Location-based Routing in Ad hoc Networks

3/3/06

Routing in ad hoc networks

- Routing protocols in communication networks obtain route information between pairs of nodes wishing to communicate.
- Proactive protocols: the protocol maintains routing tables at each node that is updated as changes in the network topology are detected.
- Reactive protocols: routes are constructed on demand. No global routing table is maintained.
  - Ad hoc on demand distance vector routing (AODV)
  - Dynamic source routing (DSR)
- However, both depend on flooding for route discovery.

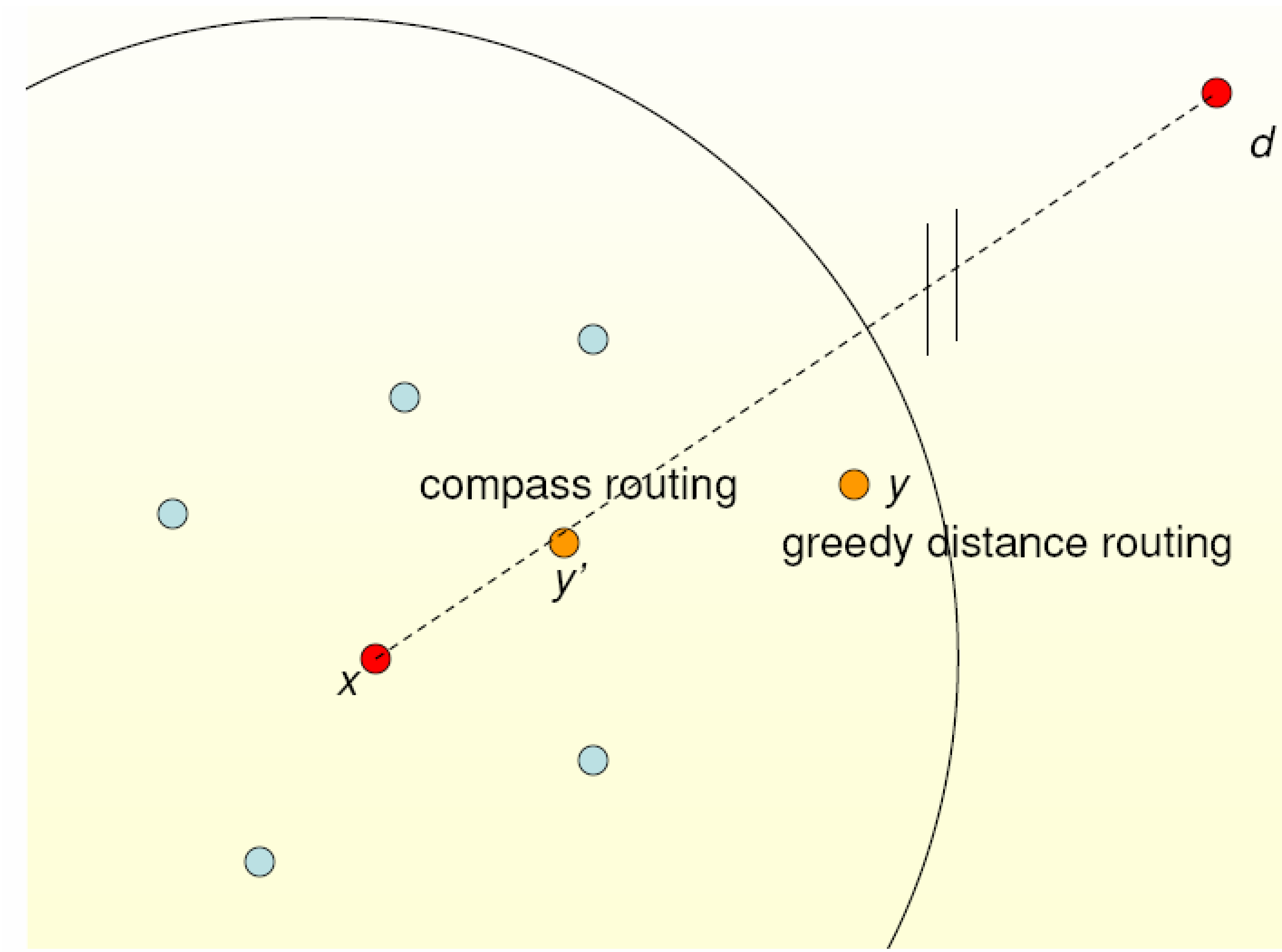
Geographical routing

- Geographical routing uses a node's location to discover path to that route.
- Assumptions:
  - Nodes know their geographical location
  - Nodes know their 1-hop neighbors
  - Routing destinations are specified geographically (a location, or a geographical region)
  - Each packet can hold a small amount (O(1)) of routing information.
  - The connectivity graph is modeled as a unit disk graph.

Geographical routing

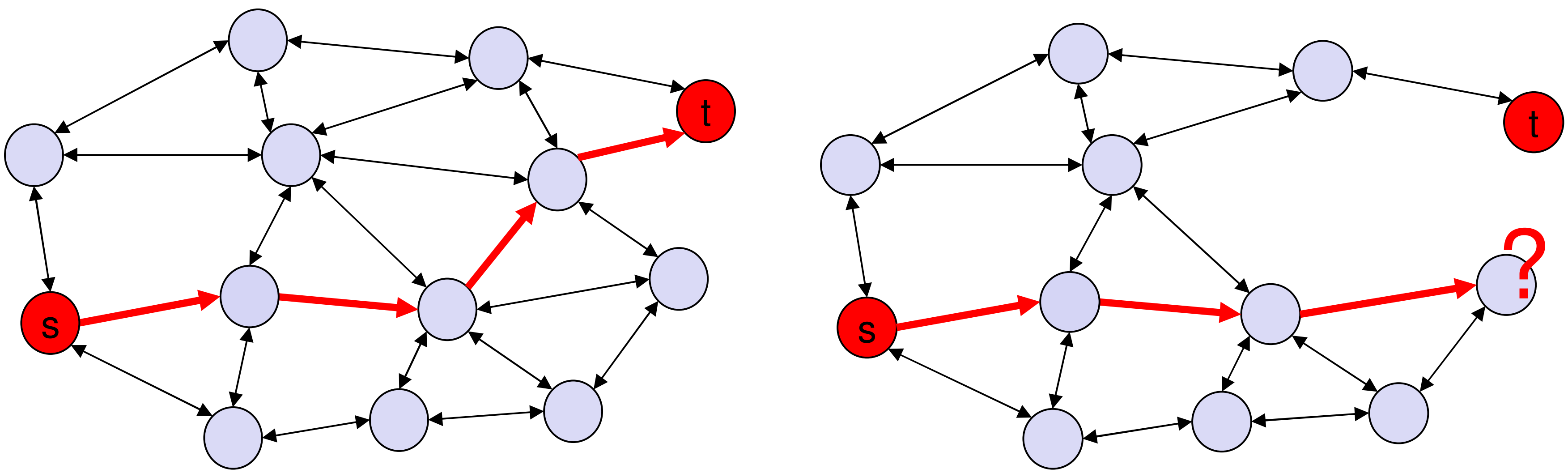
- The information that the source node has
  - The location of the destination node;
  - The location of itself and its 1-hop neighbors.
- Geographical forwarding: send the packet to the 1-hop neighbor that makes most progress towards the destination.
  - No flooding is involved.
- Many ways to measure "progress".
  - The one closest to the destination in Euclidean distance.
  - The one with smallest angle towards the destination: "compass routing".
  - Etc.

Greedy progress



Geographical routing may get stuck

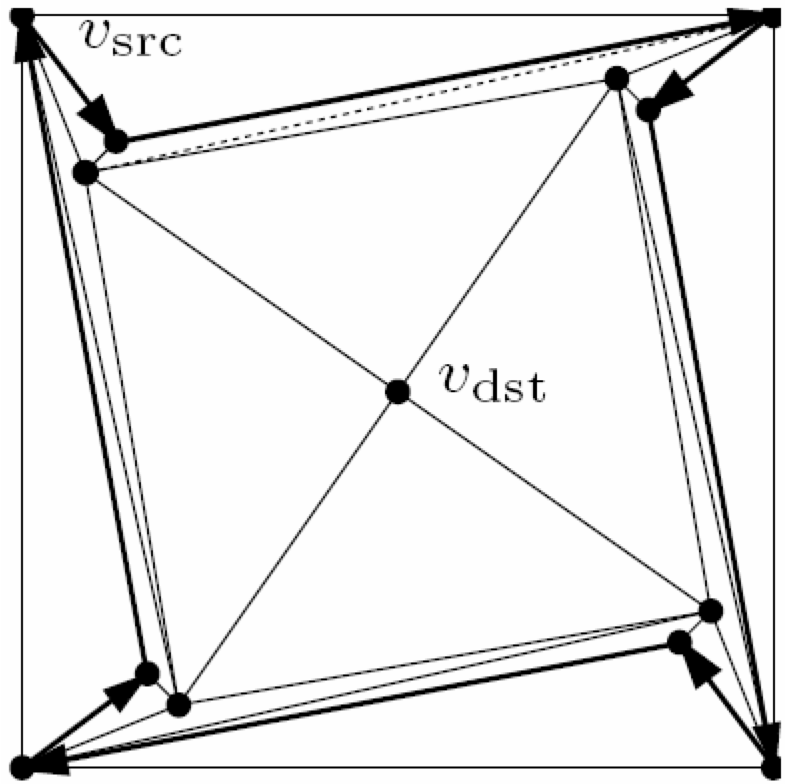
- Geographical routing may stuck at a node whose neighbors are all further away from the destination than itself.



