

Integration und Stochastik

HANNES STOPPEL, BOCHUM

Zusammenfassung: In diesem Artikel geht es um die leichte Berechnung von Flächen zwischen einem Funktionsgraphen und der x -Achse. Dies ist mithilfe stochastischer Vorgänge unter Verwendung eines Computers auf übersichtliche Art und Weise möglich. Dazu werden randomisierte Algorithmen eingesetzt und direkt am Computer unter MAPLE umgesetzt, was die Realisierung für Schülerinnen und Schüler übersichtlich leicht gestaltet. Die in diesem Artikel geschilderte Vorgehensweise wurde in der Schule bereits umgesetzt.

1 Einleitung

Die Flächenberechnung zwischen dem Graphen einer Funktion und der x -Achse wird bislang ab der Jahrgangsstufe 12 durchgenommen und nur mit Verfahren der Analysis durchgeführt. Früher war es üblich, die Differential- und Integralrechnung auf der Basis von Folgen und Reihen zu behandeln. Dies hat sich vor einiger Zeit geändert – die meisten Sachverhalte werden anschaulich begründet. Nun soll das Abitur nach der Jahrgangsstufe 12 erreicht werden. Diese Verkürzung der Schulzeit verlangt danach, Stoff vorzuziehen.

In diesem Artikel wird eine Möglichkeit vorgestellt, die Stochastik in der Integralrechnung anzuwenden. Mithilfe dieser Anwendung können relativ einfache Arten näherungsweise Flächeninhaltsberechnung mithilfe eines Computers in der Jahrgangsstufe 10 behandelt werden. Für diese Vorgehensweise sprechen drei weitere Punkte:

- Der Computereinsatz motiviert die SchülerInnen. Dieser Aspekt kann gar nicht hoch genug angesetzt werden.
- Mehrere Bereiche der Schulmathematik werden verknüpft. Diese Verknüpfung ergibt sich auf natürliche Weise aus der Sache heraus.
- Die Stochastik wird im Schulunterricht oft sehr knapp gefasst oder ganz weggelassen. Hier erfolgt die Verwendung in einem zentralen Bereich der Schulmathematik und zeigt auch die Gleichstellung der Stochastik mit anderen Teilen der Mathematik.

Die hier dargestellte Unterrichtsreihe wurde bereits mit Erfolg in derselben Reihenfolge wie in diesem Artikel beschrieben durchgeführt. Ein umfangreicher Teil wurde hierbei auch in mündlicher Zusammenarbeit bzw. Gruppenarbeit von SchülerInnen umgesetzt.

Die SchülerInnen arbeiteten hierbei deutlich schneller als erwartet und verlangten nach neuen Aufgaben und Problemstellungen; ihre Motivation war sehr hoch. Als krönender Abschluss wurde das *Nadelproblem* gelöst.

Die Programme wurden mit MAPLE geschrieben. Dies hat den Vorteil, dass sich die Algorithmen relativ einfach in einem Programm formulieren lassen. Es kann jedoch auch jede andere Programmiersprache benutzt werden, z.B. Java, JavaScript, Delphi oder C++.

Wenn MAPLE unbekannt ist, lassen sich mit [Stp] die Grundlagen erarbeiten, und in [W] befindet sich eine gute Übersicht über die Möglichkeiten der MAPLE-eigenen Programmiersprache.

2 (Pseudo-) Zufallszahlen

Bei der Berechnung von Integralen mithilfe von Computern nach dem in diesem Artikel vorgestellten Verfahren werden Zufallszahlen benötigt. Aus diesem Grund wird ein Verfahren zur Erzeugung von Zufallszahlen hier kurz vorgestellt, denn auch wenn Zufallszahlen später unter MAPLE durch Verwendung eines Befehls im Hintergrund erzeugt werden, liegt ein solches Verfahren vor.

Die Berechnung solcher Zufallszahlen mithilfe eines Computers beruht nicht wirklich auf Zufall; daher werden sie als *Pseudo-Zufallszahlen* bezeichnet. Zur Berechnung von Pseudo-Zufallszahlen¹ existieren mehrere Algorithmen, von denen hier nur das *Lineare-Kongruenz-Verfahren* vorgestellt werden soll, das in zahlreichen Programmen und Programmiersprachen benutzt wird.² Für umfassendere Informationen zu anderen Verfahren zur Erzeugung von Zufallszahlen siehe [Moe].

¹Da wir uns im Folgenden ausschließlich mit Pseudo-Zufallszahlen verwenden, werden sie als Zufallszahlen bezeichnet.

