Advanced Computer Networks

Overlay Networks

Jianping Pan
Spring 2017

Paper reading summaries

- Starting from this week
  - reading list is on xc (will be further updated)
  - summary submission schedule to be posted soon
  - 1-page reading summary on assigned papers
  - what's the problem? important then? and now?
  - main ideas? previous work? follow-on work?
  - strengths then? and now? why?
  - weaknesses then? and now? why?
  - how to improve? how to do better or differently?

* reading guideline at: http://www.cs.uvic.ca/~pan/csc466/reading.txt (template to be posted soon)

What do we “have” so far?

- Internet design and architecture
  - store-and-forward packet switching
  - end-to-end arguments
    - smart end-hosts vs dumb networks
  - best-effort services, client-server applications
- Initially, the Internet was an “overlay”
  - over telephone networks
- By design, the Internet is a “peer-to-peer”
  - for all end-hosts

Reality check

- A network of service-provider's networks
  - still mostly packet switching, end-to-end, best-effort
- But hierarchical structures almost everywhere
  - tiered service provider networks
  - hierarchies in naming, addressing, routing, service provisioning, content delivery etc
  - the (only) way to deal with scalability
- Two sides of the story
  - a lot of details/redundancy invisible to externals

Examples

- Internet routing
  - routing pathologies
    - a considerable percentage of routes is affected
  - delayed convergence
    - after a fault, it takes tens of minutes to converge
  - extended recovery
    - some faults take hours to recover
- Dependable Internet?
  - not yet

Adding ??? into the network?

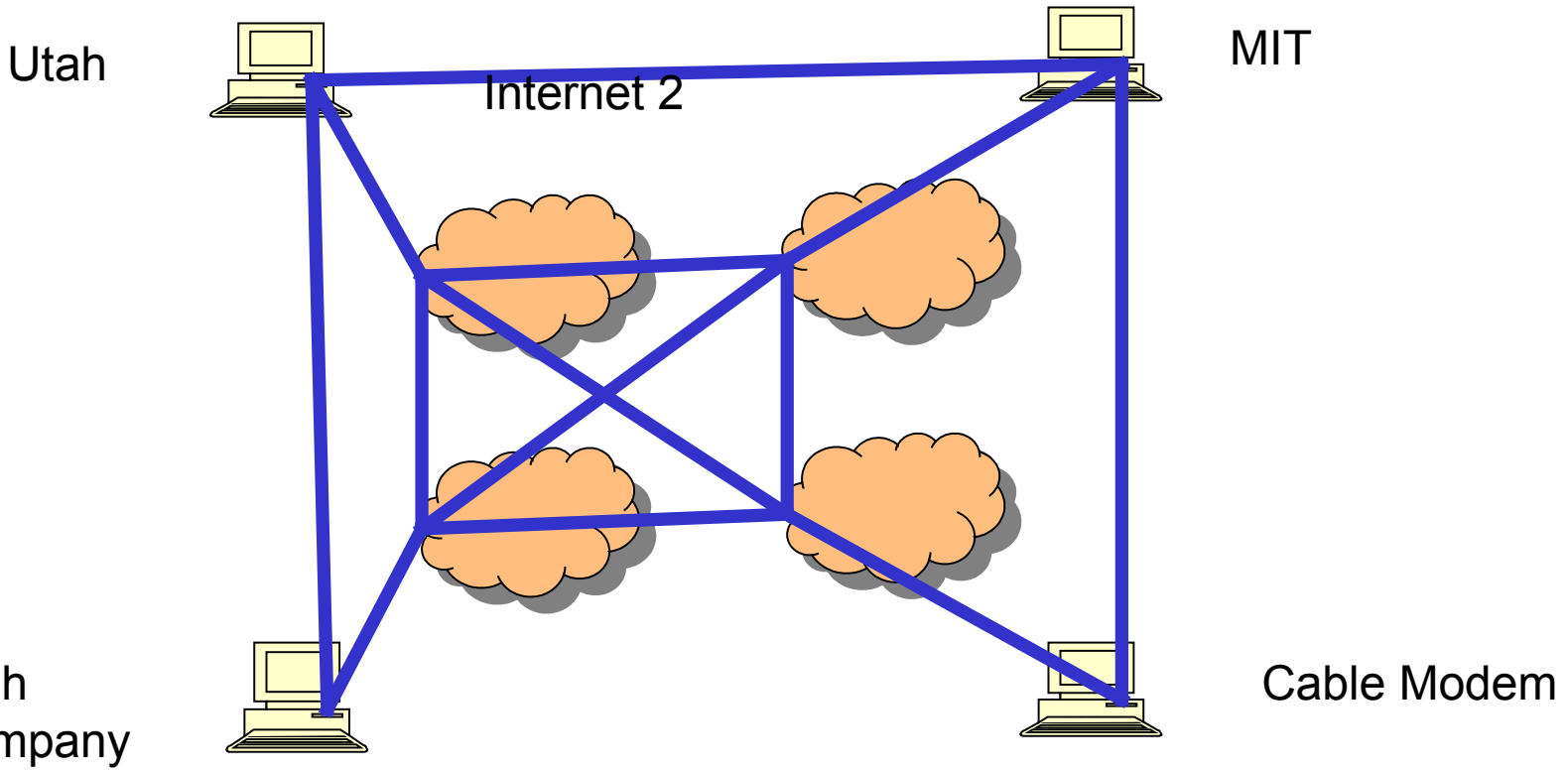
- Changing the infrastructure is difficult
  - in a competitive ISP market
  - only end-to-end counts
  - and not all applications need perfect ???
- Alternatives
  - application overlays
    - e.g., virtual private networks (VPN)
    - content delivery networks (CDN)
  - end-to-end or edge-to-edge

