CSE520: Computational Geometry
Lecture 6
Planar Graphs

Antoine Vigneron

Ulsan National Institute of Science and Technology

June 15, 2020

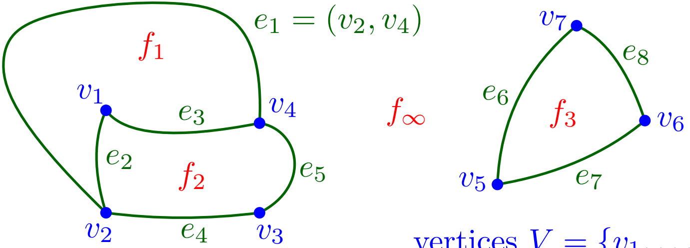
Introduction

- Reference: material taken in part from the Textbook Chapter 2, 3 and 6. Presented slightly differently.

Planar Graphs: Definition

Definition (Planar graph)

A *planar graph* is a graph that can be embedded in $\mathbb{R}^2$: It can be drawn in the plane so that no two edges intersect in their interior.



vertices $V = \{v_1, \ldots, v_7\}$

edges $E = \{e_1, \ldots, e_8\}$

faces $F = \{f_1, f_2, f_3, f_\infty\}$

Properties of Planar Graphs

- A planar graph has one unbounded face f_∞.

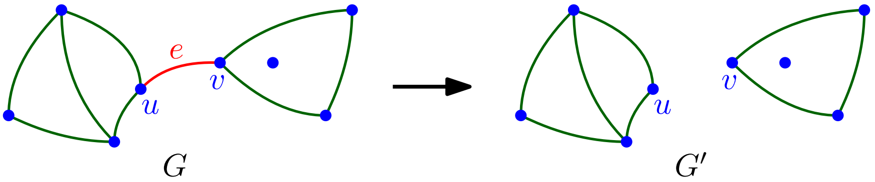
Theorem (Euler's formula)

If G is a planar graph with n vertices, m edges, p faces and c connected components, then $n - m + p - c = 1$. In particular, if G is connected, then $n - m + p = 2$.

- We now prove it by induction on m.
- **Basis step**: When $m = 0$, the graph consists of n isolated vertices.
- So $c = n$, each vertex being one connected component,
- and $p = 1$, because there is only one face: f_∞.
- It follows that $n - m + p - c = n - 0 + 1 - n = 1$.

Properties of Planar Graphs

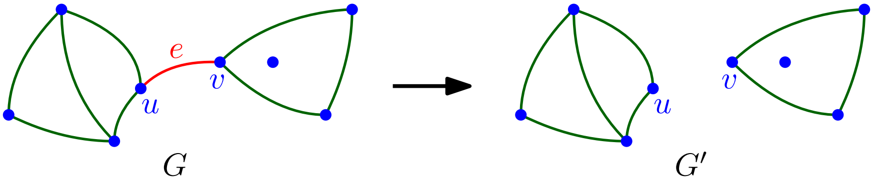
- **Inductive step.** Now suppose $m \geqslant 1$.
- Let $e = (u, v)$ be an edge of G and $G' = G$ be the graph obtained from G by removing e.



- Let n', m', p' and c' be the number of vertices, edges, faces and connected components of G'.
- In the example above, we have $n = 8$, $m = 9$, $p = 4$, $c = 2$, $n' = 8$, $m' = 8$, $p' = 4$ and $c' = 3$.

Properties of Planar Graphs

- **Case 1:** u and v are in different connected components of G'.

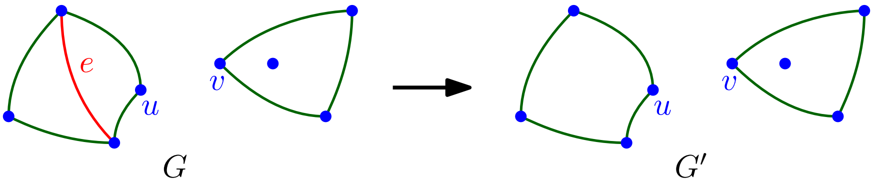


- Then $n' = n$, $m' = m - 1$, $p' = p$ and $c' = c + 1$.
- By inductive hypothesis, we have

$$\begin{aligned} 1 &= n' - m' + p' - c' \\ &= n - (m - 1) + p - (c + 1) \\ &= m - m + p - c. \end{aligned}$$

Properties of Planar Graphs

- **Case 2:** u and v are in the same connected component of G'.



