Mitigating Congestion in Wireless Sensor Networks

Bret Hull, Kyle Jamieson, Hari Balakrishnan
Networks and Mobile Systems Group
MIT Computer Science and Artificial Intelligence Laboratory

Congestion is a problem in wireless networks

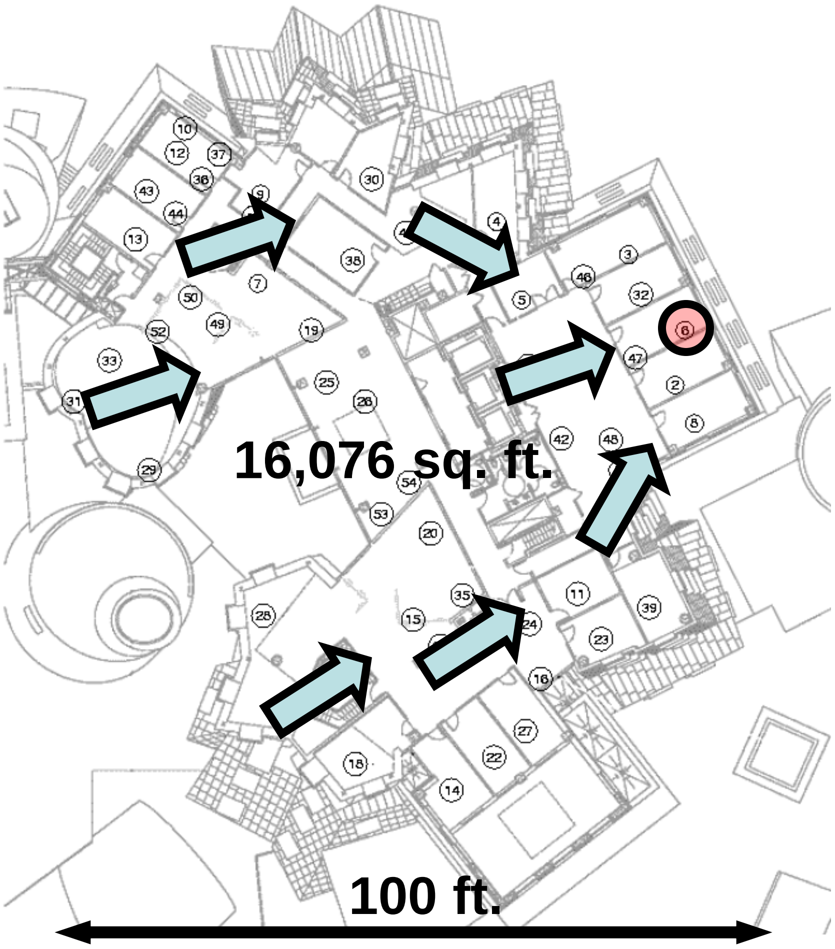
- **Difficult to provision bandwidth in wireless networks**
  - Unpredictable, time-varying channel
  - Network size, density variable
  - Diverse traffic patterns
- **The result is congestion collapse**

Outline

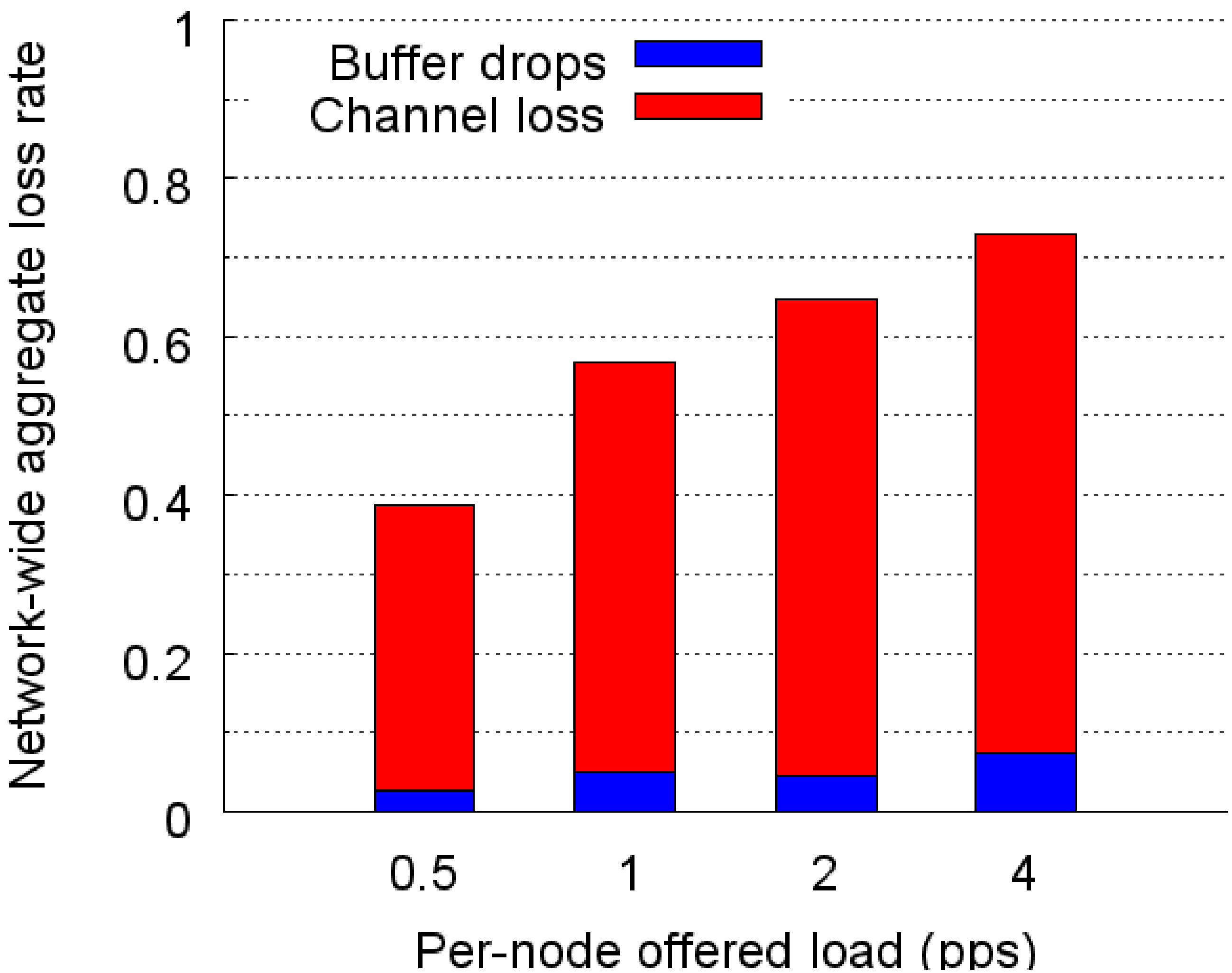
- **Quantify the problem in a sensor network testbed**
- **Examine techniques to detect and react to congestion**
- **Evaluate the techniques**
  - Individually and in concert
  - Explain which ones work and why

Investigating congestion

- 55-node Mica2 sensor network
- Multiple hops
- Traffic pattern
  - All nodes route to one sink
- B-MAC [Polastre], a CSMA MAC layer

