

Wired Geometric Routing

Jonathan Ledlie

Michael Mitzenmacher

Margo Seltzer

Harvard University

Peter Pietzuch

Imperial College London

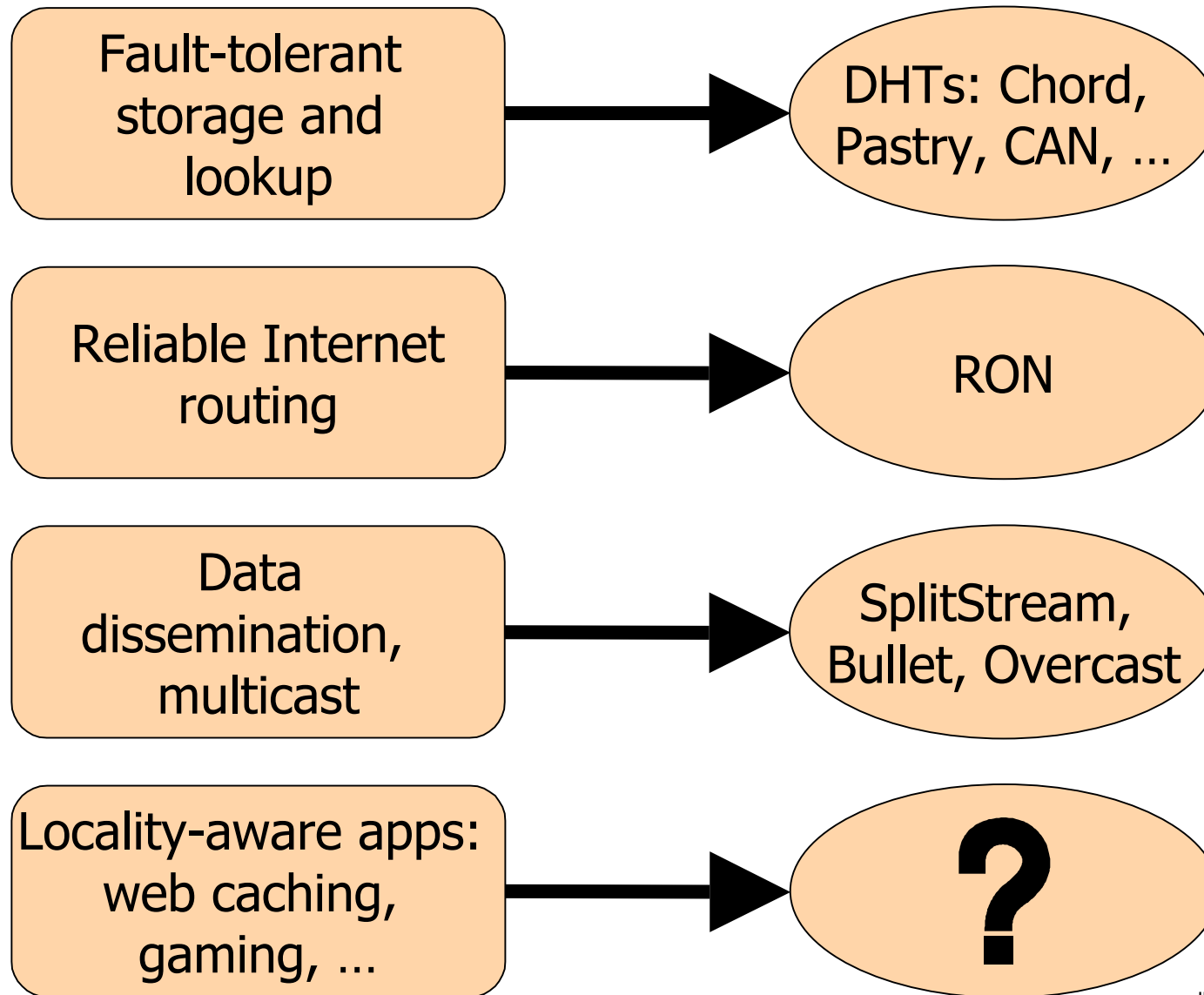
Hourglass Project

<http://www.eecs.harvard.edu/~syrah/hourglass>

IPTPS - Feb 2007

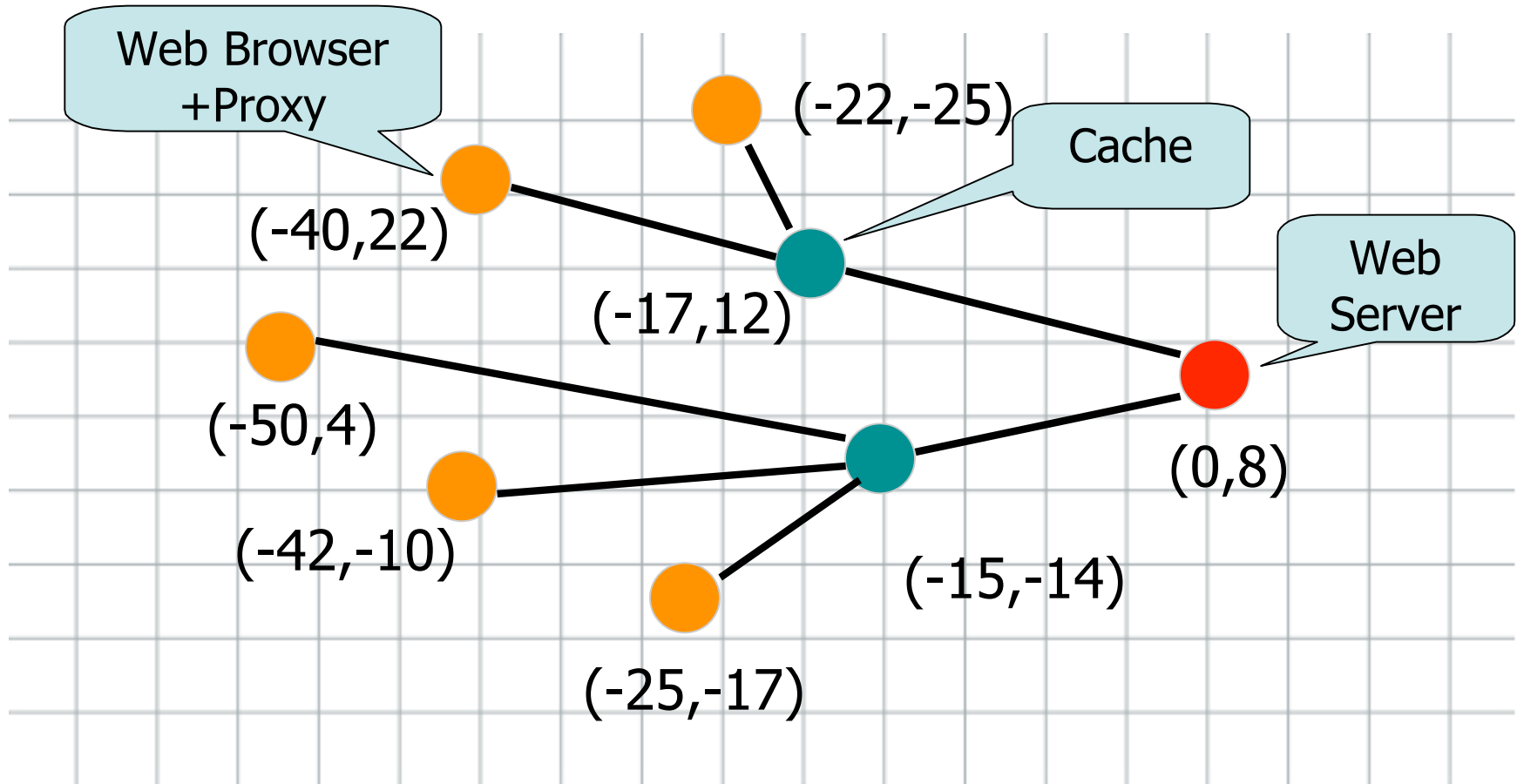


Overlay Networks Are Great



Locality-aware Web Cache

Route message toward web server's location



Routes converge at cache points

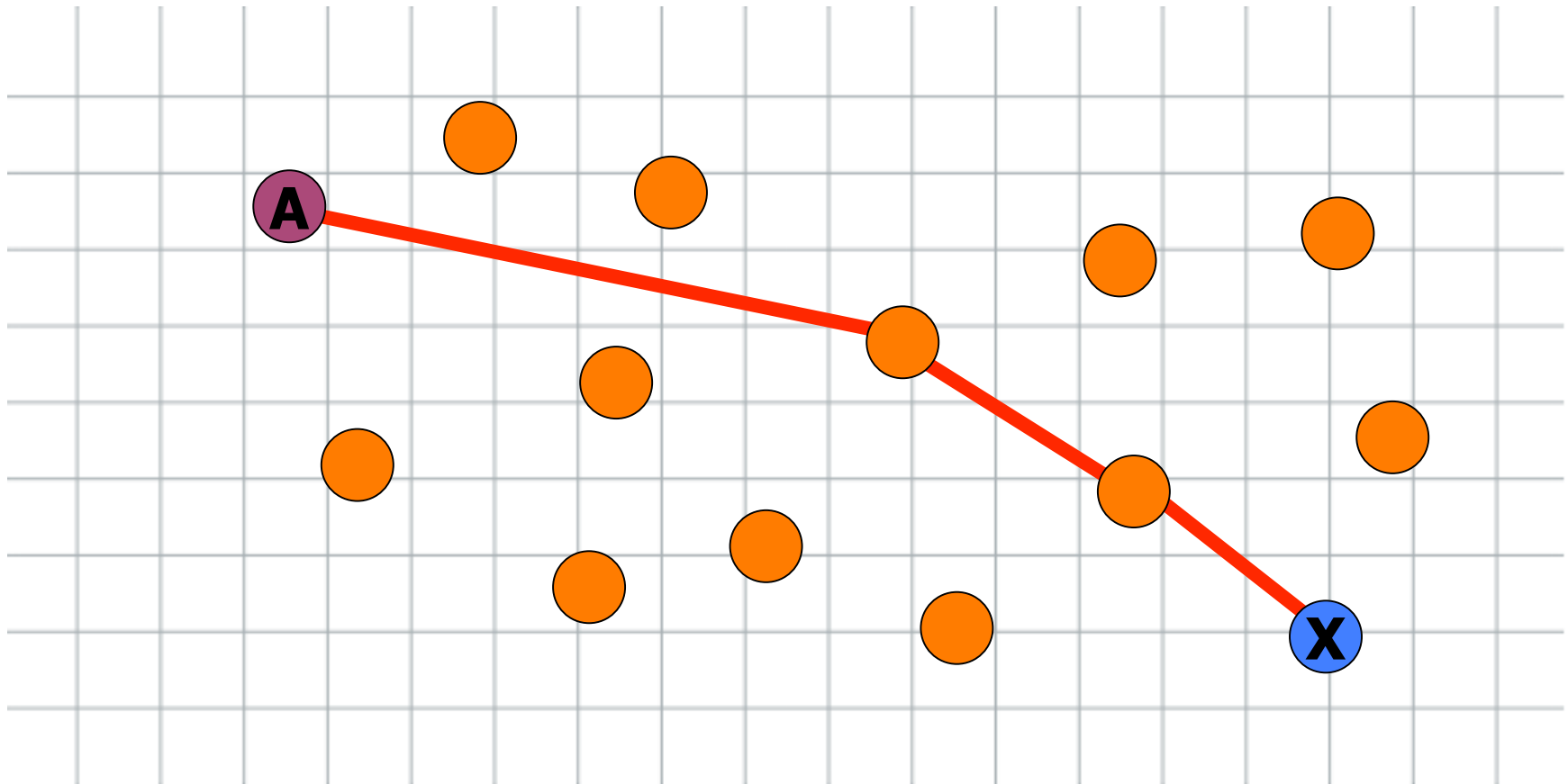


Wired Geometric Routing

- Given: each Internet host knows its location
 - Geographic location, e.g. longitude/latitude
 - **Network coordinate** using latency embedding
- Given: 2-d Euclidean multi-hop routing [Hassin '01]
 - Exploit “sense of direction”
 - Compared to wireless routing: no broadcast, long links OK
 - **Scaled-Theta Routing**: $\log(N)$ state and delay stretch
- Contributions
 - Practical implementation: k^d , RT/churn, closest node
 - Parameter tradeoffs / behavior on real network
 - Building block for e.g. locality-aware multicast, caching

