Kapitel 4
Faktorisierungsverfahren

Wie wir in §2 des letzten Kapitels gesehen haben, ist $M_{67} = 2^{67} - 1$ keine Primzahl, denn

$$13^{M_{67}-1} \equiv 81\,868\,480\,399\,682\,966\,751 \bmod M_{67} \not\equiv 1 \bmod M_{67}\,.$$

Somit ist M_{67} ein Produkt von mindestens zwei nichttrivialen Faktoren. Welche sind das?

FRANK NELSON COLE gab das Ergebnis am 31. Oktober 1903 auf einer Sitzung der American Mathematical Society bekannt: Er schrieb die Zahl

$$2^{67} - 1 = 147\,573\,952\,589\,676\,412\,927$$

auf eine der beiden Tafeln und

$$193\,707\,721 \times 761\,838\,257\,287$$

auf die andere. Dieses Produkt rechnete er wortlos aus nach der üblichen Schulmethode zur schriftlichen Multiplikation, und als er dieselbe Zahl erhielt, die auf der anderen Tafel stand, schrieb er ein Gleichheitszeichen zwischen die beiden Zahlen und setzte sich wieder. Das Ergebnis, d.h. die Faktorisierung von M_{67}, findet ein Computeralgebrasystem heute in weniger als einer Sekunde; für die damalige Zeit war sie eine Sensation! COLE gab später zu, daß er drei Jahre lang jeden Sonntag nachmittag daran gearbeitet hatte. Er versuchte M_{67} in der Form $x^2 - y^2$ darzustellen, wobei er mit Hilfe quadratischer Reste Kongruenzbedingungen für x modulo verschiedener relativ kleiner Primzahlen aufstellte und auch verwendete, daß jeder Teiler von M_{67} kongruent eins modulo 67 und kongruent ± 1 modulo acht sein muß. Dies führte zu einer ganzen Reihe

von Kongruenzen für x, die er in

$$x \equiv 1\,160\,932\,384 \bmod 1\,323\,536\,760$$

zusammenfassen konnte. Untersuchung quadratischer Reste zeigt, daß

$$x_k = 1\,323\,536\,760\ k + 1\,160\,932\,384$$

frühestens für $k = 287$ in Frage kommt, und mit $x = x_{287}$ ist tatsächlich

$$\begin{aligned} M_{67} &= 381\,015\,982\,504^2 - 380\,822\,274\,783^2 \\ &= 193\,707\,721 \times 761\,838\,257\,287\,. \end{aligned}$$

Für Einzelheiten siehe

F. N. COLE: On the factoring of large numbers, *Bull. Am. Math. Soc.* **10** (1903), 134–137 *oder* `http://www.ams.org/bull/1903-10-03/S0002-9904-1903-01079-9/home.html`

FRANK NELSON COLE (1861–1926) wurde in Massachusetts geboren. 1882 erhielt er seinen Bachelor in Mathematik von der Harvard University; danach konnte er dank eines Stipendiums drei Jahre lang bei FELIX KLEIN in Leipzig studieren. Mit einer von KLEIN betreuten Arbeit über Gleichungen sechsten Grades wurde er 1886 in Harvard promoviert. Nach verschiedenen Positionen in Harvard und Michigan bekam er 1895 eine Professor an der Columbia University in New York, wo er bis zu seinem Tod lehrte. Seine Arbeiten befassen sich hauptsächlich mit Primzahlen und mit der Gruppentheorie.

Der Auftritt von COLE schlug selbst außerhalb der Mathematik so große Wellen, daß seine Faktorisierung noch fast ein Jahrhundert später vorkommt in einer New Yorker (off-Broadway) Show von RINNE GROFF mit dem Titel *The five hysterical girls theorem.* Dort bringt sich ein junger Mathematiker um, weil er in einem Beweis von der *Primzahl* $2^{67} - 1$ ausgeht und die Tochter des Professors die obige Faktorisierung an die Tafel schreibt. Einzelheiten kann man, so man unbedingt möchte, unter `http://www.playscripts.com/play.php3?playid=551` nachlesen. (Die Show verschwand nach zwei Monaten Ende Mai 2000 in der Versenkung; sie wurde seither nur noch zweimal von Amateurgruppen aufgeführt.)

COLE konnte für seine Faktorisierung von M_{67} auf bekannte Tatsachen über die Struktur von Faktoren der MERSENNE-Zahlen zurückgreifen und auch bei den von ihm selbst gefundenen Eigenschaften potentieller Faktoren konnte er die spezielle Struktur von M_{67} ausnutzen. Ähnlich arbeiten auch heutige Mathematiker an der Faktorisierung spezieller Zahlen, beispielsweise im Rahmen des Cunningham-Projekts zur Faktorisierung von Zahlen der Form $b^n \pm 1$ für kleine Basen b. Für die Faktorisierung von RSA-Moduln kann man natürlich nicht mit solchen Techniken arbeiten. In diesem Kapitel soll es um Verfahren gehen, mit denen man eine zufällig gegebene Zahl ohne spezielle Struktur faktorisieren kann.

Es gibt kein „bestes“ Faktorisierungsverfahren; für Zahlen verschiedener Größenordnungen haben jeweils andere Verfahren ihre Stärken. Auch Vorwissen über die zu faktorisierende Zahl kann bei der Wahl eines geeigneten Verfahrens helfen: Bei einem RSA-Modul, der das Produkt zweier Primzahlen ähnlicher Größenordung ist, wird man anders vorgehen als bei einer Zahl der Form $b^n \pm 1$. Mehr noch als bei Primzahltests gilt, daß asymptotische Komplexitätsaussagen als Auswahlkriterium nutzlos sind: Das für die Faktorsierung 150-stelliger RSA-Moduln heute optimale Verfahren, das Zahlkörpersieb, wird beim Versuch eine sechsstellige Zahl zu faktorisieren, oft nicht in der Lage sein die Faktoren zu trennen, und selbst in den Fällen, in denen es erfolgreich ist, braucht es erheblich länger als einfache Probedivisionen. Es gibt definitiv kein „bestes“ Faktorisierungsverfahren; je nach Größe der zu faktorisierenden Zahl und auch nach Größe der zu erwartenden Faktoren können ganz verschiedene Strategien angebracht sind, die oft nur in der Kombination zur vollständigen Primzerlegung führen.

Im folgenden sollen einige der einfachsten gebräuchlichen Verfahren vorgestellt werden.

§ 1: Die ersten Schritte

a) Test auf Primzahl

Der schlimmste Fall für praktisch jedes Faktorisierungsverfahren tritt dann ein, wenn die zu faktorisierende Zahl eine Primzahl ist: Gerade bei

den fortgeschrittenen Verfahren gibt es oft kein anderes Abbruchkriterium als das Auffinden eines Faktors. Daher sollte (außer eventuell bei ganz kleinen Zahlen) zu Beginn einer Faktorisierung immer ein Primzahltest stehen. Da auch das Testen auf Potenzen relativ einfach ist, läßt sich eventuell auch das noch durchführen – es sei denn, daß von der Situation her (beispielsweise bei RSA-Moduln) nicht mit einer Potenz zu rechnen ist.

b) Abdividieren kleiner Primteiler

Bei kleinen zusammengesetzten Zahlen n besteht die effizienteste Art der Faktorisierung im allgemeinen darin, einfach alle Primzahlen nacheinander durchzuprobieren, indem man sie der Reihe nach so lange abdividiert, wie es geht. Sobald der Quotient kleiner ist als das Quadrat der gerade betrachteten Primzahl, kann man sicher sein, daß auch er eine Primzahl ist und hat n vollständig faktorisiert.

Die genaue Vorgehensweise ist folgende: Wir nehmen an, daß eine Liste der Primzahlen bis zu einer gewissen Grenze zur Verfügung steht; gegebenenfalls muß diese zunächst nach ERATOSTHENES erzeugt werden.

1. Schritt: Setze M gleich der zu faktorisierende Zahl und $p = 2$.

2. Schritt: Solange M durch p teilbar ist, ersetze M durch M/p und notiere p als Faktor.

3. Schritt: Falls $M = 1$, sind alle Faktoren gefunden, und der Algorithmus endet. Ansonsten wird p auf die nächste Primzahl gesetzt. Ist dann $M < p^2$, muß M prim sein und wird als weiterer Faktor notiert; sodann endet auch in diesem Fall der Algorithmus. Andernfalls geht es zurück zum zweiten Schritt.

