

# Kapitel 1

## Polynomwahl beim Zahlkörpersieb

Wir diskutieren Ansätze zur Wahl der Polynome beim Zahlkörpersieb. Wir wiederholen zunächst die schematische Darstellung des Zahlkörpersiebs, und stellen dann die Modifikationen vor. In Abschnitt 1.2 präsentieren wir einen theoretischen Ansatz, der in der Praxis allerdings dem Algorithmus in Abschnitt 1.3 unterliegt.

In diesem gesamten Abschnitt gelte zur Vereinfachung der Darstellung, dass für jede algebraische Zahl  $\alpha$  der Ring  $\mathbb{Z}[\alpha]$  nicht nur im Ring  $\mathbb{Z}_{\mathbb{Q}(\alpha)}$  der ganzen algebraischen Zahlen in  $\mathbb{Q}(\alpha)$  enthalten ist, sondern es gelte sogar Gleichheit. Geben wir ferner Laufzeiten vom Zahlkörpersieb oder Varianten an, so ist stets die vermutete Laufzeit gemeint.

### 1.1 Schematische Darstellung des Zahlkörpersiebs

Das allgemeine Zahlkörpersieb (GNFS) erhält als Eingabe eine Zahl  $n$  und soll einen nicht-trivialen Teiler von  $n$  ausgeben. Der Algorithmus und seine Laufzeit werden parametrisiert durch Schranken  $B_1, B_2, E$  für den Suchraum. Durch geeignete Wahl dieser Schranken erreicht man eine Laufzeit von

$$L_n\left[\frac{1}{3}, 1.923\right] = e^{(1.923+o(1))(\log n)^{1/3}(\log \log n)^{2/3}}$$

Im Unterschied dazu erreicht man mit dem speziellen Zahlkörpersieb (SNFS) für Zahlen der Form  $n = r^e - s$  für kleine  $r, |s|$  und großes  $e$  eine Laufzeitschranke von

$$L_n\left[\frac{1}{3}, 1.527\right] = e^{(1.527+o(1))(\log n)^{1/3}(\log \log n)^{2/3}}$$

Der Geschwindigkeitsvorteil wird u.a. dadurch erreicht, dass das Minimalpolynom  $f$  von  $\alpha$  in diesem Fall kleine Koeffizienten hat [1]. Deswegen erscheint auch der Versuch erfolgversprechend, durch eine verbesserte Polynomwahl die Laufzeit des GNFS zu verkleinern.

Algorithmus GNFS

**Schritt 1: Polynomwahl**

Wähle ein monisches, irreduzibles Polynom  $f \in \mathbb{Z}[x]$  vom Grad  $d$ , das Minimalpolynom von  $\alpha$  ist und eine "kleine" Nullstelle  $m$  modulo  $n$  hat:  $f(m) \equiv 0 \pmod{n}$ . Wähle beispielsweise die  $m$ -äre Darstellung von  $n$  mit  $m \approx n^{1/d}$ :

$$n = \sum_{i=0}^d c_i m^i, \quad f(x) = \sum_{i=0}^d c_i x^i$$

Durch geeignete Wahl von  $d$  wird erreicht, dass der führende Koeffizient 1 ist.

Die Nullstelle  $\alpha$  von  $f$  definiert einen Ringhomomorphismus

$$\varphi: \mathbb{Z}[\alpha] \rightarrow \mathbb{Z}_n, \quad \varphi\left(\sum s_i \alpha^i\right) \equiv \sum s_i m^i \pmod{n}$$

Insbesondere ist  $\varphi(a + b\alpha) \equiv a + bm \pmod{n}$ .

**Schritt 2: Sieb**

Suche Paare  $(a, b) \in \mathbb{Z}^2$  mit  $|a|, |b| < E$ ,  $\text{ggT}(a, b) = 1$ , so dass  $a + bm$  einerseits  $B_1$ -glatt und die Norm von  $N(a + b\alpha) = (-b)^d f(-\frac{a}{b})$  andererseits  $B_2$ -glatt ist.

