

**Schüler-SimuLab**  
**Kursreihe stochastische**  
**Simulationen**  
**Kurs 1**

**Die Erzeugung von**  
**Pseudo-Zufallszahlen:**  
**Der Lineare Kongruenz-Generator**  
**und Statistische Tests**

Stefan Hartmann  
Forschungszentrum caesar

15. Januar 2008



# Ein paar einleitende Worte

Habt ihr auf dem Computer schon mal ein Glücksspiel, sagen wir: Roulette, gespielt und euch gefragt, wie da eigentlich der Zufall ins Spiel kommt? Verwunderlich ist das ja schon: Wie kann es denn sein, dass ein Programm (also eine Abfolge völlig vorherbestimmter Rechenschritte) ein zufälliges, also unvorhersehbares Resultat liefert? Ist denn nicht alles vorhersehbar, was in einem Computeralgorithmus berechnet wird? Doch, das ist es!

**Ein Computer ist eine deterministische Maschine: Mit ihm lassen sich nur deterministische Prozesse durchführen.**

Falls dieser Begriff neu für euch ist: Ein Prozess heißt **deterministisch**, wenn zu jedem Zeitpunkt während des Prozesses bestimmt ist, wie der Prozess weitergeht.

*von lat.  
determinare  
= bestimmen*

In vielen Computern gibt es aber doch vorgegebene **Zufallszahlengeneratoren**, mit denen stochastische Zahlen erzeugt werden, oder etwa nicht? Wie könnte man ansonsten ein Roulette-Spiel programmieren? Irgendwie scheint das mit der Aussage oben, Computer seien deterministische Maschinen, nicht zusammen zu passen. Diesen scheinbaren Widerspruch werden wir auflösen.

*Generator  
= Erzeuger  
stochastisch  
= zufällig*

Wir werden uns in diesem Kurs mit den angesprochenen Zufallszahlengeneratoren näher beschäftigen und sie einerseits als vollkommen deterministisch entlarven. Wir werden aber andererseits sehen, dass man damit dennoch den Zufall **simulieren** kann, indem man algorithmisch Zahlenfolgen erzeugt, die wichtige Eigenschaften von echten Zufallszahlenfolgen aufweisen. Solche Kriterien, die ein Zufallszahlengenerator unbedingt erfüllen sollte, versuchen wir aufzustellen. Die durch einen Zufallsgenerator erzeugten Zahlenfolgen scheinen auf den ersten Blick zufällig zu sein, sind es aber nicht. Sie werden **pseudozufällige Zahlenfolgen** genannt. Mit ihnen lassen sich zufällige Zahlenfolgen (je nach Beschaffenheit des Generators mehr oder weniger gut) modellieren.

*von lat.  
simulare  
= nachahmen*

Anschließend werden wir eine spezielle Klasse von Zufallszahlengeneratoren, die diese Kriterien zum großen Teil erfüllen, kennenlernen, nämlich die sogenannten **Linearen Kongruenz-Generatoren**. Hinter diesen Linearen Kongruenz-Generatoren steckt ein wenig Mathematik, mit der wir uns zunächst vertraut machen müssen. Sobald wir dieses mathematische Rüstzeug beherrschen (im Prinzip handelt es sich nur um das schon aus der Grundschule bekannte „Teilen mit Rest“, aber dennoch erfordert diese Mathematik eine gewisse Übung), wird es uns leichter fallen diese Zufalls-generatoren zu verstehen und auch deren Schwächen zu erkennen. Wir werden versuchen herauszufinden, wie man diese Schwächen so gut wie möglich vermeiden kann, aber eine grundsätzliche, systematische Schwäche bleibt immer bestehen, auch das werden wir erkennen. Im weiteren Verlauf des Kurses werden wir (pseudo-)zufällige Zahlenfolgen auf ihre Güte hin testen und abschließend ein Verfahren kennenlernen, bei dem man durch Kopplung zweier durch Zufallsgeneratoren erzeugten Zahlenfolgen zum Teil bessere Zufalls-zahlen erhält.

Ihr werdet in der Lage sein, selber mit Hilfe des Computers pseudozufällige Zahlenfolgen zu generieren, die dem jeweiligen konkreten Problem angepasst sind. Ihr könnt euch also zum Beispiel euer eigenes Roulette-Spiel programmieren.

Ich wünsche euch viel Spaß bei diesem Kurs und hoffe, dass ihr eine Menge dazulernt.

Bonn, den 06.05.2004

Stefan Hartmann

# Kapitel 1

## Die Erzeugung von Zufallszahlen

*Any one who considers arithmetical  
methods of producing random digits is,  
of course, in a state of sin.*<sup>1</sup>

### 1.1 Zufallszahlen und zufällige Zahlenfolgen

Was ist eigentlich eine Zufallszahl? Ist 2 eine Zufallszahl? Oder  $\pi$ ? Oder 4,59674344543543567? Nein, sicherlich nicht. Dies sind alles deterministische Zahlenwerte. Daher ist der Begriff „eine Zufallszahl“ eigentlich irreführend. Was wir stattdessen meinen, ist eine „zufällige Zahlenfolge“, also eine Folge von unabhängigen zufälligen Zahlen. Dies bedeutet, salopp gesprochen: Jede Zahl der Folge wird zufällig erzeugt und hat mit den anderen Zahlen der Folge nichts zu tun. Wenn die Zahlen einer solchen Folge nur endlich viele Werte annehmen können, dann erwarten wir, dass jeder Wert mit der gleichen Wahrscheinlichkeit angenommen wird. Nehmen wir mal an, wir hätten eine Folge von Zufallszahlen, die die Werte  $0, 1, 2, \dots, 9$  annehmen. Dann erwarten wir, dass jeder Wert mit der Wahrscheinlichkeit  $\frac{1}{10}$  angenommen

---

<sup>1</sup>Zitat von John von Neumann (1951). John von Neumann, geboren 1903 in Budapest, nach kurzer Tätigkeit in Hamburg und Berlin als Professor für Mathematik in Princeton, USA, tätig, gestorben 1957, gilt als einer der genialsten Menschen aller Zeit, begründete unter anderem die Spieltheorie, erarbeitete die theoretischen Grundlagen zur Konstruktion eines Computers und erzielte wichtige Beiträge zur Quantenmechanik.

wird. Jedes denkbare Paar zweier aufeinander folgender Zahlen sollte mit der Wahrscheinlichkeit  $\frac{1}{100}$  angenommen werden, jedes Tripel (also drei aufeinander folgende Zahlen) mit der Wahrscheinlichkeit  $\frac{1}{1000}$ , usw. Man beachte: Wenn wir eine Folge von 1.000.000 zufälligen Zahlen haben, dann ist die Wahrscheinlichkeit sehr klein, dass wir genau 100.000 Nullen, 100.000 Einsen, usw. haben. Wenn wir aber sehr, sehr viele solcher Folgen betrachten und den Mittelwert der relativen Häufigkeiten bilden, dann wird sich dieser Mittelwert langfristig immer mehr dem erwarteten Wert, also  $\frac{1}{10}$ , annähern. Dieses Verhalten wird als **Gesetz der Großen Zahlen** bezeichnet.

Wichtig hierbei ist: Selbst wenn wir vorher 999.999 Ziffern gezogen haben, die alle nicht gleich 0 sind, bleibt die Wahrscheinlichkeit, dass die letzte Ziffer eine 0 ist, gleich  $\frac{1}{10}$ , erhöht sich also nicht. Die Vergangenheit spielt also keine Rolle. Dies liegt daran, dass die Ziffern nicht voneinander abhängen sollen.

Das waren jetzt alles ziemlich unexakte und eher umgangssprachliche Aussagen über Folgen von Zufallszahlen. Es ist gar nicht so einfach zu definieren, was eine „zufällige Zahlenfolge“ sein soll. Daran haben sich viele Mathematiker und Philosophen versucht. Der österreichische Mathematiker Richard von Mises (1883-1953) etwa versuchte es mit der folgenden Definition:

*Eine Folge von lauter Nullen und Einsen heißt zufällig, wenn es keine Regel gibt, die an irgendeiner Stelle das nächste Glied aus den vorhergehenden mit einer Wahrscheinlichkeit von mehr als 50% prognostiziert.*

Der Nachteil an dieser Definition ist, dass nicht präzisiert wird, was eine „Regel“ ist.

Die heute nach wie vor akzeptierte Definition stammt von dem berühmten russischen Mathematiker A. N. KOLMOGOROV (1903-1987) und dem noch lebenden amerikanischen Mathematiker G.J. CHAITIN aus der Mitte der 60er Jahre, die sie unabhängig voneinander entwickelten. Naiv formuliert lautet sie so:

*Eine Folge von lauter Nullen und Einsen heißt zufällig, wenn sie sich nicht durch eine kürzere Folge beschreiben lässt.*

Zur Erläuterung: Die Folge 010101... ist in diesem Sinne nicht zufällig, denn man kann sie ja durch die kürzere Folge 01 beschreiben. Natürlich ist nach dieser Definition jede algorithmisch erzeugte Zahlenfolge nicht zufällig, denn für ihre Beschreibung genügt (zusammen mit dem Algorithmus) bereits der Startwert. Mit dem Computer können wir aber Zahlenfolgen nur algorithmisch erzeugen. Daher muss man aus der jeweiligen Anforderung heraus neue, angepasste Kriterien dafür entwickeln, was man unter „guten zufälligen Zahlenfolgen“ verstehen will.



A. N. Kolmogorov



G.J. Chaitin

Nun stellen sich mindestens drei Fragen:

- (1) Wie kann man (pseudo-)zufällige Zahlenfolgen erzeugen, falls das überhaupt möglich ist?
- (2) Was kann man mit solchen (pseudo-)zufälligen Zahlenfolgen anfangen?
- (3) Wie kann man erkennen, wie „gut“ solche (pseudo-)zufälligen Zahlenfolgen sind?

Auf die letzte Frage lässt sich zumindestens schon mal eine Teilantwort geben. Wenn sich eine Person beliebige Zahlen ausdenkt, lassen sich die in der Regel nicht durch eine kürzere Zahlenfolge ausdrücken. Dennoch kann es natürlich sein, dass jemand (zum Beispiel!) ziemlich oft (aber nicht immer) nach der 1 die 2 als nächste Ziffer aussucht. Eine solche Folge wird sicherlich nicht als

„gute“ zufällige Zahlenfolge angesehen werden. Diese Schwächen sind aber nicht immer sofort zu erkennen.

### Aufgabe 1:

Es wurden auf zwei Arten Zufallszahlen zwischen 1 und 6 (zur Simulation eines Würfelexperimentes) erzeugt:

#### 1. Beispiel:

1 3 4 2 5 6 6 3 2 1 4 4 5 3 2 1 1 4 3 6  
3 2 2 4 5 3 3 1 1 4 1 5 3 2 1 6 6 5 4 3  
2 4 1 1 2 4 3 6 6 1 4 5 5 2 3 4 1 1 2 6  
4 3 3 2 1 6 6 5 4 1 3 2 2 4 3 2 1 4 6 3  
3 2 1 4 6 3 3 2 1 4 6 3 3 2 1 4 6 3 3 2  
1 4 6 3 3 2 1 4 6 3 3 2 1 4 6 3 3 2 1 4

#### 2. Beispiel:

1 1 2 3 2 4 1 2 5 2 6 3 4 6 3 4 2 4 3 4  
6 5 6 3 2 5 6 1 5 1 3 1 4 1 4 6 5 6 6 2  
3 4 6 2 4 2 1 5 3 5 4 6 4 5 6 5 3 3 5 6  
4 2 6 1 6 1 2 4 3 4 2 5 4 5 3 5 2 3 4 2  
3 6 3 1 3 2 4 3 5 3 6 2 2 1 5 1 5 6 3 6  
4 5 1 3 1 6 1 3 2 1 6 1 3 4 5 4 5 3 2 5

Würdest du diese Zahlenfolgen als „gute zufällige Zahlenfolgen“ einstufen? Wenn nein, warum nicht? Bei dem 1. Beispiel erkennst du vielleicht sofort die Schwäche. Beim 2. Beispiel ist es schwieriger zu erkennen.

(Tipp zum 2. Beispiel: Schau dir mal die Paare aufeinander folgender Zahlen an. Wie oft kommt zum Beispiel das Paar (3, 4) (also erst eine 3 und dann direkt danach eine 4) vor? Wie oft kommt ein „Pasch“ vor?)

Hier sind alle Paare zum 2. Beispiel aufgelistet:

	1	2	3	4	5	6
1	1	3	5	2	4	3
2	3	1	4	5	4	2
3	3	5	1	7	4	3
4	2	5	3	0	5	5
5	3	2	5	3	0	6
6	5	3	5	3	3	1

absolute Häufigkeiten aller Zahlenpaare

Was vielleicht ganz interessant ist: Das zweite Beispiel ist dadurch entstanden, dass man eine Person gebeten hat, eine Folge zufälliger Zahlen zwischen 1 und 6 zu nennen. Obwohl also kein Algorithmus dahintersteckt, der vielleicht zu starre Rechnungen ausführt, erhalten wir trotzdem ein schlechtes Ergebnis. Das liegt an der folgenden Tatsache: Menschen tendieren dazu alles zu vermeiden, was „nicht zufällig aussieht“, so zum Beispiel zwei gleiche Ziffern direkt hintereinander. Unser Gefühl sagt uns, dass es unwahrscheinlich ist, dass zwei gleiche Ziffern hintereinander vorkommen, also ein „Pasch“. Das ist natürlich Blödsinn. Das Ereignis, dass zweimal hintereinander eine 1 kommt, sollte genauso wahrscheinlich sein wie das Ereignis, dass erst eine 1

und dann eine 5 kommt. Umgekehrt: Würden wir einer anderen Person eine Tabelle mit ausgewürfelten, also echten Zufallszahlen geben, so würde diese Person diese Tabelle vielleicht nicht als zufällig einstufen, da man angeblich irgendwo Muster oder angeblich zu viele Pasche erkennt.

Man darf sich also nicht zu sehr auf sein Gefühl verlassen, wenn man (Pseudo-)Zufälligkeit beurteilt!

Wir werden später sehen, wie man objektive Tests auf die Güte einer (pseudo-)zufälligen Zahlenfolge systematisch durchführt.

## **1.2 Zufallszahlengeneratoren: ein historischer Überblick**

### **1.2.1 Zufallszahlentafeln und echte Zufallsgeneratoren**

Vor vielen Jahren, weit vor der Einführung von Computern, war es sehr schwierig zufällige Zahlenfolgen zu erzeugen. Man musste zufällige Experimente tatsächlich durchführen und die Ergebnisse notieren. Auf diese Weise gelangte man zu echt zufälligen Zahlenfolgen, etwa durch das Ziehen von Kugeln aus Urnen, durch Würfeln oder durch Münzwürfe. Man kann sich vorstellen, mit welchem ungeheuren Zeitaufwand das verbunden war!

Der französische Naturforscher G. BUFFON (1707-1788) warf eine Münze 4040 mal und erreichte dabei 2048 mal „Kopf“ und 1992 mal „Zahl“. Außerdem schätzte er den Wert der Zahl  $\pi$ , indem er Nadeln auf eine gestreifte Fläche warf und dabei zählte, wie oft die Nadeln die Linien berührten (dieses Experiment ist unter dem Namen „Buffon-Nadel-Experiment“ bekannt).

Der englische Biologe W. F. R. WELDON zeichnete 26 306 Würfe mit 12 Würfeln auf, während der Schweizer Naturwissenschaftler R. WOLF 100 000 Würfe mit einem einzigen Würfel durchführte und registrierte.

Der Statistiker K. PEARSON analysierte den Ausgang zahlreicher Roulette-spiele, notierte sich die Zahlen und stellte dabei fest, dass die Roulettescheibe einen systematischen Fehler (einen sogenannten „bias“) enthielt, d.h. die Zahlen traten nicht annähernd mit der gleichen relativen Häufigkeit auf. Bei dieser Gelegenheit kam ihm die Idee zu einem statistischen Test, dem sogenannten „Chi-Quadrat-Test“, der nach wie vor eine große Bedeutung hat und den wir später im Kurs auch noch kennenlernen werden.

Ähnlich zeitaufwändig war sicherlich die Methode von L. H. C. TIPPETT: Er veröffentlichte 1927 eine Tafel mit 41.600 zufälligen Dezimalziffern, die er aus Daten der Finanzbehörde gewonnen hatte.

Bei **echten Zufallszahlengeneratoren** handelt es sich um mechanische oder elektrische Geräte, die zufällige Zahlenfolgen (im Sinne der obigen Definition) liefern.