Als Beispiel wollen wir die Zahl 1 234 567 890 faktorisieren:

Im ersten Schritt werden $M = 1\,234\,567\,890$ und $p = 2$ initialisiert.

Im zweiten Schritt ist nun $p = 2$. Da M eine gerade Zahl ist, können wir durch p dividieren; wir notieren also die Zwei als Faktor und ersetzen M durch $M/2 = 617\,283\,945$. Diese Zahl ist ungerade; also geht es weiter

zum dritten Schritt, wo offensichtlich keines der Abbruchkriterien erfüllt ist. Somit wird $p = 3$ und es geht zurück zum zweiten Schritt.

Das neue M ist durch drei teilbar, genauso der Quotient $M/3 = 205\,761\,315$. Also wird nochmals durch drei dividiert, und wir erhalten den nicht mehr durch drei teilbaren Quotienten 68 587 105. Somit werden zwei Faktoren drei notiert und es geht weiter zum dritten Schritt. Dort wird $p = 5$ gesetzt, und es geht wieder zurück zu Schritt 2.

Das aktuelle $M = 68\,587\,105$ ist durch fünf teilbar mit Quotient $M/5 = 13\,717\,421$. Dieser wird das neue M; und da er nicht durch fünf teilbar ist, notieren wir nur einen Faktor fünf.

Im dritten Schritt wird wieder p erhöht und es geht zurück zum zweiten Schritt. Dort passiert nun allerdings lange Zeit nichts, denn keine der Primzahlen zwischen sieben und 3 593 teilt das aktuelle M. Erst wenn p im dritten Schritt auf 3 607 gesetzt wird, finden wir wieder einen Faktor. Wir notieren ihn, ersetzen M durch $M/3\,607 = 3803$, was offensichtlich nicht durch 3 607 teilbar ist, und gehen weiter zum dritten Schritt.

Dort ist nun offensichtlich $M < p^2$, also ist M eine Primzahl, und

$$1\,234\,567\,890 = 2 \cdot 3^2 \cdot 5 \cdot 3\,607 \cdot 3\,803$$

ist vollständig faktorisiert.

Es ist klar, daß wir schon eine zwanzigstellige Zahl nur mit viel Glück auf diese Weise mit vertretbarem Aufwand vollständig faktorisieren können. Trotzdem ist Abdividieren selbst für noch viel größere Zahlen ein sinnvoller erster Schritt, denn die auf größere Faktoren spezialisierten Verfahren schaffen es im allgemeinen nicht, auch kleine Primfaktoren voneinander zu trennen.

Um Abdividieren statt zur vollständigen Faktorisierung nur zur Identifikation „kleiner“ Primfaktoren zu verwenden, ist lediglich eine kleine Modifikation des dritten Schritts notwendig: Wir legen eine Suchgrenze S fest und brechen im dritten Schritt auch dann ab, wenn $p > S$ ist. Im letzteren Fall können wir selbstverständlich nicht behaupten, daß das verbleibende M eine Primzahl ist; M muß dann mit anderen Verfahren weiter bearbeitet werden. Die Größe von S hängt natürlich etwas ab

vom Arbeitsspeicher und der Geschwindigkeit des verwendeten Computers; ein minimaler Wert wäre etwa $2^{16} = 65\,536$; bei etwas besseren Computern läßt sich auch das Abdividieren bis zu einer Million und bei derzeit aktuellen schnellen Computern auch einer Milliarde oder etwa 2^{30} in weniger als einer Sekunde durchführen.

§ 2: Die Verfahren von Pollard und ihre Varianten

In den Jahren um 1975 entwickelte der britische Mathematiker JOHN M. POLLARD mehrere recht einfache Algorithmen zur Faktorisierung ganzer Zahlen sowie zur Berechnung diskreter Logarithmen, die auch heute noch (teils in verbesserter Form) zu den Standardwerkzeugen der algorithmischen Zahlentheorie gehören. In diesem Paragraphen sollen die beiden bekanntesten vorgestellt werden; außerdem möchte ich zumindest kurz auf mathematisch anspruchsvollere Verallgemeinerungen eingehen. Die hier behandelten Verfahren haben im Gegensatz zu denen des nächsten Paragraphen die Eigenschaft, daß sie umso schneller zum Erfolg führen, je kleiner die gesuchten Primfaktoren sind, daß sie allerdings sehr kleine Primfaktoren oft nicht finden. Sie sind also die Verfahren der Wahl für die Weiterverarbeitung eines durch Abdividieren erhaltenen „Rests“, von dem man weiß, daß er keine allzu kleinen Primfaktoren mehr hat.

JOHN M. POLLARD ist ein britischer Mathematiker, der hauptsächlich bei British Telecom arbeitete. Er veröffentlichte zwischen 1971 und 2000 rund zwanzig mathematische Arbeiten, größtenteils auf dem Gebiet der algorithmischen Zahlentheorie. Bekannt ist er auch für seine Beiträge zur Kryptographie, für die er 1999 den RSA Award erhielt. Außer den hier vorgestellten Faktorisierungsalgorithmen entwickelte er unter anderem auch das Zahlkörpersieb, eine Variante des weiter hinten vorgestellten quadratischen Siebs, dessen Weiterentwicklungen derzeit die schnellsten Faktorisierungsalgorithmen für große Zahlen sind. Seine home page, um die er sich auch jetzt im Ruhestand noch kümmert, ist `jmptidcott.googlepages.com/index.html`.

Bei den in diesem und dem nächsten Paragraphen vorgestellten Verfahren besteht das Ziel immer darin, *irgendeinen* Faktor zu finden; sobald dies erreicht ist, bricht das Verfahren ab und der gefundene Faktor sowie sein Kofaktor werden für sich weiter untersucht – wobei natürlich immer an erster Stelle ein Primzahltest stehen sollte.

a) Die Monte-Carlo-Methode

Monte Carlo ist ein Stadtteil von Monaco, der vor allem für seine Spielbank bekannt ist. An deren Spieltischen sollen Roulette-Schüsseln idealerweise rein zufällig für jedes Spiel von neuem eine Zahl zwischen 0 und 36 bestimmen.

Eine ähnliche Idee läßt sich auch für die Faktorisierung einer ganzen Zahl N verwenden: Ausgehend von einer Folge $(x_i)_{i\in\mathbb{N}}$ zufällig gewählter Zahlen zwischen 1 und N (oder 0 und $N-1$) bildet man jeweils den ggT von x_i mit N in der Hoffnung, einen nichttrivialen Teiler zu finden.

Für einen Primteiler p von N können wir erwarten, daß im Mittel jedes p-te x_i durch p teilbar ist und wir somit einen durch p teilbaren ggT mit N erhalten, der allerding möglicherweise auch noch andere Primteiler enthält.

Beim einfachen Abdividieren finden wir p, nachdem wir alle Primzahlen bis einschließlich p durchprobiert haben; wie wir im letzten Kapitel gesehen haben, sind dies etwa $p/\log p$ Stück. Für jede davon brauchen wir eine Division, verglichen mit den durchschnittlich $p/2$ EUKLIDischen Algorithmen bei der obigen Methode, die nicht einmal eine Garantie dafür bieten, den Faktor zu finden. Von daher hat die neue Methode zumindest in der bislang betrachteten Form ausschließlich Nachteile und ist keine sinnvolle Alternative zum Abdividieren.

POLLARDs Idee zur Beschleunigung beruht auf dem im Anhang genauer erklärten Geburtstagsparadoxon: Die Wahrscheinlichkeit dafür, daß eine gegebene Zufallszahl durch p teilbar ist, liegt zwar nur bei $1:p$, aber die Wahrscheinlichkeit, daß zwei der x_i modulo p gleich sind, steigt in der Nähe von etwa $\sqrt{p}$ Folgegliedern ziemlich steil von nahe null zu nahe eins. Wenn wir also anstelle der größten gemeinsamen Teiler von N mit den x_i die mit den Differenzen $x_i - x_j$ berechnen, haben wir bereits bei einer Folge der Länge um $\sqrt{p}$ gute Chancen, einen nichttrivialen ggT zu finden.