Send packets to the neighbor closest to the destination

Compass routing may get in loops

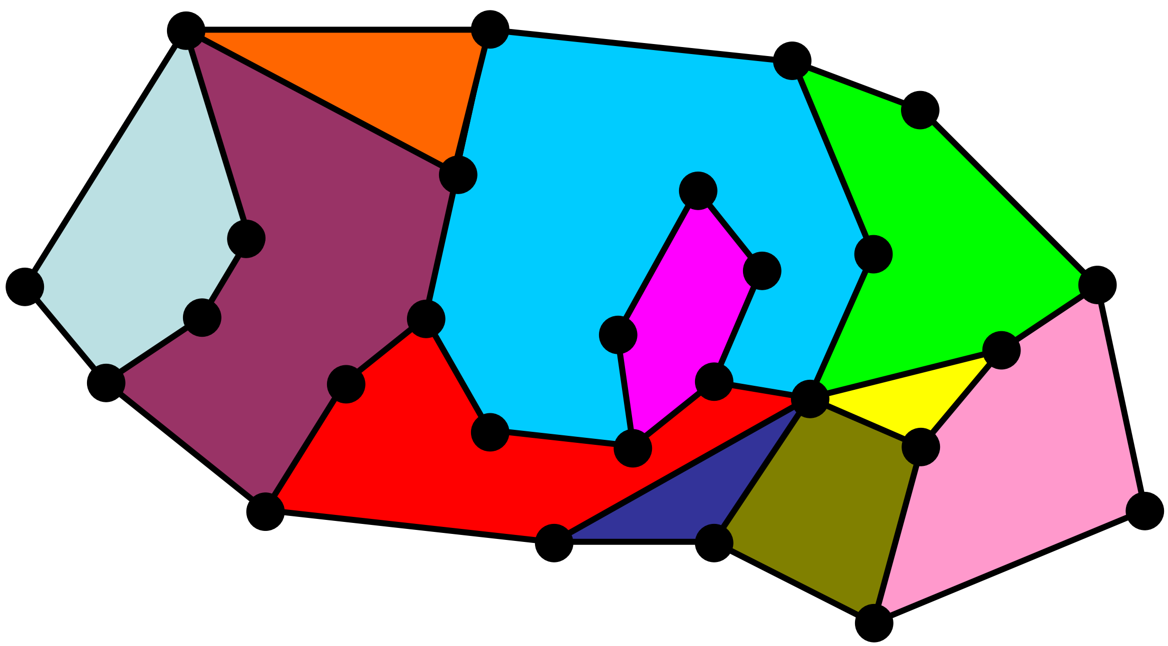
- Compass routing may get in a loop.



Send packets to the neighbor with smallest angle towards the destination

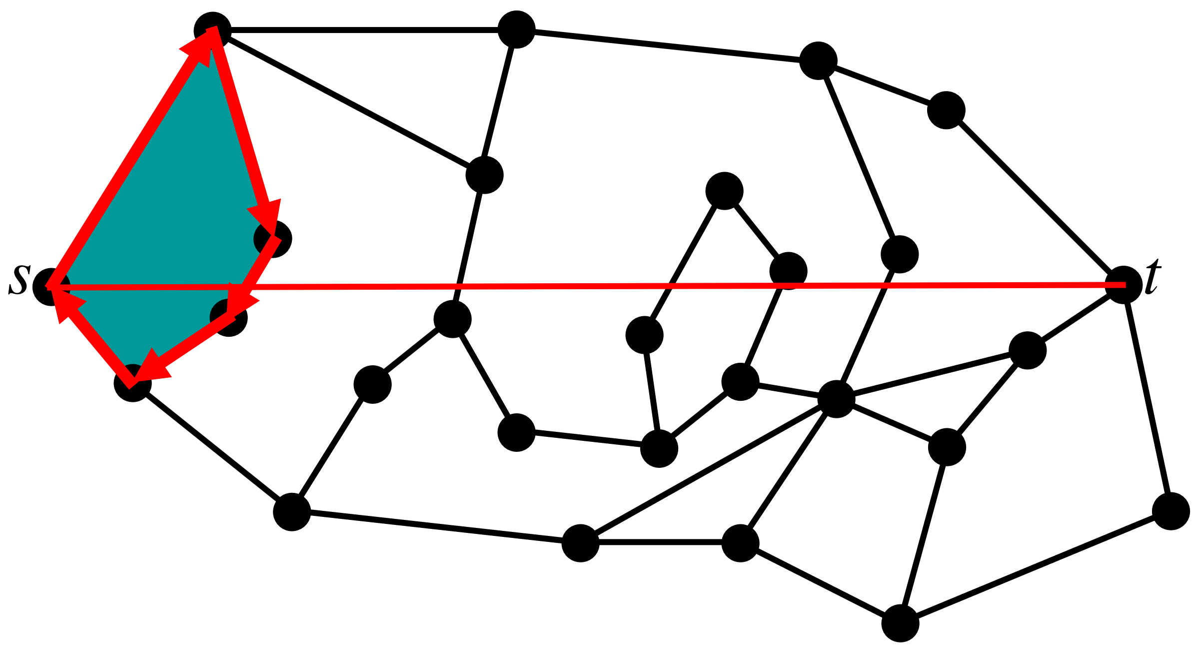
How to get around local minima?

- Use a planar subgraph: a straight line graph with no crossing edges. It subdivides the plane into connected regions called faces.