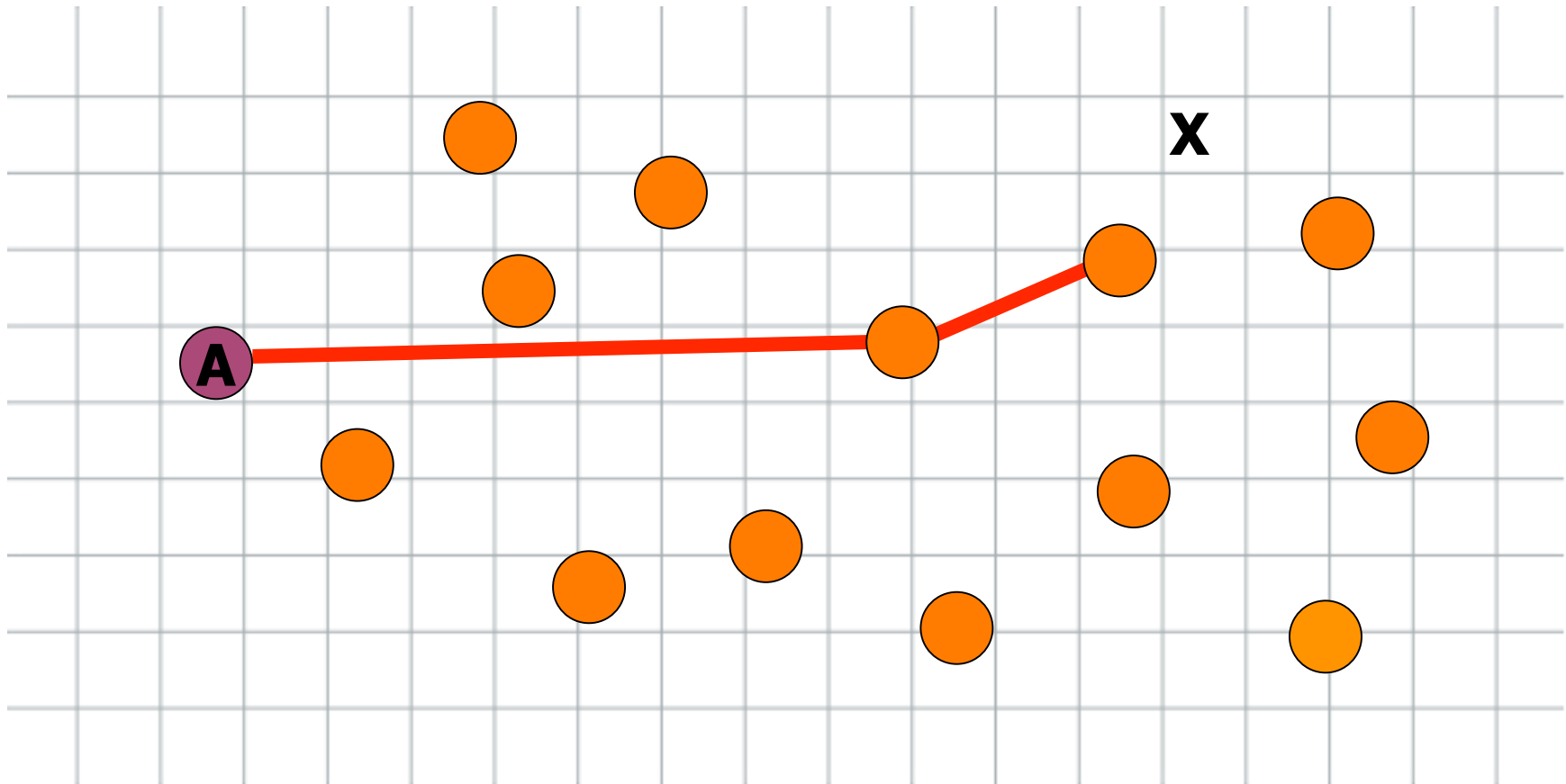


Geometric Routing: To Overlay Node



- Route message to overlay node location **X**
 - Analogous to `route(key, msg)` in DHTs
 - But routing path has low latency between **A** and **X**