Auch in dieser Form ist das Verfahren noch nicht praktikabel: Wenn wir ein neues x_i mit $i \approx \sqrt{p}$ erzeugt haben, müssen wir für alle $j < i$ den ggT von $x_i - x_j$ berechnen, was noch einmal rund $\sqrt{p}$ Schritte sind, so

daß der Gesamtaufwand nicht proportional zu $\sqrt{p}$ ist, sondern eher zu

$$\int_0^{\sqrt{p}} x\,dx = \frac{p}{2}\,,$$

was keine große Ersparnis ist. Dazu kommt, daß alle bereits berechneten Folgeglieder gespeichert werden müssen, der Algorithmus hat also auch einen Platzbedarf in der Größenordnung $\sqrt{p}$.

Dieses Problem können wir umgehen, indem wir keine echten Zufallszahlen verwenden, sondern algorithmisch eine Folge sogenannter Pseudozufallszahlen erzeugen. Typischerweise verwendet man dazu eine Rekursionsvorschrift der Form $x_{i+1} = Q(x_i) \bmod N$ mit einem quadratischen Polynom Q. (Die bei Simulationen sehr beliebten Pseudozufallsgeneratoren nach der linearen Kongruenzmethode sind für die Monte-Carlo-Methode der Faktorisierung nicht geeignet.) Meist nimmt man einfach Polynome der Form $Q(x) = x^2 + c$, wobei allerdings $c \neq 0$ und $c \neq -2$ sein sollte, denn eine genauere Untersuchung zeigt, daß diese Wahlen keine guten Pseudozufallszahlen liefern. Daß die anderen Wahlen von c stets gute Generatoren liefern ist zwar nicht bewiesen, aber die praktischen Erfahrungen sind positiv.

Wegen der speziellen Form der Rekursion hängt die Restklasse von x_{i+1} modulo p nur ab von $x_i \bmod p$; insbesondere ist also $x_{i+1} \equiv x_{j+1} \bmod p$, falls $x_i \equiv x_j \bmod p$, und entsprechend stimmen auch für jedes $r \geq 0$ die Zahlen x_{i+r} und x_{j+r} modulo p überein, d.h. die Folge wird modulo p periodisch mit einer Periode π, die $|i-j|$ teilt.

Das Problem, Periodizität in einer Folge zu entdecken, tritt nicht nur in der Zahlentheorie auf, sondern beispielsweise auch in der Zeitreihenanalyse und anderen Anwendungen. Ein möglicher Algorithmus zu seiner Lösung, auch als Hase und Schildkröte Algorithmus bekannt, stammt von FLOYD (1967) und beruht auf folgender Beobachtung:

Wird eine Folge (y_i) irgendwann periodisch, so gibt es Indizes k derart, daß $y_k = y_{2k}$ ist.

In der Tat, ist $y_{i+\pi} = y_i$ für alle $i \geq r$, so können wir für k jedes Vielfache $\ell\pi$ der Periode nehmen, das mindestens gleich r ist.

ROBERT W. FLOYD (1936–2001) beendete seine Schulausbildung bereits im Alter von 14 Jahren, um dann mit einem Stipendium an der Universität von Chicago zu studieren, wo er mit 17 einen Bachelor in *liberal arts* bekam. Danach finanzierte er sich durch Arbeit ein zweites Bachelorstudium in Physik, das er 1958 abschloß. Damit war seine akademische Ausbildung beendet; er arbeitete als Operator in einem Rechenzentrum, brachte sich selbst Programmieren bei und begann einige Jahre später mit der Publikation wissenschaftlicher Arbeiten auf dem Gebiet der Informatik. Mit 27 wurde er Assistenzprofessor in Carnegie Mellon, fünf Jahre später erhielt er einen Lehrstuhl in Stanford. Zu den vielen Entwicklungen, die er initiierte, gehört die semantische Verifikation von Programmen, Design und Analyse von Algorithmen, Refactoring, dazu kommen Arbeiten über Graphentheorie und das FLOYD-STEINBERG dithering in der Computergraphik. 1978 erhielt er den TURING-Preis, die höchste Auszeichnung der Informatik. Stanfords Nachruf auf FLOYD ist zu finden unter `news-service.stanford.edu/news/2001/november7/floydobit-117.html` .

Damit sieht der Grobablauf der Monte-Carlo-Faktorisierung einer natürlichen Zahl N folgendermaßen aus:

Schritt 0: Man wähle ein quadratisches Polynom Q und einen Startwert x_0. Setze $x = y = x_0$.
Schritt $i, i > 0$**:** Ersetze x durch $Q(x)$ und y durch $Q\big(Q(y)\big)$; berechne dann ggT$(x - y, N)$. Falls dieser weder eins noch N ist, wurde ein Faktor gefunden.

Man beachte, daß hier im i-ten Schritt $x = x_i$ und $y = x_{2i}$ ist; wir erzeugen also die Folge der x_i (Schildkröte) und die der x_{2i} (Hase) simultan, ohne Zwischenergebnisse zu speichern.

Das Teuerste an diesem Algorithmus sind die EUKLIDischen Algorithmen zur ggT-Berechnung; da wir (sofern wir kleine Primfaktoren zuvor ausgeschlossen haben) nicht wirklich erwarten, daß hier häufig ein nichttriviales Ergebnis herauskommt, liegt es nahe, deren Anzahl möglichst zu reduzieren.

Eine Strategie dazu besteht darin, jeweils mehrere Differenzen $x_{2i} - x_i$ aufzumultiplizieren und dann erst für das Produkt den ggT des mit N zu

berechnen. Die „gewisse Anzahl“ darf nicht zu groß sein, denn sonst besteht die Gefahr, daß das Produkt nicht nur durch einen, sondern gleich durch mehrere Primteiler von N teilbar ist, es sollte aber aus Effizienzgründen auch nicht zu klein sein. Wenn alle kleinen Primteiler bereits ausgeschlossen sind, zeigt die Erfahrung, daß die Zusammenfassung von etwa hundert Differenzen ein guter Kompromiß ist; wenn bereits bei den „kleinen“ Faktoren mit einer hohen Suchgrenze gearbeitet wurde, bieten sich auch höhere Werte an.

Praktisch bedeutet das, daß wir eine neue Variable P einführen mit Anfangswert eins und dann im i-ten Schritt P durch $P \cdot (x - y) \bmod N$ ersetzen. Nur falls i durch die „gewisse Anzahl“ m teilbar ist, wird anschließend der ggT von N und P berechnet; andernfalls geht es gleich weiter mit dem $(i + 1)$-ten Schritt.

Die Monte-Carlo-Methode wird auch als ρ-Methode bezeichnet, da die Folge der X_i nicht von Anfang an periodisch sein muß. Sie muß aber, da es nur p Restklassen modulo p gibt, schließlich periodisch werden, d.h. sie beginnt auf dem unteren Ast des ρ und mündet irgendwann in den Kreis. Erfahrungsgemäs ist diese Methode sehr erfolgreich im Auffinden sechs- bis achtstelliger Faktoren; danach wird sie recht langsam, und kleine Faktoren kann sie oft nicht trennen.

Als Beispiel wollen wir die sechste FERMAT-Zahl $F_6 = 2^{64} + 1 = 18\,446\,744\,073\,709\,551\,617$ betrachten. Mit dem quadratischen Polynom $Q(x) = x^2 + 1$, dem Startwert $x_0 = 2$ und einem EUKLIDischen Algorithmus nach jeweils hundert Folgegliedern findet ein handelsüblicher PC nach 900 Iterationen in Sekundenbruchteilen den Faktor 274 177, der übrigens genau wie sein Kofaktor 67 280 421 310 721 prim ist. Damit ist F_6 vollständig faktorisiert.

Anhang: Das Geburtstagsparadoxon

Angenommen, in einem Raum befinden sich n Personen. Wie groß ist die Wahrscheinlichkeit dafür, daß zwei davon am gleichen Tag Geburtstag haben?