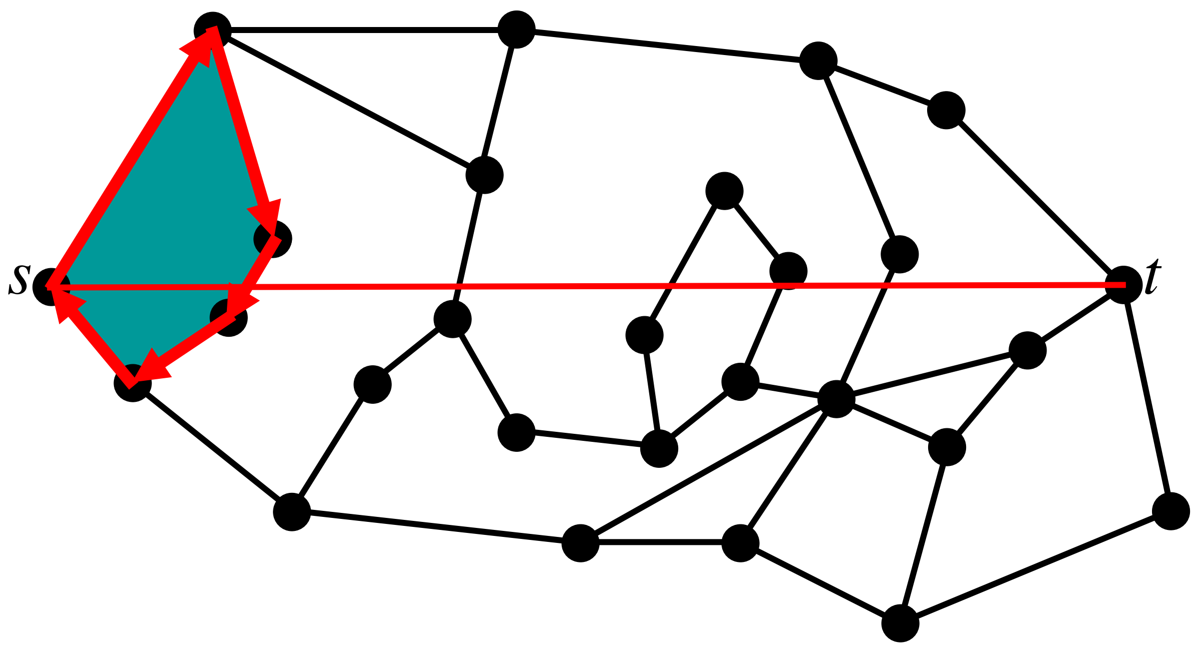


Face Routing

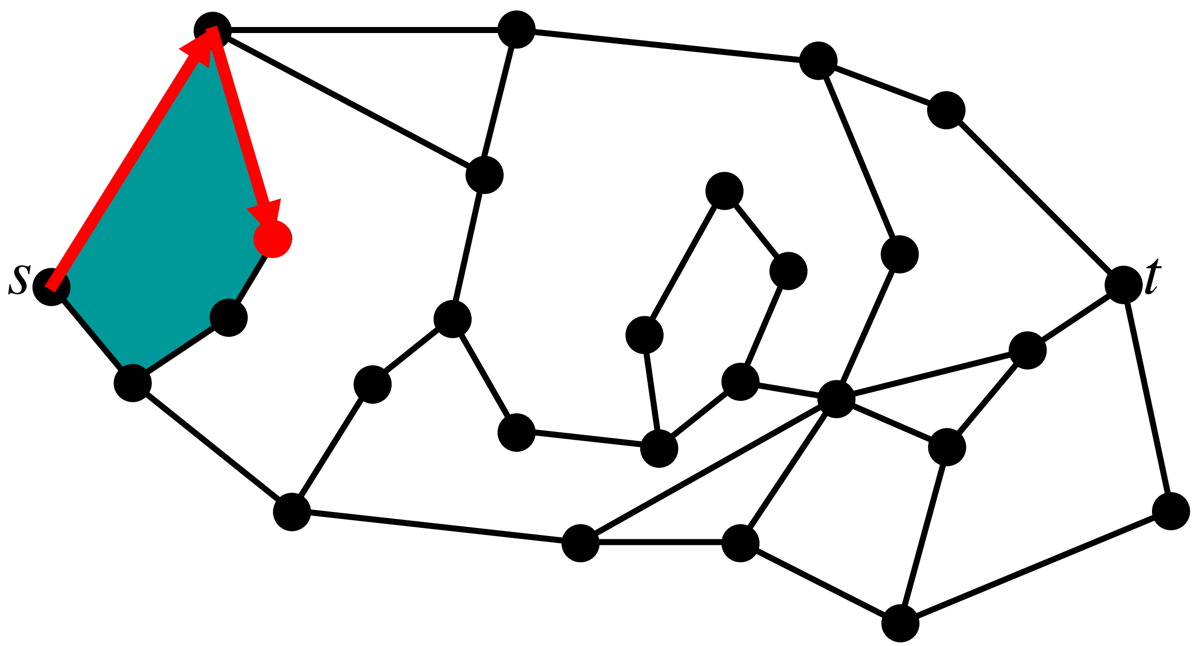
- Keep left hand on the wall, walk until hit the straight line connecting source to destination.
- Then switch to the next face.



