Subdivision Curve Interrogation Examples - Discussion

Draw

Write a proof of the pseudo-coded routine in the notes for Draw.

This involves writing a definition of what the result should be, and also identifying the Lemmas which the API routines need to satisfy. The actual proofs are the easy part.

- The result needs to be that a set of straight line segments is drawn whose Hausdorff distance from the curve is less than a specified tolerance, which we shall denote by ϵ. For some purposes, we may also require that the set should form a polygon, and that it should be traced from one end to the other. These map into different sets of Lemmas. Don't ask which is right ! Just choose an output definition. Here I take the simplest.

```
Draw(curve:a)
curve:b, c
begin   if      simple(a)
        then    draw(approximation(a))
        else    b, c := halves(a)
                draw(b)
                draw(c)
        endif
end
```

The Lemmas are:-

L1 simple(a) $\rightarrow$ h(a,App(a))$\leq \epsilon$
L2 $b \cup c = a$
L3 after a finite number of steps, simple is true for all pieces.

Given: a curve a

To Prove: $h(a, \cup(\text{calls to draw}) \leq \epsilon$

Construction: We show that
R1 if Draw does not call itself recursively it is correct
R2 Draw making recursive calls to a level n is correct
if Draw making recursive calls to a level $n-1$ is correct.

Proof:

Draw(curve:a)		
curve:b, c		
begin	if	simple(a)
	then	//A1 h(a,App(a))$\leq \epsilon$ by L1
		draw(approximation(a))
		//A2 $\cup(Draws) = App(a)$
		//A3 $h(a, \cup(Draws)) \leq \epsilon$ by A1 and A2
		//This proves R1
	else	$b, c :=$ halves(a)
		//A4 $\cup(b, c) = a$ by L2
		draw(b)
		//A5 $h(b, \cup($calls to draw inside draw$(b)) \leq \epsilon$ by R2
		draw(c)
		//A6 $h(c, \cup($calls to draw inside draw$(c)) \leq \epsilon$ by R2
		//A7 $h(\cup(b, c), \cup(\cup($calls to draw inside draw$(b))$
		$\cup($calls to draw inside draw$(c))) \leq \epsilon$ by prop of h
		//A8 $h(a, \cup($calls to draw$)) \leq \epsilon$ by reduction of A7
		//This proves R2 at level n
	endif	
end		

It is then only necessary to use induction to show that for any finite n the result holds and to combine it with L3 to prove the code.

∎

Planecut

Prove that adding the test on the hull cutting the plane in planecut does not destroy correctness if the rest of the code is correct.

- The assumption is that the required result of the procedure is the set of points in which the curve cuts the plane.

```
planecut(curve:a)
curve:b, c
begin   if      hull(a) cuts the plane
        then    if      simple(a)
                then    planecut(approximation(a))
                else    b, c := halves(a)
                        Union(planecut(b),
                              planecut(c))
                endif
        else    emptyset
        endif
end
```

We require the lemma that

$$a \subset hull(a)$$

.

Call the set of intersection points P, and the plane F.

Then $P \subset a$, so that $P \subset hull(a)$

Thus $hull(a) \cap F = \emptyset \rightarrow P = \emptyset$

■