FEHLERKORREKTUR UND VERSCHLOSSELUNG - ZWEI ASPEKTE DER CODIERUNGSTHEORIE

H. K. Kaiser, Institut für Algebra und Diskrete Mathematik,
Technische Universität Wien x)

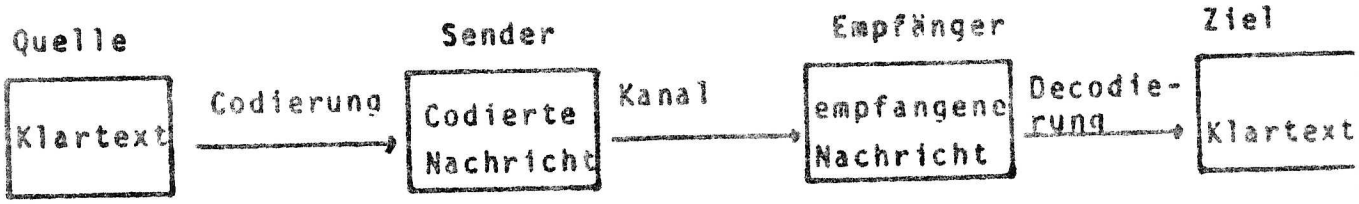
1. Einleitung.

Einige Teilgebiete der Mathematik werden, zumindest aus der Perspektive des Schulunterrichtes gesehen, für rein abstrakte Theorien oder nette Spielereien gehalten. Ich meine damit Algebra und Zahlentheorie. Die Inhalte dieser Gebiete werden zwar als innermathematisch relevant empfunden, aber die "nützlichen" Teildisziplinen der Mathematik sind eben Analysis, Wahrscheinlichkeitstheorie usw. Ich möchte in den folgenden Ausführungen einige Grundprobleme der Codierungstheorie vorstellen und exemplarisch einige Methoden zu deren Lösung diskutieren und damit diese einseitige Perspektive ein wenig korrigieren.

Die Codierungstheorie ist eine junge Teildisziplin der Mathematik und ist aus konkreten Fragen der Anwendungen entstanden. Sie bedient sich in erster Linie algebraischer und algebraisch-zahlentheoretischer Methoden. Inwiefern damit relevante Probleme unseres modernen Lebens, und vielleicht sogar Überlebens, gelöst werden können, will ich nun ausführen.

Im allgemeinen versteht man unter Codierung die Darstellung einer Nachricht in anderer Form, also etwa die Übertragung in eine Folge von Symbolen. Man assoziiert mit dem Wort "Code" Dinge wie das Morsealphabet oder Geheimcodes, wie sie von Spionen und Militärs verwendet werden. Der Vorgang ist dabei auf folgende Art schematisierbar:

Gegeben sei eine Nachricht N (Klartext). Diese wird codiert (d.h. für die jeweiligen Zwecke in eine geeignete Symbolform gebracht) und über einen sogenannten Kanal von einem Sender an einen Empfänger übermittelt. Dieser decodiert die Nachricht wiederum in den Klartext.



x) Ausarbeitung eines Vortrages, der im Rahmen des ÖMG-Lehrerfortbildungstages 1981, 3.April 1981, Mathematisches Institut der Universität Wien, gehalten wurde.

Viele Vorgänge in der modernen NachrichtenÜbertragung und Nachrichtenverarbeitung lassen sich auf diese Weise schematisch beschreiben.

Betrachten wir etwa den Vorgang der Obertragung eines Bildes aus einem Raumschiff. Dabei wird das Bild zunächst durch eine Fernsehkamera in Zeilen zerlegt, dann jede Zeile in kleine Quadrate. Jedes dieser Quadrate wird durch ein Symbol dargestellt, das durch Farbe und Intensität bestimmt wird. Zeile für Zeile wird in dieser Symbolform gesendet. Der Empfänger auf der Erde setzt das Bild zusammen, indem er den inversen Vorgang zum oben beschriebenen Codierungsvorgang durchführt.

