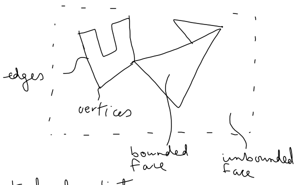
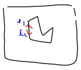
Map overlay

- Planar subdivision (map) – embedding of a graph into $\mathbb{R}^2$



- Store planar subdivision in a dcel : doubly connected edge list.
- In this approach, orientation of edges will be important.



říj 4-15:42

DCEL : 3 tables

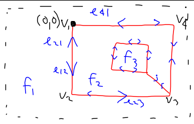


Table for vertices

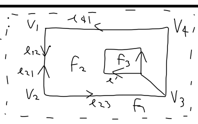
Name of vertex	Co-ordinates	Edge originating at vertex
v_1	$(0,0)$	e_{12}

Also tables for edges & faces.

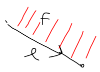
říj 4-16:09

Table for edges

Name of edges	Origin (vertex)	Twin	Next Edge	Previous Edge	Adjacent face
e_{12}	v_1	e_{21}	e_{23}	e_{41}	f_2



- Adjacent face : face to the left of (oriented) edge



- Next(e) : - origin of Next(e) is endpoint of e
  - next(e) has same adjacent face as e
  - no edge between e & next(e) with these properties
- Previous – similar.

See Fig 3.2 in E-Learning.

říj 4-16:15

Table for faces

(See Fig 3.3 in E-Learning)

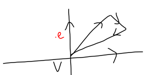
Name of face	1 edge on outer cycle	1 edge on each inner cycle
f_2	e_{23}	e
f_1	none	e_{21}

- Cycle : sequence $(e_1, \dots, e_n)$ of edges with $next(e_i) = e_{i+1}$ & $next(e_n) = e_1$.
- It is a cycle of f if $adj(e_i) = f$ for any e_i in cycle.
- Outer cycle of f if edges e_i lie on outer boundary of f.
- Inner cycle otherwise.
- Each limited (bounded) face has exactly 1 outer cycle.
- Unbounded face has no outer cycle
- All faces can have multiple inner cycles.

říj 4-16:24

Exercise:

Using dcel given a vertex v calculate all edges with origin v in clockwise order.



Find edge e with origin v
twin(e)
$e \mapsto next(twin(e)) \mapsto \dots$

Complexity of planar subdivision/DCEL is number of vertices + edges + faces.

říj 4-16:36

Algorithm for map overlay

S_1 red map .... D_1 DCEL
S_2 blue map .... D_2 DCEL
(See Fig 3.1)

- $S = \text{Overlay}(S_1, S_2)$ – calculate DCEL D for the overlay.

Algorithm has 3 steps:

1) Put tables for vertices & edges of D_1 & D_2 into a single table D. (record colour of edges)
2) Using segment intersection alg, modify tables for vertices & edges.
3) Finally create a table for faces. For each face $f \in D$ find red & blue faces f_1 & f_2 in which f lies.

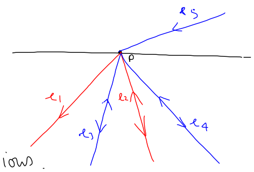
říj 4-16:41

- We give outline - details in Jankü's thesis (see E-Learning).
- Algorithm involve queue Q, binary balanced tree T of line segments (with colour), For each $p \in Q$ sets $L(p), U(p), C(p)$ of coloured line segments on which it lies.
- Add all endpoints to Q
- Various cases to consider.

řij 4-16:48

Given $p \in Q$:
if $C(p) = \emptyset$.

In tables for vertices & edges of D, do not add any new vertices or edges only update next & previous.



Original table: $next(e_2) = e_1$

New Table: $next(e_2) = e_3$

• For $C(p) \neq \emptyset$ see Fig 3.5 of E-Learning.

řij 4-16:54

Faces

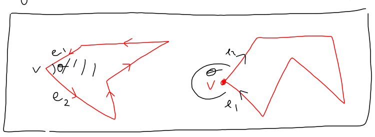
Idea: each bounded face has a unique outer cycle.
Unbounded face has no outer cycles.

- We can define the faces of D to be Outer Cycles $\cup \{C_\infty\}$ — an imaginary outer cycle for unbounded region.

Which cycles of D are inner & which are outer?

řij 4-17:06

- Calc all cycles in D using tables for vertices & edges.
- At a cycle c, choose leftmost vertex V.



Let e_1, e_2 be edges going into & out of V.
Let θ be angle between e_1, e_2 measured over adjacent region.

$\theta < 180°$ ~ Outer cycle ~ $\det\begin{pmatrix} e_{1x} & e_{1y} \\ e_{2x} & e_{2y} \end{pmatrix} > 0$

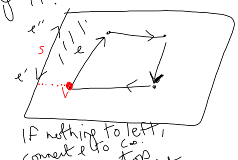
$\theta > 180°$ Inner cycle $\det(\ ) < 0$

řij 4-17:15

- Draw a graph of all the cycles (including C_∞)
- Each connected component will contain exactly one outer cycle (possibly C_∞), & then we'll be able to fill in the face information using it.

The graph

- Choose inner cycle e.
- Find leftmost point v.
- Find closest segment s to its left.
- Determines 2 edges e' & e'', one of which has same adjacent face as e, namely e'.
∴ cycle on which it lies K
- Draw edge from e to K in graph. If K outer, stop. If K inner, repeat.



If nothing to left, connect e to C_∞.

řij 4-17:21

For graph example, see Fig 3.8 in Elearning:
Warning: Some twins appear to be missing.

Complete D: Faces = Outercycles $\cup \{C_\infty\}$
- adj(e) = outer cycle or C_∞ connected to cycle on which e lies.
- Outer cycle$(f) = f$ or if $f = C_\infty$ then outer cycle$(f) = \emptyset$.
- Inner cycles(f): those to which f is connected in the graph.

řij 4-17:36

Identification of faces

- Find for each face f of overlay, a face f_1 in red map & face f_2 in blue map on which f lies.

Two cases: f has a red & blue edge in its cycle the corr. adj. faces in original graph are f_1, f_2.

- Otherwise, if f unbounded - f_1, f_2 unbounded faces in original maps.
- Else, f bounded & single colour (red) in its cycle, See Diagram 3.10 in E-Learning.

říj 4-17:41