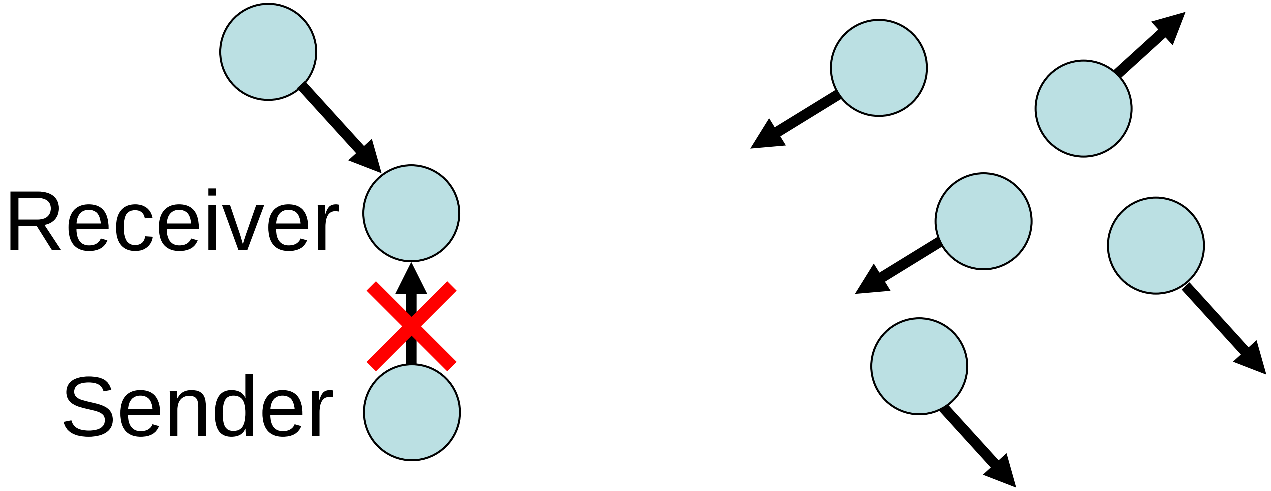


Congestion dramatically degrades channel quality

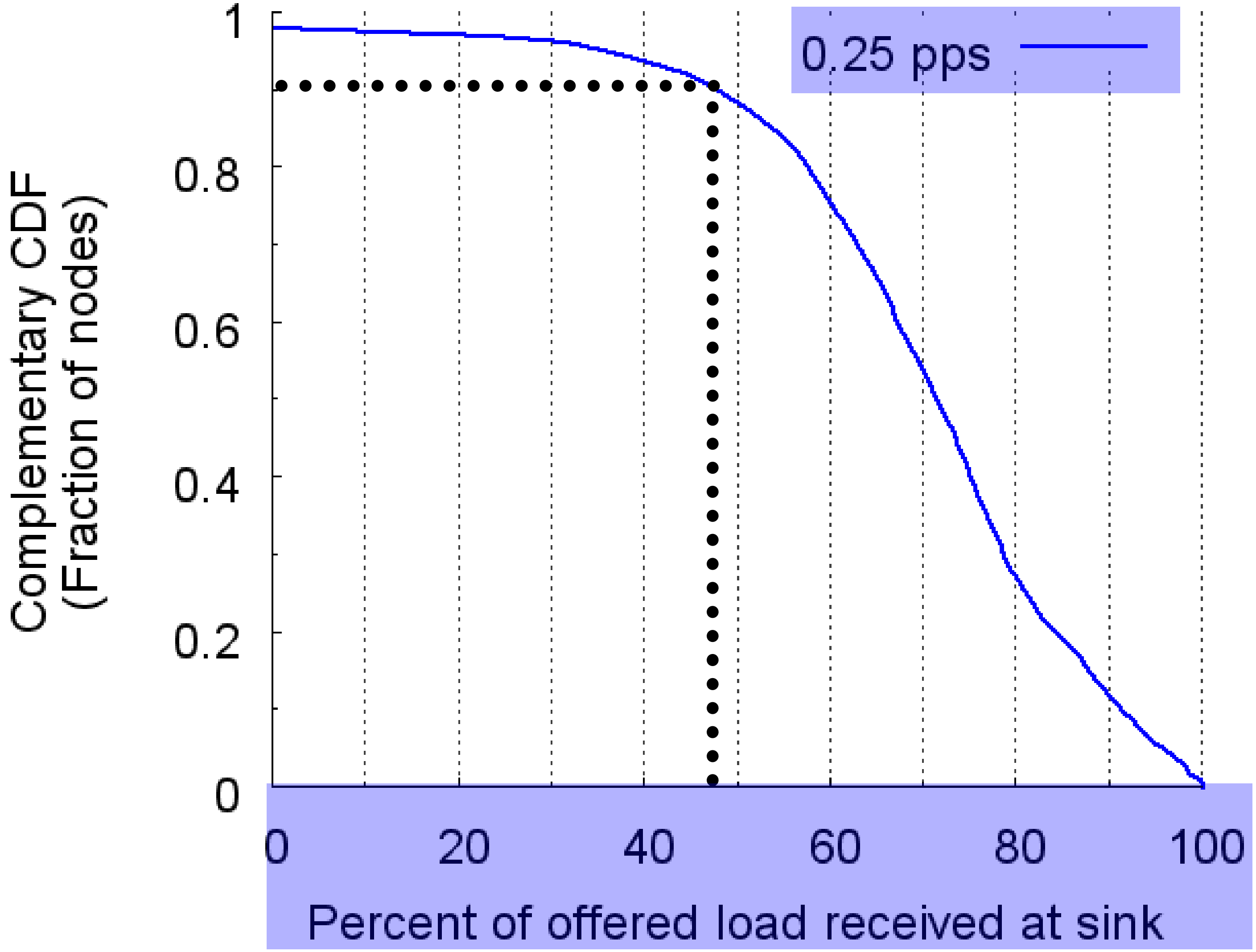


Why does channel quality degrade?

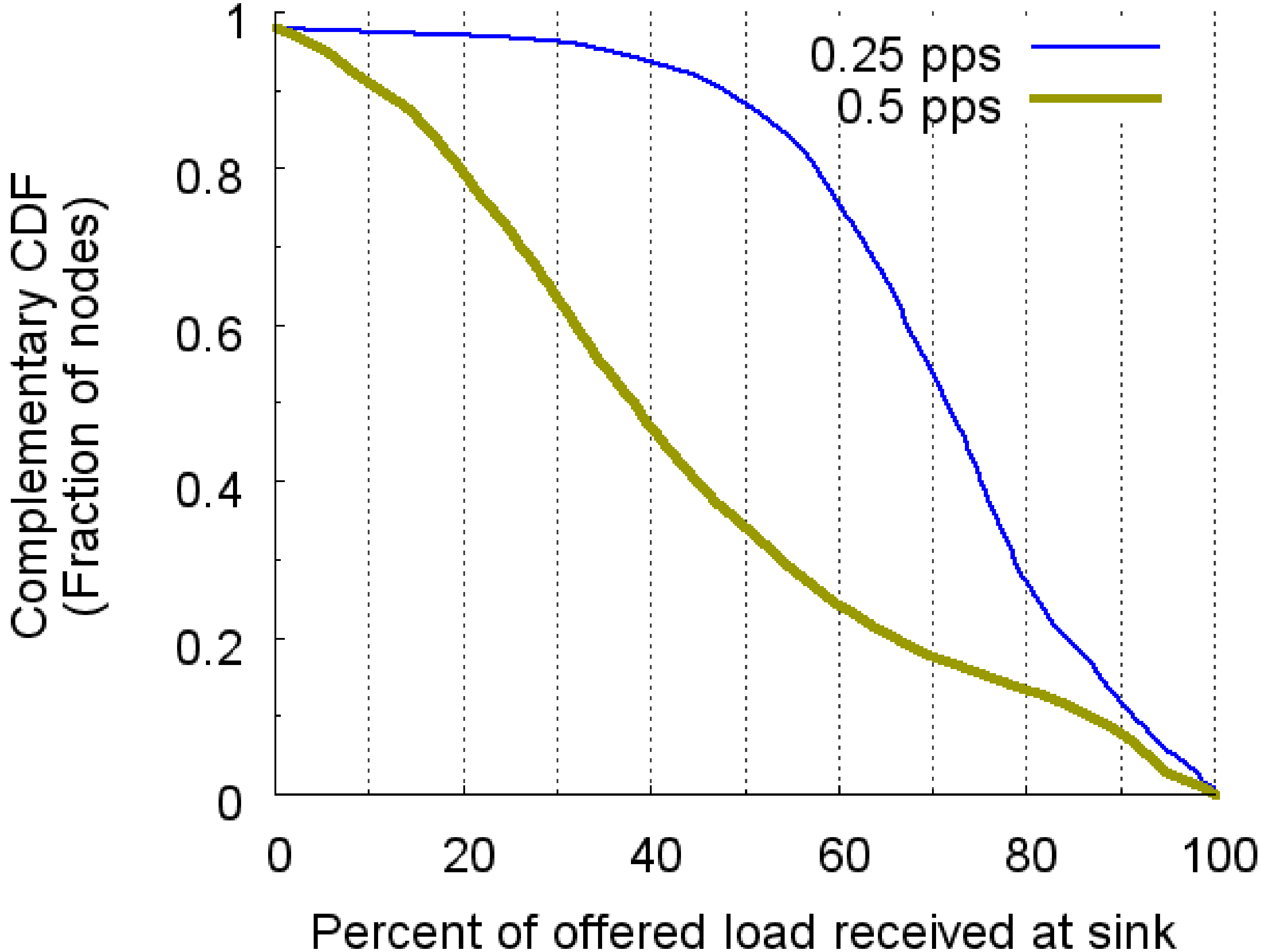
- **Wireless is a shared medium**
  - Hidden terminal collisions
  - Many far-away transmissions corrupt packets



