Modellierung und Simulation

Monte-Carlo-Simulation

Universität Hamburg

Johannes Schlundt

7. Januar 2013

Inhalt

- Motivation
- Geschichtliche Entwicklung
- Monte-Carlo-Simulation
- Zufallszahlengeneratoren(Exkurs)
- Zusammenfassung

Motivation

- Monte-Carlo-Simulation(MCS) ist eine Methode um:
  - Ein Analytisch unlösbares Probleme rein mathematischer Herkunft zu lösen

    z.B Berechnung des Integrals einer Funktion

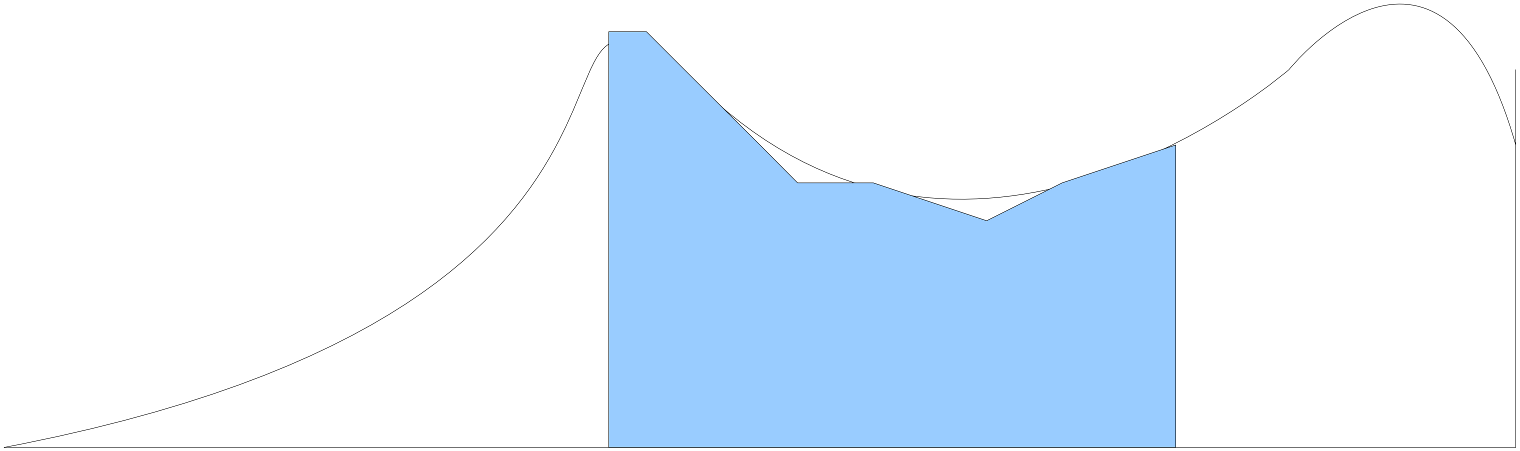


Bild: Wurde von mir erstellt.[1]

Motivation

- Monte-Carlo-Simulation(MCS) ist eine Methode um:
  - Verteilungseigenschaften von Zufallsvariablen unbekannten Verteilungstyps zu ermitteln.
  - Die Nachbildung von komplexen Prozessen, die nicht geradlinig analysiert werden können.
    z.B Wetter und Klima der Erde
  - Unsicherheiten und statistisches Verhalten simulieren
    z.b in welchem Fach ein Tischtennisball landet, wenn dieser Ball durch ein Nagelbrett fällt

Motivation

- Der Versuch der mit MCS berechnet werden soll, braucht eine große Freiheit von Eingabevariablen!
  Danach wird der Versuch ca. 1000 mal oder öfter Wiederholt durchgeführt.

- MCS hat die Vorteile:
  - Das nur ein Bruchteil der Berechnung eines gleichen Deterministischen Algorithmus nötig wäre.
  - Einfach zu Implementieren.

- Nachteil
  - Nur eine Annäherung zum exaktem Resultat.
  - Kann auch komplett daneben liegen.

