Österreichische Mathematische Gesellschaft

MATHE-BRIEF

April 2015 — Nr. 57

Herausgegeben von der Österreichischen Mathematischen Gesellschaft
http://www.oemg.ac.at/Mathe–Brief ——— mathe–brief@oemg.ac.at

SCHNELLER RECHNEN!

Es ist ein seltsames Phänomen in der Mathematik, dass man manchmal Aufgaben einfacher oder schneller lösen kann, wenn man sie vorher komplizierter macht. Wir sehen uns zwei recht überraschende Beispiele dafür an. Eine Warnung vorab: Es handelt sich nicht darum, jemandem mit Rechenschwäche das Zusammenzählen von 19 und 23 zu erleichtern, sondern um die Erklärung von Techniken, die zweckmässigerweise bei der Arbeit mit großen Zahlen per Computer angewandt werden können. Wir beginnen mit einer Erinnerung an die *modulare Arithmetik*.

1. SCHNELLER RECHNEN MIT GANZEN ZAHLEN

Wenn es 20 Uhr ist und man zählt 5 Stunden dazu, dann erhält man nicht 25 Uhr, sondern 1 Uhr. Man setzt 24 und 0 Uhr gleich und beginnt dann von neuem zu zählen. In ganzen Stunden hat man also nur die Zahlen $0,1,\ldots,23$ und die „Gleichungen“ $24=0$, $25=1$, etc.

Beim Addieren und Multiplizieren dieser Zahlen tut man dies zunächst so wie in den ganzen Zahlen und zählt dann so oft 24 ab, bis man im Bereich $\{0,1,2,\ldots,23\}$ landet. Man kann es auch so sagen: man nimmt den Rest der „gewöhnlichen“ Addition bzw. Multiplikation nach Division durch 24. Man sagt, man *rechnet modulo 24*.

Dort gelten also neue Rechenregeln wie etwa $20+5=1$ und $15\cdot 4=(60)=12$. Es gibt auch Überraschungen wie $6\cdot 8=0$. Auch Subtrahieren macht Sinn: $1-5=20$, weil ja $20+5=1$ ist. Dividieren ist problematischer, weil ja z.B. $12\cdot 3=12\cdot 5=12=12\cdot 1$ ist und man hier sicher nicht durch 12 dividieren kann, um $3=5=1$ zu bekommen. Um den Überblick nicht zu verlieren und keine Verwechslung mit den üblichen Rechenoperationen zu erzeugen, schreiben wir die obigen Rechnungen präziser an:

$$[20+5]_{24}=1,\quad [15\cdot 4]_{24}=12,\quad [6\cdot 8]_{24}=8,\quad [1-5]_{24}=20,\quad \text{und so weiter.}$$

Was man mit 24 tun kann, das kann man auch mit 12 machen (wie bei den Uhrzeiten im angelsächsischen Raum), oder überhaupt mit jeder anderen natürlichen Zahl n. Dann ist das Zahlensystem die Menge $\{0,1,2,\ldots,n-1\}$, versehen mit den zusätzlichen Gleichungen $n=0$, $n+1=1$ und so weiter. Man rechnet „modulo n“.

Es gibt immer unendlich viele Zahlen, die sich modulo n auf dasselbe reduzieren, z.B.

$$[20]_{24}=[44]_{24}=[68]_{24}=[-4]_{24}=\ldots \quad \text{und} \quad [20]_{10}=[0]_{10}=[60]_{10}=[-10]_{10}=\ldots$$

Es gilt $[a]_n=[b]_n$ genau dann, wenn $a-b$ ein ganzzahliges Vielfaches von n ist oder — was auf dasselbe hinausläuft – falls a und b bei Division durch n denselben Rest ergeben. Man sagt, b liegt dann in derselben *Restklasse* wie a modulo n.

Es ist nicht schwierig zu sehen, dass beim Rechen die verschiedenen Vertreter einer Restklasse untereinander austauschbar sind, dass also z.B. gilt:

$$[20+5]_{24} = [44+5]_{24} = [45+29]_{24} = [68-19]_{24} = \dots$$

So, jetzt zum schnelleren Rechnen. Wenn man zum Beispiel 19 und 23 addieren will, ist es egal, ob man dies im Bereich der reellen Zahlen, der rationalen Zahlen oder der natürlichen Zahlen macht. Man kann es auch modulo n rechnen, vorausgesetzt, n ist sicher größer als die erwartete Summe.