Auf ähnliche Weise läßt sich der Vorgang der Speicherung und Abruf von Daten in einem Computer beschreiben.

Entsprechend dem jeweiligen Zweck stellt man an einen Code eine oder mehrere der folgenden Forderungen:

(a) Die Codierung und Decodierung muß bequem, billig und rasch ausführbar sein.

(b) Die Obertragung soll sicher vor zufälligen Fehlern sein (menschliche Fehler bei der Bedienung; Störung bei der KanalÜbertragung, z.B. "Rauschen").

(c) Geheimhaltung der Nachricht bei der Obermittlung.

(d) Fälschungssicherheit (Authentizität der Nachricht kann gewährleistet werden).

Die erste Forderung wird z.B. durch das Morsealphabet erfüllt. Es ist leicht erlernbar und kann auf die vielfältigste Weise gesendet werden (Telegraph, Licht- oder Rauchsignale). Aber die Forderung nach Geheimhaltung und Fälschungssicherheit wird sicher nicht erfüllt, denn jeder kann den Code benützen. Auch das Prinzip (b) ist dabei nicht gewährleistet, da z.B. bei einem Senden eines Striches - anstelle eines Punktes · der Buchstabe s = ... zu d = - .. oder r = . - . nder u = .. - empfangen werde kann.

Zur Verwirklichung der Forderung (b) hat sich in den letzten 35 Jahren eine umfangreiche Theorie entwickelt, die sogenannte algebraische Codierungstheorie. Die Probleme (c) und (d) sind viel älter

Diese Aspekte sind Bestandteil der sogenannten Kryptologie, die man bis in die jüngste Vergangenheit mehr als eine Kunst und nicht als eine Wissenschaft angesehen hat. Die Entwicklung der letzten Jahre mit der Entdeckung neuer faszinierender Codes weist nun stärker in Richtung Wissenschaft, wie wir in Abschnitt 3 zeigen werden.

2. Codes zur Fehlererkennung und Fehlerkorrektur.

Benützt man einen Code, der leicht gelernt werden kann und von Hand aus gesendet wird, so muß man mit einem gewissen Prozentsatz von Fehlern rechnen. Aber im Zeitalter der Elektronik, in dem große Informationsmengen elektronisch gesendet und von Computern verarbeitet werden können, ist es möglich, komplizierte Codes zu entwerfen, die Fehler erkennen und diese sogar automatisch korrigieren zu können.

Betrachten wir folgenden Vorgang: Zur Codierung verwenden wir nur die zwei Symbole 0 und 1. Um einen normal verfaßten Text (den sogenannten Klartext) in eine Symbolfolge zu übersetzen, können wir beispielsweise jeden der 26 Buchstaben des Alphabets eindeutig durch einen Block aus fünf Stellen darstellen (das ist möglich, da es $2^5=32$ Blöcke aus fünf Stellen gibt):

A = 00000
B = 00001
C = 00010
D = 00011
⋮
Z = 11001 (=25 in binärer Form)

Auf diese Weise haben wir aber keine größere Verläßlichkeit als im Morsealphabet erreicht . Wird etwa die letzte Stelle in A fälschlich als 1 gesendet, so würden wir B decodieren und wüßten nicht, daß ein Fehler aufgetreten ist. Daher fügt man eine 6.Stelle in jedem Block an, die man auf folgende Weise mit 0 oder 1 belegt:

Ist die Summe der ersten fünf Stellen gerade, so setzen wir 0; ist sie ungerade, so schreiben wir 1. Der neue Code sieht dann so aus:

A = 000000
B = 000011
C = 000101
D = 000110
⋮
Z = 110011

Den so erhaltenen Code nennt man Quersummencode. Da die gegebene Nachricht dabei in gleichlange Blöcke aufgeteilt wird, spricht man von einem Blockcode, genauer von einem (6,1)-Blockcode, da jeder Block die Länge 6 hat, und die letzte Stelle ein Kontrollsymbol repräsentiert.