Geschichtliche Entwicklung

- 1934 von Enrico Fermi zum ersten mal Praktisch für eine Simulation einer Neutronenstreuung genutzt.

- Früher in der Stochastik schon bekannt, doch hatte noch keinen Namen und Zweck. Da MCS sich das von der Stochastik bekannte Verhalten, das „Gesetz der großen Zahlen“ zu nutzen macht.

Geschichtliche Entwicklung

- 1946 von Stan Ulam bei einem Kartenspiel wieder entdeckt.

  Stan Ulam wollte seine Gewinnchance ausrechnen.

  - Kartenspiel Canfield Solitär:
    - 52 Karten
    - Kombinatorisch aufwendig zu lösen
    - MCS: 100 Zufällige Spiele
      - Die gewonnenen Spiele durch die Gesamte Anzahl von Spielen teilen.

Geschichtliche Entwicklung

- Stan Ulam arbeitete zu dieser Zeit am Geheimen „Manhattan Project“.
- John Von Neumann arbeitete zu dieser Zeit auch im selben Projekt. Stan erzählte ihm von seiner Möglichkeit, wie man ein Kombinatorisches Problem einfacher lösen kann.
- John Von Neumann erkannte das Potenzial und sie entwickelten zusammen noch mehrere andere Algorithmen
- Da MCS im Projekt sehr hilfreich war, aber noch kein Namen hatte, kam Von Neumann auf den Namen „Monte-Carlo-Simulation“. Das auf Ulam's Onkel zurückzuführen ist, der im gleichnamigen „Monte Carlo Casino“ Geld zum spielen leihen würde.

Monte-Carlo-Simulation

- Definition
- Allgemeine Eingrenzung
- Beispiel: Approximation der Zahl Pi
- Ein Weiteres Beispiel: Nagelbrett
- Kinetisch Monte-Carlo Simulation

Definition

- Wann genau MCS vorliegt ist meistens nicht klar.
- Kalos & Whitlock beschreiben, dass ein MCS Algorithmus als eine „natürliche Simulation“ oder als eine Lösung der Gleichung betrachtet werden kann.
- Definition (nach Sawilowsky)
  - Simulation
  - Monte-Carlo-Methode
  - Monte-Carlo-Simulation

Definition

- **Simulation** ist eine fiktive Darstellung der Realität.
  - Wir generieren <u>eine</u> zufällige Zahl aus einem Zufallszahlengenerator. Dies Zahl ist einheitlich in einem Bereich von (0; 1] verteilt und kann als eine geworfene Münze angesehen werden: Wenn sie kleiner oder gleich 0.50 ist, dann ist es Kopf, aber wenn der Wert 0.50 überschreitet ist es Zahl.
  - Das genannte Beispiel ist eine Simulation, aber keine Monte-Carlo-Simulation.

Definition

- Eine **Monte-Carlo-Methode** ist eine Technik die benutzt wird, um ein mathematisches oder statistisches Problem zu lösen.
  - Eine irreguläre Figur auf einem einheitlichen Quadrat wird mit Dartpfeilen beworfen und dann wird das Verhältnis von Figur getroffenen zu insgesamt geworfenen Pfeilen berechnet.
  - Das ist ein Beispiel für eine Monte-Carlo-Methode auf einen eingegrenzten Bereich, aber keine Simulation.

Definition

- Eine **Monte-Carlo-Simulation** nutzt das mehrmalige wiederholen des Versuchs aus, um ungewollte Phänomene oder Verhalten zu entfernen.
  - Wir generieren eine <u>große Anzahl</u> von Zufallszahlen. Die einheitlich aus dem Bereich von (0; 1] stammen. Ist der Wert kleiner oder gleich 0.50 ist es Kopf und wenn größer 0.50 Zahl.
  - Das ist ein Beispiel für eine Monte-Carlo-Simulation die das Verhalten einer mehrmalige geworfenen Münze darstellt.

