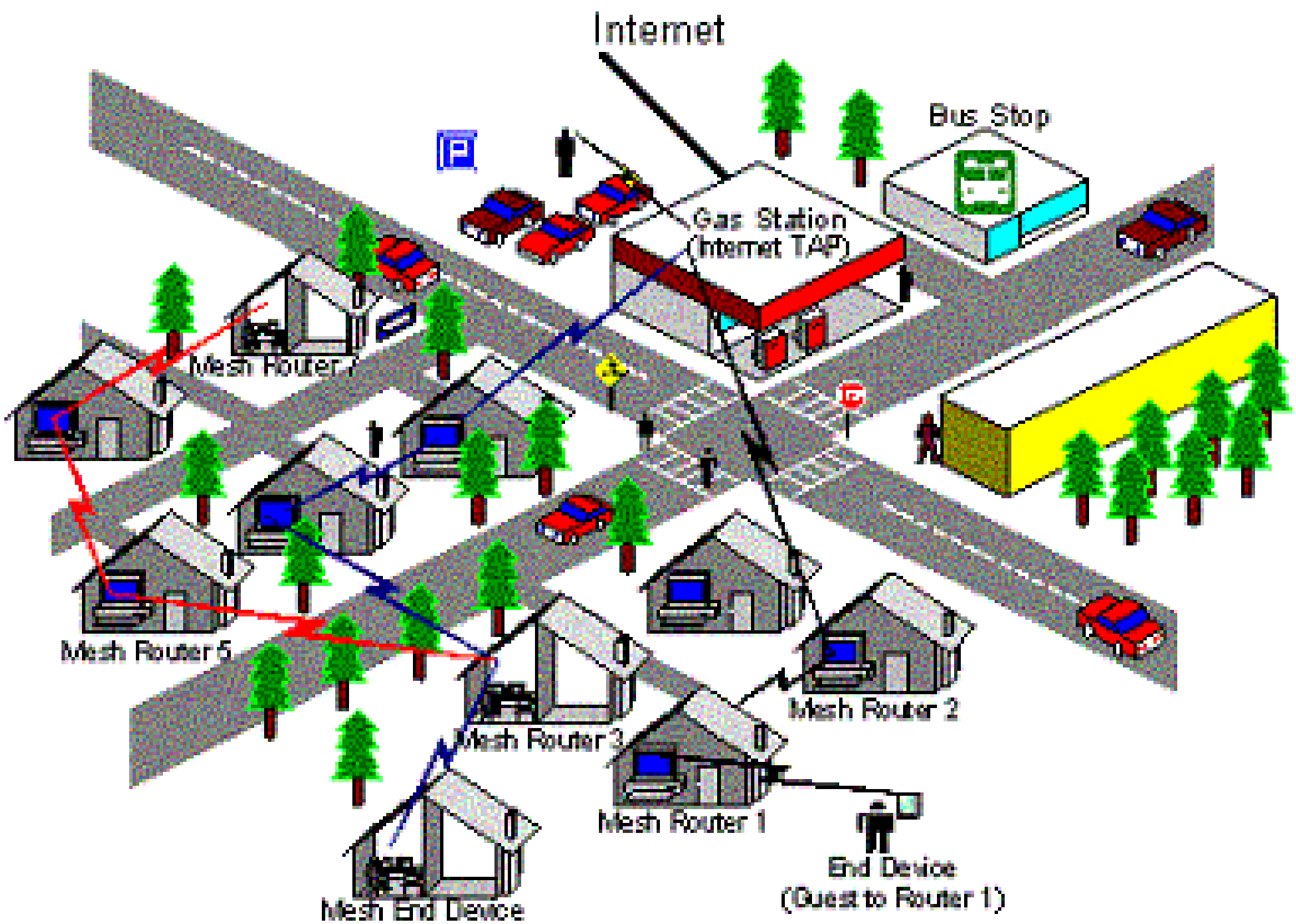
Routing in Wireless Mesh Networks

Feb. 22, 2006

Wireless Mesh Networks



Multi-hop Wireless Networks

	Stationary Nodes	Mobile Nodes
Motivating scenario	Community wireless networks (Mesh Networks)	Battlefield networks
Key challenge	Improving Network Capacity	Handling mobility, limited power.

A High-Throughput Path Metric for Multi-Hop Wireless Routing

D. S.J. Couto, D. Aguayo, J. Bicket, R. Morris

MIT

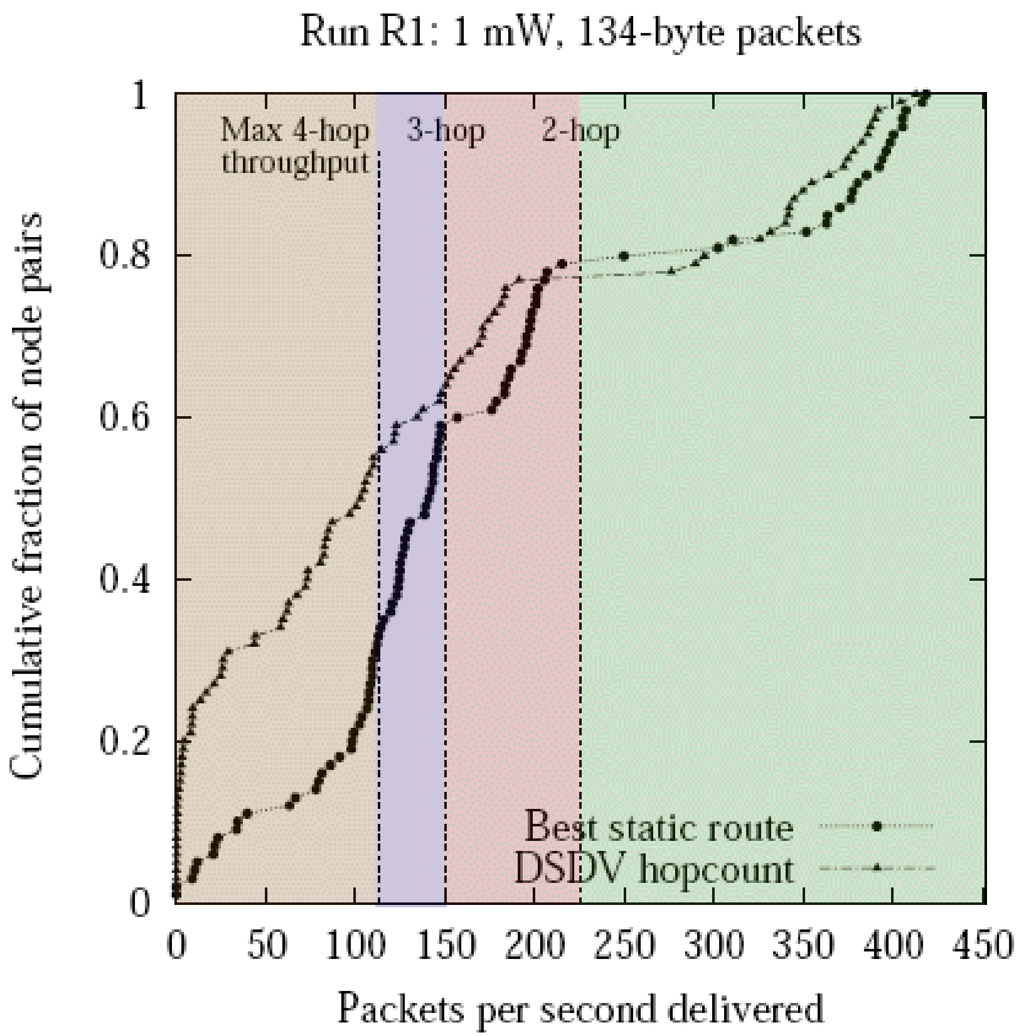
Hop Count Metric

- Maximizes the distance traveled by each hop
  - Minimizes signal strength -> Maximizes the loss ratio
  - Uses a higher TxPower -> Interference
- Possibly many shortest routes
- Avoid lossy links?

Hop Count vs. "Optimal"

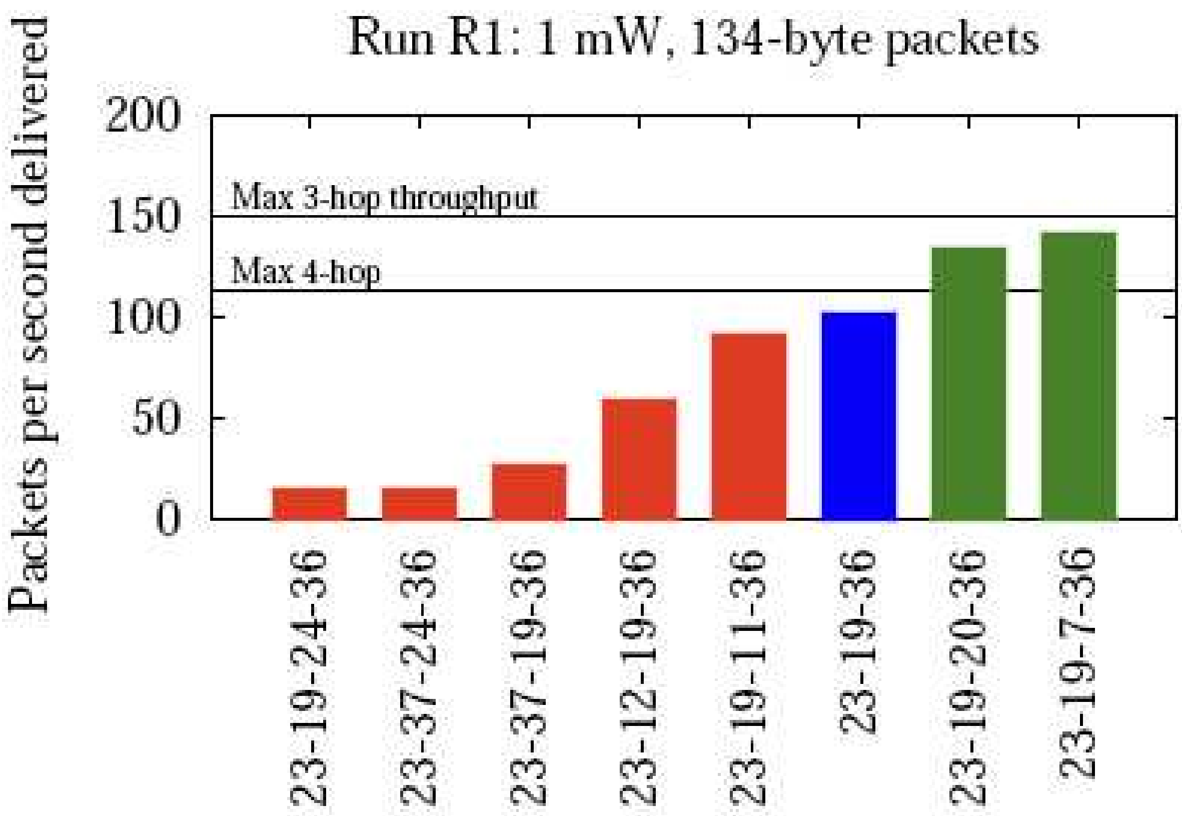
x axis:

throughput

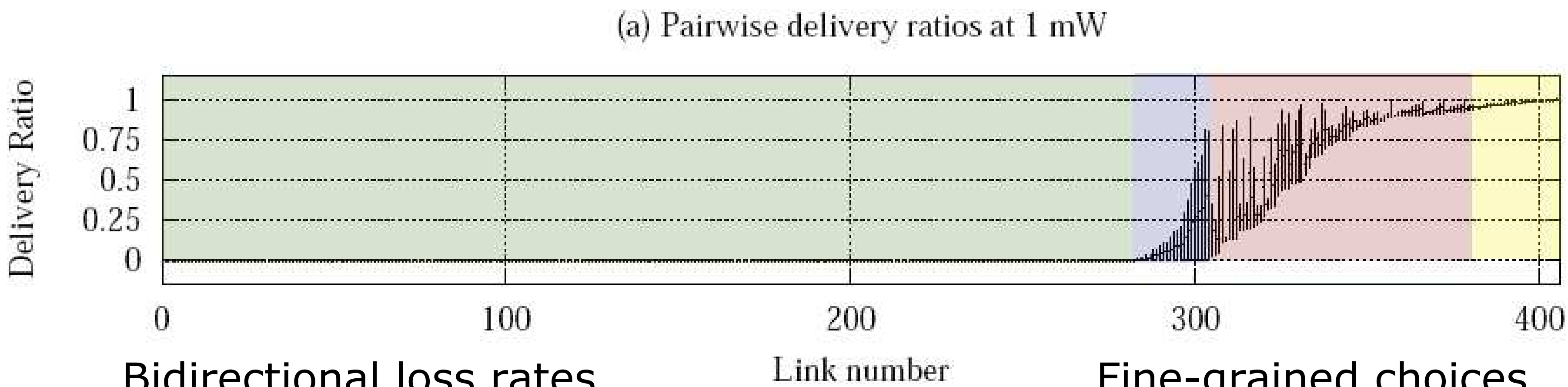


y axis: fraction of pairs with less throughput

Hop Count Route Selection

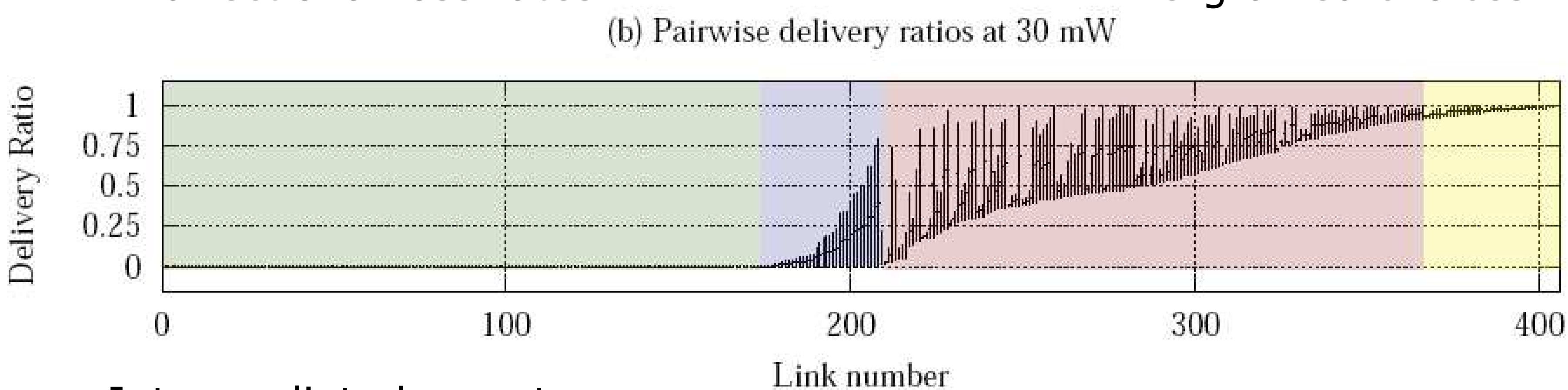


Motivation for a Better Routing Metric



Bidirectional loss rates

Fine-grained choices



Intermediate loss rates

What do you think are good routing metrics?

Potential Ideas (and their cons)

- Product of per-link delivery ratios
- Throughput of a path's bottleneck link
- End-to-End delay

Potential Ideas (and their cons)

- Product of per-link delivery ratios
  - A perfect 2-hop route is viewed as better than a 1-hop route with 10% loss ratio
- Throughput of a path's bottleneck link
  - Same as above
- End-to-End delay
  - Changes with network load as interference queue lengths vary… can cause oscillations

ETX

- The predicted number of data transmissions required to send a packet over a link
- The ETX of a path is the sum of the ETX values of the links over that path
- Examples:
  - ETX of a 3-hop route with perfect links is 3
  - ETX of a 1-hop route with 50% loss is 2

ETX continued…

- Expected probability that a transmission is successfully received and acknowledged is $d_f \times d_r$
  - d_f is forward delivery ratio
  - d_r is reverse delivery ratio
- Each attempt to transmit a packet is a Bernoulli trial, so…

$$\mathrm{ETX} = \frac{1}{d_f \times d_r}$$

Hooray for ETX!

- Based on delivery ratios, which affect throughput
- Detects and handles asymmetry by incorporating loss ratios in each direction
- Uses precise link loss ratios measurements to make fine-grained decisions between routes
  - Assumes you can measure these ratios precisely
- Penalizes routes with more hops, which have lower throughput due to inter-hop interference
  - Assumes loss rates are generally equal over links
- Tends to minimize spectrum use, which should maximize overall system capacity (reduce power too)
  - Each node spends less time retransmitting data