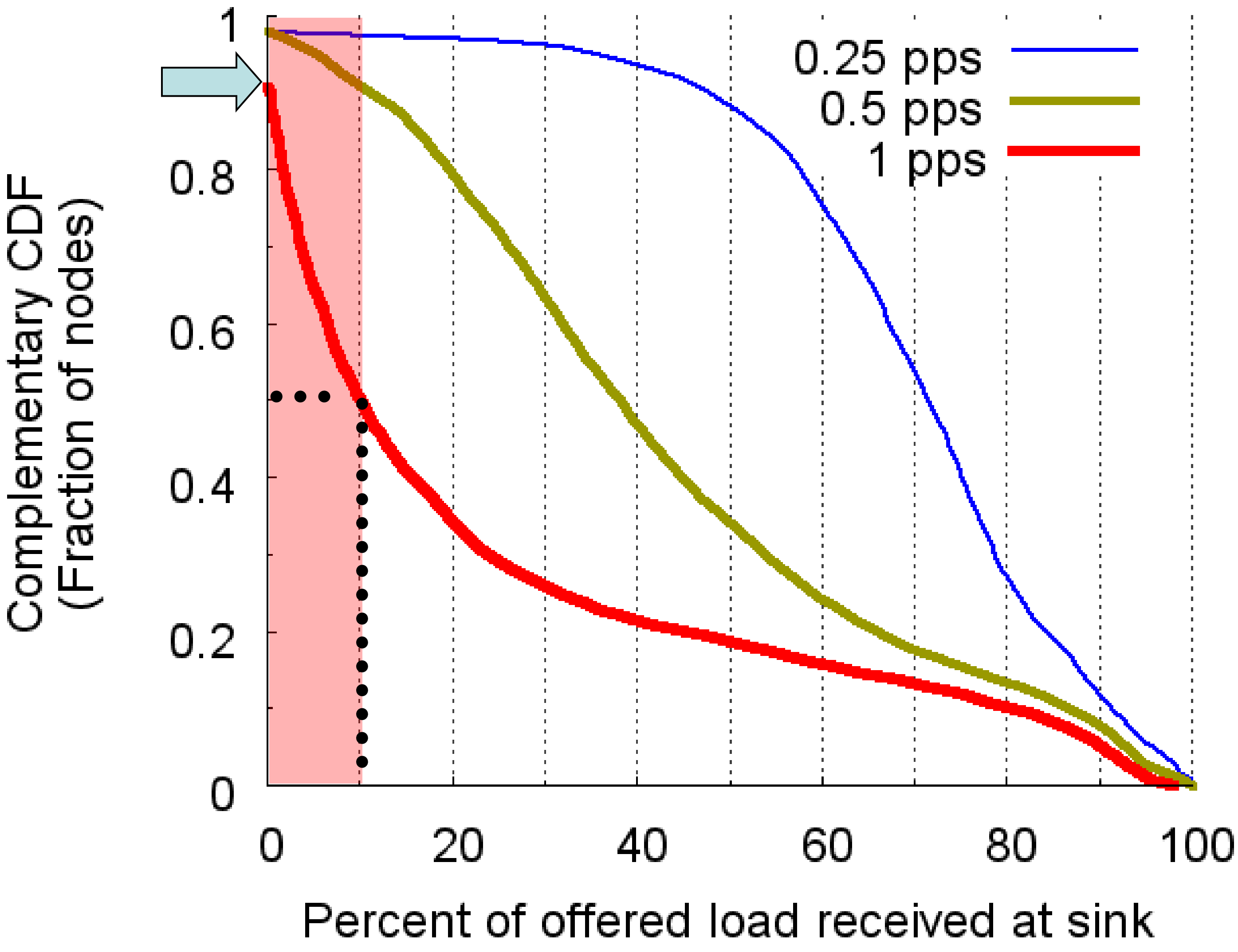
Per-node throughput distribution



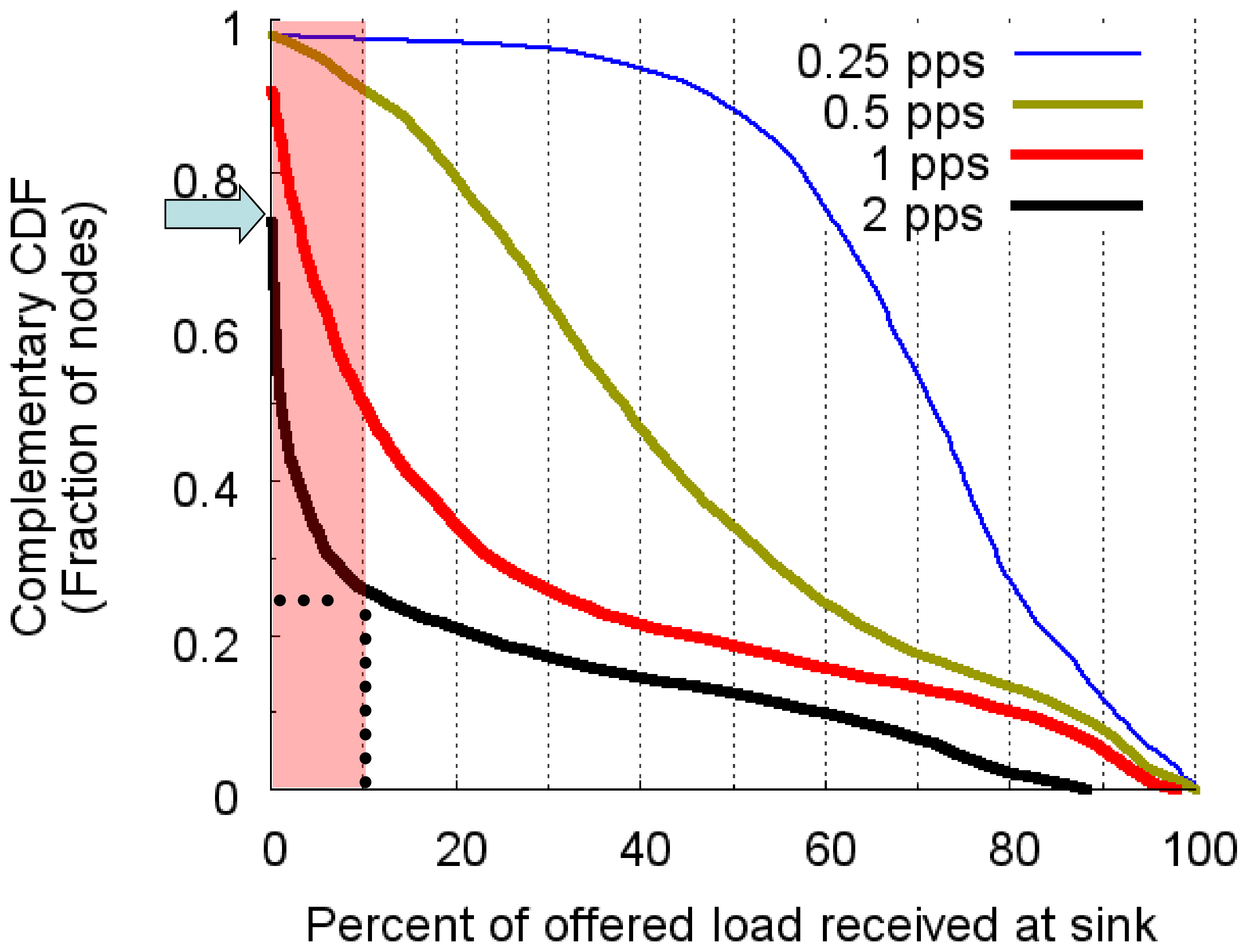
Per-node throughput distribution



Per-node throughput distribution



Per-node throughput distribution

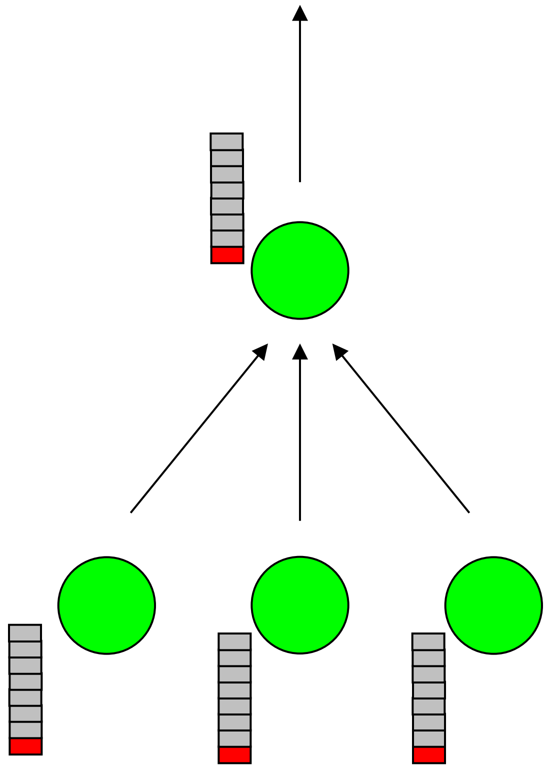


Goals of congestion control

- **Increase network efficiency**
  - Reduce energy consumption
  - Improve channel quality
- **Avoid starvation**
  - Improve the per-node end-to-end throughput distribution

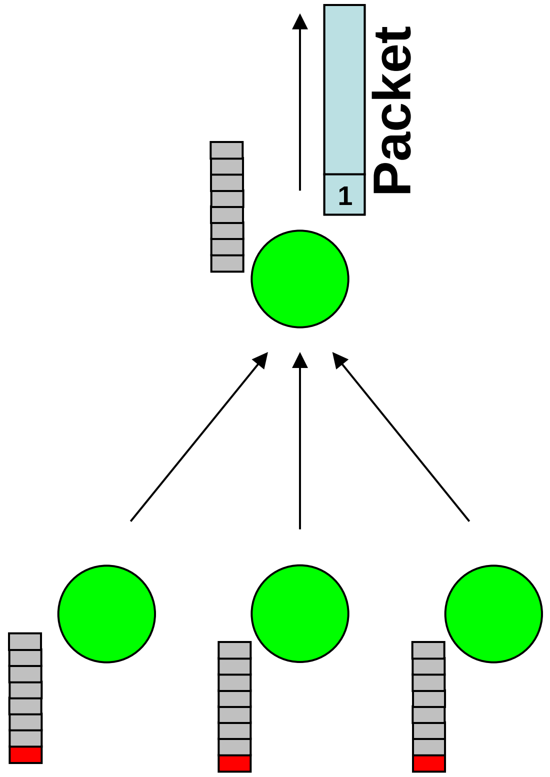
Hop-by-hop flow control

- **Queue occupancy-based congestion detection**
  - Each node has an output packet queue
  - Monitor instantaneous output queue occupancy
  - If queue occupancy exceeds α, indicate *local congestion*



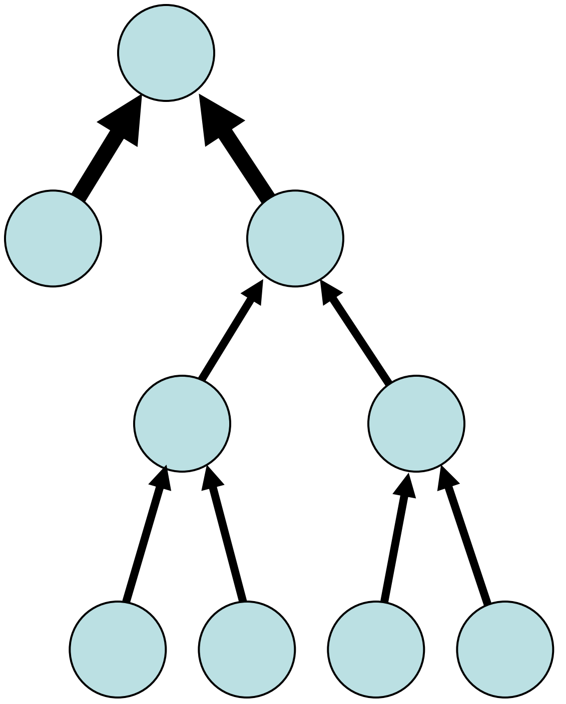
Hop-by-hop flow control

- **Hop-by-hop backpressure**
  - Every packet header has a congestion bit
  - If locally congested, set congestion bit
  - Snoop downstream traffic of parent
- **Congestion-aware MAC**
  - Priority to congested nodes



Rate limiting

- **Source rate limiting**
  - Count your **parent's number of descendents**
  - Limit your sourced traffic rate, **even if hop-by-hop flow control is not exerting backpressure**



