Path Vector Face Routing: Geographic Routing with Local Face Information

Ben Leong, Sayan Mitra, and Barbara Liskov
MIT CSAIL

Geographic Routing

- Geographic routing algorithms
  - leverage physical location information
  - scale better than other ad hoc routing algorithms (Karp, 2001)
  - state proportional to network density, not size
  - can be applied using virtual coordinates (Rao et al., 2003)

Geographic Routing

- Existing geographic routing algorithms
  - GPSR (Karp, 2001)
    GFG (Bose, 2001)
  - GOAFR+ (Kuhn, 2003)
  - nodes know only about immediate neighbors
- Can we do better if nodes have more information?

Geographic Routing

- Existing geographic routing algorithms
  - GPSR (Karp, 2001)
    GFG (Bose, 2001)
  - GOAFR+ (Kuhn, 2003)
  - nodes know only about immediate neighbors
- Can we do better if nodes have more information? **Yes!**

Greedy Path Vector Face Routing

- Our new algorithm (GPVFR):
  - stores small amount of additional local information (< 200 bytes)
  - improve maximum routing stretch over GPSR by 35 to 40%
  - improve maximum routing stretch over GOAFR+ by 20 to 25%

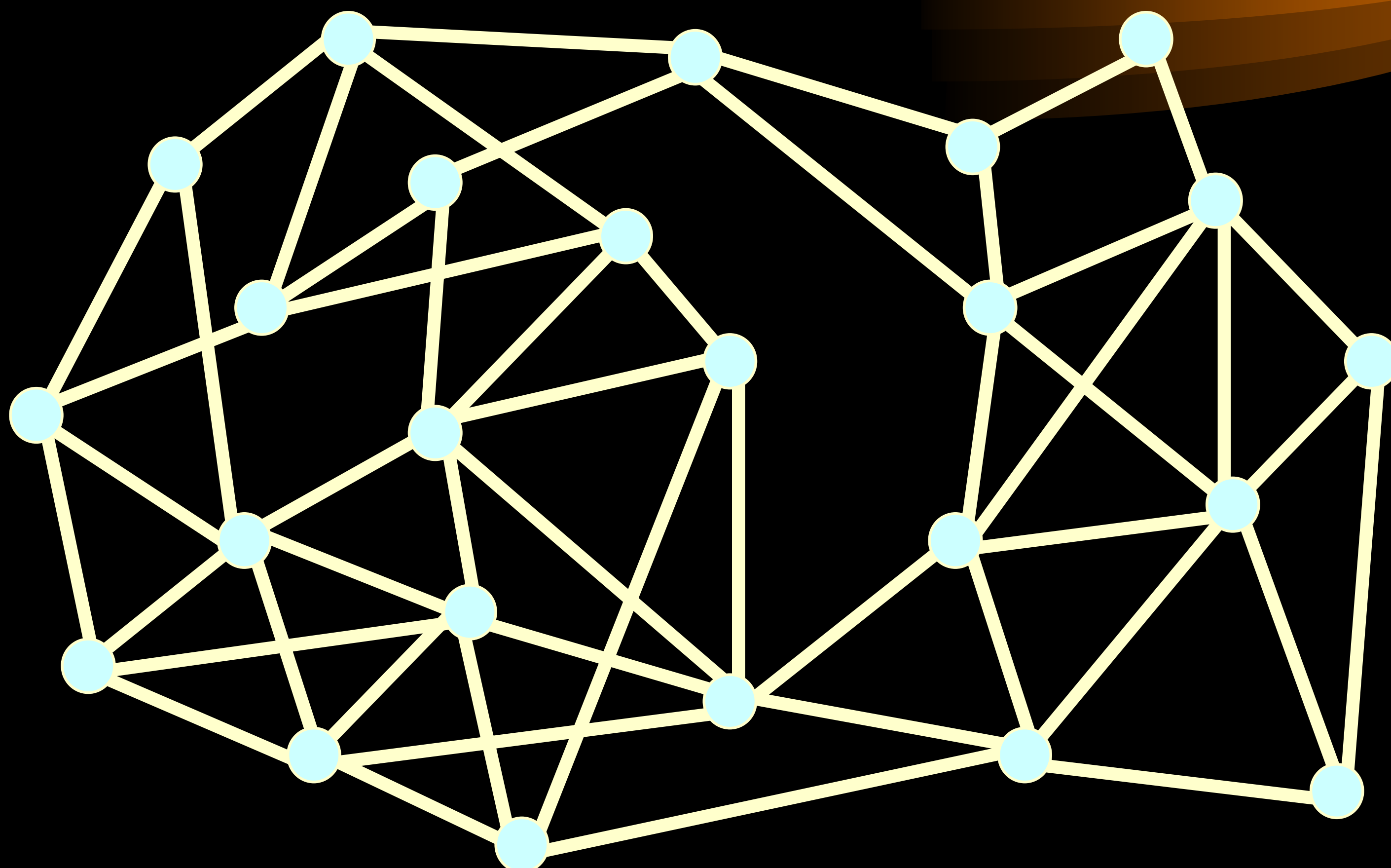
Overview

- Problem
- Approach
- Simulation Results
- Conclusion

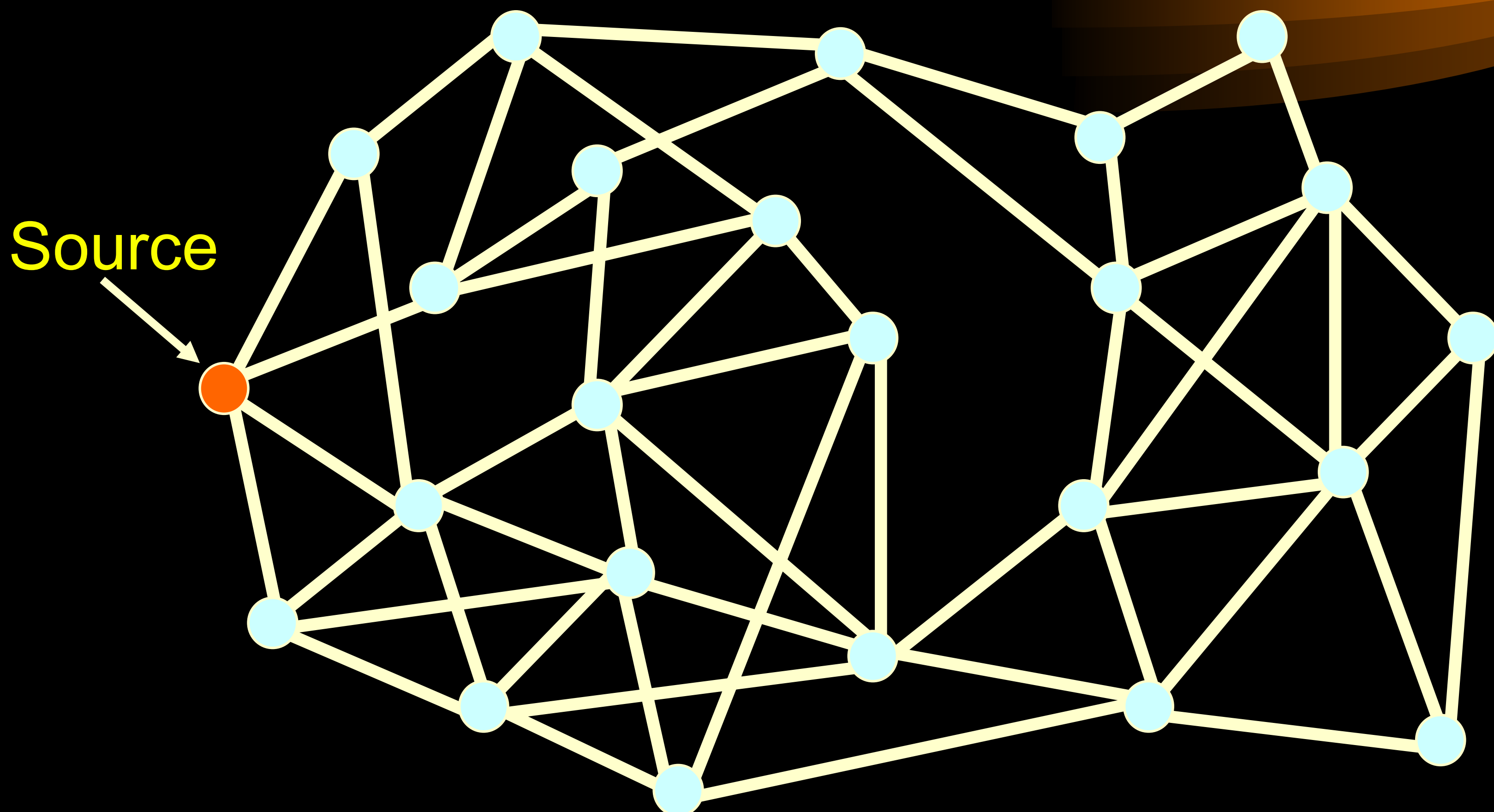
Geographic Routing

- Nodes have *x*-*y* coordinates
- Nodes know coordinates of immediate neighbors
- Packet destinations specified with *x*-*y* coordinates
- In general, forward packets greedily

