Fortgeschrittene Techniken der Kryptographie

Jürgen Fuß

Episode 1: Mathematische Werkzeuge

HAGENBERG | LINZ | STEYR | WELS

UNIVERSITY OF APPLIED SCIENCES UPPER AUSTRIA

Notation

In dieser LVA wird „mod“ in zwei verschiedenen Bedeutungen verwendet.

1. Als arithmetischer Operator wie in

$$17 \bmod 3 = 2$$

berechnet „mod“ den Rest bei der Division.

2. Als Kongruenzrelation wie in

$$7^{14} = 7 + 2 \pmod{10}$$

wird im Gegensatz dazu ausgesagt, dass diese Gleichung „modulo 10 stimmt“. Alternativ wird auch das Relationensymbol $\equiv_n$ verwendet:

$$7^{14} \equiv_{10} 7 + 2$$

Euklids Algorithmus

I	203	
II	112	
III	91	III=I-II
IV	21	IV=II-III
V	7	V=III-4IV
VI	0	VI=IV-3V

$$\text{ggT}(203, 112) = 7$$

203 : 112 = 1 x
Rest 91

112 : 91 = 1
Rest 21

91 : 21 = 4
Rest 7

Satz

Seien $a, b \in \mathbb{Z}$. Dann gibt es ganze Zahlen x und y, so dass

$$\mathrm{ggT}(a, b) = ax + by,$$

und der **erweiterte Euklidische Algorithmus** berechnet die Zahlen x und y.

$$7 = 203 \cdot x + 112 \cdot y$$

		203	112	
I	203	1	0	
II	112	0	1	
III	91	1	-1	III=I-II
IV	21	-1	2	IV=II-III
V	7	5	-9	V=III-4IV
VI	0			VI=IV-3V

$203 = 203 \cdot 1 + 112 \cdot 0$

$112 = 203 \cdot 0 + 112 \cdot 1$ $\Big] -$

$91 = 203 \cdot 1 + 112 \cdot (-1)$

$21 = 203 \cdot (-1) + 112 \cdot 2$

$$\text{ggT}(203, 112) = 7 = 203 \cdot 5 + 112 \cdot (-9)$$

Mit der Funktion `extended_gcd()` aus dem Modul `si` lassen sich diese Werte berechnen. Mit der Option `verbose=1` werden auch alle Zwischenergebnisse angezeigt.

```
> si.extended_gcd( 203, 112, verbose=1 )
'                     203     112
'              203      1       0
'              112      0       1       1
'               91      1      -1       1
'               21     -1       2       4
'                7      5      -9       3
(7, 5, -9)
```

Euklids Algorithmus

Satz von Lamé, 1844

Sind a und b natürliche Zahlen, ist $a > b$ und ist n die Bitlänge von b, so endet der Euklidsche Algorithmus zur Berechnung von $\mathrm{ggT}(a, b)$ nach spätestens $17 \cdot n$ Schritten.

Satz

Es sei c eine natürliche Zahl mit n Bit Länge. Für zufällig gewählte Zahlen a und b zwischen 1 und c ist die erwartete Anzahl an Schritten zur Berechnung von $\mathrm{ggT}(a, b)$ mit dem Euklidschen Algorithmus $0{,}584 \cdot n + 0{,}06$.

Da die Zahlen im Euklidschen Algorithmus stets kleiner werden, liegt der Hauptaufwand in den ersten Modulooperationen.

Die Eulersche Phi-Funktion

$$\begin{matrix} \cancel{0} & 1 & 2 & 3 & 4 \end{matrix}$$
$$\mathbb{Z}_5^* = \{1, 2, 3, 4\}$$

$$\begin{matrix} 0 & 1 & \cancel{2} & 3 \end{matrix}$$
$$\mathbb{Z}_4^* = \{1, 3\}$$

Definition

Für $n \in \mathbb{N}$ sei

$$\mathbb{Z}_n^* := \{k \in \mathbb{Z} \mid 0 \leq k < n \text{ und } \operatorname{ggT}(n, k) = 1\}.$$

$\operatorname{ggT}(0, 4) = 4$

$\mathbb{Z}_1^* = \{0\}$

Die Funktion

$$\begin{aligned} \varphi : \ \mathbb{N} &\to \mathbb{N} \\ n &\mapsto |\mathbb{Z}_n^*| \end{aligned}$$

$\varphi(4) = 2$

$\varphi(5) = 4$

$\varphi(13) = 12$

heißt **Eulersche φ–Funktion**.