Tritt in einem Block des obigen Quersummencodes ein einziger Fehler auf (man nennt diesen auch Einfachfehler), so erhält der Empfänger einen Block mit einer ungeraden Zahl von Einsern, der also nicht im Code ist. Also weiß der Empfänger, daß ein Fehler passiert ist. Daher nennt man diesen Code "Einfachfehlererkennend".

Tritt beispielsweise in der fünften Stelle von A ein Fehler auf, so wissen wir, daß etwas falsch ist, denn 000010 ist nicht im Code. Aber selbst wenn wir wüßten, daß genau ein Fehler aufgetreten ist, könnten wir ihn nicht korrigieren, denn auch andere Blöcke könnten bei Auftreten von Einfachfehlern als 000010 empfangen werden.

Um Fehler korrigieren zu können, muß man also die Verschiedenheit zwischen den einzelnen Codeblöcken groß machen. Naheliegenderweise erklärt man als Maß $d(\alpha,\beta)$ für die Verschiedenheit zweier Codeblocks α und β die Anzahl der Stellen, in denen α und β verschieden sind. Beispielsweise ist $d(000000,000011)=2$.

d ist also eine Abbildung von $C \times C$ (C bezeichnet die Menge der Codeblöcke) in $\mathbb{N}_0$, genannt der Hamming-Abstand von α und β (R.W.Hamming ist einer der Pioniere der algebraischen Codierungstheorie). d hat die üblichen Eigenschaften einer Distanzfunktion.

Gelingt es nun, einen Code so zu konstruieren, daß verschiedene Codeblöcke α,β stets einen Abstand ≥ 3 besitzen, dann tritt bei einem Einfachfehler bei der Übermittlung von α ein α' mit $d(\alpha,\alpha')=1$ auf, das von allen anderen Wörtern des Codes einen Abstand ≥ 2 hat. Daher kann man jeden Einfachfehler in dem Sinn korrigieren, daß man anstelle des empfangenen α' den nächstgelegenen Block, also α, als gesendet annimmt. Deshalb nennt man Blockcodes mit Minimalabstand ≥ 3 auch "Einfachfehlerkorrigierende" Codes.

Analog sind Codes mit Minimalabstand ≥5 zwischen verschiedenen Codewörtern Zweifachfehlerkorrigierend, solche mit Minimalabstand ≥7 Dreifachfehlerkorrigierend usw.

Beispiel für einen Einfachfehlerkorrigierenden Code ist etwa die dreifache Sendung jedes Nachrichtenwortes. Man nimmt beim Empfang jenes Symbol an einer Stelle des Blocks als gesendet an, das mindestens zweimal an der entsprechenden Stelle empfangen wird. Dieser Code hat aber den Nachteil, daß er sehr aufwendig ist, also sehr "teuer".

In diesem Zusammenhang ist folgendes Problem von Interesse:

Gegeben ist eine Blocklänge n und ein Hamming-Abstand h.
Man bestimme die Maximalanzahl A(n,h) von binären Wörtern (=Blocks) der Länge n, die voneinander den Mindestabstand h besitzen.

Mit Hilfe von geometrisch-kombinatorischen Methoden erhält man für h=3 beispielsweise folgendes Resultat:

n	3	4	5	6	7	8	9
A(n,3)	2	2	4	8	16	20	≥38,≤40.

Der interessanteste unter diesen Codes ist der Code der Länge 7, bestehend aus 16 Codewörtern. Er gehört zu einer Familie von Codes, die von Hamming entdeckt und nach ihm benannt wurde.