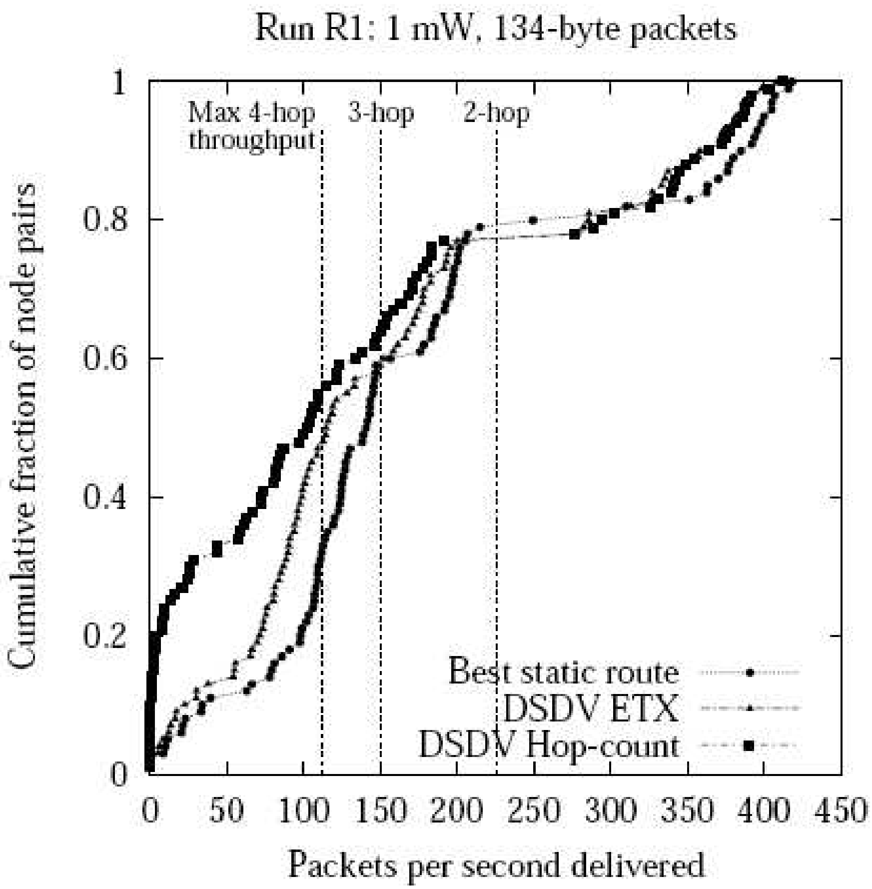
Acquiring ETX values

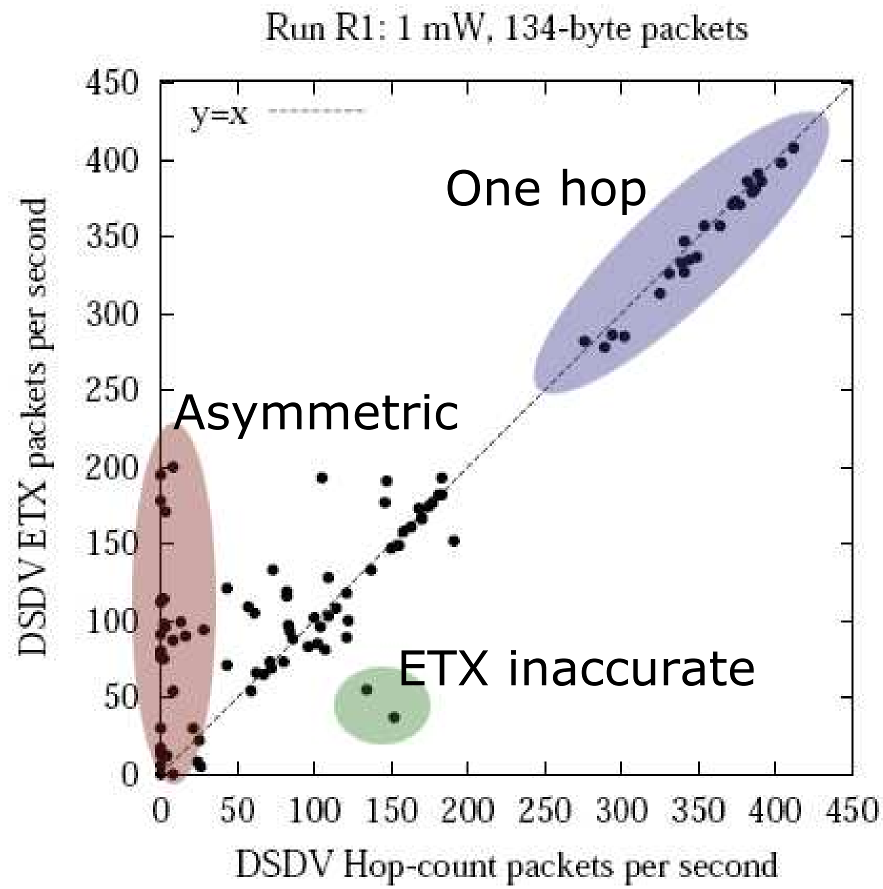
- Measured by broadcasting dedicated link probe packets with an average period τ (jittered by ±0.1τ)
- Delivery ratio: $$r(t) = \frac{\text{count}(t - w, t)}{w/\tau}$$
  - *count(t-w,t)* is the # of probes received during window *w*
  - *w/τ* is the # of probes that should have been received
- Each probe contains this information

Implementation and such…

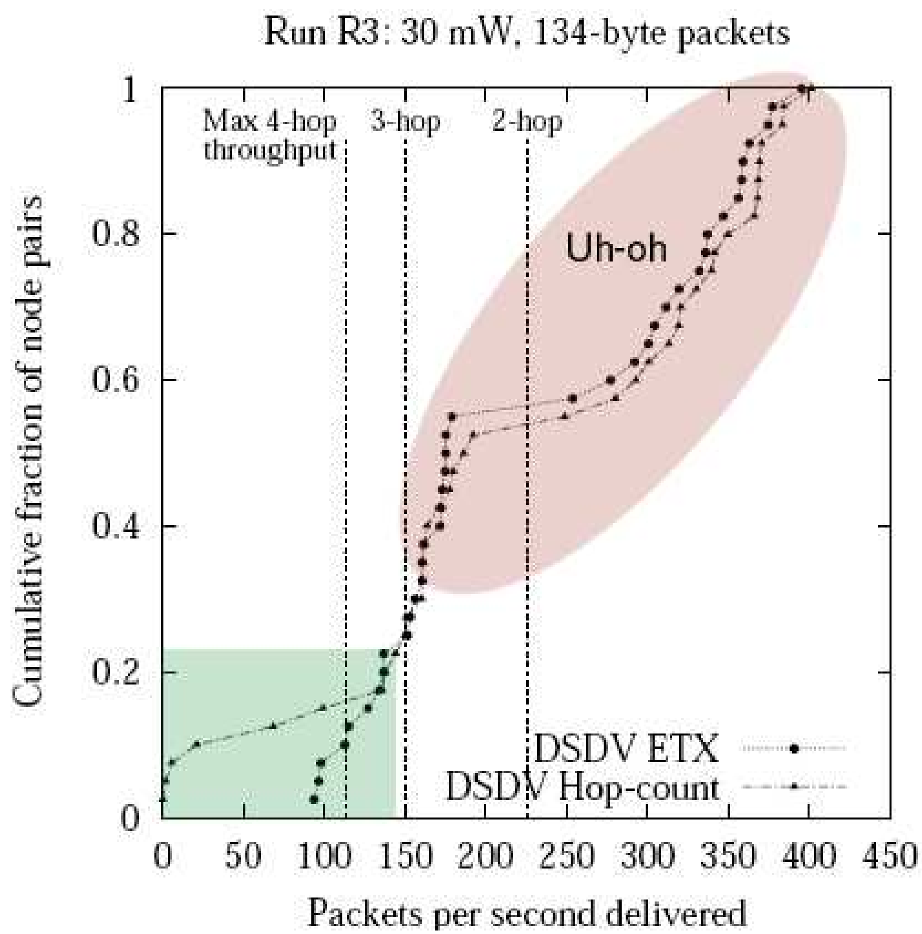
- Authors modified DSDV and DSR
- τ = 1 packet per second, w = 10 sec
- Multiple queues (different priorities)
  - Loss-ratio probes, protocol packets, data packets
- Are these experiments unfair or unrealistic?
  - In DSDV w/ ETX, route table is a snapshot taken at end of 90 second warm-up period
  - In DSR w/ ETX, source waits additional 15 sec before initiating the route request