²Zufallszahlen werden beispielsweise unter MAPLE und JAVA ebenfalls nach dem hier vorgestellten Verfahren berechnet.

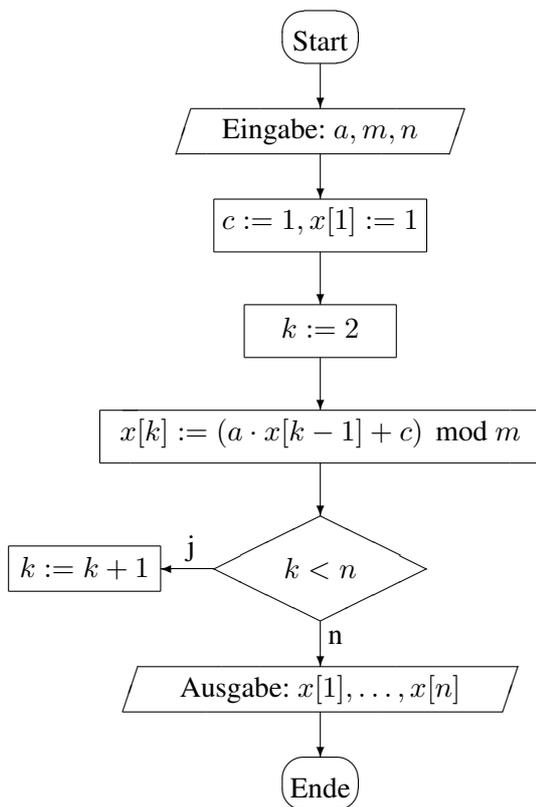


Abb. 1: Flussdiagramm zum Lineare-Kongruenz-Verfahren

Zufallszahlen werden durch Festlegung eines Startwertes und der Koeffizienten einer Funktion und dann durch Iteration der Funktion bestimmt. Dies soll hier am Beispiel des *Lineare-Kongruenz-Verfahrens* verdeutlicht werden.

Abbildung 1 zeigt das Flussdiagramm zum Lineare-Kongruenz-Verfahren.

Wie zu erkennen ist, werden Startwerte für den linearen Koeffizienten a und den Summanden c gesetzt und dann die entsprechenden linearen Operatoren ausgeführt. Dies wird mehrfach hintereinander gemacht, wobei n die Anzahl der Schritte festlegt.

Würde man ohne den Zusatz $\text{mod } m$ rechnen, müsste man jedoch feststellen, dass die Folge an Zahlen schnell anwächst und gegen das Unendliche verläuft, wie das folgende Beispiel für $a = 2, c = 1$ mit $x_1 = 1$ zeigt.

$$x_1 = 1 \Rightarrow x_2 = 3 \Rightarrow x_3 = 7 \Rightarrow x_4 = 15$$

$$\Rightarrow x_5 = 31 \Rightarrow x_6 = 63 \Rightarrow x_7 = 127 \dots$$

Damit dieser Effekt nicht eintritt, wird hier modulo einer Zahl m gerechnet. Dann liegen alle Ergebnisse im Bereich $\{0, 1, \dots, m - 1\}$.

Zusammengefasst lautet der Algorithmus:

Algorithmus

Eingabe: $n \in \mathbb{N}$ (Anzahl der Schritte)
 $m \in \mathbb{N}$ (Modulwert)
 $a \in \mathbb{N}$ (Koeffizient der x_i)
 $c \in \mathbb{N}$ (Summand der linearen Funktion)

1. Setze $x_1 = 1$.
2. Für k von 2 bis n wird folgender Schritt wiederholt:

$$x_k := (a \cdot x_{k-1} + c) \text{ mod } m.$$

Ausgabe:

$$x_1, \dots, x_n.$$

Ein mögliches Programm unter MAPLE lautet:

Programm

Zu Beginn wird der Arbeitsspeicher geleert:

```
> restart;
```

Zunächst werden hier die Werte für die Variablen eingegeben.

```
> c := 1:
> a := 137:
> m := 256:
```

Der Anfangswert lautet

```
> x[1] := 1:
```

Es folgt der eigentliche Iterationsvorgang. Nun werden die Werte der $x[i]$ berechnet.

```
> for n from 2 to 100 do
> x[n] := (a*x[n-1] + c) mod m;
> od;
```