Um diese Frage wirklich beantworten zu können, müßte man die (recht inhomogene) Verteilung der Geburtstage über das Jahr kennen; wir

beschränken uns stattdessen auf ein grob vereinfachtes Modell ohne Schaltjahre mit 365 gleich wahrscheinlichen Geburtstagen. Dann ist die Wahrscheinlichkeit dafür, daß von n Personen keine zwei am gleichen Tag Geburtstage haben,

$$\prod_{k=0}^{n-1}\left(1-\frac{k}{365}\right),$$

denn für eine Person ist das überhaupt keine Bedingung, und jede weitere Person muß die Geburtstage der schon betrachteten Personen vermeiden. (Da der Faktor mit $k = 365$ verschwindet, wird die Wahrscheinlichkeit für $n > 365$ zu null, wie es nach dem DIRICHLETschen Schubfachprinzip auch sein muß.)

Nachrechnen ergibt für $n = 23$ ungefähr den Wert 0,4927; bei 23 Personen liegt also die Wahrscheinlichkeit für zwei gleiche Geburtstage bei 50,7%. Tatsächlich dürfte sie noch deutlich höher liegen, denn bei Geburtstagen ist die Annahme einer Gleichverteilung sicherlich falsch.

Bei einer guten Folge von Zufallszahlen sollten die Restklassen modulo p in sehr guter Näherung gleichverteilt sein; die Wahrscheinlichkeit dafür, daß unter n Zufallszahlen keine zwei in der gleichen Restklasse liegen, ist somit

$$P_n = \prod_{k=0}^{n-1}\left(1-\frac{k}{p}\right).$$

Da wir uns für einigermaßen große Werte von p interessieren (die kleinen haben wir schon abdividiert), können wir davon ausgehen, daß

$$\left(1-\frac{1}{p}\right)^p \approx e \quad \text{und} \quad \left(1-\frac{1}{p}\right) \approx e^{-1/p}$$

ist; für nicht zu große Werte von k ist dann auch

$$\left(1-\frac{k}{p}\right) \approx e^{-k/p},$$

und für nicht zu große Werte von n gilt

$$P_n = \prod_{k=0}^{n-1}\left(1-\frac{k}{p}\right) \approx \prod_{k=0}^{n-1} e^{-k/p} = e^{-\frac{1}{p}\sum_{k=0}^{n-1} k} = e^{-\frac{n(n-1)}{2p}}.$$

Für $p = 365$ etwa ergibt dies den Näherungswert $p_{23} \approx 0{,}499998$ für den korrekten Wert 0,4927.

Wenn wir im Exponenten noch den Term $n(n-1)$ durch n^2 approximieren, können wir abschätzen, für welches n die Wahrscheinlichkeit P_n einen vorgegebenen Wert erreicht:

$$e^{-\frac{n^2}{2p}} = P \iff \frac{n^2}{2p} = -\ln P \iff n = \sqrt{-2p\ln P}.$$

Damit liegt P_n bei etwa 50%, falls $n \approx \sqrt{2p\ln 2} \approx 1{,}177\sqrt{p}$ ist; für $p = 365$ ergibt dies die immer noch recht gute Näherung 22,494.

Für $P = 1/1000$ ergibt sich $n \approx 3{,}717\sqrt{p}$, für $P = 999/1000$ entsprechend $n \approx 0{,}0447\sqrt{p}$. Die Wahrscheinlichkeit dafür, daß es unter n Zufallszahlen zwei mit derselben Restklasse modulo p gibt, wechselt also bei der Größenordnung $n \approx \sqrt{p}$ von sehr unwahrscheinlich zu sehr wahrscheinlich.

b) Die ($p-1$)-Methode

POLLARDs zweite Methode beruht auf dem kleinen Satz von FERMAT: Für einen Primteiler p von N, ein Vielfaches r von $p-1$ und eine zu p teilerfremde natürliche Zahl a ist $a^r \equiv 1 \bmod p$; der ggT von $(a^r-1) \bmod N$ und N ist also durch p teilbar.

Natürlich ist $p-1$ nicht bekannt, wir können aber hoffen, daß $p-1$ nur durch vergleichsweise kleine Primzahlen teilbar ist. Sei etwa B eine Schranke mit der Eigenschaft, daß $p-1$ durch keine Primzahlpotenz größer B teilbar ist. Dann ist das Produkt r *aller* Primzahlpotenzen q^e, die höchstens gleich B sind, sicherlich ein Vielfaches von $p-1$, wenn auch ein extrem großes, das sich kaum mit realistischem Aufwand berechnen läßt. Für jedes konkrete a kann $a^r \bmod N$ jedoch verhältnismäßig einfach berechnet werden: Man potenziert einfach nacheinander für jede Primzahl $q \le B$ modulo N mit deren größter Potenz, die immer noch kleiner oder gleich B ist; mit dem Algorithmus zur modularen Exponentiation aus Kapitel 1 geht das auch für sechs- bis siebenstellige Werte von B noch recht flott.

Insgesamt funktioniert POLLARDs $(p-1)$-Methode zur Faktorisierung einer natürlichen Zahl N also folgendermaßen:

Schritt 0: Wähle eine Schranke B und eine Basis a zwischen 1 und N.

Schritt 1: Erstelle (z.B. nach ERATOSTHENES) eine Liste aller Primzahlen $q \le B$.

Schritt 2: Berechne für jede dieser Primzahlen q den größten Exponenten e derart, daß auch noch $q^e \le B$ ist, d.h. $e = [\log B / \log q]$. Ersetze dann den aktuellen Wert von a durch $a^{q^e} \bmod N$.

Schritt 3: Berechne ggT$(a-1, N)$. Falls ein Wert ungleich eins oder N gefunden wird, war das Verfahren erfolgreich, ansonsten nicht.

Es ist klar, daß der Erfolg dieses Verfahrens wesentlich davon abhängt, daß N einen Primteiler p hat mit der Eigenschaft, daß alle Primfaktoren von $p-1$ relativ klein sind. Ob dies der Fall ist, läßt sich im Voraus nicht sagen; die $(p-1)$-Methode liefert daher gelegentlich ziemlich schnell sogar 20- oder 30-stellige Faktoren, während sie andererseits deutlich kleinere Faktoren oft nicht findet.

Als Beispiel betrachten wir noch einmal $M_{67} = 2^{67}-1$. Wenn wir mit der Basis $a = 17$ und der Schranke $B = 3\,000$ arbeiten, wird a modulo M_{67} potenziert zum neuen

$$a = 111\,153\,665\,932\,902\,146\,348 \text{ mit } \text{ggT}(a-1, M_{67}) = 193\,707\,721\,.$$

Damit ist (in Sekundenbruchteilen auf einem Standard-PC) eine nichttriviale Faktorisierung gefunden, und ein Primzahltest zeigt, daß sowohl der gefundene Faktor als auch sein Komplement prim sind.

Warum die Methode Erfolg hatte, sehen wir an der Faktorisierung der um eins verminderten Faktoren:

$$193\,707\,720 = 2^3 \cdot 3^3 \cdot 5 \cdot 67 \cdot 2\,677 \quad \text{und}$$

$$761\,838\,257\,286 = 2 \cdot 3^2 \cdot 29 \cdot 67 \cdot 2\,551 \cdot 8\,539\,.$$

Für jede Schranke $B \ge 2\,677$ ist also der erste Faktor ein Teiler des endgültigen $a-1$, aber für $B < 8\,539$ ist der zweite Faktor keiner.

c) Varianten

Falls $p-1$ nicht nur relativ kleine Primfaktoren hat, führt die $(p-1)$-Methode nicht zum Erfolg. In solchen Fällen hat dann aber vielleicht $p+1$ oder irgendeine andere Zahl in der Nähe von p nur kleine Primfaktoren. In solchen Fällen können Varianten der $(p-1)$-Methode zum Erfolg führen.

Um diese Varianten zu definieren, empfiehlt es sich, zunächst die $(p-1)$-Methode etwas abstrakter unter gruppentheoretischen Gesichtspunkten zu betrachten.