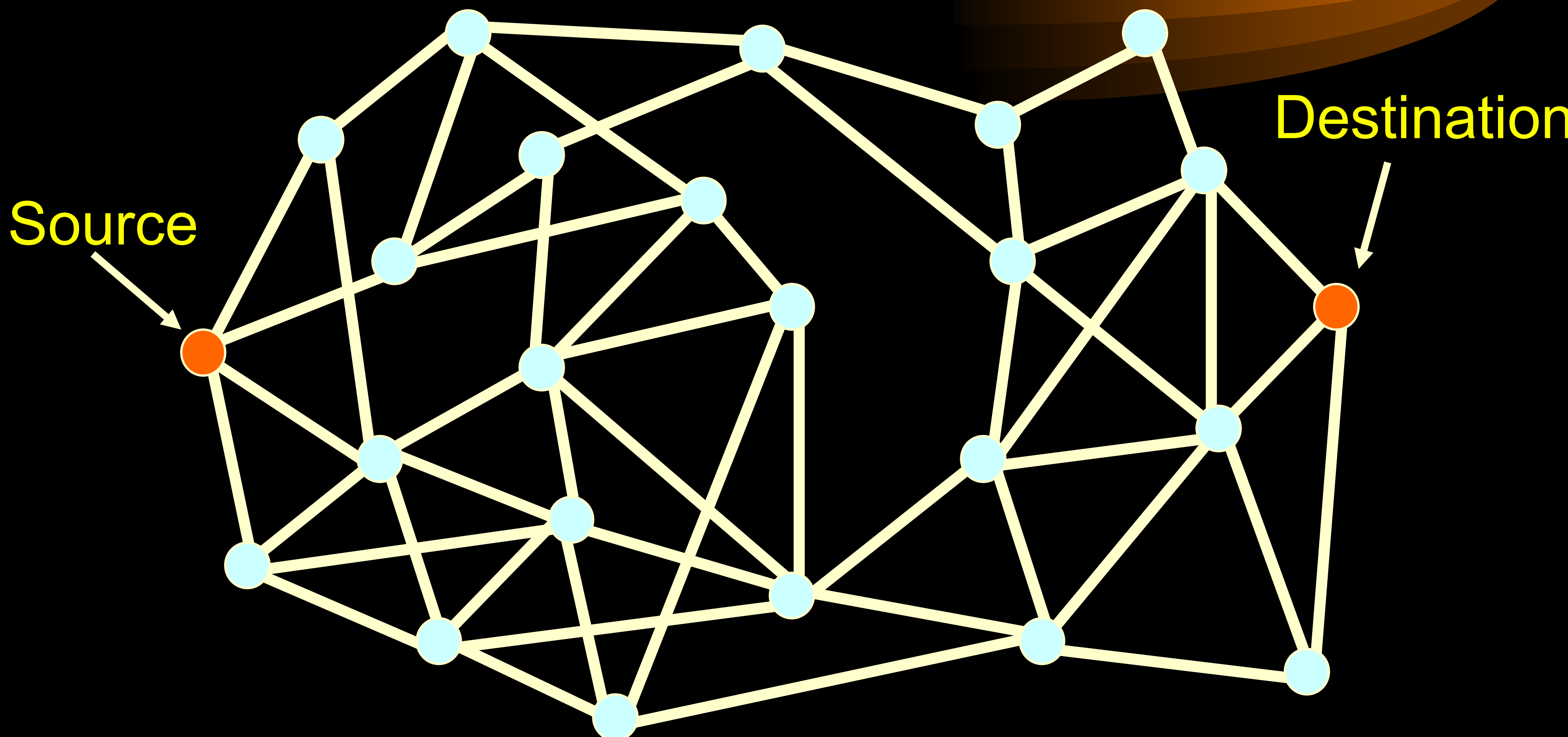
Example



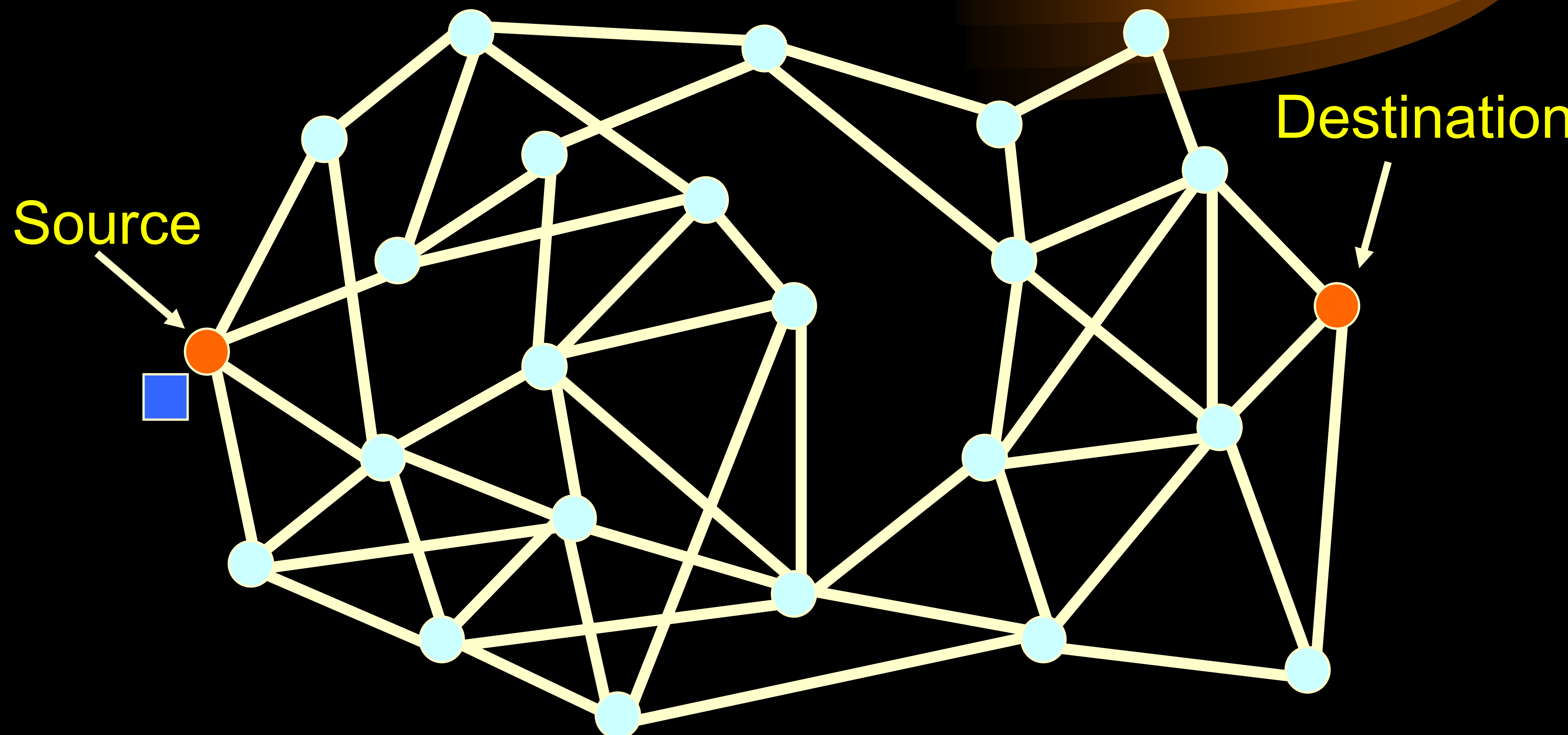
Example



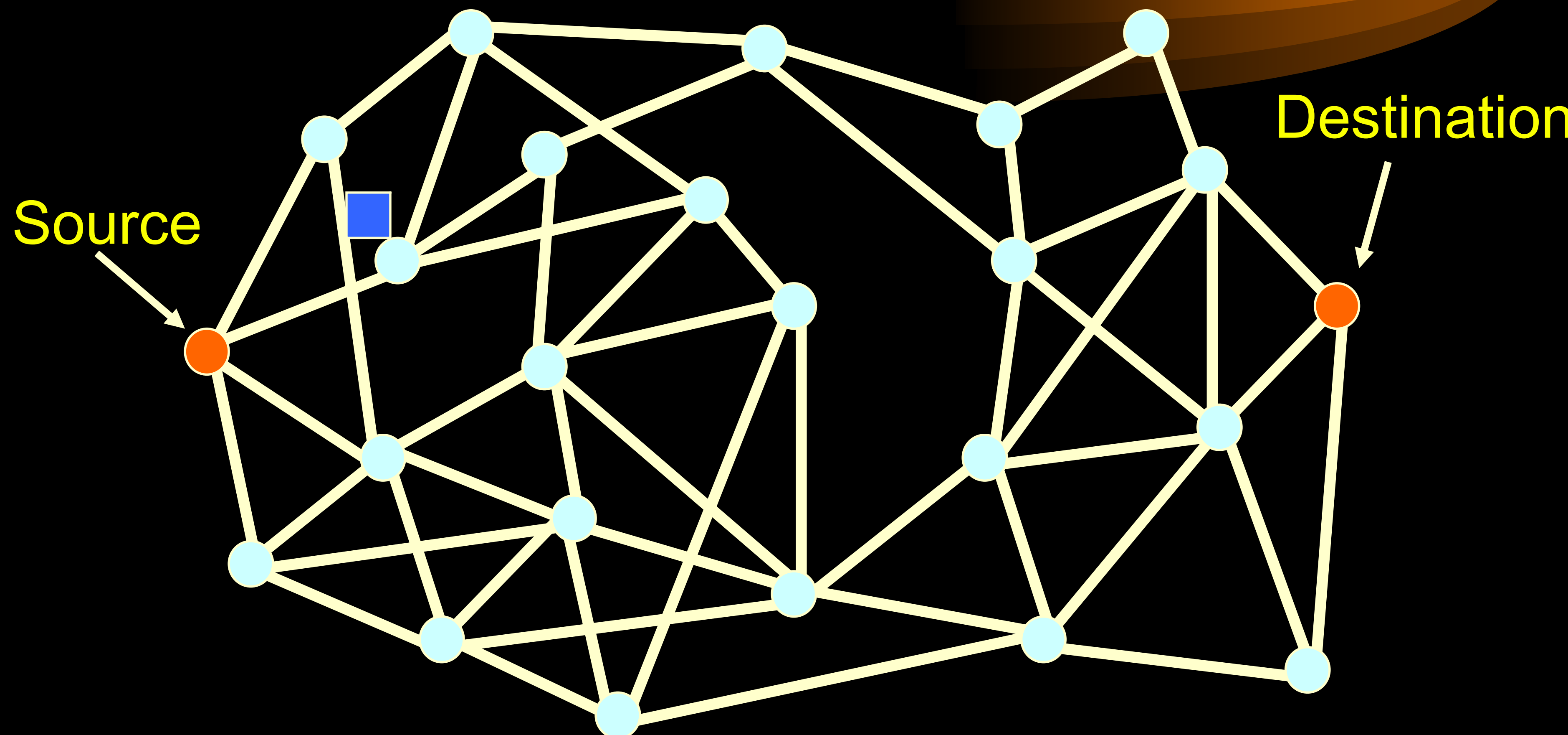
Example



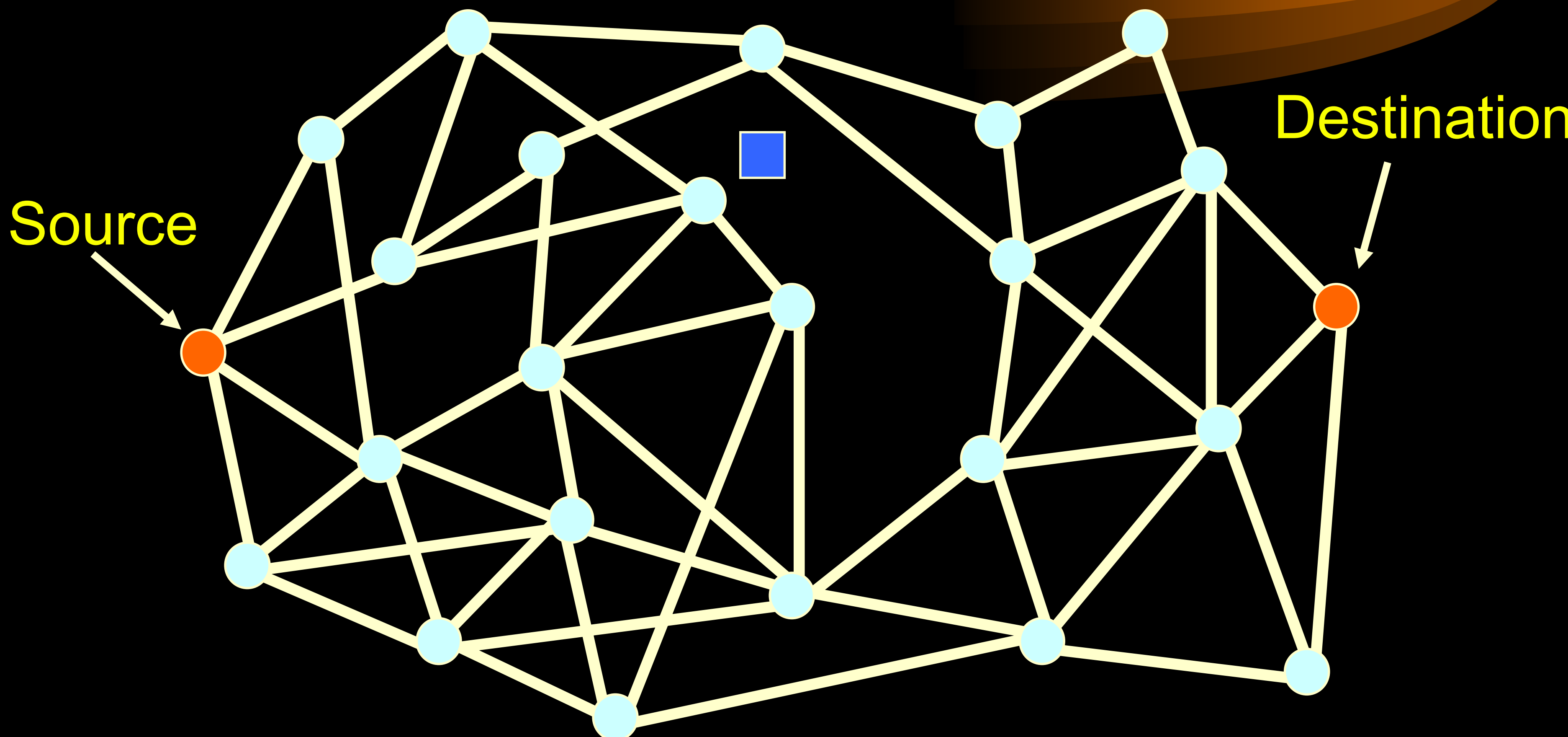
Example



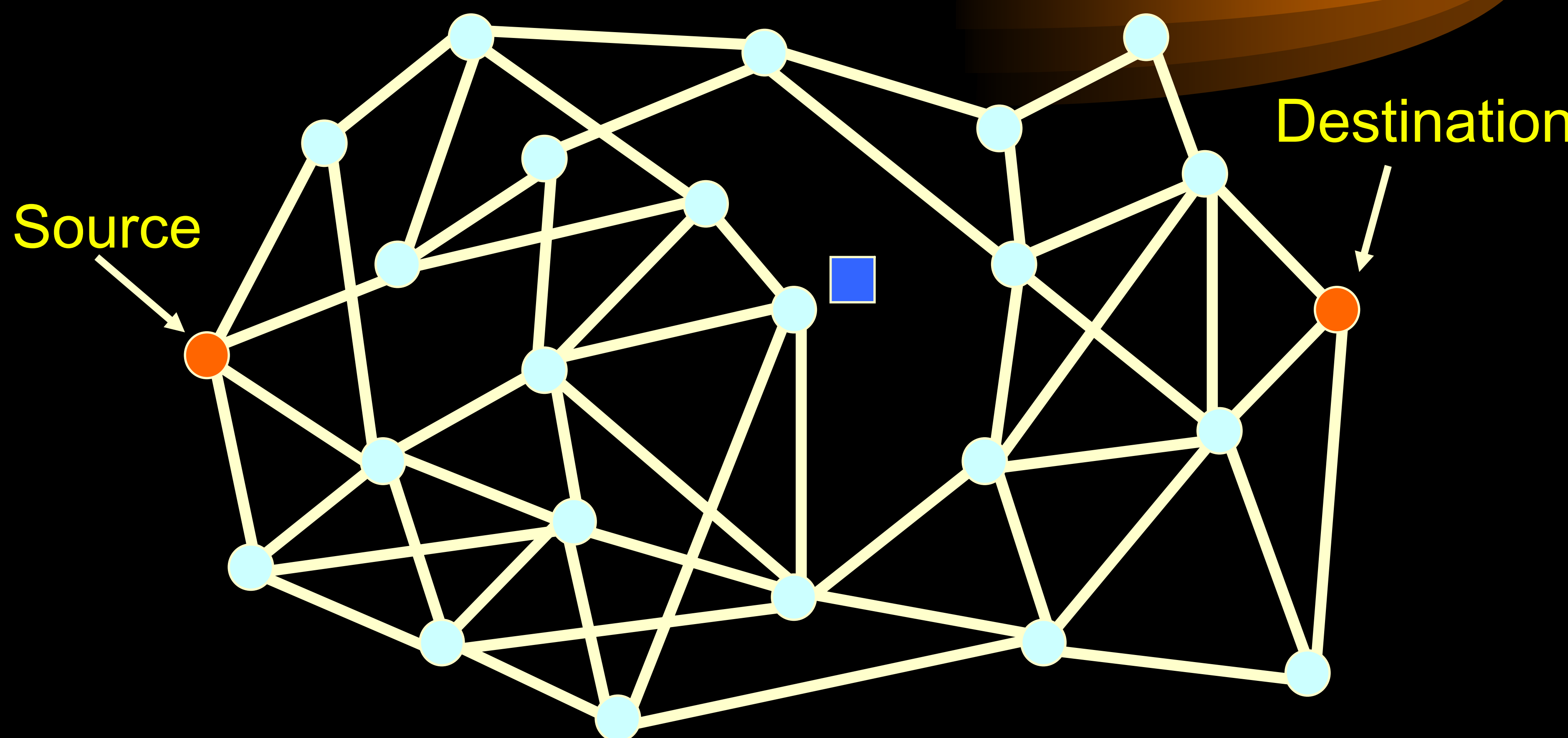
Example



Example



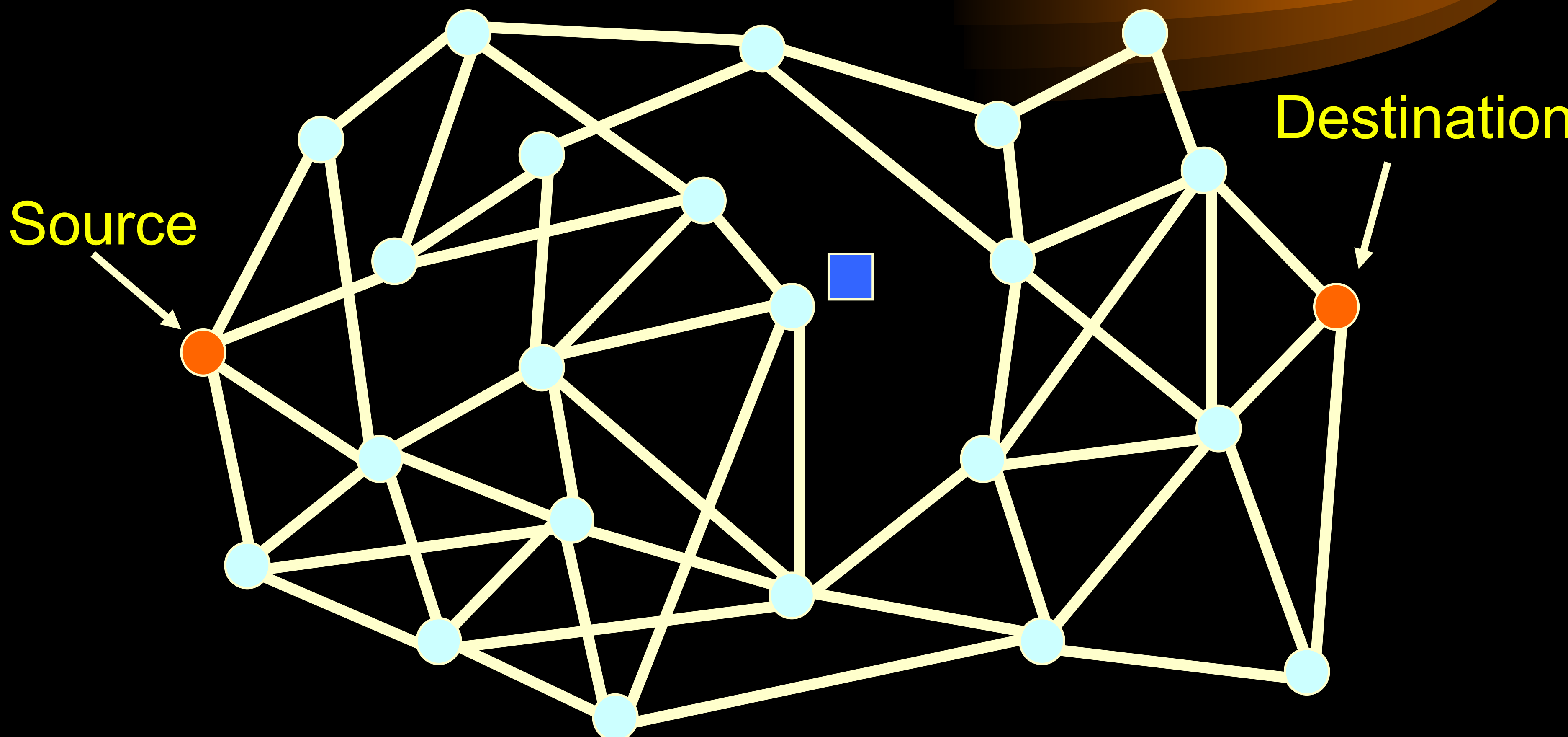
Example



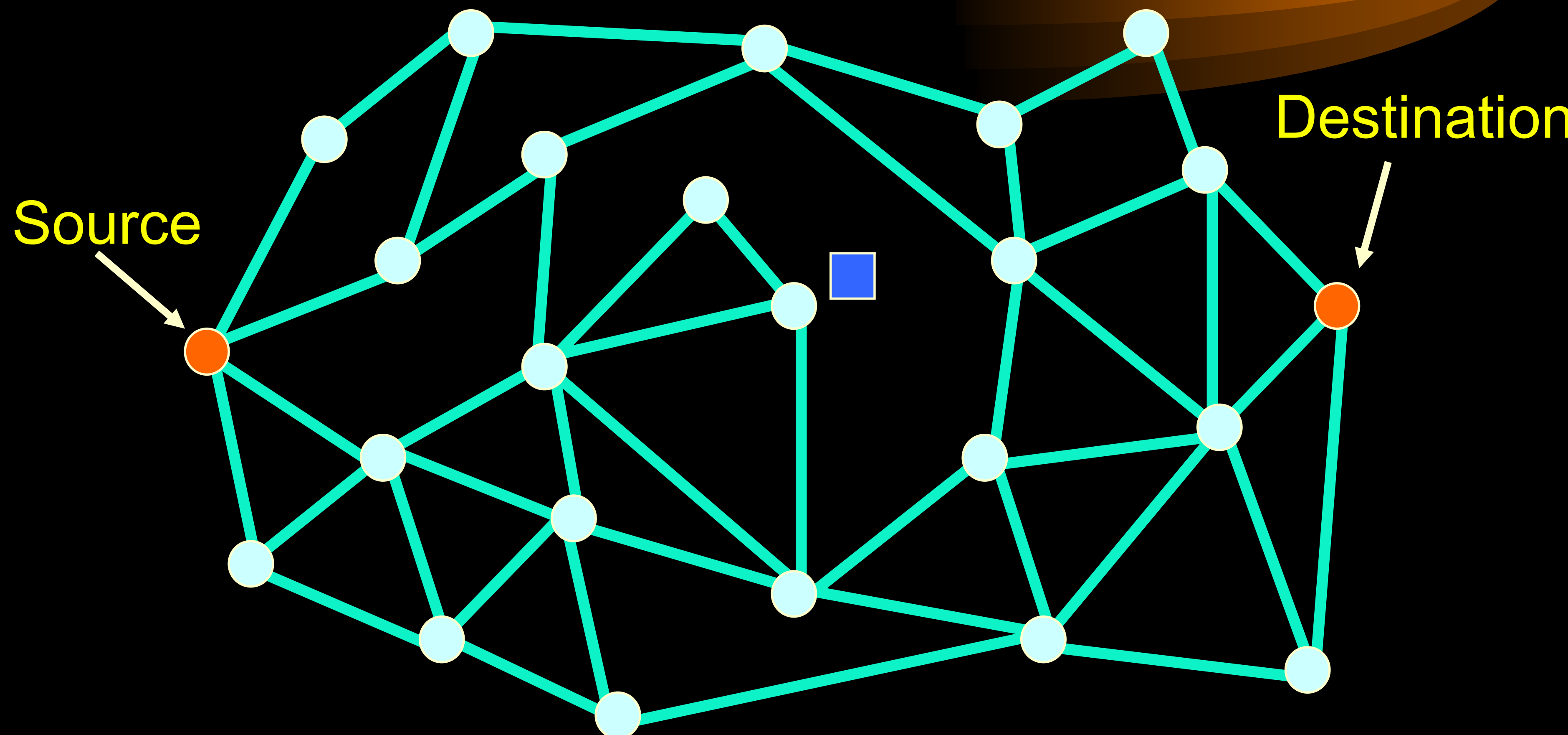
Geographic Face Routing

- Problem: sometimes a packet ends up at a local minimum.
- Face routing – route packet along faces of a planar subgraph
- Planarization:
  - Relative Neighborhood Graph (RNG)
  - Gabriel Graph (GG)
  - Cross Link Detection Protocol (CLDP) (Kim et al., NSDI 2005)