Dass es sich bei solchen Zufallszahlen nicht wirklich um etwas *Zufälliges* handelt, ist für manche SchülerInnen überraschend, für andere hingegen nicht. Dies kann zu einer Diskussion zwischen den SchülerInnen führen. Es gibt eine Möglichkeit, dies gut sichtbar zu machen. Man erzeugt einen Graphen, in dem als x -Koordinaten die x_n und als y -Koordinaten die x_{n-1} gewählt werden. In den Abbildungen 2 und 3 ist dies für die jeweils angegebenen Parameter dargestellt, wobei ein mögliches Programm zur Erzeugung der Graphen unter MAPLE folgendermaßen aussieht:

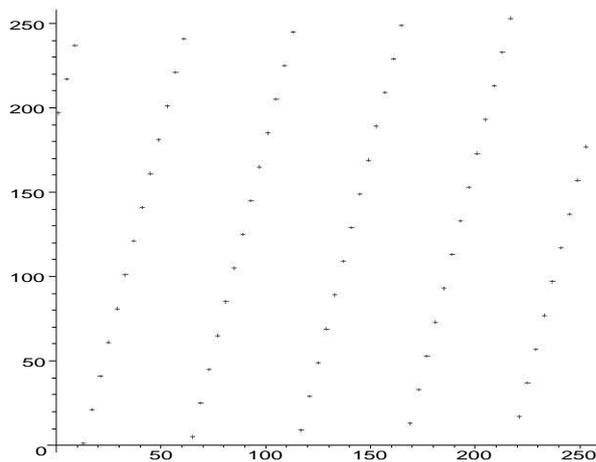


Abb. 2: $a = 13, c = 0$

Programm

```
> n:=600: m:=256: x[1]:=1:
> for a from 13 to 313 by 2 do
> for c from 0 to 1 do
> for k from 2 to n do
> x[k] := (a*x[k-1] + c) mod m:
> od:
> l := [ [x[k], x[k-1]]
$ k=2..n];
> print(plot(l, style=point,
symbol=circle));
> lprint('a'); lprint(a);
lprint('c'); lprint(c);
> od:
> od:
```

An diesem Graphen ist zu erkennen, wie sich die Pseudo-Zufallszahlen bei der Iteration verhalten. Da Iteration bedeutet, dass $x(n+1) = f(x(n))$ für eine bestimmte Funktion f gilt, sind die Punkte im Graphen als $(x(n), f(x(n)))$ aufzufassen. Die auftretende Geometrie deutet darauf hin, dass das Verhalten von Orbits vorgegeben und nicht zufällig ist.³ Diese Orbits wurden von den SchülerInnen intensiv diskutiert und führten zu einem guten Einblick in Zufallszahlen.

Da man die modulo-Rechnung bei dem Verfahren der linearen Kongruenz berücksichtigt, zeigt sich eine Periodizität des Orbits. Hierbei spielt auch die Rundung der Zahlen eines Orbits durch den Computer auf eine feste Nachkommastellenzahl eine Rolle. Die Bestimmung der Orbits deutet darauf hin, dass mit dem Lineare-Kongruenz-Verfahren keine wirklichen Zufallszahlen vorliegen. Nichtsdestotrotz handelt es

³Als *Orbit* bezeichnet man eine Folge von Funktionwerten, die bei der Iteration von Funktionen entstehen, z.B. in der nichtlinearen Dynamik oder Chaostheorie. Für genaue Hinweise vgl. [Stp2], 2.1.

⁴Eine exakte Definition und Untersuchung von Monte-Carlo-Verfahren finden sich in [Hä], Kapitel 7.

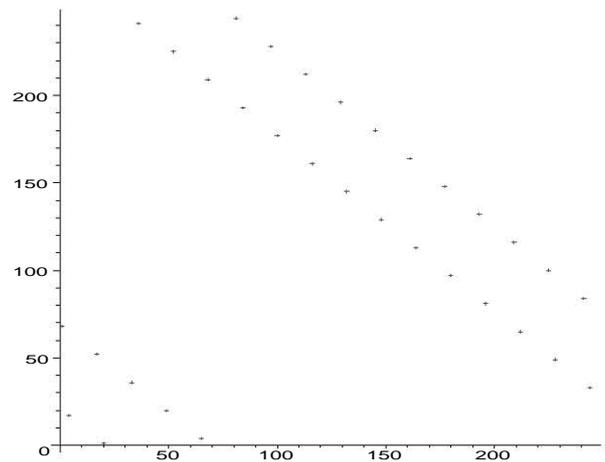


Abb. 3: $a = 15, c = 5$

sich hierbei um ein gutes Verfahren, um Pseudo-Zufallszahlen zu bestimmen, da sich für verschiedene Anfangswerte die Orbits deutlich unterscheiden. Die Startwerte werden im Allgemeinen in Abhängigkeit der Uhrzeit gesetzt und verändern sich daher im Laufe der Zeit, was zu einer Veränderung der Orbits mit der Zeit führt. Für genauere Hintergründe vgl. [Kü].