In einem weiteren Schritt zerlegen wir das n unserer Wahl in Primzahlpotenzen $n = p_1 \cdot p_2 \cdots p_k$. Zum Beispiel wird $n = 60$ in $60 = 4 \cdot 3 \cdot 5$ zerlegt. Anstatt die gewünschte Rechnung direkt durchzuführen, bestimmen wir nun zuerst die Reste der beteiligten Zahlen modulo $p_1, p_2, \dots$

$$\begin{aligned} &19 \text{ ergibt die Reste } ([19]_4, [19]_3, [19]_5) = ([3]_4, [1]_3, [4]_5), \\ &23 \text{ ergibt die Reste } ([23]_4, [23]_3, [23]_5) = ([3]_4, [2]_3, [3]_5). \end{aligned}$$

Anschließend führt man die Rechnung modulo p_1, modulo p_2 etc. durch:

$$([19+23]_4,\ [19+23]_3,\ [19+23]_5) = ([3+3]_4,\ [1+2]_3,\ [4+3]_5) = ([2]_4, [0]_3, [2]_5).$$

Man kennt nun noch immer nicht das ersehnte Ergebnis „x“ der Addition $9+23$, sondern nur die Reste von x bei Division durch 4, 3 und 5, nämlich 2, 0 und 2. Aber es gibt einen Satz, den sogenannten *Chinesischen Restsatz*, benannt nach dem chinesischen Mathematiker Sun Zi, der bereits vermutlich um das Jahr 250 ein Verfahren zur Berechnung von x (in Spezialfällen) für militärische Anwendungen beschrieb. Es geht so:

Satz (Chinesischer Restsatz)**.** *Angenommen, von einer natürlichen Zahl x kennt man die Reste $y_1 = [x]_{p_1}$ u.s.w. bis $y_k = [x]_{p_k}$ bei Division durch Primzahlpotenzen $p_1, \dots, p_k$, wobei die dazugehörigen Primzahlen alle verschieden sein sollen. Dann findet man x auf die folgende Art und Weise:*

(1) *Man berechne alle $q_i = \frac{n}{p_i}$, wobei $n = p_1 \cdot p_2 \cdots p_k$.*
(2) *Zu jedem q_i suche man ein (garantiert existierendes!) $r_i \in \{1, \dots, p_i - 1\}$ mit $[q_i \cdot r_i]_{p_i} = 1$.*
(3) *$x = y_1 q_1 r_1 + \dots + y_k q_k r_k$ ist die einzige Zahl in $\{0, 1, \dots, n-1\}$ mit den vorgegebenen Resten $y_1, \dots, y_k$.*

In unserem Beispiel mit $x = 19+23$ und $n = 60$, $p_1 = 4$, $p_2 = 3$, $p_3 = 5$ haben wir

$$q_1 = \tfrac{60}{4} = 15, \quad q_2 = \tfrac{60}{3} = 20, \quad q_3 = \tfrac{60}{5} = 12,$$

Die Zahlen r_i finden wir durch Probieren, denn es gibt z.B. für r_1 nur die Möglichkeiten $1, 2, 3$:

$$\begin{aligned} &15 \text{ mal wieviel ist 1 (modulo 4)?} \quad [15 \cdot 3]_4 = [3 \cdot 3]_4 = 1 \implies r_1 = 3, \\ &20 \text{ mal wieviel ist 1 (modulo 3)?} \quad [20 \cdot 2]_3 = [2 \cdot 2]_3 = 1 \implies r_2 = 2, \\ &12 \text{ mal wieviel ist 1 (modulo 5)?} \quad [12 \cdot 3]_5 = [2 \cdot 3]_5 = 1 \implies r_3 = 3. \end{aligned}$$

Damit haben wir die Lösung

$$x = 2 \cdot 15 \cdot 3 + 0 \cdot 20 \cdot 2 + 2 \cdot 12 \cdot 3 = 42.$$

Man wird einwenden, dass diese Rechnung viel komplizierter ist als das direkte Auswerten der Summe $x = 19+23$. Dies trifft jedoch dann nicht mehr zu, wenn die beteiligten Zahlen sehr groß sind, oder wenn man mit denselben Zahlen immer wieder rechnen muss. Hier ist es tatsächlich ein Gewinn, zuerst alle Rechnungen zuerst mit Resten durchzuführen, und erst zum Schluss den doch etwas lästigen Chinesischen Restsatz zu bemühen. Bei komplexen Rechnungen, bei denen sowohl