Funktionsweise

- Im Grunde Funktionieren alle MCS gleich:
  1. Wir grenzen die Möglichen Eingaben ein.
  2. Wir generieren zufällige Eingaben aus denn möglichen Eingabebereich.
  3. Danach führen wir eine deterministische Berechnung mit denn vorhanden Eingaben aus.
  4. Zuletzt werten wir das Ergebnis aus.

Approximation der Zahl Pi

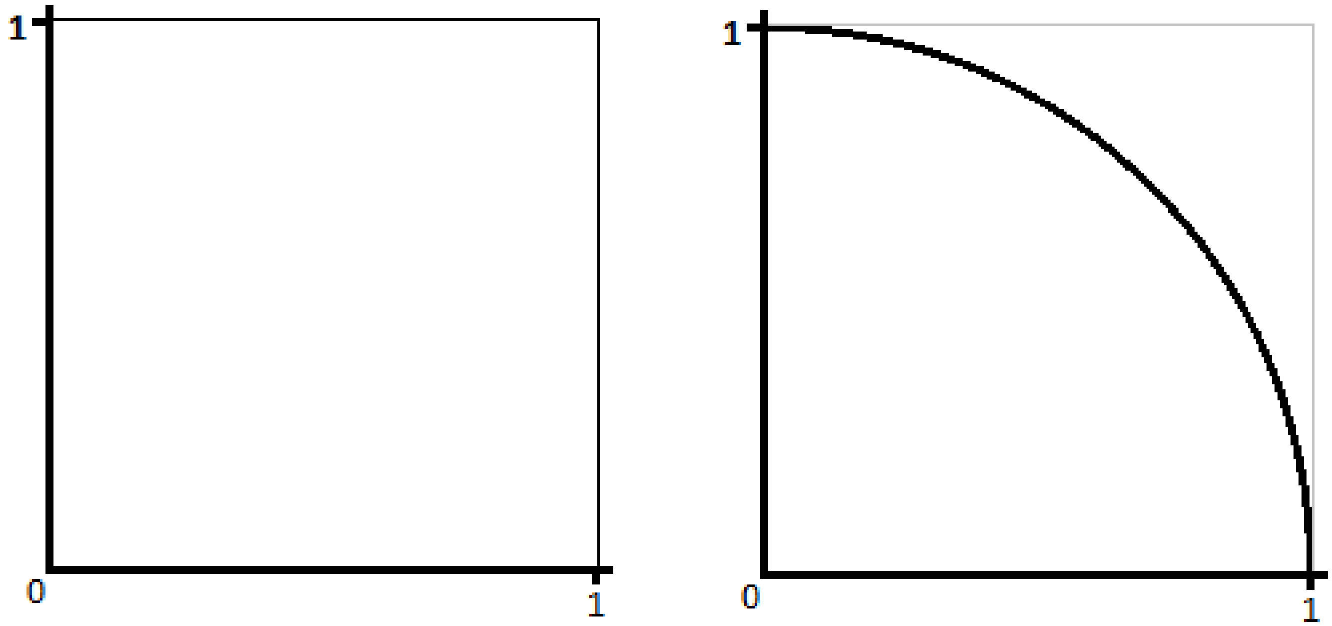


Bild: Wurde von mir erstellt.[1]