Wir rechnen in der primen Restklassengruppe $(\mathbb{Z}/N)^\times$ und damit implizit auch in $(\mathbb{Z}/p)^\times$ für jeden Primteiler p von N – egal ob wir ihn kennen, oder nicht. In $(\mathbb{Z}/p)^\times$ ist für jedes Element a die $(p-1)$-te Potenz gleich dem Einselement; genau dasselbe gibt für jede r-te Potenz, für die der Exponent r ein Vielfaches von $(p-1)$ ist. Bei der $(p-1)$-Methode wird ein r berechnet, das durch alle Primzahlpotenzen bis zu einer gewissen Schranke teilbar ist; falls in der Primzerlegung von $p-1$ keine Primzahlpotenz oberhalb der Schranke liegt, ist r ein Vielfaches von $p-1$.

Allgemeiner können wir statt in $(\mathbb{Z}/N)^\times$ und $(\mathbb{Z}/p)^\times$ auch in einem anderen Paar von Gruppen rechnen: Wir gehen aus von einer endlichen Gruppe G_n, deren Elemente sich in irgendeiner Weise als r-tupel über $(\mathbb{Z}/N)$ auffassen lassen; außerdem nehmen wir an, daß sich die Gruppenmultiplikation für zwei so dargestellte Elemente auf Grundrechenarten über $\mathbb{Z}/N$ zurückführen läßt. Dann können wir die Elemente von G_n zu Tupeln über $\mathbb{Z}/p$ reduzieren und die Menge aller so erhaltenen Tupel bildet eine Gruppe G_p. Wieder ist jede Rechnung in G_n implizit auch eine Rechnung in G_p.

Die Elementanzahl von G_p sei $N(p)$.

Wir wählen irgendein Element von G_n und potenzieren es mit demselben Exponenten r, mit dem wir bei der $p-1$-Methode die Zahl a modulo N potenziert haben. Falls r ein Vielfaches von $N(p)$ ist, erhalten wir ein Element $b \in G_n$, dessen Reduktion modulo p das Einselement von G_p ist. Ist daher b_i die i-te Koordinate von b und e_i die von e, so muß die

Differenz $b_i - e_i$ durch p teilbar sein, und mit etwas Glück können wir p als ggT von n und $b_i - e_i$ bestimmen.

Bleibt nur noch das Problem, geeignete Gruppen zu finden. Bei der $(p-1)$-Methode ist $G_n = (\mathbb{Z}/n)^\times$ und $N(p) = p - 1$.

Für die $(p+1)$-Methode benutzt POLLARD die Tatsache, daß es nicht nur zu jeder Primzahl p, sondern auch zu jeder Primzahlpotenz p^r einen Körper mit entsprechender Elementanzahl. Dieser Körper $\mathbb{F}_{p^r}$ ist natürlich verschieden vom Ring $\mathbb{Z}/p^r$; er ist ein r-dimensionaler Vektorraum über $\mathbb{F}_p$ mit geeignet definierter Multiplikation.

Speziell für $r = 2$ hat der Körper $\mathbb{F}_{p^2}$ eine multiplikative Gruppe $\mathbb{F}_{p^2}^\times$ der Ordnung $p^2 - 1 = (p+1)(p-1)$. Sie hat $\mathbb{F}_p^\times$ als Untergruppe und die Faktorgruppe $G_p = \mathbb{F}_{p^2}^\times / \mathbb{F}_p^\times$ hat die Ordnung $N(p) = p+1$. Das Rechnen in dieser Gruppe mit Reprässentanten modulo N ist etwas trickreich und benutzt die hier nicht behandelten LUCAS-Sequenzen.

Derzeit am populärsten ist eine andere Wahl von G_n und G_p: Wir nehmen für G_n eine elliptische Kurve über $\mathbb{Z}/n$. Dabei handelt es sich um die Menge aller Punkte $(x, y) \in (\mathbb{Z}/n)^2$, die einer vorgegebenen Gleichung $y^2 = x^3 - ax - b$ genügen, wobei a, b Elemente von $\mathbb{Z}_n$ sind, für die $\Delta = 4a^3 - 27b^2$ teilerfremd zu n ist; dazu kommt ein weiterer Punkt O, den wir formal als $(0, \infty)$ schreiben. G_p ist dann die entsprechende Punktmenge in $\mathbb{F}_p^2$ zusammen mit O. Nach einem Satz von HELMUT HASSE (1898–1979) ist

$$p + 1 - 2\sqrt{p} < N(p) < p + 1 + 2\sqrt{p}\,,$$

und wie man inzwischen weiß, kann man auch für jeden Wert, der diese Ungleichung erfüllt, Parameterwerte a und b finden, so daß $N(p)$ gleich diesem Wert ist. Wenn man mit hinreichend vielen verschiedenen Kurven arbeitet, ist daher die Chance recht groß, daß der Exponent r wenigstens für eine davon ein Vielfaches von $N(p)$ ist.

Die Multiplikation ist folgendermaßen definiert: Durch zwei Punkte (x_1, y_1) und (x_2, y_2) auf der Kurve geht genau eine Gerade; setzt man deren Gleichung $y = mx + c$ in die Kurvengleichung ein, erhält man ein Polynom dritten Grades in x. Dieses hat natürlich die beiden Nullstellen

x_1, x_2, und daneben noch eine dritte Nullstelle x_3. Der dritte Schnittpunkt der Geraden mit der Kurve ist somit $(x_3, mx_3 + c)$; als Summe der beiden Punkte definiert man aber

$$(x_1, y_1) \oplus (x_2, y_2) = \big(x_3, -(mx_3 + c)\big)\,.$$

Man kann zeigen, daß dies die Menge der Kurvenpunkte zu einer Gruppe mit Neutralelement O macht, in der man genauso vorgehen kann wie bei der klassischen $(p-1)$-Methode.

Unter den Faktorisierungsmethoden, deren Rechenzeit von der Größe des zu findenden Faktors abhängt, ist die Faktorisierung mit elliptischen Kurven die für große Zahlen derzeit beste bekannte Methode; sie fand schon Faktoren mit bis zu 67 Stellen. Produkte zweier ungefähr gleich großer Primzahlen wie beispielsweise RSA-Moduln sind für solche Methoden allerdings der schlechteste Fall; hierfür sind andere Methoden, deren Aufwand nur von der Größe der zu faktorisierenden Zahl abhängt, meist besser geeignet.

§ 3: Das Verfahren von Fermat und seine Varianten

Die bisher betrachteten Verfahren funktionieren vor allem dann gut, wenn die zu faktorisierende Zahl mindestens einen relativ kleinen Primteiler hat. Das hier beschriebene Verfahren von FERMAT führt genau dann schnell ans Ziel, wenn sie sich als Produkt zweier fast gleich großer Faktoren schreiben läßt. In seiner einfachsten Form beruht es auf der dritten binomischen Formel $x^2 - y^2 = (x+y)(x-y)$: Ist $N = pq$ Produkt zweier ungerader Primzahlen, so ist

$$N = (x+y)(x-y) \quad \text{mit} \quad x = \frac{p+q}{2} \quad \text{und} \quad y = \frac{p-q}{2}\,;$$

Ausmultiplizieren führt auf die Beziehung $N + y^2 = x^2$.

FERMAT berechnet für $y = 0, 1, 2, \ldots$ die Zahlen $N + y^2$; falls er auf ein Quadrat x^2 stößt, hat er zwei Faktoren $x \pm y$ gefunden. Da y gleich der halben Differenz der beiden Faktoren ist, kommt er umso schneller ans Ziel, je näher die beiden Faktoren beieinander liegen; dies erklärt die

Vorschrift der Bundesnetzagentur, daß die beiden Faktoren eines RSA-Moduls zwar ungefähr gleich groß sein sollten, daß sie aber doch einen gewissen Mindestabstand einhalten müssen.

Anstelle der Zahlen $N + y^2$ kann man auch für ein festes k die Zahlen $kN + y^2$ betrachten. Falls dies eine Quadratzahl x^2 ist, gilt entsprechend

$$kN = x^2 - y^2 = (x + y)(x - y)\,,$$

und wenn man Glück hat, sind ggT$(x \pm y, N)$ echte Faktoren von N. Wenn man Pech hat, sind es freilich einfach die beiden Zahlen eins und N. Trotzdem lassen sich darauf sehr effiziente Faktorisierungsverfahren aufbauen, denn die obige Gleichung besagt ja auch, daß wir nur *irgendwie* zwei Zahlen x, y finden müssen mit $x^2 \equiv y^2 \bmod N$ und dann eine Chance haben, daß ggT$(x \pm y, N)$ uns zwei Faktoren von N liefert. Je mehr solche Paare (x, y) wir finden, desto größer sind die Erfolgschancen.