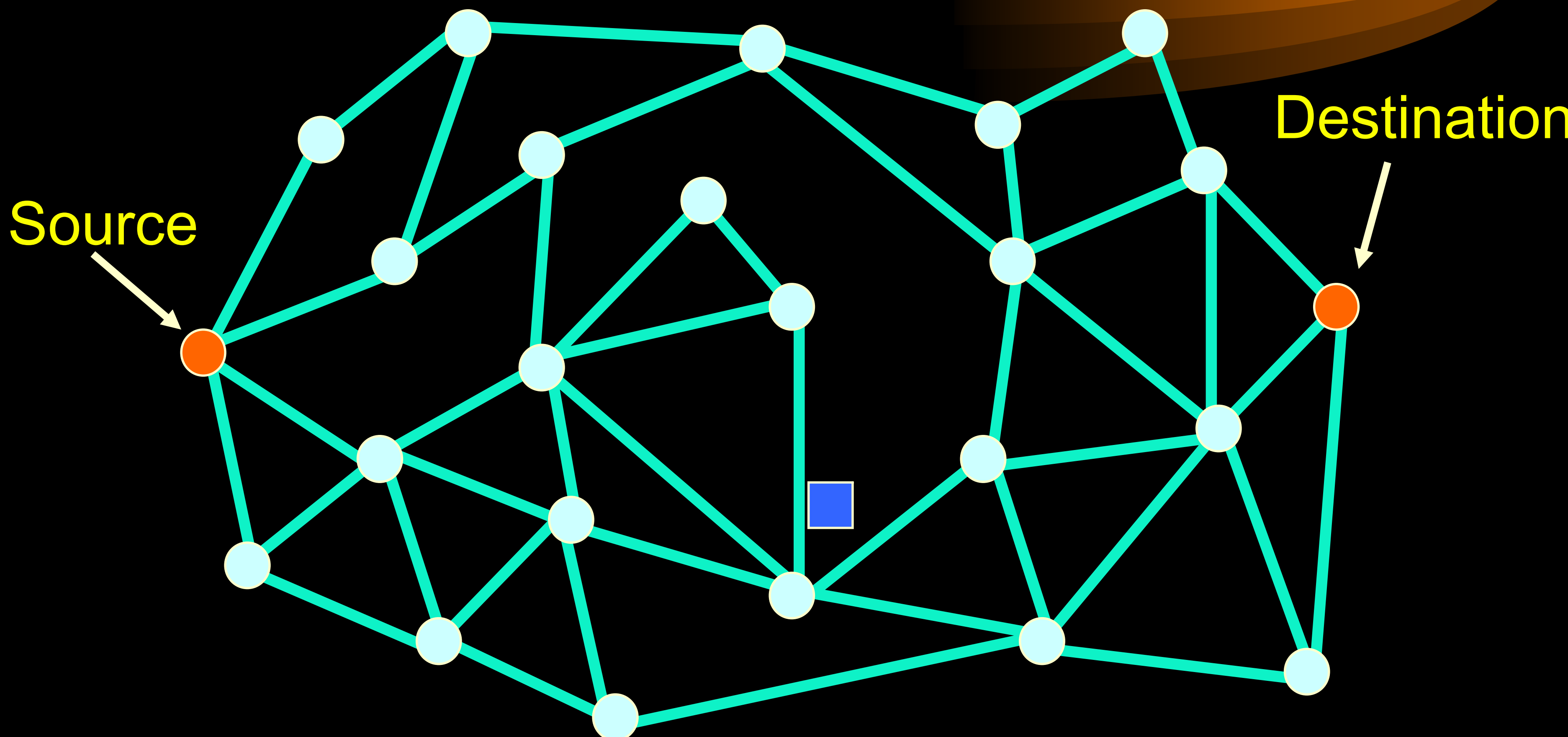
Example



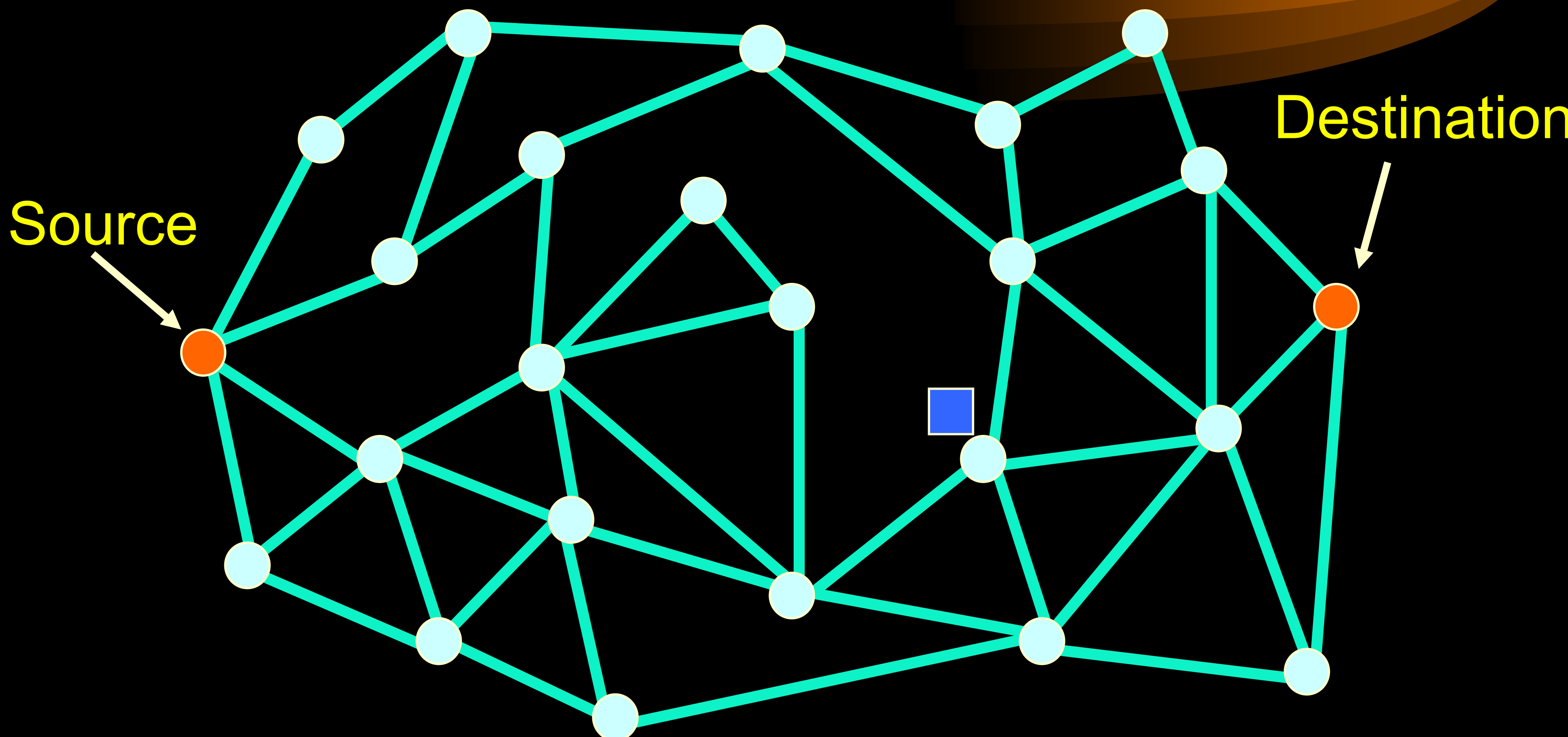
Example



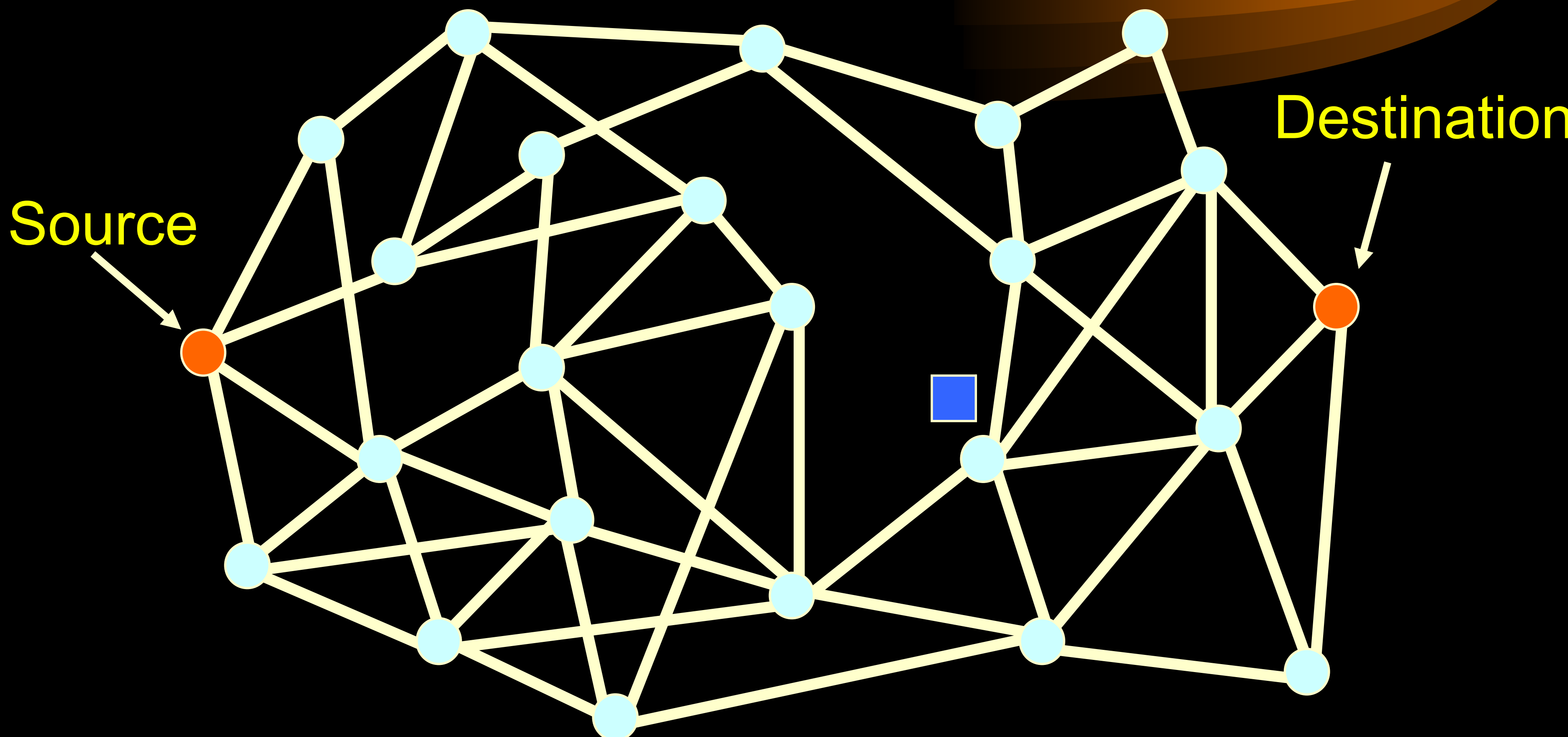
Example



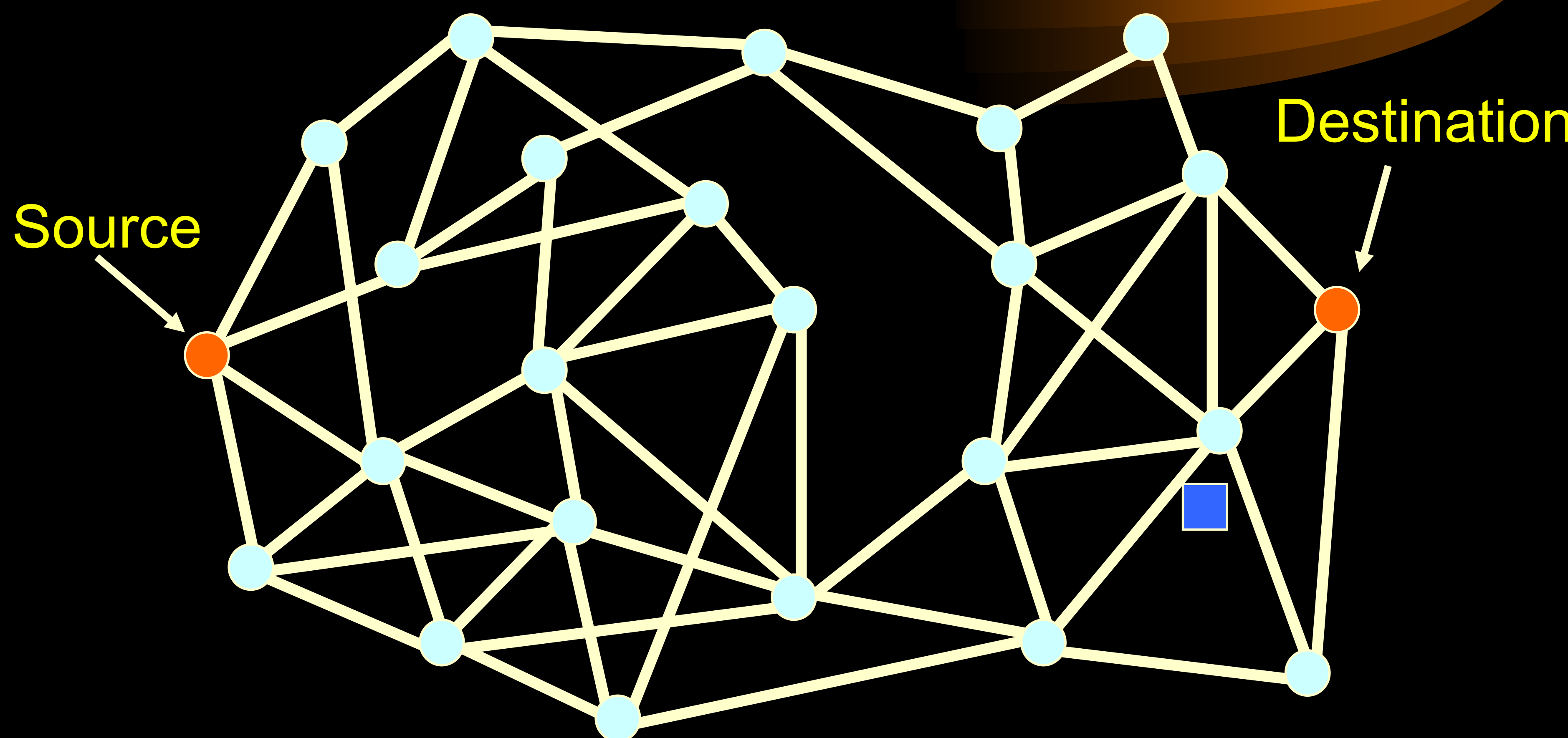
Example



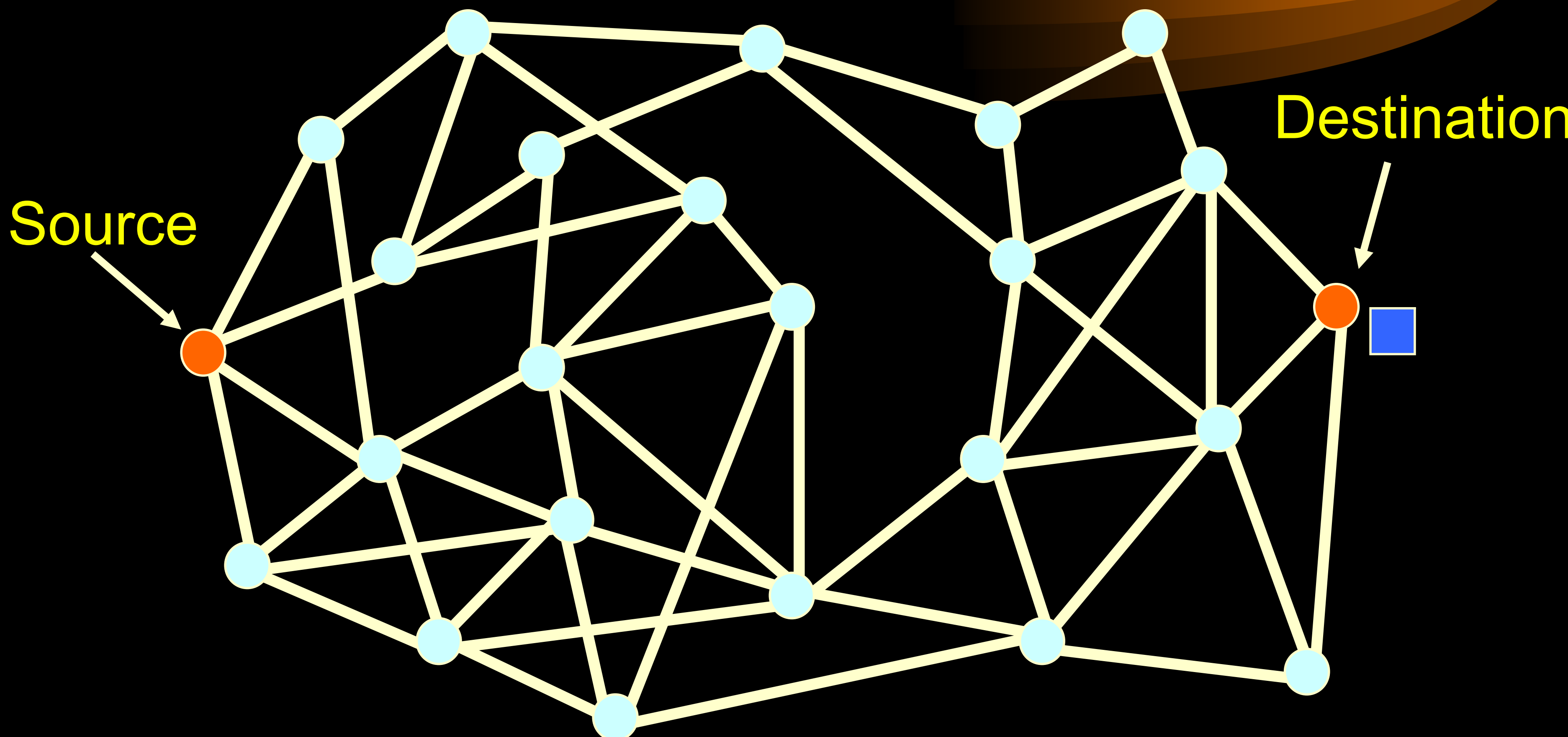
Example



Example



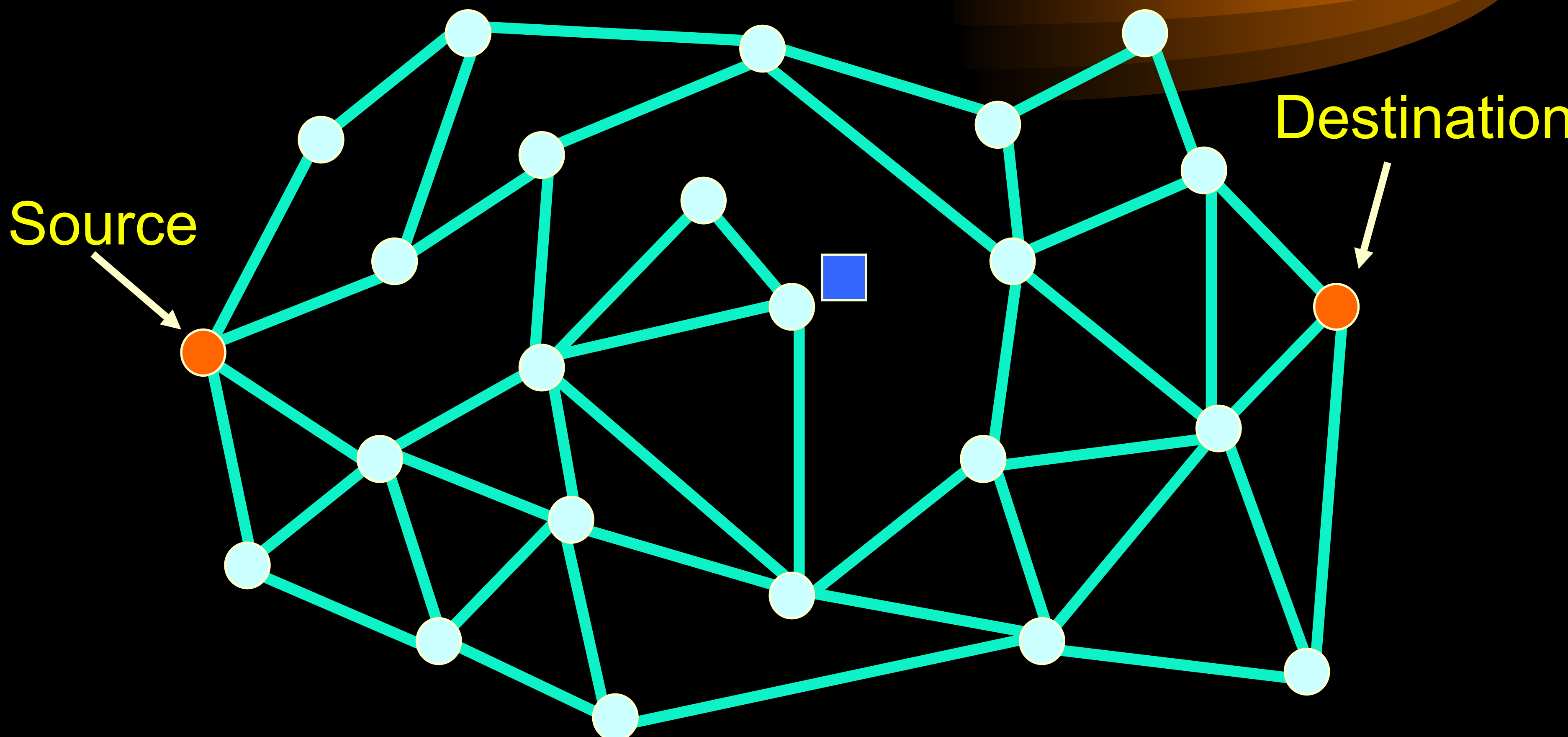
Example



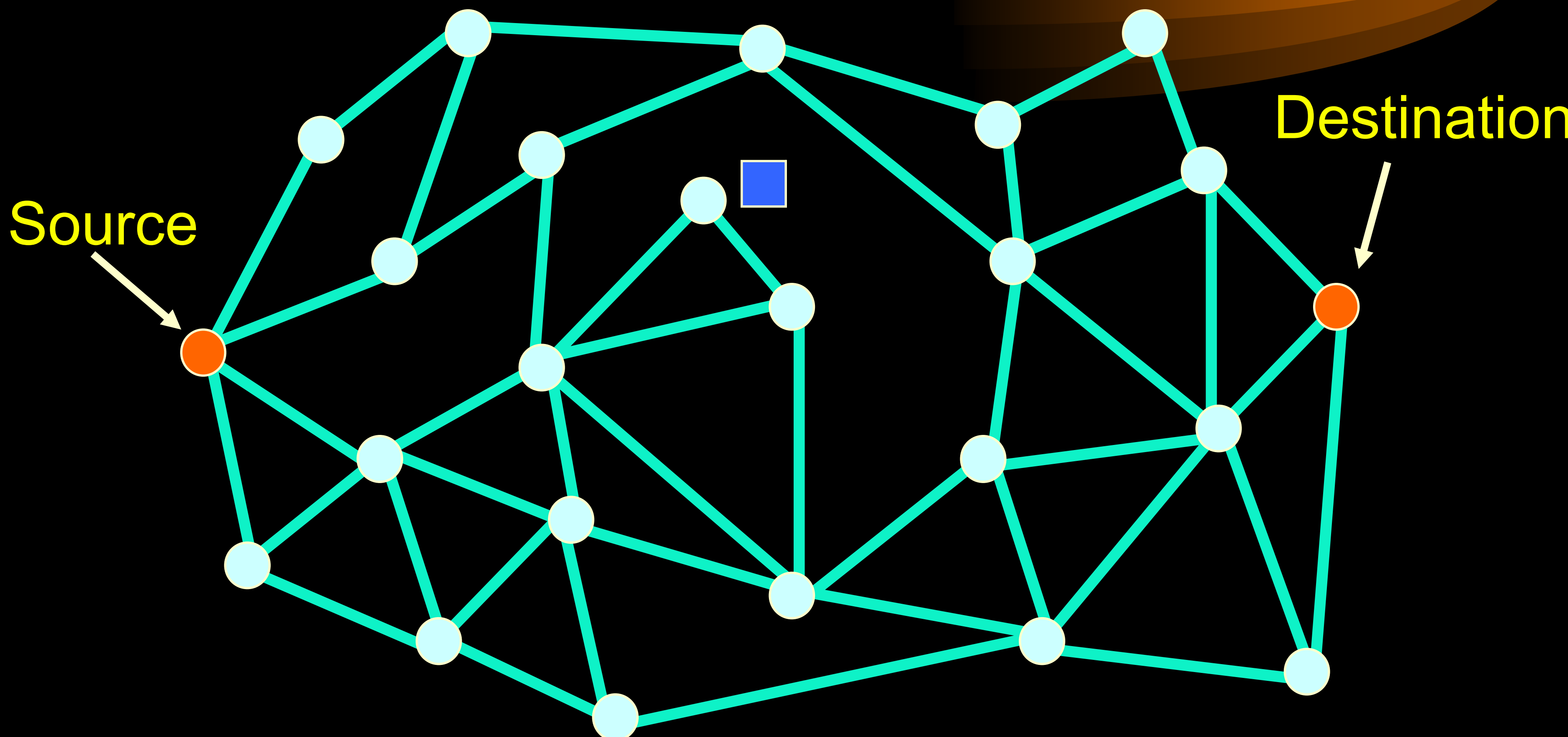
Problem

Nodes do not know enough to determine the "correct" forwarding direction.