$\varphi(p) = p - 1$

Primer Modul p

Ist $p \in \mathbb{P}$, so lässt sich $\varphi(p)$ recht einfach berechnen. Da p keine Primfaktoren (außer sich selbst) besitzt, gilt für jede ganze Zahl $k \in \{1, \ldots, p-1\}$: $\mathrm{ggT}(p, k) = 1$. Lediglich $\mathrm{ggT}(p, 0) = p > 1$. Daher ist

$$\varphi(p) = p - 1.$$

$\cancel{0}\ 1\ \cancel{2}\ \cancel{3}\ \cancel{4}\ 5$

$\mathbb{Z}_6^* = \{1, 5\}$

$\varphi(6) = 2$

$\varphi(42) = 12$

$\mathbb{Z}_{42}^* = \{1, 5, 11, 13, 17, 19, 23, 25, 29, 31, 37, 41\}$

Biprimer Modul $n = p \cdot q$

$n = 15 = 3 \cdot 5$

- $\text{ggT}(n,k) = p$: bei $k = p, 2p, 3p, \ldots, (q-1)p$.
- $\text{ggT}(n,k) = q$: bei $k = q, 2q, 3q, \ldots, (p-1)q$.

$\text{ggT}(15, k) = 3$

$3, 6, 9, 12,$

$\text{ggT}(15, k) = 5$

$5, 10,$

Bei allen übrigen k (außer 0) ist $\text{ggT}(n,k) = 1$. Das sind

$pq - 1 \quad - q + 1 \quad - p + 1$

$$\underbrace{n-1}_{\text{alle außer } 0} - \underbrace{(q-1)}_{\text{Vielfache von } p} - \underbrace{(p-1)}_{\text{Vielfache von } q} = pq - p - q + 1 = (p-1)(q-1).$$

Also ist

$$\varphi(p \cdot q) = (p-1)(q-1).$$

$\varphi(p) = p - 1$

Satz

Der Wert $\varphi(n)$ lässt sich effizient berechnen, wenn man die Primfaktorzerlegung von n kennt. Ist $n = p_1^{e_1} \cdots p_r^{e_r}$ die Primfaktorzerlegung von $n \in \mathbb{N}$, dann ist

$$\varphi(n) = n \cdot \left(1 - \frac{1}{p_1}\right) \cdots \left(1 - \frac{1}{p_r}\right).$$

$n = 42 = 2 \cdot 3 \cdot 7$

$\underset{p_1}{2^1} \cdot \underset{p_2}{3^1} \cdot \underset{p_3}{7^1}$

$\varphi(42) = 42 \cdot \left(1 - \frac{1}{2}\right)\left(1 - \frac{1}{3}\right)\left(1 - \frac{1}{7}\right)$

$= 12$

Mit der Funktion `euler_phi()` aus dem Modul `si` lässt sich $\varphi(n)$ berechnen, sofern n faktorisiert werden kann.

```
> si.euler_phi( 42 )
12
> si.prime_factors( 42 )
[2, 3, 7]
> 42 * (1-1/2) * (1-1/3) * (1-1/7)
12.000000000000002
```

Faktorisieren muss sein

$n = p \cdot q \longrightarrow q = n/p$

$\varphi(n) = (p-1)(q-1) =$

$\boxed{\varphi(n) = (p-1)\cdot\left(\frac{n}{p}-1\right)}$

$\varphi(n) = n - n/p - p + 1 \quad | \cdot p$

Ist $n = p \cdot q$ und sind sowohl n als auch $\varphi(n)$ bekannt, so lässt sich n einfach faktorisieren.