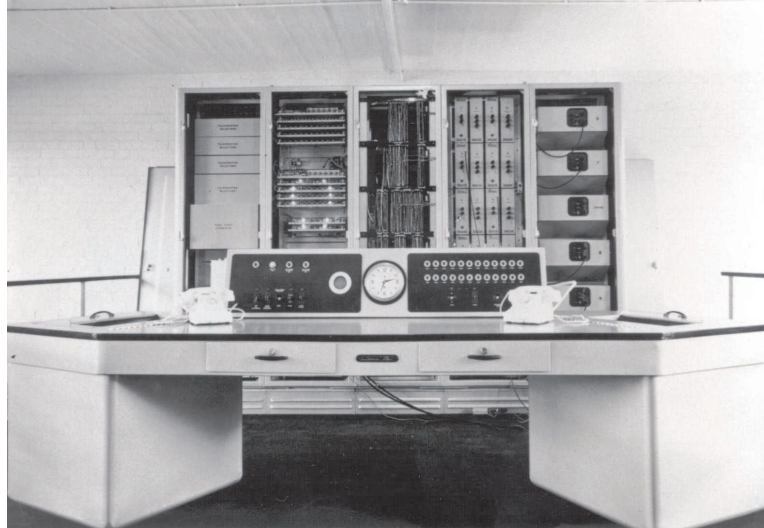
Ein Beispiel ist eine Tafel von M. G. KENDALL und B. BABINGTON-SMITH, die im Jahre 1939 durch mechanische Experimente erstellt wurde. Im ersten kommerziell hergestellten Computer, dem FERRANTI MARK I, wurden echte Zufallszahlen durch einen Rauschgenerator erzeugt.

Im Jahre 1955 druckte die RAND COOPERATION ein Buch mit 1.000.000 Zufallszahlen ab, die durch elektrisches Rauschen erzeugt wurden:

Zeile Nr.	Spalte Nr.									
	1-5	6-10	11-15	16-20	21-25	26-30	31-35	36-40	41-45	46-50
0	10097	32533	76520	15586	54675	14876	80959	09117	39292	74945
1	37342	04805	64894	74296	24805	24057	20636	10402	00822	91665
2	08422	68953	19645	09503	23209	02560	15953	34764	35080	55606
3	99019	02529	09376	70613	38311	31165	88676	74597	04436	27659
4	12807	99970	80157	56147	64052	56653	98951	16877	22171	76833
5	66065	74717	54072	76850	56697	56170	65815	59885	11199	29170
6	51060	10805	45571	82406	55505	42614	86799	07459	23403	09732
7	85269	77602	02051	65692	68665	74818	75053	85247	18623	88579
8	65575	52155	05525	47048	90553	57548	28468	28709	85491	25624
9	75796	45755	05129	64778	35808	54222	60955	20544	55275	88455
10	98520	17767	14905	68607	22109	40558	60970	95433	50500	75998
11	11805	05455	59808	27722	50725	68242	29505	24201	52775	67851
12	85412	99654	06188	98085	15746	70078	18475	40610	68711	77817
13	88685	40200	86507	18401	56766	67951	90564	76495	29609	11062
14	99594	67548	87517	64969	91826	08928	93781	61568	25478	34115
15	65481	17674	17468	50950	58047	76974	73059	57186	40218	16544
16	80124	55653	17727	08015	45518	22574	21115	78215	14585	55765
17	74550	99817	77402	77214	45236	00210	45521	64257	96286	02655
18	69916	26805	66252	29148	56956	87205	76621	15990	94400	56452
19	09893	20505	14225	68514	46427	56788	96297	78822	54522	24598
20	91499	14523	68479	27686	46162	85514	94750	89925	57089	20048
21	80336	94598	26940	56858	70297	54555	55540	55540	42050	82541
22	44204	81949	85157	47914	52979	26575	57600	40881	22222	06455
23	12550	75742	11100	02040	12860	74697	96644	89459	28707	25815
24	65606	49329	16505	54484	40219	52565	45651	77082	07207	51790
25	61196	90446	26457	47774	55924	55729	65394	59595	42582	60527
26	15474	45266	95270	79955	59567	85848	82596	10112	33211	59466
27	94557	28575	67897	54587	54622	44451	91590	42592	92927	45975
28	42481	16215	97544	08721	16868	48767	05071	12059	25701	46670
29	25525	78517	75208	89857	68955	91416	26252	29665	05522	82562
30	04495	52494	75246	55824	45864	51025	61962	79555	65557	12472
31	00549	97654	64051	88559	96159	65896	54692	82591	25287	29529
32	55965	55507	26898	09554	55551	55462	77974	50024	90505	59555
33	59808	08591	45427	26842	85609	49700	55022	24892	78565	20506
34	46058	85256	05590	92286	77281	44077	93910	85647	70617	42941
35	52179	00597	87579	25241	05567	07007	86745	17557	85594	11858
36	69254	61406	20117	45204	55956	60000	18745	92225	97518	96558
37	19565	41430	01758	75579	40419	25585	66674	56806	84962	85207
38	45555	14958	19476	07246	45667	94545	59047	90053	20826	69541
39	94864	55994	56168	10851	54888	85555	05540	55456	05014	51176
40	98086	24826	45240	28404	44999	08896	59094	75407	55441	51880
41	55185	16252	45941	50949	89455	48581	88695	45994	57548	75045
42	80951	00406	96582	70774	20551	25587	25016	25298	24624	61171
43	79752	49140	75961	28296	69861	02591	74852	20559	00587	59579
44	18655	52557	98145	06571	51050	24674	05455	61427	77958	95958
45	74029	45902	77557	52270	97790	17159	52527	58021	80814	51748
46	54178	45612	80995	57145	05555	12969	56127	19255	56040	90324
47	15664	49883	52079	84827	59581	75559	09975	55440	88461	25556
48	48524	77928	51249	64710	02295	56870	52507	57546	55020	09994
49	69074	94558	87657	95976	55584	04401	10518	25655	01848	76958

Ein Auszug aus dem Buch der RAND COOPERATION

Ein sehr berühmtes Beispiel für einen elektrischen Zufallszahlengenerator ist der Generator ERNIE (Electronic Random Number Indicator Equipment).



ERNIE

Die Erstellung von Zufallszahlen geschieht hier durch die Erzeugung von Geräuschimpulsen bei Neonröhren, die mit Hilfe von Zählrichtungen in Zufallszahlen umgewandelt werden. Verwendung findet ERNIE immer noch bei Ziehungen der Gewinner einer staatlichen Lotterie in Großbritannien.

Der Vorteil von „echten“ Zufallszahlengeneratoren ist, dass man „gute“ Zufallszahlen bekommt, mit denen man sehr realistisch simulieren kann. Perioden oder ähnliches, also typische Gefahren bei „künstlichen Zufallszahlengeneratoren“ (darauf wird später noch detailliert eingegangen) treten nicht auf.

Die Nachteile der beschriebenen Methoden liegen aber auch auf der Hand: Erstens ist die Erstellung von Tafeln sehr zeitaufwändig und zweitens ist der Vorrat an Zufallszahlen beschränkt, was Simulationen im großen Umfang nicht zulässt. Zufallszahlen, die durch mechanische oder elektrische Geräte erzeugt werden, haben den Nachteil, dass es auf diese Weise unmöglich ist, genau die gleichen Zufallszahlen noch einmal zu erzeugen. Die einmal erzeugten Zufallszahlen lassen sich nicht mehr reproduzieren, d.h. eventuell aufgetretene Fehler bei der Erzeugung können nicht mehr erkannt werden. Ein weiterer Nachteil, der bei echten Zufallszahlengeneratoren auftreten kann, ist der, dass sich die Wahrscheinlichkeiten, mit denen gewisse Zahlen erzeugt

werden, durch Verschleiß oder ähnliches im Laufe der Zeit ändern können.

Man kann sich echte Zufallszahlen im Internet unter [www.random.org](http://www.random.org) erzeugen lassen.

## 1.2.2 Pseudozufällige Zahlenfolgen

Die Nachteile von „echten“ Zufallszahlengeneratoren führten mit dem Aufkommen neuer leistungsstarker Computergenerationen zu einem steigenden Interesse an der Fragestellung, ob und wie man zufällige Zahlenfolgen mit Hilfe arithmetischer Operationen, die von einem Computer algorithmisch durchgeführt werden, erzeugen kann. Wie bereits in der Einleitung erwähnt, ist das nicht möglich, da ein Computer eine deterministische Maschine ist. Die Folgen erscheinen aber im günstigsten Fall als zufällig. Die von einem „guten“ algorithmischen Generator erzeugten Zufallszahlen besitzen zufriedenstellende (jedoch nicht beliebig gute!) statistische Fähigkeiten bezüglich Gleichverteilung (d.h. jede Zahl wird mit der gleichen Wahrscheinlichkeit erzeugt) und Unabhängigkeit (d.h. die Erzeugung einer beliebigen Zahl in der Folge hängt nicht von der Erzeugung der anderen Zahlen ab), sie sind jedoch exakt reproduzierbar. Solche Folgen werden „pseudozufällige Zahlenfolgen“ genannt, wir werden sie aber im Folgenden häufig auch einfach nur als „zufällige Zahlenfolgen“ oder auch nur als „Zufallszahlen“ bezeichnen, auch wenn das nicht ganz korrekt ist.

Der älteste algorithmische Zufallszahlengenerator ist der von JOHN VON NEUMANN (von dem das Zitat am Anfang des Kapitels stammt!) etwa 1946 vorgeschlagene „Mitten-Quadrat-Generator“. Was war die Motivation dahinter? Die Physiker der „Los Alamos Scientific Laboratory“, zu denen VON NEUMANN zählte, brauchten während des Zweiten Weltkrieges Informationen darüber, wie weit Neutronen durch verschiedene Materialien gelangten, um das geeignete Material zur Abschirmung vor solchen Neutronen zu finden. Rein aus theoretischen Berechnungen ließ sich das nicht feststellen. Daraufhin schlug VON NEUMANN vor, dieses Problem zu lösen, indem man dieses Experiment auf dem Computer modellierte und Simulationen durchführte. Dieses geheime Projekt brauchte einen Codenamen. JOHN VON NEUMANN wählte dafür den Namen „Monte Carlo“. Seitdem wird diese Methode der Simulation auch „Monte-Carlo-Methode“ genannt. Für solche Simulationen braucht man sehr viele Zufallszahlen. Von Neumann suchte also nach ge-

eigneten Algorithmen und schlug dabei (wenn auch erst nach dem Ende des Zweiten Weltkrieges) den „Mitten-Quadrat-Generator“ vor. Wie funktioniert dieser nun? Ganz einfach: Man startet zum Beispiel mit einer vierstelligen Zahl, quadriert sie und nimmt als nächste Zufallszahl die mittleren vier Ziffern des Quadrats. (Wenn das Quadrat nur aus sieben Ziffern besteht, ist vorne eine 0 zu ergänzen, wie im untenstehenden Beispiel.)



J. v. Neumann

Beispiel:

$$x_0 = 1234 \quad x_0^2 = 01 \underbrace{5227}_{=x_1} 56$$

$$x_1 = 5227 \quad x_1^2 = 27 \underbrace{3215}_{=x_2} 29$$

$$x_2 = 3215 \quad x_2^2 = 10 \underbrace{3362}_{=x_3} 25$$

usw.

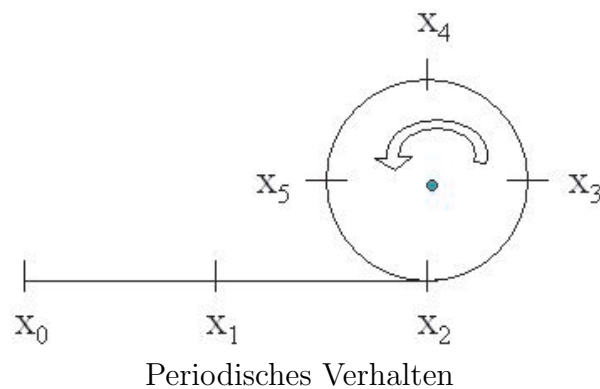
Auf den ersten Blick scheint dies ein vernünftiger Zufallszahlengenerator zu sein: Man wählt ja mittendrin ein paar Ziffern aus, auf Dauer wird schon jede vierstellige Zahl dort vorkommen, schließlich werden die Zahlen durch das Quadrieren ja jedesmal „kräftig durcheinandergewirbelt“. Aber Vorsicht: Es scheint nur so zu sein! Die Gefahr besteht darin, dass die Folge in einen sogenannten Zyklus gerät, sich also ab einem gewissen Zeitpunkt in kurzen

Abständen immer wiederholt.

### **Aufgabe 2:**

Beginne mal mit dem Startwert  $x_0 = 5283$  und führe den Algorithmus ein paar Mal durch. Was fällt dir auf? Wie ist dieser Zufallszahlengenerator auf Grund dieses Ergebnisses zu bewerten?

Das Ergebnis wird in dem folgenden Bild verdeutlicht:



Dieses Verfahren lässt sich sehr kompakt als Algorithmus schreiben. Um dies tun zu können, brauchen wir eine kleine Definition:

Für eine reelle Zahl  $r$  bezeichnen wir mit  $\text{INT}(r)$  den ganzzahligen Anteil dieser Zahl, d.h. wir „schneiden die Zahl hinter dem Komma ab“.

Beispiele:

$$\text{INT}(3,14159) = 3,$$

$$\text{INT}(0,356) = 0,$$

$$\text{INT}(-36,98) = -36.$$

Nun können wir den Algorithmus angeben:

(1) Wähle vierstelligen Startwert  $x(0)$ .

(2) Setze für  $i = 1, \dots, N$ :

$$x(i) := \text{INT}((x(i-1))^2/100)$$

$$x(i) := x(i) - \text{INT}(x(i)/10000) \cdot 10000$$

$$i := i + 1$$

### Zusatzaufgabe:

Versuche den obigen Algorithmus nachzuvollziehen: Warum passiert dort genau das, was passieren soll, sprich also: Warum wird mit Hilfe dieses Algorithmus die aktuelle Zahl quadriert, die mittleren vier Ziffern genommen und diese als neue vierstellige Zahl aufgefasst, mit der das gleiche Spielchen wiederholt wird? Wenn du das nicht direkt siehst (was normal ist), dann gehe wie folgt vor: Setze zunächst mal ein paar konkrete Zahlen ein und schaue, was passiert. Beschreibe mit eigenen Worten, „was der Algorithmus mit der vierstelligen Zahl in jedem Schritt macht“.

Versuche den Algorithmus nun (zunächst in einem Excel-Blatt und dann unter Einsatz von VBA) zu programmieren und teste seine Güte auch anhand anderer Startwerte.

Der Algorithmus hat, wie gesehen, Defizite. Man könnte nun wie folgt argumentieren: Okay, das Verfahren war ein bisschen zu einfach und nicht chaotisch genug. Ich muss den Algorithmus einfach viel komplizierter gestalten:

Einfach mal ein paar Operationen ähnlich zu der obigen hinzunehmen, sie wild durcheinander durchführen und dann werde ich schon einen guten Zufallszahlengenerator bekommen. Der US-amerikanische Informatik-Professor D. E. KNUTH behandelt in seinem Buch „The Art of Computer Programming“, *das* absolute Standardwerk der Informatik, unter anderem sehr detailliert Generatoren von Zufallszahlen. Die Informatik-Legende KNUTH, als solche darf man ihn mittlerweile sicherlich bezeichnen, gibt in diesem Buch einen sehr „chaotischen“ Zufallszahlengenerator an, um die obige These zu widerlegen:

### Algorithmus K (der „super-zufällige“ Zufallszahlengenerator)

Gegeben sei eine zehnstellige positive Zahl  $X$ . Der folgende Algorithmus beschreibt, auf welche Weise diese verändert wird. Das Ergebnis wird als neue Zahl in der zufälligen Zahlenfolge aufgefasst.

- K1.** [Zufällige Wahl der Anzahl Iterationsschritte]  $Y$  sei die „Milliardenstelstelle“ von  $X$ . Die Schritte K2 bis K13 werden nun genau  $Y + 1$  mal durchgeführt.
- K2.** [Zufälliger Sprung zu einem Iterationsschritt]  $Z$  sei die „Hundertmillionstelstelle“ von  $X$ . Gehe zum Iterationsschritt  $K(3 + Z)$ .
- K3.** [Sorgt dafür, dass  $X \geq 5\,000\,000\,000$ ] Falls  $X < 5\,000\,000\,000$  gilt, dann setze  $X = X + 5\,000\,000\,000$ .
- K4.** [„Mitten-Quadrat-Generator“] Siehe oben, nur diesmal mit zehn Stellen statt mit vier, also: Nehme das Quadrat von  $X$  und wähle davon die mittleren zehn Stellen als neues  $X$ .
- K5.** [Multiplikation] Multipliziere  $X$  mit  $1\,001\,001\,001$  und wähle von dem Ergebnis die letzten zehn Stellen als neues  $X$ .
- K6.** [Pseudo-Komplement] Falls  $X < 100\,000\,000$  gilt, setze:  $X = X + 9\,814\,055\,677$ . Andernfalls setze:  $X = 10\,000\,000\,000 - X$ .