Zufallszahlen werden ab jetzt unter MAPLE erzeugt. Hierbei werden die Startwerte aus der momentanen Zeit berechnet. Zur Bestimmung von Zufallszahlen unter MAPLE vgl. auch die Fußnote 5 zum Programm aus Abschnitt 3.

3 Monte-Carlo-Integration und Flächeninhalt

Mithilfe eines Monte-Carlo-Verfahrens⁴ zur Integration lässt sich die Fläche zwischen der x -Achse und dem Graphen einer Funktion näherungsweise bestimmen. Hier sei zunächst das relativ simple Beispiel

$$f: [0, 2] \rightarrow \mathbb{R}, x \mapsto x^2,$$

betrachtet, siehe Abbildung 4. Die Maßzahl der markierten Fläche soll bestimmt werden.

Man lässt das Programm zunächst eine gewisse Anzahl von Zufallszahlen aus dem Intervall $[0, 2]$ bestimmen; etwa 5000 oder 10000 Zufallszahlen reichen aus. Von diesen Zufallszahlen bestimmt man die Funktionswerte und bildet deren Mittelwert. Der

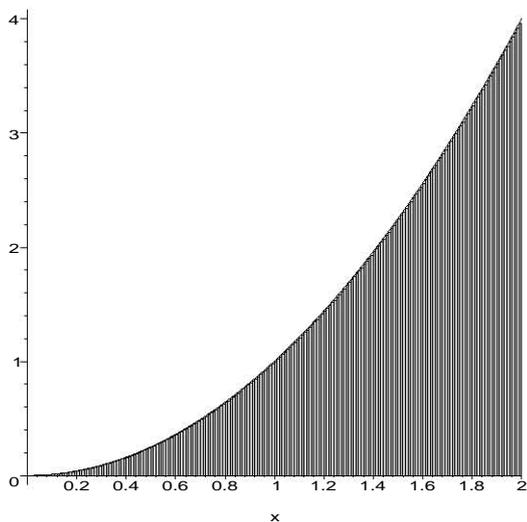


Abb. 4: Abbildung zu $f(x) = x^2$ mit markierter Fläche

Mittelwert ergibt multipliziert mit der Intervallbreite den ungefähren Wert des Flächeninhaltes. Anschaulich ist die zu bestimmende Fläche also in ein flächengleiches Rechteck umgewandelt worden.⁵ Dieser Algorithmus lautet:

Algorithmus 1

Eingabe: Funktionsterm f der Funktion, für die die Maßzahl der Fläche zwischen Funktionsgraph und x -Achse bestimmt werden soll.

1. Wähle k Zufallszahlen $x_1, \dots, x_k \in [0, 2]$.
2. Berechne $f(x_i)$ für $i = 1, \dots, k$.
3. Bestimme den Mittelwert der Funktionswerte $f(x_i)$ für $i = 1, \dots, k$.
4. Multipliziere den Mittelwert aus Schritt 3 mit der Breite 2 des Intervalls.

Ausgabe: Ergebnis der Berechnung aus den Schritten 1 bis 4.

Hinweis: Unter MAPLE werden ganzzahlige Zufallszahlen $z \in [0, 10^{12}]$ bestimmt. Im Allgemeinen wird jedoch bevorzugt mit Zufallszahlen $r \in [0, 1]$ gearbeitet, daher wird in den folgenden Programmen stets $r := \frac{z}{10^{12}}$ bestimmt.

Bemerkung: Eine weitere Möglichkeit für die Integration besteht in der folgenden Vorgehensweise:

Man bestimmt Paare (x_i, y_i) von Zufallszahlen für $i = 1, \dots, k$ und prüft dann, ob $y_i \leq f(x_i)$ gilt. Falls

⁵Ist das bestimmte Integral bereits bekannt, so wird auf diese Art der Mittelwert der Funktionswerte in einem Intervall bestimmt, vgl. [Gr], S. 195.

ja, dann liegt der Punkt (x_i, y_i) auf der Fläche unter dem Graphen von f . Mit einer analogen Berechnung zu Abschnitt 4.3 lässt sich dann die Fläche zwischen dem Funktionsgraphen und der x -Achse berechnen (falls keine Nullstellen der Funktion in dem betrachteten Intervall liegen).