DSDV Performance



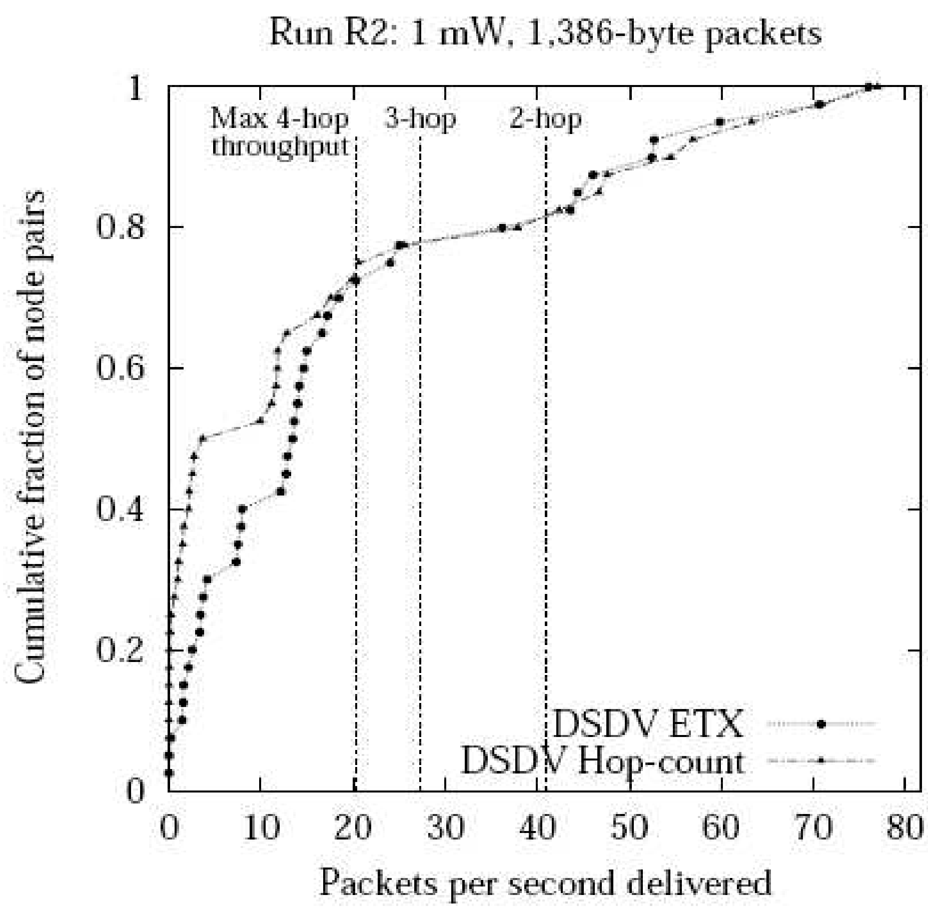


DSDV and High Transmit Power

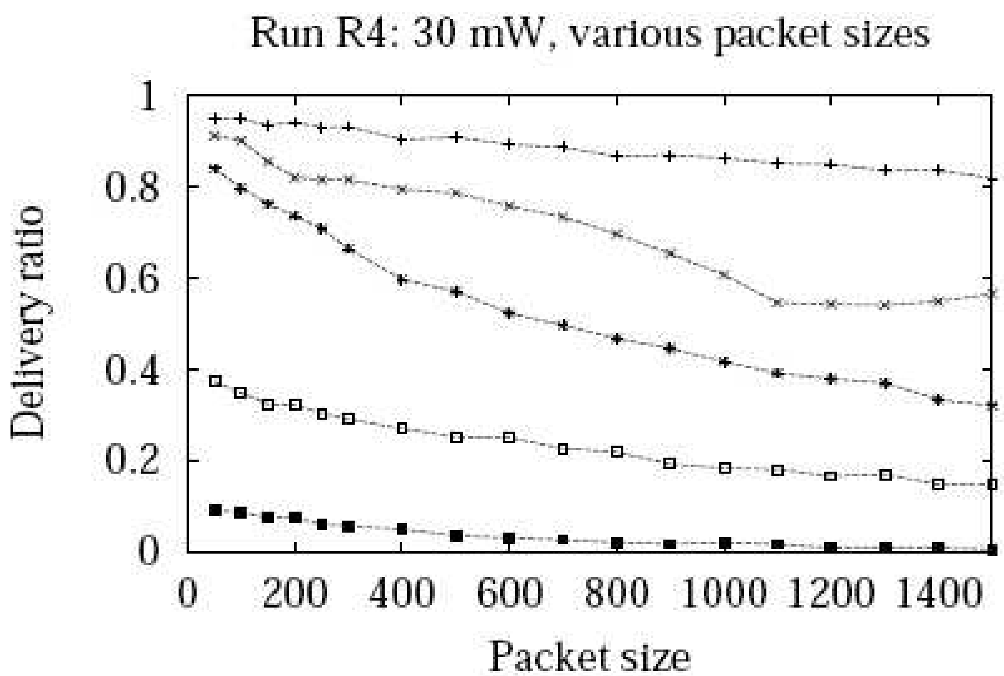


Packet Size Problems

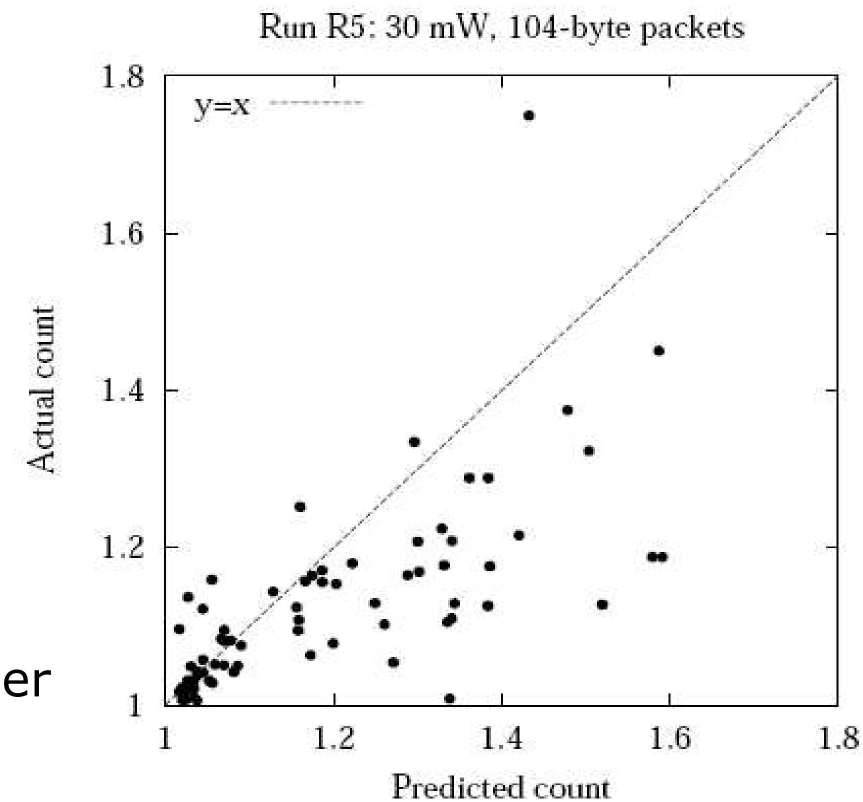
- Less throughput advantage than when data packets are smaller (134 bytes)



Packet sizes continued

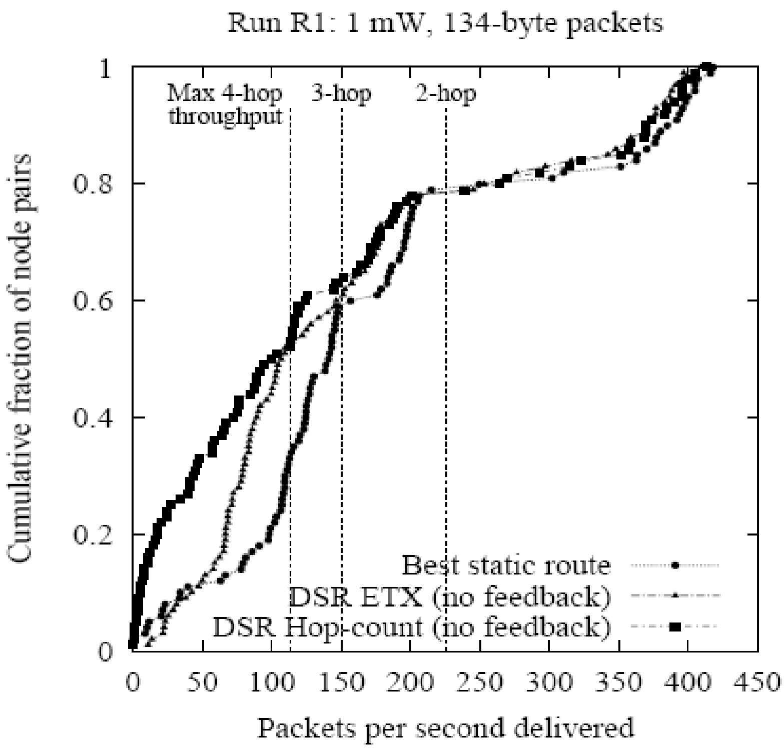


ACK's smaller than probe packets

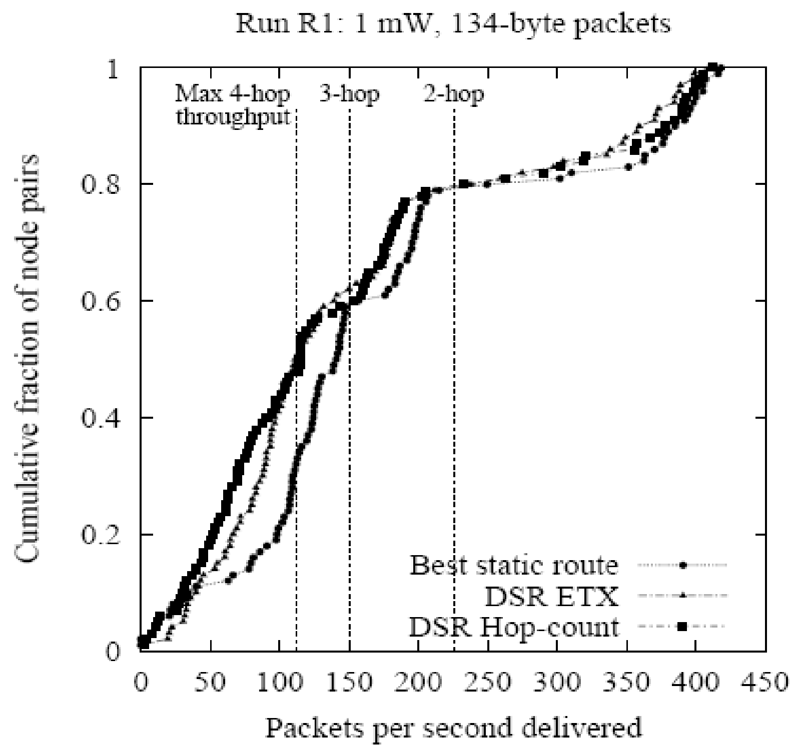


ETX underestimates ACK delivery ratios -> overestimates total number of transmissions per packet

DSR Performance



Link-layer transmission feedback disabled



Link-layer transmission feedback enabled

Discussion

- Pros?

- Cons?

Conclusions

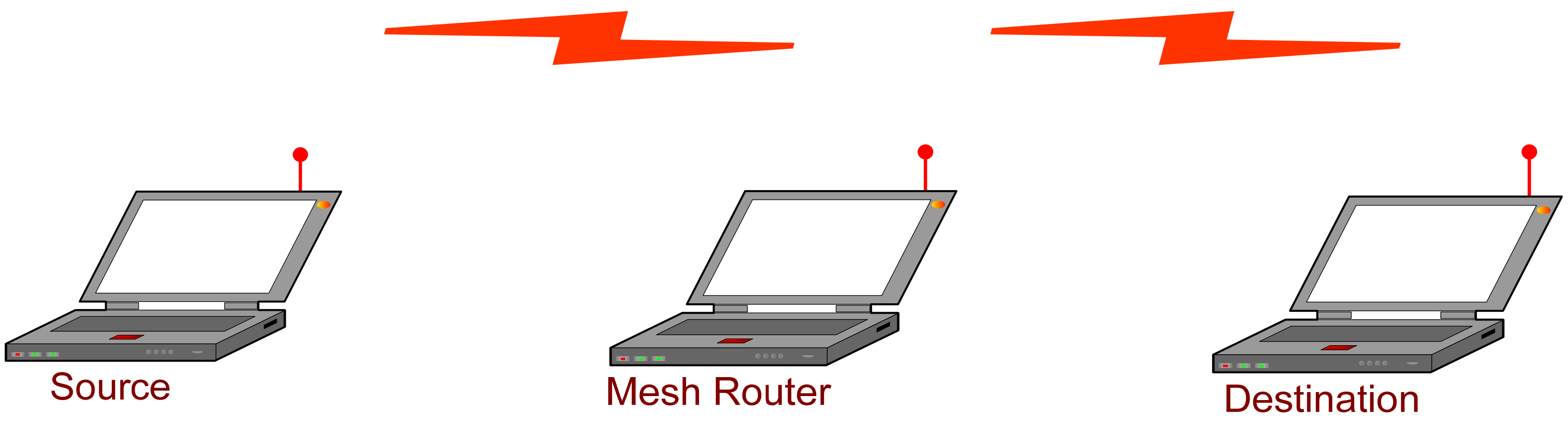
- Pros
  - ETX performs better or comparable to Hop Count Metric
    - Accounts for bi-directional loss rates
  - Can easily be incorporated into routing protocols
- Cons
  - May not be best metric for all networks
    - Mobility
    - Power-limited
    - Adaptive Rate (multi-rate)
    - Multi-radio
  - Predications of loss ratios not always accurate and incur overhead

Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks

Richard Draves, Jitendra Padhye, and Brian Zill

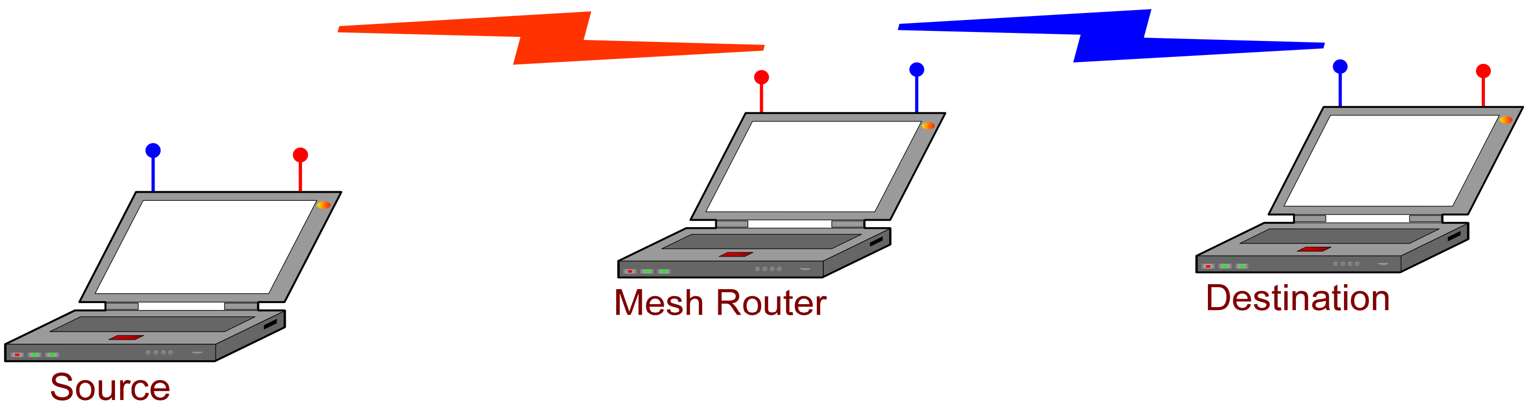
Microsoft Research

Multi-Hop Networks with Single Radio



With a single radio, a node can not transmit and receive simultaneously.

Multi-Hop Networks with Multiple Radios



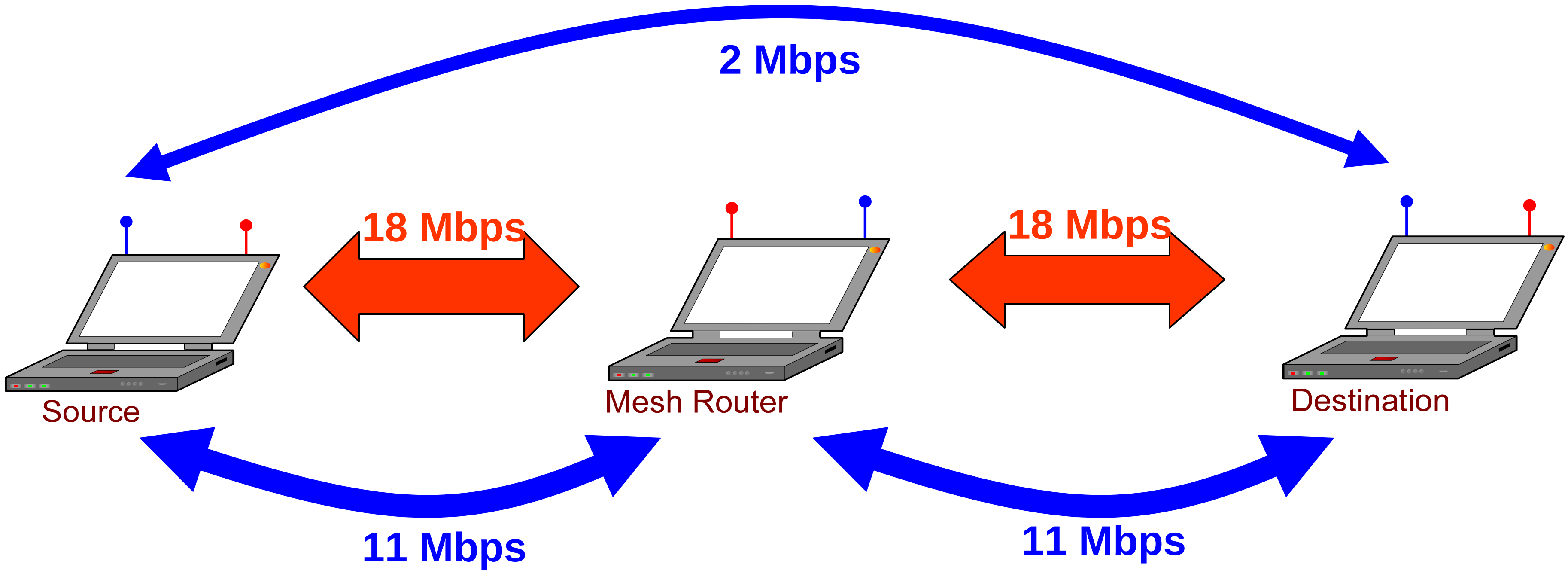
With two radios tuned to non-interfering channels, a node can transmit and receive simultaneously.

Other Advantages of Multiple Radios

- Increased robustness due to frequency diversity
  - e.g. 2.4GHz (802.11b) and 5GHz (802.11a) have different fading characteristics

- Possible tradeoff between range and data rate
  - Can be helpful during early deployment

Existing Routing Metrics are Inadequate



Shortest path: 2 Mbps

Path with fastest links: 9 Mbps

Best path: 11 Mbps

Contributions of this Paper

- New routing metric for multi-radio mesh networks
  - Weighted Cumulative Expected Transmission Time (WCETT)
- Implementation of the metric in a link-state routing protocol
  - Multi-Radio Link-Quality source routing (MR-LQSR)
- Experimental evaluation of WCETT:
  - 24-node, multi-radio mesh testbed
  - 2 radios per node, 11a and 11g
  - Side-by-side comparison with:
    - Shortest path (HOP)
    - ETX (De Couto et. al. MOBICOM 2003)

Summary of Results

- WCETT makes judicious use of two radios
  - Over 250% better than HOP
  - Over 80% better than ETX

- Gains more prominent over shorter paths and in lightly-loaded scenarios

Outline of the talk

- Design of WCETT
- Experimental results
- Conclusion

Design of Routing Metric: Assumptions

- No power constraints
- Little or no node mobility
  - Relatively stable links
- Nodes have one or more 802.11 radios
- Multiple radios on a node are tuned to non-interfering channels
  - Channel assignment is fixed

Implementation Framework

- Implemented in a source-routed, link-state protocol
  - Multi-Radio Link Quality Source Routing (MR-LQSR)
- Nodes discovers links to its neighbors; Measure quality of those links
- Link information floods through the network
  - Each node has "full knowledge" of the topology
- Sender selects "best path"
  - Packets are source routed using this path

Goal for Multi-Radio Routing Metric

Maximize throughput of a given flow:

- Prefer high-bandwidth, low-loss links
- When possible, select channel diverse paths
- Prefer shorter paths

Components of a Routing Metric

- **Link Metric:** Assign a weight to each link

WCETT: Prefer high-bandwidth, low-loss links

- **Path Metric:** Combine metrics of links on path

WCETT: Prefer short, channel-diverse paths

Link Metric: Expected Transmission Time (ETT)