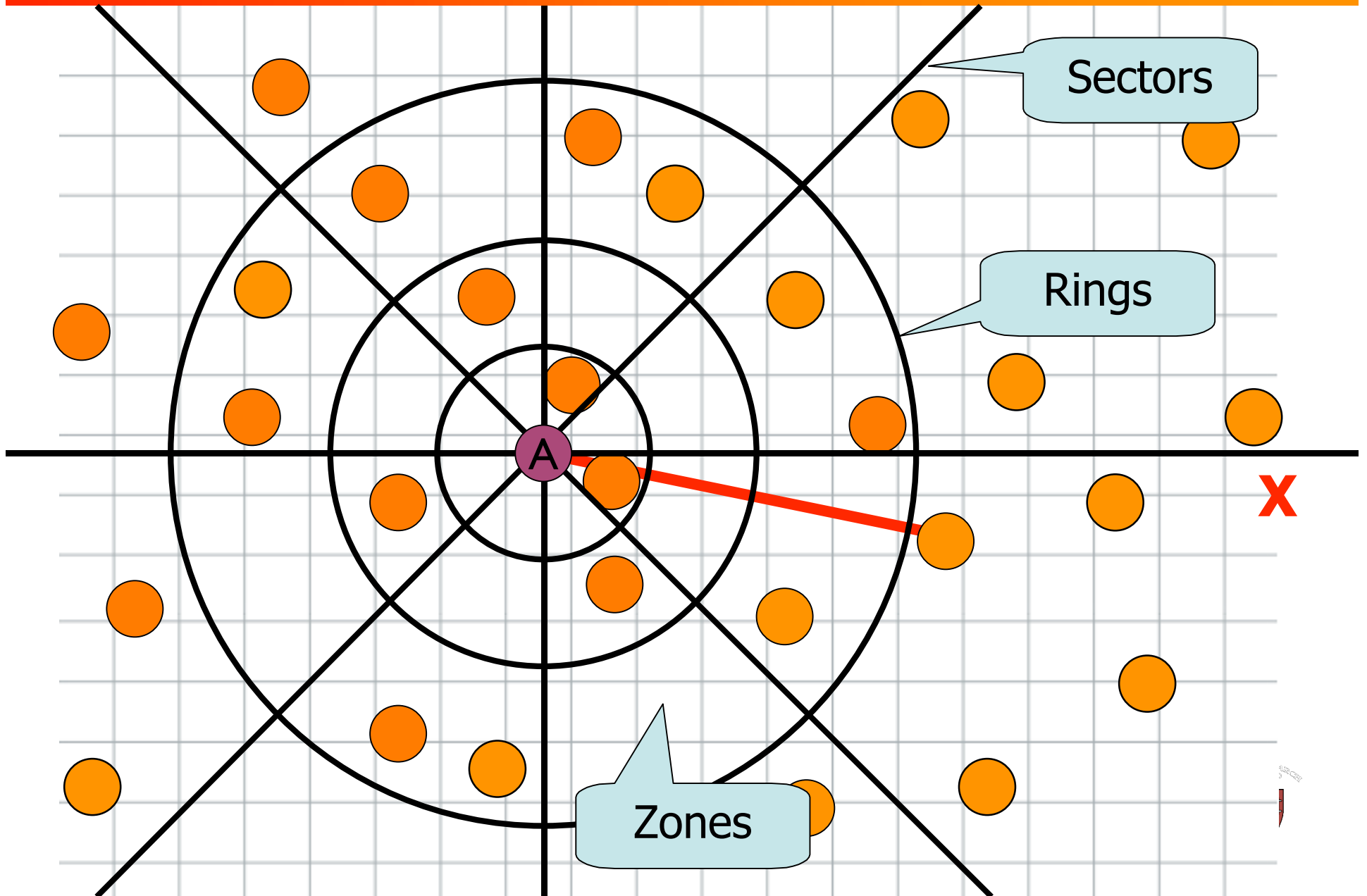
Geometric Routing: To Closest Node



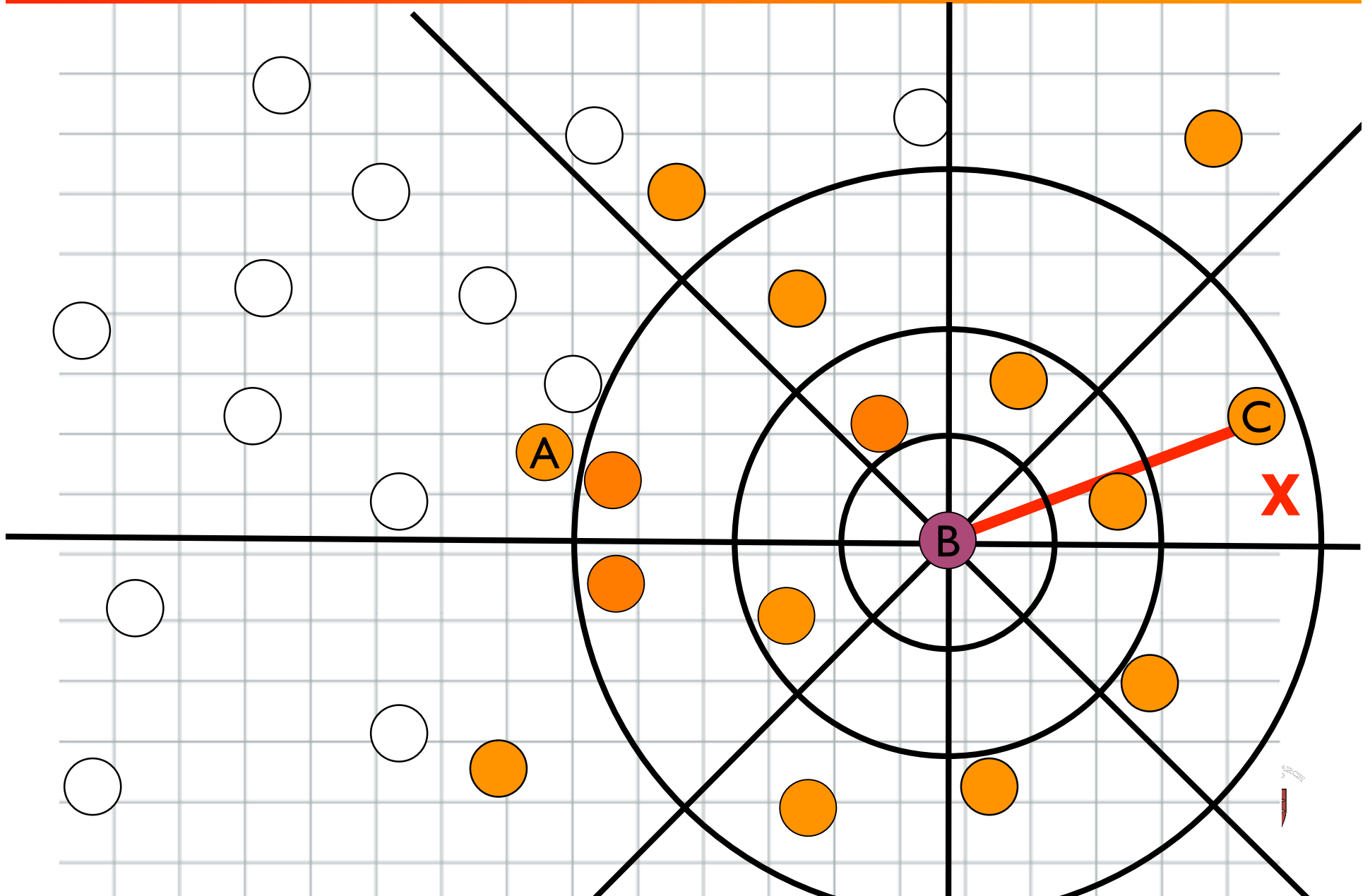
- Route message to closest existing overlay node
 - Useful when location is external to overlay network
 - e.g. finding closest web crawler to web server **X**