Es gibt jedoch zwei Gründe für den Vorzug des obigen Verfahrens:

- Die Mittelwertbestimmung sollte anhand der Funktionswerte wiederholt werden.
- Der maximale Funktionswert der untersuchten Funktion f im betrachteten Intervall sollte bekannt sein, da sonst nicht gewährleistet ist, dass für den maximalen Funktionswert $f_{\max} \leq \text{rand}_{\max}$ gilt.

Prinzipiell könnte in den hier geschilderten Programmen die Intervallunterteilung auch durch eine feste Unterteilung in n gleiche Unterintervalle ersetzt werden. Die Zufallszahlen bieten jedoch die Möglichkeit eines Übergangs zur Integration mithilfe von beliebigen Nullfolgen. Es kann so in der Oberstufe wieder aufgegriffen werden.

Mit den SchülerInnen lässt sich leicht überlegen, dass dieser Algorithmus nicht optimal ist, da er einerseits auf ein bestimmtes Intervall bezogen ist und andererseits bei Funktionen mit negativen Funktionswerten eine Fläche unterhalb der x -Achse subtrahiert wird. Dies lässt sich jedoch leicht beheben, unter anderem indem der Betrag der Funktionswerte berechnet wird. Der Algorithmus wird dann folgendermaßen verändert:

Algorithmus 2

Eingabe:

- Funktionsterm f der Funktion, für die die Maßzahl der Fläche zwischen Funktionsgraph und x -Achse berechnet werden soll.
 - Zahl r , so dass die Fläche im Intervall $[0, r]$ berechnet werden soll.
1. Wähle k Zufallszahlen $x_1, \dots, x_k \in [0, r]$.
 2. Berechne $|f(x_i)|$ für $i = 1, \dots, k$.
 3. Berechne den Mittelwert der Funktionswerte $|f(x_i)|$ für $i = 1, \dots, k$.

4. Multipliziere den Mittelwert aus Schritt 3 mit der Breite r des Intervalls.

Ausgabe: Ergebnis der Berechnung aus den Schritten 1 bis 4.

Dieser Algorithmus kann in ein Programm unter MAPLE umgesetzt werden.

Programm

```
> restart:
```

Hier wird die Funktion eingegeben.

```
> f := x -> x-1:
```

Nun soll die Anzahl der zu bestimmenden Zufallszahlen eingegeben werden:

```
> k := 10000:
```

An dieser Stelle ist die obere Grenze des Intervalls einzugeben, in dem die Fläche berechnet werden soll.

```
> r := 2:
```

```
> for i from 1 to k do
> x[i] := evalf(rand()
/10^12*r):
```

```
> y[i] := abs(f(x[i])):
```

```
> od:
```

Nun wird der Mittelwert der Funktionswerte berechnet:

```
> m := 1/k*sum(y[n], n=1..k):
```

Es folgt die Berechnung der Fläche des Rechtecks, dessen Breite r entspricht und dessen Höhe gleich dem Mittelwert der Funktionswerte ist.

```
> F := r*m:
```

Die Ergebnisse sind bei 10^4 Wiederholungen relativ genau und regen die SchülerInnen zu Überlegungen bzgl. der Berechnung von Flächeninhalten der Flächen zwischen Graphen von Funktionen und der x -Achse an. Sie können Verbesserungsvorschläge selbstständig am Computer ausprobieren und damit ihre Ideen eigenständig testen. Anwendungen auf verschiedene Funktionen bestätigen die Wertigkeit des Verfahrens. Unterschiedlich viele Zufallszahlen beeinflussen die Genauigkeit des Ergebnisses. Hier tun sich diverse Fragestellungen auf, die die SchülerInnen entwickeln und bearbeiten können.

Eine interessante Anwendung der behandelten Verfahren findet sich im nächsten Abschnitt.

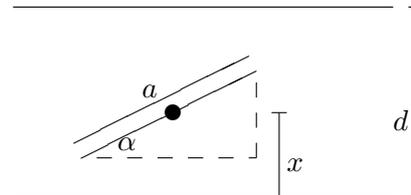


Abb. 5: Skizze zum Nadelproblem

4 Das Nadelproblem

Bei dem Buffonschen Nadelproblem handelt es sich um ein „klassisches“ Problem. Mit seiner Lösung lässt sich π näherungsweise bestimmen. Das Nadelproblem wird im Folgenden zunächst allgemein beschrieben; für die Schule würde man vermutlich von Beginn an Bedingungen an a und d stellen. Dabei taucht eine Flächenberechnung auf, die mithilfe des in Abschnitt 3 beschriebenen Verfahren näherungsweise gelöst werden kann.

4.1 Problem

Die Ebene sei mit Parallelen vom Abstand d überdeckt. In diese Ebene lässt man eine Nadel der Länge a fallen. Wie groß ist die Wahrscheinlichkeit, dass die Nadel eine der Parallelen schneidet?