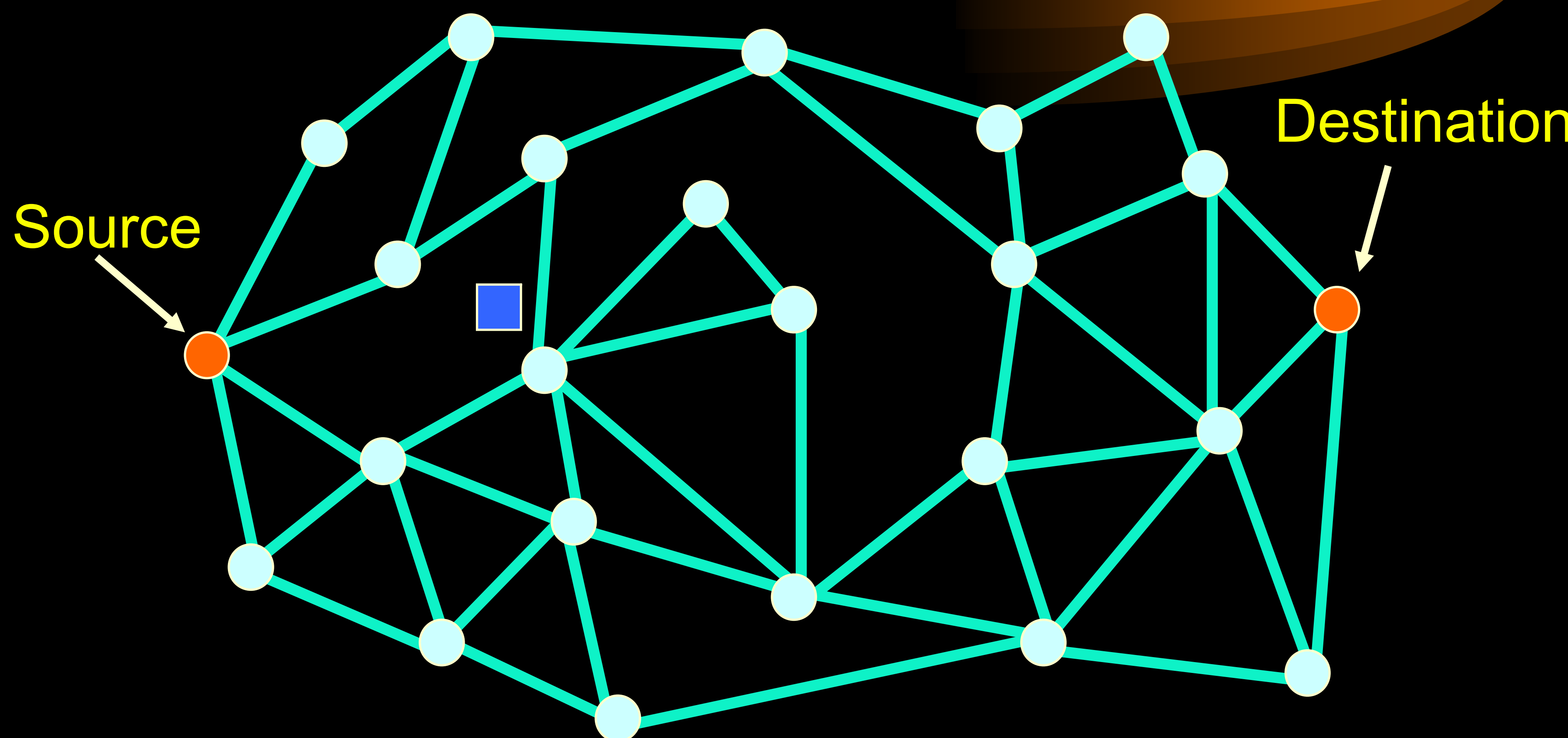
Bad Choice Example



Bad Choice Example



Bad Choice Example



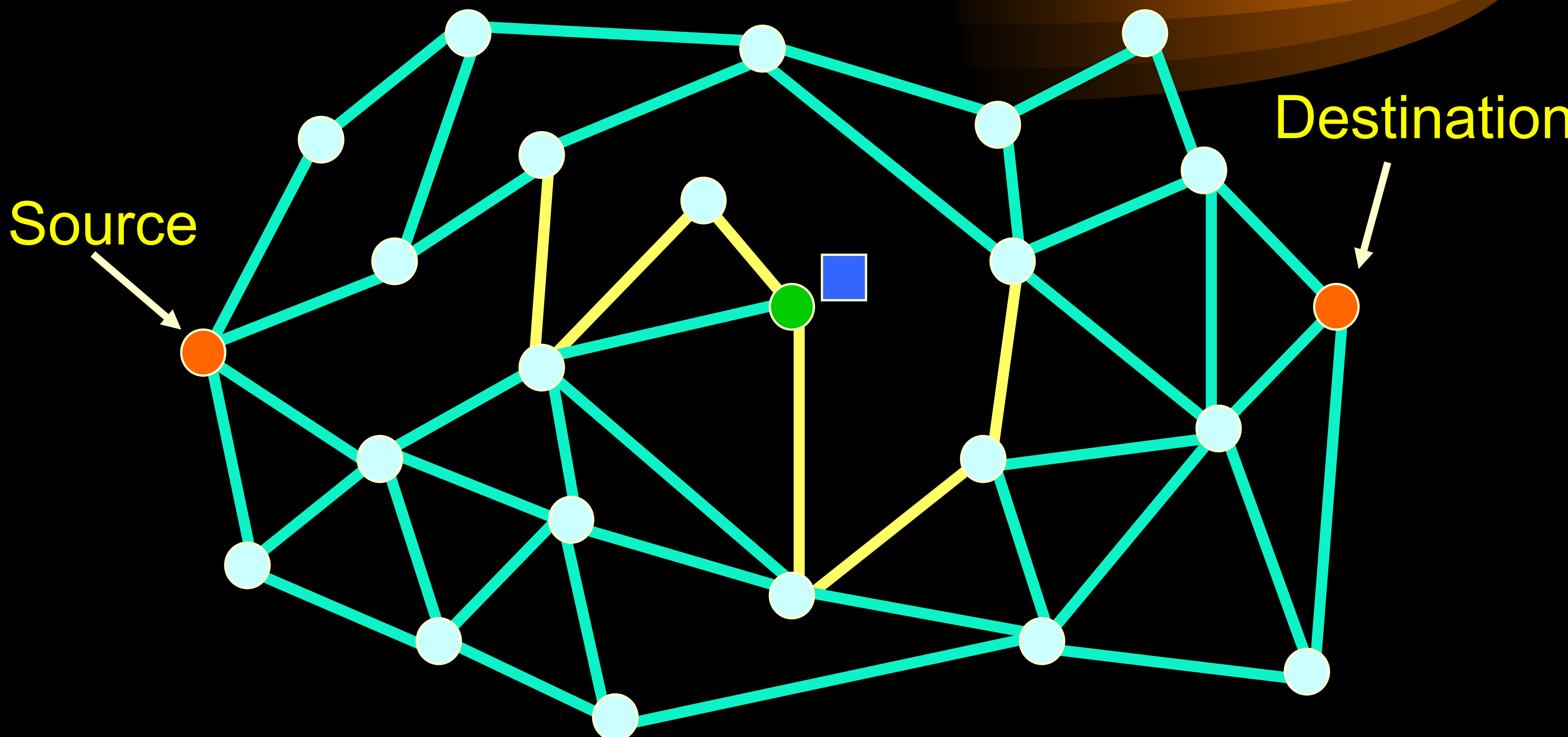
Hypothesis

By maintaining several hops of information along each planar face, we can make a better choice when deciding how to traverse a face

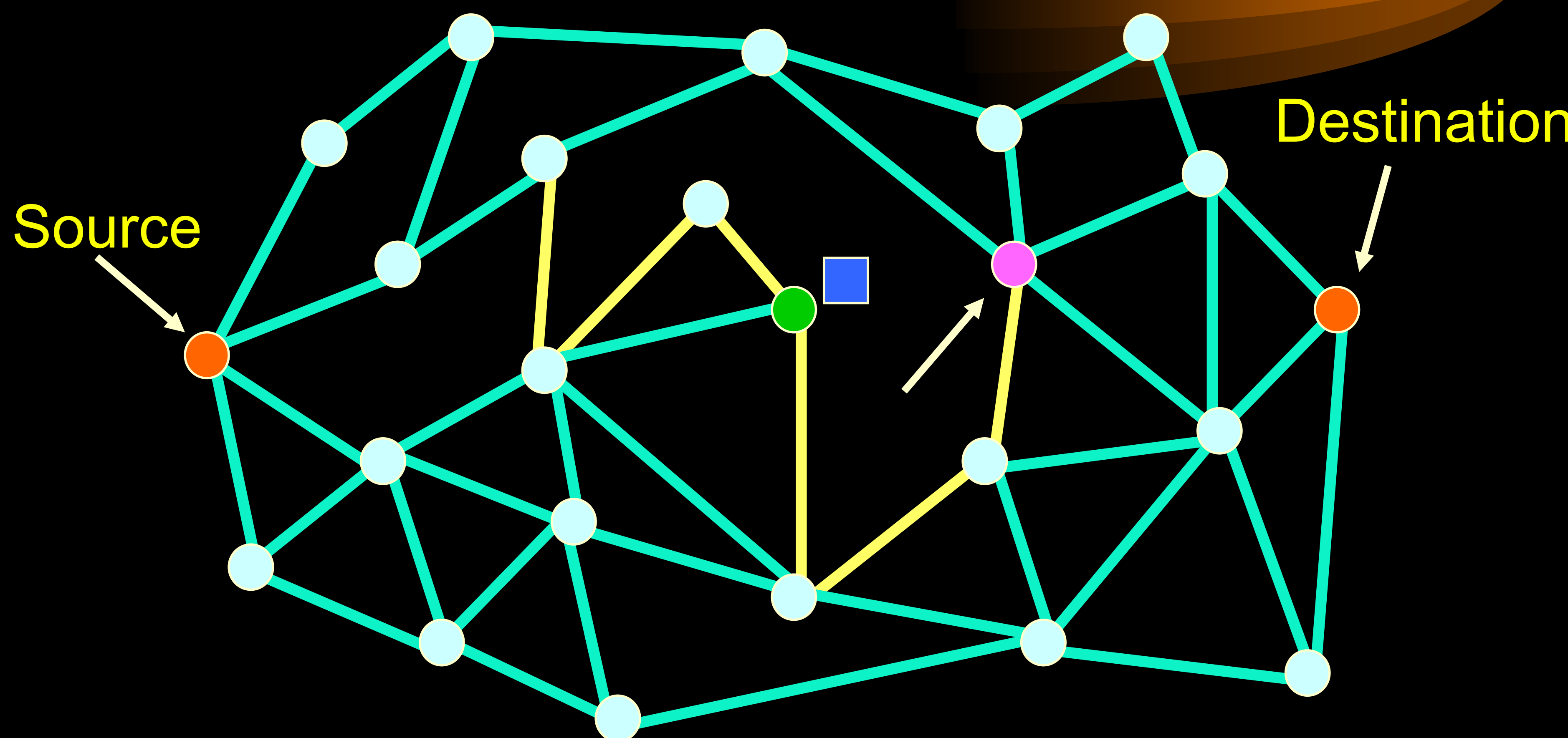
Greedy Path Vector Face Routing (GPVFR)

- Three modes:
  1. Forward greedily if possible.
  2. Use face information to forward along existing face
  3. Fallback on face traversal (GPSR)
- Revert to greedy forwarding as soon as it is feasible