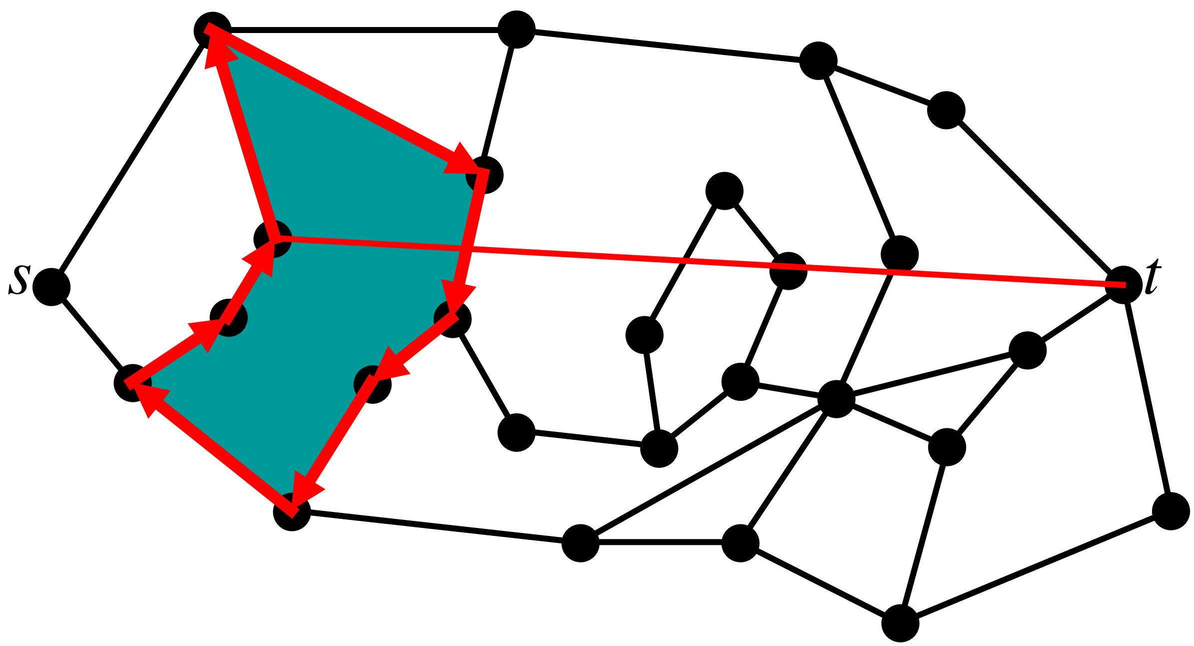
Face Routing



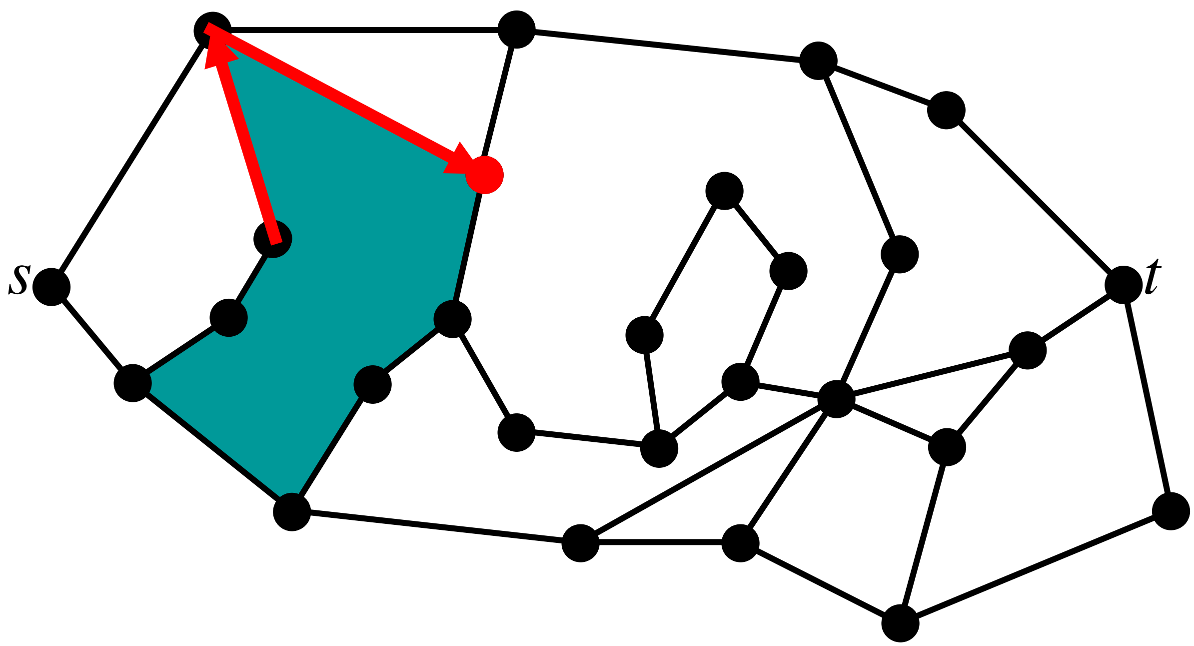
Face Routing



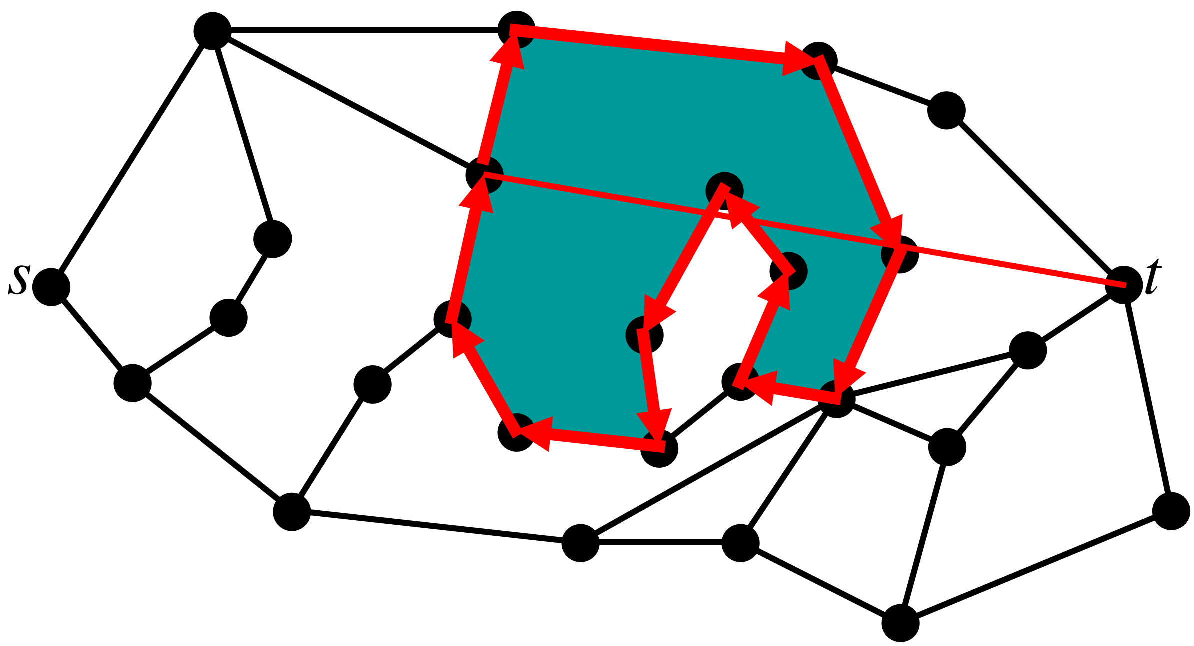
Face Routing



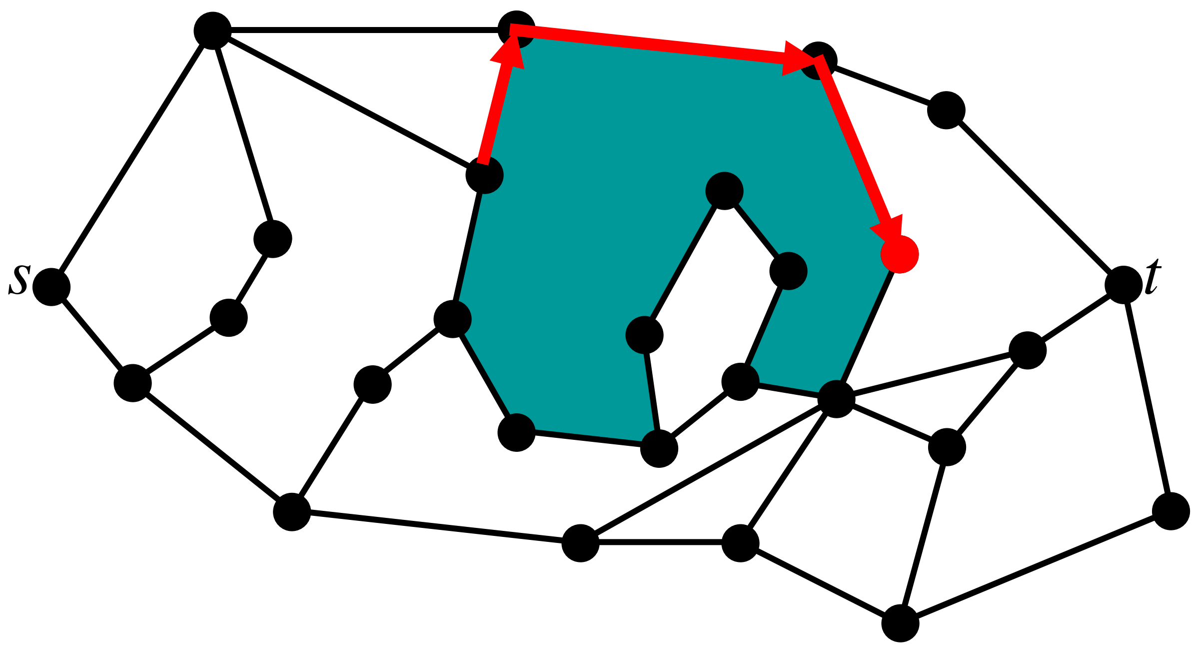
Face Routing



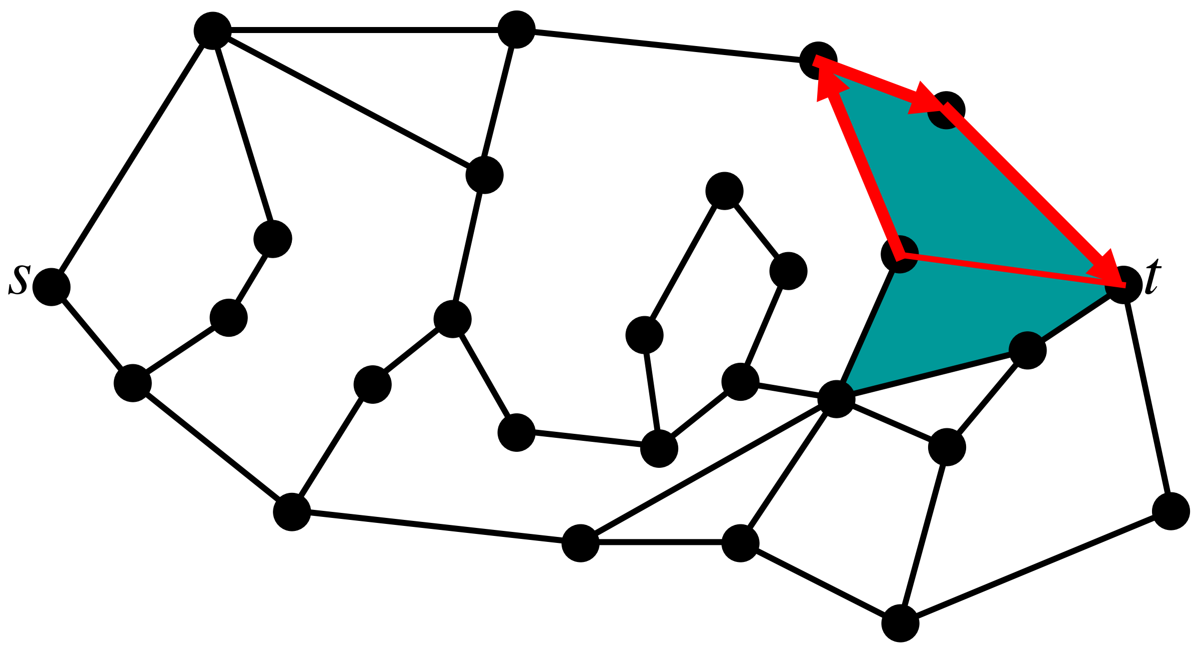
Face Routing



Face Routing

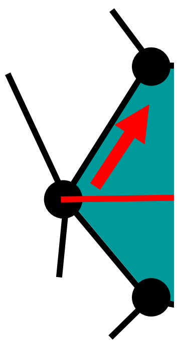


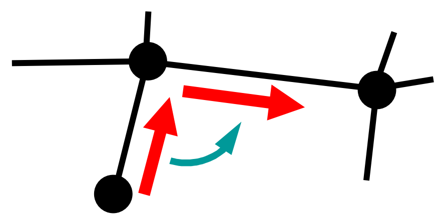
Face Routing



Face Routing Properties

- All necessary information is stored in the message
  - Source and destination positions
  - The node when it enters the perimeter mode.
  - The first edge on the current face.
- Completely local:
  - Knowledge about direct neighbors' positions is sufficient
  - Faces are implicit. Only local neighbor ordering around each node is needed

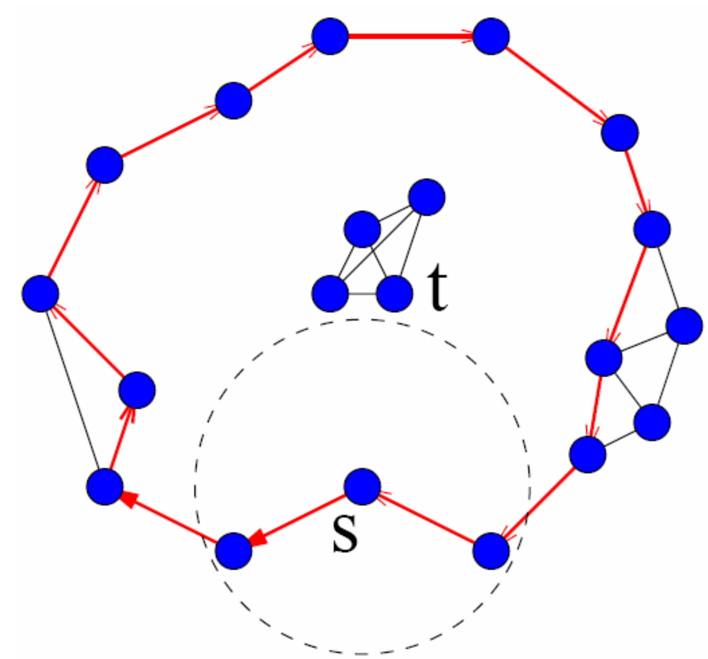




"Right Hand Rule"

What if the destination is disconnected?

- The perimeter routing will get back to where it enters the perimeter mode.
- Failed – no way to the destination.
- Guaranteed delivery of a message if there is a path.



