Algorithms in Number Theory

Jonathan L.F. King
University of Florida, Gainesville FL 32611-2082, USA
15 April, 2022 (at *08:50*)

Iterated Lightning-bolt (Euclidean algorithm)

Fix integers J_0 and J_1, and set $D := \text{GCD}(J_0, J_1)$. A pair (S,T) of integers is "a ***Bézout pair*** for J_0, J_1" if

1a: $$SJ_0 + TJ_1 \;=\; D\,.$$

Bézout's lemma says: *There always exists a Bézout pair.* (Alternative term: S and T are ***Bézout multipliers***.)

A Bézout pair (S,T) is not unique; it is (except in the boring $J_0{=}0{=}J_1$ case) part of a one-parameter family

1b: $$S_{\langle k\rangle} \;:=\; S + \Big[k\cdot\frac{J_1}{D}\Big] \quad\text{and}$$
$$T_{\langle k\rangle} \;:=\; T - \Big[k\cdot\frac{J_0}{D}\Big]\,,$$

of Bézout pairs $(S_{\langle k\rangle}, T_{\langle k\rangle})$, for each $k \in \mathbb{Z}$.

1c: *Exercise.* Prove that (1b) describes *all* the Bézout pairs for J_0,J_1. □

GCD of several integers. Given a list of integers, $\vec{\mathbf{J}} = (J_0, J_1, \ldots, J_L)$, use

2a: $$\text{GCD}(J_0, J_1, \ldots, J_L) \text{ or } \text{GCD}(\vec{\mathbf{J}})$$

to denote the greatest common divisor, D, of the list. Our goal is to simultaneously compute D and a Bézout-tuple $\vec{\mathbf{s}} := (S_0, \ldots, S_L)$ such that

2b: $$\sum_{\ell=0}^{L} [S_\ell \cdot J_\ell] \;=\; D\,.$$

We'll accomplish this with L applications of LBolt:

$$D \overset{\text{note}}{=\!=} \text{GCD}\Big(\ldots\text{GCD}(\text{GCD}(J_0,J_1),J_2)\ldots,J_L\Big)\,.$$

> Algorithm: From integers $\vec{\mathbf{J}} = (J_0, J_1, \ldots, J_{L-1}, J_L)$, set
> $$C \;:=\; \text{GCD}(J_0, J_1, \ldots, J_{L-1}) \quad \textit{and}$$
> $$D \;:=\; \text{GCD}(J_0, J_1, \ldots, J_{L-1}, J_L) \overset{\text{note}}{=\!=} \text{GCD}(C, J_L)\,.$$
> Apply LBolt $L{-}1$ times to produce integers $\mathsf{v}_0, \ldots, \mathsf{v}_{L-1}$ with $\sum_{\ell=0}^{L-1}[\mathsf{v}_\ell \cdot J_\ell] = C$, and an L^{th} time to produce $\alpha,\beta \in \mathbb{Z}$ with $\alpha C + \beta J_L = D$. Then
>
> 2c: $$S_0 := \alpha\mathsf{v}_0,\; S_1 := \alpha\mathsf{v}_1,\; \ldots,\; S_{L-1} := \alpha\mathsf{v}_{L-1},\; S_L := \beta,$$
>
> gives a tuple $\vec{\mathbf{s}}$ satisfying (2b).

Proof. From the above defns of $\vec{\mathbf{v}}$, and of α and β,

$$D \;=\; \alpha C + \beta J_\ell \;=\; \Big[\alpha \cdot \sum_{\ell=0}^{L-1}[\mathsf{v}_\ell \cdot J_\ell]\Big] + \beta J_L$$
$$=\; \Big[\sum_{\ell=0}^{L-1} \alpha\mathsf{v}_\ell \cdot J_\ell\Big] + \beta J_L\,.$$ ♦

Shorthand. Given two equal-length tuples of numbers, $\vec{\mathbf{a}} = (a_0, a_1, \ldots, a_N)$ and $\vec{\mathbf{c}} = (c_0, c_1, \ldots, c_N)$, define their ***dot product*** to be

$$\vec{\mathbf{a}} \bullet \vec{\mathbf{c}} \;:=\; \sum_{n=0}^{N} a_n \cdot c_n\,.$$

Worked LBolt. Consider $\vec{\mathbf{J}} := (525, 150, 350, 210)$. Using 3 applications of LBolt, we will compute a Bézout tuple $\vec{\mathbf{s}}^3$ such that $\vec{\mathbf{s}}^3 \bullet \vec{\mathbf{J}} = \text{GCD}(\vec{\mathbf{J}})$.