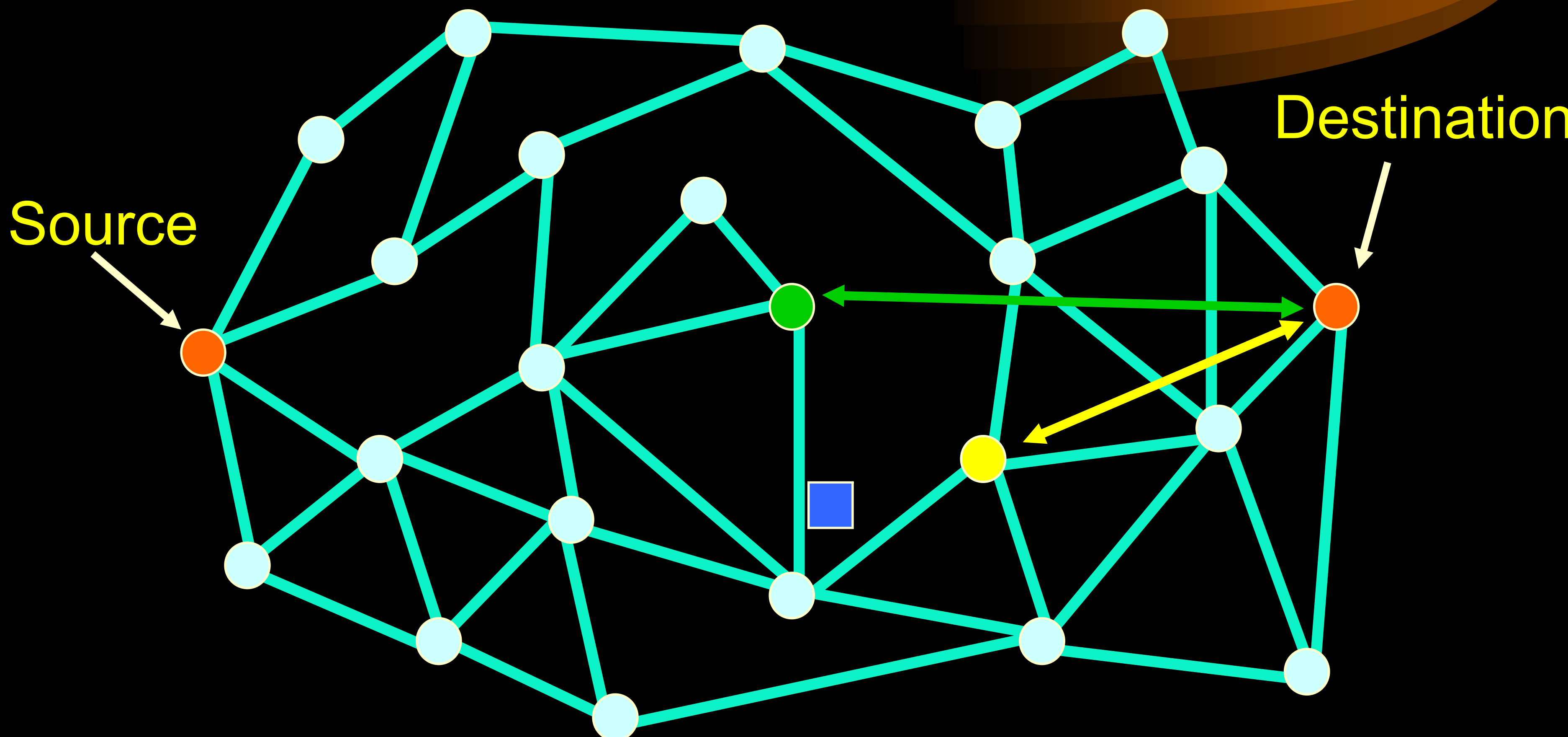
Using Face Information



Using Face Information



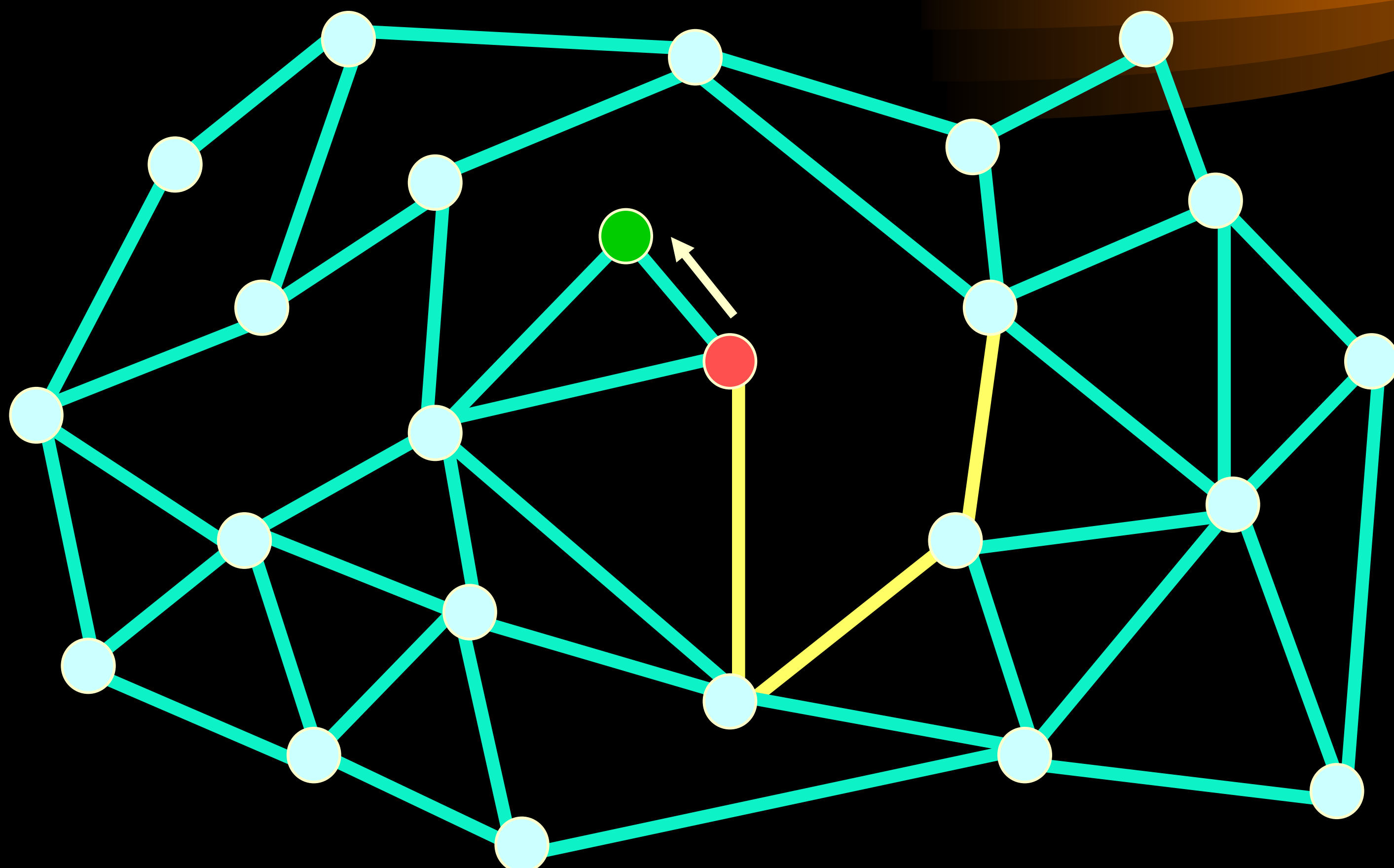
Revert to Greedy Mode



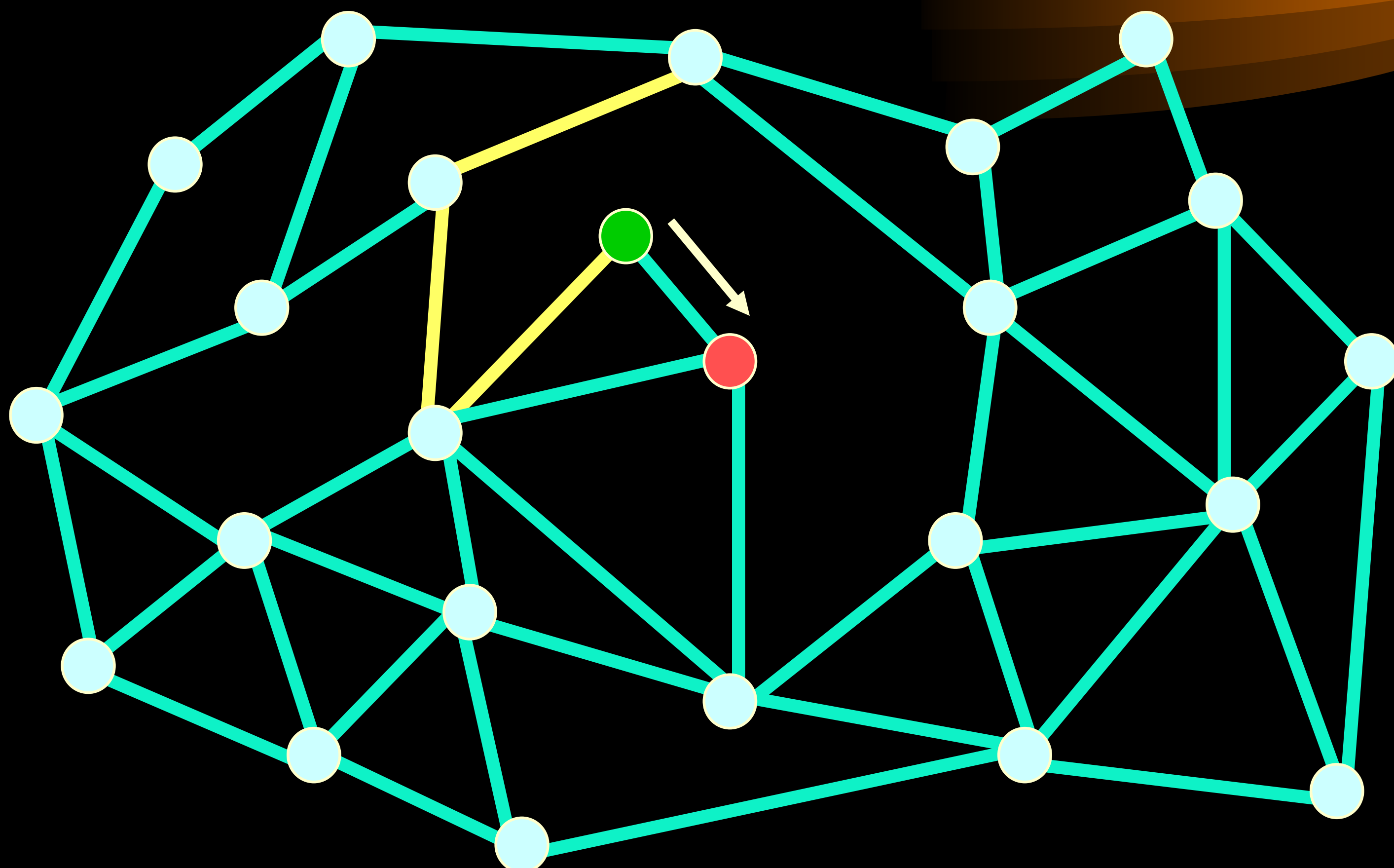
Path Vector Exchange (PVEX)

- Protocol for maintaining face information
- Nodes periodically exchange path vectors with planar neighbors
  - *h* hops of information
- Information is piggybacked on *keepalive* messages

Maintaining Face Information



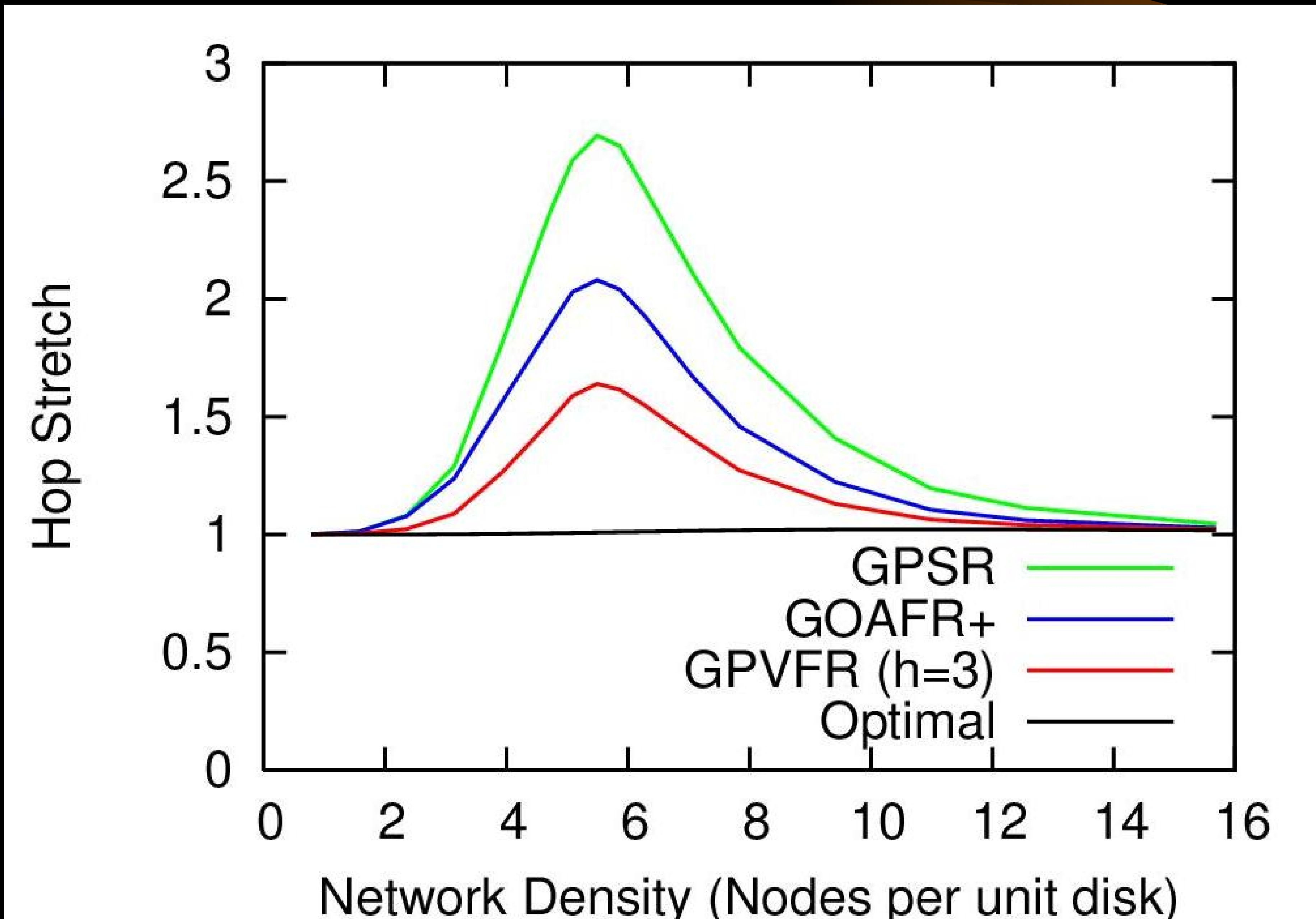
Maintaining Face Information



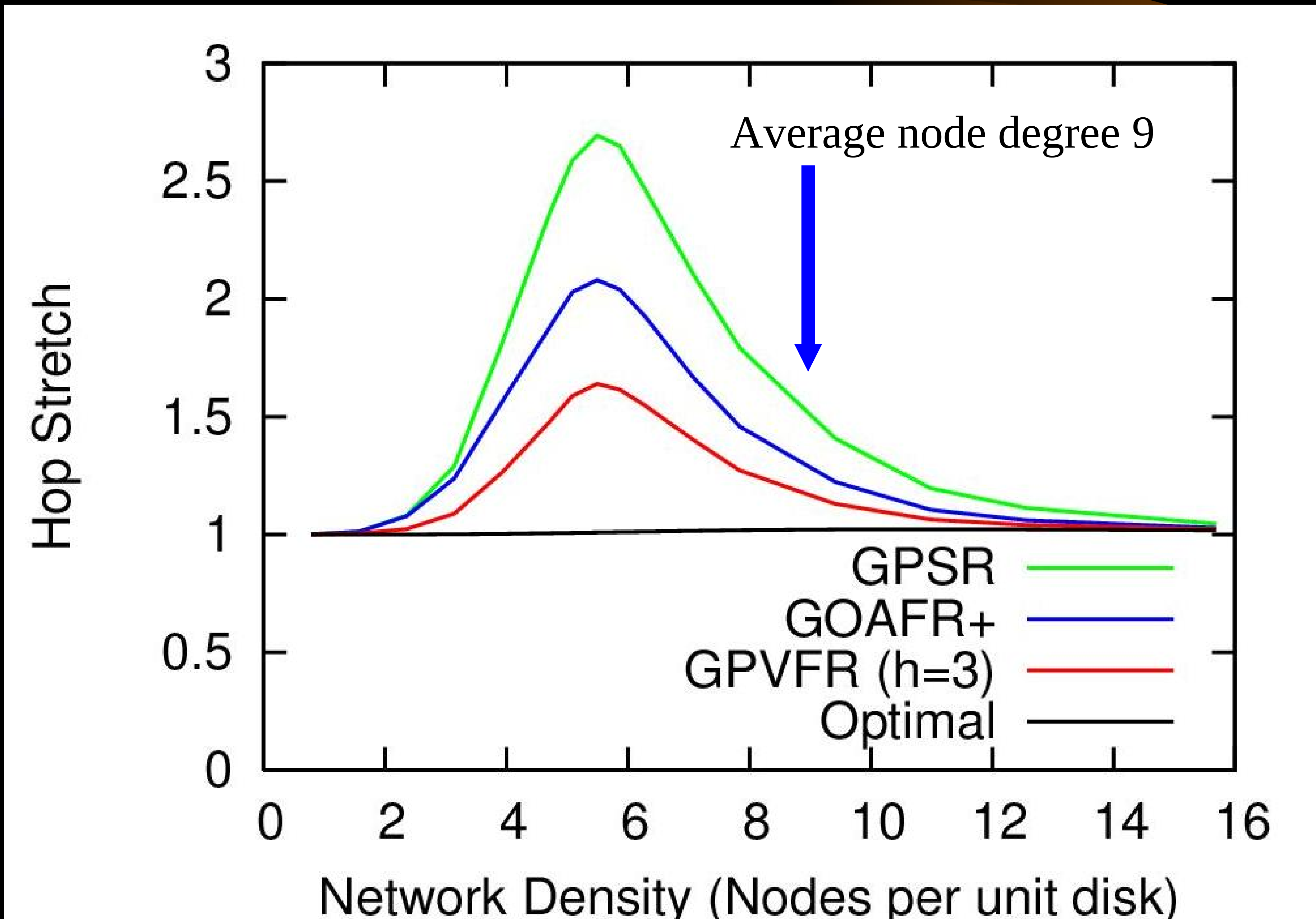
Simulation Results

- Measured 2 routing metrics:
  - Path Stretch
  - Hop Stretch
- Random networks over a range of network densities
- Compare to GPSR (Karp, 2001) and GOAFR+ (Kuhn, 2003)
- Results for RNG and GG planarization in paper

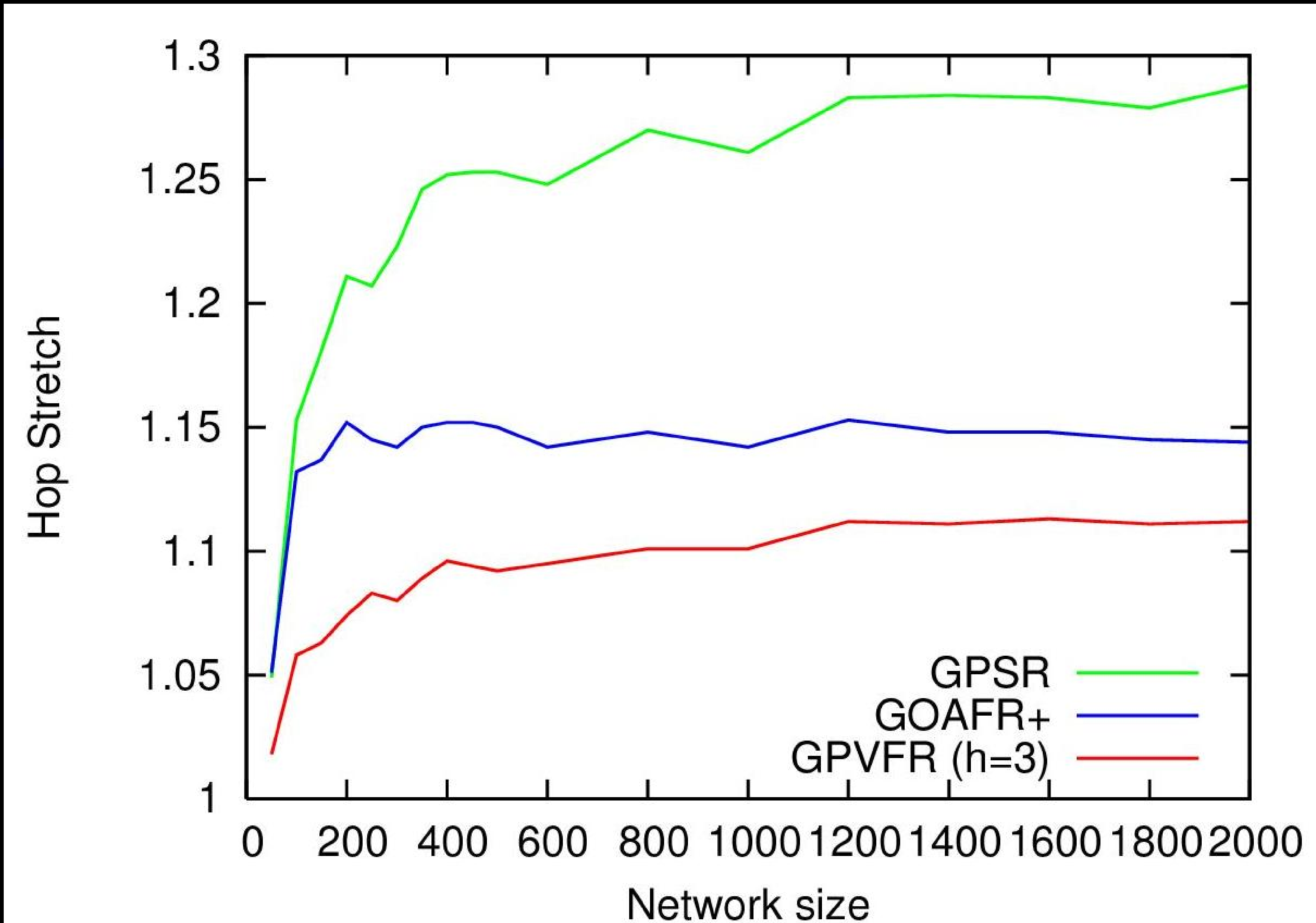
Hop Stretch (CLDP Planarization)



Hop Stretch (CLDP Planarization)