Planar Graph Subtraction

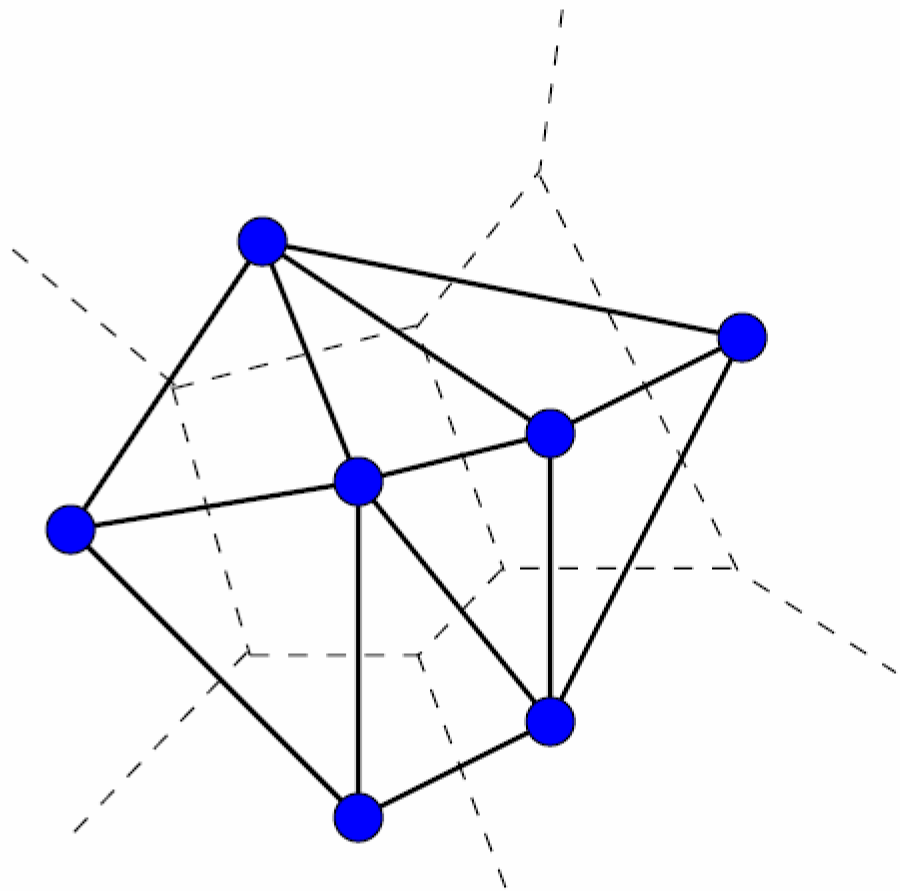
Compute a planar subgraph of the unit disk graph.

- Preserves connectivity.
- Distributed computation.

A little detour on Delaunay triangulation

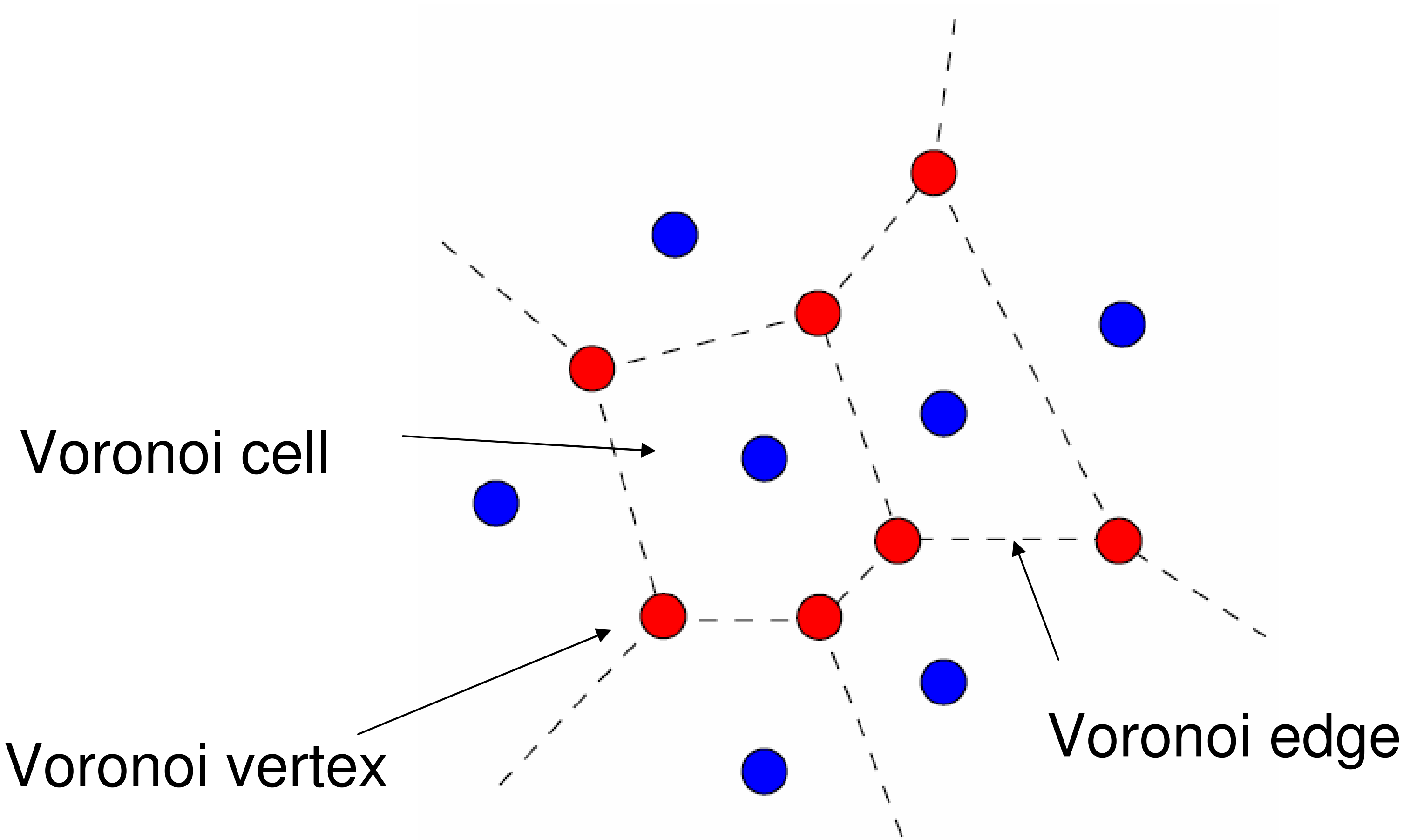
Delaunay triangulation

- First proposed by B. Delaunay in 1934.
- Numerous applications since then.



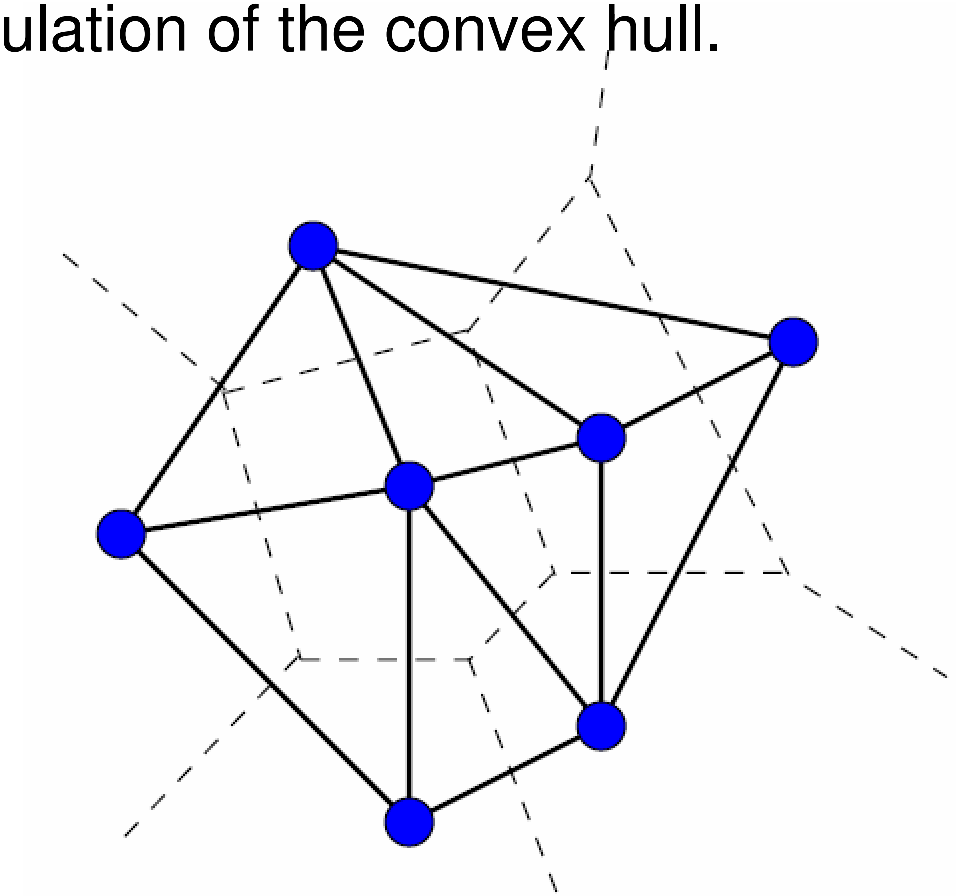
Voronoi diagram

- Partition the plane into cells such that all the points inside a cell have the same closest point.