Related work

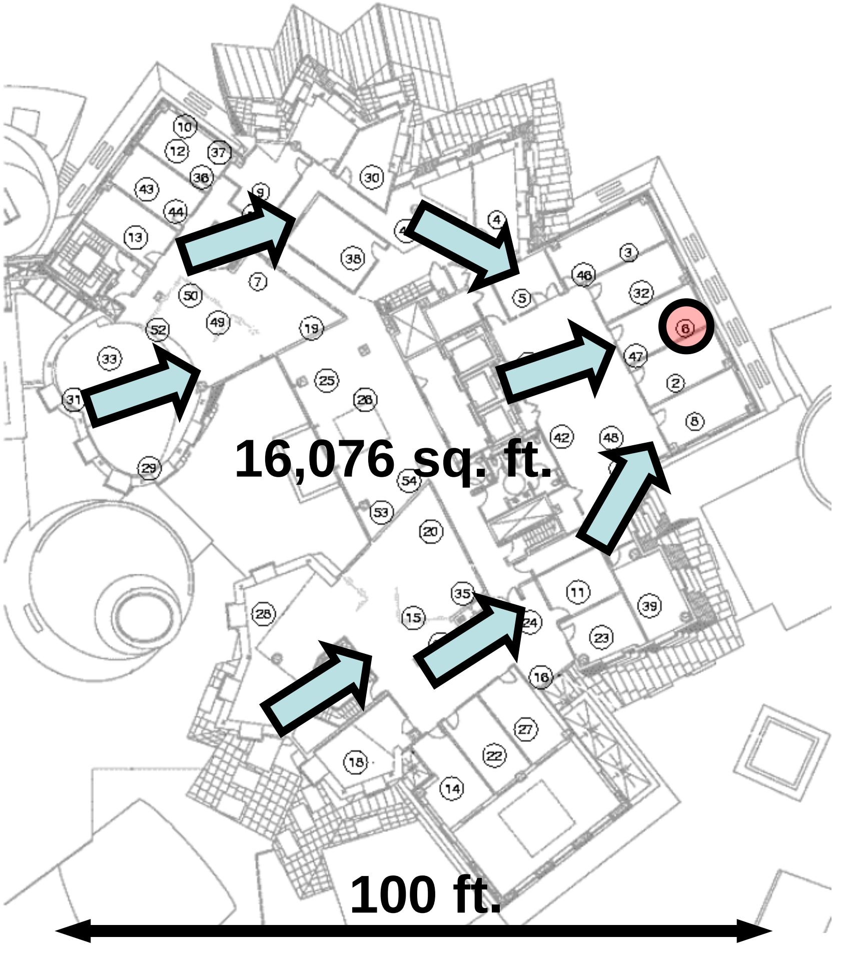
- **Hop-by-hop flow control**
  - Wan *et al*., SenSys 2003
  - ATM, switched Ethernet networks
- **Rate limiting**
  - Ee and Bajcsy, SenSys 2004
  - Wan *et al*., SenSys 2003
  - Woo and Culler, MobiCom 2001
- **Prioritized MAC**
  - Aad and Castelluccia, INFOCOM 2001

Congestion control strategies

No congestion control	Nodes send at will
Occupancy-based hop-by-hop flow control	Detects congestion with queue length and exerts hop-by-hop backpressure
Source rate limiting	Limits rate of sourced traffic at each node
Fusion	**Combines occupancy-based hop-by-hop flow control with source rate limiting**

Evaluation setup

- Periodic workload
- Three link-level retransmits
- All nodes route to one sink using ETX
- Average five hops to sink
- –10 dBM transmit power
- 10 neighbors average

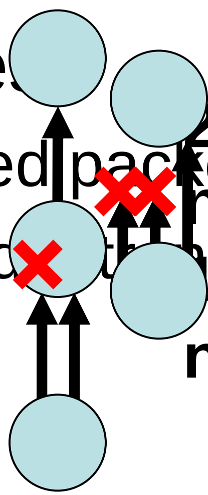


Metric: network efficiency

$$\eta = \frac{\sum_{p \in Received} \mathbf{hops}(p)}{\mathbf{total\ transmit\ count}}$$

Interpretation: the fraction of transmissions that contribute to data delivery.

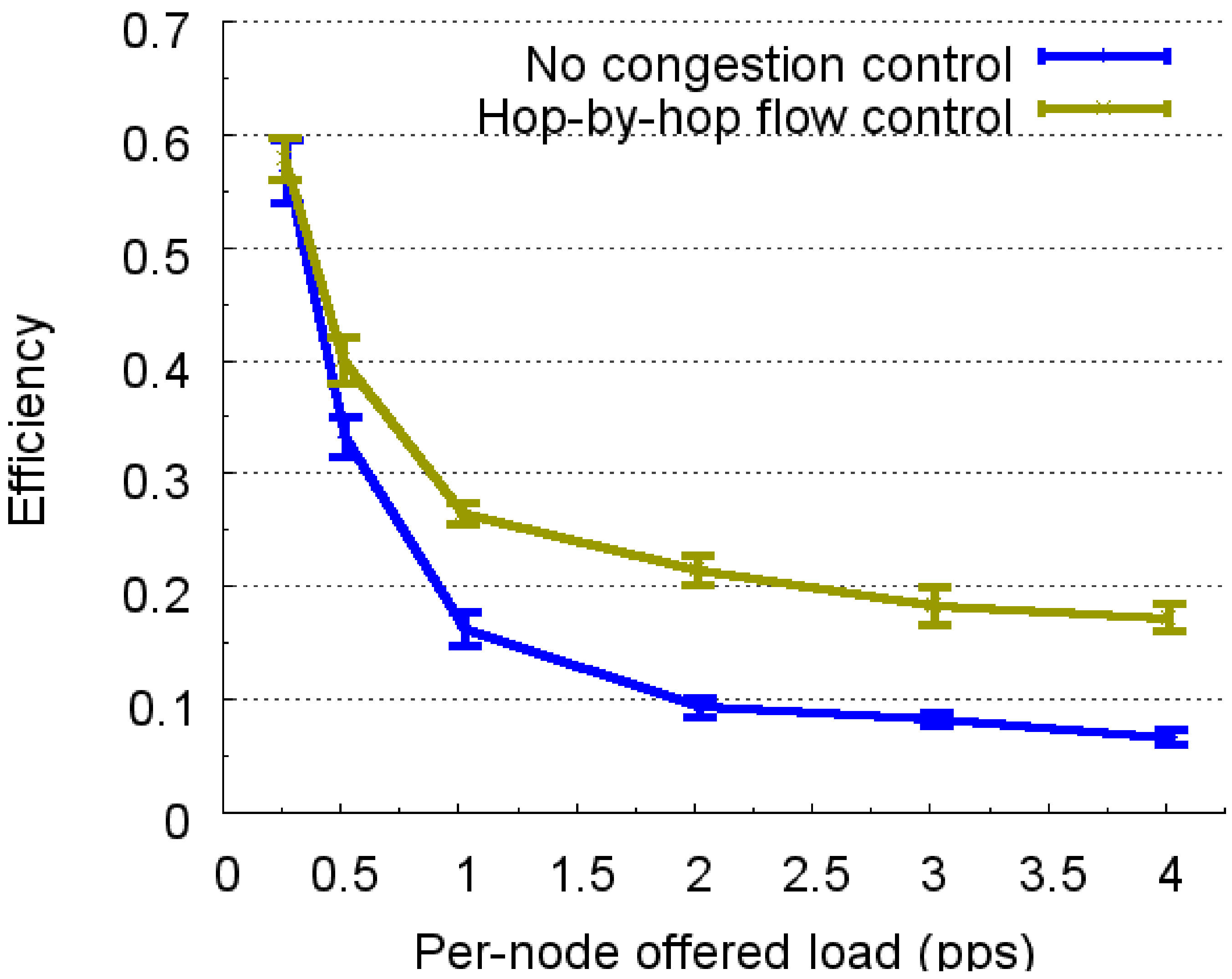
- Penalizes
  - Dropped packets (buffer drops, channel losses)
  - Wasted transmissions



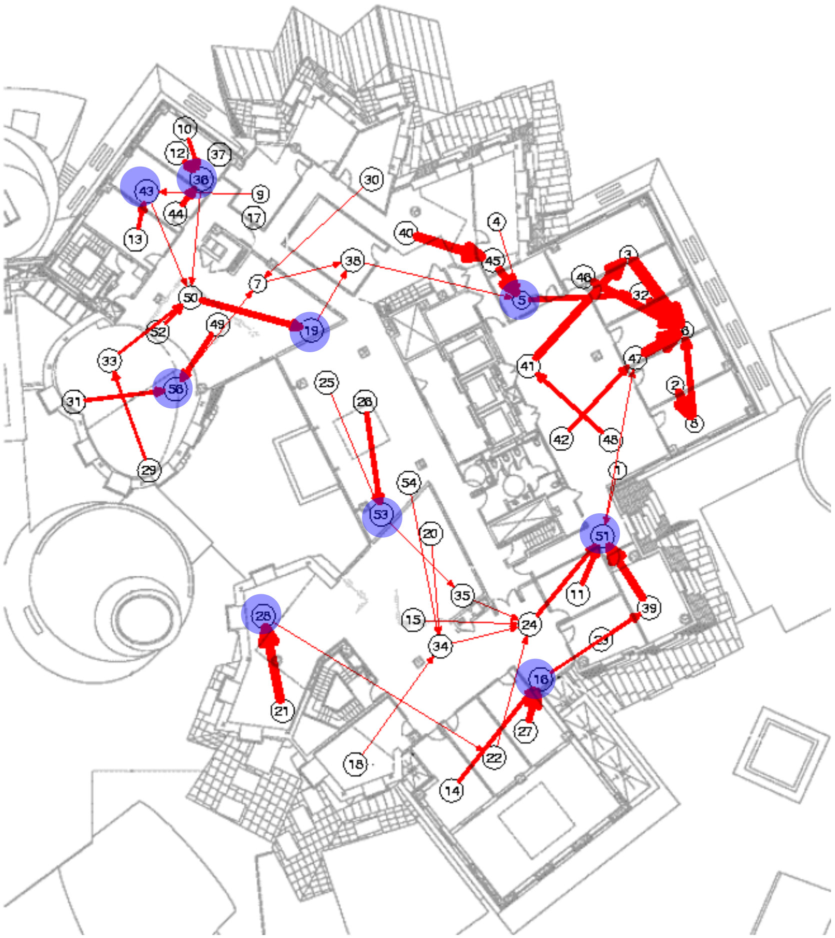
1 packet, 3 transmits, 1 received: η = 1/3

2 packets from bottom node, no channel loss, 1 buffer drop, 1 received: η = 2/(1+2) = 2/3