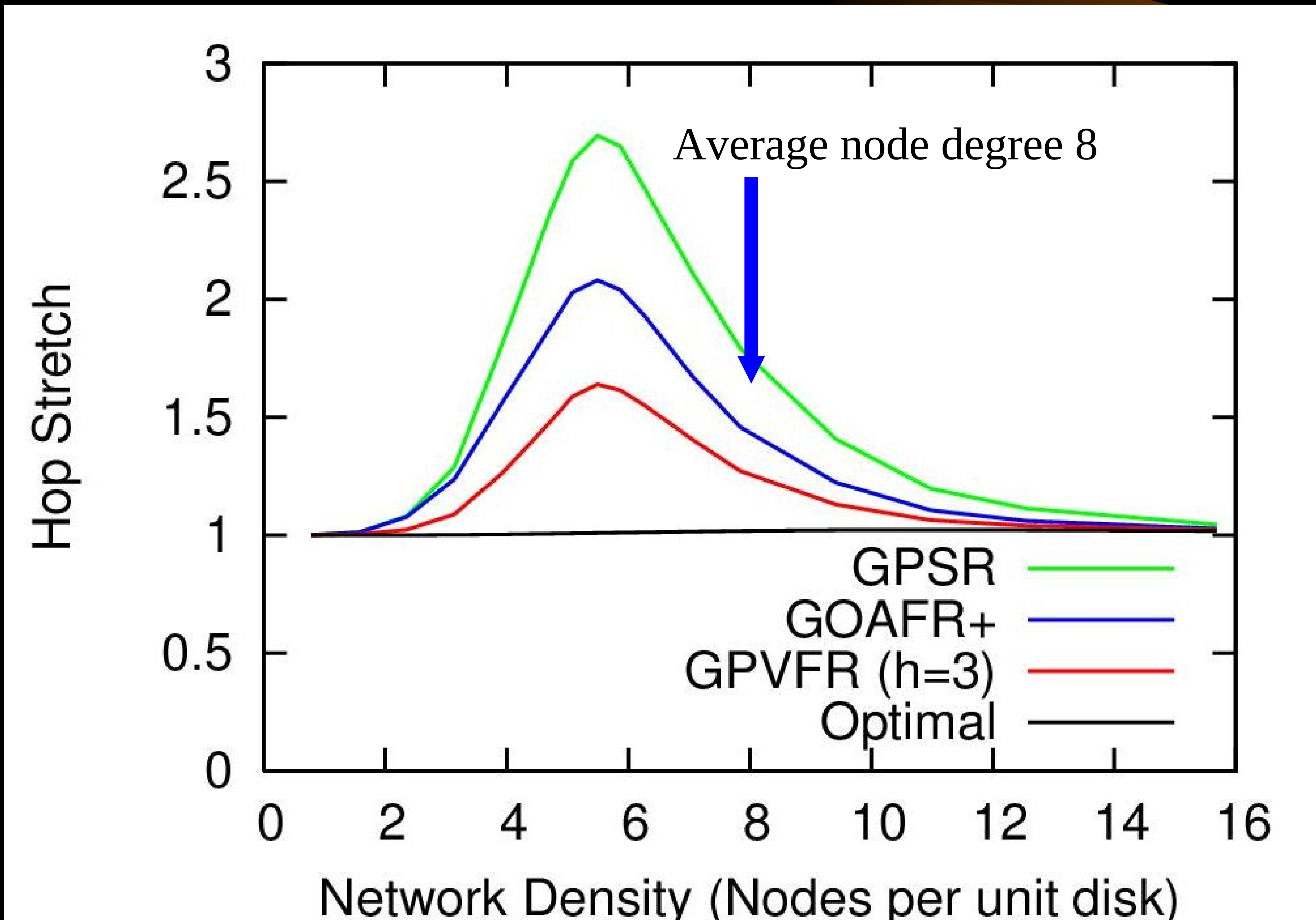


Scaling up

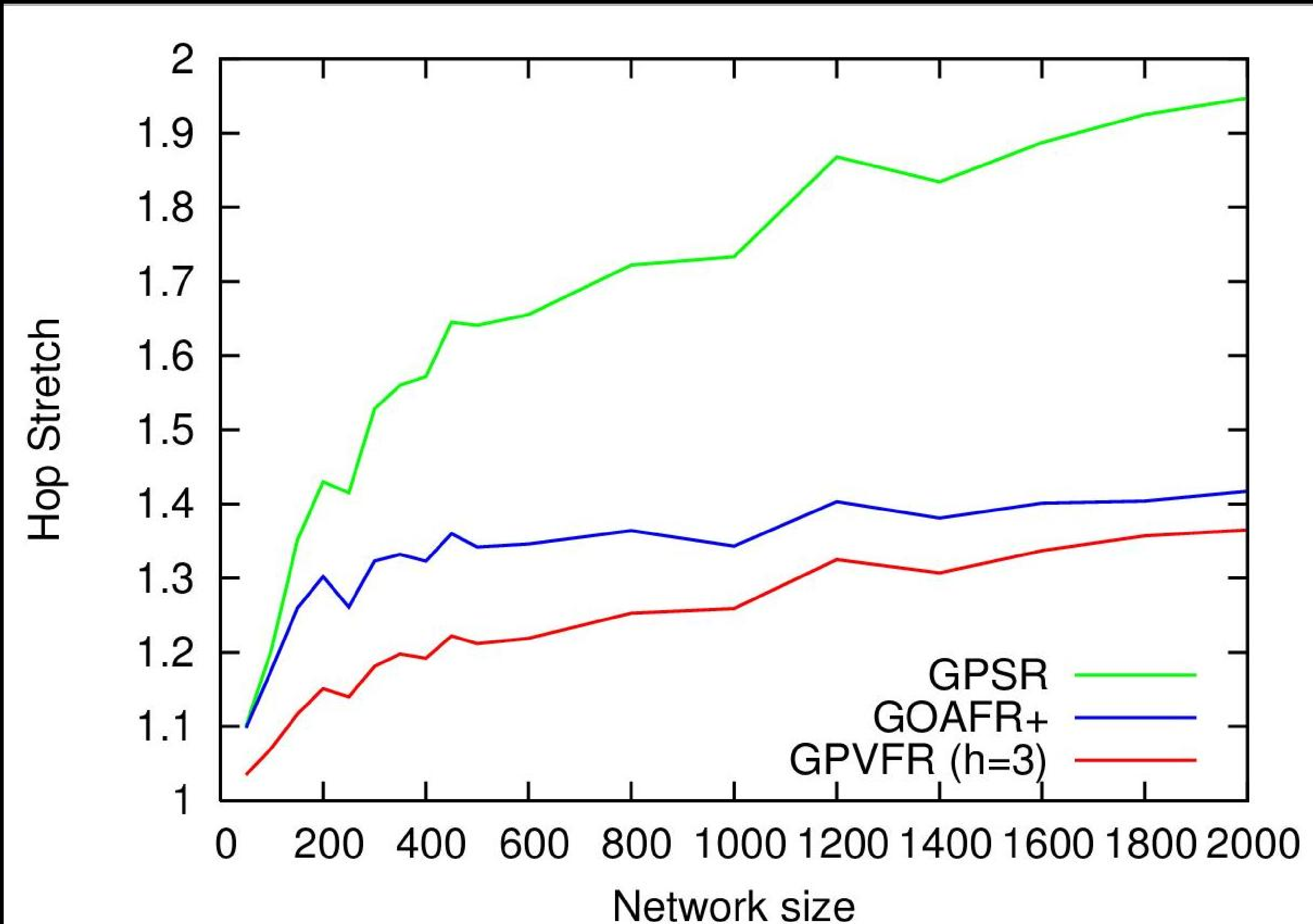


Average node density = 9

Hop Stretch (CLDP Planarization)

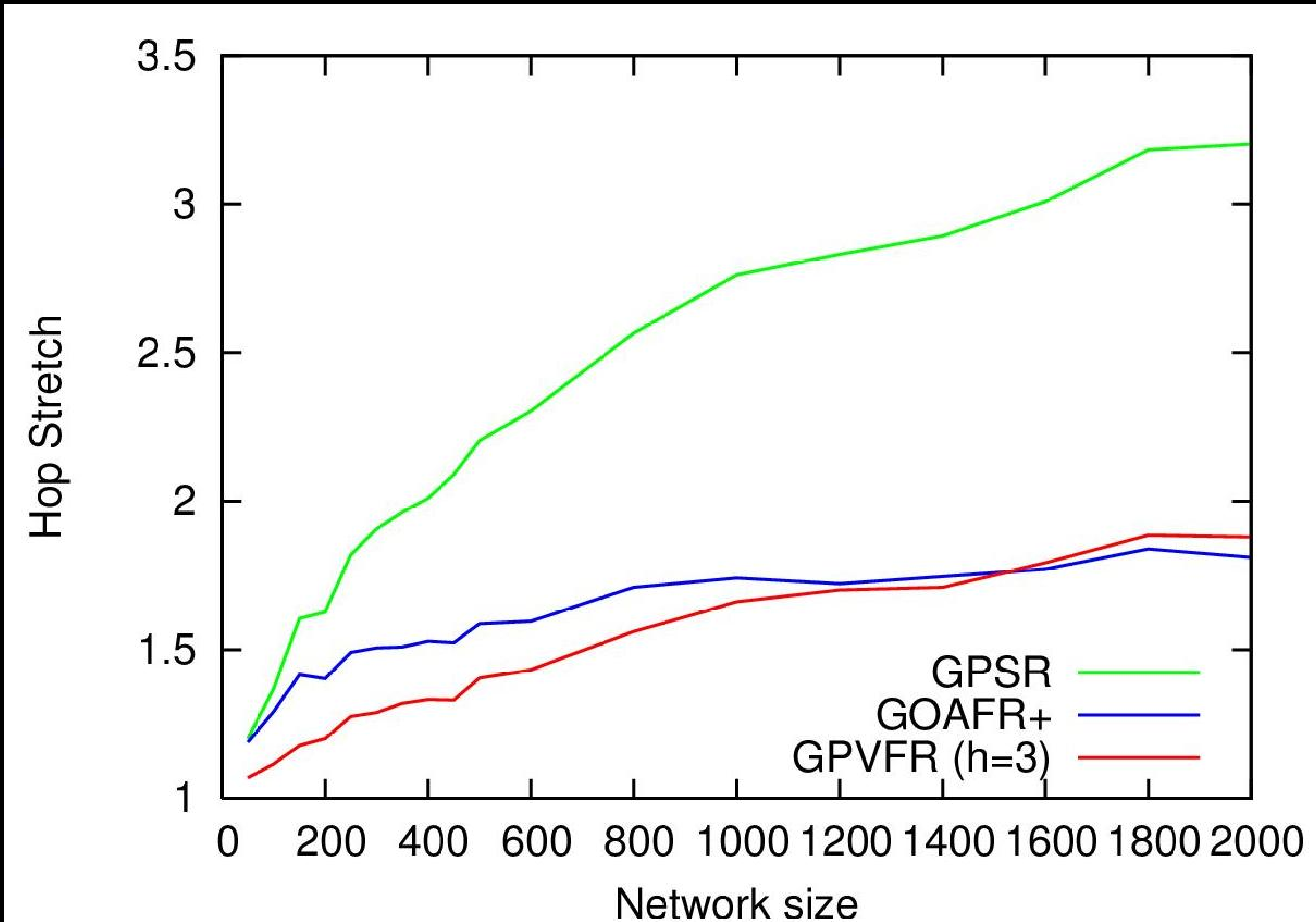


Scaling up



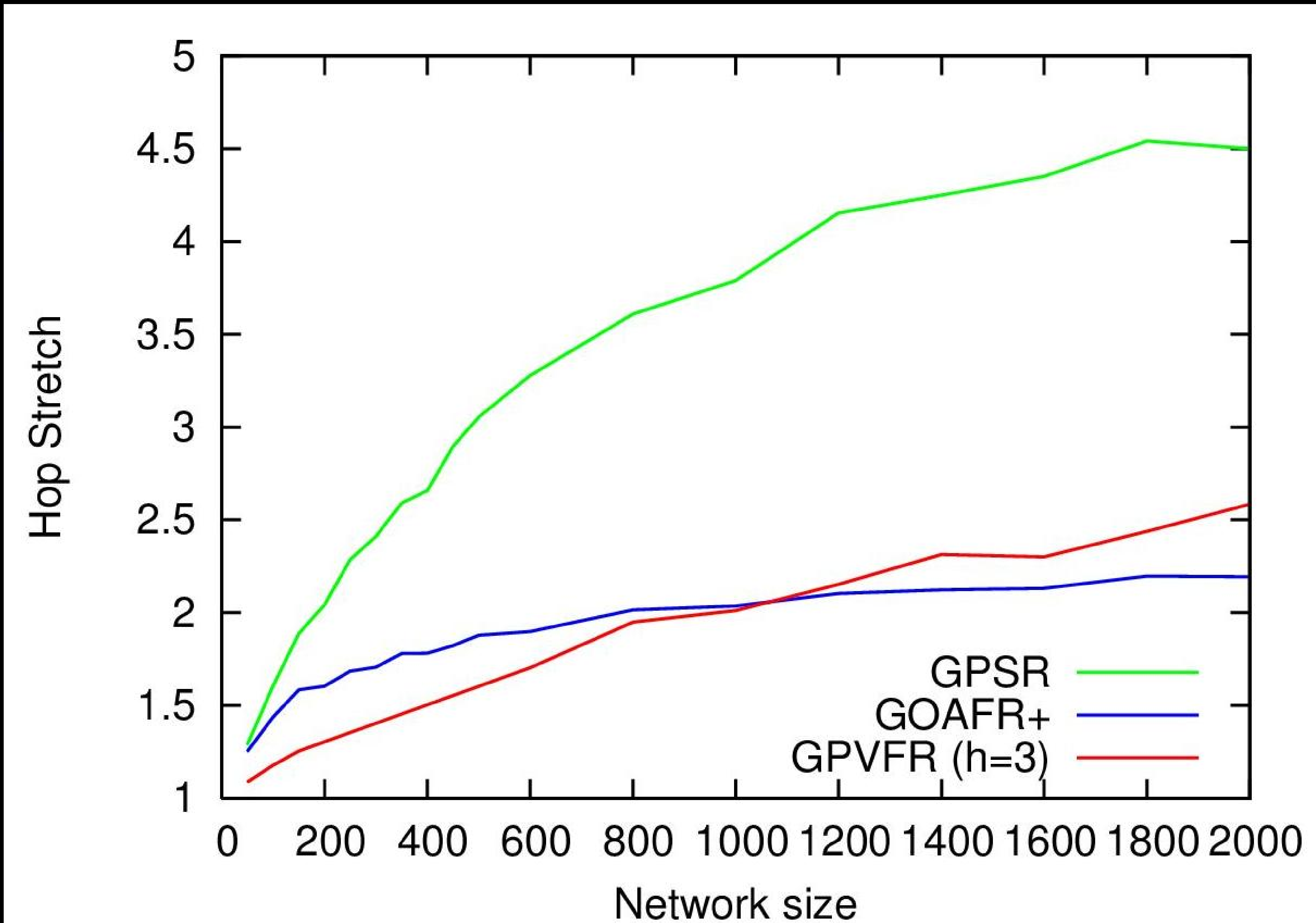
Average node density = 8

Scaling up



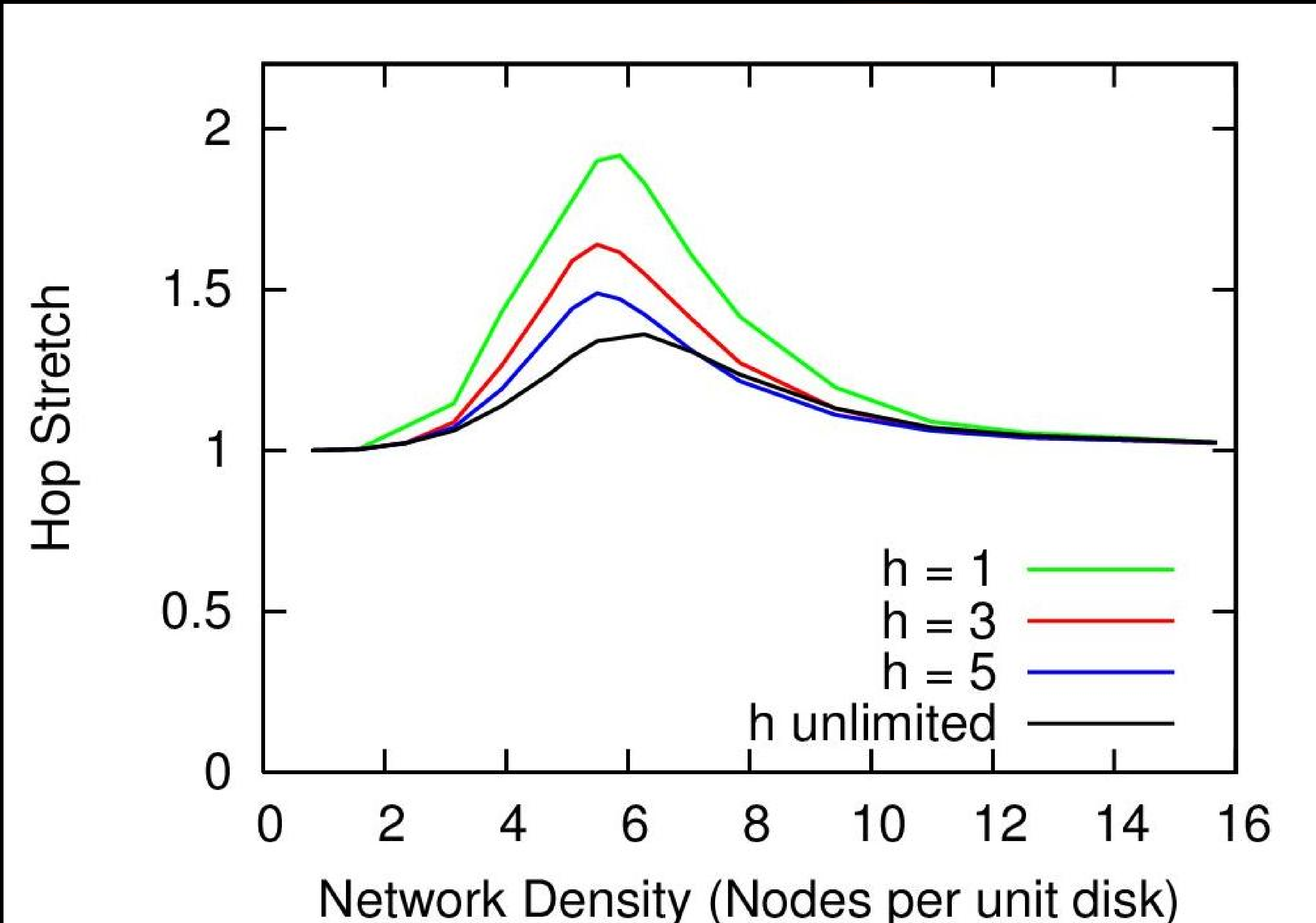
Average node density = 7

Scaling up



Average node density = 6.5

Varying Path Vector Length



Maintenance Cost

- Additional storage:
  - Small (15 to 20 extra nodes on average, < 200 bytes)
  - proportional to number of planar neighbors
  - independent of network density
- Additional bandwidth:
  - *h* message exchanges (each < 200 bytes)
- Planarization cost >> PVEX cost

Theoretical results

1. With full face information, we can route obliviously;
2. Without full face information, it is impossible to route obliviously.

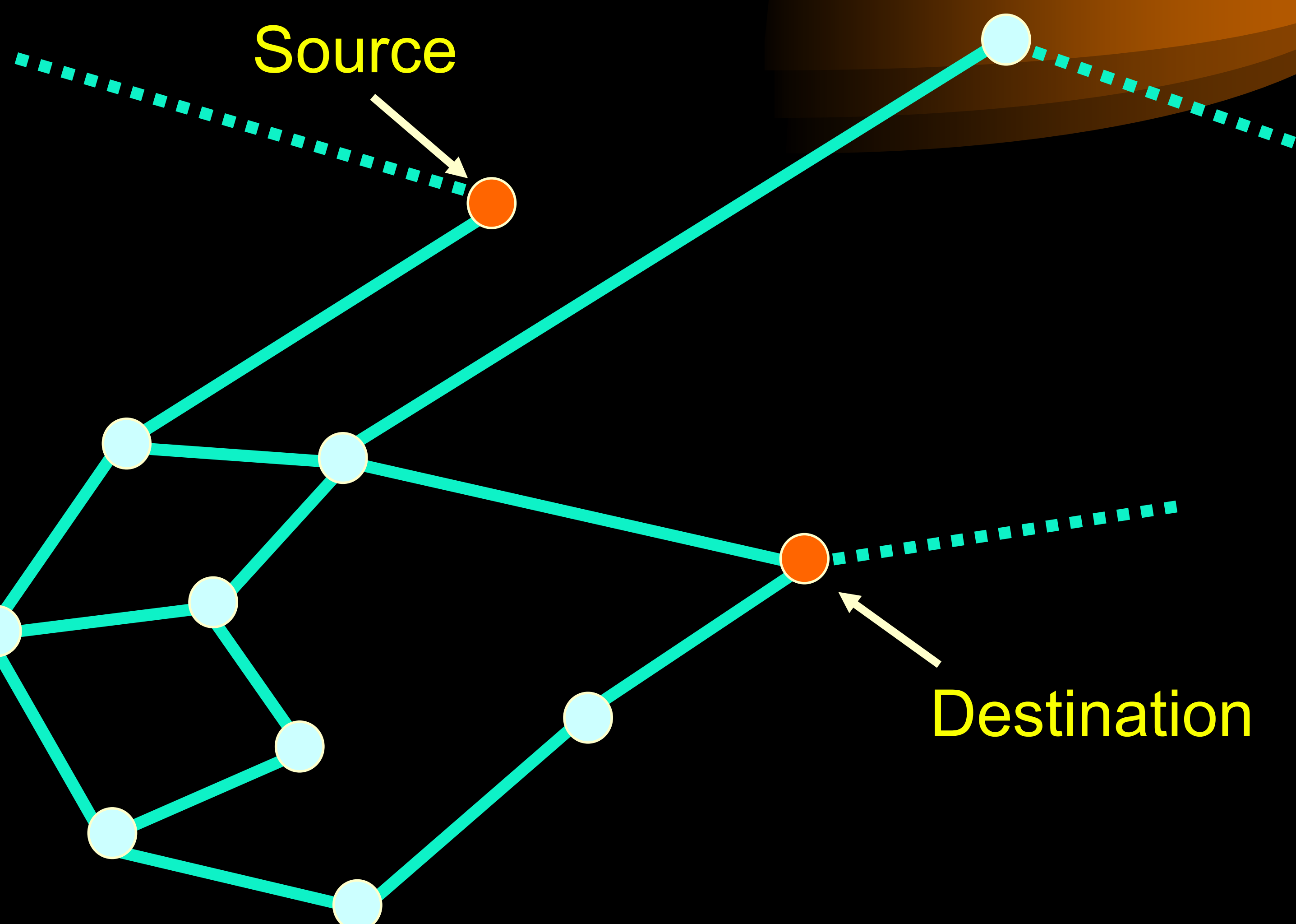
Conclusion

- Forwarding direction is critical for good performance
- GPVFR achieves significantly improved routing stretch with a little extra storage.