Approximation der Zahl Pi

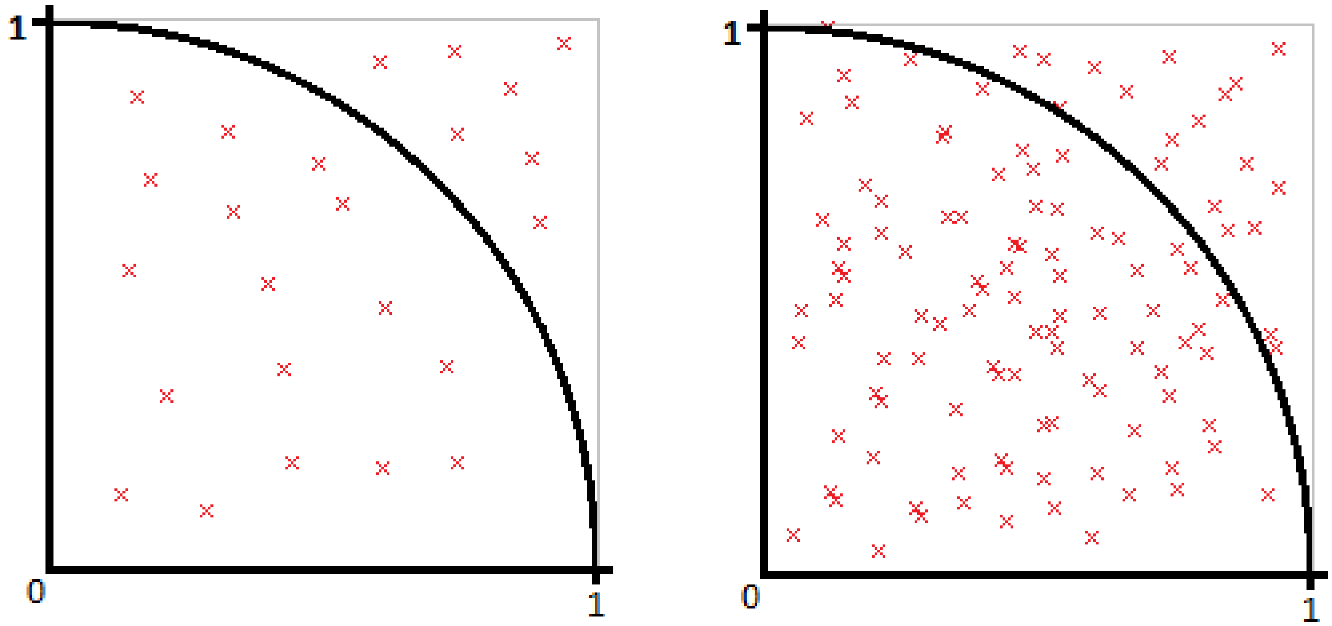
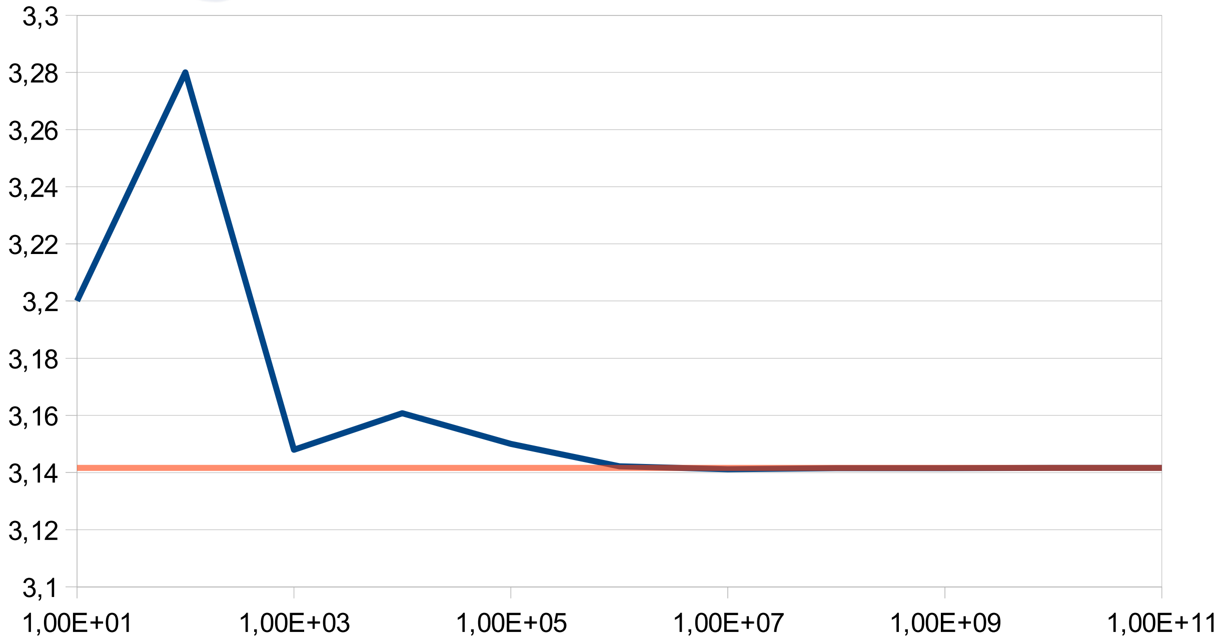