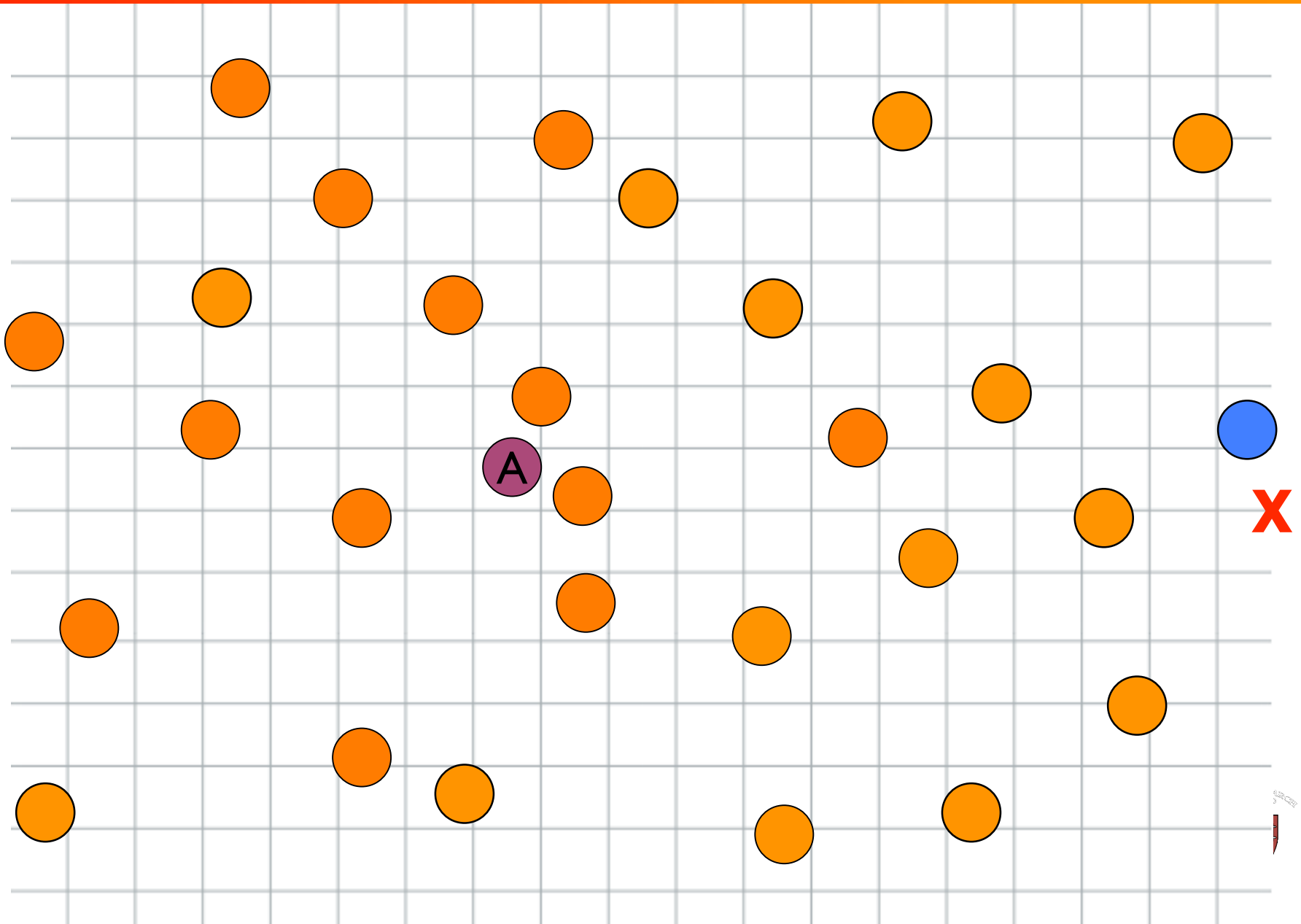
Scaled Theta Routing



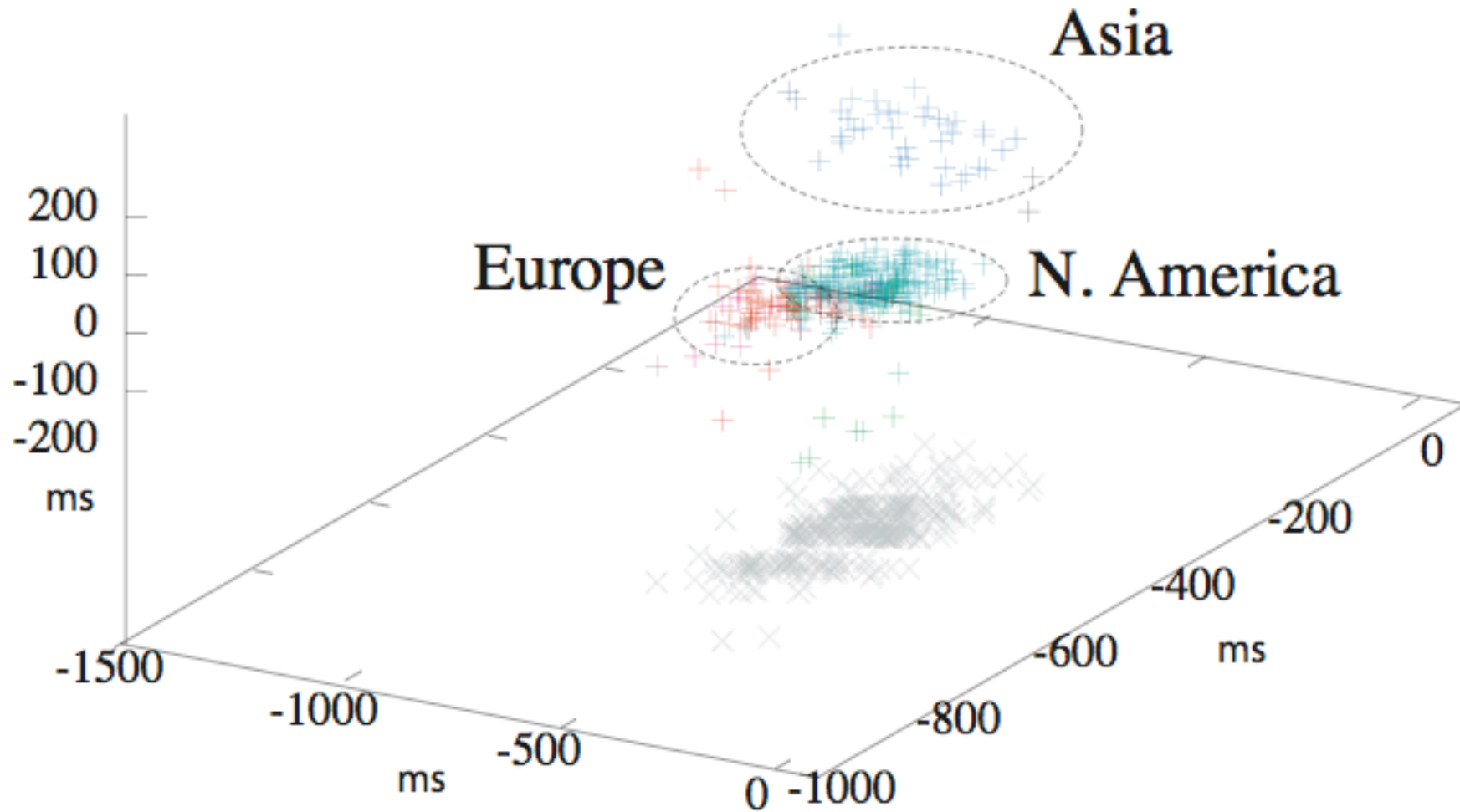
Scaled Theta Routing



Scaled Theta Routing



Network Coordinates on PlanetLab



- High embed accuracy w/few dimensions (3-5)
- Overlay node churn; maintenance



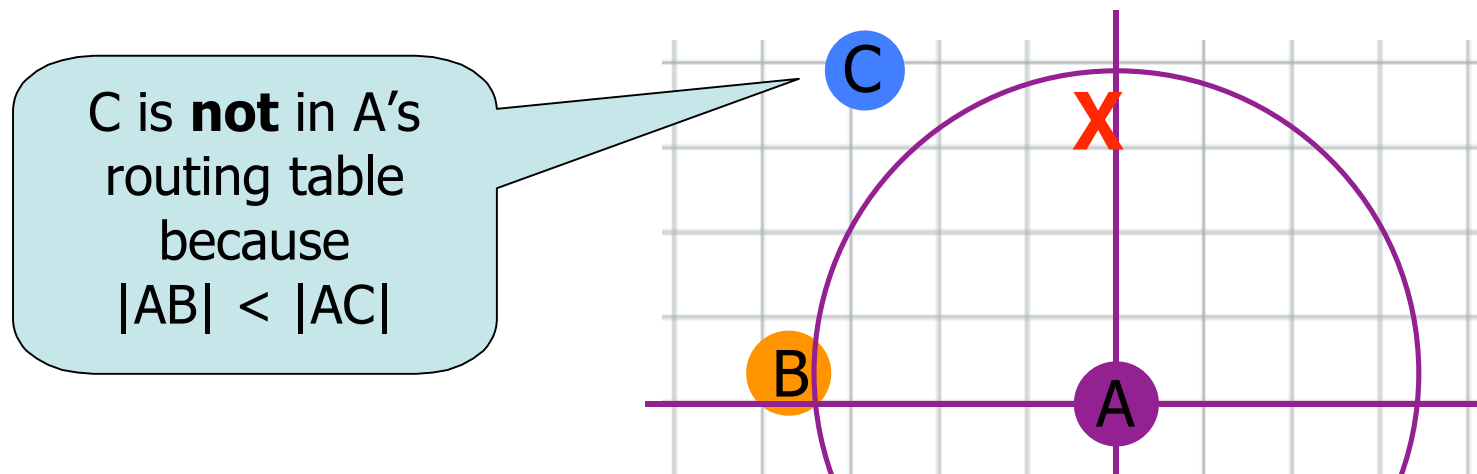
Practical Routing on Network Coordinates

- From theory to practice
 - ✓ Non-omniscient routing table formation
 - Inspired by Pastry join protocol
 - ✓ Generalized k^d zone assignment
 - Hyperspherical coordinates: $d-1$ angles
 - x Local minima in closest node lookups
- Evaluation
 - Parameter trade-offs: effect sectors and rings
 - Closest node on real network
 - Closest node with churn



