Algorithmische Methoden 1

Wolfgang Windsteiger

RISC Institut

Das Thema der heutigen Vorlesung

Der Euklid'sche Algorithmus, diophantische Gleichungen und Kongruenzklassen modulo m

Grundbegriffe

- Grundbegriffe Teilbarkeit, ggT, etc. und die Grundversion des Euklidschen Algorithmus $\rightsquigarrow$ Lineare Algebra.
- Erweiterter Euklidscher Algorithmus berechnet zusätzlich Koeffizienten $x, y \in \mathbb{Z}$ so, dass

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

- Anwendung zur Lösung einfacher diophantischer Gleichungen (linear, in zwei Unbekannten) der Form

$$ax + by = c.$$

Definitionen

Definition (Kongruenz modulo m)

Sei $m \in \mathbb{N}$. Die Zahlen $a, b \in \mathbb{Z}$ heißen kongruent modulo m genau dann, wenn $m|(a-b)$ gilt. Für die Kongruenz von a und b modulo m schreiben wir $a \equiv_m b$.

$\equiv_m$ ist eine Kongruenzrelation auf $\mathbb{Z}$ ist, d.h. eine Äquivalenzrelation auf $\mathbb{Z}$, die mit Addition und Multiplikation auf $\mathbb{Z}$ verträglich ist im Sinne von

$$a \equiv_m b \text{ und } a' \equiv_m b' \implies (a + a' \equiv_m b + b' \quad \text{und} \quad a \cdot a' \equiv_m b \cdot b').$$

Definitionen

Definition (Kongruenzklassen, Rechnen modulo m)

Für $a \in \mathbb{Z}$ nennt man die Äquivalenzklasse von a bzgl. $\equiv_m$ auch Kongruenzklasse (oder Restklasse) von a modulo m und bezeichnet sie mit $[a]_m$. Weiters ist

$$\mathbb{Z}_m := \{[a]_m \mid a \in \mathbb{Z}\},$$

und in $\mathbb{Z}_m$ heißt m der Modul. Auf $\mathbb{Z}_m$ können für $\diamond \in \{+, -, \cdot\}$ durch

$$[a]_m \diamond [b]_m := [a \diamond b]_m \tag{29}$$

selbst wieder Addition, Subtraktion und eine Multiplikation definiert werden.

Ein paar Fakten

- Für jedes $b \in [a]_m$ ist $b \bmod m = r_a = a \bmod m$. Damit haben alle Elemente in $[a]_m$ bei Division durch m denselben Rest, nämlich r_a, daher auch der Name Restklasse.
- Da auch $r_a + q \cdot m \in [a]_m$ für jedes $q \in \mathbb{Z}$ gilt, ist

$$[a]_m = \{r_a + q \cdot m \mid q \in \mathbb{Z}\}.$$

- In $\mathbb{Z}_m$ gibt es genau m verschiedene Klassen, also

$$\mathbb{Z}_m = \{[0]_m, \ldots, [m-1]_m\}.$$

Beispiel

Beispiel

In $\mathbb{Z}_3$ lauten die Kongruenzklassen

$$[0]_3 = \{\dots, -6, -3, 0, 3, 6, \dots\}$$
$$[1]_3 = \{\dots, -5, -2, 1, 4, 7, \dots\}$$
$$[2]_3 = \{\dots, -4, -1, 2, 5, 8, \dots\}.$$

und jede Klasse hat die Gestalt $[a]_3 = \{a + q \cdot 3 \mid q \in \mathbb{Z}\}$.