- Then $n' = n$, $m' = m - 1$, $p' = p - 1$ and $c' = c$.
- By inductive hypothesis, we have

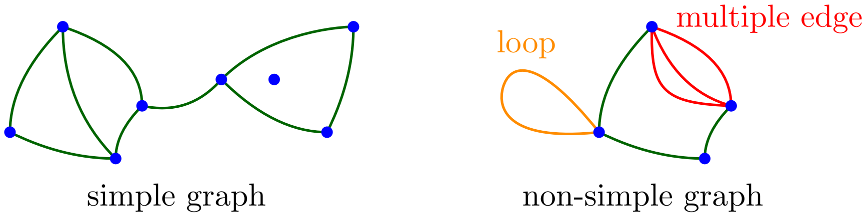
$$\begin{aligned} 1 &= n' - m' + p' - c' \\ &= n - (m-1) + (p-1) - c \\ &= m - m + p - c. \end{aligned}$$

□

Properties of Planar Graphs

Definition

A *simple graph* is a graph without loops or multiple edges.

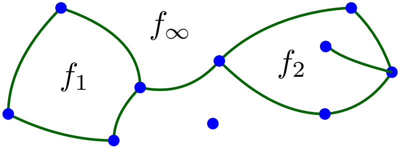


Theorem

If G is a simple planar graph and $n \geqslant 3$, then $m \leqslant 3(n-2)$ and $p \leqslant 2(n-2)$

Properties of Planar Graphs

- Before proving the theorem above, we need this definition.



$\deg(f_1) = 4$
$\deg(f_2) = 6$
$\deg(f_\infty) = 10$

Definition

The degree of a face is the number of edges on its boundary. If an edge appears twice, it is counted twice.

Properties of Planar Graphs

- If $m \leqslant 2$, then we have $m \leqslant 3(n-2)$ because $n \geqslant 3$, and we have $p = 1 \leqslant 2(n-2)$.
- So we may assume that $m \geqslant 3$.
- As there are no multiple edges or loops, it follows that each face has degree $\geqslant 3$.
- Then we have

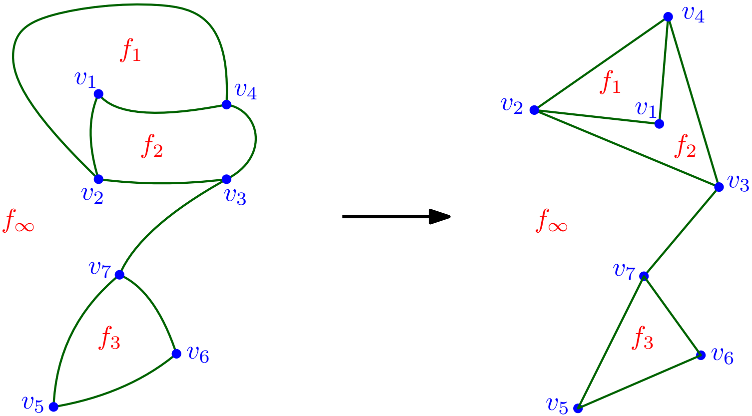
$$2m = \sum_{f \in F} \deg(f) \geqslant 3p.$$

- By Euler's formula, $n - m + p - c = 1$, so $m - p = n - c - 1 \leqslant n - 2$.
- It follows that $\frac{3}{2}p - p \leqslant n - 2$ and thus $p \leqslant 2(n-2)$.
- It also yields $m \leqslant 3(n-2)$. $\square$

Properties of Planar Graphs

Theorem

Every planar graph has a straight line embedding.

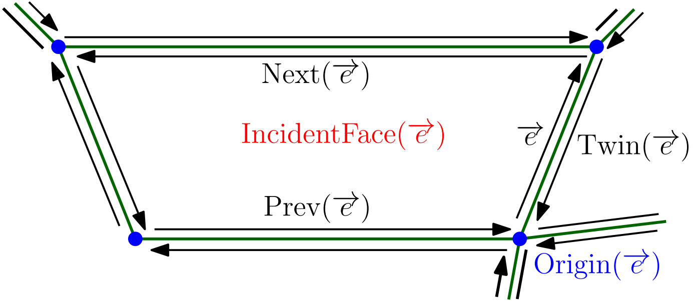


- Not proved in this course.

Planar Straight Line Graphs

Definition (Planar straight line graph)

A planar straight line graph (PSLG) is an embedded planar graph with only straight line edges. It is also called a *planar subdivision*.