4.2 Allgemeine Lösung

In diesem Abschnitt wird zunächst das allgemeine Verfahren zur Lösung besprochen. Dies lässt sich dann durchführen, wenn die Sinusfunktion integriert werden kann.

Es sei x der Abstand des Mittelpunkts der Nadel von der nächsten Geraden. Damit gilt

$$0 \leq x \leq \frac{d}{2} \quad \text{und für den Winkel} \quad 0 \leq \alpha \leq \pi. \quad (1)$$

Der maximale Abstand x_{max} , so dass die Nadel eine der Parallelen schneiden kann, ist $x_{max} = \frac{a}{2}$. Schließt die Nadel mit den Parallelen den Winkel α ein, so überlagert sie im Fall

$$x \leq \frac{a}{2} \sin \alpha \quad (2)$$

eine der Parallelen.

Um eine Wahrscheinlichkeit berechnen zu können, muss zuerst geklärt werden, was der Ereignisraum ist. Hierfür ergibt sich nach (1)

$$\Omega = \{(\alpha, x): 0 \leq \alpha \leq \pi, 0 \leq x \leq \frac{d}{2}\}.$$

Das Ereignis zum Schneiden einer der Parallelen ist nach (2)

$$\begin{aligned} A &= \{\text{Die Nadel schneidet eine Parallele}\} \\ &= \{(\alpha, x) \in \Omega: 0 \leq x \leq \frac{a}{2} \sin \alpha\}. \end{aligned}$$

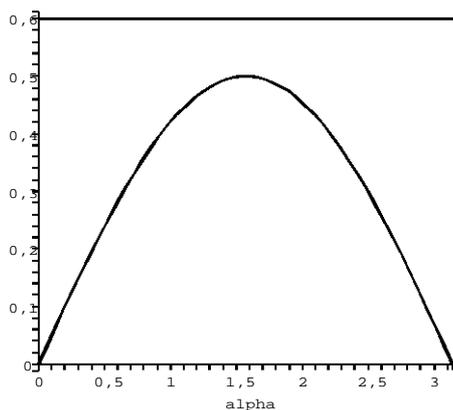


Abb. 6: Graphen zu A und Ω

Stellt man A und Ω durch Funktionen dar – vgl. hierzu Abbildung 6, wobei $a = 1$ und $d = 1.2$ gesetzt wurden – so ergibt sich die Eintrittswahrscheinlichkeit von A zu

$$P(A) = \frac{P(A)}{P(\Omega)} = \frac{\text{Fläche}(A)}{\text{Fläche}(\Omega)}.$$

Für den Zähler gilt

$$\text{Fläche}(A) = \int_0^\pi \frac{a}{2} \sin \alpha \, d\alpha = -\frac{a}{2} \cos \alpha \Big|_0^\pi = a,$$

und der Nenner ist

$$\text{Fläche}(\Omega) = \pi \frac{d}{2},$$

also gilt

$$P(A) = \frac{a}{\pi \frac{d}{2}} = \frac{2a}{\pi d}.$$

Wählt man $d = 2a$, so folgt

$$P(A) = \frac{1}{\pi}.$$

4.3 Näherungsweise Bestimmung von π

Es werden zufällig n Punkte in der Ω -Fläche gewählt (Stichprobenumfang). Dann wird die Anzahl k dieser Punkte bestimmt, die innerhalb der A -Fläche liegen (Trefferquote).

Für einen großen Stichprobenumfang gilt dann nach der Laplace-Regel

$$\frac{\text{\# Punkte zwischen Kurve und } x\text{-Achse}}{\text{\# Punkte insgesamt}} = \frac{k}{n} \approx P(A),$$

$$\pi = \frac{1}{P(A)} \approx \frac{n}{k}$$

Dieses lässt sich auch mithilfe eines Computers durchführen, auswerten und graphisch darstellen.

4.4 Verfahrensweise in der Sekundarstufe I

Die obige Abschätzung unter Verwendung der Näherung von $\int_0^\pi \frac{a}{2} \sin \alpha \, d\alpha$ aus Abschnitt 4.3 für π lässt sich auch in der Sekundarstufe I durchführen, wenn π nach der Definition und einer simplen Abschätzung etwas genauer bestimmt werden soll. Dabei sollte von vornherein der Abstand zweier benachbarter Parallelen gleich der doppelten Länge der Nadel gewählt werden.