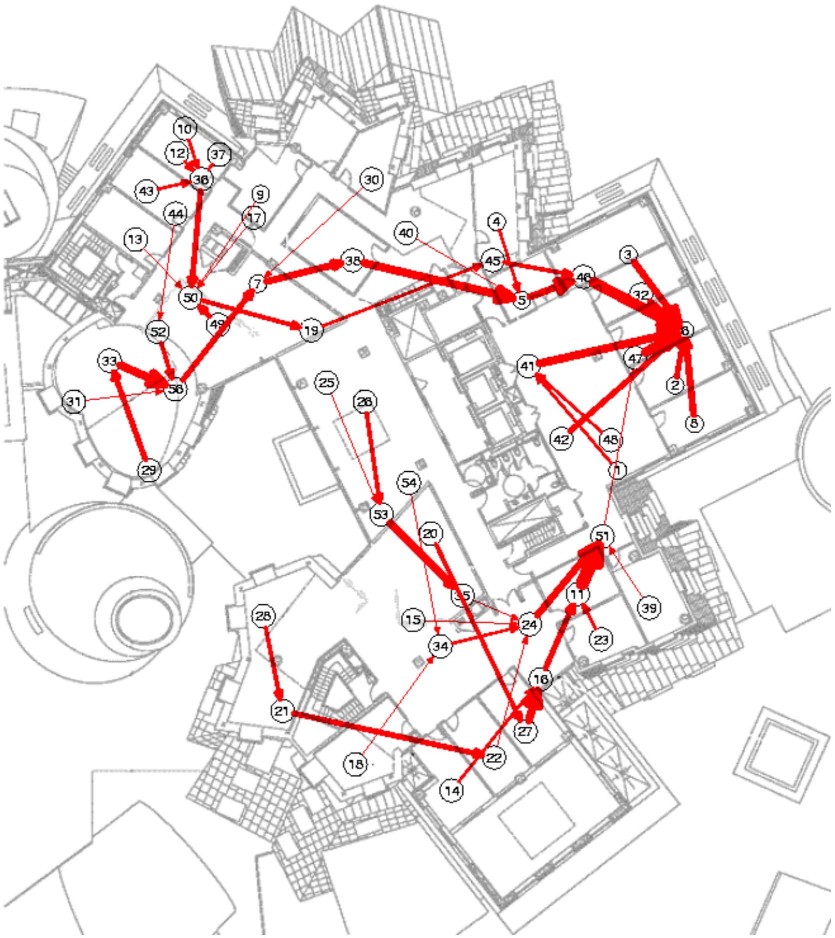
Hop-by-hop flow control improves efficiency



Hop-by-hop flow control conserves packets



No congestion control



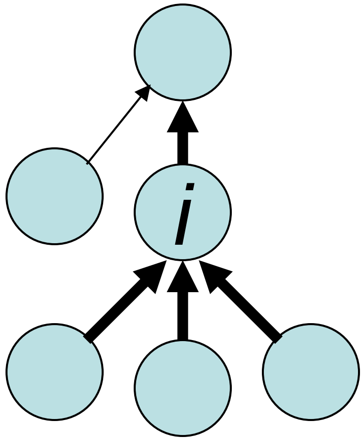
Hop-by-hop flow control

Metric: imbalance

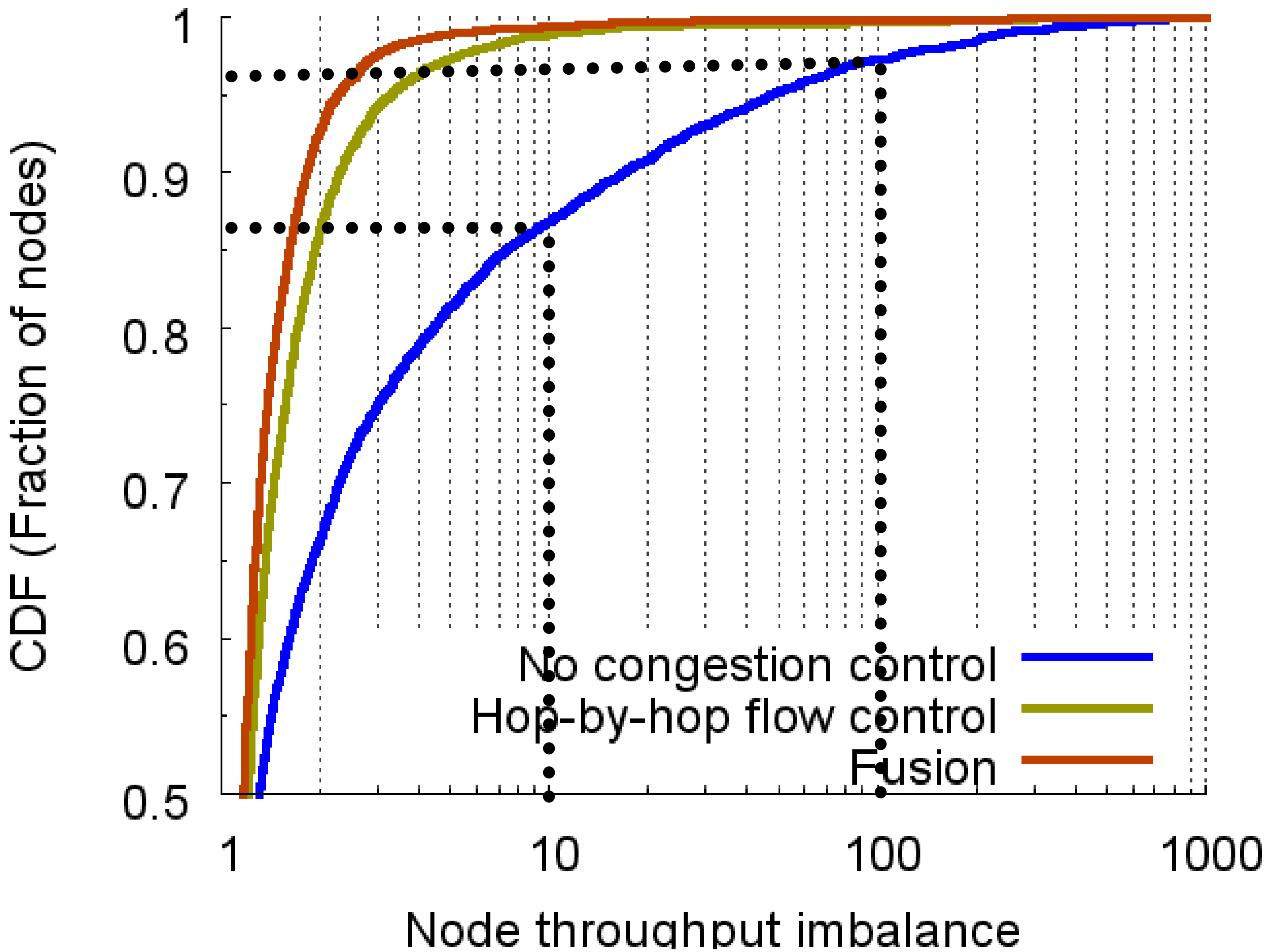
$$\zeta(i) = \frac{\mathbf{received}_*(i)}{\mathbf{received}_i(\mathbf{parent}(i))}$$

Interpretation: measure of how well a node can deliver received packets to its parent

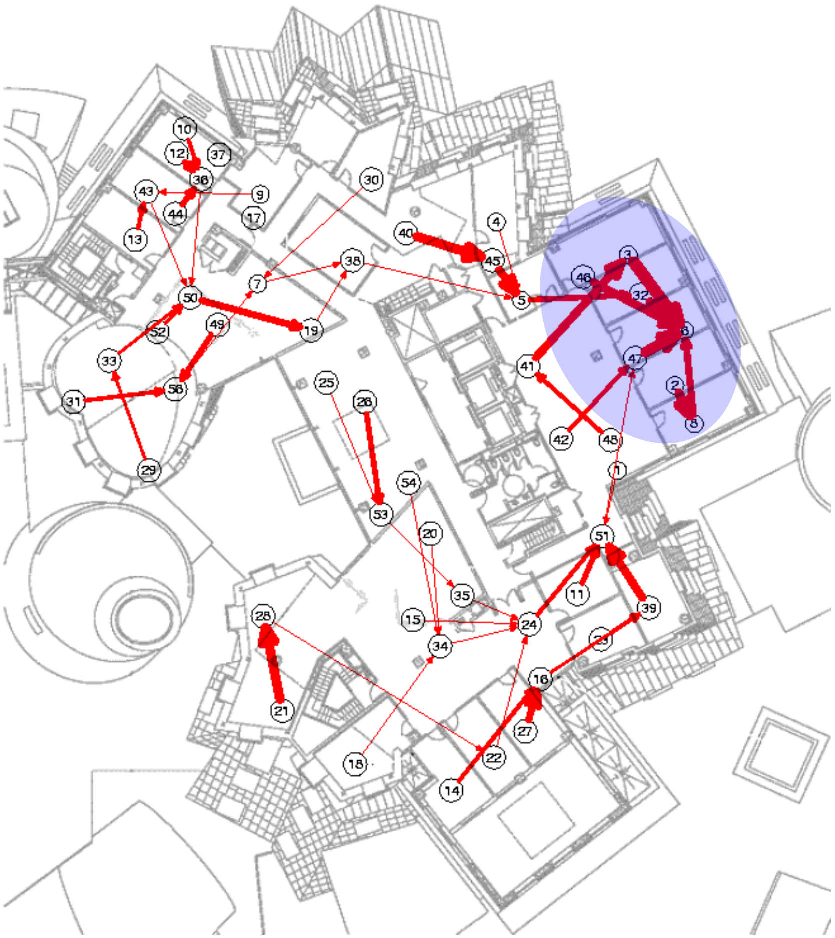
- ζ=1: deliver all received data
- ζ ↑: more data not delivered