> We LBolt to compute gcd and multipliers:
> $$D_1 \;:=\; \text{GCD}(J_0, J_1) \;=\; \text{GCD}(525, 150) \;=\; 75\,;$$
> $$(\alpha,\beta) \;:=\; (1,\text{-}3)\,.$$
> Setting $\vec{\mathbf{s}}^1 := (1,\text{-}3)$, then, $\vec{\mathbf{s}}^1 \bullet (525,150) = 75$.

> Apply the algorithm again to produce
> $$D_2 \;:=\; \text{GCD}(D_1, J_2) \;=\; \text{GCD}(75, 350) \;=\; 25\,;$$
> $$(\alpha,\beta) \;:=\; (5,\text{-}1)\,.$$
> Multiply $\alpha \cdot \vec{\mathbf{s}}^1 \overset{\text{note}}{=\!=} (5, \text{-}15)$, then adjoin β, to produce $\vec{\mathbf{s}}^2 := (5,\text{-}15,\text{-}1)$. So now $\vec{\mathbf{s}}^2 \bullet (525,150,350) = 25$.

> A third application of LBolt gives
> $$D_3 \;:=\; \text{GCD}(D_2, J_3) \;=\; \text{GCD}(25, 210) \;=\; 5\,;$$
> $$(\alpha,\beta) \;:=\; (17,\text{-}2)\,.$$
> Multiply $\alpha \cdot \vec{\mathbf{s}}^2 \overset{\text{note}}{=\!=} (85,\text{-}255,\text{-}17)$, then adjoin β, to produce $\vec{\mathbf{s}}^3 := (85,\text{-}255,\text{-}17,\text{-}2)$.

The upshot is that

$$\vec{\mathbf{s}}^3 \bullet \vec{\mathbf{J}} \overset{\text{def}}{=\!=} \vec{\mathbf{s}}^3 \bullet (525,150,350,210)$$
$$=\; 5 \;=\; \text{GCD}(\vec{\mathbf{J}})\,,$$

as desired.

Remark. Each of the three Bézout (α,β) pairs is actually part of a 1-parameter family specified by (1b). It follows that the above $\vec{\mathbf{s}}^3$ is just one member of a

3-parameter family of (integer) 4-tuples $\vec{\mathbf{s}}$ that satisfy $\vec{\mathbf{s}}^3 \bullet \vec{\mathbf{J}} = \text{GCD}(\vec{\mathbf{J}})$.

In other words, there is an injective (i.e, 1-to-1) function $f{:}\mathbb{Z}{\times}\mathbb{Z}{\times}\mathbb{Z}{\to}\mathbb{Z}{\times}\mathbb{Z}{\times}\mathbb{Z}{\times}\mathbb{Z}$ with the property that

$$f(a,b,c) \bullet \vec{\mathbf{J}} \quad = \quad \text{GCD}(\vec{\mathbf{J}}),$$

for each triple a,b,c of integers. □

Solving a linear congruence

Having fixed a *modulus* $M \in \mathbb{Z}_+$, as well as a *coefficient* and *target* B,$T \in \mathbb{Z}$, our goal is to find all solutions x to

3: $$B{\cdot}x \;\equiv_M\; T, \quad \text{where } x \in [0\,..\,M).$$

Our algorithm has three STEPs.

This congruence has a solution IFF there exists a *pair* (x,k) solving eqn

3∗: $$B{\cdot}x + M{\cdot}k = T, \quad \text{where } x,k \in \mathbb{Z}.$$

Evidently $D := \text{GCD}(B,M)$ divides LhS(3∗). Hence if $T \nmid D$, then (3∗) has no soln-pair. Whence

STEP A: If $D := \text{GCD}(B,M)$ fails to divide T, then (3) has no soln. Else, define

$$\beta := \frac{B}{D},\; \mu := \frac{M}{D} \text{ and } \tau := \frac{T}{D}$$

and study this "reduced congruence":

4: $$\beta{\cdot}y \;\equiv_\mu\; \tau, \quad \text{where } y \in [0\,..\,\mu).$$

We have gained that $\boxed{\beta \perp \mu}$.