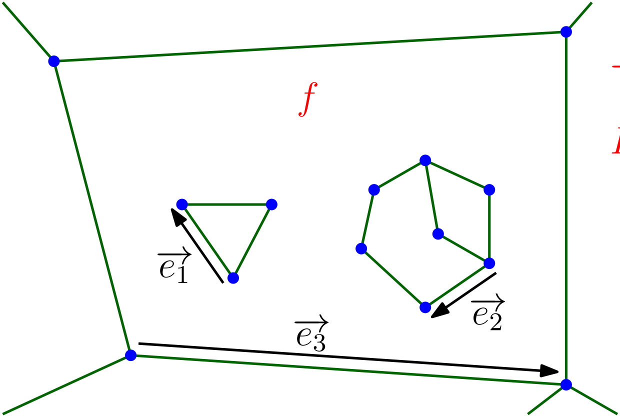


- A data structure: *doubly connected edge list*.
  - Each edge is replaced by two directed half-edges.
  - The half-edges enclosing a face form a counterclockwise cycle.

Doubly Connected Edge List

- Vertex v
  - Coordinates
  - An incident half-edge IncidentEdge$(v) = (v, w)$
- Half edge $\overrightarrow{e}$
  - 3 edges Twin($\overrightarrow{e}$), Next($\overrightarrow{e}$), Prev($\overrightarrow{e}$)
  - Vertex Origin($\overrightarrow{e}$)
  - A face IncidentFace($\overrightarrow{e}$)
- Face f
  - A half-edge $\overrightarrow{e}(f)$ of its exterior boundary
  - A half-edge of the boundary of each hole in f; they are stored in a list $L(f)$

Faces in Doubly Connected Edge Lists



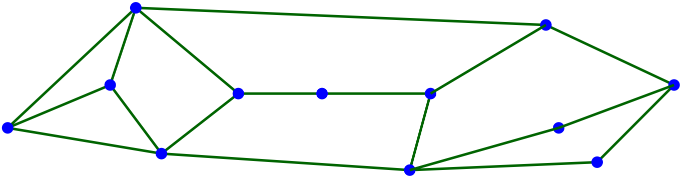
$$\overrightarrow{e}(f) = \overrightarrow{e_3}$$

$$L(f) = \{\overrightarrow{e_1}, \overrightarrow{e_2}\}$$

Special Cases

- *Polyline*: The edges form a chain.

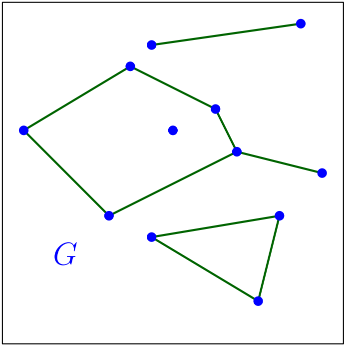
- *Convex subdivision*: All faces are convex, except f_∞.

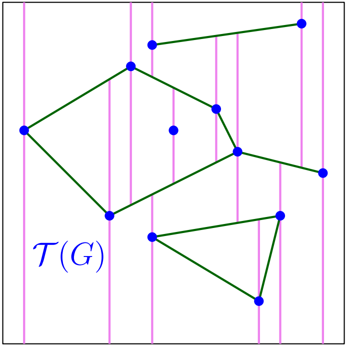


- *Polygon*: a bounded face of a PLSG (see next lecture).

Trapezoidal Map

- Start with a PSLG G.
- The *trapezoidal map* $\mathcal{T}(G)$ is the convex subdivision obtained by drawing vertical edges downward and upward from each vertex.

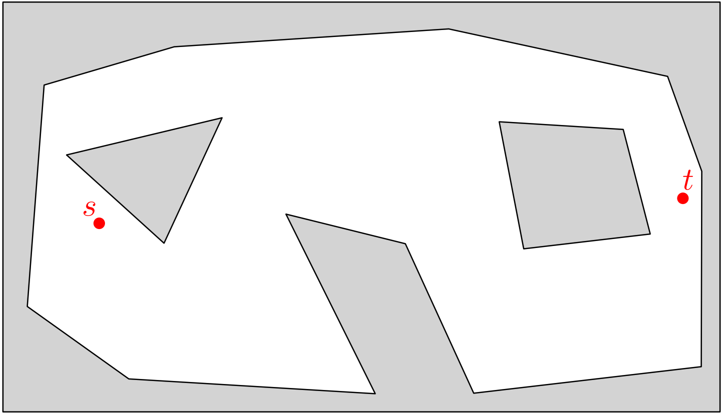




- We draw a bounding box around G so that there is no infinite face, hence all faces of $\mathcal{T}(G)$ trapezoids.