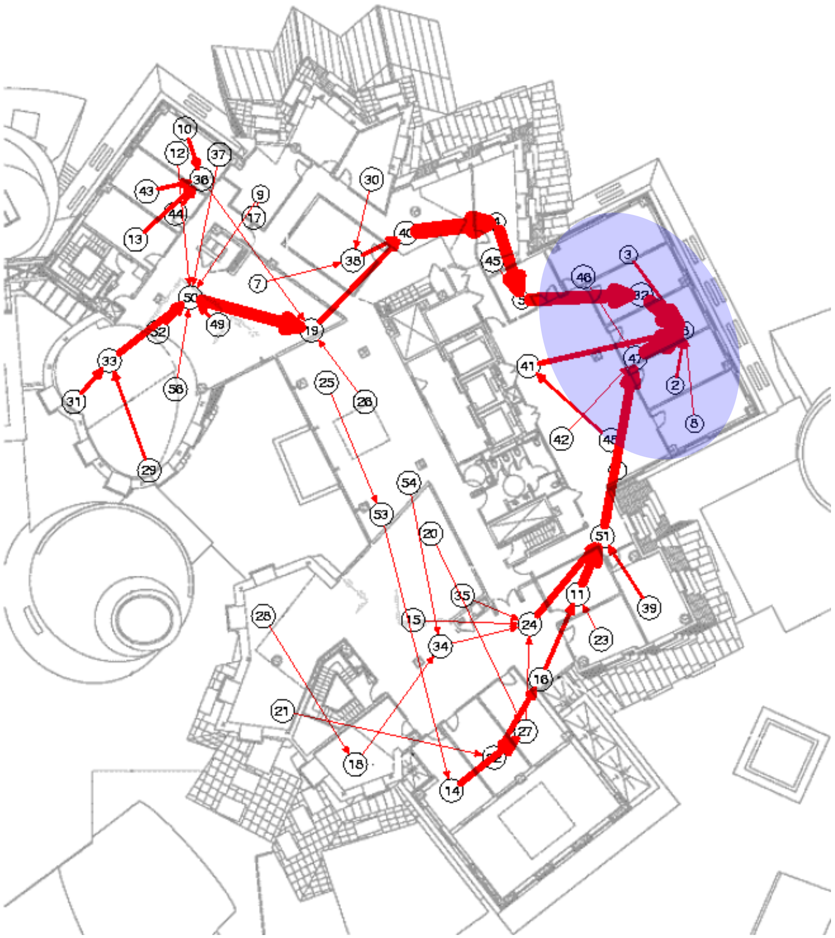
Periodic workload: imbalance



Rate limiting decreases sink contention

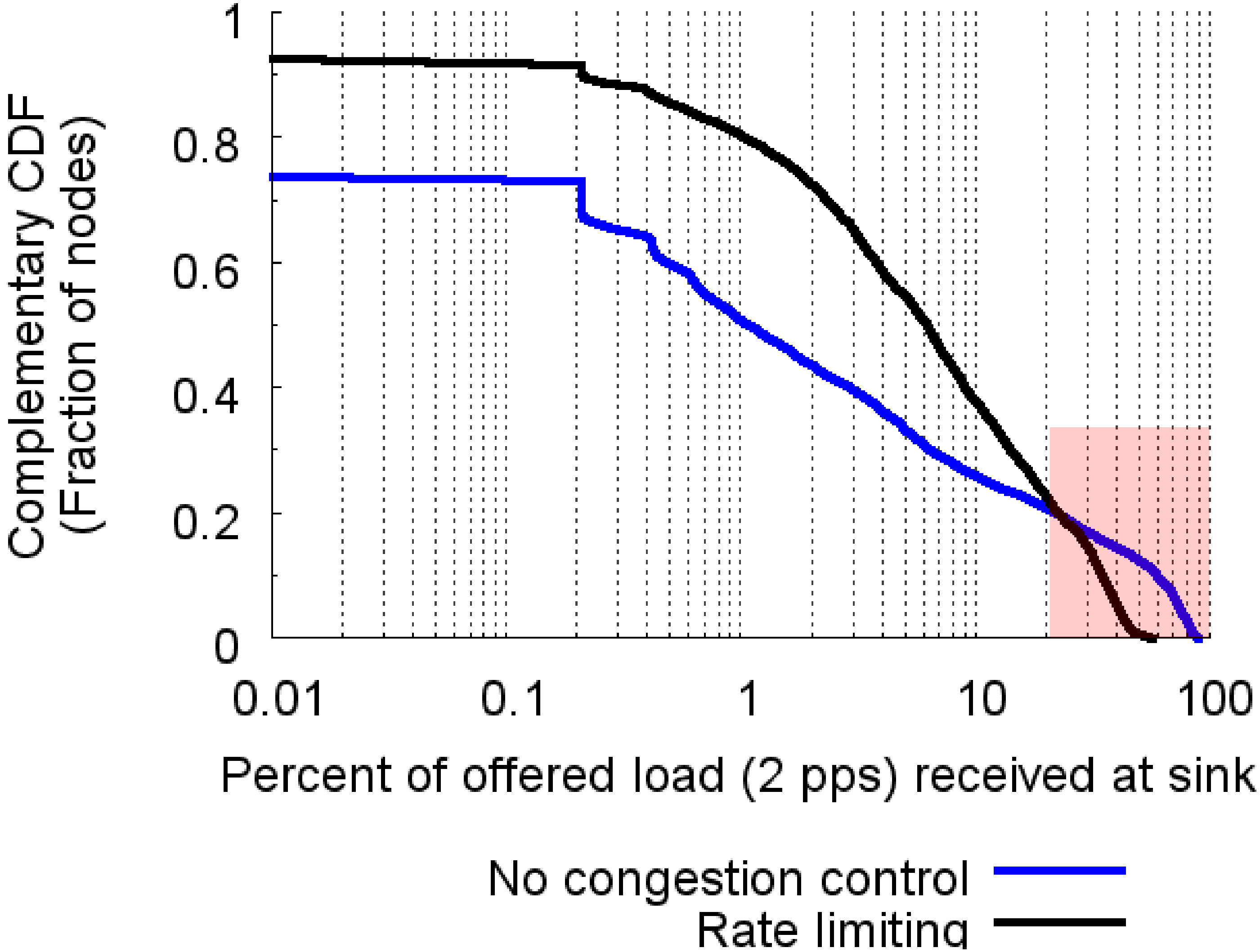


No congestion control

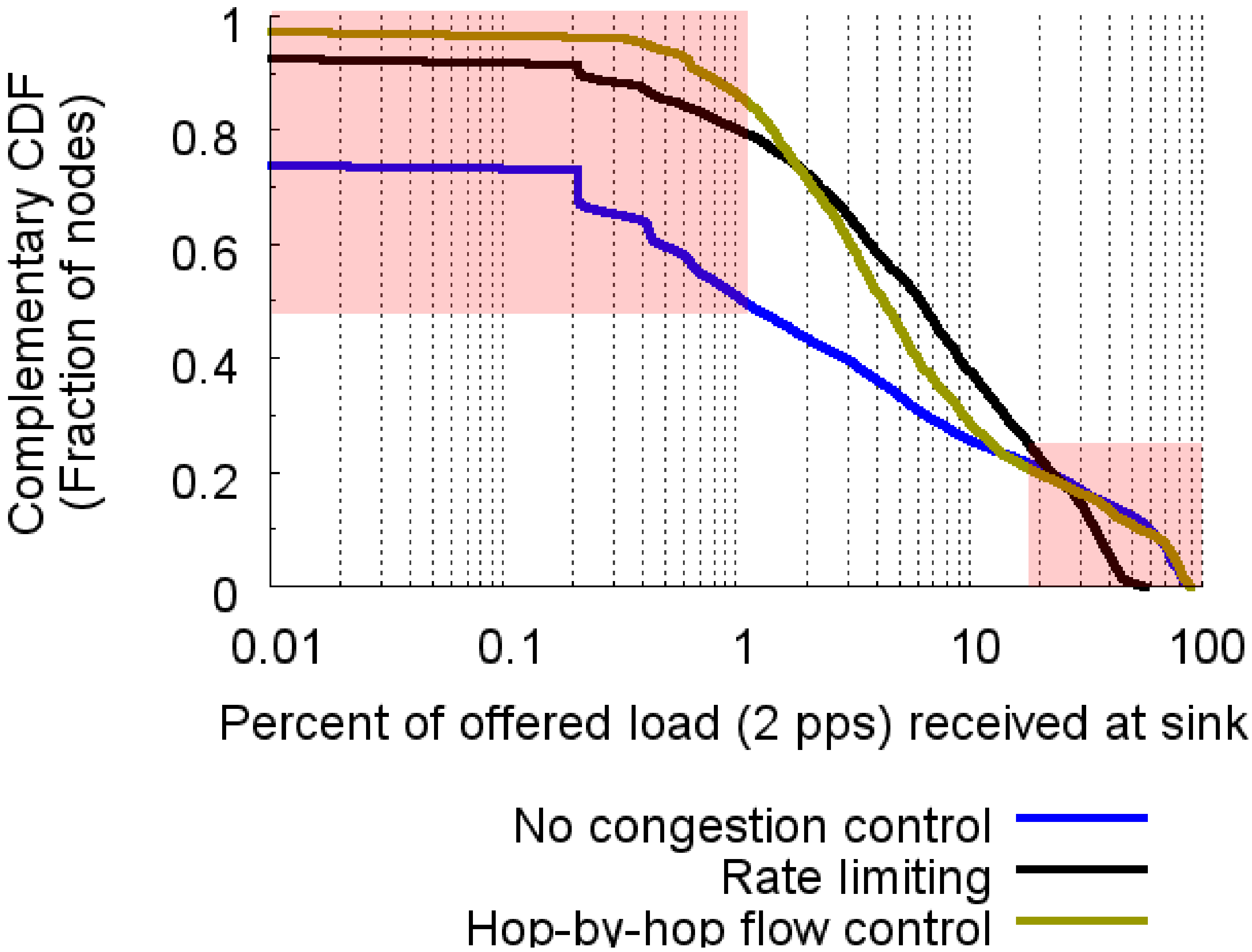


With *only* rate limiting

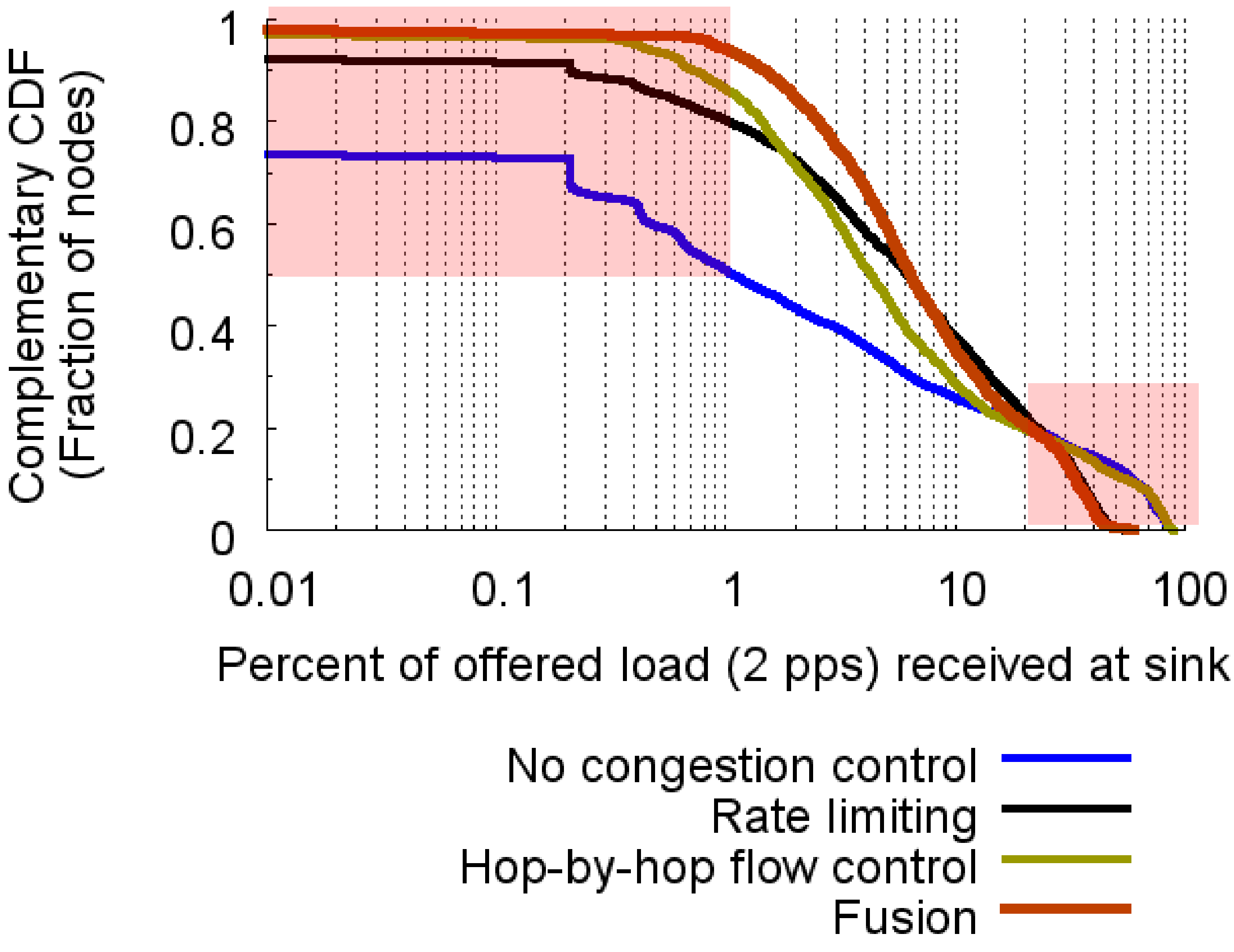
Rate limiting provides fairness



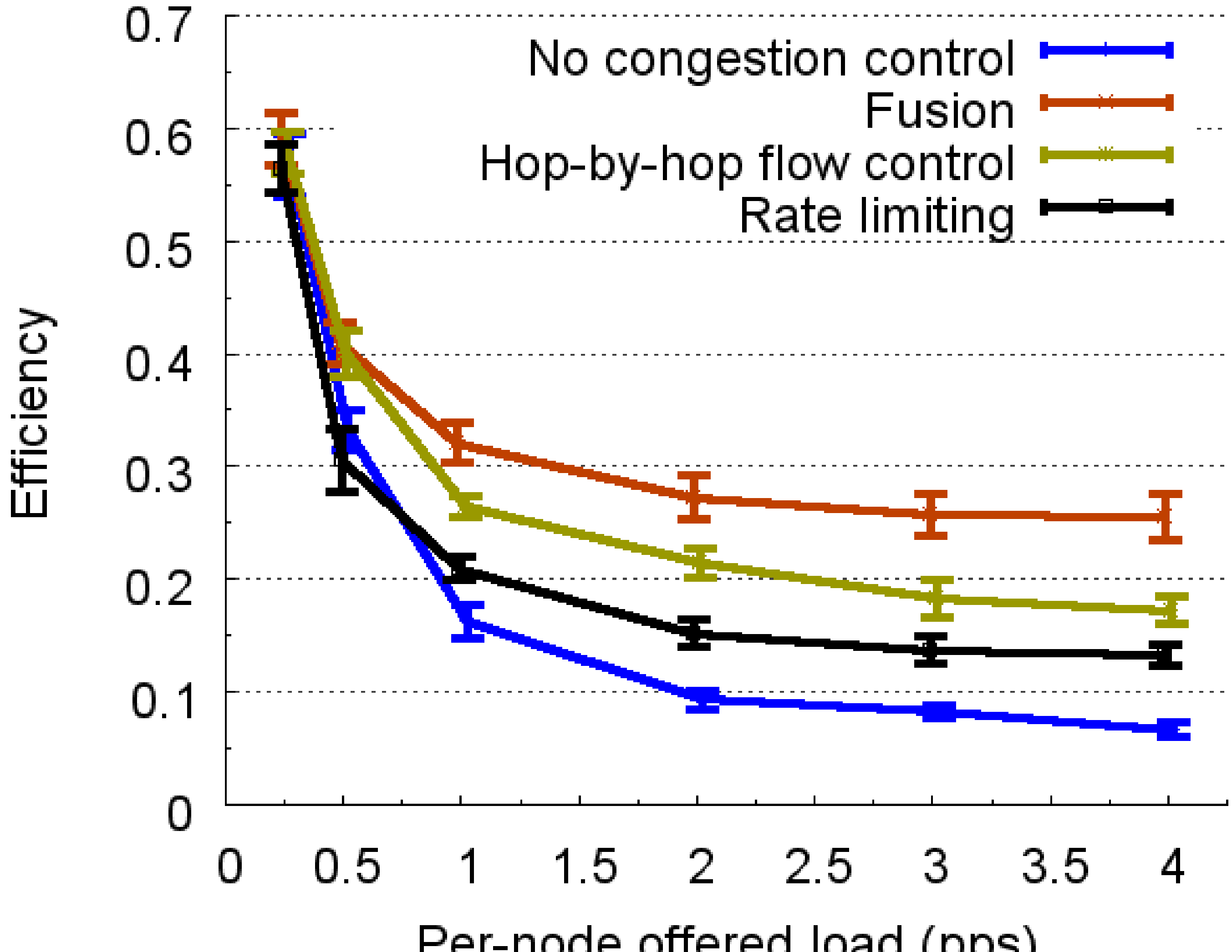
Hop-by-hop flow control prevents starvation



Fusion provides fairness and prevents starvation



