A bagatelle on Euclid's Algorithm

If $X \underline{\text{gcd}}\, Y = 1$, the Diophantine equation

$$(0) \qquad a, b:\ aX + bY = 1$$

is solvable. I am almost certain that the first proof I saw of this theorem was non-constructive and of the following structure. Consider the Y remainders $(0 \le i < Y)$

$$r_i = (i.X)\ \underline{\text{mod}}\ Y \qquad (0 \le r_i < Y) .$$

Since

$$\begin{array}{ll}
 & r_i = r_j \\
= & \quad \{\text{def. and arithmetic}\} \\
 & Y \mid (i.X) - (j.X) \\
= & \quad \{\text{arithmetic}\} \\
 & Y \mid (i-j).X \\
\Rightarrow & \quad \{X \underline{\text{gcd}}\, Y = 1\} \\
 & Y \mid (i-j) \\
= & \quad \{0 \le i, j < Y\} \\
 & i = j
\end{array}$$

,

all the r_i are distinct; because of $0 \le r_i < Y$, each value occurs once, in particular, one of the remainders, r_n say, equals 1 , i.e. for some k , we have

$$nX = kY + r_n = kY + 1 \quad .$$

Thus $a, b := n, -k$ yields a solution of (0)

* * *

My first step is to generalize the theorem by removal of the constant 1 : the Diophantine equation

$$(1) \qquad a, b: \quad aX + bY = q \qquad \text{where} \quad q = X \underline{\text{gcd}}\, Y$$

is solvable. This time we give a constructive proof by actually solving equation (1). On Euclid's Algorithm we superpose the additional invariant

$$(2) \qquad aX + bY = x \quad \wedge \quad cX + dY = y \qquad :$$

```
|[ var x, y, a, b, c, d : int
 ; x, y, a, b, c, d := X, Y, 1, 0, 0, 1
 ; do x > y → x, a, b := x-y, a-c, b-d
    [] y > x → y, c, d := y-x, c-a, d-b
   od {(2) ∧ x=q , hence (1)}
]| .
```

It is just another example of "Proving theorems with Euclid's algorithm." (see [0])

* * *

The above is so elementary that I hesitated to devote an EWD to it. But I think the constructive proof interesting for a number of reasons.

- We should not complain about what I believe to be the general state of affairs that, for the same existence theorem, the constructive proof tends to be longer than the nonconstructive one: after all, it delivers more.

Here, however, the constructive argument seems to me as short—if not shorter— as the nonconstructive argument.

- The constructive argument is beautifully forced. Upon completion the goal (1)

$$aX + bY = q$$

has to follow from $x = q$ and the invariant. What else than $aX + bY = x$ could that be? The second conjunct in (2) is added to maintain the symmetry.

- The Diophantine equation

$$(3) \quad a, b, c:\ aX + bY + cZ = q \quad \text{where} \quad q = X \ \underline{\text{gcd}}\ Y \ \underline{\text{gcd}}\ Z$$

is solvable. This theorem can be proved from the program for the gcd of 3 positive numbers

```
|[ var x, y, z: int
 ; x, y, z := X, Y, Z
 ; do x > y → x := x-y
    [] y > z → y := y-z
    [] z > x → z := z-x
   od
 ]|       ,
```

and now the constructive proof is much more elegant than the nonconstructive argument I could come up with (which suffers from an even more severe destruction of the symmetry).

* * *

Remark To satisfy my curiosity, I looked this theorem up in Courant/Robbins "What is mathematics?". They gave — a sketch of — a constructive argument! One up for them. But their text vividly shows that half a century ago, there was no tradition of an adequate style of describing algorithms. Consequently, the prevailing style of reasoning about algorithms was no better. It is nice to see that there has been progress since 1941. (End of Remark.)

[0] A.J.M. van Gasteren "On the shape of mathematical arguments", LNCS 445, Springer-Verlag

Austin, 10 Sep. 1993

prof. dr. Edsger W. Dijkstra
Department of Computer Sciences
The University of Texas at Austin
Austin, TX 78712-1188
USA