**Schritt 3: Gleichungssystem**

Suche eine Teilmenge  $S$  der Paare  $(a, b)$  von Schritt 2, so dass

$$\prod_{(a,b) \in S} (a + bm) \quad \text{und} \quad \prod_{(a,b) \in S} (a + b\alpha)$$

Quadrate in  $\mathbb{Z}$  bzw.  $\mathbb{Z}[\alpha]$  sind, und sich eine Wurzel  $x$  bzw.  $\beta$  in der Faktorbasis  $p_1, p_2, \dots$  für  $\mathbb{Z}$  bzw.  $\pi_1, \pi_2, \dots$  für  $\mathbb{Z}[\alpha]$  darstellen läßt.

**Schritt 4: Wurzelberechnung**

Sei  $\varphi(\beta) \equiv y \pmod{n}$ . Wegen  $\varphi(a + b\alpha) \equiv a + bm \pmod{n}$  gilt  $\varphi(\beta^2) \equiv y^2 \equiv x^2 \pmod{n}$ . Versuche  $n$  durch die Berechnung von  $\text{ggT}(x \pm y, n)$  zu faktorisieren.

Ist das Polynom  $f$  aus Schritt 1 nicht irreduzibel, also  $f = gh$ , dann kann wegen  $f(m) \equiv g(m)h(m) \equiv 0 \pmod{n}$  bereits einer der Faktoren  $g(m), h(m)$  einen nicht-trivialen Teiler von  $n$  liefern. Wir können daher vereinfachend davon ausgehen, dass ein beliebig gewähltes Polynom  $f$  tatsächlich irreduzibel ist.

## 1.2 Polynomwahl nach Coppersmith

Dieser Abschnitt basiert auf einer Arbeit von Don Coppersmith [3].

Coppersmith' Modifikation des GNFS beruht auf der Idee, statt eines Polynoms  $f(x)$  mehrere Polynome  $f_i(x)$  vom Grad  $d$  mit Nullstelle  $m$  zu verwenden. Dies wird uns eine theoretisch bessere Laufzeitschranke liefern. Buhler et al. [1] vermuten allerdings, dass sich die so erzielte asymptotische Verbesserung in der Praxis nicht bemerkbar macht, etwa für Zahlen mit weniger als 1000 Dezimalstellen.

Coppersmith GNFS

### Schritt 0': Preprocessing-Sieb

Suche Paare  $(a, b) \in \mathbb{Z}^2$  mit  $|a|, |b| < E$ ,  $\text{ggT}(a, b) = 1$  und  $a + bm$  ist  $B_1$ -glatt.

### Schritt 1': Polynomwahl

Wähle wie im GNFS ein Polynom  $f \in \mathbb{Z}[x]$  vom Grad  $d$  mit Nullstelle  $m$  modulo  $n$  durch  $m$ -äre Darstellung von  $n$ . Definiere

$$f_i(x) = f(x) + i(x - m), \quad i = 0, 1, 2, \dots$$

und betrachte entsprechende  $\alpha_i$  sowie  $\varphi_i : \mathbb{Z}[\alpha_i] \rightarrow \mathbb{Z}_n$ .

### Schritt 2': Sieb

Für jedes Polynom  $f_i$  suche aus den Paaren  $(a, b)$  von Schritt 0' diejenigen, für die  $N_{f_i}(a - b\alpha_i) = (-b)^d f_i(-\frac{a}{b})$  glatt bezüglich  $B_2$  ist.

### Schritt 3': Gleichungssystem

Suche eine Teilmenge  $S$  der Tripel  $(a, b, f_i)$  von Schritt 2', so dass

$$\prod_{(a,b,f_i) \in S} (a + bm) \quad \text{und} \quad \prod_{(a,b,f_i) \in S} (a + b\alpha_i)$$

Quadrate in  $\mathbb{Z}$  bzw.  $\mathbb{Z}[\alpha_1, \alpha_2, \dots]$  sind, und sich eine Wurzel  $x$  bzw.  $\beta$  in der Faktorbasis  $p_1, p_2, \dots$  für  $\mathbb{Z}$  bzw.  $\pi_{1,i}, \pi_{2,i}, \dots$  für  $\mathbb{Z}[\alpha_i]$  darstellen läßt.