Closest Node Lookups

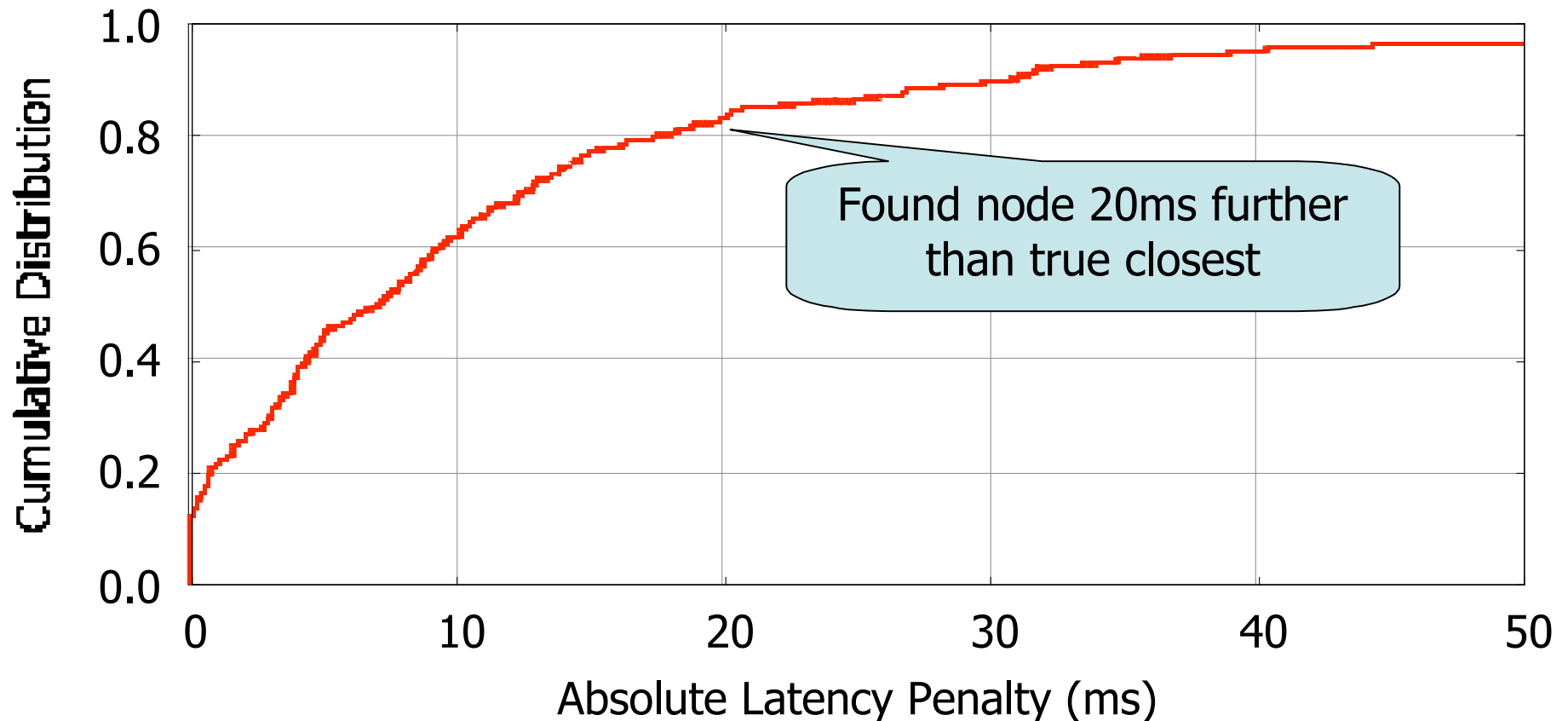
- How does a node know it's the closest node?
 - Closest node lookups mean **holes** in routing tables
 - **Greedy** strategy doesn't work anymore



- Reduce probability of this by increasing number of sectors
 - At most one node off in practice
 - Use sensor net approach?



Evaluation: Closest Node



Closest coordinate vs. true closest node

- 4d+h embed of MIT King data set (1740 DNS Servers)
- Designate 10% as targets
- Assigned "perfect" routing tables (rings=8; base=4; sectors=6)
- Found closest coordinate (1/10000); thus: embed error dominates



Conclusions + Open Questions

- First cut at efficient routing on NC substrate
 - Fundamental building block like DHTs
 - Geometric meaning for solving network problems
- Open Questions
 - Geographic coordinates?
 - Behaviours/optimizations different NC sets / graphs?
 - Quantify/analyze path convergence?
 - Fleshing out applications; not Meridian/OASIS competitor
 - Unify routing between wireless and wired networks??



Thanks & Pyxida



Check out Pyxida:
A Network Coordinate Service
for PlanetLab

<http://pyxida.sourceforge.net>

Thanks. Any Questions?

Jonathan Ledlie
jonathan@eecs.harvard.edu