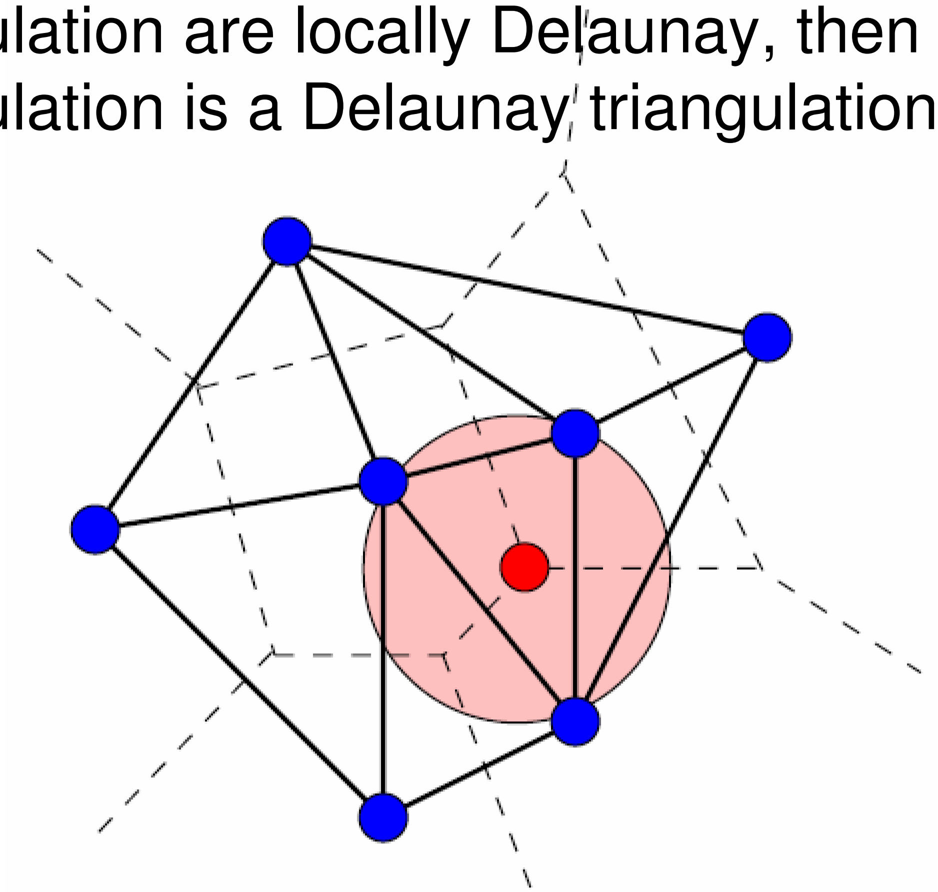
Delaunay triangulation

- Dual of Voronoi diagram: Connect an edge if their Voronoi cells are adjacent.
- Triangulation of the convex hull.



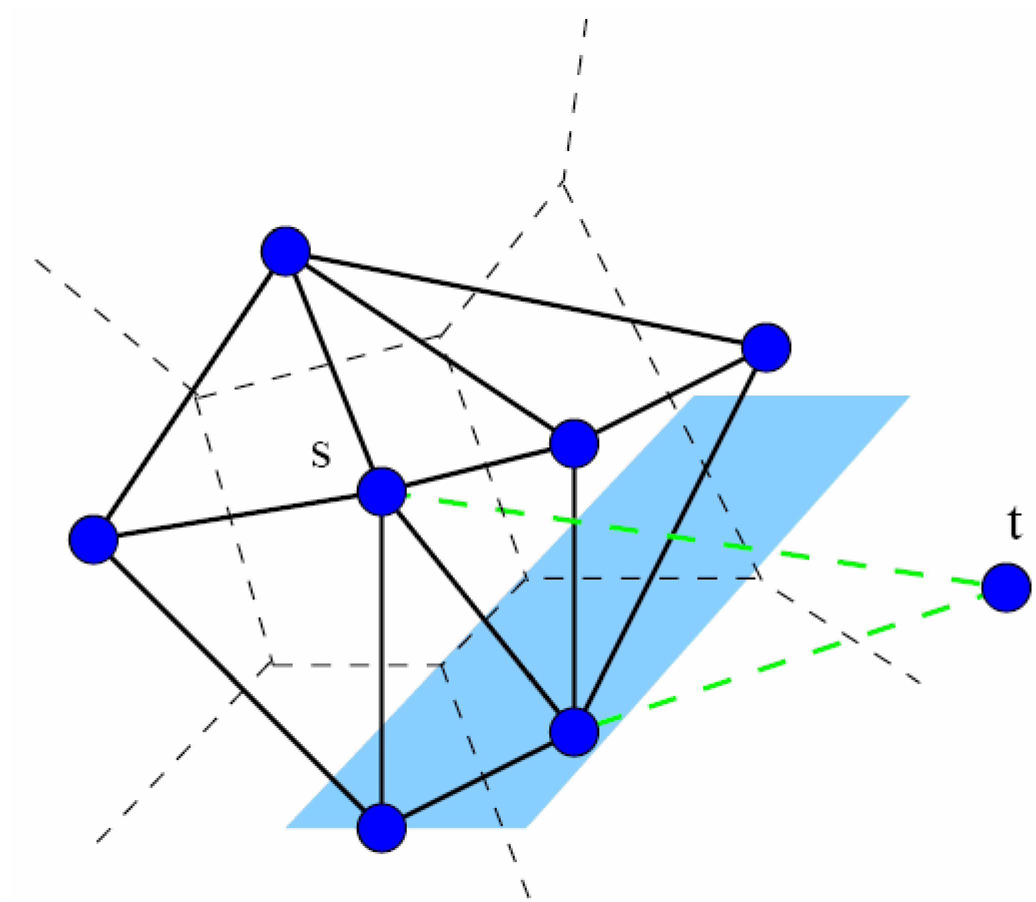
Delaunay triangulation

- "Empty-circle property": the circumcircle of a Delaunay triangle is empty of other points.
- The converse is also true: if all the triangles in a triangulation are locally Delaunay, then the triangulation is a Delaunay triangulation.



Greedy routing on Delaunay triangulation

- Claim: Greedy routing on DT never gets stuck.



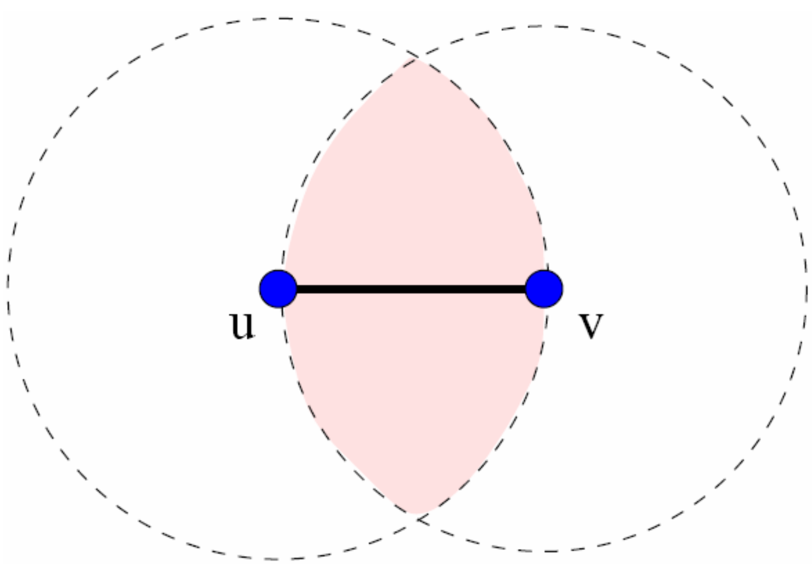
Delaunay triangulation

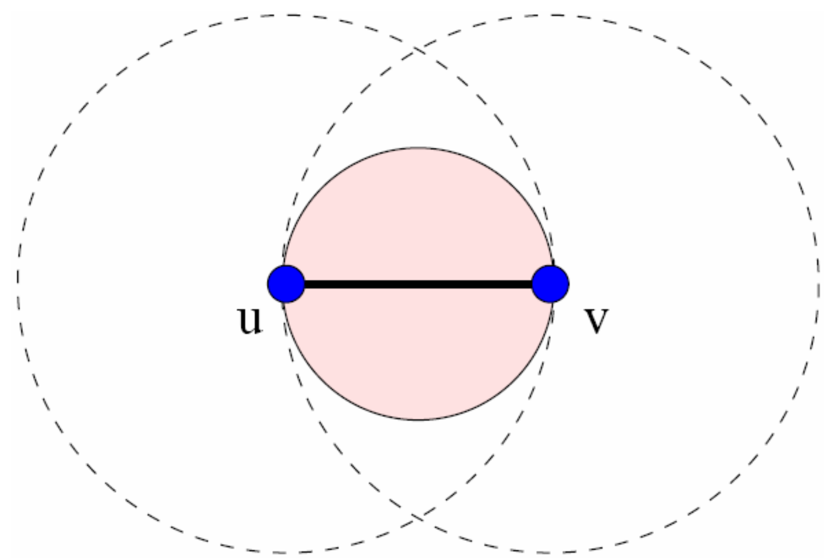
- For an arbitrary point set, the Delaunay triangulation may contain long edges.
- Centralized construction.