- Link loss rate = p
  - Expected number of transmissions

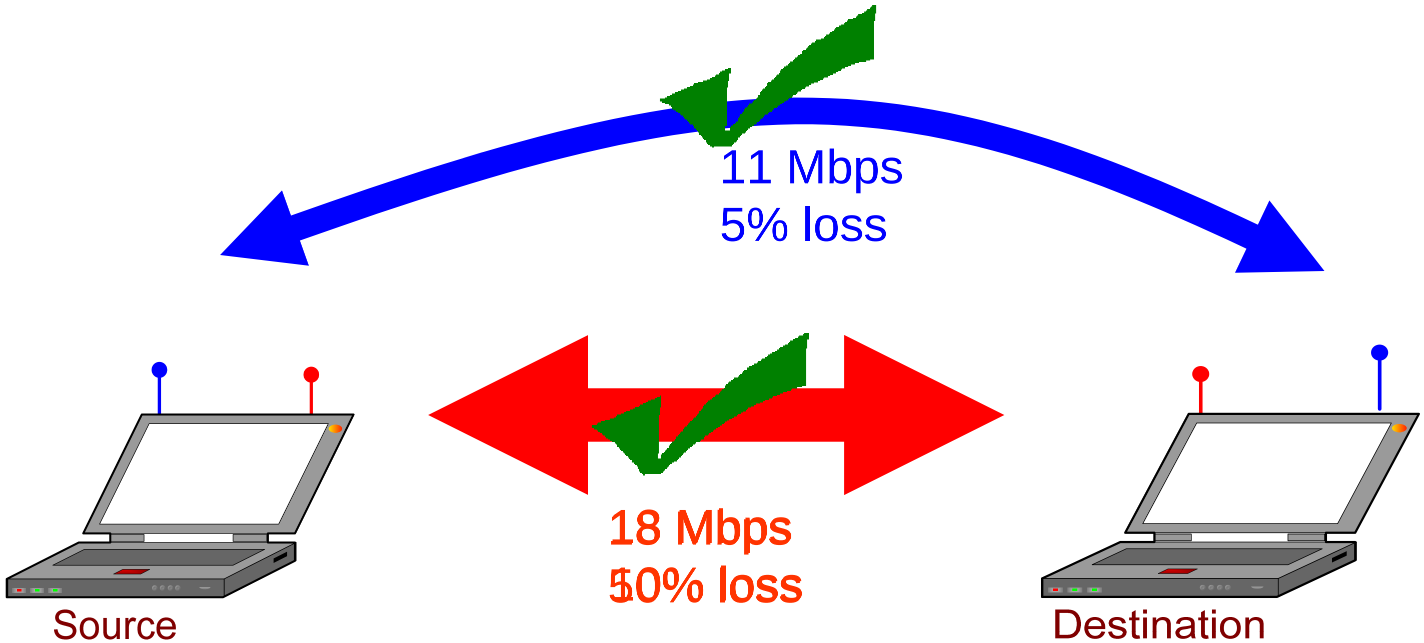
$$\mathrm{ETX} = \frac{1}{1-p}$$

- Packet size = S, Link bandwidth = B
  - Each transmission lasts for S/B

$$\mathrm{ETT} = \left(\frac{S}{B}\right) * \mathrm{ETX}$$

- Lower ETT implies better link

ETT: Illustration



1000 Byte Packet

ETT : 0.77 ms

ETT : 0.89 ms

Combining Link Metric into Path Metric Proposal 1

- Add ETTs of all links on the path
- Use the sum as path metric

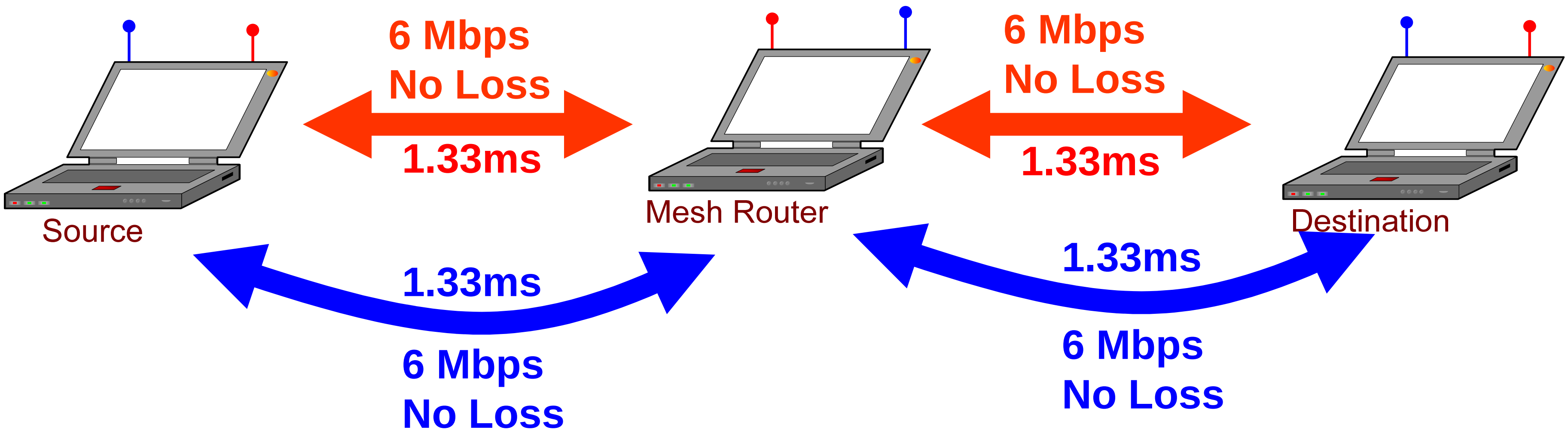
SETT = Sum of ETTs of links on path

(Lower SETT implies better path)

Pro: Favors short paths

Con: Does not favor channel diversity

SETT does not favor channel diversity



Path	Throughput	SETT
Red-Blue	6 Mbps	2.66 ms
Red-Red	3 Mbps	2.66 ms

Impact of Interference

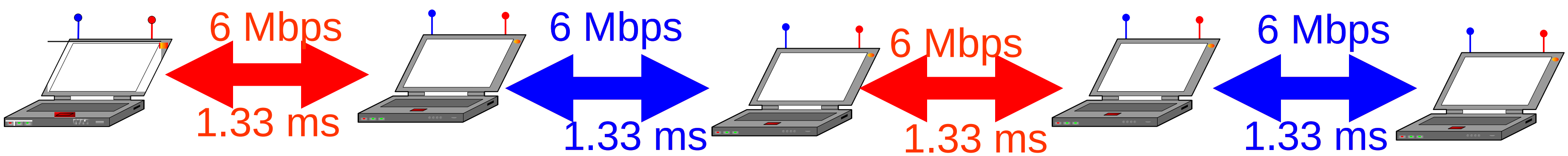
- Interference reduces throughput
- Throughput of a path is lower if many links are on the same channel
  - Path metric should be worse for non-diverse paths
- *Assumption: All links that are on the same channel interfere with one another*
  - Pessimistic for long paths

Combining Link Metric into Path Metric Proposal 2

- Group links on a path according to channel
  - Links on same channel interfere
- Add ETTs of links in each group
- Find the group with largest sum.
  - This is the “bottleneck” group
  - Too many links, or links with high ETT (“poor quality” links)
- Use this largest sum as the path metric
  - **Lower value implies better path**

“Bottleneck Group ETT” (BG-ETT)

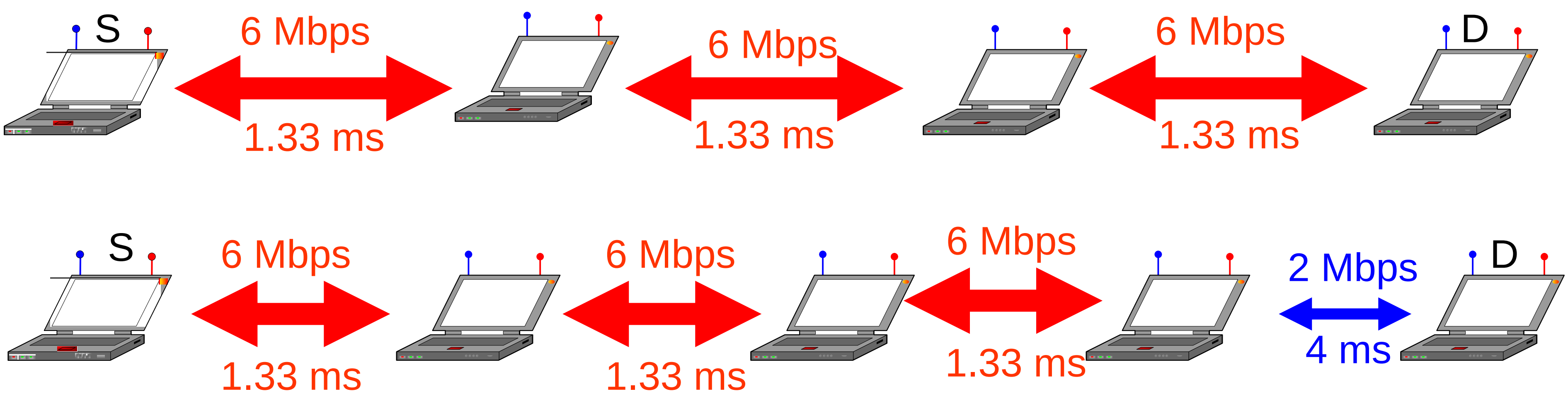
BG-ETT Example



Path	Throughput	Blue Sum	Red Sum	BG-ETT
All Red	1.5 Mbps	0	5.33 ms	5.33 ms
1 Blue	2 Mbps	1.33 ms	4 ms	4 ms
Red-Blue	3 Mbps	2.66 ms	2.66 ms	2.66 ms

BG-ETT favors high-throughput, channel-diverse paths.

