Bsp. Quadratisches Sieb

Bsp: Wir faktorisieren die Zahl $91 = 7 \cdot 13$.

- Als Glattheitsschranke wählen wir $b = 5$.
- Wir faktorisieren nur positive Zahlen $z_i := x_i^2 - n = (10 + i)^2 - n$.
- Daher wählen wir $B = \{2, 3, 5\}$. Es gilt $(\frac{n}{p}) = 1$ für alle $p \in B$.
- Wir wollen die Zahlen z_i im Intervall $0 \leq i \leq 9$ sieben.
- Damit gilt $z_i \leq z_9 = 19^2 - n = 270$.
- Wir berechnen alle Lösungen von $x^2 \equiv 91 \bmod p^r$ mit $p^r \leq 270$.

$p \backslash r$	1	2	3	4	5
2	1	–	–	–	–
	(11)				
3	± 1	± 1	± 19	± 46	± 127
	$(10, 11)$	$(10, 17)$	$(19, 35)$	$(46, 35)$	$(127, 35)$
5	± 1	± 4	± 29		
	$(11, 14)$	$(29, 21)$	$(29, 96)$		

Bsp. Quadratisches Sieb

- Für eine Lösung $\pm x_{p^r}$ steht in der Klammer das kleinste $x_i \geq 10$ mit $x_i \equiv x_{p^r} \mod p^r$ bzw. $x_i \equiv -x_{p^r} \mod p^r$.
- Bsp: z_{10} ist durch 3^2 teilbar und damit auch alle $z_{10+3^2\mathbb{Z}}$.
- Wir erhalten die folgenden partiellen Faktorisierungen.

x_i	$z_i = x_i^2 - n$	teilbar durch	Cofaktor
10	9	3^2	1
11	30	$2 \cdot 3 \cdot 5$	1
12	53	–	53
13	78	$2 \cdot 3$	13
14	105	$3 \cdot 5$	7
15	134	2	67
16	165	$3 \cdot 5$	11
17	198	$2 \cdot 3^2$	11
18	233	–	233
19	270	$2 \cdot 3^3 \cdot 5$	1

Bsp. Quadratisches Sieb

- Die Zeilen 11 und 19 liefern die Kongruenz
$$(11 \cdot 19)^2 \equiv 27^2 \equiv (2 \cdot 3^2 \cdot 5)^2 = 90^2 \equiv (-1)^2 \bmod 91.$$
- Es gilt $27 \not\equiv \pm 1 \bmod 91$ und $\mathrm{ggT}(27 \pm 1, 91) = \{7, 13\}$.

Anmerkungen:

- In der “Large Prime”-Variante des Siebs werden Zeilen mit demselben Co-Faktor verwendet.
- Bsp.: Für $x_i = 16$ und 17 erhalten wir die zusätzliche Relation
$$(16 \cdot 17 \cdot 11^{-1})^2 \equiv 2 \cdot 3^3 \cdot 5 \bmod 91.$$
- **Laufzeit:** Das Quadratischen Sieb benötigt Zeit $e^{\sqrt{\ln n \ln \ln n}}$. (unter geeigneten Glattheitsannahmen)
- Dies ist superpolynomiell aber supexponentiell in $\ln n$.

Pollards $p-1$ Methode

Idee:

- Sei $n = pr$ mit $1 < p < n$, p prim, $p \nmid r$. D.h. $\mathbb{Z}/n\mathbb{Z} \cong \mathbb{Z}/p\mathbb{Z} \times \mathbb{Z}/r\mathbb{Z}$.
- Sei $p-1$ b-glatt, d.h. $p-1 = \prod_{p \in B} p^{e_B}$.
- Sei k ein Vielfaches von $\prod_{p \in B} p^{e_B}$. Dann gilt
$$a^k \equiv 1 \bmod p \text{ für alle } a \in U_n.$$
- Falls zusätzlich $a^k \not\equiv 1 \bmod r$ folgt $p \leq \mathrm{ggT}(a^k - 1, n) < n$.

Algorithmus Pollards $p-1$-Methode

EINGABE: $n = pr$ zusammengesetzt, p prim, Schranke C mit $p \leq C$.

1. Wähle b geeignet, so dass $p-1$ b-glatt ist. Sei $B = \{p_1, \ldots, p_s\}$.
2. Wähle $a \in_R \{2, \ldots, n-1\}$. Falls $\mathrm{ggT}(a, n) > 1$, Ausgabe des ggTs.
3. Für $i = 1 \ldots s$
   1. Wähle e_i maximal mit $p_i^{e_i} < C$. Berechne $a := a^{p_i^{e_i}} \bmod N$.
4. Falls $\mathrm{ggT}(a-1, N) \notin \{1, N\}$, Ausgabe des ggTs.

Analyse von Pollards $p-1$-Methode

Korrektheit:

- In Schritt 3.1 wird $a^k \bmod N$ berechnet mit $k = \prod_{i=1}^{s} p_i^{e_i}$.
- Falls $p-1$ b-glatt ist, gilt $p-1|k$.
- Damit ist $\mathrm{ggT}(a^k-1, n) \geq p$.
- D.h. wir finden einen nicht-trivialen Teiler, falls $\mathrm{ggT}(a^k-1, n) < n$.
- Sei q ein Primteiler von r, so dass $q-1$ nicht b-glatt ist.
- Damit existiert ein $q'|q-1$, $q' \in \mathbb{P}$ mit $q' > b$.
- Ferner gelte $q'|\mathrm{ord}(a)$ in U_q. Dann gilt
$$a^k \not\equiv 1 \bmod q \text{ und damit } \mathrm{ggT}(a^k-1, n) < n.$$
- Wir berechnen die Ws, dass $q'|\mathrm{ord}(a)$ in U_q.
- Sei U_q zyklisch mit Generator g. Wir schreiben $a \equiv g^i \bmod q$.
- Es folgt $\mathrm{ord}(a) = \frac{q-1}{\mathrm{ggT}(i,q-1)}$ in U_q. Falls $q' \nmid i$, gilt $q'|\mathrm{ord}(a)$.
- Da a zufällig gewählt ist, geschieht dies mit Ws $1-\frac{1}{q'}$.

Analyse von Pollards $p-1$-Methode

Laufzeit:

- Schritt 3 benötigt Zeit $\mathcal{O}(s \log C \log N^2) = \mathcal{O}(s \log^3 N)$.

Problem der $p-1$-Methode:

- Die Laufzeit ist abhängig von der Ordnung von U_p.
- Sei $\frac{p-1}{2} \in \mathbb{P}$ mit $\frac{p-1}{2} \approx \sqrt{n}$.
- Dann benötigen wir $p_s \approx \sqrt{n}$ und damit
$$s = \left|\{x \in \mathbb{P} \mid x \leq p_s\}\right| \approx \frac{\sqrt{n}}{\log n}.$$
- In diesem Fall ist die Laufzeit nicht besser als bei Probedivision.