Around Bresenham

We consider the task of plotting a curve in the Euclidean plane by marking pixels from a grid covering that plane. More specifically, given a real-valued function f on some finite interval, we wish to mark, for each integer x in that interval, a pixel with integer coordinates (x, y) such that y is as close to f.x as possible. This latter requirement can be formulated as

$$| y - f.x | \leq \tfrac{1}{2} ,$$

or, equivalently, as the conjunction of Q0 and Q1, given by

Q0: $\quad y \leq \frac{1}{2} + f.x$

Q1: $\quad -\frac{1}{2} + f.x \leq y$.

We furthermore assume that f satisfies a kind of "smoothness" property on the interval, to wit

$$0 \leq f.(x+1) - f.x \leq 1 ,$$

which can also be formulated as the conjunction of D0 and D1, given by

D0: $\quad f.x \leq f.(x+1)$

D1: $\quad f.(x+1) \leq 1 + f.x$

It is this property that suggests that the curve be plotted in the order of increasing (or decreasing) value of because the pixel to be marked for x+1 is not too far away from the pixel to be marked for x (thus yielding the nicer picture).

Thus, with the interval given by two integers A and B, $A \le B$, we will consider a marking program of the form

$$x := A$$
$$; \underline{do}\ x \ne B \rightarrow Mark(x, y) ;\ x := x+1\ \underline{od}\ ,$$

and regard it as our task to extend this program with operations on y such that $Q0 \wedge Q1$ is an invariant of the repetition.

The reader may recognize the above problem statement as one from the world of graphics. We wish to emphasize, however, that in this essay neither this specific problem nor its origin is of much concern to us. We are concerned with the development of a program meeting the specification, by paying attention to uninterpreted formulae only, i.e. without further reference —mental or otherwise— to pixels, pictures, or whatever.

* * *

The program above requires two adaptations, one for the initialization of y and one for y's adjustment in the step. As for the initialization, we cannot say much without taking specificities of f into account. For the time being, we may record it as

$$x, y :\quad x = A \ \wedge\ Q0 \wedge Q1\ .$$

For the step, we consider an adaptation of the form $x, y := x+1, y+\xi$, and try to find out for what integer values of ξ this maintains

$Q0 \wedge Q1$. We will deal with the invariances of $Q0$ and $Q1$ in turn.

$Q0$'s invariance

The weakest precondition for $x, y := x+1, y+\xi$ to establish $Q0$ is

$G.\xi: \quad y + \xi \le \frac{1}{2} + f.(x+1)$.

We now investigate for which (integer) values of ξ the required precondition $G.\xi$ is implied by the actual precondition, which is $Q0 \wedge Q1 \wedge D0 \wedge D1$. To that end, we observe

$$
\begin{array}{ll}
 & y + \xi \le \frac{1}{2} + f.(x+1) \\
\Leftarrow & \quad \{ \text{using } Q0 \text{ and the transitivity of } \le \} \\
 & \frac{1}{2} + f.x + \xi \le \frac{1}{2} + f.(x+1) \\
\equiv & \quad \{ \text{arithmetic} \} \\
 & f.x + \xi \le f.(x+1) \\
\Leftarrow & \quad \{ \text{using } D0 \text{ and the transitivity of } \le \} \\
 & f.x + \xi \le f.x \\
\equiv & \quad \{ \text{arithmetic} \} \\
 & \xi \le 0 \quad .
\end{array}
$$

As a result, statement $x, y := x+1, y+\xi$ "automatically" maintains $Q0$ for $\xi \le 0$, but for other values of ξ the statement had better be guarded by $G.\xi$.

Now we expect that, in view of the properties D of f, we will never need to consider increments of y, i.e. values for ξ, outside the range $0, 1$. This expectation will turn out to be true, and, anticipating that, we can summarize the above analysis as

$$
\begin{array}{ll}
(0) & \{Q0\}\{D0\} \\
& \underline{\text{if}}\ G.1 \rightarrow x, y := x+1, y+1 \\
& [\!]\ \text{true} \rightarrow x := x+1 \quad (\text{i.e.}\ \ x, y := x+1, y+0) \\
& \underline{\text{fi}} \\
& \{Q0\}\ .
\end{array}
$$

So much for the invariance of $Q0$.