Der Grundalgorithmus zum Finden solcher Paare ist das sogenannte quadratische Sieb, mit dem wir uns als nächstes beschäftigen wollen.

In seiner einfachsten Variante wählen wir uns ein quadratisches Polynom, z.B. das Polynom

$$f(x) = \left(x + \left\lceil\sqrt{N}\right\rceil\right)^2 - N\,.$$

Für jedes x ist dann $f(x) \equiv \left(x + \left\lceil\sqrt{N}\right\rceil\right)^2 \bmod N$, wobei links und rechts verschiedene Zahlen stehen. Insbesondere steht links im allgemeinen keine Quadratzahl.

Falls wir allerdings Werte $x_1, x_2, \ldots, x_r$ finden können, für die das Produkt der $f(x_i)$ eine Quadratzahl ist, dann ist

$$\prod_{i=1}^{r} f(x_i) \equiv \prod_{i=1}^{r} \left(x + \left\lceil\sqrt{N}\right\rceil\right)^2 \bmod N$$

eine Relation der gesuchten Art.

Diese Strategie erklärt die Wahl des Polynoms f: Wir betrachten sowohl die Zahlen $x + \left\lceil\sqrt{N}\right\rceil$ als auch

$$f(x) = \left(x + \left\lceil\sqrt{N}\right\rceil\right)^2 - N = x^2 + 2x\left\lceil\sqrt{N}\right\rceil + \left\lceil\sqrt{N}\right\rceil^2 - N\,.$$

Für x-Werte, die deutlich kleiner als $\sqrt{N}$ sind (und nur mit solchen werden wir es im folgenden zu tun haben) liegen beide Zahlen in der Größenordnung $\sqrt{N}$; bei den meisten anderen Polynomen, die ein Quadrat minus N produzieren, wäre entweder die zu quadrierende Zahl oder deren Funktionswert deutlich größer. Natürlich kann anstelle von $\left\lceil\sqrt{N}\right\rceil$ genauso gut eine andere Zahl ähnlicher Größenordnung verwendet werden; es kommt nur darauf an, daß ihr Quadrat in der Nähe von N liegt.

Um geeignete x_i zu finden, betrachten wir eine Menge $\mathcal{B}$ von Primzahlen, die sogenannte Faktorbasis. Typischerweise enthält $\mathcal{B}$ für die Faktorisierung einer etwa hundertstelligen Zahl zwischen 100 und 120 Tausend Primzahlen, deren größte somit, wie die folgende Tabelle zeigt, im einstelligen Millionenbereich liegt.

n	n-te Primzahl	n	n-te Primzahl
100 000	1 299 709	600 000	8 960 453
200 000	2 750 159	700 000	10 570 841
300 000	4 256 233	800 000	12 195 257
400 000	5 800 079	900 000	13 834 103
500 000	7 368 787	1 000 000	15 485 863

Beim quadratischen Sieb interessieren nur x-Werte, für die $f(x)$ als Produkt von Primzahlen aus $\mathcal{B}$ (und eventuell auch Potenzen davon) darstellbar ist. Ist $f(x_i) = \prod_{p\in\mathcal{B}} p^{e_{ip}}$, so ist

$$\prod_{i=1}^{r} f(x_i)^{\varepsilon_i} = \prod_{p\in\mathcal{B}} p^{\sum_{i=1}^{r} e_{ip}\varepsilon_i}$$

genau dann ein Quadrat, wenn $\sum_{i=1}^{r} e_{ip}\varepsilon_i$ für alle $p \in \mathcal{B}$ gerade ist. Dies hängt natürlich nur ab von den ε_i mod 2 und den e_{ip} mod 2; wir können ε_i und e_{ip} daher als Elemente des Körpers mit zwei Elementen

auffassen und bekommen dann über $\mathbb{F}_2$ die Bedingungen

$$\sum_{i=1}^{r} e_{ip}\varepsilon_i = 0 \quad \text{für alle } p \in \mathcal{B}\,.$$

Betrachten wir die ε_i als Variablen, ist dies ein homogenes lineares Gleichungssystem in r Variablen mit soviel Gleichungen, wie es Primzahlen in der Faktorbasis gibt. Dieses Gleichungssystem hat nichttriviale Lösungen, falls die Anzahl der Variablen die der Gleichungen übersteigt, falls es also mehr Zahlen x_i gibt, für die $f(x_i)$ über der Faktorbasis faktorisiert werden kann, als Primzahlen in der Faktorbasis.

Für jede nichttriviale Lösung ist

$$\prod_{i=1}^{r} f(x_i)^{\varepsilon_i} = \prod_{i=1}^{r} \left(x + \left\lfloor\sqrt{N}\right\rfloor\right)^{2\varepsilon_i} \bmod N$$

eine Relation der Form $x^2 \equiv y^2 \bmod N$, die mit einer Wahrscheinlichkeit von etwa ein halb zu einer Faktorisierung von N führt. Falls wir zehn linear unabhängige Lösungen des Gleichungssystems betrachten, führt also mit einer Wahrscheinlichkeit von etwa 99,9% mindestens eine davon zu einer Faktorisierung.

Da ε_i nur die Werte 0 und 1 annimmt, stehen in obigem Produkt natürlich keine echten Potenzen: Man multipliziert einfach nur die Faktoren miteinander, für die $\varepsilon_i = 1$ ist. Außerdem interessieren nicht die links- und rechtsstehenden Quadrate, sondern deren Quadratwurzeln; tatsächlich also berechnet man (hier natürlich in $\mathbb{N}_0$)

$$x = \prod_{p\in\mathcal{B}} p^{\frac{1}{2}\sum_{i=1}^{r} \varepsilon_i e_{ip}} \bmod N \quad \text{und} \quad y = \prod_{i=1}^{r} \left(x + \left\lfloor\sqrt{N}\right\rfloor\right)^{\varepsilon_i} \bmod N\,.$$

Zum besseren Verständnis des Grundprinzips wollen wir versuchen, damit die Zahl 15 zu faktorisieren. Dies ist zwar eine sehr untypische Anwendung, da das quadratische Sieb üblicherweise erst für mindestens etwa vierzigstellige Zahlen angewandt wird, aber zumindestens das Prinzip sollte auch damit klarwerden.

Als Faktorbasis verwenden wir die Menge

$$\mathcal{B} = \{2, 3, 7, 11\}\,;$$

die Primzahl fünf fehlt, da $3 \cdot 5 = 15$ ist und daher bei einer Faktorbasis, die sowohl drei als auch fünf enthält, die Gefahr zu groß ist, daß die linke wie auch die rechte Seite der Kongruenz durch fünfzehn teilbar ist. Bei realistischen Anwendungen muß man auf solche Überlegungen keine Rücksicht nehmen, denn dann sind die Elemente der Faktorbasis höchstens siebenstellig und somit erheblich kleiner als die gesuchten Faktoren.