Resilient Overlay Networks

- http://nms.lcs.mit.edu/ron
  - [ABKM01] D. Anderson, H. Balakrishnan, F. Kaashoek, R. Morris, Resilient Overlay Networks, In Proc. of SOSP '01. [RON]
- Design goals
  - fast failure detection and recovery
    - active probing, re-routing
  - tighter integration with applications
    - application-specific, e.g., video conferencing
  - expressive policy routing
    - e.g., “no commercial traffic on Internet2”

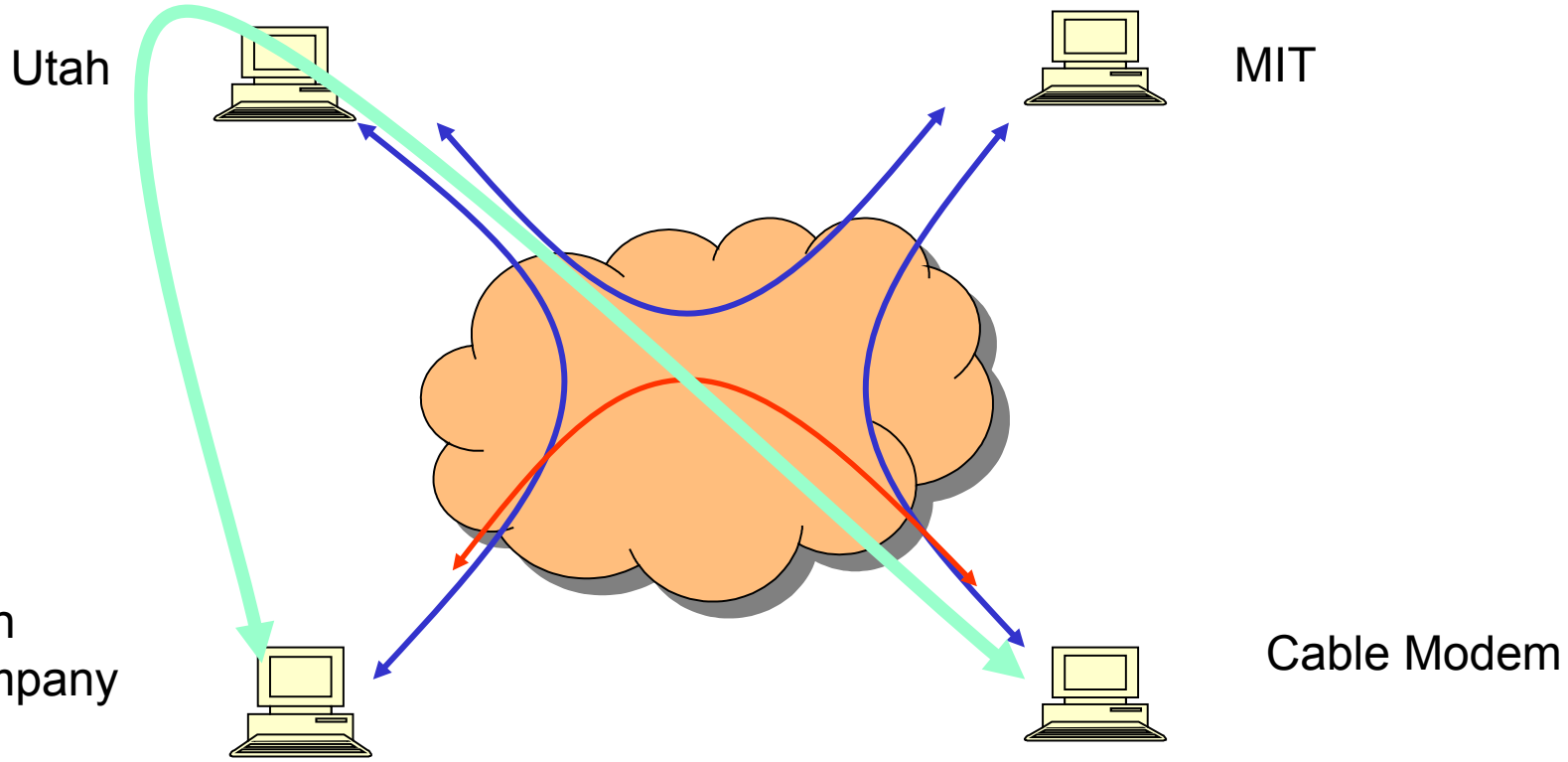
Observations

- Network redundancy, invisible to applications



Ideas

- Route around failures



Approaches

- Characterize “links” between nodes
  - active probing: delay, loss
- Disseminate link characteristics
  - “link-state” advertisement
- Choose the “best” route
  - only at the entry node
  - with possibly one intermediate node
- Forward the packets
  - RON encapsulation