Q1's invariance

The weakest precondition for $x, y := x+1, y+\xi$ to establish $Q1$ is

$H.\xi: \quad -\frac{1}{2} + f.(x+1) \leq y + \xi$.

As before, we investigate for which values of ξ this is implied by the actual precondition:

$$
\begin{array}{ll}
& -\frac{1}{2} + f.(x+1) \leq y + \xi \\
\Leftarrow & \quad \{\text{using } Q1 \text{ and the transitivity of } \leq\} \\
& -\frac{1}{2} + f.(x+1) \leq -\frac{1}{2} + f.x + \xi \\
\equiv & \quad \{\text{arithmetic}\} \\
& f.(x+1) \leq f.x + \xi \\
\Leftarrow & \quad \{\text{using } D1 \text{ and the transitivity of } \leq\} \\
& 1 + f.x \leq f.x + \xi \\
\equiv & \quad \{\text{arithmetic}\} \\
& 1 \leq \xi \quad .
\end{array}
$$

In summary, we derived

$$
\begin{array}{ll}
(1) & \{Q1\}\{D1\} \\
& \underline{\text{if}}\ \text{true} \rightarrow x, y := x+1, y+1 \\
& [\!]\ H.0 \rightarrow x := x+1 \\
& \underline{\text{fi}} \\
& \{Q1\}\ .
\end{array}
$$

So much for the invariance $Q1$.

Now we combine (0) and (1) into a single program fragment maintaining both $Q0$ and $Q1$:

$\{Q0 \wedge Q1\}\ \{D0 \wedge D1\}$
<u>if</u> $G.1 \rightarrow x, y := x+1, y+1$
▯ $H.0 \rightarrow x := x+1$
<u>fi</u>
$\{Q0 \wedge Q1\}$.

The only remaining proof obligation is to verify that this program fragment does not suffer from the danger of abortion, i.e. we have to check that the disjunction of the guards is implied by the precondition of the if-statement. In order to examine this we expand the guards (while simplifying them):

$G.1: \quad y + \frac{1}{2} \leq f.(x+1)$

$H.0: \quad f.(x+1) \leq y + \frac{1}{2}$,

and, lo and behold, we have $G.1 \vee H.0$, so that there is no danger of abortion. (At the same time, this also confirms our expectation that we need not consider increments of y beyond 0 or 1.)

Collecting the pieces, we have derived that the program depicted in Figure 0 plots the curve as demanded.

* * *

$\{D0 \wedge D1\}\ \{A \leq B\}$
$;\ x, y :\ \ x = A \ \wedge\ Q0 \wedge Q1$
$\{inv\ \ Q0 \wedge Q1\}$
$;\ \underline{do}\ \ x \neq B \rightarrow$
$\quad\quad Mark\ (x, y)$
$\quad\quad ;\ \underline{if}\ G.1 \ \rightarrow\ x, y := x+1,\ y+1$
$\quad\quad\ \ [\!]\ \ H.0 \ \rightarrow\ x := x+1$
$\quad\quad\ \ \underline{fi}$
$\underline{od}$.

So much for the development of this algorithm.

* * *

There is, however, a little bit more to be said. The evaluation of the guards G.1 and H.0 will, in general, demand floating-point arithmetic, and for many functions f there is no escaping it. But for some class of curves —most notoriously the conic sections— there is an opportunity to transform the algorithm so that its execution will require integer arithmetic only. And this is sometimes considered an advantage. We next show such a transformation for the case of a (specific) hyperbola.

Consider the curve given by

$$0 \leq f.x \quad \wedge \quad (f.x)^2 - x^2 = C \quad ,$$

for some (large) positive integer constant C. It is the "positive branch" of a hyperbola. We wish to plot it on the interval $[A, B)$, with A and B integers satisfying $0 \leq A \leq B$.

