierten Markov Kette auf S angenommen. Für ein $\lambda \geq 1$ wird die Verteilung

$$\pi_\lambda(s) = \frac{\lambda^{f(s)}}{Z(\lambda)}$$

definiert, wobei $Z(\lambda) := \sum_{s \in S} \lambda^{f(s)}$ die Normalisierungskonstante ist, welche π_λ zu einer Wahrscheinlichkeitsverteilung macht. Gerade diese Konstante $Z(\lambda)$ ist es, welche eine analytische oder numerisch exakte Lösung durch einen viel zu großen Berechnungsaufwand unerreichbar macht. Die Ausführung des Hill climb Algorithmus dahingegen erfordert nicht die Berechnung von $Z(\lambda)$. Aufgrund der Definition von $\pi_\lambda(s)$ werden solche Zustände s mit großen Werten $f(s)$ einer hohen Wahrscheinlichkeit ausgestattet. Die Akzeptanz-Wahrscheinlichkeit im Fall

$f(s) < f(s_n)$ bei einem Übergang $s_n \rightarrow s$ sei $\lambda^{-[f(s_n)-f(s)]}$. Für $\lambda \rightarrow \infty$ geht diese Wahrscheinlichkeit gegen Null. Die Menge der globalen Maxima sei definiert durch

$$S^* := \{s \in S : f(s) = f^* := \max_{t \in S} f(t)\}.$$

Die folgende Grenzwertbetrachtung zeigt, dass π_λ für $\lambda \rightarrow \infty$ gegen die Gleichverteilung der globalen Maxima von f konvergiert:

$$\begin{aligned} \lim_{\lambda \rightarrow \infty} \pi_\lambda(s) &= \lim_{\lambda \rightarrow \infty} \frac{\lambda^{f(s)}}{\sum_{t \in S} \lambda^{f(t)}} = \lim_{\lambda \rightarrow \infty} \frac{\lambda^{f(s)}}{|S^*| \cdot \lambda^{f^*} + \sum_{t \in S \setminus S^*} \lambda^{f(t)}} \\ &= \lim_{\lambda \rightarrow \infty} \frac{\lambda^{f(s)}/\lambda^{f^*}}{|S^*| + \sum_{t \in S \setminus S^*} \lambda^{f(t)}/\lambda^{f^*}} = \frac{1_{s \in S^*}}{|S^*|}. \end{aligned}$$

4 Beispiel: Solid disks in a box

Die Arbeit [4] von Metropolis et. al. aus dem Jahre 1953 war nicht nur namensgebend für eine mögliche Ausführung einer MCMC Methode, dem sogenannten Metropolis-Algorithmus. Sie stellt auch ein gutes Beispiel für die möglichen Anwendungsfälle der Methodik dar. Behandelt wird eine Fragestellung der statistischen Physik, in welcher eine bestimmte Zustandsgröße eines realen Gases in einer Kammer bestimmt werden soll. In gewisser Weise findet hier wieder die Methodik des Hill climb Algorithmus Anwendung, da die Energie des Zustandes in der Kammer minimiert werden soll. Die Energie berechnet sich hierbei als

$$E(s) = \frac{1}{2} \sum_i \sum_{j \neq i} V(d_{i,j}),$$

wobei V ein Potential zwischen zwei Teilchen in der Kammer darstellt, welches nur vom Abstand der beiden abhängt. Ein Kandidat für einen neuen Zustand wird erzeugt, indem ein einzelnes Teilchen ein wenig verschoben wird. Der Kandidat wird direkt akzeptiert, wenn sich die Gesamtenergie des Zustandes verringert, d.h. $\Delta E = E(s) - E(s_n)$. In den übrigen Fällen wird der Kandidat mit der Akzeptanz-Wahrscheinlichkeit $\exp(-\Delta E/kT)$ übernommen, wobei T den Einfluss der Temperatur des Gases beschreibt und k die Boltzmann-Verteilung ist.