$+$	$[0]_3$	$[1]_3$	$[2]_3$
$[0]_3$	$[0]_3$	$[1]_3$	$[2]_3$
$[1]_3$	$[1]_3$	$[2]_3$	$[0]_3$
$[2]_3$	$[2]_3$	$[0]_3$	$[1]_3$

$\cdot$	$[0]_3$	$[1]_3$	$[2]_3$
$[0]_3$	$[0]_3$	$[0]_3$	$[0]_3$
$[1]_3$	$[0]_3$	$[1]_3$	$[2]_3$
$[2]_3$	$[0]_3$	$[2]_3$	$[1]_3$

Invertieren in $\mathbb{Z}_m$

In $\mathbb{Z}_4$ ist $[2]_4$ wegen

$$[2]_4 \cdot [1]_4 = [2]_4 \qquad [2]_4 \cdot [2]_4 = [0]_4 \qquad [2]_4 \cdot [3]_4 = [2]_4$$

nicht invertierbar bzgl. der Multiplikation.

Satz

Seien $b \in \mathbb{Z}$ und $m \in \mathbb{N}$. Dann ist $[b]_m \in \mathbb{Z}_m$ invertierbar bzgl. der Multiplikation genau dann, wenn $\mathrm{ggT}(m, b) = 1$.

Beweis.

$$[b]_m \cdot [x]_m = [1]_m \iff bx \equiv_m 1 \iff bx - km = 1 \text{ für ein } k \in \mathbb{Z}.$$

Diophantische Gleichung: lösbar genau dann, wenn $\mathrm{ggT}(m, b) | 1$, also $\mathrm{ggT}(m, b) = 1$. □

Kanonische Form

Definition (Repräsentantensystem)

Ein Repräsentantensystem einer Äquivalenzrelation $\sim$ auf einer Menge M ist eine Menge $R \subseteq M$, in der aus jeder Äquivalenzklasse genau ein Element enthalten ist.

Es gibt zu jedem $a \in \mathbb{Z}$ genau ein $a' \in R$ mit $a \equiv_m a'$. Die Operation

$$\text{kanonisch}_{\mathbb{Z}_m} : \mathbb{Z} \to R,\ a \mapsto a'$$

nennt man in der Regel einen kanonischen Simplifikator für $\equiv_m$ und a' nennt man die kanonische Form von a bzgl. $\equiv_m$.

Rechnen in $\mathbb{Z}_m \rightsquigarrow$ Rechnen in R.

Darstellung

Computerrepräsentation (Datenstrukturen $\mathcal{Z}_m^+$ und $\mathcal{Z}_m^\pm$)

1. $R = \{0, \ldots, m-1\}$. *Die auf diesem System beruhende Datenstruktur nennen wir* $\mathcal{Z}_m^+$, *der zugehörige kanonische Simplifikator dazu lautet*

$$\text{kanonisch}_{\mathcal{Z}_m^+}(a) := a \bmod m.$$

2. $R = \{-((m-1) \operatorname{div} 2), \ldots, 0, \ldots, m \operatorname{div} 2\}$. *Die darauf aufbauende Datenstruktur nennen wir* $\mathcal{Z}_m^\pm$, *der kanonische Simplifikator hierfür lautet*

$$\text{kanonisch}_{\mathcal{Z}_m^\pm}(a) := \begin{cases} a \bmod m & \text{falls } a \bmod m \leq m \operatorname{div} 2 \\ (a \bmod m) - m & \text{sonst.} \end{cases}$$

Rechnen in $\mathbb{Z}_m$

- $$r \diamond s := \text{kanonisch}(rep(r) \diamond rep(s))$$
- Invertieren:

Algorithmus *InversZmEuklid*: Invertieren in $\mathbb{Z}_m$

$b \leftarrow rep(r)$
$(t, -k, x) \leftarrow ErwGGTZEuklid(m, b)$
$rep(s) \leftarrow \text{kanonisch}_{\mathcal{Z}_m^+}(x)$
return s

Aufruf: $InversZmEuklid(r)$
Eingabe: $r \in \mathcal{Z}_m^+$
mit: $\text{ggT}(m, rep(r)) = 1$
Ausgabe: $s \in \mathcal{Z}_m^+$
mit: $r \cdot s = 1 \in \mathcal{Z}_m^+$.