### Schritt 4': Wurzelberechnung

—wie zuvor, da  $\varphi_i(a + b\alpha_i) \equiv (a + bm) \pmod{n}$  für alle  $i = 0, 1, 2, \dots$  —

In Schritt 3 im ursprünglichen GNFS lösen wir ein Gleichungssystem  $Mx = 0$ , bei dem die Zeilen der Matrix  $M$  durch  $\pi_1, \pi_2, \dots$  und  $p_1, p_2, \dots$  beschriftet werden, und die Spalten durch Paare  $(a, b)$ :

$$M = \begin{array}{c} \left( \begin{array}{c} \leftarrow \quad (a, b) \quad \rightarrow \\ \pi_1 \\ \pi_2 \\ \vdots \\ \hline p_1 \\ p_2 \\ \vdots \end{array} \right) \\ \begin{array}{c} \epsilon_i \bmod 2 \\ \\ \\ e_i \bmod 2 \end{array} \end{array}$$

wobei  $\epsilon_i, e_i$  die maximalen Potenzen sind, mit denen  $a + b\alpha$  bzw.  $a + bm$  von  $\pi_i$  bzw.  $p_i$  geteilt werden. Anschaulich beginnen wir mit einer leeren Matrix und fügen durch das Sieb gefallene Spalten  $(a, b)$  hinzu, bis lineare Abhängigkeiten uns ein  $x \neq 0$  mit  $Mx = 0$  liefern. Diese gesiebten Paare  $(a, b)$  müssen simultan die Glattheitsschranken für  $a + bm$  und  $(-b)^d f(-a/b)$  erfüllen.

In der Variation von Coppersmith indizieren wir nun die Spalten der Matrix  $M$  durch Tripel  $(a, b, f_i)$ . Insbesondere nehmen wir jedes  $B_1$ -glatte Paar  $(a, b)$  auf, das die  $B_2$ -Glattheitsschranke für *mindestens* ein Polynom  $f_i$  erfüllt; eventuell nehmen wir ein Paar  $(a, b)$  sogar für mehrere Polynome auf. Daher erreichen wir schneller lineare Abhängigkeiten, so dass wir die Parameter  $B_1, B_2, E$  etwas kleiner wählen können. Die resultierende Laufzeit beträgt dann

$$L_n[\frac{1}{3}, 1.902] = e^{(1.902+o(1))(\log n)^{1/3}(\log \log n)^{2/3}}$$

statt  $L_n[\frac{1}{3}, 1.923]$  wie im GNFS. Dabei wird allerdings eine relativ große Konstante im Term  $o(1)$  versteckt, so dass das Verfahren —wie oben angemerkt— zur Zeit für die Praxis keine Relevanz hat.

### 1.3 Polynomwahl für RSA-155

Der Ansatz von Cavallar et al. [2], der zur Rekordfaktorisierung einer Zahl mit 155 Dezimalstellen führte, beruht im wesentlichen auf einer verbesserten Wahl des Polynoms  $f$  im GNFS. Unser Ziel ist es,  $f(x) = \sum c_i x^i$  so zu wählen, dass die Werte  $a + bm$  und  $(-b)^d f(-a/b)$  für möglichst viele Paare  $(a, b)$  klein werden und somit die Glattheitsschranken erfüllen.

Wir schreiben  $F(v, w) \in \mathbb{Z}[v, w]$  für das Polynom  $F(v, w) = (-w)^d f(-v/w)$ , das die Norm von  $a + b\alpha$  beschreibt. Dieses Polynom  $F$  soll zwei Eigenschaften erfüllen:

- *size property*: für viele  $(a, b)$  sollen die Werte  $F(a, b)$  klein sein  
Dies kann man z.B. erreichen, indem man die Koeffizienten von  $f$  klein wählt.
- *root property*:  $F$  soll viele Nullstellen modulo kleiner Primzahlpotenzen  $p^e$  haben, so dass die Werte  $F(a, b)$  “eigentlich kleiner sind, als sie aussehen”.

Beispiel: Ist  $f(x) \equiv x^4 - 1 \pmod{5}$ , so hat  $f$  vier Nullstellen 1, 2, 3, 4 modulo 5. Dadurch kann man für ca. 4/5 der Paare  $(a, b)$  den Teiler 5 aus  $F(a, b)$  kürzen.