Die ersten vier Stellen $a_1a_2a_3a_4$ sind jeweils durch ein Element aus $\{0,1\}^4$ gegeben. Die letzten drei Stellen errechnet man aus den ersten vier Stellen auf folgende Weise:

$$a_5 = a_1+a_2+a_4 \quad (\text{mod}2)$$
$$a_6 = a_1+a_3+a_4 \quad (\text{mod}2)$$
$$a_7 = a_2+a_3+a_4 \quad (\text{mod}2).$$

Wir haben gesehen, daß der Minimalabstand zwischen verschiedenen Codewörtern RUckschlüsse auf die GUte eines Codes zur Fehlerkorrektur zuläßt. Wie bestimmt man diesen Minimalabstand auf einfache Weise? Dazu definiert man als Gewicht $w(\alpha)$ eines Codewortes α die Anzahl der Einser in α. Weiters wählt man die Codewörter so, daß man sie mit einer algebraischen Struktur versehen kann. Man faßt die Codewörter (Länge n) als Elemente von $\langle\{0,1\}^n,+\rangle$ auf (+ bezeichnet die Addition modulo 2). Bildet die Menge C der Codewörter bezüglich der

komponentenweisen Addition modulo 2 eine Gruppe, so spricht man von einem Gruppencode.

Dann kann man folgenden Satz zeigen:

Ist C ein Gruppencode mit den Codewörtern $\alpha_1,\ldots,\alpha_n$, so ist der kleinste Abstand $d(\alpha_i,\alpha_j)$ zweier Codewörter $\alpha_i \neq \alpha_j$ gleich dem kleinsten Gewicht $w(\alpha_z)$ der von Null verschiedenen Codewörter (mit Null bezeichnen wir jenes Codewort, das neutrales Element der Gruppe $<C,+>$ ist).

Gruppencodes bieten weiter den Vorteil, daß sie auf effiziente Weise decodiert werden können. Man geht dabei so vor:

1.Schritt: Man schreibt eine Liste aller Codewörter auf, die die Gruppe C bilden, beginnend mit Null:

z.B. 000000 100110 010101 001011 110011 101101 011110 111000

2.Schritt: Man wählt ein Wort $x \in \{0,1\}^n$ von kleinstem Gewicht unter jenen Wörtern aus $\{0,1\}^n$, die noch nicht in der Liste verwendet worden sind. Man schreibt die Linksnebenklasse x+C in die nächste Zeile, wobei man x+C stets unter $c \in C$ anschreibt:

000000 100110 010101 001011 110011 101101 011110 111000
100000 000110 110101 101011 010011 001101 111110 011000

3.Schritt: Man wiederholt den zweiten Schritt so lange, bis alle Elemente von $\{0,1\}^n$ erfaßt sind:

000000 100110 010101 001011 110011 101101 011110 111000
100000 000110 110101 101011 010011 001101 111110 011000
010000 110110 000101 011011 100011 111101 001110 101000
001000 101110 011101 000011 111011 100101 010110 110000
000100 100010 010001 001111 110111 101001 011010 111100
000010 100100 010111 001001 110001 101111 011100 111010
000001 100111 010100 001010 110010 101100 011111 111001
100001 000111 110100 101010 010010 001100 111111 011001

Diese Liste nennt man Standardanordnung oder Decodierungstafel.

4.Schritt: Man decodiert jedes empfangene Wort α durch das oberste Wort jener Spalte, in der α steht.

Die Elemente der ersten Spalte heißen Anführer der Nebenklassen. Daher nennt man das Verfahren auch "Decodierung durch Anführer der Nebenklassen".

Definiert man ein Fehlerwort e als jenes Wort, das zum gesendeten Wort α addiert das empfangene Wort α' ergibt, d.h. $\alpha+e=\alpha'$, so gilt für die Decodierung mittels Standardanordnung:

Genau jene Fehlerwörter werden korrigiert, die Anführer von Nebenklassen sind.

Ich hoffe, mit diesen Ausführungen einen Einblick in die Grundprobleme der algebraischen Codierung gegeben zu haben. Natürlich werden die Methoden heute verfeinert und ausgebaut angewendet. Es liegt auf der Hand, die Codewörter als Vektoren über dem endlichen Körper GF(2) anzusehen, oder - bei Verwendung anderer Übertragungskanäle - allgemein als Elemente eines Vektorraumes über GF(q).