- $\rightsquigarrow$ *Mathematica*

Der Chinesische Restalgorithmus

Problemstellung (Chinesisches Restproblem der Dimension n).

$$\begin{aligned} \text{Gegeben:} \quad & r, m \in \mathbb{Z}^n \\ \text{mit:} \quad & \operatorname{ggT}(m_i, m_j) = 1 \text{ für } i \neq j. \\ \text{Gesucht:} \quad & z \in \mathbb{Z} \\ \text{mit:} \quad & z \equiv_{m_i} r_i \quad \text{für } i = 1, \ldots, n. \end{aligned}$$

Beispiel

Bei gegebenen $r = (5, 12)$ *und* $m = (17, 31)$ *erfüllt etwa* $z = 260$ *die geforderten Eigenschaften, da* $260 \equiv_{17} 5$ *und* $260 \equiv_{31} 12$. *Weitere Lösungen sind etwa durch* 787 *oder* -267 *gegeben. Es gilt aber* $z \equiv_{17 \cdot 31} 787$ *und* $z \equiv_{17 \cdot 31} -267$.

Existenz einer Lösung

Satz (Chinesischer Restsatz)

Seien $r, m \in \mathbb{Z}^n$ *und* $\mathrm{ggT}(m_i, m_j) = 1$ *für* $i \neq j$. *Dann existiert ein* $z \in \mathbb{Z}$ *mit*

$$z \equiv_{m_i} r_i \quad \text{für } i = 1, \ldots, n. \tag{30}$$

Die Menge aller Lösungen von (30) *in* $\mathbb{Z}$ *ist dann durch* $[z]_{m_1 \cdot \ldots \cdot m_n}$ *gegeben.*

Lemma (Chinesischer Restsatz für 2 Kongruenzen)

Für $r_1, m_1, r_2, m_2 \in \mathbb{Z}$ *mit* $\mathrm{ggT}(m_1, m_2) = 1$ *existiert ein* $z' \in \mathbb{Z}$ *mit* $z' \equiv_{m_1} r_1$ *und* $z' \equiv_{m_2} r_2$. *Weiters gilt für alle* $z \in \mathbb{Z}$

$$z \equiv_{m_1} r_1 \text{ und } z \equiv_{m_2} r_2 \iff z \equiv_{m_1 m_2} z'. \tag{31}$$

Beweis Lemma

Jede Kongruenz $z \equiv_m r$ ist gleichbedeutend mit der Existenz eines $x \in \mathbb{Z}$ mit $z = mx + r$. Die Existenz von z' ist daher äquivalent zur Existenz von $x', y' \in \mathbb{Z}$ mit $m_1x' + r_1 = z' = m_2y' + r_2$. Das wiederum entspricht der Lösbarkeit der diophantischen Gleichung $m_1x' - m_2y' = r_2 - r_1$, die wegen $\mathrm{ggT}(m_1, m_2) = 1$ und $1|(r_2 - r_1)$ gegeben ist. Zum Nachweis von (31) sei $z \in \mathbb{Z}$ beliebig aber fix. Falls $z \equiv_{m_1} r_1$ und $z \equiv_{m_2} r_2$ gilt, so existieren $x, y \in \mathbb{Z}$ mit $m_1x + r_1 = z = m_2y + r_2$, womit

$$z - z' = m_1(x - x') = m_2(y - y')$$

gilt. Wegen $\mathrm{ggT}(m_1, m_2) = 1$ teilt m_2 aber $x - x'$, also $x - x' = m_2q$ für ein geeignetes $q \in \mathbb{Z}$, sodass schlussendlich $z - z' = m_1m_2 \cdot q$ und damit $z \equiv_{m_1m_2} z'$ ist. Gilt umgekehrt $z \equiv_{m_1m_2} z'$, so ist für ein passendes $q \in \mathbb{Z}$

$$z = z' + m_1m_2 \cdot q = m_1(x' + m_2q) + r_1, \text{ d.h. } z \equiv_{m_1} r_1.$$

Analog dazu zeigt man $z \equiv_{m_2} r_2$.