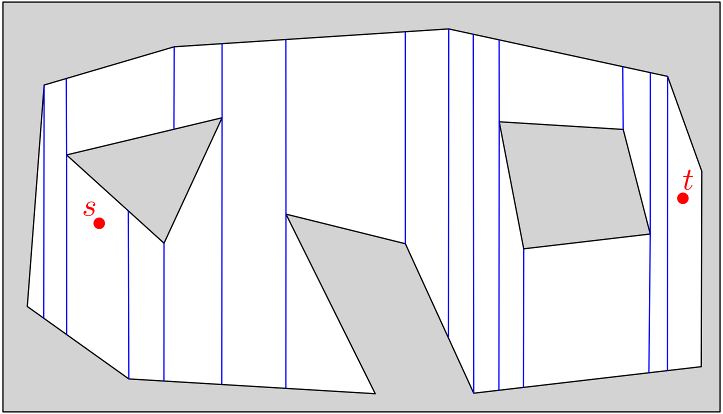
Trapezoidal Map: Application

Robot motion planning: find a path from s to t.



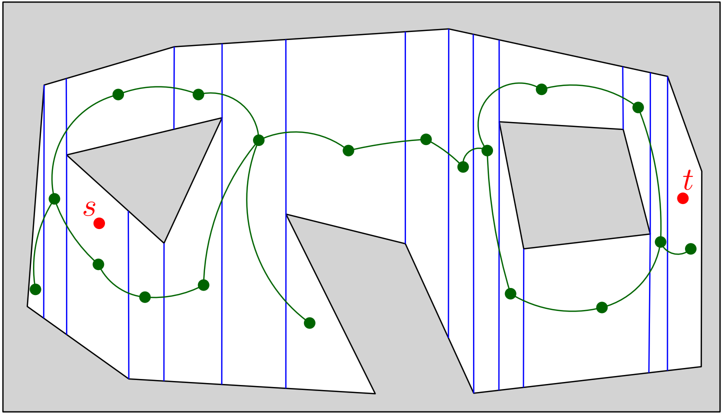
Trapezoidal Map: Application

Compute the trapezoidal map.



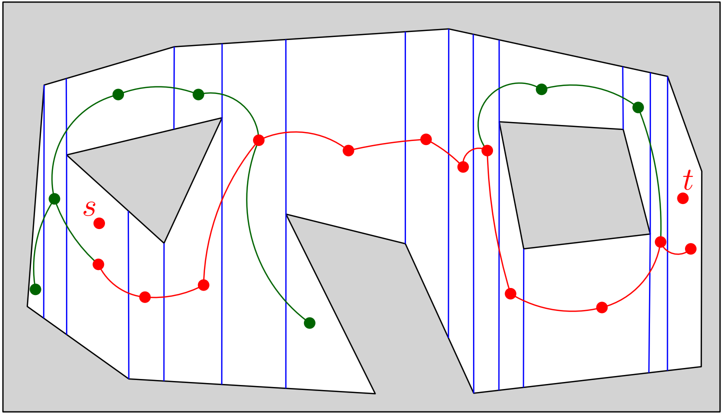
Trapezoidal Pap: Application

Compute the corresponding connectivity graph.



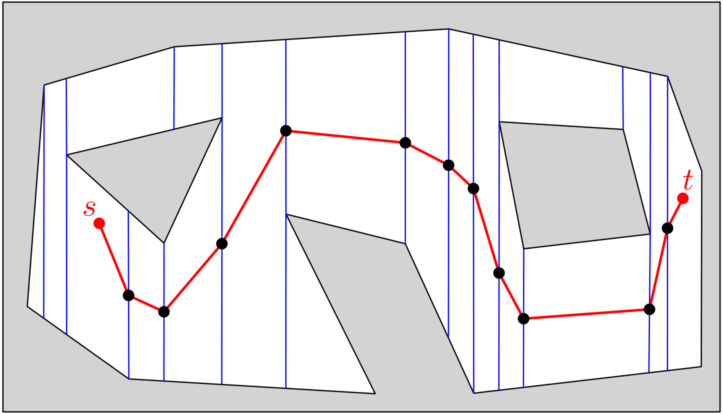
Trapezoidal Map: Application

Find a path in this graph from the cell containing s to the cell containing t.