Bild: Wurde von mir erstellt.[1]

Approximation der Zahl Pi

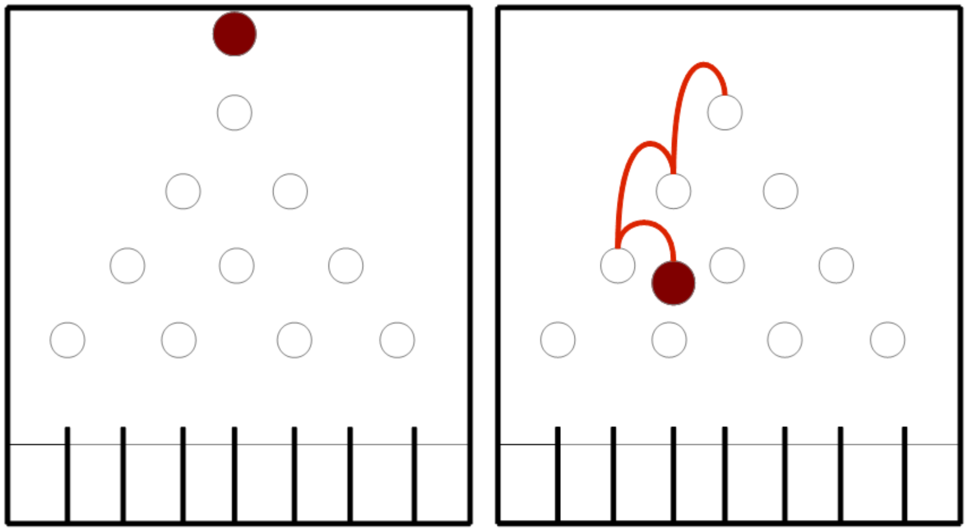


- Erst nach ca. 1Mio Wiederholungen, nähert sich der Wert an PI.
- Meine Annäherung war nach 100 Milliarden Wiederholungen, auf 4 stellen nach denn Komma genau.

Meine: **3,1415**8731...
PI: 3,1415926536....

Bild: Wurde von mir erstellt.[1]

Nagelbrett



1. Wir grenzen die Möglichen Eingaben ein.
   Je Stufe fällt der Ball links oder rechts.

2. Wir generieren zufällige Eingaben aus denn möglichen Eingabebereich.
   Der Ball fällt durch Zufall nach links oder rechts.

3. Danach führen wir eine deterministische Berechnung mit denn vorhanden eingaben aus.
   Wir Zählen wie oft er in welches Fach fällt und dividieren durch die Gesamtanzahl der Durchläufe.

4. Zuletzt werten wir das Ergebnis aus.
   Aus dem Ergebnis entwickelt man eine Verteilung. In welches Fach mit welcher Wahrscheinlichkeit der Ball fällt.

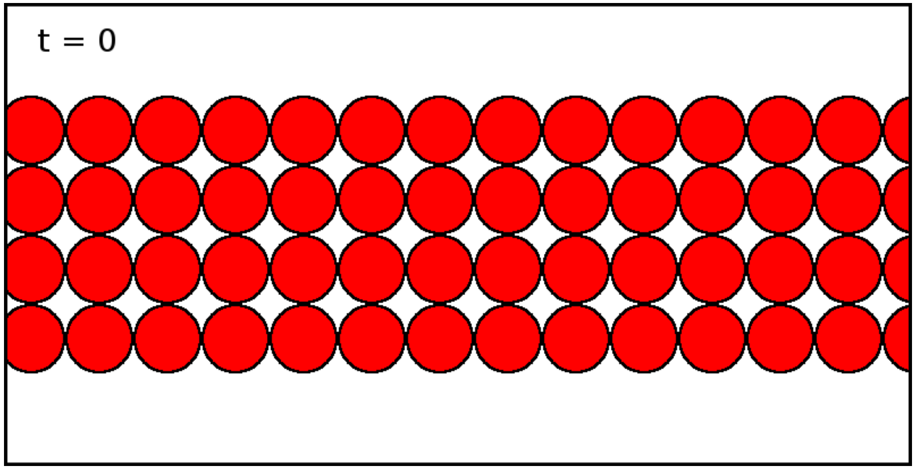
Bild: Wurde von mir erstellt.[1]