- K7. [Vertauschung der Hälften]** Vertausche die ersten fünf Ziffern der zehnstelligen Zahl  $X$  mit den letzten fünf Ziffern.
- K8. [Multiplikation 2]** Mache das gleiche wie in K5.
- K9. [Verkleinerung der Ziffern]** Verkleinere jede der Ziffern von  $X$ , die nicht gleich 0 sind, um eins.
- K10. [Modifizierung mit 99 999]** Falls  $X < 100\,000$  gilt, setze:  $X = X^2 + 99\,999$ . Andernfalls setze:  $X = X - 99\,999$ .
- K11. [Normierung]** Falls  $X < 100\,000\,000$  gilt, dann setze  $X = 10 \cdot X$  und wiederhole diesen Schritt.
- K12. [Modifizierter „Mitten-Quadrat-Generator“]** Multipliziere  $X$  mit  $X - 1$  und wähle von dem Ergebnis die mittleren zehn Stellen als neues  $X$ .
- K13. [Wiederholung?]** Falls  $Y > 0$  gilt, dann erniedrige  $Y$  um 1 und kehre zu Schritt K2 zurück. Wenn  $Y = 0$  gilt, dann ist der Algorithmus beendet. Das aktuelle  $X$  ist das nächste Glied der „zufälligen“ Zahlenfolge.



D. E. Knuth

Es ist jetzt nicht erforderlich, dass man sich jeden Schritt im Algorithmus ganz genau anschaut. Wichtig ist, dass man erkennt, wie „chaotisch“ hier verfahren wird. Da ja häufig „chaotisch“ mit „stochastisch“ gleichgesetzt wird, sollte man meinen, dass hier ein besonders guter Zufallszahlengenerator entstanden ist. Solche Zyklen wie im „Mitten-Quadrat-Generator“ sollten hier wohl nicht vorkommen, gerade bei zehnstelligen Zahlen! Vermutlich wird dieser Generator, unabhängig davon, was man als Anfangswert wählt, eine riesige Anzahl von Zufallszahlen liefern.

**Aber: Meistens kommt es anders als man denkt!**

Bei manchen Startwerten konvergiert der Algorithmus relativ schnell gegen 6 065 038 420. Und, der Zufall will es so, dieser Wert (und damit auch alle nachfolgenden) wiederholt sich dann jeweils nach 44 Iterationsschritten (probiere es doch einfach mal aus, wenn dir langweilig ist). Wir haben also einen Zyklus der Länge 44 vorliegen. Bei anderen Startwerten wiederum wiederholt sich der Algorithmus erstmals nach 7401 Werten, dort liegt dann ein Zyklus der Länge 3178 vor. Man sieht also: Dieser Algorithmus hat relativ kleine Zyklen, die man zudem nicht vorhersehen kann. Er ist also zur Erzeugung von Zufallszahlen völlig unbrauchbar: Das Chaos, das ja eigentlich dazu dienen sollte, „besonders gute zufällige Zahlen zu erzeugen“, ist zu groß, als dass man diese Zyklen vorhersehen könnte.

**Merke: Zufallszahlen sollten niemals mit einer zufälligen Methode erzeugt werden!**

Im folgenden werden wir die Erzeugung von Zufallszahlen nicht mehr dem Zufall überlassen. Wir werden systematisch vorgehen und Zufallszahlengeneratoren kennenlernen, deren Eigenschaften man ganz genau analysieren kann und bei denen man nachweisen kann, dass sie gewisse Eigenschaften haben, die man von einem Zufallszahlengenerator fordert. Solche unliebsamen Überraschungen wie bei Algorithmus K werden dann nicht mehr auftreten.

Aber, mal was anderes: Warum waren die Menschen eigentlich schon immer so sehr an der Erzeugung von Zufallszahlen interessiert?

Fällt dir eine Anwendung ein? Denk mal kurz darüber nach. Anschließend kannst du dann auf der nächsten Seite nachschauen.

## 1.3 Wofür braucht man Zufallszahlen?

Zufällige Zahlenfolgen sind in vielen, völlig verschiedenen Anwendungsgebieten nützlich. Beispiele:

- (a) **Simulationen.** Es können sowohl Naturphänomene (etwa im Bereich der Atomphysik) als auch zeitliche Dynamiken, die in der (Finanz-)Welt des Menschen vorkommen (zum Beispiel Aktienverläufe) stochastisch simuliert werden. Man erhält auf diese Weise eine Vielzahl realistisch erscheinender Szenarien, die einem ein Gefühl dafür geben, „was theoretisch alles passieren kann“ und mit welcher Wahrscheinlichkeit gewisse Ereignisse eintreten.
- (b) **Stichproben.** Da es etwa bei Qualitätskontrollen nicht möglich ist, alle Objekte zu überprüfen, begnügt man sich mit zufällig ausgewählten Stichproben.
- (c) **Computerprogrammierung.** Als zufällige Eingabedaten zum Testen von programmierten Algorithmen sind zufällige Zahlenfolgen sehr nützlich. Zudem gibt es *zufällige Algorithmen*, die sehr effektiv wirken. Auch um Computerspiele, bei denen der Zufall eine Rolle spielt (zum Beispiel Skat oder Roulette) zu programmieren, benötigt man zufällige Zahlenfolgen.
- (d) **Ästhetik.** Ein zufälliger Einfluss lässt Bilder und Musik lebendiger erscheinen. Schau dir zum Beispiel den caesar-Schriftzug an: Bei ihm sind die Abstände zwischen den Buchstaben zufällig.
- (e) **Entscheidungsfindung.** Was glaubst du eigentlich, wie dein Mathelehrer auf deine Note gekommen ist? Okay, das war ein Scherz. Aber bei gewissen Entscheidungsfindungen werden in der Praxis tatsächlich Zufallszahlen herangezogen.

# Kapitel 2

## Ein bisschen Mathematik

Bevor wir uns mit dem Linearen Kongruenz-Generator näher beschäftigen, müssen wir uns zunächst einmal die dahinter liegende mathematische Theorie etwas genauer anschauen.

### 2.1 Division mit Rest

Man weiß, dass die folgende Aussage gilt:

Es seien  $a$  und  $m$  beliebige natürliche Zahlen,  $m > 0$ . Dann gibt es genau zwei natürliche Zahlen  $q$  und  $r$  mit den Eigenschaften:

$$a = q \cdot m + r \quad \text{und} \quad 0 \leq r < m. \quad (2.1)$$

Bevor du weiterliest, denk doch einmal über diese Aussage nach. Ist sie dir „anschaulich“ klar? Wähle doch mal zwei natürliche Zahlen  $a$  und  $m$  mit  $m > 0$ . Hast du welche? Wie lauten jetzt  $r$  und  $q$ ? Wenn du es noch nicht verstehst: Kein Problem, es wird gleich erklärt!

Dieses  $r$  ist durch die obigen Bedingungen also eindeutig bestimmt, wenn man  $a$  und  $m$  erst einmal gewählt hat. Daher können wir die folgende Be-

zeichnung einführen:

**Bezeichnung 2.1 ( $\text{Rest}_m(a)$ )** *Es sei  $m$  eine beliebig gewählte natürliche Zahl mit  $m > 0$ . Für eine natürliche Zahl  $a$  bezeichnen wir im Folgenden mit*

$$\text{Rest}_m(a) = r$$

*den Rest, der auftritt, wenn wir  $a$  durch  $m$  mit Rest teilen.*

Das hört sich viel schwieriger an als es ist. Es bedeutet einfach folgendes: Wir haben zwei natürliche Zahlen,  $a$  und  $m$ , wobei  $m$  nicht gleich 0 sein soll. Nun wollen wir  $a$  durch  $m$  teilen. Wenn  $m$  größer als  $a$  ist, wählen wir einfach  $q = 0$  und  $r = a$ . Nun sei  $m$  kleiner oder gleich  $a$ . Weil  $m$  nicht gleich 0 ist, dürfen wir grundsätzlich schon einmal durch  $m$  teilen. Es kann sein, dass diese Division glatt aufgeht. Dann gibt es eine natürliche Zahl  $q$  mit  $a = q \cdot m$ . In diesem Fall ist  $r = 0$ . Es kann aber auch sein, dass die Division nicht glatt aufgeht. Dann bleibt ein Rest übrig. Man kann es aber erreichen, dass der Rest zumindestens kleiner als  $m$  ist. In diesem Fall sind sowohl  $q$  als auch  $r$  eindeutig bestimmt.

Diese abstrakte Argumentation wollen wir mal an einem konkreten Beispiel verdeutlichen:

Es sei  $a = 10$  und  $m = 3$ . Dann gilt zum Beispiel:

$$10 = 2 \cdot 3 + 4,$$

also:

$$a = q \cdot m + r \quad \text{mit} \quad a = 10, \quad m = 3, \quad q = 2 \quad \text{und} \quad r = 4.$$

Die erste Bedingung in (2.1), also die Gleichung  $a = q \cdot m + r$ , ist also erfüllt. Dagegen gilt aber nicht:  $0 \leq r < m$ , denn wir haben ja  $r = 4 > 3 = m$ . Die

zweite Bedingung in (2.1) ist also nicht erfüllt. Wie können wir es erreichen, dass sie erfüllt ist? Nun, wie kann man die Gleichung:  $10 = 2 \cdot 3 + 4$  interpretieren? Lasst sie uns doch mal so schreiben:

$$10 - 2 \cdot 3 = 4.$$

Wir ziehen also von der 10 zweimal die 3 ab und landen bei 4. Dummerweise ist  $4 > 3$ . Wir haben also die 3 nicht oft genug abgezogen. Wir hätten sie noch ein drittes Mal abziehen können. Tun wir das also! Wir erhalten:

$$10 - 3 \cdot 3 = 1,$$

also:

$$10 = 3 \cdot 3 + 1.$$

Nun ist

$$a = q \cdot m + r \quad \text{mit} \quad a = 10, \quad m = 3, \quad q = 3 \quad \text{und} \quad r = 1.$$

In diesem Fall sind beide Bedingungen in (2.1) erfüllt, denn es gilt ja

$$r = 1 < 3 = m.$$

Beachte bitte: Es gibt für  $a = 10$  und  $m = 3$  keine andere Darstellung der Form (2.1) als die, die wir gerade gefunden haben (also mit  $q = 3$  und  $r = 1$ ).

**Die Darstellung in (2.1) existiert immer und ist eindeutig.**

Anschaulich ist das in unserem Beispiel klar: Würden wir etwa einmal mehr  $m$  von  $a$  abziehen, also viermal, so wäre unser Rest bereits negativ.

Weitere Beispiele:

- $a = 20, m = 7$ :  $20 = 2 \cdot 7 + 6 \Rightarrow \text{Rest}_7(20) = 6,$
- $a = 3, m = 10$ :  $3 = 0 \cdot 10 + 3 \Rightarrow \text{Rest}_{10}(3) = 3,$
- $a = 45, m = 9$ :  $45 = 5 \cdot 9 + 0 \Rightarrow \text{Rest}_9(45) = 0.$

### Gruppenaufgabe

Finde für die folgenden Paare die Division mit Rest gemäß (2.1) und gib  $\text{Rest}_m(a)$  an::

- (a)  $a = 30, m = 6,$
- (b)  $a = 77, m = 8,$
- (c)  $a = 0, m = 1000,$
- (d)  $a = 100, m = 99.$

## 2.2 \*Aus dem Alltagsleben

Die beiden nächsten Abschnitte sind nicht unbedingt für das Verständnis des Linearen Kongruenz-Generators notwendig. Es ist also nicht schlimm, wenn ihr nicht alles versteht. Im Prinzip reicht die aus der Grundschule bekannte „Division mit Rest“ aus dem letzten Kapitel für natürliche Zahlen aus. Aber: Dann wüsste man anschließend gar nicht, was denn das „kongruent“ in dem „Linearen Kongruenz-Generator“ eigentlich bedeutet. Und das fänden wir eigentlich ziemlich schade. (Ihr kennt sicherlich „kongruente Dreiecke“. Hat das was mit dem „kongruent“ im „Linearen Kongruenz-Generator“ zu tun? Ja, aber darauf können wir hier nicht eingehen. Dennoch wollen wir euch

erklären, was man hier speziell unter „kongruent“ versteht.) Kurzum: Wir haben uns zu diesem kleinen mathematischen Ausflug entschieden.

Es geht hier um die sogenannte **Kongruenzrechnung**, auch **Modulorechnung** genannt. Im Wesentlichen kennt ihr das aber schon aus dem Alltagsleben. Schaut euch mal eine Digitaluhr mit einer 24-Stundenanzeige an. Nehmen wir mal an, wir haben 18 Uhr. Wieviel Uhr haben wir dann 10 Stunden später? Genau, 4 Uhr. Man kann sich das zum Beispiel so überlegen:

$$18 + 10 = 28 = 24 + 4 \equiv 4.$$

Das „ $\equiv$ “-Zeichen wird später noch erklärt. Es soll so etwas wie bedeuten wie „zwar nicht unbedingt ganz gleich, aber wenigstens in einem gewissen Sinne ähnlich“. Das werden wir aber später noch ganz genau präzisieren. Hier die Rechnung in Worten erklärt: Wenn die Uhr immer weiter laufen würde, hätte man 28 Uhr. Da sie aber um 24 Uhr auf 0 Uhr umspringt, muss man sozusagen bei 24 Uhr wieder bei 0 Uhr anfangen, d.h. die 28 Uhr entsprechen 4 Uhr. Man kann einfach von der 28 die 24 abziehen und landet bei der 4, und somit einer Zahl zwischen 0 und 23, die von der Uhr angezeigt wird.

Anderes Beispiel: Wir haben 23 Uhr. Wie viel Uhr haben wir 100 Stunden später? Wir rechnen wie eben:

$$23 + 100 = 123 = 120 + 3 = 5 \cdot 24 + 3 \equiv 3.$$

Antwort: Wir haben 100 Stunden später 3 Uhr.

Was haben wir gemacht? Würde die Uhr immer weiter laufen, hätte man nach 100 Stunden 123 Uhr. Wir brauchen aber eine Zahl zwischen 0 und 23. Wir teilen also 123 durch 24 mit Rest. Wie oft passt die 24 in die 123? Fünf Mal, bleibt Rest 3. Anders ausgedrückt: Wir ziehen von der 123 so oft die 24 ab, bis wir zwischen 0 und 23 landen. Wenn wir von 123 nun 5 mal 24 abziehen, landen wir in der Tat bei 3. In diesem Sinne ist die 123 zur 3 „ähnlich“. Beide lassen bei der Division durch 24 den gleichen Rest, nämlich 3. Oder, anders ausgedrückt: Die Differenz der beiden Zahlen, also  $123 - 3 = 120$ , ist durch 24 teilbar.

Wir haben in diesen Beispielen nichts anderes gemacht als „modulo 24“ zu rechnen.

Weiter kennt ihr doch die folgenden Regeln:

- „gerade  $\cdot$  gerade = gerade“,
- „gerade  $\cdot$  ungerade = gerade“,
- „ungerade  $\cdot$  ungerade = ungerade“.

Lasst uns einen Moment mal alle **geraden** natürlichen Zahlen (unter den „natürlichen Zahlen“ verstehen wir an dieser Stelle alle Elemente der Menge  $\mathbb{N}_0 = \{0, 1, \dots\}$ , obwohl die 0 häufig nicht zu den natürlichen Zahlen gezählt wird) mit der 0 identifizieren und alle **ungeraden** natürlichen Zahlen mit 1. Formal ist das so etwas ähnliches wie oben: Wie ziehen von einer natürlichen Zahl so lange die 2 ab, bis wir bei 0 oder 1 landen. Dann entsprechen die obigen Regeln gerade den wohlbekannteren Rechenregeln:

- $0 \cdot 0 = 0$ ,
- $0 \cdot 1 = 0$ ,
- $1 \cdot 1 = 1$ .

In diesem Beispiel haben wir nichts anderes gemacht als „modulo 2“ zu rechnen.

Dieses Konzept, was wir in den Beispielen kennengelernt haben, wollen wir nun verallgemeinern.