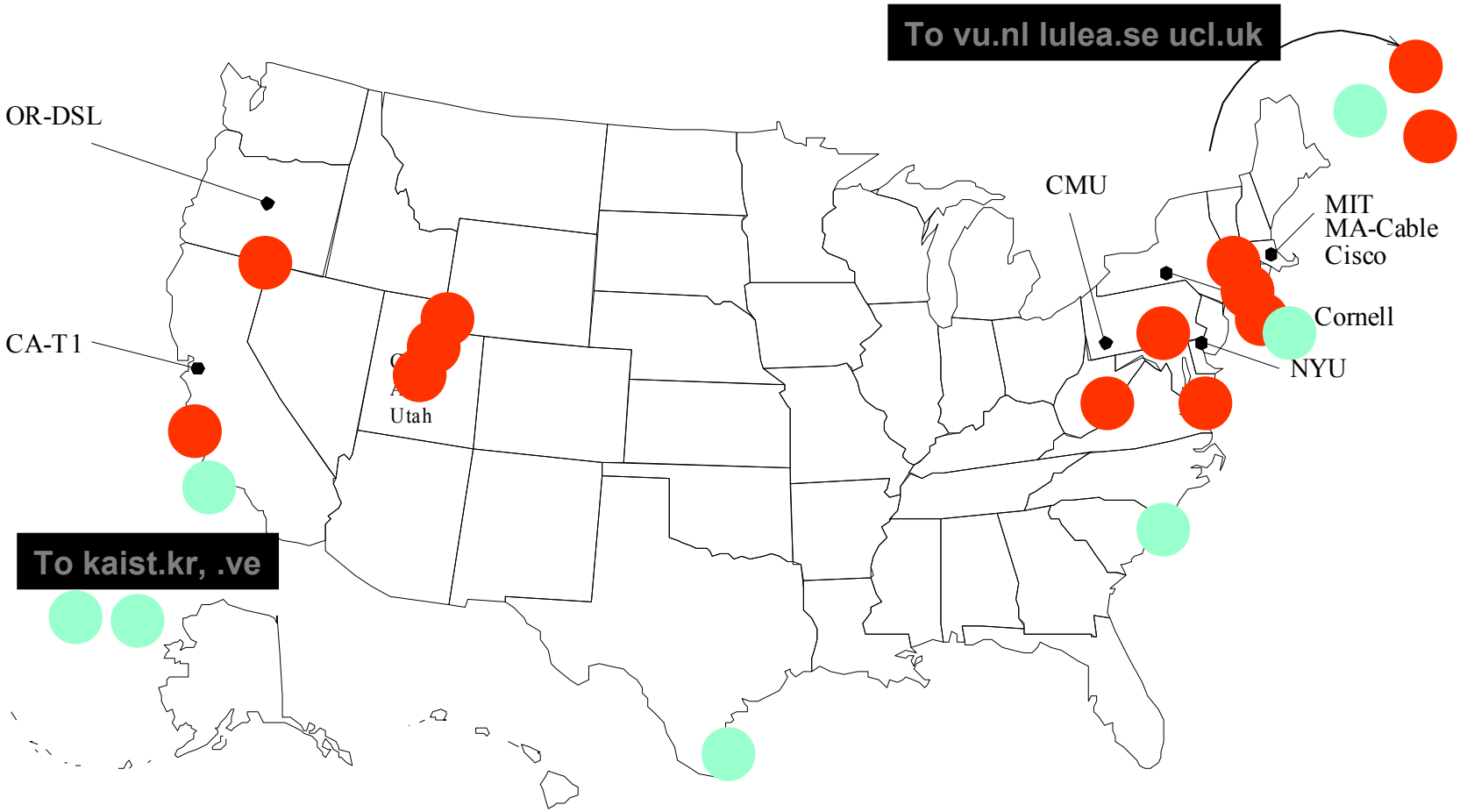
Design details

- Path selection
  - Ĥ delay
    - exponentially weighted moving average (EWMA)
    - $delay_{i+1} = a * delay_i + (1-a) * last\_rtt$, a = 0.9
  - Ĥ loss: moving window average
    - window size: 100
  - Ĥ throughput
    - TCP-like, proportional to 1/(rtt * sqrt(p))
  - Ĥ application-specific
    - priority among delay, loss, throughput, etc