Hier muss mithilfe eines Näherungsverfahrens (z.B. des Mehreckverfahrens oder einer praktischen Umfangstimmung eines Kreises von gegebenem Radius) eine ungefähre Bestimmung von π auf etwa eine Nachkommastelle (d.h. $\pi \approx 3.1$) vorgenommen worden sein. Damit lässt sich dann der Wert für $\int_0^\pi \frac{a}{2} \sin \alpha \, d\alpha$ berechnen. Dieses Programm ist ähnlich wie das Programm aus Abschnitt 3.1 aufgebaut und sieht unter MAPLE folgendermaßen aus:

Programm

```
> restart;
```

Zuerst wird die Funktion eingegeben.

```
> f := x -> a/2*sin(x);
```

Nun wird noch der Parameter a und die Anzahl n der Wiederholungen eingegeben:

```
> a := 3: n := 10000:
```

An dieser Stelle ist die obere Grenze des Intervalls einzugeben (ungefähr der Wert für π), in dem das Integral berechnet werden soll.

```
> r := 3.1:
```

```
> for i from 1 to n do
> x[i] := evalf(rand()
/10^12*r):
```

```
> y[i] := abs(f(x[i])):
```

```
> od:
```

Dann wird der Mittelwert der Funktionswerte berechnet:

```
> m := 1/n*sum(y[k], k=1..n):
```

Es folgt die Berechnung der Fläche des Rechtecks mit der Breite des Intervalls und der Höhe des Mittelwerts der Funktionswerte.

```
> F := r*m;
```

Mit diesem Programm unter MAPLE lässt sich näherungsweise die Fläche von A ermitteln, und man erhält nach zahlreichen Durchläufen für unterschiedliche a das Ergebnis $F(A) \approx a$.

Damit wurde der Flächeninhalt zwischen dem Graphen von A und der x -Achse bestimmt. $F(\Omega) = \pi a$ wird wie oben beschrieben bestimmt, und damit ergibt sich $P(A) = \frac{1}{\pi}$. Andererseits gilt $\frac{1}{\pi} \approx \frac{k}{n}$, also

$$\pi \approx \frac{n}{k}$$

Ein Versuch lässt sich einerseits praktisch durchführen, jedoch auch mit einem Computer simulieren. Letzteres kann mit einem Programm umgesetzt werden, das durch folgende Algorithmendarstellung beschrieben wird:

Algorithmus

Eingabe: $n \in \mathbb{N}$ (Anzahl der Schritte)

$a \in \mathbb{N}$ (Länge der Nadel)

1. Setze $k = 0$ (Anzahl der Überschneidungen der Nadel mit einer Kante).
2. n -mal werden die folgenden Schritte wiederholt:
 - a) Wähle zufällig einen Winkel $\alpha \in [0^\circ, 180^\circ[$.
 - b) Wähle zufällig ein $x \in [0, d]$.
 - c) Es sei $y := \frac{a}{2} \cdot \sin \alpha$.
 - d) Wenn $x + y \geq d$ oder $x - y \leq 0$, dann setze die Anzahl der Überschneidungen der Nadel mit einer Kante um 1 hoch.

Ausgabe:

$$\frac{n}{k}$$

Das zugehörige Programm kann unter MAPLE folgendermaßen aussehen:

Programm

```
> restart:
```

Nachfolgend wird eine Prozedur definiert, mit der die Umrechnung von Winkel- in Bogenmaß durchgeführt wird. Dies sollte für die SchülerInnen nicht unmittelbar zugänglich sein, da hierdurch die Aufmerksamkeit in die falsche Richtung geleitet wird.

```
> alpha := proc()
> convert(180*degrees*
evalf(rand()/10^12), radians)
> end proc:
```

Es wird die Anzahl der Würfe der Nadel festgelegt:

```
> n := 10000:
```

Nun werden die Länge der Nadel und der Abstand der Kanten definiert:

```
> a := 1: d := 2*a:
```

Der Zähler für die Überschneidungen der Nadel an der Kante wird gleich null gesetzt:

```
> k := 0:
```

Hier beginnt die eigentliche Schleife:

```
> for i from 1 to n do
> x := d*evalf(rand()/10^12):
> y := a/2*evalf(sin(alpha())):
> if x+y >= d or x-y <= 0 then
>     k := k+1:
> fi:
> od:
> evalf(n/k);
```

Das Programm ist jedoch nicht exakt. Führt man es mit $n = 10000$ Schritten aus, so ergibt als Näherungswert $\pi \approx 3.15457$. Für einen Umfang von $n = 2 \cdot 10^6$ erhält man $\pi \approx 3.1436$. Hierbei beträgt die Rechenzeit an einem Computer mit einem Pentium-4-Prozessor $t \approx 40$ min; es ist daher in diesem Umfang während des Unterrichts nicht gut durchführbar und kann besser als Ergebnis, das man zuhause erhalten hat, erwähnt werden.