## 2.3 \*Kongruenzrechnung

Zunächst einmal sollte man wissen, dass man das Konzept der „Division mit Rest“, das wir ja im letzten Abschnitt nur für natürliche Zahlen formuliert haben, ohne Probleme auch auf ganze Zahlen ausweiten kann. Mit anderen Worten:  $a$  und  $q$  dürfen auch negativ sein, aber Vorsicht: Für  $m$  wird nach wie vor vorausgesetzt, dass es größer als 0 ist (ansonsten hat man keine Eindeutigkeit mehr). Wir wollen das Ganze mal als „Satz“ formulieren. Ein mathematischer „Satz“ ist – wenn man es ganz einfach (und etwas unexakt) formulieren will – eine wahre Behauptung, die man streng logisch beweisen

kann. Eine Behauptung wie „In Bonn regnet es häufiger als in Florenz“ ist zwar offenbar richtig, wie uns die Erfahrung lehrt, aber kein mathematischer Satz, denn wir können ihn nicht streng logisch beweisen.

**Satz 2.2 (Division mit Rest)** *Es seien  $a$  und  $m$  ganze Zahlen, wobei zusätzlich  $m > 0$  vorausgesetzt sei. Dann gibt es genau ein Paar  $(q, r)$  ganzer Zahlen mit den Eigenschaften*

$$a = q \cdot m + r \quad \text{und} \quad 0 \leq r < m. \quad (2.2)$$

Wir verzichten hier auf einen Beweis. Anschaulich ist die Sache aber wieder ziemlich klar: Man subtrahiert von  $a$  (bzw. addiert zu  $a$ ) so lange  $m$ , bis man im Bereich zwischen 0 und  $m - 1$  landet.

□

### Beispiel

Wie teilt man  $a = -77$  durch  $m = 5$  mit Rest? Wir könnten erst einmal 77 durch 5 mit Rest teilen. Wie oft passt die 5 in die 77? 15 Mal, bleibt Rest 2. Also:

$$77 = 15 \cdot 5 + 2.$$

Also könnte man jetzt glauben, dass es bei  $a = -77$  dann eben  $q = -15$  sein muss. Probieren wir das aus:

$$-77 = (-15) \cdot 5 - 2.$$

Hier wäre also  $r = -2$ . Das ist aber nicht das, was wir haben wollen. Wir hätten ja gerne, dass die Ungleichung  $0 \leq r < 5$  erfüllt ist. Wie können wir das retten? Indem wir einfach einmal mehr zur  $-77$  die 5 addieren, also 16 mal. Dann erhalten wir:

$$-77 + 16 \cdot 5 = 3,$$

also

$$-77 = (-16) \cdot 5 + 3$$

und damit die gewünschte Darstellung

$$a = q \cdot m + r \quad \text{mit} \quad a = -77, \quad m = 5, \quad q = -16 \quad \text{und} \quad r = 3.$$

### Gruppenaufgabe

Finde für die folgenden Paare die Division mit Rest gemäß (2.2):

(a)  $a = -100, m = 6,$

(b)  $a = -77, m = 11,$

(c)  $a = -31, m = 10.$

Nun wollen wir endlich erklären, was dieses komische „kongruent“ bedeuten soll. Genauer: Wir definieren, wann wir zwei Zahlen (bei einem vorher gegebenen  $m$ ) „kongruent modulo  $m$ “ nennen.

**Definition 2.3 (kongruent modulo  $m$ )** *Es sei  $m > 0$  beliebig vorgegeben. Dann nennen wir zwei ganze Zahlen  $a$  und  $b$  kongruent modulo  $m$ , wenn sie bei der Division durch  $m$  den gleichen Rest lassen.*

*Hier noch die Schreibweise für „ $a$  ist kongruent zu  $b$  modulo  $m$ “:*

$$a \equiv b \pmod{m}.$$

*Wenn klar ist, welches  $m$  gerade gemeint ist, kann man auch einfach  $a \equiv b$  schreiben.*

Aha! Wenn ich also jetzt feststellen will, ob zwei Zahlen  $a$  und  $b$  kongruent modulo  $m$  sind, muss ich also sowohl  $a$  als auch  $b$  durch  $m$  mit Rest teilen (mit der Darstellung wie in (2.2)). Wenn die beiden Reste (also die  $r$ 's) gleich sind, dann sind  $a$  und  $b$  kongruent modulo  $m$ . Wenn die beiden Reste verschieden voneinander sind, dann sind  $a$  und  $b$  nicht kongruent modulo  $m$ . In diesem Fall schreibt man auch  $a \not\equiv b \pmod{m}$ .

Aber muss ich das denn wirklich jedes Mal so machen? Gibt es denn da kein einfacheres Kriterium? Doch, das gibt es. Wir formulieren es wieder in einem mathematischen Satz, brauchen dazu aber zunächst einmal den Begriff eines Teilers für ganze Zahlen.

**Definition 2.4 (Teiler)** *Es seien  $a$  und  $b$  ganze Zahlen,  $a \neq 0$ . Man sagt,  $a$  teilt  $b$ , wenn es eine ganze Zahl  $q$  gibt mit  $b = q \cdot a$ . Dann nennt man  $a$  einen **Teiler** von  $b$  und  $b$  ein **Vielfaches** von  $a$ .*

Mit anderen Worten: Teilt man  $b$  durch  $a$  mit Rest und gilt für den Rest  $r$ :  $r = 0$ , dann ist  $a$  ein Teiler von  $b$ . Die anschauliche Erklärung eines Teilers für natürliche Zahlen wird also einfach auf die ganzen Zahlen erweitert.

**Satz 2.5** *Es gilt genau dann  $a \equiv b \pmod{m}$ , wenn  $m$  ein Teiler von  $a - b$  ist.*

BEWEIS. Wir müssen zwei Dinge beweisen:

**Behauptung 1:** *Wenn  $a \equiv b \pmod{m}$  gilt, dann ist  $m$  ein Teiler von  $a - b$ .*

Beweis der Behauptung 1:

Es gelte also:  $a \equiv b \pmod{m}$ . Das heißt,  $a$  und  $b$  lassen bei der Division durch  $m$  den gleichen Rest. Mit anderen Worten: Es gibt  $q, q'$  und ein gemeinsames  $r$  mit  $0 \leq r < m$ , so dass folgendes gilt:

$$a = q \cdot m + r$$

und

$$b = q' \cdot m + r.$$

Dann ist aber:

$$a - b = (q \cdot m + r) - (q' \cdot m + r) = q \cdot m - q' \cdot m + r - r = (q - q') \cdot m,$$

also  $m$  in der Tat ein Teiler von  $a - b$ .

**Behauptung 2:** *Wenn  $m$  ein Teiler von  $a - b$  ist, dann gilt  $a \equiv b \pmod{m}$ .*

Beweis der Behauptung 2:

Es sei also  $m$  ein Teiler von  $a - b$ . Dann gibt es ein  $q'$  mit

$$(*) \quad a - b = q' \cdot m.$$

Nun teilen wir  $a$  durch  $m$  mit Rest und erhalten:

$$(**) \quad a = q'' \cdot m + r$$

mit einem  $0 \leq r < m$ . Wir müssen nun zeigen, dass wir den gleichen Rest  $r$  erhalten, wenn wir  $b$  durch  $m$  mit Rest teilen. Aus (\*) und (\*\*) folgt aber:

$$b = a - q' \cdot m = q'' \cdot m + r - q' \cdot m = (q'' - q') \cdot m + r = q \cdot m + r$$

mit  $q = q'' - q'$ . Wir haben also in der Tat bei  $b$  den gleichen Rest  $r$  wie bei  $a$ . Daher sind  $a$  und  $b$  kongruent modulo  $m$ .

□

Was bedeutet das nun? Wenn wir feststellen wollen, ob  $a$  und  $b$  kongruent modulo  $m$  sind, müssen wir einfach deren Differenz bilden, also  $a - b$ , und schauen, ob diese Differenz ein Vielfaches von  $m$  ist. Wenn ja, dann sind  $a$  und  $b$  kongruent modulo  $m$ . Wenn nein, dann sind  $a$  und  $b$  nicht kongruent modulo  $m$ . Umgekehrt: Wie erhalten wir aus einer beliebigen ganzen Zahl  $a$  alle Zahlen, die kongruent zu  $a$  modulo  $m$  sind? Indem wir beliebig oft  $m$  zu  $a$  addieren bzw. von  $a$  subtrahieren. Mit anderen Worten: Alle Zahlen, die einen Abstand zu  $a$  haben, der gerade ein Vielfaches von  $m$  ist, sind kongruent zu  $a$  modulo  $m$ .

### Beispiele

- (a) Sind  $a = -10$  und  $b=11$  kongruent modulo 7?

Ja, denn es gilt:  $a - b = -10 - 11 = -21$  und  $-21$  ist ein Vielfaches von 7 wegen  $-21 = (-3) \cdot 7$ .

- (b) Welche Zahlen sind kongruent zu 8 modulo 11?

Da die Differenz der gesuchten Zahlen zu  $a = 8$  ein Vielfaches von  $m = 11$  sein soll, brauchen wir einfach nur zu 8 beliebig oft 11 zu addieren beziehungsweise von 8 beliebig oft 11 abzuziehen. Die zu 8 kongruenten Zahlen sind also:

$$\dots, -25, -14, -3, 8, 19, 30, \dots$$

### Gruppenaufgabe

- (a) Sind 6 und -15 kongruent modulo 3? Sind 10 und 210 kongruent modulo 1000?
- (b) Welche Zahlen sind kongruent zu 2 modulo 13?

**Bezeichnung 2.6 ( $\text{Rest}_m(a)$ )** *Es sei  $m$  eine beliebig gewählte ganze Zahl mit  $m > 0$ . Für eine ganze Zahl  $a$  bezeichnen wir im folgenden mit*

$$\text{Rest}_m(a)$$

*den Rest, der auftritt, wenn wir  $a$  durch  $m$  mit Rest teilen. Aus dem letzten Abschnitt wissen wir jetzt:*

*$\text{Rest}_m(a)$  ist also genau diejenige Zahl, die kongruent zu  $a$  modulo  $m$  ist (davon gibt es unendlich viele Zahlen) und für die zusätzlich*

$$0 \leq \text{Rest}_m(a) < m$$

*gilt (dadurch ist sie dann eindeutig bestimmt).*

## Kapitel 3

# Der Lineare Kongruenz-Generator

So, jetzt haben wir erst einmal genug Mathematik gemacht. Nun wollen wir unsere Zufallszahlen aber auch mal langsam erzeugen!

Der Lineare Kongruenz-Generator wurde im Jahr 1949 von dem amerikanischen Mathematiker D. H. LEHMER (1905-1991), der sich auch im Bereich der Zahlentheorie große Dienste erworben hat, eingeführt.



D. H. Lehmer

Der Lineare Kongruenz-Generator ist häufig immer noch ein Bestandteil vieler Zufallszahlengeneratoren, wobei es mittlerweile auch bessere, sogenannte Nichtlineare und Inverse Kongruenzgeneratoren, gibt. Wir werden später sehen, wo die Schwächen des Linearen Kongruenz-Generators liegen.

Der Lineare Kongruenz-Generator arbeitet nach dem folgenden Algorithmus:

**1. Schritt:** Wähle

- einen **Modul**  $m$  mit  $m > 0$ ,
- einen **Multiplikator**  $a$  mit  $0 \leq a < m$ ,
- eine **Verschiebung**  $c$  mit  $0 \leq c < m$ ,
- einen **Startwert**  $X_0$  mit  $0 \leq X_0 < m$ .

**2. Schritt:** Berechne für  $i \geq 0$ :

$$X_{i+1} = \text{Rest}_m(a \cdot X_i + c).$$

**3. Schritt:** Breche ab, sobald eine Periode eintritt.

In Worten:

Wähle dir  $m$ ,  $a$ ,  $c$  und einen Startwert  $X_0$ .

Berechne dann  $a \cdot X_0 + c$ . Teile diesen Ausdruck dann durch  $m$  mit Rest. Den entstehenden Rest nennst du  $X_1$ .

Jetzt berechnest du  $a \cdot X_1 + c$ . Teile diesen Ausdruck dann durch  $m$  mit Rest. Den entstehenden Rest nennst du  $X_2$ .

Jetzt berechnest du  $a \cdot X_2 + c$ . Teile diesen Ausdruck dann durch  $m$  mit Rest. Den entstehenden Rest nennst du  $X_3$ .

Und so weiter.

Breche ab, sobald ein Folgenglied auftritt, das vorher schon mal vorgekom-

men ist.

### **Gruppenaufgabe:**

Führe den Algorithmus durch für  $m = 10$ ,  $X_0 = 1$ ,  $a = 7$  und  $c = 7$ . Was fällt dir auf?

Man sieht an dem letzten Beispiel, dass man, wenn man  $m$ ,  $X_0$ ,  $a$  und  $c$  willkürlich wählt, sehr viel „Pech“ mit den erzeugten zufälligen Zahlenfolgen haben kann. Es kann durchaus passieren, dass relativ frühzeitig eine Periode auftritt. Solche zufälligen Zahlenfolgen sind natürlich nicht besonders brauchbar. Wir müssen also Sorge tragen, dass so etwas nicht vorkommt und unsere Werte von  $m$ ,  $a$ ,  $c$  und  $X_0$  mit Bedacht wählen, so dass wir eine sehr gute zufällige Zahlenfolge erhalten. Aber wie genau? Gibt es da vielleicht Regeln? (Wenn nicht, dann wäre der Lineare Kongruenz-Generator keinesfalls besser als der Mitten-Quadrat-Generator!) Und: Was heißt eigentlich in diesem Zusammenhang „sehr gut“? Was ist das Beste, was wir erreichen können?

### **Gruppenaufgabe:**

Nehmen wir mal an, wir haben  $m$ ,  $X_0$ ,  $a$  und  $c$  irgendwie gewählt.

Nach wie vielen Schritten tritt allerspätestens wieder eine Wiederholung auf? Was ist also die größtmögliche Periodenlänge?

Jetzt wissen wir also, was wir bestenfalls erwarten dürfen. Wie aber können wir das erreichen? Zum Glück haben die Mathematiker dafür Regeln herausgefunden. Sie klingen auf den ersten Blick etwas kompliziert, sind aber in der Praxis recht einfach zu überprüfen:

**Satz 3.1** Die durch einen Linearen Kongruenz-Generator erzeugte Zahlenfolge hat genau dann die größtmögliche Periodenlänge, also  $m$ , wenn die folgenden Bedingungen erfüllt sind:

- (a)  $c$  und  $m$  sind teilerfremd.
- (b)  $a - 1$  ist durch jede Primzahl teilbar, durch die  $m$  teilbar ist.
- (c) Wenn  $m$  durch 4 teilbar ist, dann ist auch  $a - 1$  durch 4 teilbar.

Man sieht also: Die Werte von  $a$ ,  $m$  und  $c$  sind mit Bedacht zu wählen. Sie sollten zumindestens den obigen Bedingungen genügen. Dagegen ist es egal, wie man den Startwert  $X_0$  wählt. Für jedes  $X_0$  bekommt man eine andere Folge von Zufallszahlen.

### **Aufgabe 3:**

Wir betrachten den Linearen Kongruenz-Generator mit  $a = 5$ ,  $m = 8$  und  $c = 1$ . Sind dann die obigen Bedingungen erfüllt? Führe den Algorithmus jetzt mal für ein beliebiges  $X_0$  mit  $0 \leq X_0 < m$  durch und schaue, ob du wirklich eine Periode der maximalen Länge  $m = 8$  erhältst.

Kann man denn jetzt sagen: Okay, wenn ich  $c$ ,  $m$  und  $p$  so wähle, dass die obigen Bedingungen erfüllt sind, dann ergibt sich automatisch eine „gute“ zufällige Zahlenfolge? Versucht mal die folgende Aufgabe, dann habt ihr eine vorläufige Antwort auf diese Frage.

### Zusatzaufgabe:

Nun betrachten wir den Linearen Kongruenz-Generator mit  $a = 1$ ,  $m = 8$  und  $c = 1$ . Sind dann die obigen Bedingungen erfüllt? Führe den Algorithmus jetzt mal für ein beliebiges  $X_0$  mit  $0 \leq X_0 < m$  durch und schaue, ob du wirklich eine Periode der maximalen Länge  $m = 8$  erhältst. Wie beurteilst du diese „zufällige“ Zahlenfolge? Handelt es sich um einen „guten“ Zufallszahlengenerator?