Damit ist es möglich, die Codes den realen Anforderungen besser anzupassen. Man kann sie beispielsweise zur Korrektur von Fehlerbündel usw. einsetzen. Für Details sei auf die angegebene Literatur verwiesen.

3. Kryptologie.

Wir wollen uns nun noch mit der mathematischen Behandlung der in der Einleitung aufgestellten Forderungen (c) und (d) beschäftigen. Hauptanliegen ist dabei zunächst, die Nachricht für dritte Personen nicht entzifferbar zu machen. Es ist üblich, in der Kryptologie für Decodierung "Entzifferung" oder "Dechiffrierung" zu sagen, für Codierung "Verschlüsselung" oder "Chiffrierung".

Eine klassische Verschlüsselung wurde von Cäsar verwendet. Darüber berichtet Suetonius in seinem Buch "Die 12 Cäsaren". Demnach soll Cäsar seine Botschaften durch zyklische Verschiebung der Buchstaben des Alphabets um 3 Stellen verschlüsselt haben, also nach folgendem Code:

A B C D ..... W X Y Z
D E F G ..... Z A B C

Ganz allgemein nennt man eine solche zyklische Verschiebung der Buchstaben des Alphabets um n Stellen (wobei natürlich (n,26)=1 sein muß, um Eindeutigkeit zu erreichen) eine Cäsar-Verschiebung.

Allerdings ist dieser Code von Gegnern relativ leicht zu brechen, wenn sie die Botschaft abfangen. Die Buchstaben des Alphabets treten in den natürlichen Sprachen mit unterschiedlicher Häufigkeit auf. So machen zum Beispiel E, T, A, O, N, R etwa 50% der Buchstaben in den Wörtern der englischen Sprache aus, E allein ungefähr 13%. Durch eine Häufigkeitsanalyse an Hand von wenigen Dutzend Symbolen kann man einen dieser häufigen Buchstaben bestimmen und daraus die Verschiebung berechnen. Damit ist das ganze System geknackt.

1586 konzipierte Blaise de Vigenère ein Verschlüsselungssystem, das ebenfalls auf der Verschiebung von Buchstaben des Alphabets beruht. Die Länge der Verschiebung ändert sich aber periodisch. Das folgende Beispiel soll den Vorgang illustrieren:

Nachricht	S E N D E T	M E H R	T R U P P E N	U N D	W A F F E
Verschiebungsfolge	1 7 4 13 5 1	7 4 13 5	1 7 4 13 5 1 7	4 13 5	1 7 4 13 5
Verschlüsselte Nachricht	T L R Q J U	T I U W	U Y Y C U F U	Y A I	X H J S J

Um diesen Code zu brechen, bemüht man sich zunächst, die Länge der Verschiebungsfolge zu bestimmen. Dazu versucht man, sich wiederholende Buchstabenblöcke zu finden und aus deren Abständen die Länge zu eruieren. Sodann führt man eine Häufigkeitsanalyse wie bei der Cäsarverschlüsselung für Buchstaben im Abstand der gefundenen Länge durch.

Als Verallgemeinerung der Methode von Vigenère entwickelte G.S. Vervam 1926 die Einweg-Schablone mit Zufallsziffern. Er ging dabei von Idee aus, die Verschiebungsfolge beliebig lang und zufällig zu wählen. Diese Methode ist mit beträchtlichen Schwierigkeiten verbunden. Sender und Empfänger müssen eine lange Folge von Zufallsverschiebungen kennen (so lange wie die Nachricht ist), sie ist nur einmal verwendbar, aber man kann sie durch Abfangen der Botschaft allein nicht knacken, denn die Zufälligkeit der Verschiebungsfolge impliziert, daß je zwei Nachrichtenfolgen derselben Länge gleich wahrscheinlich der chiffrierten Folge zugrunde liegen.