The reader may verify that, on this interval, f satisfies properties D0 and D1.

We first expand guard G.1, seeking to express it with integer subexpressions only:

$$
\begin{array}{ll}
 & G.1 \\
\equiv & \quad \{ \text{definition of } G \} \\
 & y+1 \leq \frac{1}{2} + f.(x+1) \\
\equiv & \quad \{ \text{arithmetic} \} \\
 & y+\frac{1}{2} \leq f.(x+1) \\
\equiv & \quad \{ \text{both sides are nonnegative; for } y+\frac{1}{2} \\
 & \quad\ \ \text{this is so by } Q1 \} \\
 & (y+\frac{1}{2})^2 \leq (f.(x+1))^2 \\
\equiv & \quad \{ \text{arithmetic and definition of } f \} \\
 & y^2+y+\frac{1}{4} \leq (x+1)^2+C \\
\equiv & \quad \{ \text{arithmetic} \} \\
 & \frac{1}{4} \leq x^2+2*x-y^2-y+C+1 \\
\equiv & \quad \{ \bullet \text{ by } P \text{, given below} \} \\
 & \frac{1}{4} \leq h
\end{array}
$$

For the "complementary" guard H.0, we find

$$H.0 \equiv \frac{1}{4} \geq h \ .$$

The reason for introducing the additional invariant

$$P: \qquad h = x^2+2*x-y^2-y+C+1 \ ,$$

is that the repeated updating of h is assumed to be less costly than the repeated evaluation of P's right-hand side for the successive values of x and y. Furthermore note that h is an integer — as demanded — .

Now we have almost succeeded in expressing G.1 and H.0 in terms of integer expressions only, be it for the occurrence of that $\frac{1}{4}$. Because h is an integer, we can eliminate this $\frac{1}{4}$ at a bargain because of

$$\tfrac{1}{4} \leq h \quad \equiv \quad 1 \leq h$$

and $$\tfrac{1}{4} \geq h \quad \equiv \quad 0 \geq h \,.$$

Thus, we find

$$G.1 \;\equiv\; 1 \leq h \qquad \text{and} \qquad H.0 \;\equiv\; 0 \geq h$$

We now give the final program at once, leaving the standard proof of the invariance of P to the reader. (All that is needed for this is the axiom of assignment.)

```
   x, y := A, round.(sqrt.(A² + C))
 ; h := x² + 2*x - y² - y + C + 1
 ; do  x ≠ B →
         Mark (x,y)
       ; if 1 ≤ h →
              x, y, h := x+1, y+1, h + 2*x - 2*y + 1
         [] 0 ≥ h →
              x, h := x+1, h + 2*x + 3
         fi
   od .
```

The program text can be further embellished, but we leave it at this. For some interesting details concerning the marking we refer to, for instance, [vdS 93].

* * *

The reader that feels like constructing a plotting algorithm himself, may try to do so for, say, a straight line segment in the "first octant" of the plane. The exercise can be quite rewarding, since he will find himself designing the famous algorithm that was invented in the early sixties by J. E. Bresenham [Bre65].

Acknowledgement We thank Rob Hoogerwoord and Anne Kaldeway for their comments.

Postscript This note is a rewrite of WF119, which was written early 1990 because I felt intrigued but not satisfied by [Wi90].

References

[Bre65] J.E. Bresenham. Algorithm for computer control of a digital plotter. IBM Systems Journal, 4(1): 25-30, 1965.

[vdS93] J.L.A. van de Snepscheut. What computing is all about. Springer-Verlag New York, 1993.

[Wi90] N. Wirth. Drawing lines, circles, and ellipses in a raster. In "Beauty is our Business, a Birthday Salute to Edsger W. Dijkstra". Springer-Verlag New York, 1990.

WHJ Feijen
Sterksel
31 August 1995