Man sieht anhand der letzten Aufgabe, dass man auch zu wenig brauchbaren Zufallszahlen kommen kann, wenn die obigen Bedingungen an den Operator formal erfüllt sind. Dann ist die Periodenlänge zwar optimal, also gleich  $m$ , aber die Zahlen wirken trotzdem nicht zufällig, z.B. wenn sie (wie in der Zusatzaufgabe nach Aufgabe 3) ständig anwachsen. (Es gibt auch deutlich kompliziertere Tücken, die man dem Algorithmus nicht sofort ansieht.) Wir brauchen also irgendwelche Verfahren, die irgendwie „messen“, wie gut unsere Zufallszahlengeneratoren sind. Solche Verfahren, sogenannte „statistische Verfahren“, werden wir später kennenlernen. Sie sagen was darüber aus, ob nebeneinander liegende Zahlen „nicht zu sehr in Zusammenhang“ stehen, also in gewisser Weise „unabhängig“ voneinander zu sein scheinen. Wenn –wie in einer der obigen Aufgaben– die Folge immer um 3 anwächst, dann deutet das auf eine „hohe Abhängigkeit“ aufeinander folgender Zahlen hin. Der Test sollte dann also negativ ausfallen. Wir werden uns später damit beschäftigen. Erst einmal dürft ihr jetzt aktiv werden, und zwar auch auf dem Computer!

#### Aufgabe 4:

Versuche mal mit Hilfe des Linearen Kongruenz-Generators in Excel Zufallszahlen zu erzeugen. Mache dir erst einmal klar, was du alles brauchst:

- Felder, wo man  $m$ ,  $a$ ,  $c$  und  $X_0$  eintragen kann
- die Formel  $X_1 = \text{Rest}_m(a \cdot X_0 + c)$
- ganz viele Kopien dieser Formel, nur mit  $X_i$  anstatt  $X_0$  und  $X_{i+1}$  anstatt  $X_1$ .

Beachte bitte: In Excel schreibt man

$$\text{Rest}_m(a \cdot X_i + c)$$

als

$$= \text{REST}((a * X_i + c); m)$$

(das „=-“-Zeichen muss in Excel vor jeder Formel stehen), wobei für  $a$ ,  $c$  und  $m$  feste Felder und für  $X_i$  ein variables Feld einzusetzen ist.

Jetzt erzeugst du mit deinem Zufallsgenerator 20000 Zufallszahlen mit den Parametern

$$a = 313 \quad , \quad m = 16384 \quad , \quad c = 3271 \quad \text{und} \quad X_0 = 0.$$

Es ist normal, dass du das vielleicht nicht alleine kannst. Aber du willst ja was lernen!

#### Zusatzaufgabe:

Versuche diesen Algorithmus nun auch unter dem Einsatz von VBA zu programmieren.

# Kapitel 4

## Transformation von Zufallszahlen

Mit unserem Zufallszahlengenerator können wir nun Zufallszahlen erzeugen, die in der Menge

$$\{0, 1, 2, \dots, m - 1\}$$

liegen. Aber es kann ja sein, dass wir gerade diese Zahlen gar nicht wollen, sondern uns einen anderen Wertebereich wünschen. Häufig will man reelle (eigentlich rationale, da die Dezimalbruchentwicklung auf dem Computer immer abbricht) Zufallszahlen zwischen 0 und 1 erzeugen.

### Gruppenaufgabe:

Überlegt euch ein Verfahren, wie man aus einer beliebigen Folge natürlicher Zufallszahlen eine Folge reeller Zufallszahlen konstruieren kann, deren Werte zwischen 0 und 1 liegen. Geht jetzt noch mal in eure Excel-Programme. Erweitert eure Programme so, dass ihr nun auch Zufallszahlen aus dem Bereich  $[0, 1) = \{x \in \mathbb{R} : 0 \leq x < 1\}$  erzeugen könnt und führt dieses Verfahren für eure bereits erzeugten Zufallszahlen durch.

Jetzt können natürlich Probleme der unterschiedlichsten Arten vorkommen. Es könnte zum Beispiel sein, dass wir ein Lottospiel simulieren wollen. Dann bräuchten wir als Zufallszahlen die ganzzahligen Werte zwischen 1 und 49. Ein anderes Beispiel lernst du in der nächsten Aufgabe kennen.

### Gruppenaufgabe:

Wir nehmen einmal an, wir wollen 100 zufällige Würfelzahlen erzeugen. Dazu brauchen wir Zufallszahlen zwischen 1 und 6. Naja, aber eigentlich genügen uns ja Zahlen zwischen 0 und 5, zu denen wir jeweils 1 addieren. Es gibt zunächst einmal mehrere Möglichkeiten dies zu planen:

#### Versuch 1

Wenn man direkt mit dem Modul  $m = 6$  rechnet, dann bekommt man ja sofort Zufallszahlen zwischen 0 und 5, also Zufallszahlen, die direkt im richtigen Bereich liegen. Nun suchen wir uns einen Linearen Kongruenz-Generator, der den obigen Bedingungen genügt, damit wir die maximale Periodenlänge  $m$  bekommen, also auch wirklich alle Zahlen zwischen 0 und 6 treffen. Es müssen ja (erste Bedingung) der Modul  $m$  und die Verschiebung  $c$  teilerfremd sein. Wir wählen jetzt mal  $c = 5$ . Die nächste Bedingung lautet ja: „ $a - 1$  ist durch jede Primzahl teilbar, die durch  $m = 6$  teilbar ist. Wir müssen also dafür sorgen, dass  $a - 1$  durch 6 teilbar ist, wobei  $a$  der Multiplikator ist. Wir wählen  $a = 1$ , um das zu erreichen. Als Startwert  $X_0$  wählen wir einfach  $X_0 = 0$ . Dann erhalten wir den folgenden Linearen Kongruenz-Generator:

$$X_{i+1} = \text{Rest}_6(1 \cdot X_i + 2).$$

Du kannst das Verfahren ja mal ausprobieren. Benutze dazu bitte die entsprechende Spalte des Excel-Arbeitsblattes „Linearer Kongruenz-Generator“.

#### Versuch 2

Wir brauchen ja 100 Werte. Also ist es eventuell sinnvoll als Modul  $m = 101$  zu wählen, denn dann bekommen wir ja zumindestens schon mal 101 verschiedene Werte, wenn wir uns an die Bedingungen an die Parameter halten. Wir könnten auch  $m = 100$  wählen, aber  $m = 101$  ist unter naiven (!) Gesichtspunkten (scheinbar!) besser, weil 101 eine Primzahl ist und dann die Bedingungen einfacher zu überprüfen sind. Wir könnten zum Beispiel  $a = 1$ ,  $m = 101$  und  $c = 3$  nehmen.

Aber: Diese 101 Werte haben natürlich noch nicht den Wertebereich, den wir gerne hätten. Sie nehmen ja Werte zwischen 0 und 100 an, wir brauchen aber Werte zwischen 0 und 5. Was machen wir also? Wir teilen anschließend noch mal alle Werte durch 6 mit Rest, rechnen also „modulo 6“. Dann haben wir also den Generator

$$X_{i+1} = \text{Rest}_6(\text{Rest}_{101}(1 \cdot X_i + 5)).$$

Probiere auch dieses Verfahren aus. Benutze dazu bitte die entsprechende Spalte des Excel-Arbeitsblattes „Linearer Kongruenz-Generator“.

### Versuch 3

Wir machen das gleiche wie bei Versuch 2, nur jetzt mit den Parametern aus Aufgabe 6, also mit  $a = 313$ ,  $m = 16384$  und  $c = 3271$ . Benutze dazu bitte die entsprechende Spalte des Excel-Arbeitsblattes „Linearer Kongruenz-Generator“.

Wie beurteilst du die drei Verfahren? Was sind die Nachteile?

Wir sehen also: Sowohl ein schlechter Ansatz als auch ein schlechter Zufallszahlengenerator können zu schlechten Ergebnissen führen. Wir müssen unsere Zufallszahlengeneratoren also gut testen, bevor wir sie einsetzen, sonst sind unsere Simulationen auch mies.

Der Versuch 3 liefert die beste Folge. Aber es können Tücken auftreten, mit denen man nicht unbedingt rechnet! Nicht immer sieht man den Zufallszahlengeneratoren direkt an, ob sie für den jeweiligen Zweck gut geeignet sind. Denn: Auch wenn die erzeugten Zufallszahlen insgesamt nicht schlecht sind, also einen „zufälligen“ Eindruck machen, kann es sein, dass einzelne Ziffern, insbesondere die letzte Ziffer, alles andere als zufällig wirken. Schau dir dazu bitte die folgende Aufgabe an:

### Zusatzaufgabe:

Nehmen wir einmal an, wir wollen Zufallszahlen von 0 bis 9 erzeugen, also zufällige Ziffern. Dann können wir uns ja zunächst einen Linearen Kongruenz-Generator mit  $m = 10000$ ,  $a = 3141$  und  $c = 1$  wählen (überprüfe bitte, dass die Kriterien für eine maximale Periodenlänge erfüllt sind) und dann

$$X_{i+1} = \text{Rest}_{10}(\text{Rest}_{10000}(3141 \cdot X_i + 1))$$

setzen. Führe das bitte (für einen beliebigen Startwert  $X_0$ ) durch. Was fällt dir auf?

(Für den Mathecrack: Kannst du deine Vermutung auch beweisen?)

Man sieht also: Auch wenn die Zufallszahlen auf den ersten Blick „gut“ wirken, können Tücken auftreten, wenn wir diese Zufallszahlen wie hier durch erneutes Dividieren mit Rest auf einzelne Zifferblöcke reduzieren, die dann vielleicht nicht mehr zufällig wirken.

Daher ist es gut, wenn man auch noch andere Methoden kennt. Eine möchten wir euch jetzt vorstellen.

Was wir zum Beispiel machen können, ist Folgendes: Wir transformieren eine beliebige Folge von Zufallszahlen,

$$X_0, X_1, X_2, \dots,$$

die wir durch einen Linearen Kongruenz-Generator mit Modul  $m$  erzeugt haben, zunächst durch  $U_n = \frac{X_n}{m}$  auf eine Folge

$$U_0, U_1, U_2, \dots$$

mit Wertebereich  $[0, 1)$ . (Wenn wir schon Zufallszahlen aus  $[0, 1)$  vorliegen haben, wie bei den Excel-Zufallszahlen, dann starten wir an diesem Punkt.) Nun nehmen wir mal an, wir wollen daraus für ein beliebiges  $d \in \mathbb{N}$  Zufallszahlen aus der Menge

$$0, 1, 2, \dots, d - 1$$

erzeugen. Wie können wir das tun? Bevor du weiterliest, solltest du zunächst einmal kurz selber darüber nachdenken.

Die Idee ist die folgende: Wir zerlegen das Intervall  $[0, 1)$  in  $d$  gleich große Teilintervalle und ordnen jedem Teilintervall eine Zahl zu: dem ersten Teilintervall die 0, dem zweiten Teilintervall die 1, usw. Da alle Teilintervalle gleich groß sind und im Idealfall in jedem Teilintervall gleich viele Zahlen liegen, hoffen wir darauf, eine recht gute Verteilung der Zahlen  $0, 1, \dots, d - 1$  erhalten.

Wie kann man das mathematisch ausdrücken? Erinnerst du dich noch an die Funktion INT? (Für eine reelle Zahl  $r$  haben wir mit  $\text{INT}(r)$  den ganzzahligen Anteil dieser Zahl bezeichnet, d.h. wir haben „die Zahl hinter dem Komma abgeschnitten“.)

Nun setzen wir einfach:

$$Y_n = \text{INT}(d \cdot U_n).$$

### Zusatzaufgabe:

(a) Es sei  $d = 7$ . Welche Werte nimmt dann  $Y_n$  an, wenn  $U_n = 0.05$ ,  $U_n = 0.178$ ,  $U_n = 0.29$ ,  $U_n = 0.75$  und  $U_n = 0.99$  ist?

(b) Kannst du für ein beliebiges  $U_n$  genau die Bereiche  $I_0, I_1, \dots, I_6$  (als Intervalle) angeben, für die

$$\text{INT}(6 \cdot U_n) = i \quad (i = 0, 1, \dots, 6)$$

gilt?

(c) Kannst du (b) auch für ein beliebiges  $d$  verallgemeinern?

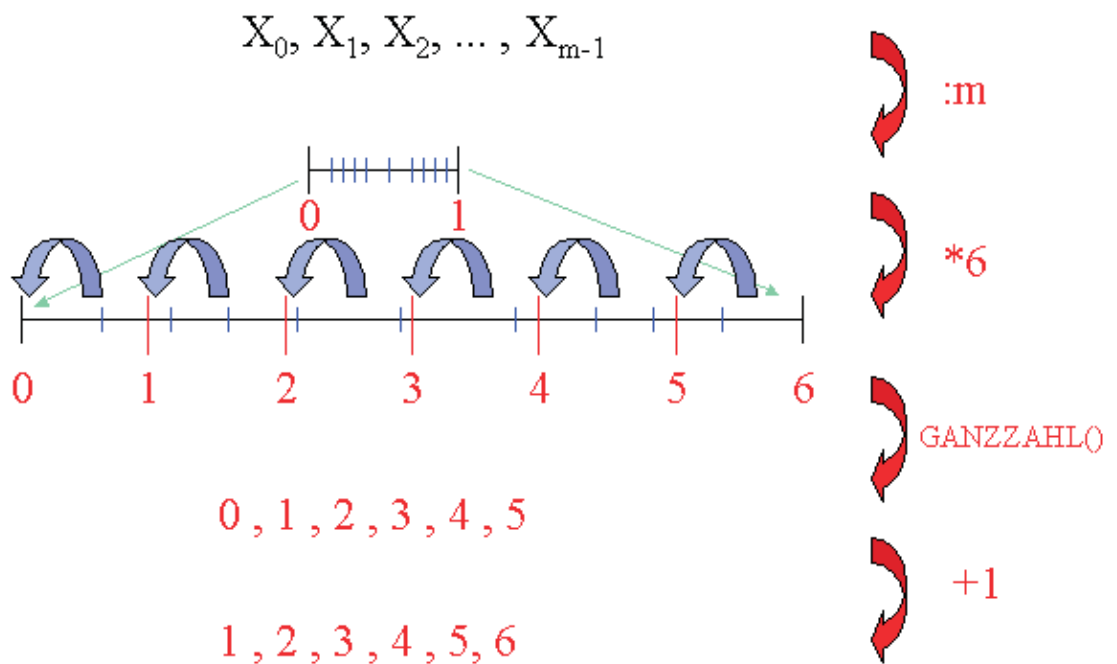
Nun wollen wir uns zum Abschluss mit unserem erworbenen Wissen überlegen, wie man einen Würfel simulieren kann:

Wir erzeugen uns zunächst Zufallszahlen aus dem Bereich  $[0, 1)$  (siehe oben) und dann mittels

$$Y_n = \text{INT}(6 \cdot U_n)$$

zufällige Zahlen aus der Menge  $\{0, 1, 2, 3, 4, 5\}$ . Dann müssen wir nur noch 1 hinzu addieren.

Schematisch sieht das Verfahren also wie folgt aus:



Schema zur Erzeugung zufälliger Würfelzahlen

### Aufgabe 5:

Simuliere nun mit dem gerade erlernten und oben noch einmal skizzierten Verfahren tausend zufällige Würfe mit einem Würfel. Verwende dazu die entsprechende Spalte des Excel-Arbeitsblattes „Linearer Kongruenz-Generator“ und die Parameter

$$a = 313 \quad , \quad m = 16384 \quad , \quad c = 3271 \quad \text{und} \quad X_0 = 0,$$

wie in Aufgabe 4.

Lass den Computer zählen, wie oft eine „1“, „2“, „3“ usw. vorkam. Verwende dabei den Befehl ZÄHLENWENN (siehe Excel-Hilfe).

Versuche das Ergebnis zu interpretieren: Von wie vielen Versuchen hast du wie oft eine bestimmte Zahl „gewürfelt“? Bilde die Verhältnisse, also die relativen Häufigkeiten. Überlege nun: Wie groß sind die (theoretischen) Wahrscheinlichkeiten? Vergleiche die beiden Werte. Mache das Gleiche mit zehntausend zufälligen Würfeln. Vergleiche die beiden Ergebnisse und plotte alle Werte in geeigneten Balkendiagrammen.