Bei den bisher vorgestellten Methoden hängt die Sicherheit des Systems von der Geheimhaltung des Codes ab. Auch muß der Code Sender und Empfänger bekannt sein, die also vor dem Austauschen der Nachricht zur Vereinbarung des Codes Kontakt aufnehmen müssen. Obendrein kann der Forderung (d) nach Authentizität nicht Folge geleistet werden.

1976 hatten W.Diffie und M.Hellman die Idee für eine Chiffriermethode, die auf überraschende Weise die eben genannten Schwierigkeiten löste. Sie entwickelten dabei die sogenannten öffentlichen Chiffriersysteme:

Jeder Person A wird ein öffentlicher Schlüssel S_A und ein privater Schlüssel T_A zugeordnet. Natürlich soll für jede Nachricht N die Hintereinanderanwendung der Schlüssel wiederum N ergeben, d.h.: $T_A \circ S_A(N) = S_A \circ T_A(N) = N$. S_A wird allgemein zugänglich gemacht (etwa in einer Art Telefonbuch), T_A bleibt nur A (oder eventuell nur dessen Computer) bekannt. Sendet A an B eine Nachricht N, so sieht er S_B im "Telefonbuch" nach und sendet $S_B(N)$ an B. Nur dieser kennt T_B, kann also $T_B \circ S_B(N) = N$ bestimmen.

Auch die Authentizität der Nachricht kann man nach dieser Methode gewährleisten. Dazu verschlüsselt A die Nachricht N zuerst mit T_A, dann mit S_B, sendet also $S_B \circ T_A(N)$ an B. B bildet $T_B \circ S_B \circ T_A(N)$, besorgt dann den öffentlichen Schlüssel S_A und bildet $S_A \circ T_B \circ S_B \circ T_A(N) = N$. Diese Nachricht kommt sicher von A, denn nur dieser kennt T_A.

Neben der Forderung der leichten und raschen Handhabung der Schlüssel T und S ist natürlich zur Geheimhaltung notwendig, daß es hoffnungslos schwer ist, T_A aus dem allgemein bekannten S_A zu bestimmen. 1977 haben R.Rivest, A.Shamir und L.Adleman dieses Problem mittels "einseitiger Probleme" ("trapdoor problems") gelöst. Dabei handelt es sich um folgendes: Jeder von uns hat schon am eigenen Leib verspürt, daß die Zerlegung einer gegebenen Zahl n in Primfaktoren eine sehr langwierige und mit großem Rechenaufwand verbundene Aufgabe sein kann. Die Probe jedoch ist sehr rasch und einfach durchführbar. Das Faktorisierungsproblem einer Zahl n ist also von "einseitigem" Schwierigkeitsgrad.

Diese Tatsache wird im Rivest-System verwendet. Zunächst wird die Nachricht als natürliche Zahl N unter Benützung des folgenden Schemas dargestellt:

a = 00	b = 01	c = 02	d = 03	e = 04	f = 05	g = 06
h = 07	i = 08	j = 09	k = 10	l = 11	m = 12	n = 13
o = 14	p = 15	q = 16	r = 17	s = 18	t = 19	u = 20
v = 21	w = 22	x = 23	y = 24	z = 25	A = 26	B = 27
C = 28	D = 29	E = 30	F = 31	G = 32	H = 33	I = 34
J = 35	K = 36	L = 37	M = 38	N = 39	O = 40	P = 41
Q = 42	R = 43	S = 44	T = 45	U = 46	V = 47	W = 48
X = 49	Y = 50	Z = 51	0 = 52	1 = 53	2 = 54	3 = 55
4 = 56	5 = 57	6 = 58	7 = 59	8 = 60	9 = 61	= 62
. = 63	, = 64	; = 65	? = 66	...		

