Stelsels

Stelsels van lineaire congruenties

Beschouw het stelsel:

$$\begin{cases} x \overset{m_1}{=} a_1 \\ x \overset{m_2}{=} a_2 \\ \dots \\ x \overset{m_n}{=} a_n \end{cases}$$

Met $a_i \in \mathbb{Z}, m_i \in \mathbb{N}_0, m_i$ onderling priem, $x \in \mathbb{Z}$

Bemerk: de coëfficiënten van x zijn allemaal 1 (= algemene vorm van lineaire congruentie)

Chinese reststelling:

Dit stelsel bezit een unieke oplossing modulo $M = m_1 \cdot m_2 \cdots m_n$

Stelsels van lineaire congruenties

Bewijs:

1. <u>Constructie van een oplossing</u>

   Stel $M_i = \frac{M}{m_i}$ voor alle i

   $\Rightarrow \text{ggd}(m_i, M_i) = 1$

   $\Rightarrow$ de lineaire congruentie $M_i \cdot y \overset{m_i}{=} 1$ heeft een oplossing y_i

$$\begin{cases} x \overset{m_1}{=} a_1 \\ x \overset{m_2}{=} a_2 \\ \dots \\ x \overset{m_n}{=} a_n \end{cases}$$

   Definieer nu $x = a_1 M_1 y_1 + a_2 M_2 y_2 + \cdots + a_n M_n y_n$

   Dan is x inderdaad een oplossing van het stelsel

   $\Rightarrow x = (a_1 M_1 y_1 + a_2 M_2 y_2 + \cdots + a_n M_n y_n) \bmod M$ is een oplossing van het stelsel

   en $x \in \{0, 1, 2, \dots, M-1\}$

Stelsels van lineaire congruenties

$$\begin{cases} x \overset{m_1}{=} a_1 \\ x \overset{m_2}{=} a_2 \\ \dots \\ x \overset{m_n}{=} a_n \end{cases}$$

Bewijs:

2. <u>De gevonden oplossing is uniek</u>

   Ongerijmde: stel dat er twee oplossingen mod M zijn: x_1 en x_2

   $\Rightarrow z = x_1 - x_2$ is een oplossing van

$$\begin{cases} z \overset{m_1}{=} 0 \\ z \overset{m_2}{=} 0 \\ \dots \\ z \overset{m_n}{=} 0 \end{cases}$$

   $\Rightarrow z = x_1 - x_2$ is een veelvoud van m_1, van m_2, ...

   $\Rightarrow z = x_1 - x_2$ is een veelvoud van M (want alle m_i onderling ondeelbaar)

   Dus: x_1 en x_2 moeten minstens een veelvoud van M verschillen: contradictie

Voorbeeld 1

Beschouw het stelsel:

$$\begin{cases} x \overset{3}{=} 2 \\ x \overset{5}{=} 3 \\ x \overset{7}{=} 2 \end{cases}$$

3, 5 en 7 zijn onderling priem, dus er is één unieke oplossing modulo $M = 105 = 3 \cdot 5 \cdot 7$

We vinden:

$M_1 = 35, M_2 = 21, M_3 = 15$

De unieke oplossing in $\{0,1,2, \ldots 104\}$ is:

$x = a_1 M_1 y_1 + a_2 M_2 y_2 + a_3 M_3 y_3 \bmod M = 2 \cdot 35 \cdot y_1 + 3 \cdot 21 \cdot y_2 + 2 \cdot 15 \cdot y_3 \bmod 105$

waarbij y_i de inverse is van M_i modulo m_i

Voorbeeld 1

Beschouw het stelsel:

$$\begin{cases} x \overset{3}{=} 2 \\ x \overset{5}{=} 3 \\ x \overset{7}{=} 2 \end{cases}$$

De unieke oplossing in $\{0,1,2, \ldots 104\}$ is:

$x = a_1 M_1 y_1 + a_2 M_2 y_2 + a_3 M_3 y_3 \bmod M = 2 \cdot 35 \cdot y_1 + 3 \cdot 21 \cdot y_2 + 2 \cdot 15 \cdot y_3 \bmod 105$

waarbij y_i de inverse is van M_i modulo m_i

Via Euclides vinden we deze inversen (reken na of programmeer): $y_1 = 2, y_2 = 1, y_3 = 1$

En dus: $x = 2 \cdot 35 \cdot 2 + 3 \cdot 21 \cdot 1 + 2 \cdot 15 \cdot 1 \bmod 105 = 23$

Controleer door $x = 23$ terug in te vullen in het stelsel!