STEP B: Use LBolt to compute a mod-μ multiplicative-inverse, I, of β; so $I{\cdot}\beta \equiv_\mu 1$. Thus

$$y \;\equiv_\mu\; I{\cdot}\beta{\cdot}y \;\equiv_\mu\; I{\cdot}\tau\,.$$

Let y_0 be the unique value in $[0\,..\,\mu)$ st. $y_0 \equiv_\mu I{\cdot}\tau$.

This y_0 is in the *unique* mod-μ residue class solving (4). But mod-M, this residue class splits into D many residue classes. So here is the last step:

STEP C: The D many solutions to (3) are

$$x \;=\; y_0,\, y_1,\, y_2,\, y_3,\, \ldots,\, y_{D-2},\, y_{D-1}\,,$$

where $y_k := y_0 + [k\mu]$.

A worked example. I use an arrow over a letter to abbreviate a sequence, e.g

$$\vec{\mathbf{b}} := (b_0, b_1, b_2, \ldots)\,.$$

We consider

$$35x \;\equiv_{21}\; 55\,.$$

Let's apply STEP A. Since $\text{GCD}(35,21) \overset{\text{note}}{=\!=} 7$ does not divide 55, the above congruence has no soln. [The same computation shows that congr. $21x \equiv_{35} 55$ has no solution.]

A congruence with solns. Consider congruence

3′: $$33{\cdot}x \;\equiv_{114}\; 18\,, \quad \text{where } x \in [0\,..\,114).$$

For STEP A, we compute just the $\vec{\mathbf{r}}$ and $\vec{\mathbf{q}}$ columns:

n	r_n	q_n
0	114	▬
1	33	3
2	15	2
3	3	5
4	0	∞

Since $D := \text{GCD}(33,114) \overset{\text{note}}{=\!=} 3$ divides the target, 18, we divide each of the numbers in (3′) by $D{=}3$ to obtain the reduced congruence

4′: $$11{\cdot}y \;\equiv_{38}\; 6, \quad \text{where } y \in [0\,..\,38).$$

For STEP B, we compute (using $\vec{\mathbf{q}}$) just[♡1] the $\vec{\mathbf{t}}$ column. (Note: We have $\vec{\mathbf{q}}$ from the previous table.)

n	r_n	q_n	s_n	t_n
0	38	▬	1	0
1	11	3	0	1
2	5	2	1	-3
3	1	5	-2	7
4	0	∞	11	-38

So the mod-38 reciprocal of 11 is 7. From STEP B, then,

$$y \;\equiv_{38}\; 7 \cdot 6 \;=\; 42 \;\equiv_{38}\; 4\,.$$

So we set $\boxed{y_0 := 4}$.

[♡1] We do not need to compute $\vec{\mathbf{s}}$ nor $\vec{\mathbf{r}}$. Of course, the new $\vec{\mathbf{r}}$ is just the old $\vec{\mathbf{r}}$ divided by D. I have grayed-out the superfluous columns.

Finally, Step C tells us that these three,

$$\mathbf{4}, \quad 4+38 \overset{\text{note}}{=\!=} \mathbf{42}, \quad 42+38 \overset{\text{note}}{=\!=} \mathbf{80}$$

are the $D{=}3$ many solutions to $(3')$.

Checking. We calculate:

$$\begin{aligned} 33\cdot\mathbf{4} &= 132 &&= 1{\cdot}114 + 18\,; \\ 33\cdot\mathbf{42} &= 1386 &&= 12{\cdot}114 + 18\,; \\ 33\cdot\mathbf{80} &= 2640 &&= 23{\cdot}114 + 18\,. \end{aligned}$$

Copasetic!

Filename: `Problems/NumberTheory/nt-algorithms.latex`
As of: *Thursday 24Sep2009.* Typeset: *15Apr2022* at *08:50.*