Die gesuchte natürliche Zahl N erhält man durch Obersetzung der einzelnen Zeichen des Klartextes in die jeweiligen zwei Dezimalziffern und deren Juxtaposition.

Im Rivest-Schema besteht der öffentliche Schlüssel S aus zwei natürlichen Zahlen s und m, der private Schlüssel T aus einer natürlichen Zahl t. Wir nehmen an, daß N<m ist (andernfalls zerlegt man N in Blöcke der Länge <m). Wir bestimmen nun zwei große Primzahlen (siehe Anhang) p und q und bilden m=p.q. Man wählt beispielsweise p und q beide 100-stellig. Ein so entstehendes 200-stelliges m kann aber nach den heute bekannten Methoden in einer vernünftigen Zeit nicht zerlegt werden. Ein Rechner, der 10^6 Rechenschritte pro Sekunde durchführt, würde nach den heute bekannten Algorithmen für die Zerlegung eines solchen m rund $3{,}8.10^8$ Jahre benötigen. Also kann m mit ruhigem Gewissen publiziert werden. Nun wählt man zufällig eine große Zahl t, die zu (p-1)(q-1) teilerfremd ist. Schließlich errechnet man s so, daß $s.t \equiv 1 \bmod ((p-1)(q-1))$ gilt.

Mit den so bestimmten Zahlen m,s,t sieht der Vorgang der Übermittlung so aus:

Ein Partner sendet $V=N^s \pmod m$. Der Empfänger (nur er kennt t) berechnet $V^t \pmod m$. Damit kennt er die Nachricht N, denn

$$V^t=(N^s)^t=N^{st}=N^{1+k(p-1)(q-1)} \equiv N \pmod m.$$

Die Anwendungen dieses Systems liegen auf der Hand, etwa bei elektronisch durchgeführten Bankgeschäften (in den USA werden dabei jährlich rund 300 Millionen Dollar gestohlen, meist von Computerspezialisten). Ich möchte hier noch auf eine Anwendungsmöglichkeit eingehen, nämlich beim Atomsperrvertrag. Diese Verwendung wird von A.Engel in seinem Artikel "Datenschutz durch Chiffrieren" (siehe Literaturverzeichnis) ausführlich beschrieben. Im Zuge des Atomsperrvertrages zwischen USA und der Sowietunion wurde es jedem der Vertragspartner gestattet, seismische Geräte im Land des Partners aufzustellen. Diese Geräte registrieren jede Erschütterung und damit jede unterirdische Atomexplosion. Man kann diese Instrumente vor Manipulationen schützen, indem man sie so konstruiert, daß sie sich selbst zerstören, wenn Unbefugte an ihnen hantieren. Aber der Informationskanal, über den die Geräte ihre Nachricht in die Heimat senden, ist nicht sicher. Denn das jeweilige Land, in dem die Geräte aufgestellt sind, könnte falsche Nachrichten unterlegen. Daher

werden die Signale verschlüsselt. Nun befürchtet aber die jeweilige Nation, daß die Geräte zur Spionage verwendet werden und auch andere Informationen senden.

Das Rivestsystem bietet sich hier als Ausweg an. Jede Nation verschlüsselt die Daten mit ihren privaten Schlüssel und kann damit die Nachrichten sofort mit dem jeweiligen öffentlichen Schlüssel entziffern. Damit wäre der Verdacht auf Spionage entkräftet. Um falsche Daten zu substituieren, müßte man das Rivestsystem knacken.

Anhang: Ein Verfahren zur Bestimmung von großen Primzahlen.