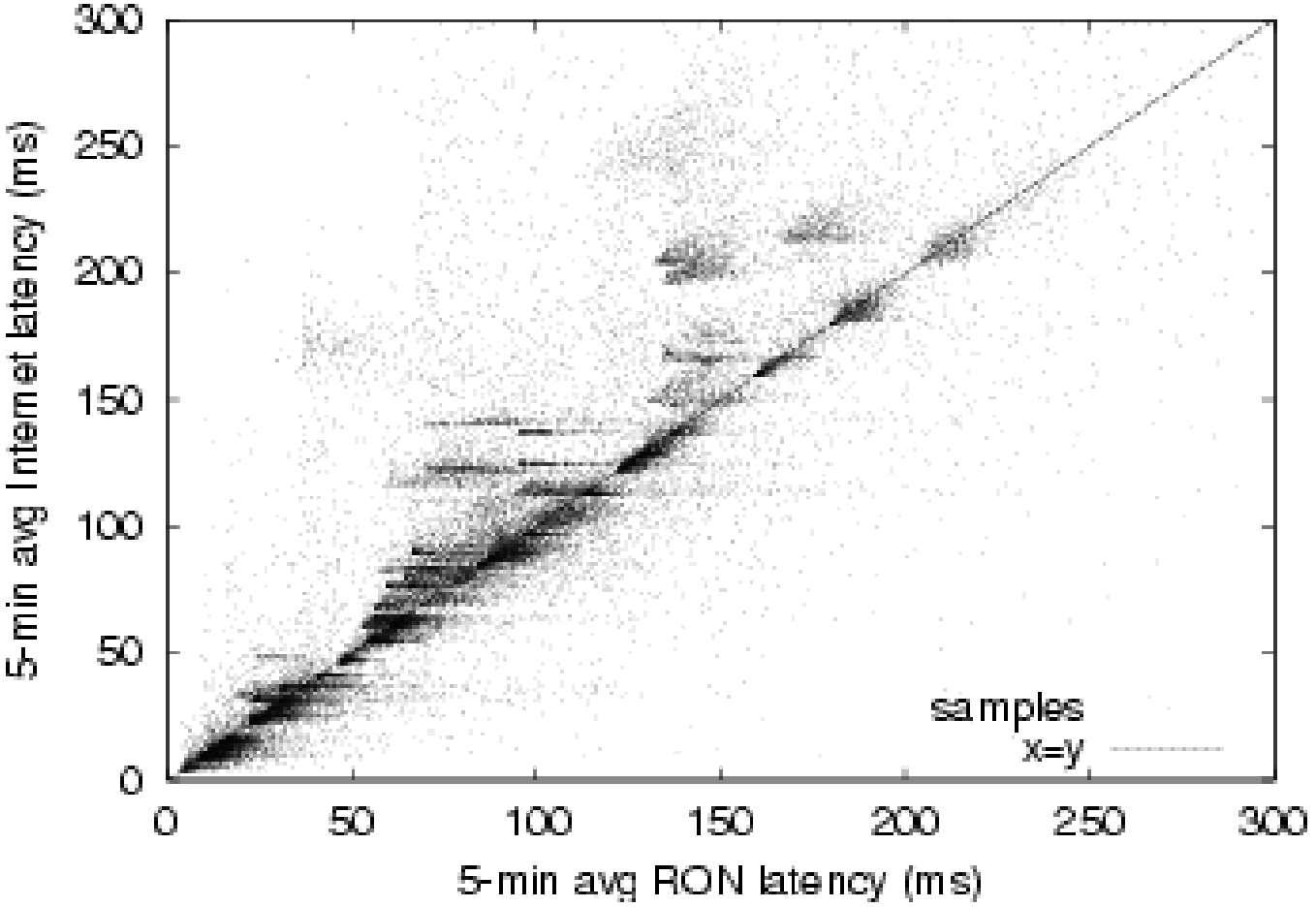
Membership management

- Static membership
  - load other peer nodes from a configuration file
- Announcement-based, soft-state membership
  - know at least one peer node
  - announce its existence by broadcast
  - soft-state
    - flood peer node list every 5 minutes
    - if a node is not heard for 60 minutes, the node has left
- Search?