Wir berechnen $f(x)$ für $x = 1, 2, \ldots$, bis wir einige Funktionswerte haben, die über der Faktorbasis faktorisiert werden können. Die faktorisierbaren Werte sind in folgender Tabelle zusammengestellt:

x	$x + \left\lfloor\sqrt{N}\right\rfloor$	$f(x)$	*Faktorisierung*
1	4	1	
3	6	21	$3 \cdot 7$
5	8	49	7^2
6	9	66	$2 \cdot 3 \cdot 11$
10	13	154	$2 \cdot 7 \cdot 11$
54	57	3234	$2 \cdot 3 \cdot 7^2 \cdot 11$

Die erste und die dritte Zeile sind selbst schon Relationen der gesuchten Art, nämlich

$$4^2 \equiv 1 \bmod 15 \quad \text{und} \quad 8^2 \equiv 7^2 \bmod 15\,.$$

Die zweite Relation ist nutzlos, denn $8 - 7 = 1$ und $8 + 7 = 15$. Die erste dagegen führt zur Faktorisierung, denn

$$\text{ggT}(4 + 1, 15) = 5 \quad \text{und} \quad \text{ggT}(4 - 1, 15) = 3\,.$$

Da dies aber ein Zufall ist, der bei großen Werten von N so gut wie nie vorkommt, wollen wir das ignorieren und mit den Relationen zu $x = 3, 6, 10$ und 51 arbeiten:

$$\begin{aligned} 6^2 &\equiv 3 \cdot 7 && \bmod 15 \\ 9^2 &\equiv 2 \cdot 3 \cdot 11 && \bmod 15 \\ 13^2 &\equiv 2 \cdot 7 \cdot 11 && \bmod 15 \\ 57^2 &\equiv 2 \cdot 3 \cdot 7^2 \cdot 11 && \bmod 15 \end{aligned}$$

Multipliziert man die ersten drei dieser Relationen miteinander, folgt

$$(6 \cdot 9 \cdot 13)^2 \equiv (2 \cdot 3 \cdot 7 \cdot 11)^2 \bmod 15$$

oder $702^2 \equiv 462^2 \bmod 15$. Da

$$\text{ggT}(702 - 462, 15) = \text{ggT}(240, 15) = 15$$

ist, bringt das leider nichts.

Wir erhalten auch dann rechts ein Quadrat, wenn wir das Produkt der ersten, dritten und vierten Relation bilden; dies führt auf

$$(6 \cdot 13 \cdot 57)^2 \equiv (2 \cdot 3 \cdot 7^2 \cdot 11)^2 \bmod 15$$

oder $4446^2 \equiv 3234^2 \bmod 15$. Hier ist

$$\text{ggT}(4446 - 3234, 15) = \text{ggT}(1212, 15) = 3\,,$$

womit wir die Zahl 15 faktorisiert haben – wenn auch nicht unbedingt auf die einfachstmögliche Weise.

Bei realistischen Beispielen sind die Funktionswerte $f(x)$ deutlich größer als die Primzahlen aus der Faktorbasis; außerdem liegen die vollständig faktorisierbaren Zahlen viel dünner als hier: Bei der Faktorisierung einer hundertstelligen Zahl etwa muß man davon ausgehen, daß nur etwa jeder 10^9-te Funktionswert über der Faktorbasis zerfällt.

Daher ist es wichtig, ein Verfahren zu finden, mit dem diese wenigen Funktionswerte schnell und einfach bestimmt werden können. Das ist zum Glück möglich:

Der Funktionswert $f(x)$ ist genau dann durch p teilbar, wenn

$$f(x) \equiv 0 \bmod p$$

ist. Für ein Polynom f mit ganzzahligen Koeffizienten ist offensichtlich $f(x) \equiv f(y) \bmod p$, falls $x \equiv y \bmod p$ ist. Daher ist für ein x mit $f(x) \equiv 0 \bmod p$ auch

$$f(x + kp) \equiv 0 \bmod p \quad \text{für alle } k \in \mathbb{Z}\,.$$

Es genügt daher, im Bereich $0 \le x < p - 1$ nach Werten zu suchen, für die $f(x)$ durch p teilbar ist.

Dazu kann man f auch als Polynom über dem Körper mit p Elementen betrachten und nach Nullstellen in diesem Körper suchen. Für Polynome

großen Grades und große Werte von p kann dies recht aufwendig sein; hier, bei einem quadratischen Polynom, müssen wir natürlich einfach eine quadratische Gleichung lösen: In $\mathbb{F}_p$ wie in jedem anderen Körper auch gilt

$$f(x) = \left(x + \left\lceil\sqrt{N}\right\rceil\right)^2 - N = 0 \iff \left(x + \left\lceil\sqrt{N}\right\rceil\right)^2 = N\,,$$

und diese Gleichung ist genau dann lösbar, wenn es ein Element $w \in \mathbb{F}_p$ gibt mit Quadrat N, wenn also in $\mathbb{Z}$

$$w^2 \equiv N \bmod p$$

ist. Für $p > 2$ hat $f(x) = 0$ in $\mathbb{F}_p$ dann die beiden Nullstellen

$$x = -\left\lceil\sqrt{N}\right\rceil \pm w\,;$$

andernfalls gibt es keine Lösung.

Insbesondere kann also $f(x)$ nur dann durch p teilbar sein, wenn N modulo p ein Quadrat ist; dies ist für etwa die Hälfte aller Primzahlen der Fall. Offensichtlich sind alle anderen Primzahlen nutzlos, und sollten daher gar nicht erst in die Faktorbasis aufgenmmen werden.

Im Kapitel über quadratische Reste werden wir sehen, daß sich auch für große N und p leicht und schnell entscheiden läßt, ob N modulo p ein Quadrat ist; der Aufwand entspricht ungefähr dem eines auf N und p angewandten EUKLIDischen Algorithmus. Wir werden dort auch sehen, daß sind für solche Quadrate relativ schnell die beiden Quadratwurzeln modulo p berechnen lassen.

Das eigentliche Sieben zum Auffinden der komplett über der Faktorbasis zerlegbaren Funktionswerte $f(x)$ geht dann folgendermaßen vor sich: Man legt ein Siebintervall $x = 0, 1, \ldots, M$ fest und speichert in einem Feld der Länge $M + 1$ für jedes x eine Approximation von $\log_2 |f(x)|$.

Für jede Primzahl p aus der Faktorbasis berechnet man dann die beiden Nullstellen $x_{1/2}$ von f modulo p im Intervall von 0 bis $p - 1$ und subtrahiert von jedem Feldelement mit Index der Form $x_1 + kp$ oder $x_2 + kp$ eine Approximation von $\log_2 p$.

Falls $f(x)$ über der Faktorbasis komplett faktorisierbar ist, sollte dann am Ende der entsprechende Feldeintrag bis auf Rundungsfehler gleich

null sein; um keine Fehler zu machen, untersucht man daher für alle Feldelemente, die betragsmäßig unterhalb einer gewissen Grenze liegen, durch Abdividieren, ob sie wirklich komplett faktorisieren, und man bestimmt auf diese Weise auch *wie* sie faktorisieren. Damit läßt sich dann das oben erwähnte Gleichungssystem über $\mathbb{F}_2$ aufstellen und, falls genügend viele Relationen gefunden sind, nichttrivial so lösen, daß eine der daraus resultierenden Gleichungen $x^2 \equiv y^2 \bmod p$ zu einer nichttrivialen Faktorisierung von N führt.

Als zwar immer noch untypisch kleines Beispiel, das besser und schneller durch Abdividieren faktorisiert werden könnte, betrachten wir die Zahl $N = 5\,352\,499$. Wir nehmen als Faktorbasis alle Primzahlen kleiner hundert modulo derer N ein Quadrat ist; Nachrechnen zeigt, daß

$$\mathcal{B} = \{3, 5, 11, 13, 17, 19, 23, 31, 41, 43, 53, 59, 83, 89\}$$

dann 14 Elemente enthält. Für jedes davon müssen wir die quadratische Gleichung $f(x) \equiv 0 \bmod p$ lösen, was hier natürlich selbst durch Ausprobieren recht schnell möglich wäre. Die Lösungsmengen sind

$p =$	3	5	11	13	17	19	23
Lösungen	$\{1,2\}$	$\{0,4\}$	$\{0,5\}$	$\{4,11\}$	$\{6,9\}$	$\{2,8\}$	$\{0,20\}$
$p =$	31	41	43	53	59	83	89
Lösungen	$\{27,28\}$	$\{3,4\}$	$\{26,35\}$	$\{39,52\}$	$\{2,33\}$	$\{35,70\}$	$\{23,68\}$