Um die root-Eigenschaft zu erfüllen, wird vorgeschlagen (siehe Murphys Dissertation [4]), dass man  $c_d$  als Produkt kleiner Potenzen  $p^e$  wählt. Wir lassen daher die Forderung fallen, dass  $f$  monisch sein soll. In diesem Fall ist zwar nicht mehr gesichert, dass  $\alpha \in \mathbb{Z}_{\mathbb{Q}(\alpha)}$  ist, aber durch geeignete Modifikation des GNFS-Algorithmus bereitet dies keine Probleme. Ebenso sollen die weiteren Koeffizienten  $c_{d-1}, c_{d-2}, \dots$  nicht zu groß sein, beispielsweise  $c_{d-1} \approx (n - c_d m^d) / m^{d-1}$ . Die Koeffizienten  $c_0, c_1, c_2, \dots$  müssen dafür bei festem  $d$  allerdings größer gewählt werden, gegebenenfalls sogar größer als  $m$ .

Durch die ungleichmäßige Wahl der Koeffizienten ändert sich auch der Suchbereich für die Paare  $(a, b)$ . Da das Polynom  $F(v, w)$  die Form

$$F(v, w) = c_d v^d - c_{d-1} v^{d-1} w + c_{d-2} v^{d-2} w^2 + \dots + (-1)^{d-1} c_1 v w^{d-1} + (-1)^d c_0 w^d$$

hat, und  $v$  in den Termen der kleinen Koeffizienten  $c_d, c_{d-1}, \dots$  in größerer Potenz als  $w$  eingeht (und umgekehrt für  $c_0, c_1, \dots$ ), erlauben wir größere Absolutwerte für  $a$  und kleinere für  $b$ . Mit anderen Worten, vom quadratischen Suchbereich mit  $|a|, |b| < E$  gehen wir über zu einem rechteckigen Bereich für solche verzerrten (*skewed*) Polynome  $f$ .

Für das Verfahren, um geeignete Polynome zu erzeugen, verweisen wir auf [2, 4]. So wurde beispielsweise das Polynom

$$\begin{array}{rcl} F(v, w) & = & \begin{array}{r} 11\,93771\,38320\ v^5 \\ -80\,16893\,72849\,97582\ v^4\ w \\ -66269\,85223\,41185\,74445\ v^3\ w^2 \\ +1\,18168\,48430\,07952\,18803\,56852\ v^2\ w^3 \\ +745\,96615\,80071\,78644\,39197\,43056\ v\ w^4 \\ -40\,67984\,35423\,62159\,36191\,37084\,05064\ w^5 \end{array} \\ m & = & 3912\,30797\,21168\,00077\,13134\,49081 \end{array}$$

vom Grad  $d = 5$  zur Faktorisierung von RSA-155 gefunden. Das Verhältniss Höhe/Breite des Suchraums für  $a, b$  beträgt 10 800.

Aus [2] stammt die folgende Tabelle, die die letzten Faktorisierungsrekorde darstellt. Ein MIPS-Jahr ist die Anzahl der Operation eines Jahres, wenn eine Millionen Operationen pro Sekunde ausgeführt werden:

#Dezimal- stellen	Datum	Algorithmus	Aufwand in MIPS-Jahren
RSA-120	Juni 1993	Quadratisches Sieb	825
RSA-129	April 1994	Quadratisches Sieb	5000
RSA-130	April 1996	Zahlkörpersieb	1000
RSA-140	Februar 1999	Zahlkörpersieb	2000
RSA-155	August 1999	Zahlkörpersieb	8400

Bemerkenswert ist, dass GNFS im Vergleich zum Quadratischen Sieb mit weniger als dem doppeltem Aufwand 26 Dezimalstellen mehr faktorisieren kann.

# Literaturverzeichnis

- [1] J.BUHLER, H.LENSTRAS, C.POMERANCE: Factoring Integers with the Number Field Sieve, in *A.Lenstra, H.Lenstra (Ed.): The Development of the Number Field Sieve, Lecture Notes in Mathematics, Vol. 1554, Springer-Verlag, pp. 50–94, 1993.*
- [2] S.CAVALLAR, B.DODSON, A.LENSTRAS, ...: Factorization of a 512-Bit RSA Modulus, *Eurocrypt 2000, Lecture Notes in Computer Science, Vol. 1807, Springer-Verlag, pp. 1–18, 2000.*
- [3] D.COPPERSMITH: Modifications to the Number Field Sieve, *Journal of Cryptology, Vol. 6, pp. 169–180, 1993.*
- [4] B.MURPHY: Polynomial Selection for the Number Field Sieve, *Ph.D. Thesis, Australian National University, Juli 1999.*