5 Fazit

Wie in den Abschnitten 2 bis 4 zu erkennen war, sind die Möglichkeiten des Computers begrenzt. SchülerInnen ist dies heutzutage nicht klar; sie trauen dem Computer alles zu. Um diese Erwartungshaltung zu relativieren, ist der Umgang mit dem Computer in Bezug auf Zufallszahlen sinnvoll.

In den Abschnitten 3 und 4 wurden auch die Grenzen der einzelnen Programme deutlich. Es sind jedoch auch die Möglichkeiten zu erkennen, dass überhaupt so große Stichprobenumfänge und gute Veranschaulichungen möglich sind. Auf diese Art und Weise machen Computer-Simulationen von stochastischen Experimenten didaktisch Sinn.

Außerdem lassen sich mithilfe des Computereinsatzes mehrere Problemstellungen bearbeiten, die normalerweise erst in der Sekundarstufe II behandelt würden. Und es können mehrere Bereiche der Schulmathematik einbezogen werden:

- näherungsweise Bestimmung der Kreiszahl π
- Wahrscheinlichkeitsrechnung, Zufallsexperimente
- Simulation stochastischer Experimente am Computer
- Grundlagen von Flächenberechnung und Integration in der Sekundarstufe I

Insbesondere lässt sich dieses Konzept in der Klasse 9 oder 10 der Sekundarstufe I umsetzen, wodurch sich

- Möglichkeiten des Vorziehens mancher Bereiche der Mathematik aus der Oberstufe in die Mittelstufe,
- sinnvoller Zusammenbau von Mathematik unterschiedlicher Bereiche und
- fächerübergreifender Unterricht der Fächer Mathematik und Informatik

ergeben.

Damit lässt sich ein Computer in der Sekundarstufe I problem- und schülerorientiert einsetzen.

Der oben beschriebene Kurs wurde bereits mit einer Gruppe von SchülerInnen aus der Jahrgangsstufe 10 von Realschulen und Gymnasien durchgeführt; anschließend wurde eine ausführliche anonyme Befragung der SchülerInnen vorgenommen. Die Ergebnisse waren, dass sie das Thema interessant, einleuchtend und von der Durchführung nicht zu schwer fanden. Die SchülerInnen fühlten sich zwar sehr gefordert, das Thema fesselte Sie jedoch so sehr, dass sie bereit waren, hohes Engagement zu zeigen. Dabei waren sie sehr überrascht über die Pseudo-Zufallszahlen, da sie sich bis dahin keine Gedanken über die Bestimmung von Zufallszahlen durch den Computer gemacht hatten. Es erschien ihnen

merkwürdig, dass man Zahlen, die so bestimmt werden, überhaupt Zufallszahlen nennt.

Insbesondere schien den SchülerInnen die Berechnung von Flächen zwischen einem Funktionsgraphen und der x -Achse nützlich. Das Nadelproblem sahen sie als eine schöne Anwendung des zuvor erarbeiteten Stoffes. Einigen SchülerInnen wurde auf diese Art der Sinn der Flächenberechnung mithilfe von bestimmten Integralen klar, was sie als eine Steigerung ihrer Motivation auffassten.

Literatur

- [Gr] H. Griesel et al.: *Elemente der Mathematik, LK Analysis*. Schrödel Verlag 2001.
- [Hä] O. Häggström: *Finite Markov Chains and Algorithmic Applications*. Oxford University Press, 2002.
- [Kü] H.R. Künsch: *Stochastische Simulation*. Skript zur Vorlesung im WS 03/04. ETH Zürich, 2003,
<http://stat.ethz.ch/~kuensch/>
- [Moe] O. Moeschlin et al: *Experimental Stochastics*. Second Edition. Springer, 2003.
- [Stp] H. Stoppel: *Mathematik anschaulich, Brückenkurs mit Maple*. Oldenbourg, 2002.
- [Stp2] H. Stoppel: *Nichtlineare Dynamik und Mandelbrotmenge*. Skriptum zu Mathematik AG, 2002.
- [W] A. Walz: *Maple 7, Rechnen und Programmieren*. Oldenbourg, 2002.

Anschrift des Verfassers

Hannes Stoppel

Fakultät für Mathematik

Ruhr-Universität Bochum

NA 2/26

D-44780 Bochum

stoppel@math.ruhr-uni-bochum.de