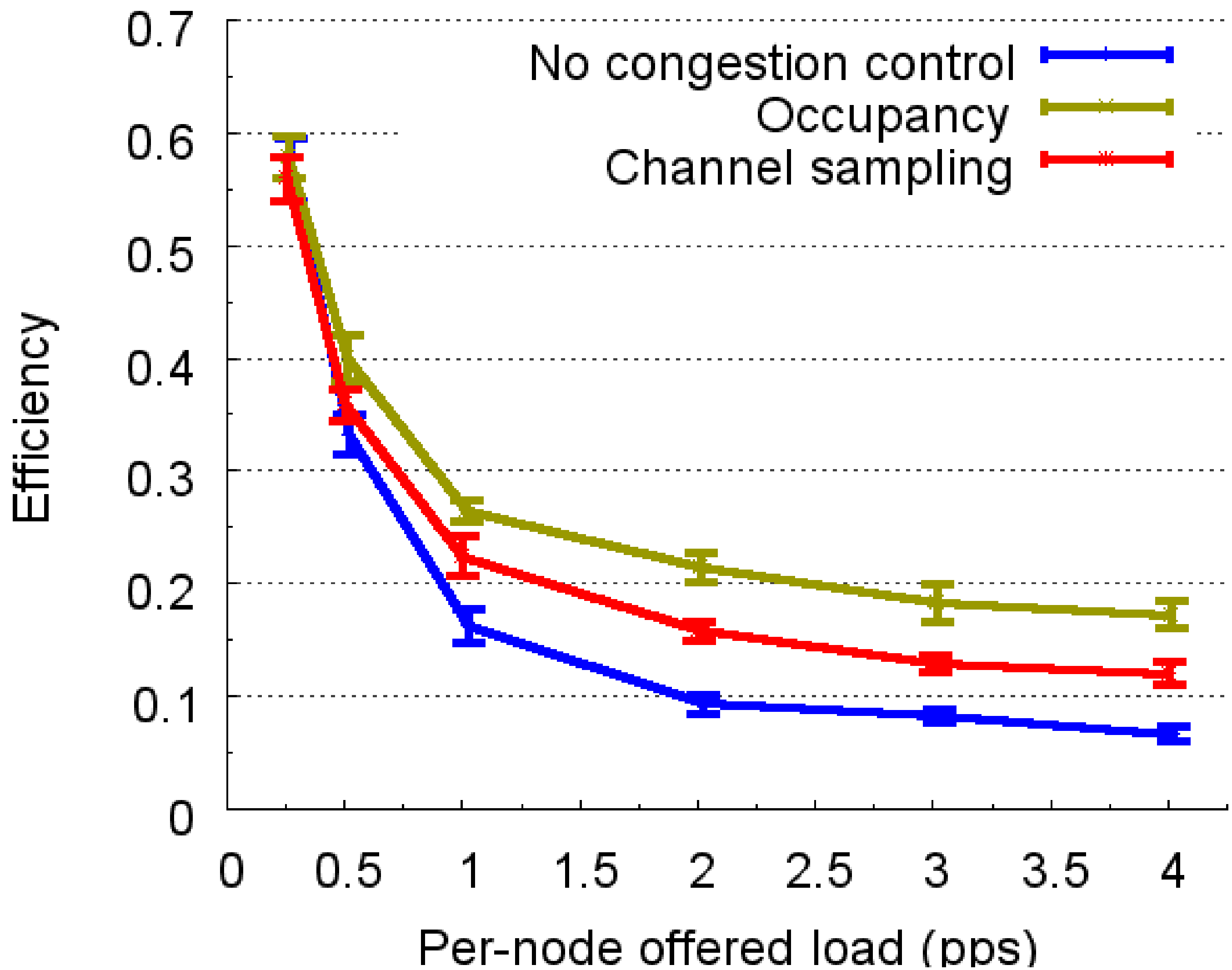
Synergy between rate limiting and hop-by-hop flow control



Alternatives for congestion detection

- **Queue occupancy**
- **Packet loss rate**
  - TCP uses loss to infer congestion
  - Keep link statistics: stop sending when drop rate increases
- **Channel sampling [Wan03]**
  - Carrier sense the channel periodically
  - *Congestion*: busy carrier sense more than a fraction of the time