Im Anhang findet sich noch eine handschriftliche Skizze dieses Beispiels, welche es aus Zeitgründen nicht mehr in diese Version der Ausarbeitung geschafft hat.

Für Interessierte findet sich unter <https://github.com/TheRealMephisto/SolidDisksInABox> das Git-Repository des vom Autor verwendeten Codes, geschrieben in Python.

Literatur

- [1] Steve Brooks u. a. »Handbook of Markov Chain Monte Carlo«. In: (Jan. 2011). DOI: 10.1201/b10905.
- [2] Markov Chains und Mixing Times. *Levin, David A. and Peres, Yuval and Wilmer, Elisabeth L.* 2. Aufl.
- [3] Olle Häggström. *Finite Markov Chains and Algorithmic Applications*. 2001.
- [4] Nicholas Metropolis u. a. »Equation of State Calculations by Fast Computing Machines«. In: (Juni 1953).

Example: Hard disks in a box

Metropolis, Rosenbluth, 1953

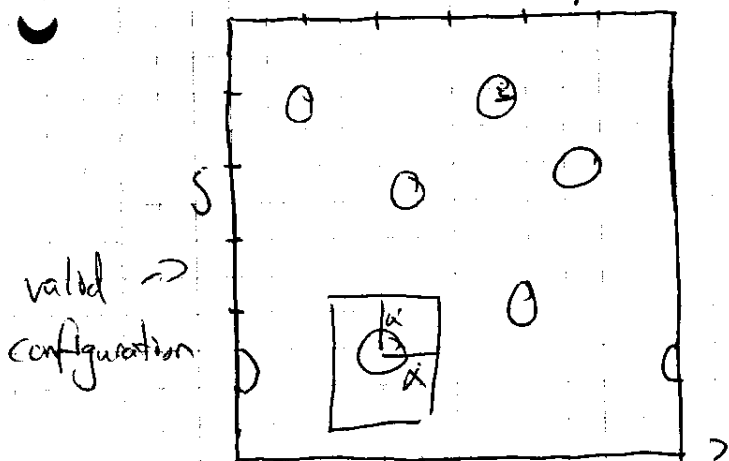
Most important aspect for my lecture:

Work out by example

Why is this not possible to solve analytically / exactly numerically?

→ MCMC ftw!

Disks of radius $\epsilon > 0$, in total N disks/particles
Box with discrete positions



valid configuration

v.c.: No overlapping

$X \in \mathbb{R}^{2N}$

$X = (r_1, s_1, r_2, s_2, \dots, r_N, s_N)$

Boltzmann Verteilung

$$\tilde{\pi}(X) = \frac{1}{Z} e^{-E(X)/kT} \cdot I(X \text{ v.c.})$$

$E(X)$ Energy of configuration

(choose particle i randomly (uniformly dist.) and choose new pos. in the box of length a randomly (uniform))

Trying to estimate expected value of some fct. f
 $E(f(X)), X \sim \pi$
Calc. of Pressure given volume and temperature

$$Q(x, x') = \frac{1}{N} I(\text{moved 1 particle})$$

$$= \frac{1}{N} I((r_i, s_i) = (r'_i, s'_i) \forall i \text{ except for } i=k, \text{ where } |r_k - r'_k| \leq a \text{ and } |s_k - s'_k| \leq a)$$

with $c = \#$ pts in $2a \times 2a$ box

1) Sample x from Q

2) sample u from $[0, 1]$

3) if $u < \frac{\tilde{\pi}(x)}{\tilde{\pi}(x_i)}$ then $x_{\text{new}} = x$, else $x_{\text{new}} = x_i$

$$Z = \sum_{X \text{ v.c.}} e^{-E(X)/kT}$$

$E(X), f(X)$ else falls complicated Fct.

Explore probability configs and approximate distribution $\tilde{\pi}(X)$