Trapezoidal Map: Application

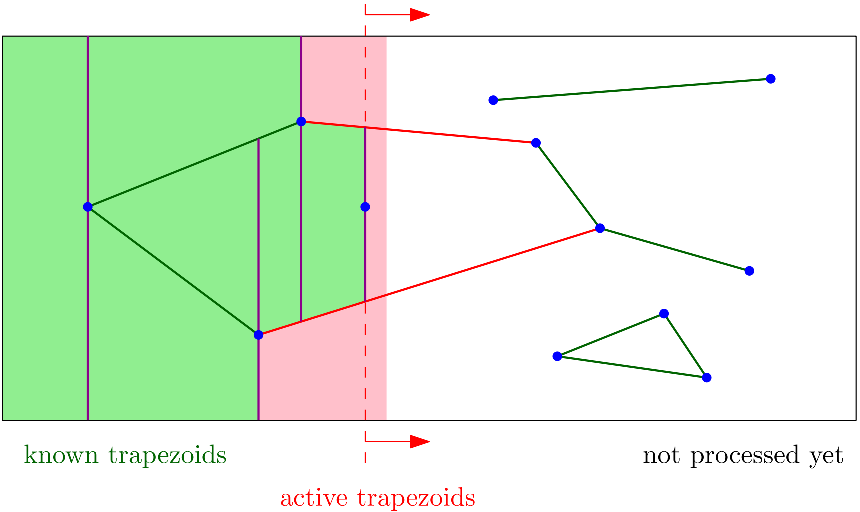
Convert it into a path from s to t through midpoints of vertical edges.



Computing the Trapezoidal Map $\mathcal{T}(G)$

- Assume G has n vertices.
- INPUT: A representation of G (for instance, a doubly connected edge list).
- OUTPUT: A representation of $\mathcal{T}(G)$.
- Idea: We will use plane sweep.
- A modified version of the intersection detection algorithm.
- First step: Sort vertices by increasing x-coordinate.
- An event: The sweep line reaches a vertex of G.

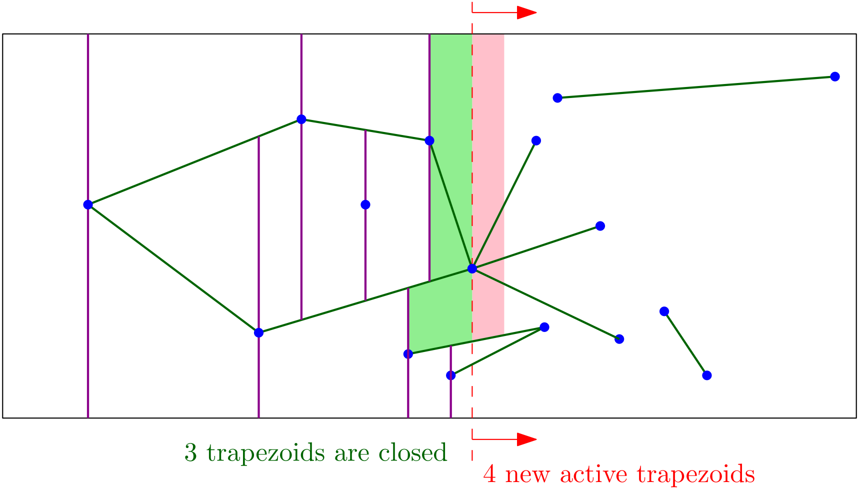
Computing $\mathcal{T}(G)$



Computing $\mathcal{T}(G)$

- Invariants:
  - We know the trapezoids that lie to the left of the sweep line.
  - *Active trapezoids*: Trapezoids that intersect the sweep line.
  - We know the order of the active trapezoids along the sweep line.
  - We know the left, top and bottom edges of each active trapezoid.
- An event: Close some active trapezoids and create new ones.

Computing $\mathcal{T}(G)$



Computing $\mathcal{T}(G)$

- At event i suppose k_i trapezoids are closed or created.
- Event i can be handled in $O(k_i \log n)$ time.
- Amortized analysis.
  - $\mathcal{T}(G)$ has at most $3n$ vertices.
  - So there are $O(n)$ trapezoids.
  - Each trapezoid is created and closed one time only.
  - So $\sum k_i = O(n)$
- Overall, the algorithm runs in $O(n \log n)$ time.