# Kapitel 5

## Visualisierungen und statistische Tests

### 5.1 Visualisierungen

Die Linearen Kongruenz-Generatoren haben einen systematischen Fehler, also einen Fehler, der im Verfahren selbst begründet ist und sich nicht durch eine noch so gute Wahl eines Moduls  $m$ , eines Multiplikators  $a$  und einer Verschiebung  $c$  beheben lässt. Diesen Fehler wollen wir uns jetzt mal näher anschauen, und zwar machen wir das wie folgt:

Wir betrachten eine Folge von Zufallszahlen:

$$X_0, X_1, X_2 \dots, X_i \dots, X_{m-1}$$

Nun interpretieren wir je zwei aufeinander folgende Zahlen als Koordinaten eines Punktes der Ebene, also:

$$\begin{aligned} P_0 &= (X_0, X_1), \\ P_1 &= (X_1, X_2), \\ P_2 &= (X_2, X_3), \\ &\vdots \quad \quad \quad \vdots \\ P_{m-2} &= (X_{m-2}, X_{m-1}). \end{aligned}$$

Diese Punkte zeichnen wir nun in ein Koordinatensystem der Ebene ein. Die Punkte sollten nun im Idealfall, also wenn es sich um „wirkliche“ Zufallszahlen handeln würde, einerseits in jedem Gebiet mit ungefähr der gleichen Anzahl befinden (d.h. die Dichte der Punktwolke sollte überall ungefähr gleich groß sein), andererseits sollten sich keine Muster erkennen lassen. Muster sind grundsätzlich schlecht, weil sie bedeuten, dass aufeinander folgende Zufallszahlen nicht besonders „zufällig“ gezogen werden. Man kann sagen, dass eine Zufallszahl zu sehr durch ihre Vorgänger bestimmt ist. In Fällen „richtiger“ Zufallszahlen wäre das nicht so. Dann wäre die Erzeugung einer Zufallszahl völlig unabhängig von den Zufallszahlen, die vorher erzeugt wurden und es würden mit großer Wahrscheinlichkeit keine Muster entstehen. Natürlich kann man die Punkte auch erst in das Intervall  $[0, 1]$  transformieren (wie ging das nochmal?) und sich die folgenden Punktepaare anzeigen lassen

$$(U_i, U_{i+1}) \in [0, 1] \times [0, 1] = \{(x, y) : x \in [0, 1], y \in [0, 1]\}.$$

### **Gruppenaufgabe:**

Experimentiere selber mal ein bisschen mit diesen graphischen Darstellungen. Lasse dir für verschiedene Lineare Kongruenz-Generatoren die zweidimensionale Visualisierung im Excel-Programm anzeigen. Was stellst du fest? Siehst du irgendwelche Muster? Versuche es mal mit

$$a = 1229 \quad , \quad m = 2048 \quad , \quad c = 1 \quad \text{und} \quad X_0 = 0.$$

Was beobachtest du?

Bildet man nun analoge Punkte im Raum aus drei aufeinander folgenden Zufallszahlen, also:

$$\begin{aligned}
P_0 &= (X_0, X_1, X_2), \\
P_1 &= (X_1, X_2, X_3), \\
P_2 &= (X_2, X_3, X_4), \\
&\vdots \quad \vdots \quad \vdots \\
P_{m-3} &= (X_{m-3}, X_{m-2}, X_{m-1}),
\end{aligned}$$

so erhält man eine dreidimensionale Darstellung: eine Punktwolke in einem Würfel. Bei geringer Punktzahl wirken die Punktwolken noch zufällig, bei größerer Punktzahl zeigen sich schnell Muster.

Beachtet bitte: Diese Muster treten bei jedem Linearen Kongruenz-Generator auf, man bekommt diese Schwäche nicht weg, egal wie man den Modul  $m$ , den Multiplikator  $a$ , die Verschiebung  $c$  und den Startwert  $X_0$  auch immer wählt. Es ist ein systematischer Fehler, der durch die Art entsteht, wie man mit dem Linearen Kongruenz-Generator Zufallszahlen erzeugt. Hier sind wir bei den Grenzen des Linearen Kongruenz-Generators angekommen! Auf Grund dieser Schwäche hat man mittlerweile bessere Zufallszahlengeneratoren entwickelt, auf die wir in diesem Kurs aber nicht näher eingehen. Wir werden aber im nächsten Kapitel ein anderes Verfahren kennenlernen, mit dem man die Musterbildung zumindestens einschränkt und die Qualität der zufälligen Zahlenfolge verbessert.

Die Frage ist jetzt: Kann man diese Eindrücke, die man von den Graphiken her bekommt, auch mathematisch nachweisen? Wie kann man objektiv testen, ob ein Zufallsgenerator „gut“ ist? Damit beschäftigen wir uns im nächsten Abschnitt.

## 5.2 Der $\chi^2$ -Test

Unser Hauptziel ist es – daran sei noch einmal erinnert – Folgen von Zahlen zu erzeugen, die sich so verhalten als seien sie zufällig. Ein Kriterium ha-

ben wir bei den Linearen Kongruenz-Generatoren bereits kennengelernt: Es tritt dort mit Sicherheit eine Periode auf, also muss man wenigstens darauf achten, dass diese Periode möglichst lang ist. Das kann man erreichen, wenn man gewisse Regeln einhält, die wir kennengelernt haben. Die Länge der Periode ist zwar ein wichtiges Kriterium, aber: Eine lange Periode garantiert uns trotzdem noch nicht, dass wir auch eine für unsere jeweiligen Zwecke nützliche Zahlenfolge erhalten. Die Folge  $1, 2, 3, 4, \dots$  hat auch keine Periode, wird aber sicherlich nicht als zufällig angenommen. Im letzten Abschnitt haben wir uns auf unser Einschätzungsvermögen verlassen. Wir haben gesagt: Wenn wir uns die Zahlenfolgen zwei- und dreidimensional visualisieren und dann irgendwelche Muster erkennen, dann handelt es sich um eine schlechte zufällige Zahlenfolge. Aber: Zur Beurteilung von Zufälligkeit sollten wir uns nicht allzu sehr auf unseren gesunden Menschenverstand und unser Gefühl verlassen. Schöner wäre es doch, wir hätten unvoreingenommene, objektive Tests.

Es gibt ein großes Gebiet in der Mathematik, in der man solche Tests entwickelt: die **Statistik**. Wir werden uns jetzt einen speziellen Test, den sogenannten  $\chi^2$ -Test (sprich: „Chi-Quadrat-Test“) näher anschauen und ihn dann auf unser Problem anwenden.

Erst einmal etwas zum grundsätzlichen Wesen von statistischen Tests: Nehmen wir einmal an, wir hätten schon einige Tests durchgeführt und bei all diesen Tests hätte sich unsere zufällige Zahlenfolge als „gut“ erwiesen. Dann können wir trotzdem nicht sicher sein, dass die Folge auch bei jedem weiteren Test nicht miserabel abschneidet. Trotzdem gibt uns natürlich jeder weitere Test mehr Sicherheit bezüglich der Güte der zufälligen Zahlenfolge. In der Praxis werden ungefähr sechs verschiedene statistische Tests auf eine Folge angewendet. Wenn sie alle Tests besteht, dann betrachtet man sie als zufällig. Es gilt also die „Unschuldvermutung“: Solange wir nicht statistisch aufgezeigt haben (von einem Beweis kann man in diesem Zusammenhang nicht sprechen), dass eine Zahlenfolge „schlecht“ ist, gilt sie als „gut“. Wir möchten uns nun zwei dieser Tests näher anschauen. Beide Tests beruhen auf dem  $\chi^2$ -Test, den wir uns daher zunächst einmal getrennt anschauen.

Der  $\chi^2$ -Test ist der bekannteste statistische Test. Wir möchten ihn an einem einfachen Beispiel näher erläutern:

Wir betrachten das Werfen von zwei Würfeln und interessieren uns für die Augensumme  $s$  der beiden Würfel. Neben den möglichen Ausgängen des Ex-

perimentes führen wir die Wahrscheinlichkeiten auf:

Ausgang $s$	W'keit $p_s$
2	$\frac{1}{36}$
3	$\frac{2}{36}$
4	$\frac{3}{36}$
5	$\frac{4}{36}$
6	$\frac{5}{36}$
7	$\frac{6}{36}$
8	$\frac{5}{36}$
9	$\frac{4}{36}$
10	$\frac{3}{36}$
11	$\frac{2}{36}$
12	$\frac{1}{36}$

Zur Erläuterung: Insgesamt sind bei zwei Würfeln  $6 \cdot 6 = 36$  Ausgänge möglich:  $(1, 1), (1, 2), \dots, (6, 6)$ . Wenn man auf die Augenzahl 4 kommen möchte, braucht man die folgenden drei Ausgänge:  $(1, 3), (2, 2), (3, 1)$ . Daher kommen wir auf die Wahrscheinlichkeit von

$$p = \frac{\text{Anzahl der günstigen Fälle}}{\text{Anzahl der möglichen Fälle}} = \frac{3}{36}$$

für das Ereignis „Die Augensumme ist 4“.

Wir können daraus ableiten, wie oft wir die Augensumme 4 bei  $n$  Würfeln erwarten dürfen:

Bei 36 Würfeln sollte ungefähr 3-mal die Augensumme 4 vorkommen (beachte:  $3 = 36 \cdot \frac{3}{36}$ ).

Bei 72 Würfeln sollte ungefähr 6-mal die Augensumme 4 vorkommen (beachte:  $6 = 72 \cdot \frac{3}{36}$ ).

Bei 144 Würfeln sollte ungefähr 12-mal die Augensumme 4 vorkommen (beachte:  $12 = 144 \cdot \frac{3}{36}$ ).

Allgemein:

Bei  $n$  Würfeln sollte ungefähr

$$n \cdot \frac{3}{36} - \text{mal}$$

die Augensumme 4 vorkommen.

Noch allgemeiner:

Bei  $n$  Würfeln sollte jede Summe  $s$  ungefähr

$$n \cdot p_s - \text{mal}$$

vorkommen. Dies ist der sogenannte **Erwartungswert** für die Anzahl der Würfe mit Augensumme  $s$ .

Noch einmal zurück zu unserem konkreten Beispiel: Wenn wir 144 mal würfeln, sollte als Augensumme ungefähr  $144 \cdot \frac{3}{36}$ -mal, also 12-mal, die 4 erscheinen. Natürlich wird im Regelfall nicht genau 12-mal als Augensumme 4 herauskommen, aber im Durchschnitt schon. Wenn der Würfel fair, also in Ordnung ist, dann sollte was in der Nähe von 12 rauskommen.

Hier die Tabelle, wie oft bei  $n = 144$  Würfeln mit beiden Würfeln erwartungsgemäß jede Augensumme auftritt:

Ausgang $s$	Erwartungswert: $144 \cdot p_s$
2	4
3	8
4	12
5	16
6	20
7	24
8	20
9	16
10	12
11	8
12	4

Nehmen wir mal an, eine Person hat wirklich 144 mal gewürfelt. (Ich möchte euch das jetzt nicht zumuten.) Dann wird sie mit großer Wahrscheinlichkeit Häufigkeiten herausbekommen, die zwar nicht unbedingt genau den obigen Werten  $144 \cdot p_s$  entsprechen, aber irgendwo in der Nähe davon liegen. Nehmen wir einmal an, sie hat die folgenden Häufigkeiten des Auftretens der Augensummen beobachtet (ganz rechts, fett gedruckt):

Ausgang $s$	Erwartungswert: $144 \cdot p_s$	Wert $Y_s$ aus Experiment
2	4	<b>2</b>
3	8	<b>4</b>
4	12	<b>10</b>
5	16	<b>12</b>
6	20	<b>22</b>
7	24	<b>29</b>
8	20	<b>21</b>
9	16	<b>15</b>
10	12	<b>14</b>
11	8	<b>9</b>
12	4	<b>6</b>

Beachte bitte, dass die beobachtete Anzahl in keinem Fall exakt der erwarteten Anzahl entspricht. Ein Zufallsexperiment wird selten die exakt richtigen Häufigkeiten liefern!

In Anbetracht dessen stellt sich die Frage, ob und wie wir überhaupt testen können, ob die beiden Würfel fair sind, d.h. ob tatsächlich bei beiden Würfeln alle sechs Ziffern mit der gleichen Wahrscheinlichkeit und damit alle Augensummen mit den theoretisch oben ausgerechneten Wahrscheinlichkeiten auftreten.

**Wichtig: Wir können auf keinen Fall ein sicheres, endgültiges Urteil abgeben! Wir können anhand statistischer Tests nur sagen, dass *wahrscheinlich* etwas gilt oder nicht gilt. Wir können allerdings genau angeben, *wie groß* diese Wahrscheinlichkeit ist. Das heißt, wir können die *Fehlerwahrscheinlichkeit* angeben, also die Wahrscheinlichkeit, mit der wir uns geirrt haben.**

Naheliegender ist es, sich für alle möglichen Augensummen  $s$  die Abstände zwischen dem theoretisch zu erwartenden Wert  $144 \cdot p_s$  und dem im Experiment beobachteten Wert  $Y_s$  anzuschauen. Je „schlechter“ das Paar Würfel ist, desto größer sollten diese Differenzen (im Durchschnitt) sein. Damit die Differenzen immer positiv sind, bilden wir die Quadrate der Differenzen. Anschließend summieren wir alles auf, d.h. wir rechnen folgendes aus:

$$Q = (Y_2 - 144 \cdot p_2)^2 + (Y_3 - 144 \cdot p_3)^2 + \dots + (Y_{12} - 144 \cdot p_{12})^2$$

oder für allgemeines  $n$ :

$$Q = (Y_2 - n \cdot p_2)^2 + (Y_3 - n \cdot p_3)^2 + \dots + (Y_{12} - n \cdot p_{12})^2.$$

Ein schlechtes Paar Würfel sollte eher einen hohen Wert für  $Q$  ergeben, ein gutes Paar Würfel eher einen niedrigen Wert (weil dann die Beobachtungen in der Nähe der erwarteten Werte liegen und daher die Differenzen klein werden). Für jeden Wert von  $Q$  können wir uns jetzt fragen: „Wie wahrscheinlich ist es, dass  $Q$  so hoch ist, wenn echte Würfel verwendet werden?“ Für sehr hohe Werte von  $Q$  sollte diese Wahrscheinlichkeit klein sein. Wenn diese Wahrscheinlichkeit sehr klein ist, zum Beispiel  $\frac{1}{100}$ , dann wissen wir, dass nur in ungefähr einem unter 100 Fällen eine so deutliche Abweichung von den erwarteten Werten auftreten sollte und wir haben Grund misstrauisch zu werden. Aber: Es kann ja nun auch sein, dass wir gerade diesen einen

von 100 Fällen erwischt haben. Es kann also sein, dass wir zu Unrecht misstrauisch geworden sind und einen Fehler machen, wenn wir aufgrund dieses Misstrauens die Würfel für schlecht halten. Die Wahrscheinlichkeit, dass wir die Würfel zu Unrecht ablehnen, also die Fehlerwahrscheinlichkeit, ist gerade  $\frac{1}{100}$ , nämlich gleich der Wahrscheinlichkeit, dass wir mit unserer Beobachtung in dem „Ablehnungsbereich“ liegen.

Nun müssen wir aber noch eine Sache beachten: Nehmen wir mal an, wir würden ganz oft würfeln und wir würden erwarten eine bestimmte Augensumme tritt 10 000 mal auf. Dann führt man das Experiment durch und diese Augensumme tritt genau 9 999 mal auf. Was würdest du dann denken? Klar, ein sehr gutes Ergebnis! Man wird davon ausgehen, dass die Annahme, die man vorher gemacht hat (dass es sich um einen fairen Würfel handelt), richtig war. Dagegen wird man nicht so euphorisch sein, wenn man vorher erwartet hat, dass eine Augensumme 3 mal eintritt und dann im Experiment 2 mal beobachtet wird. In beiden Fällen ist aber der Abstand der beiden Werte gleich 1. In dem Ausdruck

$$Q = (Y_2 - n \cdot p_2)^2 + (Y_3 - n \cdot p_3)^2 + \dots + (Y_{12} - n \cdot p_{12})^2$$

würden also beide Terme den gleichen Beitrag zur Summe leisten. Das kann nicht sinnvoll sein! Eigentlich sind wir nämlich gar nicht an den absoluten Fehlern, sondern an den relativen Fehlern interessiert. Sprich: Wir empfinden nicht dann ein Ergebnis automatisch als schlecht, wenn die Differenz zwischen erwarteten und beobachtbaren Wert groß ist, sondern wenn der beobachtbare Wert **prozentual** weit von dem erwarteten Wert abweicht. Dies kann man dadurch berücksichtigen, dass man den Ausdruck

$$\chi^2 = \frac{(Y_2 - n \cdot p_2)^2}{n \cdot p_2} + \frac{(Y_3 - n \cdot p_3)^2}{n \cdot p_3} + \dots + \frac{(Y_{12} - n \cdot p_{12})^2}{n \cdot p_{12}}$$

betrachtet. Nun gilt: Wenn  $n \cdot p_2$  doppelt so groß ist wie  $n \cdot p_3$ , dann wird die quadrierte Differenz zwischen  $Y_2$  und  $n \cdot p_2$  nur halb so viel gewichtet wie die quadrierte Differenz zwischen  $Y_3$  und  $n \cdot p_3$ . Also: Wenn der erwartete Wert größer wird, dann werden eventuelle absolute Abweichungen zum beobachteten Wert weniger stark gewichtet.