- Next: 2 planar subgraphs that can be constructed in a distributed way: relative neighborhood graph and the Gabriel graph.

Relative Neighborhood Graph and Gabriel Graph

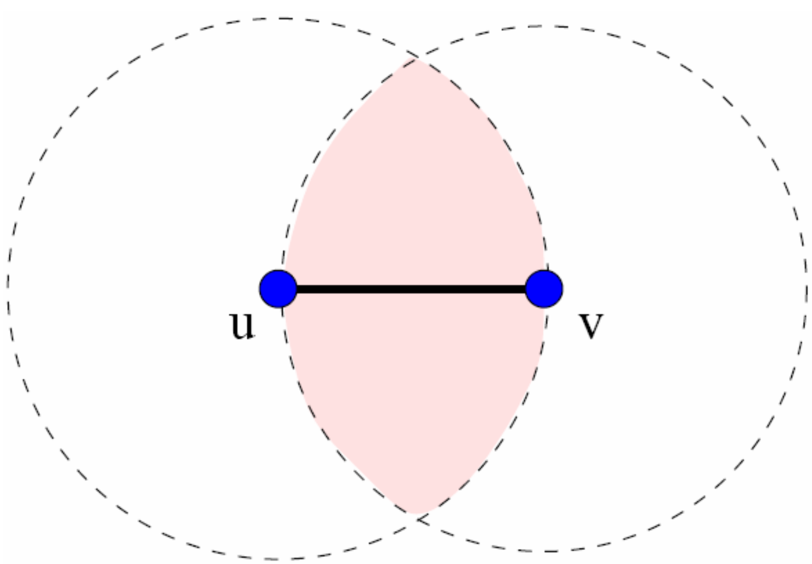
- Relative Neighborhood Graph (RNG) contains an edge uv if the lune is empty of other points.
- Gabriel Graph (GG) contains an edge uv if the disk with uv as diameter is empty of other points.
- Both can be constructed in a distributed way.

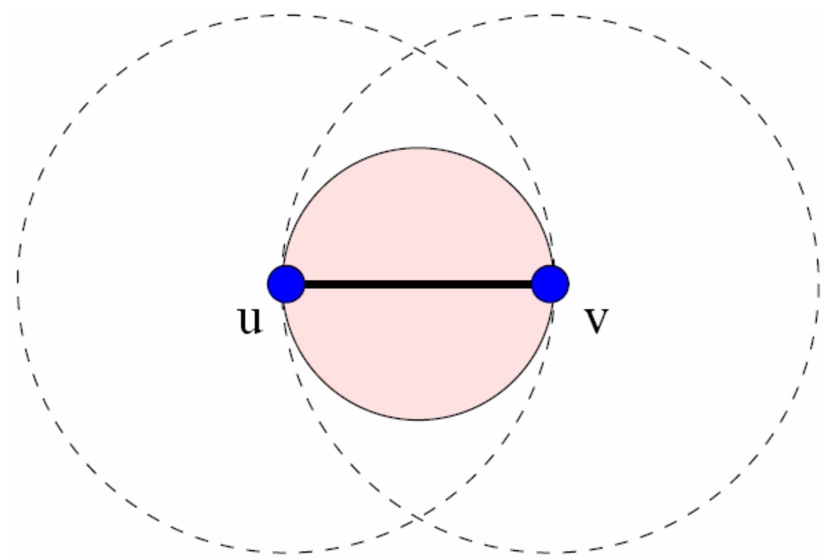




Relative Neighborhood Graph and Gabriel Graph

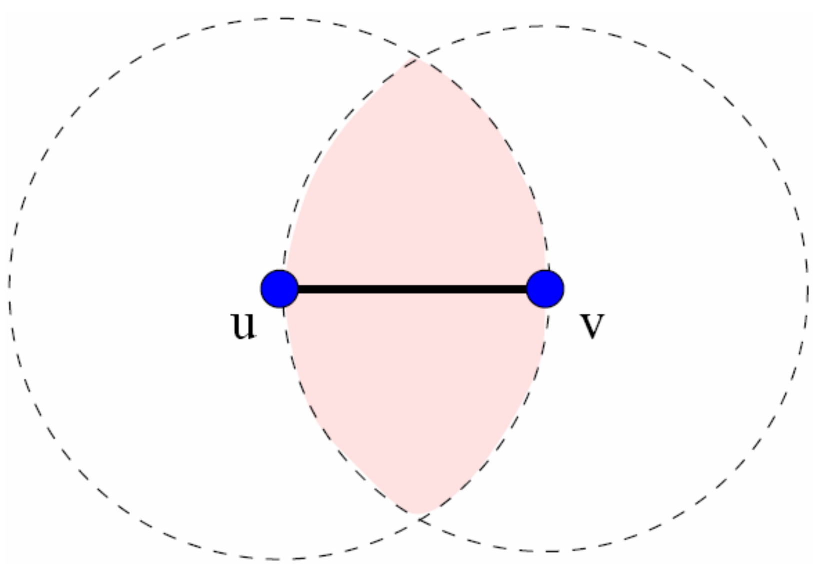
- Claim: MST $\subseteq$ RNG $\subseteq$ GG $\subseteq$ Delaunay
- Thus, RNG and GG are planar (Delaunay is planar) and keep the connectivity (MST has the same connectivity of UDG).

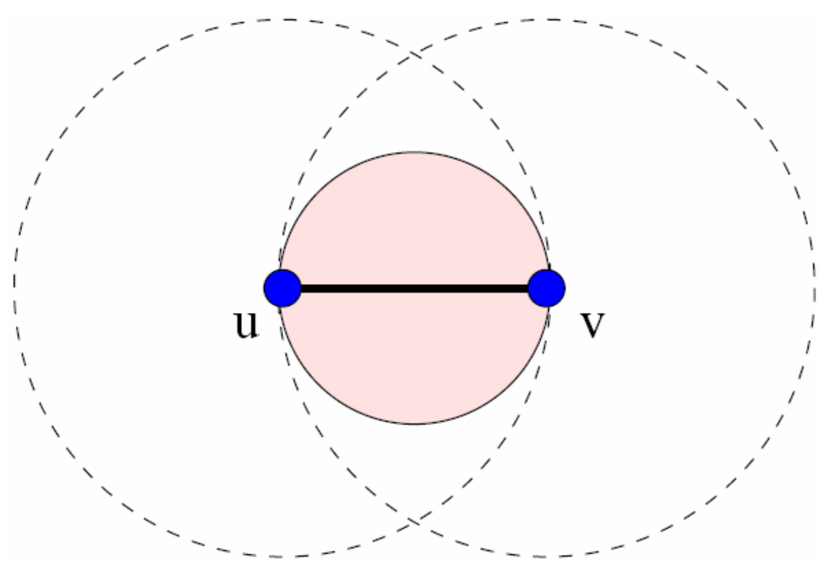




MST $\subseteq$ RNG $\subseteq$ GG $\subseteq$ Delaunay

1. RNG $\subseteq$ GG: if the lune is empty, then the disk with uv as diameter is also empty.
2. GG $\subseteq$ Delaunay: the disk with uv as diameter is empty, then uv is a Delaunay edge.