kinetisch Monte-Carlo Simulation

- Kinetischer Monte Carlo (kMC) ist eine Monte Carlo Simulation die häufig in der Physik genutzt wird.
- Mit kinetische Monte Carlo können wir Natürliche zeitabhängige Entwicklungen simulieren.

- Zum leichteren Verständnis, erkläre ich den kinetischen Monte Carlo mit dem Beispiel des Plateau-Rayleight Instability. Dabei transformiert sich ein Zylindrisches-Gebilde in einzelne Tropfen.

kinetisch Monte-Carlo Simulation



1. Wir setzten die Zeit auf $t=0$
2. Wir erstellen eine Liste aller beweglichen Objekte im System $r_{i.}$
3. Wir berechnen eine kumulative Funktion

$$R_i=\sum_{j=1}^{i} r_j : i=1,\ldots,N$$

   N stellt die Gesamtzahl der Übergänge dar.
4. Wir wählen eine zufällige Zahl $u\in(0;1]$ aus.

Bild: Wurde von mir erstellt.[1]

kinetisch Monte-Carlo Simulation

5. Wir wählen das Event i, dass wir mithilfe der Funktion finden
$$R_{i-1} < uR_N \leqslant R_i$$
6. Wir führen Event i aus.
7. Wir wählen eine neue zufällige Zahl $u' \in (0;1]$ aus.
8. Wir aktualisieren die Zeit mit $t = t + \Delta t \quad \Delta t = R_N^{-1} \ln(1/u')$
9. Wir Berechnen die Liste aller Objekte in r_i neu, die sich während des Übergangs geändert haben und noch beweglich sind.
10. Zurück zu Schritt 2

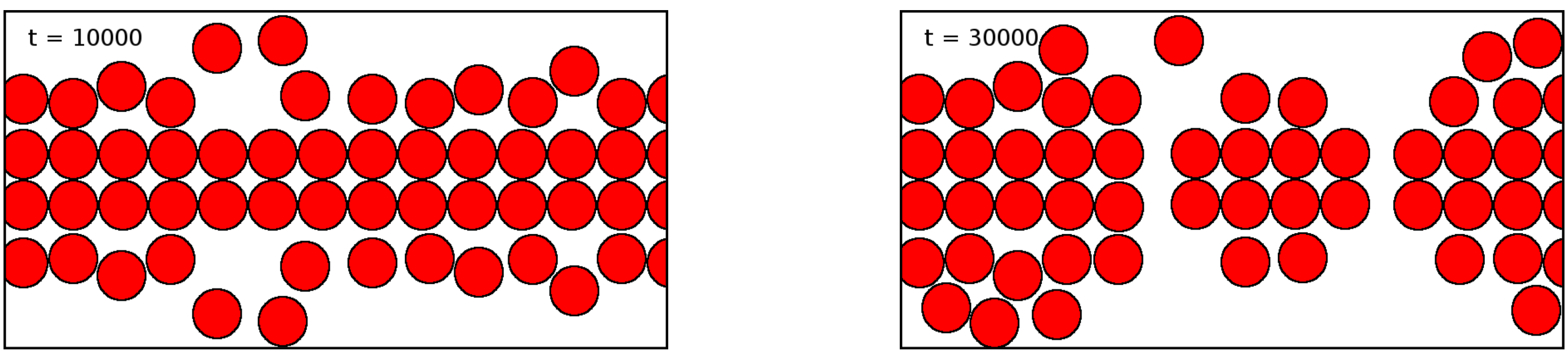


Bild: Wurde von mir erstellt.[1]