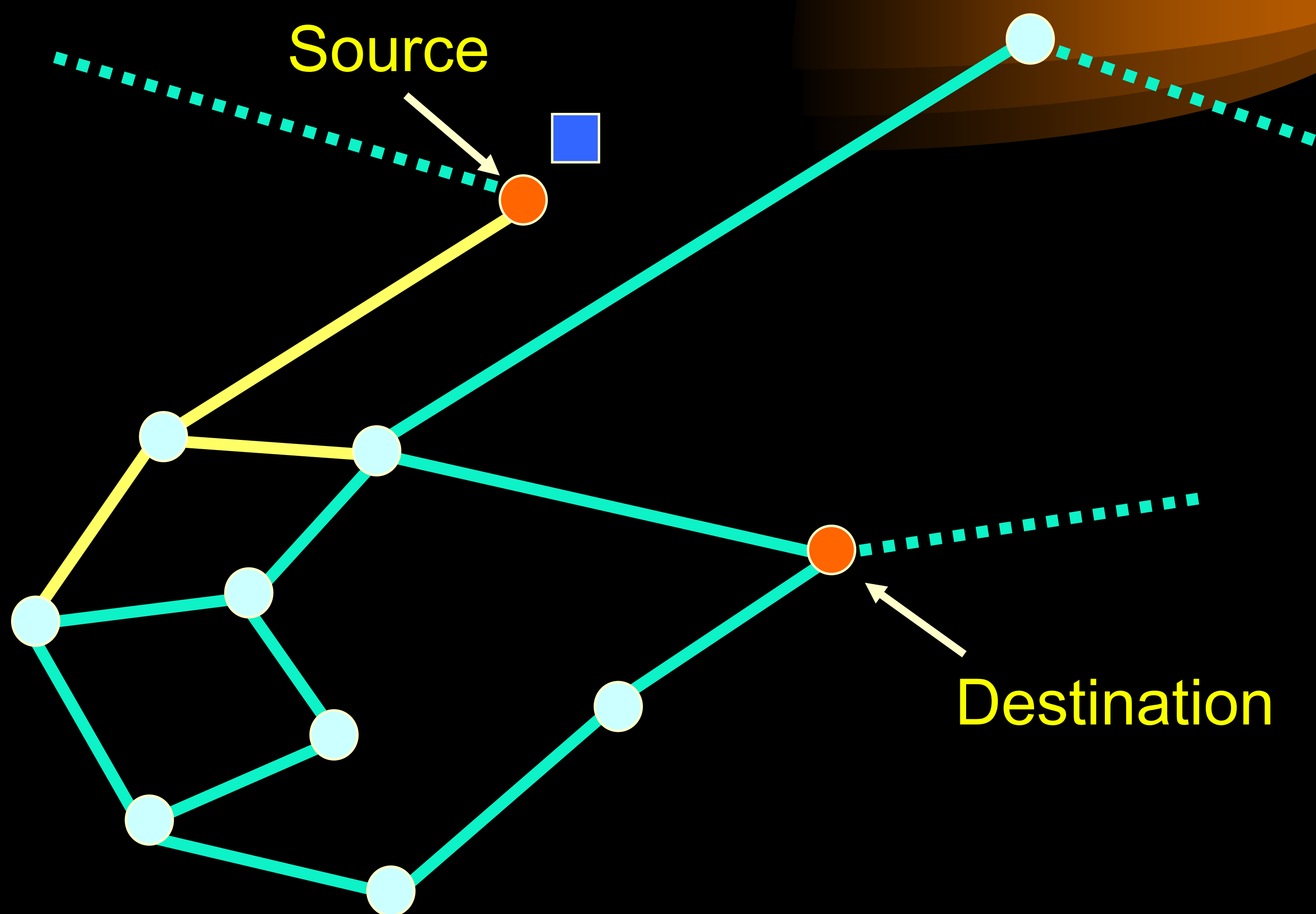
Path Vector Face Routing: Geographic Routing with Local Face Information

Ben Leong, Sayan Mitra, and Barbara Liskov
MIT CSAIL

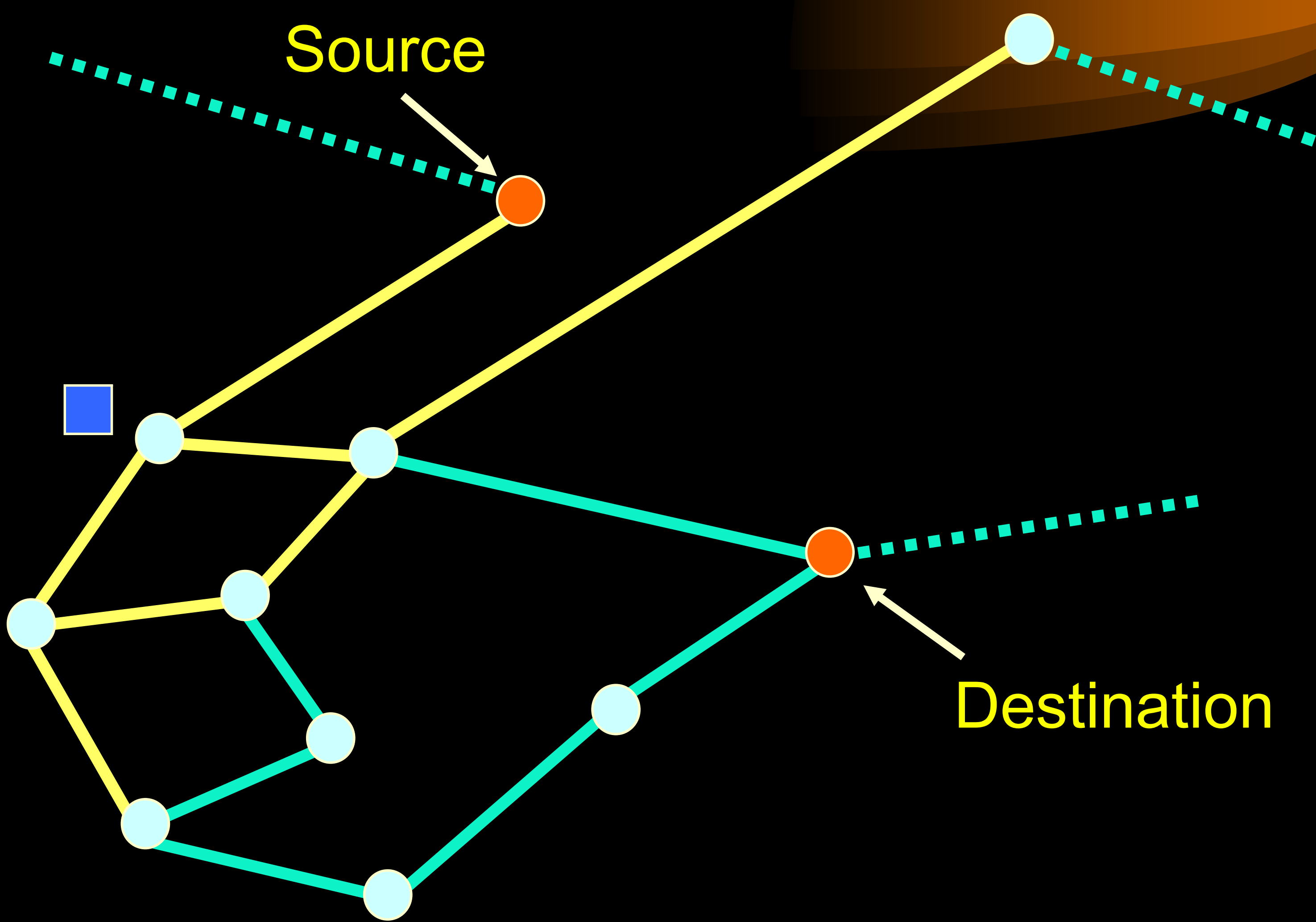
Why unlimited face information can be bad



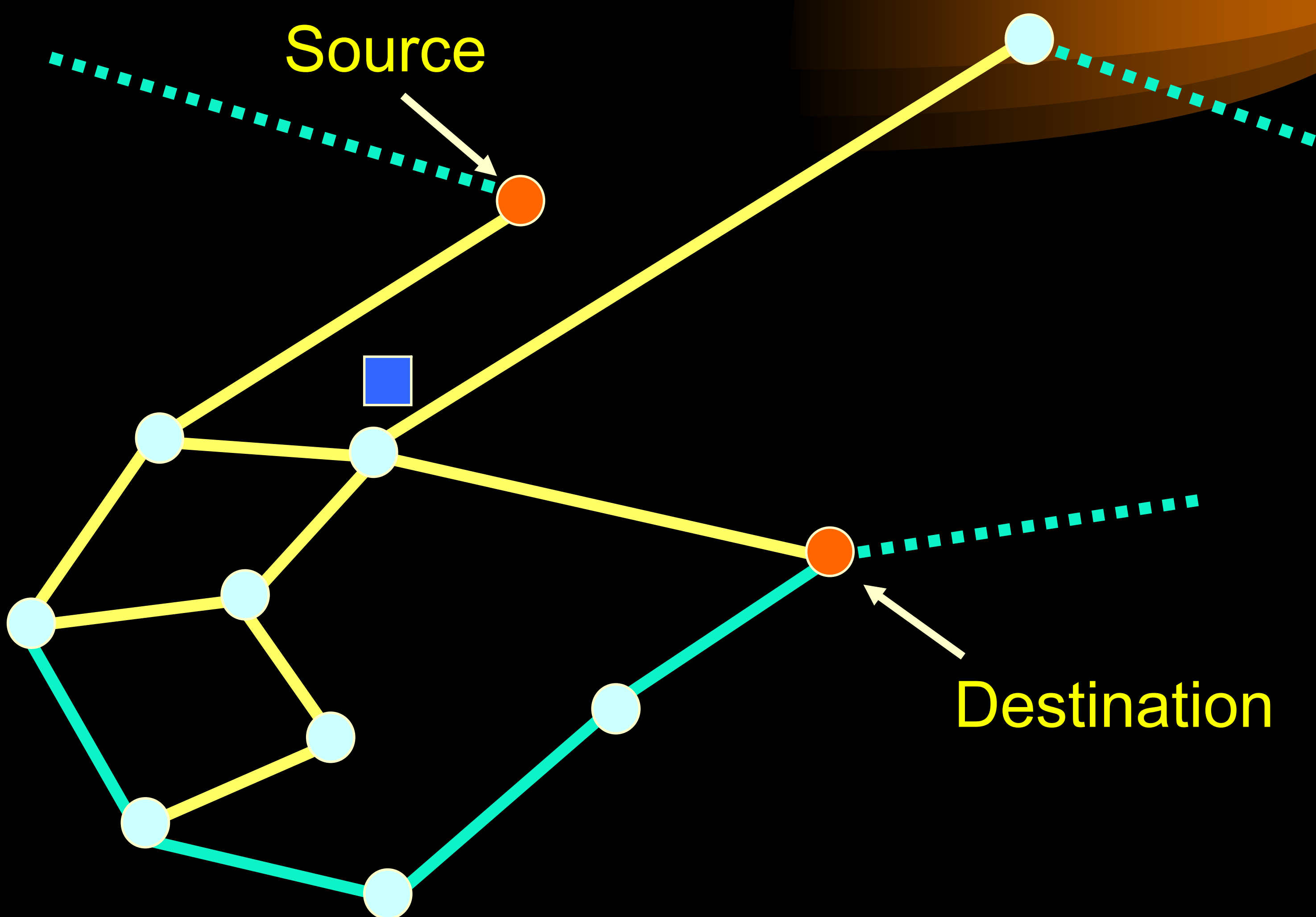
Why unlimited face information can be bad



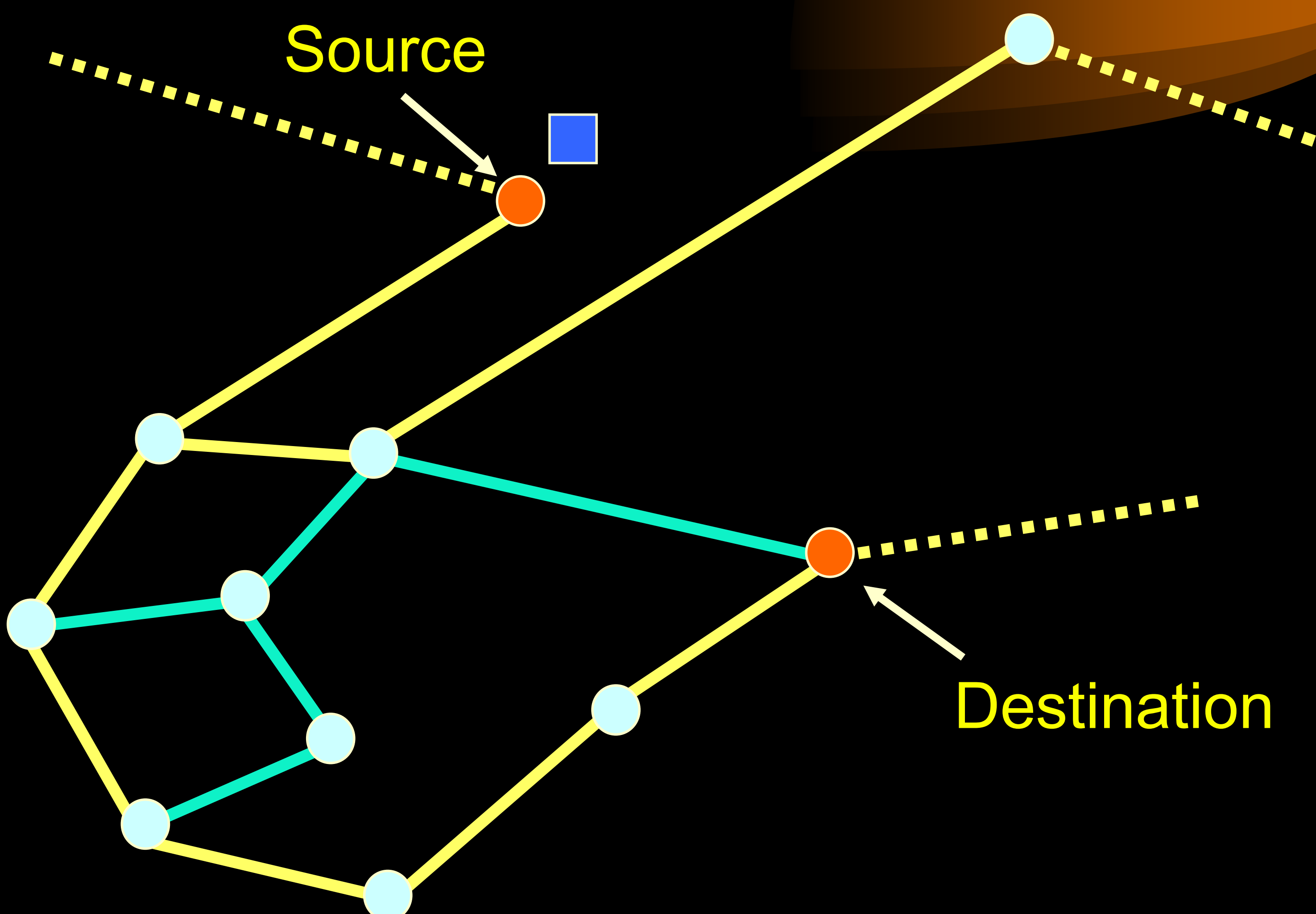
Why unlimited face information can be bad



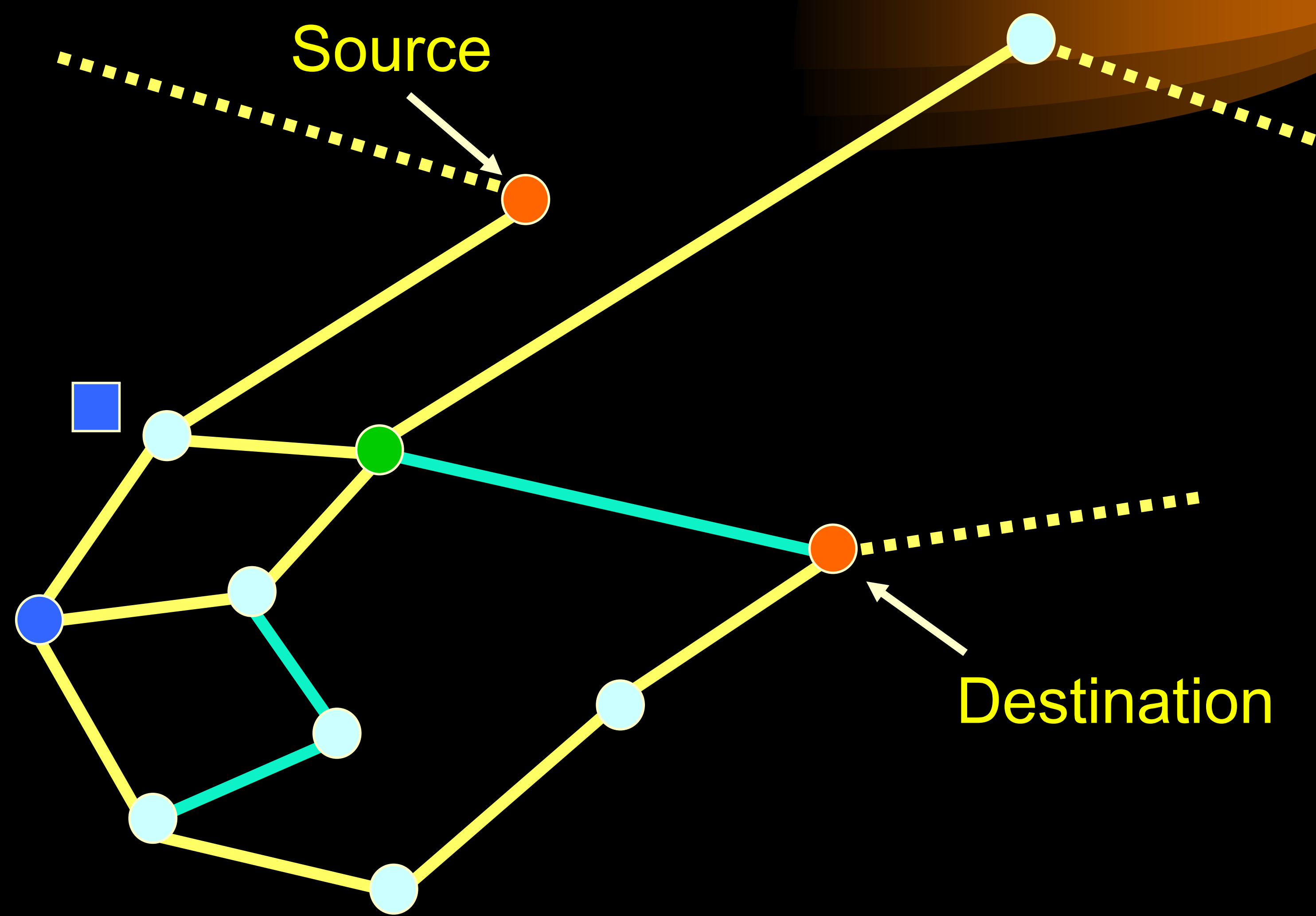
Why unlimited face information can be bad