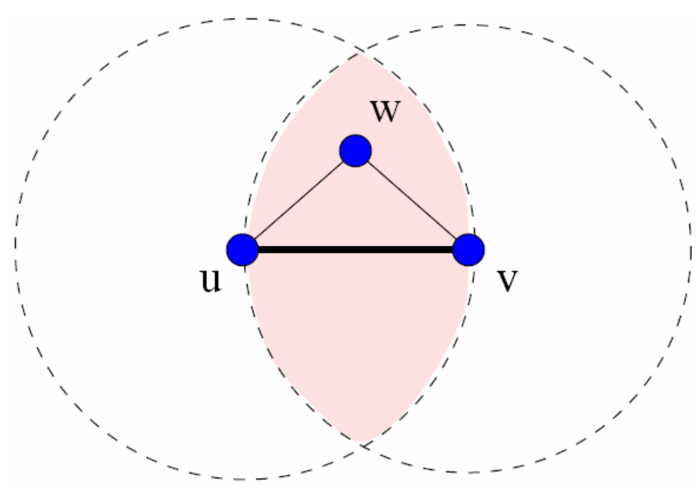




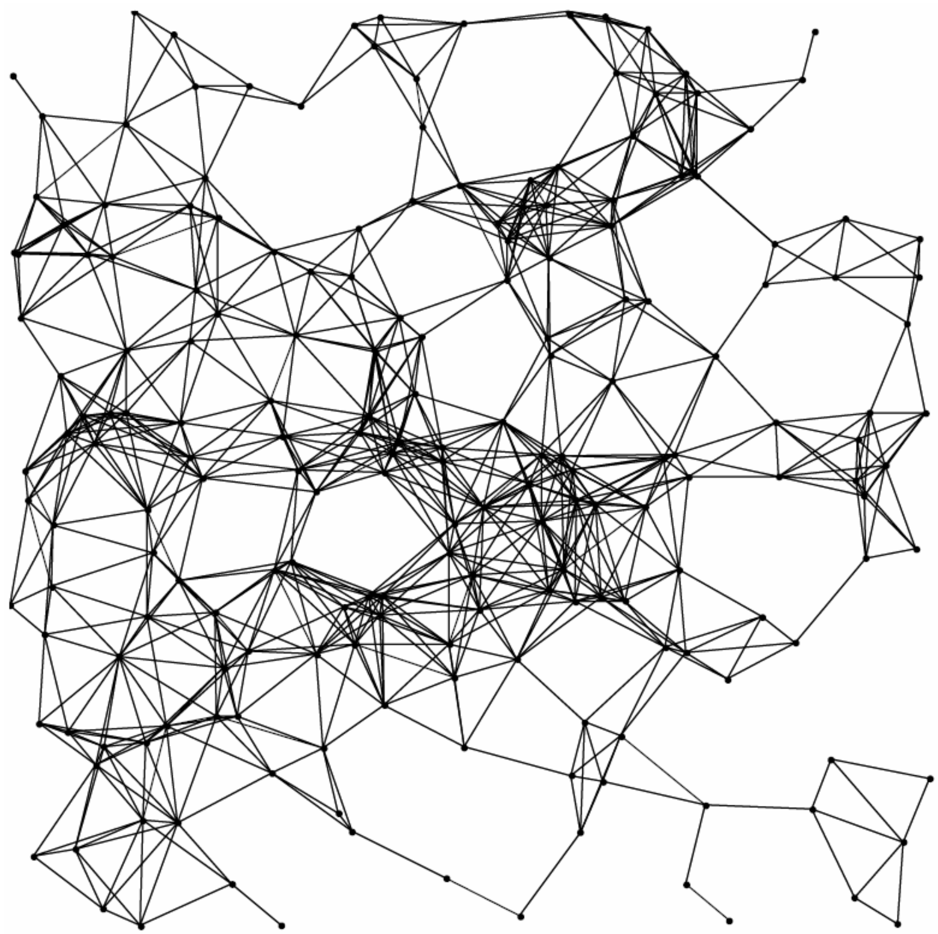
MST $\subseteq$ RNG $\subseteq$ GG $\subseteq$ Delaunay

3. MST $\subseteq$ RNG:
   - Assume uv in MST is not in RNG, there is a point w inside the lune. $|uv|>|uw|$, $|uv|>|vw|$.
   - Now we delete uv and partition the MST into two subtrees.
   - Say w is in the same component with u, then we can replace uv by wv and get a lighter tree. ➔ contradiction.

RNG and GG are planar (Delaunay is planar) and keep the connectivity (MST has the same connectivity of UDG).

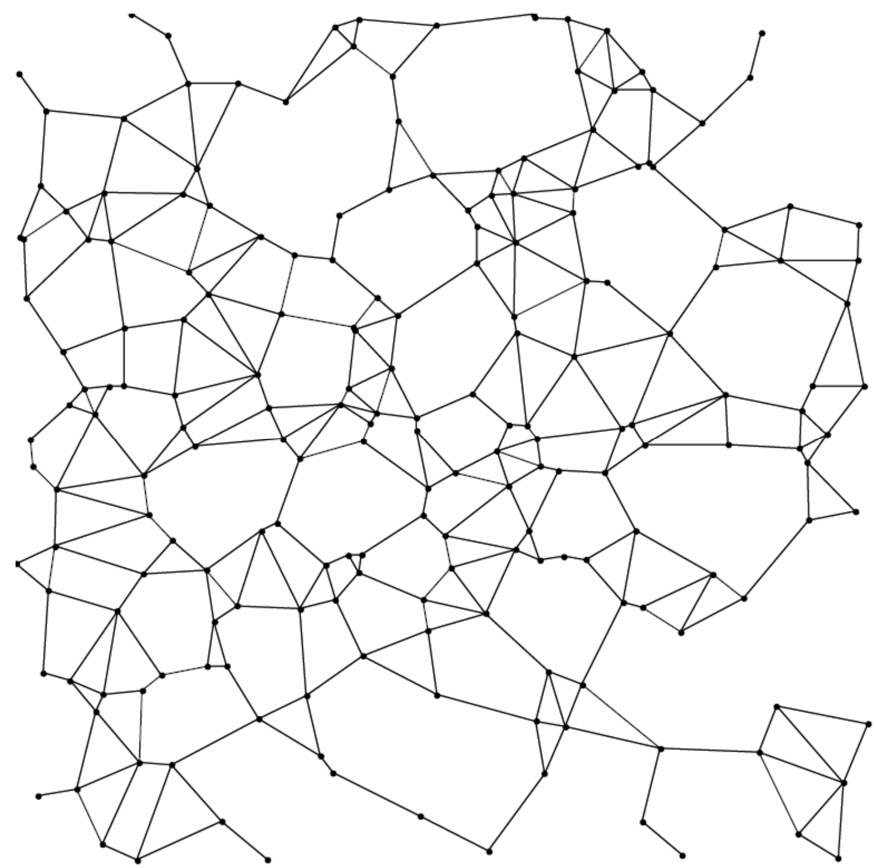


An example of UDG

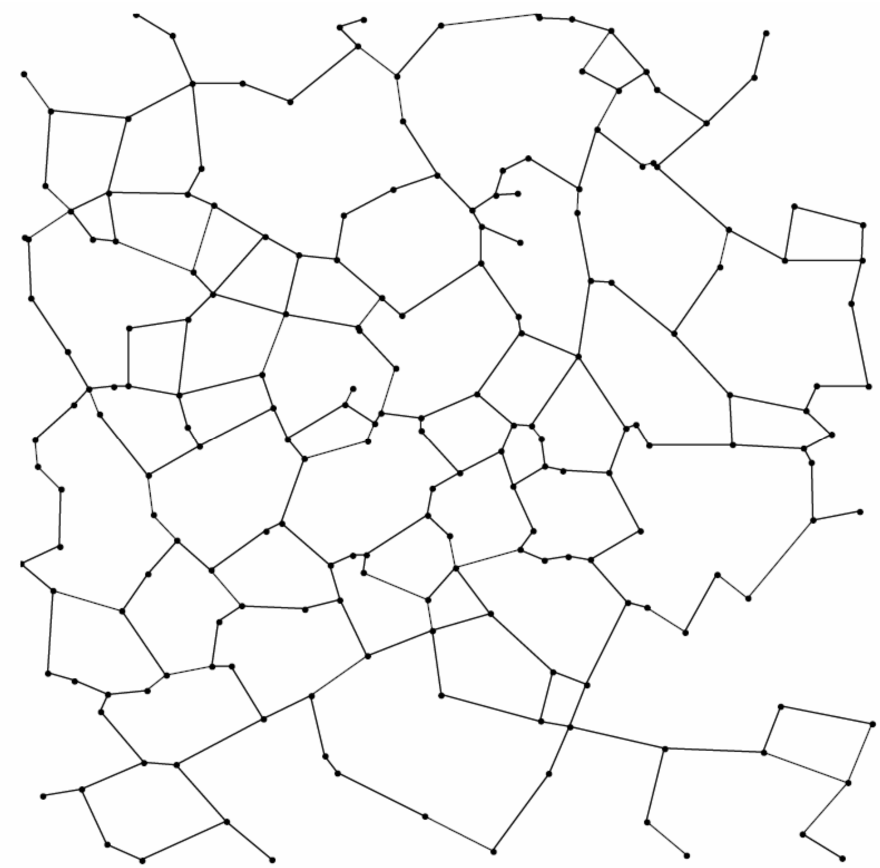


200 nodes randomly deployed in a 2000×2000 meters region.
Radio range =250meters

An example of GG and RNG



GG



RNG

Two problems remain

- A subgraph G' of G is a α-spanner if the shortest path in G' is bounded by a constant α times the shortest path length in G.

- Both RNG and GG are not spanners $\rightarrow$ a short path may not exist!

- Even if the planar subgraph contains a short path, can greedy routing and face routing find a short one?

Other problems

- Localization:
  - Nodes need to know their geographical locations.

- Location service:
  - How does a source know the location of destination?
  - What if the nodes move around?

- Planar graph construction:
  - Requires a unit disk graph assumption, which is not always the case in practice.
  - What if the nodes are in 3d?

Summary

- Location-based routing
- Greedy forwarding
- Planar graph routing
- How to construct a planar graph?