Voorbeeld 2

Beschouw het stelsel:

$$\begin{cases} x \overset{3}{=} 2 \\ x \overset{6}{=} 5 \end{cases}$$

3 en 6 zijn **niet** onderling priem, dus het stelsel moet eerst gereduceerd worden naar een vorm waarbij alle m_i onderling priem zijn

De laatste vergelijking valt uiteen in $x \overset{2}{=} 5$ en $x \overset{3}{=} 5$ wat equivalent is met $x \overset{2}{=} 1$ en $x \overset{3}{=} 2$

Het stelsel wordt dus:

$$\begin{cases} x \overset{3}{=} 2 \\ x \overset{2}{=} 1 \\ x \overset{3}{=} 2 \end{cases} \Leftrightarrow \begin{cases} x \overset{3}{=} 2 \\ x \overset{2}{=} 1 \end{cases}$$

Voorbeeld 3

Beschouw het stelsel:

$$\begin{cases} x \overset{3}{=} 2 \\ 3x \overset{9}{=} 12 \end{cases}$$

3 en 9 zijn **niet** onderling priem, dus het stelsel moet eerst gereduceerd worden naar een vorm waarbij alle m_i onderling priem zijn

In de laatste vergelijking kunnen we alle getallen delen door 3: $x \overset{3}{=} 4$ of nog $x \overset{3}{=} 1$

Het stelsel wordt dus:

$$\begin{cases} x \overset{3}{=} 2 \\ x \overset{3}{=} 1 \end{cases}$$

Dit is duidelijk een strijdig stelsel

Toepassingen

Residu-talstelsels

Probleem:

bewerkingen + en $\times$ bij enorm grote gehele getallen m.b.v. computer

mogelijke manieren :

gewone methode (decimaal, binair of hexadecimaal): sequentieel…

parallelle methode?

Residu-talstelsels

Beschouw $m_1, m_2, \ldots, m_n$ allen onderling priem, en $M = m_1 m_2 \cdots m_n$

Chinese reststelling: elk getal $x < M$ kan nu op unieke wijze voorgesteld worden als

$$(x \bmod m_1, x \bmod m_2, \ldots, x \bmod m_n)$$

Bijvoorbeeld:

Kies $m_1 = 99$, $m_2 = 98$, $m_3 = 97$ en $m_4 = 95$

Dan kan men alle getallen $< 89\,403\,930$ voorstellen

Het getal 123684 wordt uniek voorgesteld door (33, 8, 9, 89)

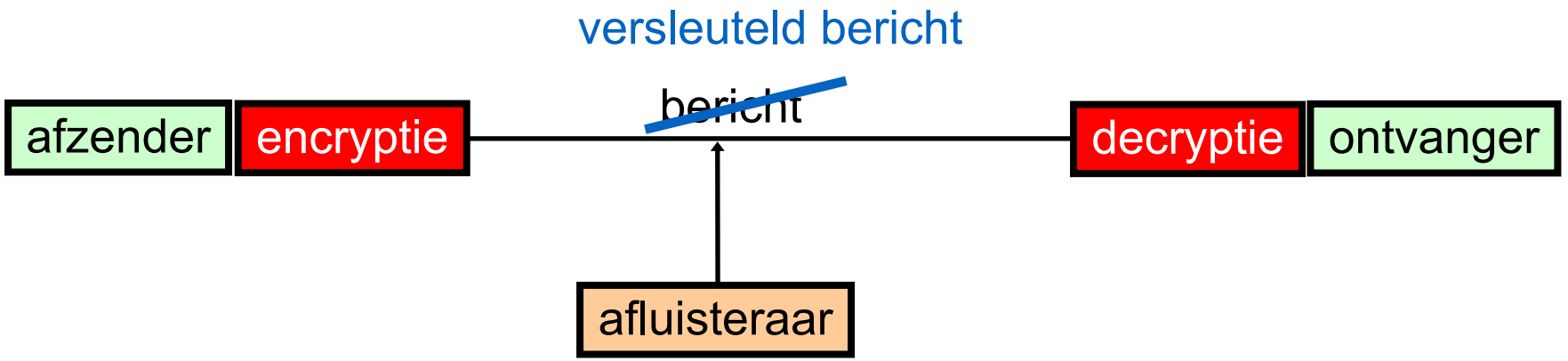
… want $33 = 123684 \bmod 99$, enz.