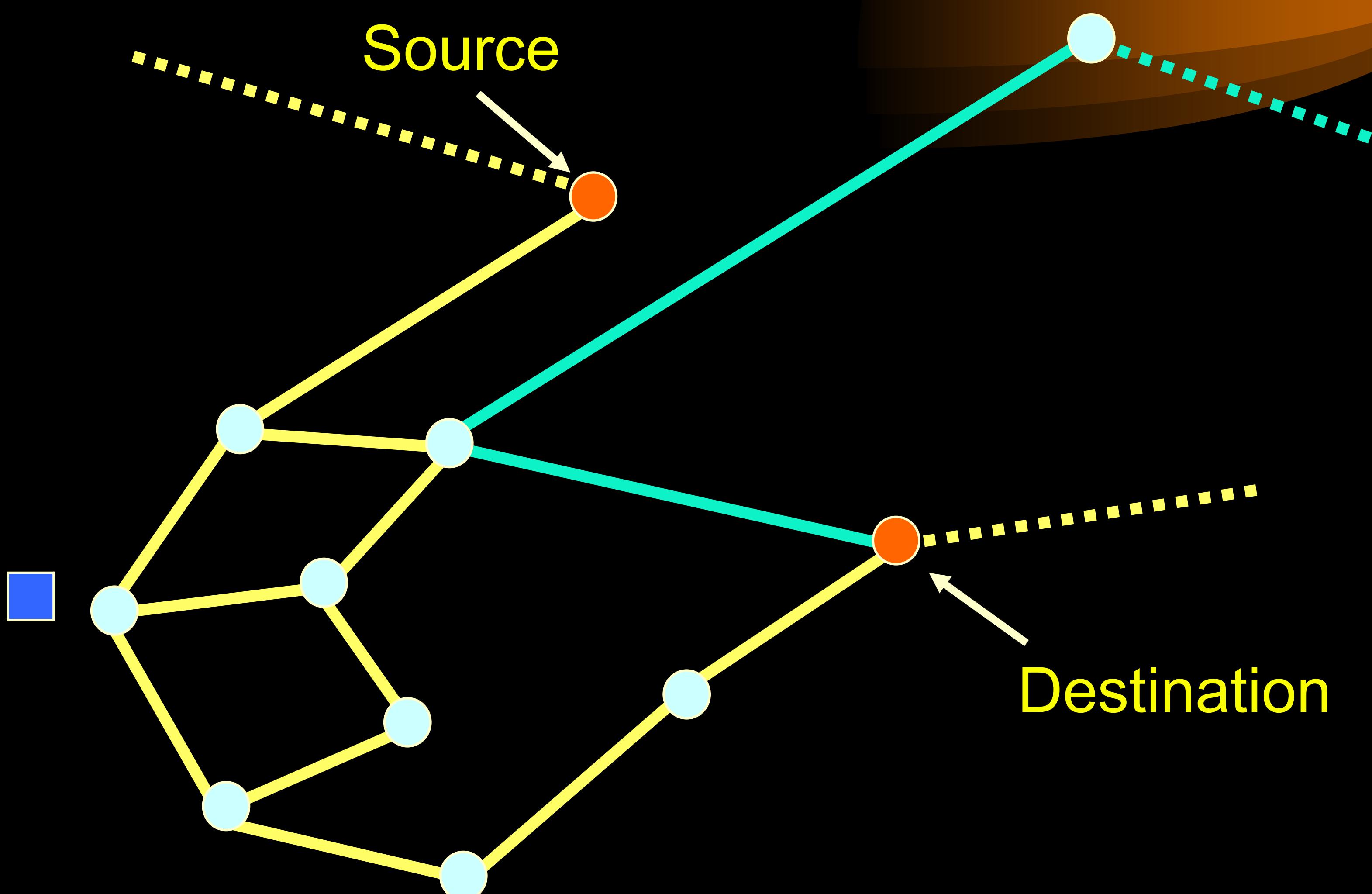
Why unlimited face information can be bad



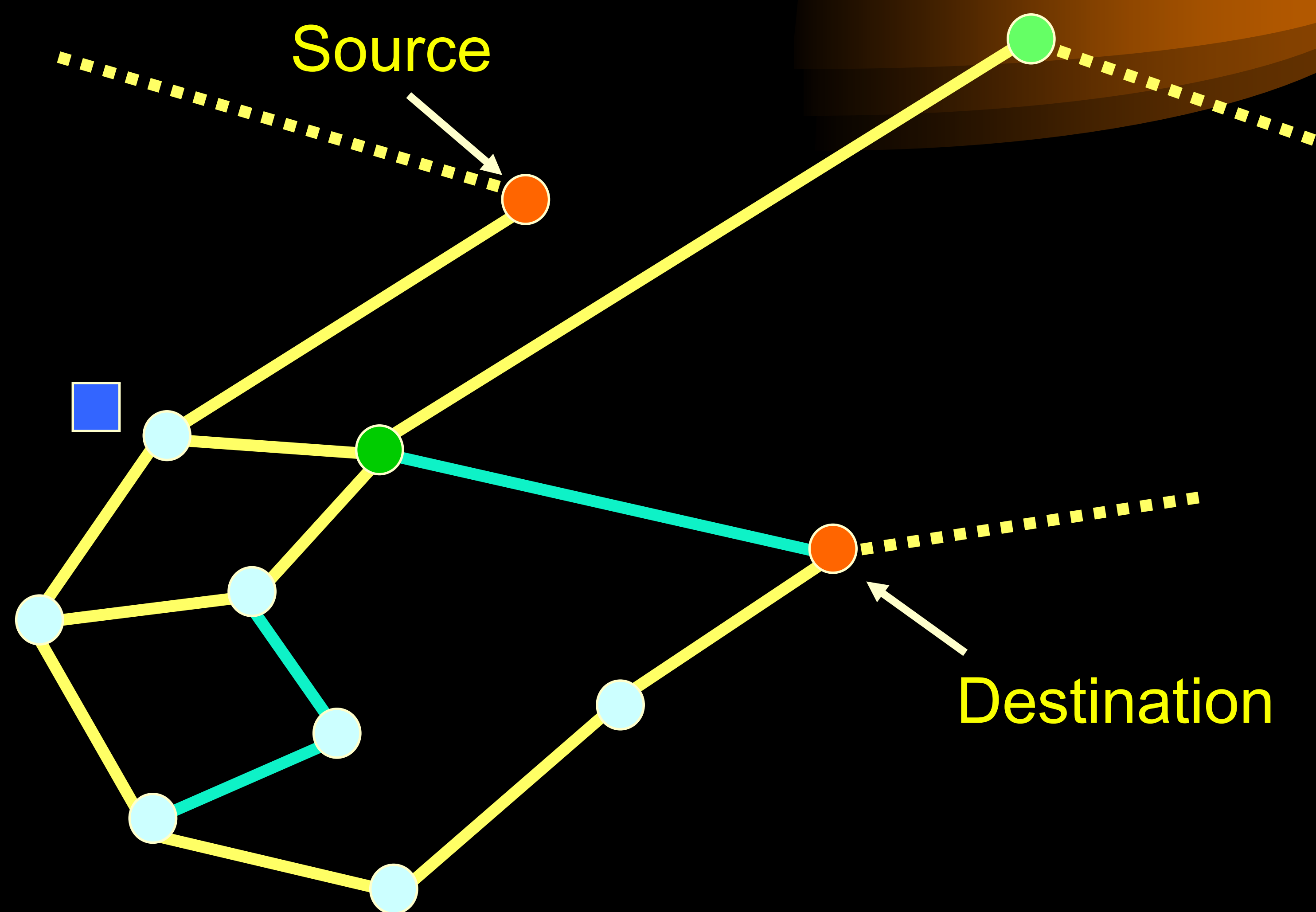
Why unlimited face information can be bad



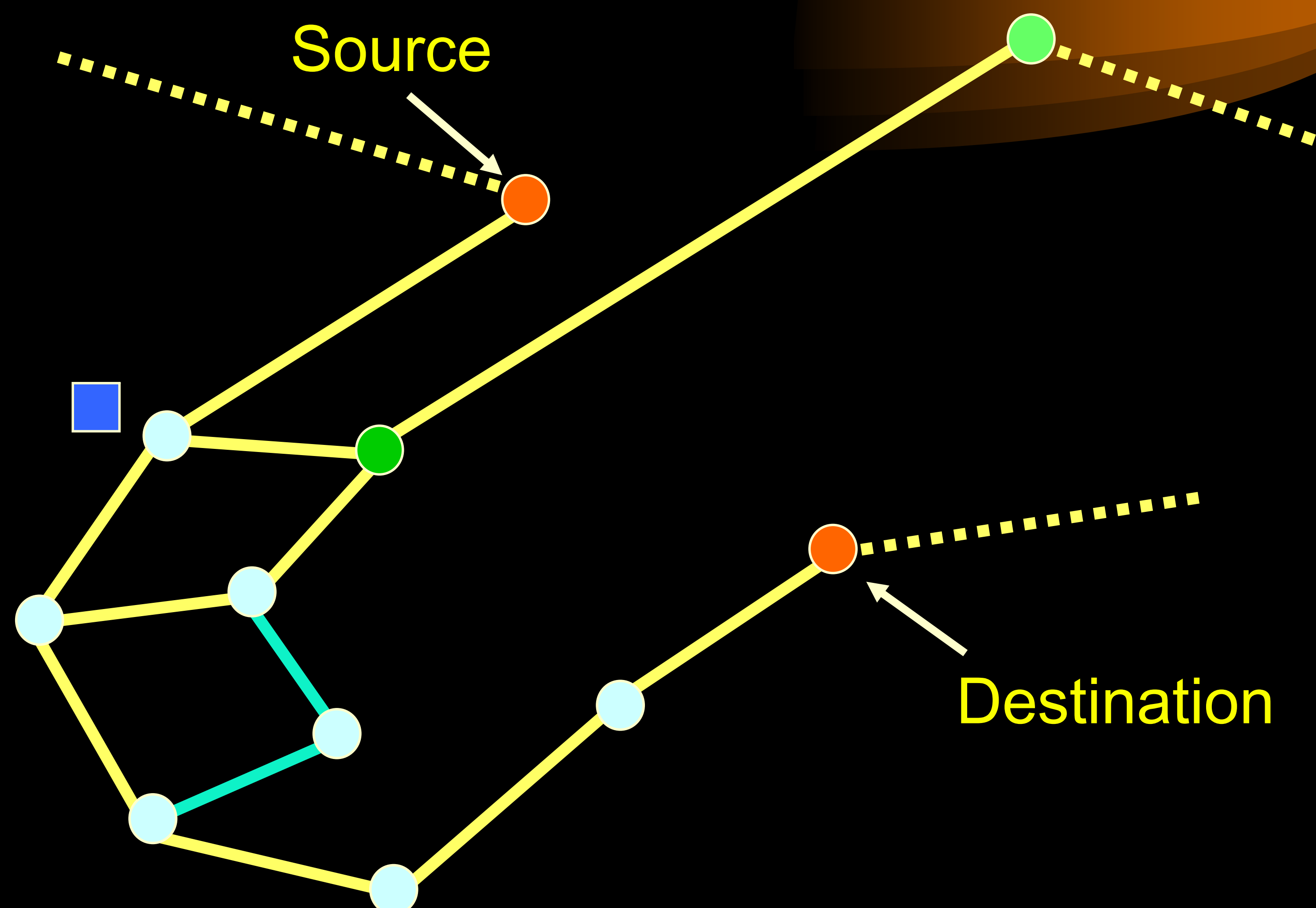
Why unlimited face information can be bad



Why unlimited face information can be bad



Why unlimited face information can be bad



Theorem 1

Given a connected pair of nodes v and t in a planar graph G, assuming that every node in G completely knows all its faces, we can route from v to t obliviously

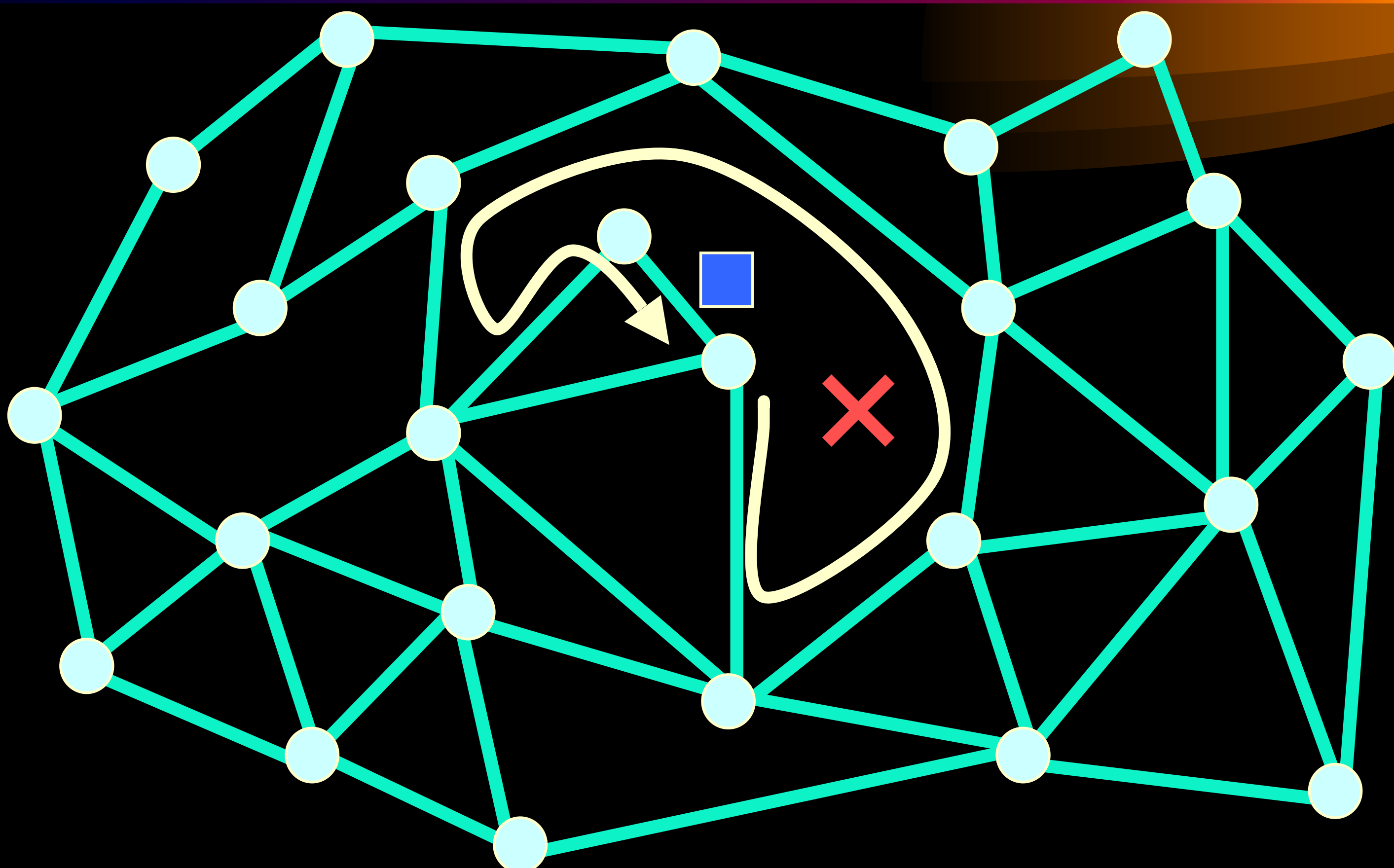
Theorem 1 Paraphrased

With full face information at each node, we can route without storing state in the packets

Oblivious Routing with Full Face Information (OPVFR)

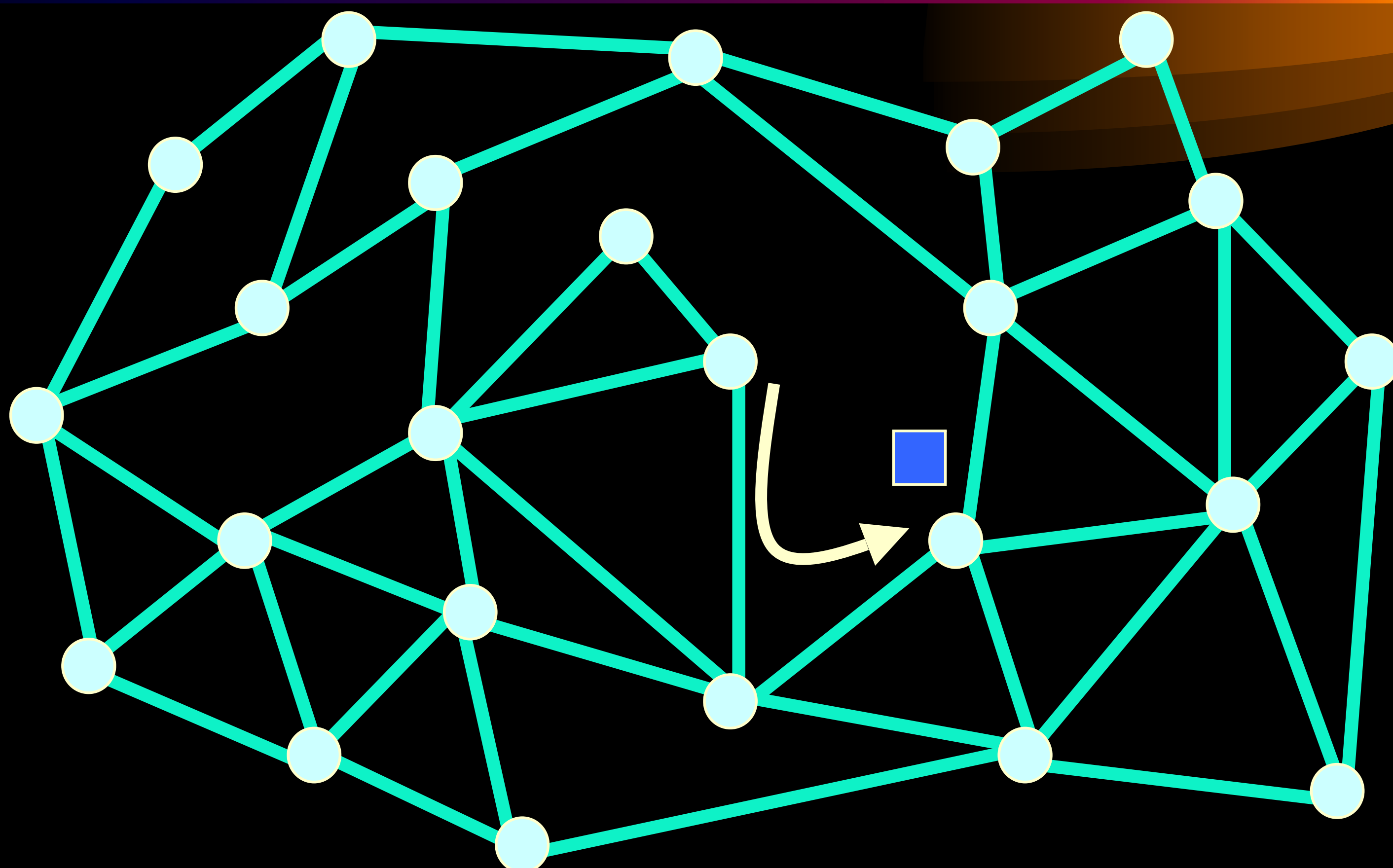
- Suppose all nodes have full face information
- Do:
  - Find **target node** and route towards it.
  - To find **target node**: find edge that is nearest to destination node among all faces. Node on edge that is nearer destination is target node.
- Break ties in some consistent way.

Non-oblivious Routing



- Need to know when we come back to the same node!

Non-oblivious Routing



- Need to know when to switch back to greedy

Theorem 2

For any given non-negative integer h, there does not exist a deterministic oblivious routing algorithm that guarantees packet delivery for all planar graphs if nodes are limited to knowing only about nodes that are up to h hops away

Theorem 2 Paraphrased

If nodes do not have full face information, it is impossible to always route correctly without storing some state in the packets.