Man nennt

$$\chi^2 = \frac{(Y_2 - n \cdot p_2)^2}{n \cdot p_2} + \frac{(Y_3 - n \cdot p_3)^2}{n \cdot p_3} + \dots + \frac{(Y_{12} - n \cdot p_{12})^2}{n \cdot p_{12}}$$

die  $\chi^2$ -Statistik der Beobachtungen  $Y_2, \dots, Y_{12}$ . Nun kann man näherungsweise bestimmen, mit welcher Wahrscheinlichkeit  $\chi^2$  bestimmte Werte annimmt. Für sehr große Werte von  $n$  sind diese angenäherten Wahrscheinlichkeiten ziemlich gut. Diese Wahrscheinlichkeiten werden in Tabellen gelistet. Für unseren Fall mit den Augensummen ergibt sich die folgende (auszugsweise abgedruckte) Tabelle:

Wert $x$	Wahrscheinlichkeit: $P(\chi^2 \leq x)$
2.558	0.01
3.940	0.05
6.737	0.25
9.342	0.5
12.55	0.75
18.31	0.95
23.21	0.99

Wie ist diese Tabelle zu lesen? Links steht ein Wert und rechts die Wahrscheinlichkeit dafür, dass  $\chi^2$  **höchstens** diesen Wert annimmt. Die Wahrscheinlichkeit, dass  $\chi^2$  also höchstens gleich 18.31 ist, ist gleich  $P(\chi^2 \leq 18.31) = 0.95$ , also gleich 95 Prozent. Wenn wir also bei dem obigen Würfelversuch für  $\chi^2$  einen Wert beobachten, der größer als 18.31 ist, etwa gleich 19, dann ist dies ein sehr unwahrscheinliches Ereignis. Denn die Wahrscheinlichkeit, dass  $\chi^2$  höchstens gleich 18.31 ist, war ja gleich 0.95, also 95 Prozent.

Also muss ja umgekehrt die Wahrscheinlichkeit, dass  $\chi^2$  größer als 18.31 ist, gleich  $1 - 0.95 = 0.05$  sein, also gleich 5 Prozent. Wenn wir also beobachten, dass  $\chi^2$  gleich 19 ist, dann ist dies, vorausgesetzt wir haben einen fairen Würfel, nur mit weniger als 5%-iger Wahrscheinlichkeit der Fall. Da wir diesen unwahrscheinlichen Fall beobachtet haben, lässt uns das misstrauisch werden. Es scheint so, also ob unsere Vermutung falsch und der Würfel nicht fair war. Aber es kann ja nun mal sein, dass dieses unwahrscheinliche Ereignis eingetreten ist. Wir können mit unserem Misstrauen also auch falsch liegen. Die Wahrscheinlichkeit, dass wir damit falsch liegen, beträgt höchstens 5 Prozent. Wenn wir sehr oft würfeln, viel mehr als 144 mal und dann am Ende ein so hoher Wert von  $\chi^2$  rauskommt, dass die Wahrscheinlichkeit für einen so hohen Wert unter der Annahme fairer Würfel nur 5 Prozent beträgt, dann werden wir misstrauisch. Wir glauben dann zunächst einmal nicht mehr an unsere Vermutung, dass es sich um einen fairen Würfel handelt. Um etwas sicherer zu werden, wiederholen wir den Test vielleicht häufiger.

Wir rechnen den Wert von  $\chi^2$  für zwei Beispiele aus. Wir nehmen dabei an, dass wir zweimal jeweils 144 mal zwei Paar Würfel werfen, also erst 144 mal das erste Würfelpaar und dann 144 mal das zweite Würfelpaar. Dann erhalten wir für beide Versuche verschiedene beobachtete Werte, die auf der nächsten Seite aufgelistet sind. Wir bilden mal von beiden Größen die  $\chi^2$ -Statistiken und erhalten:

$$\begin{aligned}\chi_1^2 &= \frac{(4-4)^2}{4} + \frac{(10-8)^2}{8} + \frac{(10-12)^2}{12} + \frac{(13-16)^2}{16} + \frac{(20-20)^2}{20} + \frac{(18-24)^2}{24} \\ &\quad + \frac{(18-20)^2}{20} + \frac{(11-16)^2}{16} + \frac{(13-12)^2}{12} + \frac{(14-8)^2}{8} + \frac{(13-4)^2}{4} \\ &= 29 \frac{59}{120},\end{aligned}$$

$$\begin{aligned}\chi_2^2 &= \frac{(3-4)^2}{4} + \frac{(7-8)^2}{8} + \frac{(11-12)^2}{12} + \frac{(15-16)^2}{16} + \frac{(19-20)^2}{20} + \frac{(24-24)^2}{24} \\ &\quad + \frac{(21-20)^2}{20} + \frac{(17-16)^2}{16} + \frac{(13-12)^2}{12} + \frac{(9-8)^2}{8} + \frac{(5-4)^2}{4} \\ &= 1 \frac{17}{120}.\end{aligned}$$

Ausgang $s$	Erwartungswert: $144 \cdot p_s$	Wert $Y_{s_1}$ aus Experiment 1	Wert $Y_{s_2}$ aus Experiment 2
2	4	4	3
3	8	10	7
4	12	10	11
5	16	13	15
6	20	20	19
7	24	18	24
8	20	18	21
9	16	11	17
10	12	13	13
11	8	14	9
12	4	13	5

Der Wert von  $\chi_1^2$  (zur Erinnerung:  $\chi_1^2 = 29 \frac{59}{120}$ ) ist sehr groß. Unter der Annahme, dass faire Würfel vorliegen, kommt ein solch hoher Wert – wie man der obigen Tabelle entnimmt – in weniger als einem Prozent aller Fälle vor. Mit mindestens 99 Prozent Wahrscheinlichkeit ist der Wert von  $\chi_1^2$  kleiner. Dies lässt uns misstrauisch werden. Wir können jetzt davon ausgehen, dass es sich nicht um faire Würfel handelt. Dann liegen wir mit einer Wahrscheinlichkeit von mindestens 99 Prozent richtig mit unserer Vermutung, es verbleibt höchstens ein Prozent Fehlerwahrscheinlichkeit.

Der Wert von  $\chi_2^2$  (zur Erinnerung:  $\chi_2^2 = 1 \frac{17}{120}$ ) liegt nicht im Ablehnungsbe-

reich. Ein Wert, der so klein (oder kleiner) ist, tritt zwar nur in weniger als 1 Prozent der Fälle auf, spricht jedoch nicht gegen die von uns zu überprüfende Hypothese, dass es sich um faire Würfel handelt. Daher dürfen wir unsere Annahme, dass es sich um faire Würfel handelt, aufgrund dieses Ergebnisses nicht ablehnen. Dies heißt jedoch nicht, dass wir damit in irgendeiner Form bewiesen hätten, dass es sich um faire Würfel handelt!

Schon mal eine Bemerkung in Vorgriff auf unsere statistischen Tests für die zufälligen Zahlenfolgen: Dort werden wir auch bei zu niedrigen Werten von  $\chi^2$  (wie es in diesem Beispiel der Fall wäre) misstrauisch. Dies hängt damit zusammen, dass wir nicht nur die „gleichmäßige Verteilung“ der Zahlen (die hier der Fairness der Würfel entspricht) überprüfen, sondern auch, wie „zufällig“ die Zahlen erscheinen. Dies ist aber eine Besonderheit bei der Überprüfung von zufälligen Zahlenfolgen. Beim eigentlichen  $\chi^2$ -Test wird man nur bei auffallend hohen Werten der Größe  $\chi^2$  misstrauisch.

### **Gruppenaufgabe:**

Führe den Test jetzt selber einmal mit 72 Würfeln durch. Erstelle eine Tabelle, die die Erwartungswerte und die tatsächlich erwürfelten Anzahlen für alle möglichen Augensummen enthält.

Berechne nun die  $\chi^2$ -Statistik und schaue in der Tabelle nach. Sind die Würfel (wahrscheinlich) fair?

Damit, so hoffen wir, ist euch das Prinzip des  $\chi^2$ -Test klar geworden. Wir möchten es nun auf unsere Zufallsgeneratoren anwenden, d.h. wir möchten unsere Zufallsgeneratoren einigen statistischen Tests unterwerfen, die alle auf dem  $\chi^2$ -Test beruhen.

## 5.3 Statistische Tests für zufällige Zahlenfolgen

Wir stellen Tests vor, mit denen man die von Zufallszahlengeneratoren erzeugten Zahlenfolgen auf ihre Güte hin testen kann. Dabei ist der Baustein immer der  $\chi^2$ -Test.

### 5.3.1 Eindimensionaler Test

Theoretische, also „richtige“ Zufallszahlen haben die Eigenschaft, dass die Wahrscheinlichkeit für das Eintreffen jeder Zahl gleich groß ist. Daher ist die erste Anforderung an eine Zufallsvariable, dass die Zahlen näherungsweise gleichverteilt sind. Das bedeutet für eine Folge endlich vieler natürlicher Zufallszahlen: Jede Zahl sollte ungefähr mit der gleichen Häufigkeit auftreten. Für einen Linearen Kongruenz-Generator mit maximaler Periodenlänge  $m$  ist das natürlich automatisch erfüllt, wenn die Anzahl der erzeugten Zufallszahlen ein Vielfaches von  $m$  ist, da dann jede Zahl genau gleich oft vorkommt. In diesem Fall macht ein reiner Test auf „gleichmäßige Verteilung“ keinen Sinn. Für andere Zufallszahlengeneratoren könnte ein solcher Test aber sinnvoll sein. Wir testen hier aber nicht nur auf eine „gleichmäßige Verteilung“ der Zahlen, sondern möchten auch dann eine zufällige Zahlenfolge ablehnen, wenn sie „zu regelmäßig“ (wie beim Linearen Kongruenz-Generator) und damit „zu wenig zufällig“ erscheint. Daher möchten wir den eindimensionalen Test kurz beschreiben:

Die Idee ist die folgende: Der Test wird auf eine Folge  $(U_n)$  angewendet, die nur Zahlen zwischen 0 und 1 annimmt, von der wir überprüfen wollen, ob die Zahlen gleichmäßig auf dem Intervall  $[0, 1)$  verteilt sind und auch „zufällig erscheinen“. „Gleichmäßig verteilt“ bedeutet mathematisch: Die Wahrscheinlichkeit, dass eine beliebig gewählte Zahl kleiner gleich  $x$  ist, sollte gerade gleich  $x$  sein. Beispiel: Die Wahrscheinlichkeit, dass eine Zahl kleiner gleich 0.25 ist, sollte gerade gleich 0.25 sein. Jede einzelne Zahl tritt übrigens mit Wahrscheinlichkeit 0 ein. Dies ist ein Phänomen, mit dem wir uns in einem späteren Kurs noch beschäftigen werden. Also: Die Wahrscheinlichkeit, dass eine Zahl in einem gewissen Teilstück (zum Beispiel  $[0.25; 0.75]$ ) liegt, ist gerade so groß wie die Länge dieses Teilstücks (in diesem Beispiel: 0.5).

Wie können wir nun überprüfen, ob unsere Folge von erzeugten Zufallszahlen annähernd gleichmäßig auf dem Intervall  $[0, 1)$  verteilt ist? Nun: Wir zerlegen das Intervall  $[0, 1)$  in  $m$  gleich große Teilintervalle. Nehmen wir mal an, wir wählen  $m = 100$ . Dann zerlegen wir das Intervall  $[0, 1)$  also in die folgenden Teilintervalle:

$$[0; 0.01), [0.01; 0.02), [0.02; 0.03), \dots, [0.99, 1.00).$$

Wie groß ist dann die Wahrscheinlichkeit, in einem bestimmten Teilintervall zu liegen? Da wir  $m$  Teilintervalle haben, ist die Wahrscheinlichkeit gerade  $p = \frac{1}{m}$ . (In unserem obigen Beispiel ist  $p = \frac{1}{100}$ .)

### Gruppenaufgabe:

Nehmen wir einmal an, wir haben das Intervall in  $m$  Teilintervalle zerlegt und wir haben  $n$  Zufallszahlen erzeugt. Wie viele Zufallszahlen sollten dann erwartungsgemäß in jedem Teilintervall liegen?

Nun können wir einen  $\chi^2$ -Test anwenden: Wir zählen für jedes Teilintervall  $s = 0, \dots, m-1$  die Anzahl  $Y_s$  der Zahlen, die in diesem Teilintervall liegen. Andererseits wissen wir, dass in jedem der Teilintervalle erwartungsgemäß  $\frac{n}{m}$  Zufallszahlen liegen sollten. Daher betrachten wir die Größe:

$$\chi^2 = \frac{(Y_0 - \frac{n}{m})^2}{\frac{n}{m}} + \frac{(Y_1 - \frac{n}{m})^2}{\frac{n}{m}} + \dots + \frac{(Y_{m-1} - \frac{n}{m})^2}{\frac{n}{m}}.$$

Nun rechnen wir den Wert von  $\chi^2$  aus. Wir nehmen zunächst an, es würde sich um echte Zufallszahlen handeln. Dann kann man ziemlich genau ausrechnen, mit welcher Wahrscheinlichkeit  $\chi^2$  welchen Wert annimmt. Ein zu hoher Wert von  $\chi^2$  (dann erscheinen die Zahlen nicht gleichmäßig verteilt) oder ein zu niedriger Wert von  $\chi^2$  (dann erscheinen die Zahlen nicht zufällig) lässt uns misstrauisch werden und uns an der Güte unserer zufälligen Zahlenfolge (und damit an der Güte unseres eingesetzten Zufallszahlengenerators) zweifeln. Man sollte dabei als Faustregel beachten, dass  $\frac{n}{m} \geq 5$  gilt, sonst ist der  $\chi^2$ -Test zu ungenau. Also: Man muss gegebenenfalls, wenn dies nicht erfüllt ist, entweder die Anzahl der zu untersuchenden Zahlen erhöhen oder die Intervallbreite verringern. Genauer gehen wir wie bei dem eindimensionalen Test wie folgt vor:

Wenn der Wert von  $\chi^2$  so groß (oder klein) ist, dass die Wahrscheinlichkeit für einen so großen (oder so kleinen) Wert von  $\chi^2$  bei richtigen Zufallszahlen kleiner als 1% ist, dann lehnen wir unsere Zahlenfolge sofort als nicht ausreichend zufällig ab.

Ist der Wert von  $\chi^2$  so groß (oder klein), dass die Wahrscheinlichkeit für einen größeren (oder kleineren) Wert von  $\chi^2$  bei richtigen Zufallszahlen größer gleich 1% und kleiner als 5% ist, dann sind die Zahlen „verdächtig“. Sollten sie auch bei anderen Tests „verdächtig“ sein, dann werden wir sie auch ablehnen.

Es handelt sich also **nicht um einen reinen Gleichverteilungstest**. Dann dürften wir die zufällige Zahlenfolge nur bei sehr hohen Werten von  $\chi^2$  verwerfen. Stattdessen wird **zusätzlich** überprüft, wie „zufällig“ (im Sinne von „nicht zu gleichmäßig“) die Zahlen erscheinen.

Wir wollen das kurz zusammenfassen:

Ist  $\chi_p^2$  der Wert, für den

$$P(\chi^2 \leq \chi_p^2) = p$$

gilt, so ist

$$[0; \chi_{0.01}^2) \cup (\chi_{0.99}^2; +\infty)$$

der **Ablehnungsbereich** und

$$[\chi_{0.01}^2, \chi_{0.05}^2) \cup (\chi_{0.95}^2, \chi_{0.99}^2]$$

der **verdächtige Bereich**.

Dies soll noch einmal an einem Beispiel demonstriert werden:

Nehmen wir einmal an, wir haben das Intervall in 100 gleich große Teilintervalle zerlegt. Dann hätten wir die folgende Tabelle :

Wert $\chi_p^2$	Wahrscheinlichkeit: $p = P(\chi^2 \leq \chi_p^2)$
69.230	0.01
77.046	0.05
89.181	0.25
98.334	0.5
108.093	0.75
123.225	0.95
134.642	0.99

Man kann nun sagen:

- Wenn  $\chi^2$  kleiner als 69.230 oder größer als 134.642 ist, dann lehnen wir unsere erzeugte zufällige Zahlenfolge ab.
- Wenn  $\chi^2$  mindestens gleich 69.230 und zugleich kleiner als 77.046 oder aber größer als 123.225 und zugleich höchstens so groß wie 134.642 ist, dann gilt unsere Zahlenfolge als verdächtig.