Um das zuvor beschriebene öffentliche Chiffriersystem in der Praxis wirklich verwendbar zu machen, benötigt man ein rasches Verfahren zur Bestimmung von großen Primzahlen. Normalerweise ist es nur mit hohem Rechenaufwand möglich festzustellen, ob eine vorgegebene Zahl eine Primzahl ist. 1976 gelang es M.O.Rabin, einen Algorithmus zu entwickeln, der mit hoher Wahrscheinlichkeit richtig entscheidet, ob eine gegebene Zahl eine Primzahl ist. Er benützte dazu den Begriff der starken Pseudoprimzahl. Sei eine Zahl $N=2^s t+1$ (t ungerade) vorgegeben. Dann nennen wir N eine starke Pseudoprimzahl zur Basis b, wenn entweder

$$b^t \equiv 1 \pmod N,$$

oder

$$b^{t2^r} \equiv -1 \pmod N \text{ für ein } r \text{ mit } 0 \le r < s.$$

Natürlich ist - wie man unter Benützung des kleinen Satzes von Fermat leicht nachrechnet - jede Primzahl p eine starke Pseudoprimzahl zur Basis b für jedes b mit $1 \le b \le p-1$. 2047 ist (das kleinste) Beispiel für eine zerlegbare Zahl, die starke Pseudoprimzahl zur Basis 2 ist.

Für zerlegbares N ist N für mindestens 75% aller Zahlen $1 \le b \le N-1$ keine starke Pseudoprimzahl zur Basis b (wie M.O.Rabin zeigte). Man geht daher bei der Aufgabe der Oberprüfung, ob eine gegebene Zahl N eine Primzahl ist, auf folgende Weise vor:

Man wählt zufällig k verschiedene Zahlen b mit $1 \le b \le N-1$ aus und sieht nach, ob N eine starke Pseudoprimzahl zur Basis b für jedes der ausgewählten b ist (dafür gibt es einen "guten" Algorithmus). Ist dies der Fall, so ist N eine Primzahl. Diese Aussage ist nur mit einer Fehlerwahrscheinlichkeit von $\le 4^{-k}$ behaftet.

Mit großem Rechenaufwand zeigten Selfridge und Wagstaff, daß für $N < 2{,}5 \cdot 10^{10}$ die einzige zerlegbare starke Pseudoprimzahl zur Basis b für b=2,3,5,7 die Zahl 3 215 031 751 ist.

Somit kann man mit Hilfe des Rabin-Tests leicht große Primzahlen berechnen.

Literatur

[1] G.Birkhoff - T.Bartee: Angewandte Algebra.
R.Oldenburg Verlag, München-Wien 1973.

[2] W.Diffie - M.Hellman: New directions in cryptography.
IEEE Transactions on information theory, IT-22; 644-654 (1976).

[3] L.Dornhoff-E.Hohn: Applied modern algebra.
Mac Millan, New York 1978.

[4] A.Engel: Datenschutz durch Chiffrieren: Mathematische und algorithmische Aspekte. MU,Jg.25, Heft 6 (1979).

[5] J.L.Fisher: Application oriented algebra.
Dun-Donnelly Publ., New York 1977.

[6] A.Gill: Applied algebra for the computer sciences.
Prentice-Hall, New Jersey 1976.

[7] R.W.Hamming: Coding and informationtheory.
Prentice-Hall, New Jersey 1980.

[8] M.E.Hellman: Die Mathematik neuer Verschlüsselungssysteme.
Spektrum der Wissenschaft 10, 92-101 (1979).

[9] E.Henze - H.Homuth: Einführung in die Codierungstheorie.
Uni-Text,Vieweg 1974.

[10] H.Kautschitsch: Eine genetische Einführung in die Zahlentheorie.
Preprint, Universität Klagenfurt

[11] W.Peterson - E.Weldon: Error correcting codes.
MIT-Press, Cambridge, Mass. 1962.

[12] M.O.Rabin: Probabilistic algorithms. In J.F.Traub (ed.):
Algorithms and complexity, recent results and new directions.
Academic Press 1976, 21-40.

[13] R.Rivest - A.Shamir - L.Adleman: A method for obtaining digital signatures and publik-key cryptosystems. Communications of the ACM, Feb.1978, 120-126.

H.K.Kaiser
Institut für Algebra und Diskrete Mathematik
Technische Universität Wien