Wenn wir damit das Intervall der natürlichen Zahlen von 1 bis 20 000 sieben, erhalten wir 18 Zahlen, die über der Faktorbasis komplett zerfallen:

i	x_i	$f(x_i)$	*Faktorisierung*
1	23	104397	$3 \cdot 17 \cdot 23 \cdot 89$
2	121	571857	$3 \cdot 11 \cdot 13 \cdot 31 \cdot 43$
3	533	2747217	$3 \cdot 11 \cdot 17 \cdot 59 \cdot 83$
4	635	3338205	$3 \cdot 5 \cdot 13 \cdot 17 \cdot 19 \cdot 53$
5	741	3974417	$31 \cdot 41 \cdot 53 \cdot 59$
6	895	4938765	$3 \cdot 5 \cdot 13 \cdot 19 \cdot 31 \cdot 43$
7	2013	13361777	$11 \cdot 13 \cdot 41 \cdot 43 \cdot 53$
8	2185	14879505	$3 \cdot 5 \cdot 17 \cdot 23 \cdot 43 \cdot 59$
9	2477	17591601	$3 \cdot 31 \cdot 43 \cdot 53 \cdot 83$

10	2649	19268945	$5 \cdot 19 \cdot 43 \cdot 53 \cdot 89$
11	4163	36586077	$3 \cdot 11 \cdot 19 \cdot 23 \cdot 43 \cdot 59$
12	4801	45256497	$3 \cdot 11 \cdot 13 \cdot 31 \cdot 41 \cdot 83$
13	5497	55643601	$3 \cdot 13 \cdot 17 \cdot 23 \cdot 41 \cdot 89$
14	6253	68023857	$3 \cdot 11 \cdot 19 \cdot 23 \cdot 53 \cdot 89$
15	10991	171643917	$3 \cdot 17 \cdot 23 \cdot 41 \cdot 43 \cdot 83$
16	11275	179281245	$3 \cdot 5 \cdot 11 \cdot 13 \cdot 19 \cdot 53 \cdot 83$
17	14575	279852045	$3 \cdot 5 \cdot 11 \cdot 17 \cdot 19 \cdot 59 \cdot 89$
18	18535	429286605	$3 \cdot 5 \cdot 11 \cdot 23 \cdot 31 \cdot 41 \cdot 89$

Ein Vektor $\vec{\varepsilon} \in \mathbb{F}_2^{18}$, für den $\prod f(x_i)^{\varepsilon_i}$ ein Quadrat ist, löst daher über $\mathbb{F}_2$ das lineare Gleichungssystem mit Matrix

$$\begin{pmatrix}
1&1&1&1&0&1&0&1&1&0&1&1&1&1&1&1&1&1\\
0&0&0&1&0&1&0&1&0&1&0&0&0&0&0&1&1&1\\
0&1&1&0&0&0&1&0&0&0&1&1&0&1&0&1&1&1\\
0&1&0&1&0&1&1&0&0&0&0&1&1&0&0&1&0&0\\
1&0&1&1&0&0&0&1&0&0&0&0&1&0&1&0&1&0\\
0&0&0&1&0&1&0&0&0&1&1&0&0&1&0&1&1&0\\
1&0&0&0&0&0&0&1&0&0&1&0&1&1&1&0&0&1\\
0&1&0&0&1&1&0&0&1&0&0&1&0&0&0&0&0&1\\
0&0&0&0&1&0&1&0&0&0&0&1&1&0&1&0&0&1\\
0&1&0&0&0&1&1&1&1&1&1&0&0&0&1&0&0&0\\
0&0&0&1&1&0&1&0&1&1&0&0&0&1&0&1&0&0\\
0&0&1&0&1&0&0&1&0&0&1&0&0&0&0&0&1&0\\
0&0&1&0&0&0&0&0&1&0&0&1&0&0&1&1&0&0\\
1&0&0&0&0&0&0&0&0&1&0&0&1&1&0&0&1&1
\end{pmatrix}.$$

Der GAUSS-Algorithmus führt auf die Lösungen

$$(\sigma, \mu+\rho, \mu+\rho, \mu+\rho, \sigma+\rho+\tau, \nu+\sigma, \lambda+\rho, \sigma,$$
$$\nu+\mu+\lambda, \nu+\sigma+\mu+\tau, \mu+\sigma+\tau, \lambda, \nu+\sigma, \mu, \nu, \rho, \sigma, \tau)$$

mit sechs freien Parametern $\lambda, \mu, \nu, \rho, \sigma, \tau \in \mathbb{F}_2$. Setzen wir zunächst $\lambda = \mu = \sigma = 1$, $\nu = \rho = \tau = 0$, so erhalten wir die Lösung

$$\vec{\varepsilon} = (1,1,1,1,1,1,1,1,0,0,0,1,1,1,0,0,1,0)\,.$$

Sie führt auf die beiden Zahlen

$$x = \prod_{p\in\mathcal{B}} p^{\frac{1}{2}\sum_{i=1}^{r}\varepsilon_i e_{ip}} \bmod N = 854\,237 \quad \text{und}$$

$$y = \prod_{i=1}^{r}\left(x + \left\lfloor\sqrt{N}\right\rfloor\right)^{\varepsilon_i} \bmod N = 3\,827\,016\,.$$

Leider ist die Differenz dieser beiden Zahlen teilerfremd zu N.

Setzen wir in einem zweiten Versuch $\nu = 1$ statt $\nu = 0$, so erhalten wir die weitere Lösung

$$\vec{\varepsilon} = (1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 0, 1, 0)\,,$$

die uns die Zahlen

$$x = \prod_{p\in\mathcal{B}} p^{\frac{1}{2}\sum_{i=1}^{r}\varepsilon_i e_{ip}} \bmod N = 1\,020\,903 \quad \text{und}$$

$$y = \prod_{i=1}^{r}\left(x + \left\lfloor\sqrt{N}\right\rfloor\right)^{\varepsilon_i} \bmod N = 4\,093\,611$$

liefert. Nun ist der ggT der Differenz mit N gleich 1 237, womit wir die Faktorisierung

$$5\,352\,499 = 1237 \times 4327$$

gefunden haben. In diesem Fall sind die beiden Faktoren sogar bereits Primzahlen; das wird natürlich im allgemeinen nicht der Fall sein – es sei denn, man startet wie hier mit dem Produkt zweier Primzahlen.

Natürlich hätte uns jede der bisher behandelten Methoden dieses Ergebnis mit erheblich geringerem Aufwand und auch erheblich schneller geliefert; das quadratische Sieb entwickelt seine Stärken erst bei erheblich größeren Zahlen, für die es dann oft tagelang rechnet.

Dabei verwendet man das quadratische Sieb meist nicht in der hier vorgestellten Einfachstsversion, sondern mit verschiedenen Optimierungen.

Bei realistischen Anwendungen wird der überwiegende Teil der Rechenzeit für das Sieben gebraucht. Dies läßt sich relativ einfach parallelisieren, indem man das Sieben für verschiedene Teilintervalle auf verschiedene Computer verteilt. Auf diese Weise können mehrere Tausend

Computer jeweils ein Teilintervall sieben und anschließend die gefundenen Faktorisierungen an eine Zentrale melden. Sobald genügend viele eingegangen sind, kann diese ein lineares Gleichungssystem aufstellen und dieses lösen.

Eine weitere Verbesserung, die zu kürzeren Suchintervallen und damit auch kleineren Zahlen führt, besteht darin, anstelle des einen Polynoms f mehrere Polynome zu verwenden. Auch diese können wieder auf verschiedene Computer verteilt werden. Falls einige der Polynome auch negative Werte annehmen können, muß auch das berücksichtigt werden, indem man bei der Faktorisierung der nach dem Sieben übrig gebliebenen Zahlen auch noch die -1 als zusätzliche „Primzahl“ in die Faktorbasis aufnimmt.

Ab etwa 120 bis 130 Stellen wird eine Variante schneller, bei der auch mit komplizierteren als nur quadratischen Polynomen gearbeitet wird, das sogenannte Zahlkörpersieb. Es hat seinen Namen daher, daß die dahinterstehende Theorie mit algebraischen Zahlkörpern arbeitet; konkret gerechnet wird allerdings weiterhin mit ganzen Zahlen. Dieses Zahlkörpersieb ist die derzeit beste bekannte Methode zur Faktorisierung von Zahlen, die Produkte zweier Primzahlen ähnlicher Größenordnung sind; von diesem Verfahren geht also die größte Gefahr für RSA aus. Der derzeitige Rekord für dieses Verfahren ist die Faktorisierung einer zweihundertstelligen *challenge number* der Firma RSA.