Beweis Chinesischer Restsatz

Wir verwenden Induktion nach n. Für $n = 1$ ist die Aussage klarerweise wahr. Für den Induktionsschritt von $n-1$ auf n seien $n > 1$ und $r, m \in \mathbb{Z}^n$ mit $\text{ggT}(m_i, m_j) = 1$ für $i \neq j$ beliebig aber fix. Dann ist (30) aufgrund des Lemmas äquivalent zu

$$z \equiv_{m_1 m_2} z' \text{ und } z \equiv_{m_i} r_i \text{ für } i = 3, \ldots, n \tag{32}$$

mit $z' \in \mathbb{Z}$ wie im Lemma konstruiert. Da (32) nur mehr aus $n-1$ Kongruenzen besteht und die Module $m_1 m_2$ und m_i (für $i = 3, \ldots, n$) wieder paarweise relativ prim sind, existiert laut Induktionsvoraussetzung so ein $z \in \mathbb{Z}$ und die Menge aller Lösungen von (32) – und damit auch von (30) – ist durch $[z]_{(m_1 m_2) \cdot m_3 \cdot \ldots \cdot m_n} = [z]_{m_1 \cdot \ldots \cdot m_n}$ gegeben.

Chinesischer Restalgorithmus

- Existenzbeweis mit Induktion $\rightsquigarrow$ rekursiver Lösungsalgorithmus.
- In jedem Schritt: Reduktion der Problemgröße mittels Lemma.

Algorithmus *CRAZ*: Chinesischer Restalgorithmus in $\mathbb{Z}$

$n \leftarrow |r|$
if $n = 1$
  $z \leftarrow r_1 \bmod m_1$
else
  $(x', y') \leftarrow$
    $LöseLinDiophant(m_1, -m_2, r_2 - r_1)$
  $z' \leftarrow m_1 \cdot x' + r_1 \bmod m_1 \cdot m_2$
  $r_2 \leftarrow z'$, $m_2 \leftarrow m_1 \cdot m_2$
  $z \leftarrow CRAZ(r_{2:n}, m_{2:n})$
return z

Aufruf:	$CRAZ(r, m)$
Eingabe:	$r, m \in \mathbb{Z}^n$
mit:	$\mathrm{ggT}(m_i, m_j) = 1$
	für $i \neq j$.
Ausgabe:	$z \in \mathbb{Z}$
mit:	$z \equiv_{m_i} r_i$
	für $i = 1, \ldots, n$
	$0 \leq z < m_1 \cdot \ldots \cdot m_n$.

Beispiel

Gesucht sei eine ganze Zahl z mit

$$z \equiv_{17} 5 \quad z \equiv_{31} 12 \quad z \equiv_{23} 11.$$

Zur Lösung dieser Aufgabe rufen wir mit $r = (5, 12, 11)$ und $m = (17, 31, 23)$ den Algorithmus $CRAZ(r, m)$ auf. Im ersten Rekursionsschritt wird eine Lösung $x' = 77$ und $y' = 42$ der diophantischen Gleichung $17x' - 31y' = 7$ berechnet. Daraus ergibt sich mit $m_1 m_2 = 527$ das Zwischenresultat $z' = 17 \cdot 77 + 5 \bmod 527 = 260$. Im rekursiven Aufruf lauten die neuen Eingaben dann $r = (260, 11)$ und $m = (527, 23)$. Die diophantische Gleichung $527x' - 23y' = -249$ besitzt eine Lösung $x' = -2739$ (und $y' = -62748$), aus der sich mit $m_1 m_2 = 12121$ dann $z' = 527 \cdot (-2739) + 260 \bmod 12121 = 11327$ ergibt. Mit $r = (11327)$ und $m = (12121)$ ist der Basisfall der Rekursion erreicht und $CRAZ(r, m)$ liefert letztlich $z = 11327$.