$\varphi(n) \cdot p = np - n - p^2 + p$

$p^2 + p(\varphi(n) - n - 1) + n = 0$

Das bedeutet umgekehrt: Kann n nicht faktorisiert werden, so kann $\varphi(n)$ nicht bestimmt werden.

$p^2 + p \cdot 17 + 24 = 0$

$p_{1/2} = - \pm \sqrt{}$

Der chinesische Restsatz

Die Reduktion modulo n ist eine recht einfache Operation, schnell ergibt sich

$$\begin{aligned} 42 \bmod 3 &= 0 \\ 42 \bmod 4 &= 2 \\ 42 \bmod 5 &= 2 \end{aligned}$$

Umgekehrt stellen wir uns jetzt die Frage, ob und wie sich aus den Gleichungen

$$\begin{aligned} z &= 0 \pmod 3 \\ z &= 2 \pmod 4 \\ z &= 2 \pmod 5 \end{aligned}$$

$z = \ldots \pmod{12}$

$z = 2 \pmod 5$

z bestimmen lässt. Schon wieder ein schwieriges Problem. Müssen wir hier probieren? Ist 42 die einzige Lösung?

Chinesischer Restsatz für zwei Gleichungen

Es seien $n_1, n_2 \in \mathbb{N}$, so dass $\operatorname{ggT}(n_1, n_2) = 1$. Weiterhin seien $z_1, z_2 \in \mathbb{Z}$. Dann erhält man alle Lösungen des Restklassengleichungssystems

$n_1 = 5$
$n_2 = 7$
$n = 35$

$$\begin{cases} z = z_1 \pmod{n_1} \\ z = z_2 \pmod{n_2} \end{cases}$$

auf folgende Weise:

1. Berechne $n = n_1 \cdot n_2$.
2. Berechne mithilfe des erweiterten Euklidschen Algorithmus ganze Zahlen x_1 und x_2, so dass $n_1 x_1 + n_2 x_2 = 1$.
3. Berechne

$$z := z_1 n_2 x_2 + z_2 n_1 x_1 \mod n.$$

Dieses z ist die eindeutige Lösung des Restklassengleichungssystems modulo n. Die Menge aller Lösungen ist $\{z + kn \mid k \in \mathbb{Z}\}$.

UNIVERSITY OF APPLIED SCIENCES UPPER AUSTRIA

$$z := z_1 n_2 x_2 + z_2 n_1 x_1 \mod n \quad (\text{mit } n_1 x_1 + n_2 x_2 = 1).$$

$n_2 x_2 = 1 - n_1 x_1$

$n_1 x_1 = 1 - n_2 x_2$

ist Lösung des Gleichungssystems

$$\begin{aligned} z &= z_1 \pmod{n_1} \\ z &= z_2 \pmod{n_2}, \end{aligned}$$

denn

$$\begin{aligned} z &= z_1 n_2 x_2 + z_2 \underbrace{n_1}_{=0} x_1 = z_1 \underbrace{n_2 x_2}_{=1-n_1 x_1} = z_1(1 - \underbrace{n_1}_{=0} x_1) = z_1 \pmod{n_1} \quad \text{und} \\ z &= z_1 \underbrace{n_2}_{=0} x_2 + z_2 n_1 x_1 = z_2 \underbrace{n_1 x_1}_{=1-n_2 x_2} = z_2(1 - \underbrace{n_2}_{=0} x_2) = z_2 \pmod{n_2} \end{aligned}$$

Chinesischer Restsatz (I)

Es seien $n_1, n_2, \ldots, n_s$ paarweise relativ prime natürliche Zahlen. Weiterhin seien $z_1, z_2, \ldots, z_s \in \mathbb{Z}$. Dann erhält man alle Lösungen des Restklassengleichungssystems

$$\begin{aligned} z &= z_1 \pmod{n_1} \\ &\vdots \\ z &= z_s \pmod{n_s} \end{aligned}$$

auf folgende Weise:

1. Berechne $n = n_1 \cdot n_2 \cdots n_s$. Modulo n ist die Lösung eindeutig.
2. Berechne für $i = 1, \ldots, s$ die Zahlen

$$q_i := \frac{n}{n_i}.$$

Chinesischer Restsatz (II)

3. Berechne für $i = 1, \ldots, s$ das inverse Element r_i von q_i modulo n_i (mit dem erweiterten Euklidschen Algorithmus), also

$$r_i := q_i^{-1} \bmod n_i.$$

4. Berechne

$$z := z_1 q_1 r_1 + z_2 q_2 r_2 + \cdots + z_s q_s r_s \bmod n.$$

Dieses z ist die eindeutige Lösung des Restklassengleichungssystems modulo n. Die Menge aller Lösungen ist $\{z + kn \mid k \in \mathbb{Z}\}$.

Mit der Funktion `chinese_remainder()` aus dem Modul `si` lässt sich dieses Restklassengleichungssystem lösen. Übergeben werden eine Liste mit den Moduln (3, 4 und 5) und eine Liste mit den Resten (0, 2 und 2).

```
> from si import chinese_remainder
> chinese_remainder( [3,4,5], [0,2,2] )
42
```

Die Sätze von Fermat und Euler

$$a \equiv b \pmod p \quad | \cdot z$$
$$az \equiv b \cdot z \pmod p \quad | \cdot z^{-1}$$
$$a \equiv b \pmod p$$

Kleiner Satz von Fermat

Ist $p \in \mathbb{P}$, ist $z \in \mathbb{Z}$ und ist $\mathrm{ggT}(z, p) = 1$, dann gilt

$$z^{p-1} = 1 \pmod p.$$

$$z^p \equiv z \pmod p$$

$$\boxed{1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot \ldots \cdot (p-1)} \pmod p$$

$\cdot z \downarrow \quad \cdot z \downarrow \quad \cdot z \downarrow \quad \cdot \downarrow \quad \cdot z \downarrow$

$$1 \cdot z \cdot 2z \cdot 3z \cdot 4 \cdot z \cdot 5 \cdot z \cdot \ldots (p-1) \cdot z \pmod p$$

$$z^{p-1} \cdot \cancel{1} \cdot \cancel{2} \cdot \cancel{3} \cdot \cancel{4} \cdot \ldots \cdot \cancel{(p-1)} \equiv \cancel{1} \cdot \cancel{2} \cdot \cancel{3} \cdot 4 \ldots \cdot \cancel{(p-1)} \pmod p$$

$$z^{p-1} \equiv 1 \pmod p$$

UNIVERSITY OF APPLIED SCIENCES UPPER AUSTRIA

Der Satz von Euler

Satz von Euler für $n = p \cdot q$

Sind $p, q \in \mathbb{P}$, $n = pq$ und $z \in \mathbb{Z}$, dann gilt für alle $z \in \mathbb{Z}$ mit $\operatorname{ggT}(z, n) = 1$:

$$z^{(p-1)(q-1)} = 1 \pmod{n}.$$

$$z^{(p-1)(q-1)} = 1 \mod p$$

$$z^{(p-1)(q-1)} = 1 \mod q$$

$$\Downarrow$$

$$\left(\boxed{z^{p-1}}\right)^{(q-1)} = 1$$

UNIVERSITY OF APPLIED SCIENCES UPPER AUSTRIA

Satz von Euler

Sind $n \in \mathbb{N}$ und $z \in \mathbb{Z}$ und ist $\mathrm{ggT}(z, n) = 1$, dann gilt

$$z^{\varphi(n)} = 1 \pmod{n}.$$

Korollar

Sind $n \in \mathbb{N}$ und $z, a, b \in \mathbb{Z}$ und ist $\mathrm{ggT}(z, n) = 1$, dann gilt:

1. Ist $a = b \pmod{\varphi(n)}$, dann ist $z^a = z^b \pmod{n}$.
2. $z^a = z^{a \bmod \varphi(n)} \pmod{n}$.