BG-ETT does not favor short paths



Path	Throughput	Blue Sum	Red Sum	BG-ETT
3-Hop	2 Mbps	0	4 ms	4 ms
4-Hop	2 Mbps	4 ms	4 ms	4 ms

Path Metric: Putting it all together

- SETT favors short paths
- BG-ETT favors channel diverse paths

Weighted Cumulative ETT (WCETT)

$$WCETT = (1-\beta) * SETT + \beta * BG\text{-}ETT$$

β is a tunable parameter

Higher value: More preference to channel diversity
Lower value: More preference to shorter paths

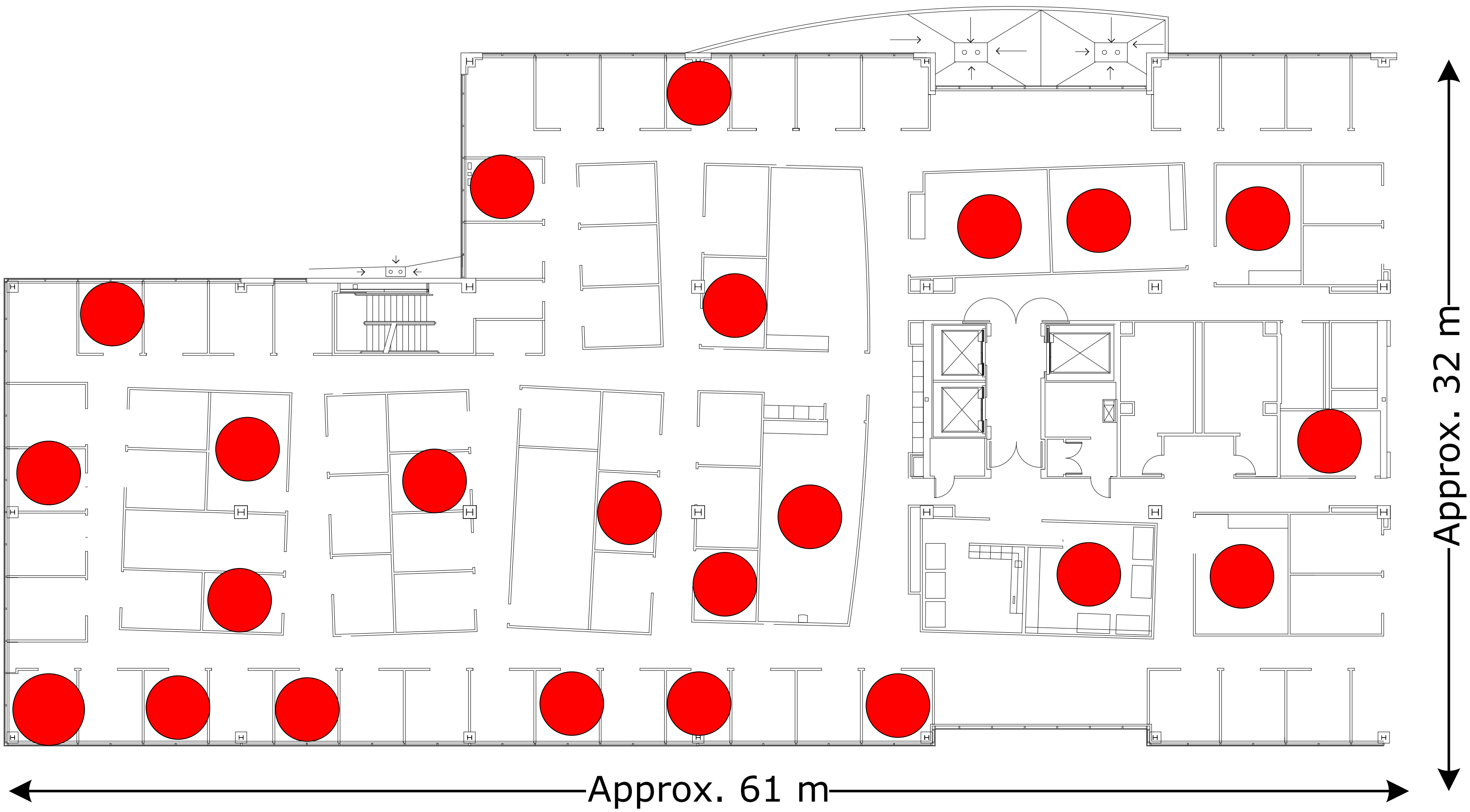
How to measure loss rate and bandwidth?

- Loss rate measured using broadcast probes
  - Similar to ETX
  - Updated every second
- Bandwidth estimated using periodic packet-pairs
  - Updated every 5 minutes

Outline of the talk

- Design of WCETT
- Experimental results
- Conclusion

Mesh Testbed



23 nodes running Windows XP.

Two 802.11a/b/g cards per node: Proxim and NetGear (Autorate)

Diameter: 6-7 hops.

Experiment Setting

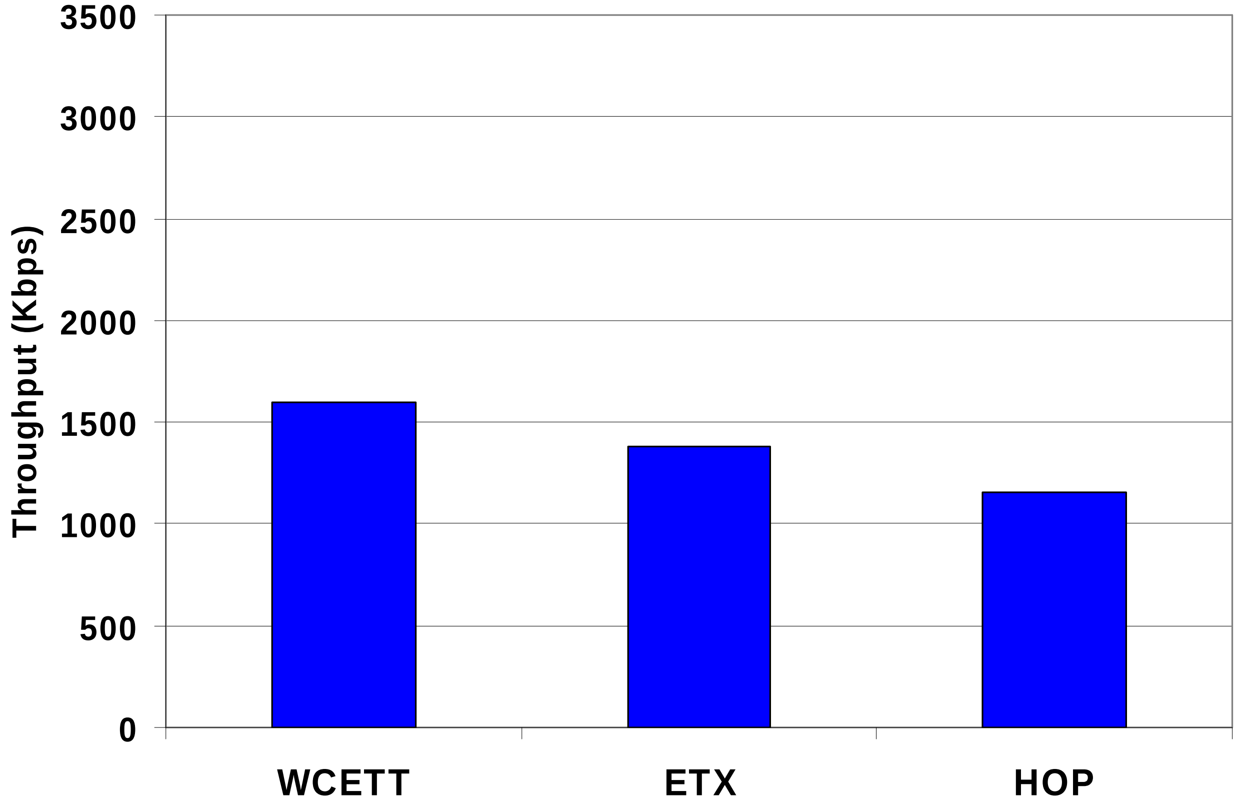
- 2-Minute TCP transfer between 100 randomly selected node pairs (Out of 23x22 = 506)
- Only one transfer active at a time
- Performance metric:
  - **Median throughput of 100 transfers**

Baseline (Single Radio)
NetGear on 802.11a (Channel 36), Proxim OFF

Two Radio
NetGear on 802.11a (Chan 36), Proxim on 802.11g (Chan 10)

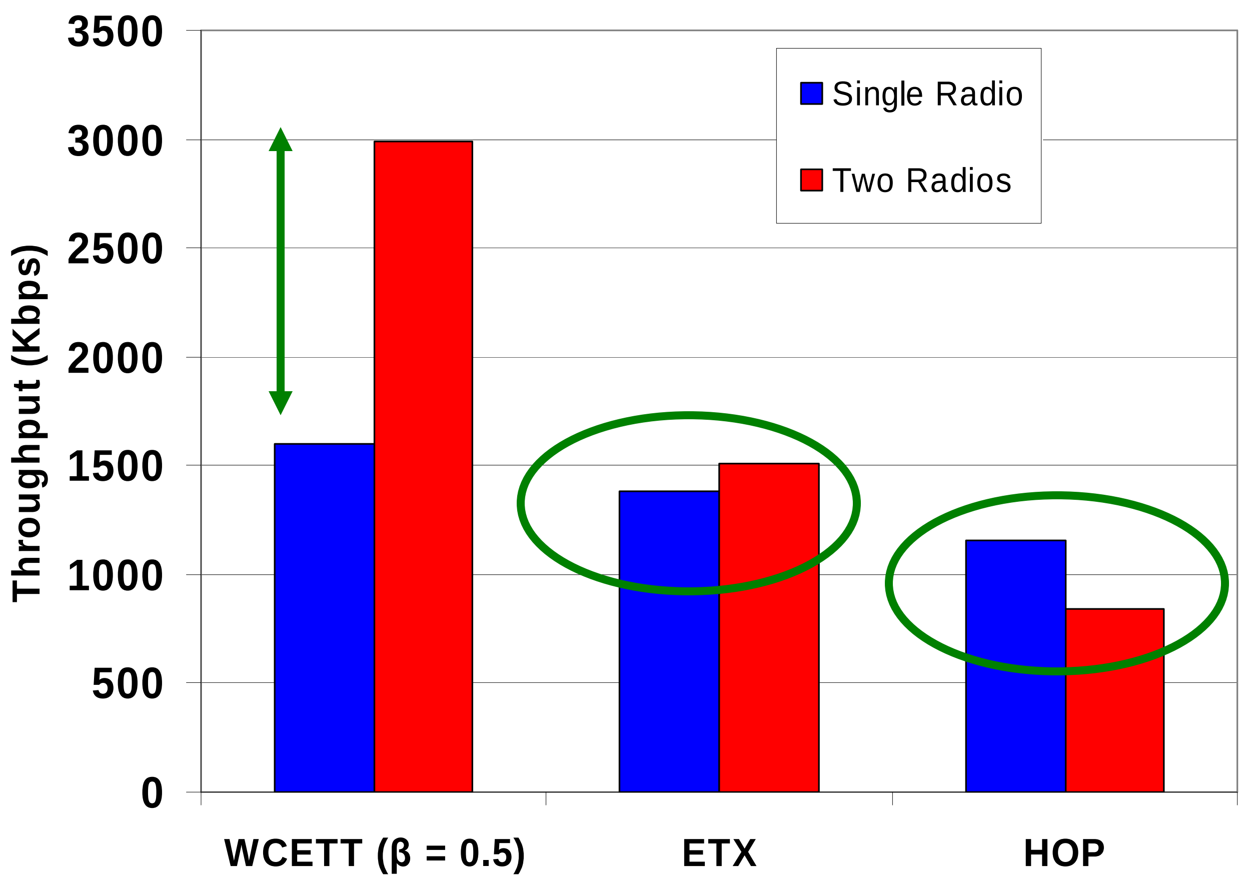
(802.11g radios have longer range, lower bandwidth)