Speicherplatz als auch Rechenzeit eines Computers an ihre Grenzen kommen, so ist die Verkleinerung der Zahlen, mit denen tatsächlich hantiert werden muss, ein essentieller Beitrag zur Lösung. Es ist auch möglich, die Lösung eines großen Problems auf mehrere parallel arbeitende Rechner aufzuteilen. Eine lange Liste von Anwendungen des Chinesischen Restsatzes findet sich z.B. auf der Webseite *http://mathoverflow.net/questions/10014/applications-of-the-chinese-remainder-theorem*. In der Praxis wird man mit der Wahl von $n = 60$ kaum auskommen. Wählt man jedoch

$$\begin{aligned} n &= 2^5 \cdot 3^3 \cdot 5^2 \cdot 7^2 \cdot 11 \cdot 13 \cdot 17 \cdot 19 \cdot 23 \cdot 29 \cdot 31 \cdot 37 \cdot 41 \cdot 43 \cdot 47 \\ &= 32 \cdot 27 \cdot 25 \cdot 49 \cdot 11 \cdot 13 \cdot 17 \cdot 19 \cdot 23 \cdot 29 \cdot 31 \cdot 37 \cdot 41 \cdot 43 \cdot 47 \approx 3 * 10^{21}, \end{aligned}$$

so kann man im ganzen Zahlbereich von 0 bis $3 \cdot 10^{21}$ unbeschränkt addieren, subtrahieren und multiplizieren, ohne mit einzelnen Zahlen größer als 48 hantieren zu müssen, bevor abschließend der Chinesische Restsatz angewendet wird.

2. Schnelles Multiplizieren von Polynomen

Als der Autor dieses Briefes die folgende Methode, zwei Polynome p, q zu multiplizieren, zum ersten Mal sah, dachte er, das wäre so ziemlich die ungeschickteste Methode, dies zu tun:

(1) Man wähle ausreichend viele Stellen $x_0, \ldots, x_n$ und berechne Werte $p(x_i) = u_i$ und $q(x_i) = v_i$. Dabei muss die Anzahl der Stellen mindestens $n+1 = \text{Grad}(p) + \text{Grad}(q) + 1$ sein.
(2) Durch die Lagrangesche Interpolationsformel suche man ein Polynom r, das an den Stellen x_i die Werte $u_i \cdot v_i$ annimmt. Dann gilt $r = p \cdot q$.

Dazu ein paar Bemerkungen:

- Ist $p = a_0 + a_1 x + a_2 x^2 + \ldots + a_m x^m$, so berechnet man für eine Zahl c den Funktionswert $p(c)$ am besten durch $p(c) = a_0 + c\big(a_1 + c(a_2 + c(\ldots))\big)$.
- Die Lagrangesche Interpolationsformel zum Bestimmen des Polynoms r aus den Werten $w_0 = r(x_0), \ldots, w_n = r(x_n)$ lautet

$$r = w_0 l_0 + \cdots w_n l_n, \quad \text{wobei} \quad l_i(x) = \frac{(x-x_0)\,(x-x_1)\,\cdots\,\cancel{(x-x_i)}\cdots(x-x_n)}{(x_i-x_0)(x_i-x_1)\cdots\cancel{(x_i-x_i)}\cdots(x_i-x_n)}.$$

- Es ist eine gute Idee, mit jeder Stelle x_i auch die Stelle $-x_i$ zu nehmen. Denn sind $\text{ger}(p)$ die Summe der Terme in p mit geradem Exponenten und $\text{unger}(p)$ die Summer der restlichen Terme, so ist $p(c) = \text{ger}(c) + \text{unger}(c)$ und $p(-c) = \text{ger}(c) - \text{unger}(c)$. Durch $\text{ger}(c)$ und $\text{unger}(c)$ kann man also gleich die zwei Werte p(c) und $p(-c)$ bekommen.

Man kann also das obige Polynom p entweder durch seine Koeffizienten $a_0, \ldots, a_m$, oder alternativ durch seine Werte $u_0, \ldots, u_m$ an Stellen $x_0, \ldots, x_m$ eindeutig charakterisieren. Die zweite Folge nennt man die Spektralform von p. Der Unterschied zwischen den beiden Darstellungen wird besonders deutlich, wenn man die zeitraubende Multiplikation von Polynomen (welche durch ihre Koeffizienten gegeben sind) vergleicht mit der direkt möglichen Multiplikation von Werten:

Polynom	Koeffizienten	Werte
p	$(a_0, a_1, a_2, \ldots)$	$(u_0, u_1, u_2, \ldots)$
q	$(b_0, b_1, b_2, \ldots)$	$(v_0, v_1, v_2, \ldots)$
$p+q$	$(a_0+b_0,\ a_1+b_1,\ a_2+b_2, \ldots)$	$(u_0+v_0,\ u_1+v_1,\ u_2+v_2, \ldots)$
$p \cdot q$	$(a_0b_0,\ a_0b_1+a_1b_0,\ a_0b_2+a_1b_1+a_2b_0,\ \ldots)$	$(u_0 \cdot v_0, u_1 \cdot v_1, u_2 \cdot v_2, \ldots)$

G. Pilz