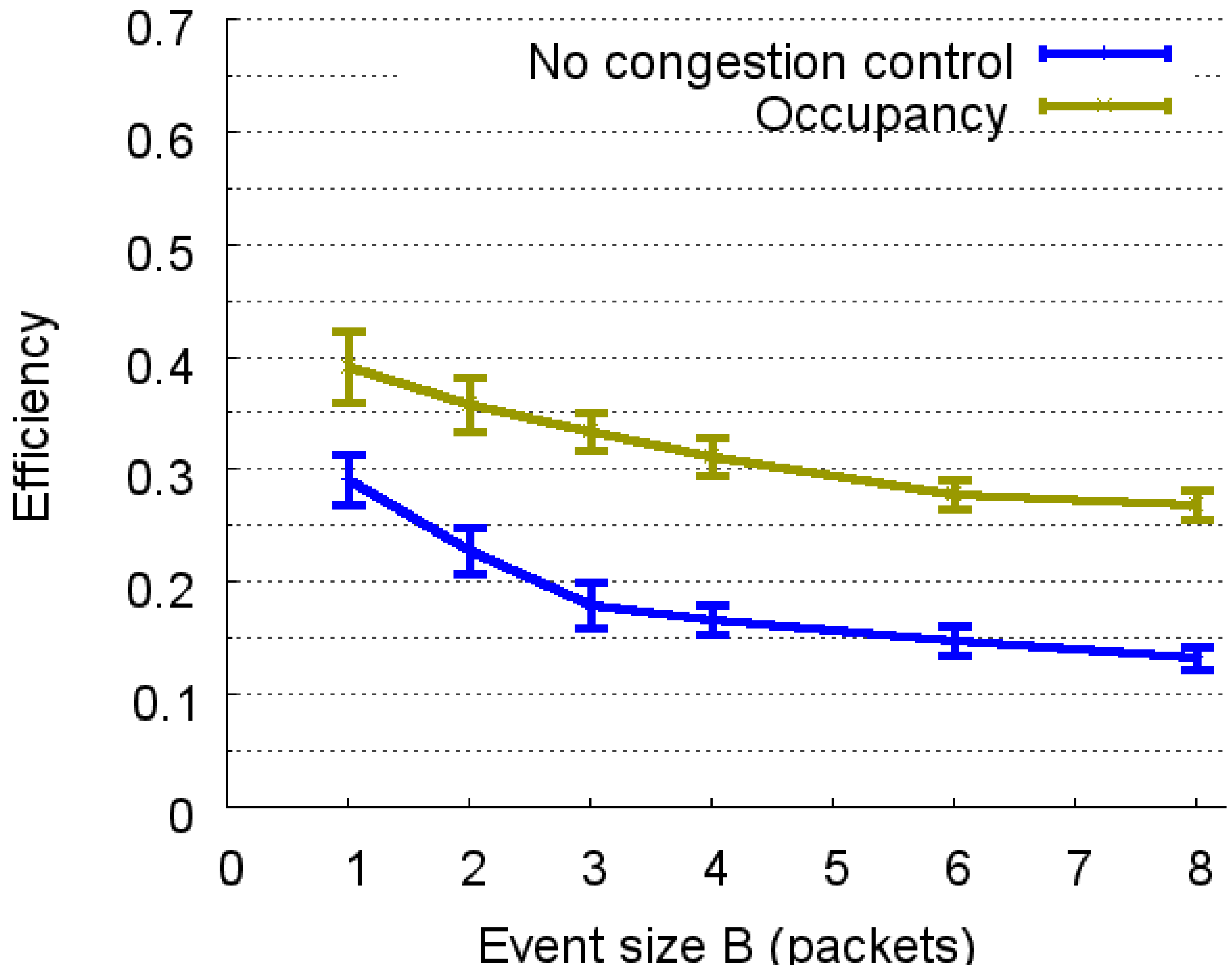
Comparing congestion detection methods



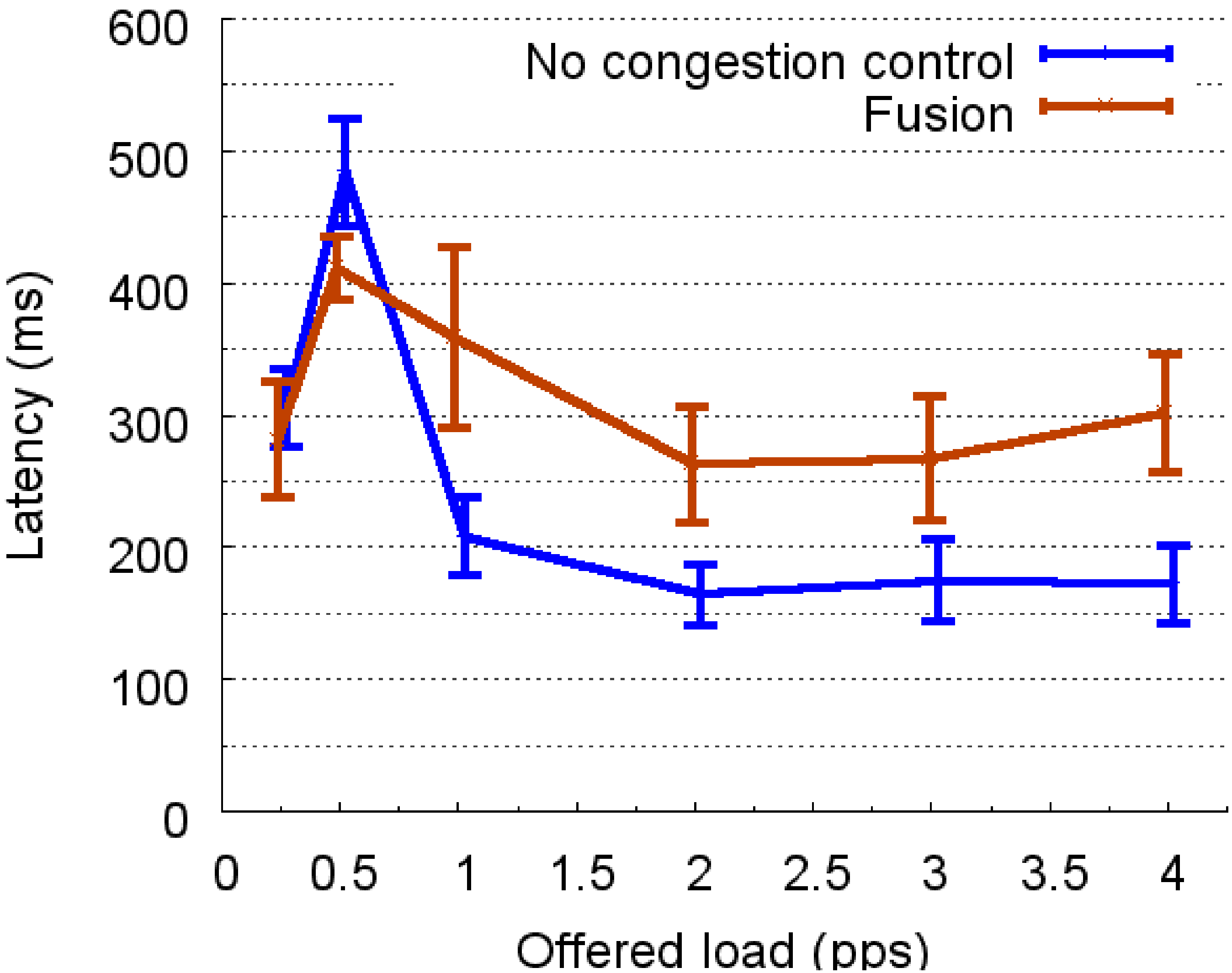
Correlated-event workload

- **Goal: evaluate congestion under an impulse of traffic**
  - Generate *events* seen by **all** nodes at the **same time**
  - At the event time each node:
    - Sends *B* back-to-back packets (“event size”)
    - Waits long enough for the network to drain

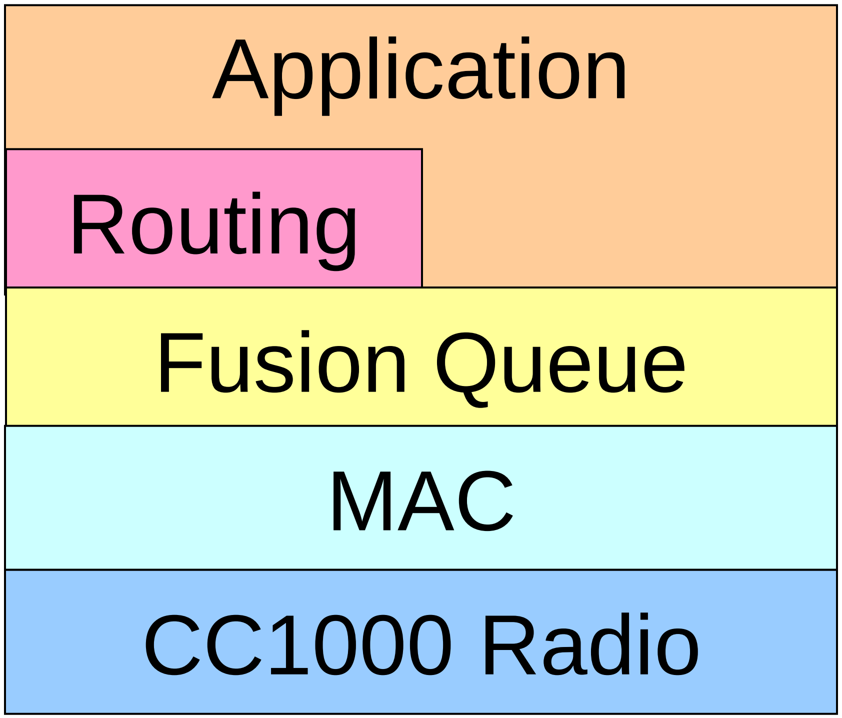
Small amounts of event-driven traffic cause congestion



Congestion control slows down the network



Software architecture



- Fusion implemented as a congestion-aware queue above MAC
- Apps need not be aware of congestion control implementation

Summary

- Congestion is a problem in wireless sensor networks
- Fusion's techniques mitigate congestion
  - Queue occupancy detects congestion
  - Hop-by-hop flow control improves efficiency
  - Source rate limiting improves fairness
- Fusion improves efficiency by 3× and eliminates starvation

http://nms.csail.mit.edu/fusion