Residu-talstelsels

Vergelijking met gewone talstelsels :

- \+ bewerkingen + en $\times$ veel eenvoudiger en dus sneller (let wel op voor overflow)
- \+ bewerkingen + en $\times$ inherent parallel (parallel computing)
- – enkel in $\mathbb{Z}$, niet in $\mathbb{R}$
- – andere bewerkingen veel complexer (dus enkel specifieke toepassingen)

Cryptologie



Versleutelen van berichten

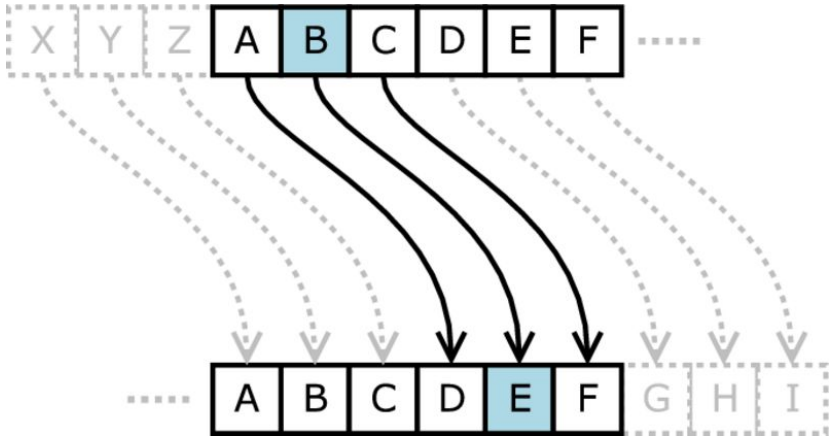
- Symmetrische encryptie: er is één sleutel om te encrypteren en decrypteren – '**private key**'
- Asymmetrische encryptie: er zijn twee verschillende sleutels – '**public key**'

Private key: Caesar's encryptiemethode

Encryptie:

Elke letter A … Z wordt omgezet in een getal $n = 0 \ldots 25$

$f(n) = (n + k) \bmod 26$ (k = sleutel)



Decryptie :

$g(n) = (n - k) \bmod 26$

Elk getal $n = 0 \ldots 25$ wordt terug een letter A … Z

Voordelen en nadelen:

\+ heel eenvoudig

– eenvoudig te kraken (bijv. door afluisteren van de sleutel, of door frequentieanalyse)

Voorbeeld: "brexit" met $k = 24$ wordt "zpcvgr"

Public key: RSA

Idee: de afzender kiest/berekent een sleutel en verspreid deze publiekelijk
De sleutel is dus gekend door iedereen, maar enkel de afzender kan deze decrypteren

RSA-encryptie (1976, MIT)

ontvanger kiest 2 grote priemgetallen p en q en een exponent e [ggd(e, (p – *1*) x (q – 1)) = 1]
ontvanger berekent $n = p \times q$
ontvanger stuurt sleutel (n, e) naar afzender
afzender encrypteert het bericht M tot de code C: $C = M^e \bmod n$
ontvanger berekent d = inverse van e modulo (p – *1*) x (q – 1)
ontvanger decrypteert code C tot bericht M: $M = C^d \bmod n$
(zonder bewijs: gebaseerd op de kleine stelling van Fermat)

Tegenwoordig wordt aangeraden om voor n een getal met 2048 bits ≈ 600 cijfers te nemen
De sterkte van RSA is gebaseerd op het moeilijk kunnen ontbinden van een groot getal

RSA: voorbeeld

We wensen het karakter ‘Y’ te verzenden, met ASCII-code 89

1. Kies twee priemgetallen $p = 11$ en $q = 29$ en exponent $e = 3$ [ggd(3, 10 x 28) = 1]
2. Bereken $n = p \times q = 319$
3. Stuur de sleutel $(n, e) = (319, 3)$ naar de afzender
4. Afzender encrypteert $M = 89$ als volgt: $C = M^e \bmod n = 89^3 \bmod 319 = 298$
5. Afzender stuurt $C = 298$ over het communicatiekanaal naar de bestemmeling
6. Bestemmeling berekent inverse van e modulo $(p - 1) \times (q - 1) = d = 187 \bmod 280$
7. Bestemmeling decrypteert het bericht als volgt: $M = C^d \bmod n = 298^{187} \bmod 319 = 89$
8. 89 als ASCII-code komt inderdaad overeen met de oorspronkelijke letter ‘Y’