Monte-Carlo-Simulation

Geschichtliche Entwicklung

- **Zufallszahlengeneratoren(Exkurs)**
- Zusammenfassung
- Literatur

Zufallszahlengeneratoren

- Da MCS mit Zufallszahlen arbeitet, um zufällige Versuche zu generieren, brauchen wir einen Zufallszahlengenerator.
- John Von Neumann hat in denn Anfangszeiten selber ein Entwickelt, da es noch keine Zufallszahlengeneratoren gab. Die „Mittquadratmethode“, dabei wird in jeder Iteration eine Zahl quadriert, die mittlere-Werte dann für die neu Berechnung übergeben und Wiederholt.
  0**08**0=>0**16**0=>2**56**0=>31360=>...
  - Die Nachteile sind, dass der Generator nach einigen Iteration eine Nullfolge bildet oder die generierten Zahlen nicht gleichmäßig verteilt liegen.

Zufallszahlengeneratoren

- Güte von deterministische Zufallszahlengeneratoren laut BSI
  - K1 - Eine Folge von Zufallszahlen mit einer geringen Wahrscheinlichkeit mit identischen aufeinander folgenden Zahlen.
  - K2 - Eine Folge von Zahlen, die nicht von einem „wahren Zufallszahl“ in einem statistischem Test unterschiedenen werden kann.

Zufallszahlengeneratoren

- K3 - Für jeden Angreifer sollte es unmöglich sein von jeder gegebener Sub-Sequenz, frühere oder zukünftige Werte in der Folge, noch einen inneren Zustand des Generators zu berechnen oder sonst zu erhalten.
- K4 - Für einen Angreifer sollte es unmöglich sein aus einem inneren Zustand des Generators, etwaige frühere Nummern in der Sequenz oder alle früheren inneren Generator Zustände zu berechnen oder zu erhalten.

Zufallszahlengeneratoren

- MCS reichen K1 und K2 aus.
- Wenn der Zufallsgenerator nicht gleichverteilt arbeitet, ist dieser für manche Simulation nicht zu gebrauchen.

Zufallszahlengeneratoren

- MCS reichen K1 und K2 aus.
- Wenn der Zufallsgenerator nicht gleichverteilt arbeitet, ist dieser für manche Simulation nicht zu gebrauchen.

Monte-Carlo-Simulation

Geschichtliche Entwicklung

Zufallszahlengeneratoren(Exkurs)

- **Zusammenfassung**
- Literatur

Zusammenfassung

- Monte-Carlo-Simulation ist geeignet Prozesse zu Simulieren, die mit einem deterministischen Algorithmus schwer oder gar nicht lösbar sind.
- Bei einem gutem Zufallszahlengenerator kann man gute natürliche Ereignisse simulieren.
- Wenn man durch die große Freiheit der Variablen kein Deterministischen Algorithmus entwickeln kann, dann kann man mit MCS dennoch ein Versuch simulieren.

Literatur

[1]Alle Grafiken wurden von mir selber erstellt und verdeutlichen nur Skizzenhaft. Haben deshalb auch keine große Aussagekraft.

http://en.wikipedia.org/wiki/Monte_Carlo_method

http://de.wikipedia.org/wiki/Monte-Carlo-Simulation

http://en.wikipedia.org/wiki/Kinetic_Monte_Carlo

[HLA]H. L. Anderson: "Metropolis, Monte Carlo and the MANIAC. Los Alamos Science, Nr. 14, 1986, S. 96–108, 1986. http://library.lanl.gov/cgi-bin/getfile?00326886.pdf

Metropolis, N. (1987). "The beginning of the Monte Carlo method". Los Alamos Science (1987 Special Issue dedicated to Stanisław Ulam): 125–130.

http://en.wikipedia.org/wiki/Pseudorandom_number_generator

Ende

Vielen Dank für Ihre Aufmerksamkeit.

Noch Fragen?