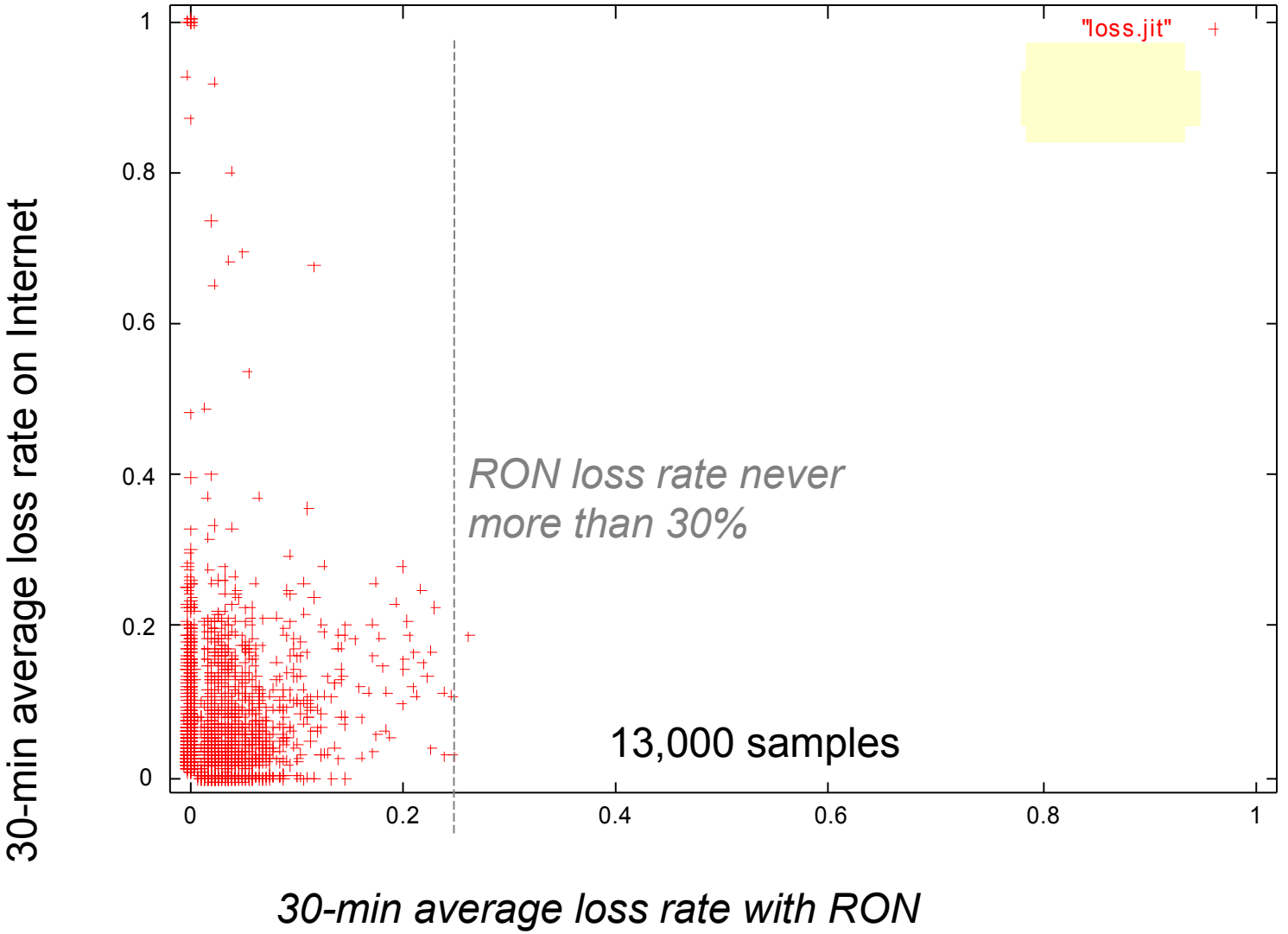
Performance evaluation



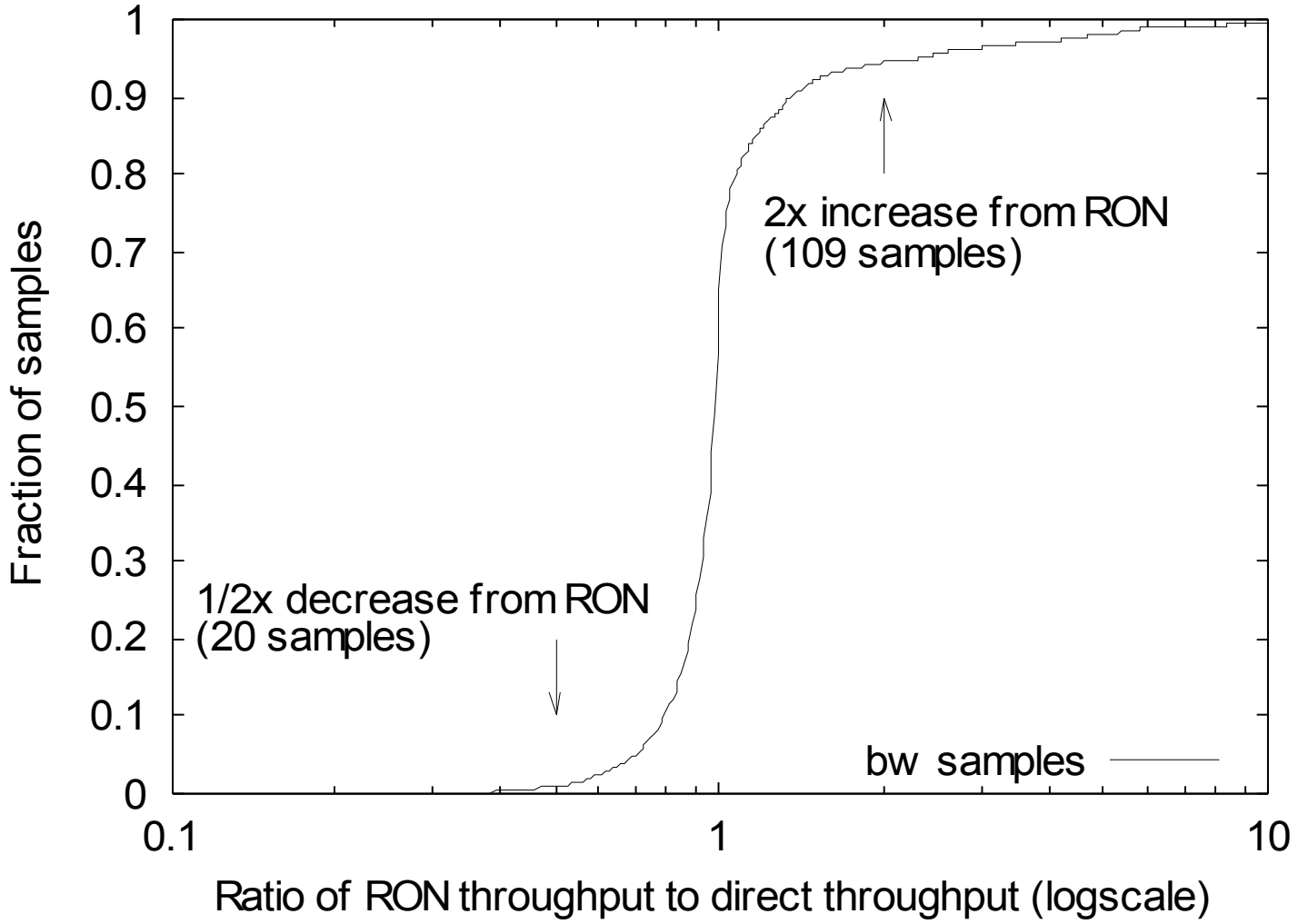
Reduced delay



Reduced loss



Improved throughput



Overhead

- Link probing
  - Ĥ size: 69 bytes; interval: 12 seconds
- Link advertisement
  - Ĥ size: 60+20*(N-1); interval: 14 seconds
- Recovery time: 12~25 seconds (N=50)

10 nodes	20 nodes	30 nodes	40 nodes	50 nodes
1.8 Kbps	5.9 Kbps	12 Kbps	21 Kbps	32 Kbps

More discussion

- One hop?
- Route stability
  - hysteresis
- Path selection
  - tradeoff between delay, loss, etc
- Routing policy
- Scalability
- NAT (network address translator)

More overlay networks

- Planet-lab network testbed and GENI
- Peer-to-peer applications
  - Napster: with centralized directory server
  - Gnutella: distributed flooding search (ERS)
  - KaZaA: hierarchy introduced; supernode
  - BitTorrent: trackers; files in chunks; tit-for-tat
  - Skype
  - Structured P2P
    - Distributed Hash Table (HDT): Chord, CAN, Pastry, etc

Next lectures

- DHT
  - Required reading
    - Chord
    - [RFHKS01] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A scalable content-addressable network. In SIGCOMM," Aug. 2001. [CAN]
    - [RD01] Rowstron and P. Druschel, "Pastry: Scalable, distributed object location and routing for largescale peer-to-peer systems," Proc. 18th IFIP/ACM Int'l. Conf. Distributed Systems Platforms (Middleware), 2001. [Pastry]
- Gnutella, BitTorrent, Skype