Also:

$\chi^2 < 69.230$	$\Rightarrow$	<b>Zahlenfolge ablehnen!</b>
$69.230 \leq \chi^2 < 77.046$	$\Rightarrow$	Zahlenfolge verdächtig!
$123.225 < \chi^2 \leq 134.642$	$\Rightarrow$	Zahlenfolge verdächtig!
$134.642 < \chi^2$	$\Rightarrow$	<b>Zahlenfolge ablehnen!</b>

Liegt  $\chi^2$  irgendwo dazwischen, so können wir keine Aussagen machen. Wir können dann die Zahlenfolge jedenfalls nicht ablehnen und auch nicht für verdächtig halten. Das heißt aber noch lange nicht, dass es sich um eine „gute“ Zahlenfolge handelt. Es kann ja auch sein, dass sie nur zufällig im erlaubten Bereich liegt. Auch bei dieser Beurteilung können wir also einen (zum Teil heftigen) Fehler machen, aber das Risiko müssen wir in Kauf nehmen. Das Motto lautet: „In dubio pro reo!“\* Solange eine zufällige Zahlenfolge mit statistischen Tests nicht als schlecht nachgewiesen ist, gilt sie auch nicht als schlecht. Wenn sie alle statistischen Tests, auf die man sich vorher geeinigt hat, bestanden hat, dann gilt sie als hinreichend gut.

*\*lat.  
Im Zweifel  
für den  
Angeklagten!*

### Gruppenaufgabe:

Schaue dir das Excel-Arbeitsblatt zum eindimensionalen Test für zufällige Zahlenfolgen an. Versuche zu verstehen, was in dem Excel-Arbeitsblatt genau gemacht wird. Hierbei wird das Intervall  $[0, 1)$  in 100 Teilintervalle zerlegt:

$$[0; 0.01), [0.01; 0.02), [0.02; 0.03), \dots, [0.99, 1.00).$$

Führe den eindimensionalen Test nun für die von Excel selbst erzeugten Zufallszahlen durch. Die entsprechende  $\chi^2$ -Tabelle ist die bereits oben genannte und wird im direkten Anschluss an diese Aufgabe erneut angegeben.

Wert $\chi_p^2$	Wahrscheinlichkeit: $p = P(\chi^2 \leq \chi_p^2)$
69.230	0.01
77.046	0.05
89.181	0.25
98.334	0.5
108.093	0.75
123.225	0.95
134.642	0.99

### 5.3.2 Serientest

Wir wollen nicht nur, dass die einzelnen Zahlen gleichmäßig verteilt sind, sondern auch, dass Paare aufeinander folgender Zahlen gleichmäßig verteilt sind. Dadurch wollen wir insbesondere Musterbildungen in der Ebene, die wir bei der Visualisierung erkannt haben, mit mathematischen Methoden aufdecken. Bei Linearen Kongruenz-Generatoren treten ja immer Scharen paralleler Geraden auf. Wenn der Abstand zwischen den Geraden nun groß genug ist, sollte dies durch ein schlechtes Testergebnis aufgedeckt werden.

Was machen wir nun? Wir zerlegen das Einheitsquadrat

$$[0, 1) \times [0, 1) = \{ (x, y) : x \in [0, 1), y \in [0, 1) \}$$

in  $m^2$  Teilquadrate:

$$\begin{array}{ccccccc}
[0, \frac{1}{m}) \times [0, \frac{1}{m}) & , & [0, \frac{1}{m}) \times [\frac{1}{m}, \frac{2}{m}) & , & \dots & , & [0, \frac{1}{m}) \times [\frac{m-1}{m}, 1) & , \\
[\frac{1}{m}, \frac{2}{m}) \times [0, \frac{1}{m}) & , & [\frac{1}{m}, \frac{2}{m}) \times [\frac{1}{m}, \frac{2}{m}) & , & \dots & , & [\frac{1}{m}, \frac{2}{m}) \times [\frac{m-1}{m}, 1) & , \\
\vdots & , & \vdots & , & \dots & , & \vdots & , \\
[\frac{m-1}{m}, 1] \times [0, \frac{1}{m}) & , & [\frac{m-1}{m}, 1] \times [\frac{1}{m}, \frac{2}{m}) & , & \dots & , & [\frac{m-1}{m}, 1] \times [\frac{m-1}{m}, 1) & .
\end{array}$$

Wie groß ist die Wahrscheinlichkeit für ein Paar  $(x, y) \in [0, 1) \times [0, 1)$  von Zufallszahlen, in einem bestimmten Teilquadrat zu liegen? Da wir  $m^2$  Teilquadrate haben, ist die Wahrscheinlichkeit gerade  $p = \frac{1}{m^2}$ .

Wir erzeugen mit unserem Zufallsgenerator nun  $2 \cdot n$  Zufallszahlen  $Y_0, Y_1, \dots, Y_{2n-2}, Y_{2n-1}$  und transformieren diese in  $2n$  Zufallszahlen  $U_0, U_1, \dots, U_{2n-2}, U_{2n-1}$  aus  $[0, 1)$ . Nun bilden wir  $n$  Paare:

$$\begin{array}{c}
(U_0, U_1) \\
(U_2, U_3) \\
\vdots \\
(U_{2n-2}, U_{2n-1}).
\end{array}$$

(Also: Wir bilden nicht die Paare  $(U_0, U_1), (U_1, U_2), \dots, (U_{2n-2}, U_{2n-1})$  wie bei der Visualisierung! Dies hängt mit den Anforderungen an den  $\chi^2$ -Test zusammen, nämlich, dass keine Abhängigkeit zwischen den zufälligen Ziehungen bestehen darf.)

### Gruppenaufgabe:

Nehmen wir einmal an, wir haben das Quadrat in  $m^2 = 100 \cdot 100 = 10\,000$  Teilquadrate zerlegt und wir haben  $2n = 200\,000$  Zufallszahlen erzeugt. Wie viele Paare von Zufallszahlen sollten dann erwartungsgemäß in jedem Teilquadrat liegen?

Nun können wir wieder einen  $\chi^2$ -Test anwenden. Wir zählen für jedes Teilquadrat  $s = 0, \dots, m^2 - 1$  die Anzahl  $Y_s$  der Zahlen, die in diesem Teilintervall liegen. Andererseits wissen wir, dass in jedem der Teilintervalle erwartungsgemäß  $\frac{n}{m^2}$  Zufallszahlen liegen sollten. Daher betrachten wir die Größe:

$$\chi^2 = \frac{(Y_0 - \frac{n}{m^2})^2}{\frac{n}{m^2}} + \frac{(Y_1 - \frac{n}{m^2})^2}{\frac{n}{m^2}} + \dots + \frac{(Y_{m^2-1} - \frac{n}{m^2})^2}{\frac{n}{m^2}}.$$

Nun rechnen wir den Wert von  $\chi^2$  aus. Wir nehmen zunächst an, es würde sich um echte Zufallszahlen handeln. Dann kann man ziemlich genau ausrechnen, mit welcher Wahrscheinlichkeit  $\chi^2$  welchen Wert annimmt. Ein zu hoher Wert von  $\chi^2$  (dann erscheinen die Zahlenpaare nicht gleichmäßig verteilt) oder ein zu niedriger Wert von  $\chi^2$  (dann erscheinen die Zahlenpaare nicht zufällig) lässt uns misstrauisch werden und uns an der Güte unserer zufälligen Zahlenfolge (und damit an der Güte unseres eingesetzten Zufallszahlengenerators) zweifeln. Man sollte dabei als Faustregel beachten, dass  $\frac{n}{m^2} \geq 5$  gilt, sonst ist der  $\chi^2$ -Test zu ungenau. Also: Man muss gegebenenfalls, wenn dies nicht erfüllt ist, entweder die Anzahl der zu untersuchenden Zahlen erhöhen oder die Intervallbreite erniedrigen.

Wie beim eindimensionalen Test treffen wir auch hier die folgende Vereinbarung:

Ist  $\chi_p^2$  der Wert, für den  $P(\chi^2 \leq \chi_p^2) = p$  gilt, so ist

$$[0; \chi_{0.01}^2) \cup (\chi_{0.99}^2; +\infty)$$

der **Ablehnungsbereich** und

$$[\chi_{0.01}^2, \chi_{0.05}^2) \cup (\chi_{0.95}^2, \chi_{0.99}^2]$$

der **verdächtige Bereich**.

Beim Serientest ändert sich im Vergleich zum eindimensionalen Test nur die  $\chi^2$ -Tabelle (da wir mehr Unterteilungen haben, nämlich  $m^2$  statt vorher  $m$ ). Zum Beispiel haben wir für  $m = 100$  (also bei der Zerlegung des Quadrates in  $100 \cdot 100 = 10\,000$  Einzelquadrate):

Wert $\chi_p^2$	Wahrscheinlichkeit: $p = P(\chi^2 \leq \chi_p^2)$
9672.965	0.01
9767.537	0.05
9903.258	0.25
9998.333	0.5
10094.016	0.75
10232.737	0.95
10330.917	0.99

Man kann nun sagen:

- Wenn  $\chi^2$  kleiner als 9672.965 oder größer als 10330.917 ist, dann lehnen wir unsere erzeugte zufällige Zahlenfolge ab.
- Wenn  $\chi^2$  mindestens gleich 9672.965 und zugleich kleiner als 9767.537 oder aber größer als 10232.737 und zugleich höchstens so groß wie 10330.917 ist, dann gilt unsere Zahlenfolge als verdächtig.

Also:

$\chi^2 < 9672.965$	$\Rightarrow$	<b>Zahlenfolge ablehnen!</b>
$9672.965 \leq \chi^2 < 9767.537$	$\Rightarrow$	Zahlenfolge verdächtig!
$10343.737 < \chi^2 \leq 10330.917$	$\Rightarrow$	Zahlenfolge verdächtig!
$10330.917 < \chi^2$	$\Rightarrow$	<b>Zahlenfolge ablehnen!</b>

Liegt  $\chi^2$  irgendwo dazwischen, so können wir keine Aussagen machen. Ansonsten gelten die gleichen Aussagen, die wir schon beim eindimensionalen Test gemacht haben.

### Gruppenaufgabe:

Schaue dir das Excel-Arbeitsblatt zum Serientest für zufällige Zahlenfolgen an. Versuche zu verstehen, was in dem Excel-Arbeitsblatt genau gemacht wird. Hierbei wird das Quadrat  $[0, 1) \times [0, 1)$  in  $100 \cdot 100 = 10\,000$  Teilquadrate zerlegt.

Führe den Serientest nun für die von deinem eigenen Linearen Kongruenz-Generator erzeugten Zufallszahlen und für die von Matlab, C und Excel erzeugten Zufallszahlen durch.

Selbstverständlich kann man sich nun auch andere Serientests überlegen, z.B. indem man drei aufeinander folgende Zufallszahlen zu einem Punkt des Raumes zusammenfasst und den Würfel  $[0, 1) \times [0, 1) \times [0, 1)$  in  $m^3$  Teilwürfel unterteilt. Außerdem kann man sich noch viele andere statistische Tests für zufällige Zahlenfolgen überlegen, zum Beispiel einen **Permutationstest**, was wie der Gleichverteilungs- und der Serientest ein spezieller  $\chi^2$ -Test ist. Darauf wollen wir aber an dieser Stelle nicht näher eingehen. Stattdessen lernen wir jetzt zum Abschluss des Kurses noch eine Methode kennen, wie man beim Einsatz eines Linearen Kongruenz-Generators die zweidimensionale Musterbildung, die wir sowohl graphisch erkannt als auch statistisch nachgewiesen haben, ein wenig besser in den Griff bekommt.

# Kapitel 6

## Kopplung von Zufallszahlengeneratoren

Es gibt viele Möglichkeiten die typischen Schwächen des Linearen Kongruenz-Generators, die wir ausführlich kennengelernt haben, zu verringern. Eine Möglichkeit möchten wir euch kurz vorstellen: Wir koppeln einfach zwei Zufallszahlengeneratoren!

Was verstehen wir darunter? Wir wollen es mal an einem einfachen Beispiel erklären:

Wir betrachten die beiden Linearen Kongruenz-Generatoren

$$\begin{aligned} (1) \quad X_{i+1} &= \text{Rest}_7(50 \cdot X_i + 9), & X_0 &= 5, \\ (2) \quad X_{i+1} &= \text{Rest}_7(50 \cdot X_i + 9), & X_0 &= 6, \end{aligned}$$

also zweimal den gleichen Linearen Kongruenz-Generator, nur mit zwei verschiedenen Startwerten.

Die erste Folge hat (bis zum Eintreten der Periode) die Gestalt:

$$5, 0, 2, 4, 6, 1, 3,$$

die zweite Folge sieht wie folgt aus:

$$6, 1, 3, 5, 0, 2, 4.$$

Wir interpretieren die beiden Folgen nun wie folgt: Die erste Folge liefert unsere Zufallszahlen, mit der zweiten Folge wird die Position der Zahl, die aus der ersten Folge gewählt wird. Da es keine 0-te Position gibt, addieren wir zu allen Zahlen der zweiten Folge eine 1:

Die erste Folge hat dann nach wie vor die Gestalt:

5, 0, 2, 4, 6, 1, 3,

die zweite Folge sieht jetzt folgt aus:

7, 2, 4, 6, 1, 3, 5.

Also: In der zweiten Folge steht zuerst eine 7. Das heißt, wir wählen zunächst die siebte Zahl der ersten Folge, also die 3.

Dann steht in der zweiten Folge eine 2. Das heißt, wir wählen nun die zweite Zahl der ersten Folge, also die 0.

Nun folgt in der zweiten Folge eine 4. Wir wählen also die vierte Zahl der ersten Folge, also die 4.

Und so weiter...

### **Gruppenaufgabe:**

Wie lautet also jetzt die vollständige Folge, die wir auf diese Art durch Kopplung der beiden Zufallszahlenfolgen erhalten?

Anhand einer so kurzen Folge wie in der letzten Aufgabe kann man den Vorteil, den eine solche Kopplung möglicherweise bringt, natürlich noch nicht sehen.

Interessant wird dies erst bei längeren Zahlenfolgen. Verschwinden dann vielleicht die für einen Linearen Kongruenz-Generator typischen Linienmuster in der zweidimensionalen Darstellung?

Probiere es doch einfach selber mal aus!

### Gruppenaufgabe:

Weiter oben haben wir einen Linearen Kongruenz-Generator mit

$$a = 1229 \quad , \quad m = 2048 \quad , \quad c = 1 \quad \text{und} \quad x_0 = 0$$

betrachtet, bei dem in der zweidimensionalen Darstellung nur wenige, relativ weit voneinander entfernte Linien zu sehen waren. Lass dir die zweidimensionale Visualisierung ruhig noch einmal anzeigen. Sieht das nicht furchtbar aus? Schau dir nun an, was passiert, wenn man diesen Zufallszahlengenerator mit einem anderen Generator koppelt.

Man sieht an dem Ergebnis, und zwar besonders deutlich bei der Kopplung von schlechten Linearen Kongruenz-Generatoren, dass die typischen (eigentlich unerwünschten) Muster des Linearen Kongruenz-Generators in der zweidimensionalen Darstellung aufgelöst werden.

Man kann sich nun auch andere Kopplungen von Zufallszahlengeneratoren und weitere Verfahren zur Qualitätsverbesserung der Zahlenfolgen überlegen. Mittlerweile gibt es, wie bereits angesprochen, zudem bessere (nicht-lineare) Zufallszahlengeneratoren. Wir wollen es aber an dieser Stelle bei diesem einfachen Verfahren bewenden lassen.

# Schlussworte und Ausblick

Ich hoffe, dass euch dieser Kurs einen ersten Einblick in die Technik der stochastischen Simulation und die Theorie der statistischen Tests vermittelt hat. Wir haben dabei auch ein wenig in die Historie und hinter die Kulissen der Zufallszahlengeneratoren geschaut. Ihr habt gelernt, dass man sich bei der Beurteilung vielleicht nicht nur auf sein Gefühl verlassen sollte, sondern dass es für eine Bewertung der Güte einer (pseudo-)zufälligen Zahlenfolge ganz objektive mathematische Tests gibt. Zwei davon haben wir kennengelernt. Man muss solche Testergebnisse dann aber auch geeignet interpretieren, sonst sind sie wertlos.

Jetzt ist euch vielleicht klar, dass es gar nicht so einfach ist, mit einem Computer den Zufall gut zu simulieren. Man muss schon so allerhand beachten, sonst kann man in die verschiedensten Fallen tappen.

In weiteren SimuLab-Kursen wollen wir dann zufällige Zahlenfolgen für stochastische Simulationen nutzen und damit unter anderem näherungsweise einige Wahrscheinlichkeiten und Flächen berechnen. Ebenso werden wir dabei Anwendungen aus der Finanzwelt kennenlernen, denn man kann zum Beispiel auch Aktienkurse und Zinsverläufe stochastisch simulieren.

Zu guter Letzt aber hoffe ich – und das ist das Wichtigste – dass euch der Kurs Spaß gemacht hat und dass ihr bald mal wiederkommt.