Median Throughput (Baseline, single radio)



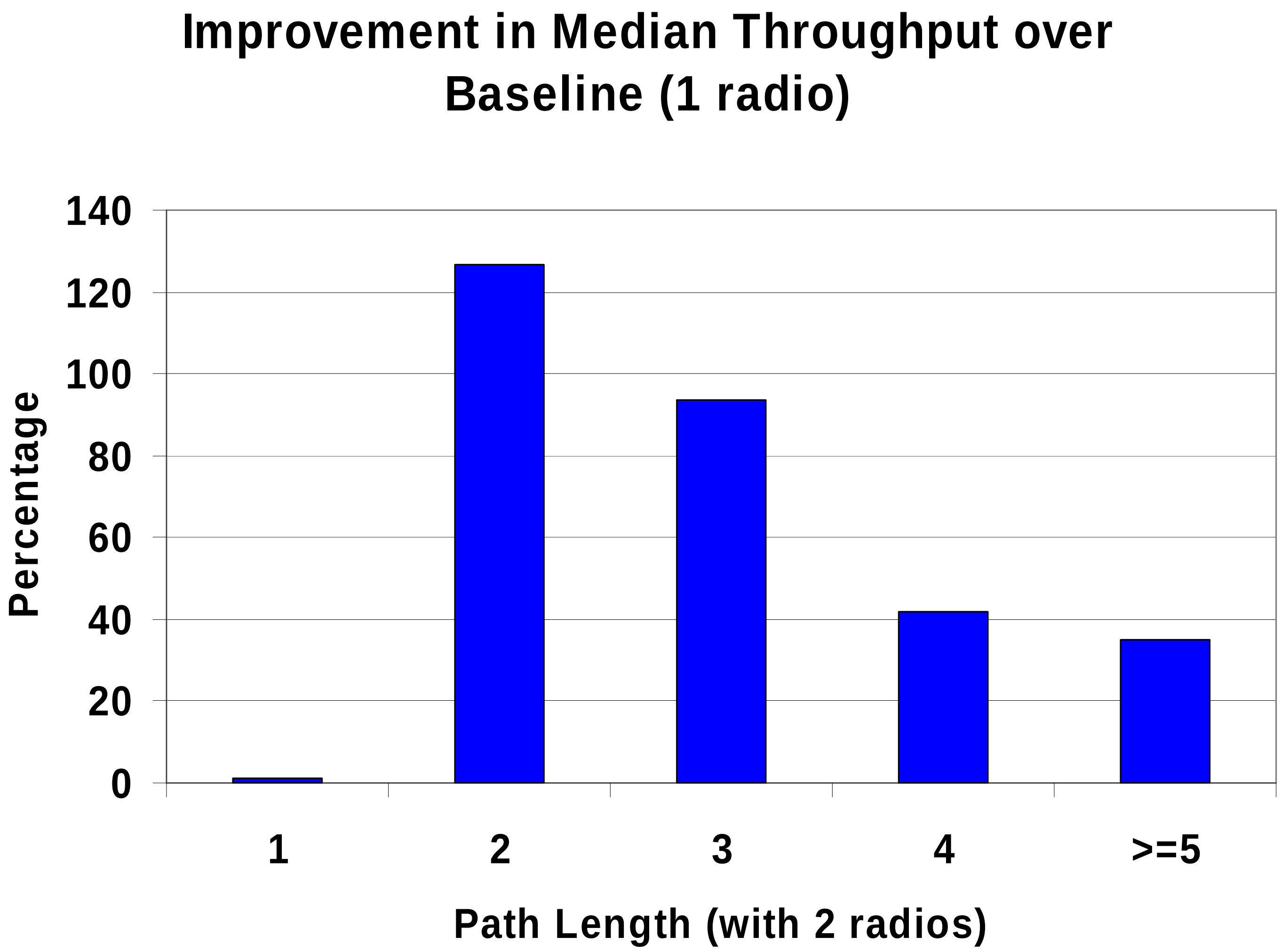
WCETT provides performance gain even with one radio.

Median Throughput (Two radios)



WCETT r Performance of HOP worsens with 2nd radio! r baseline

Do all paths benefit equally with WCETT?

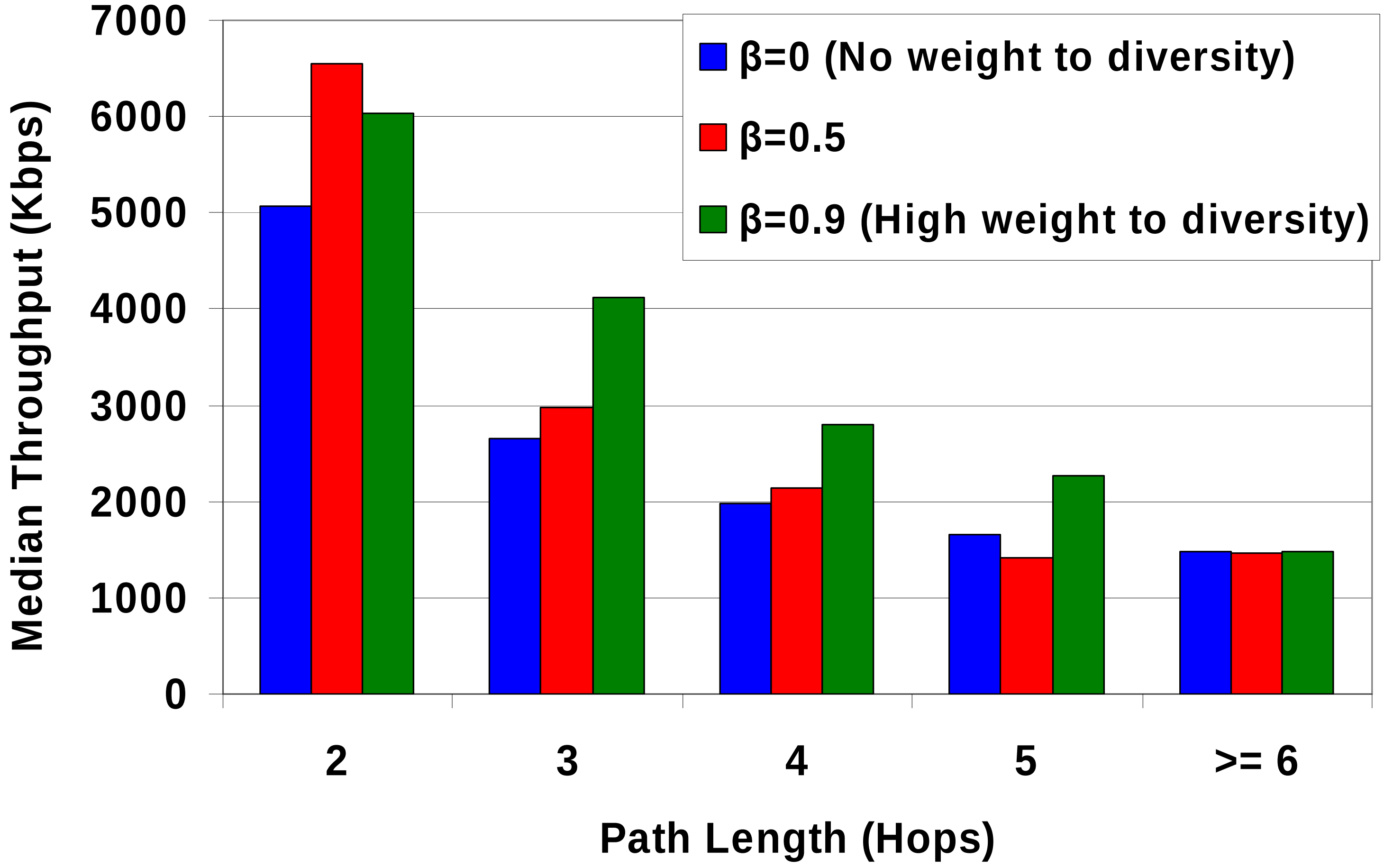


WCETT gains are more prominent for shorter paths

Announcements

- Mid-term next Monday
  - Open book/open notes
  - Anyone has class after 4:30?

Impact of β value

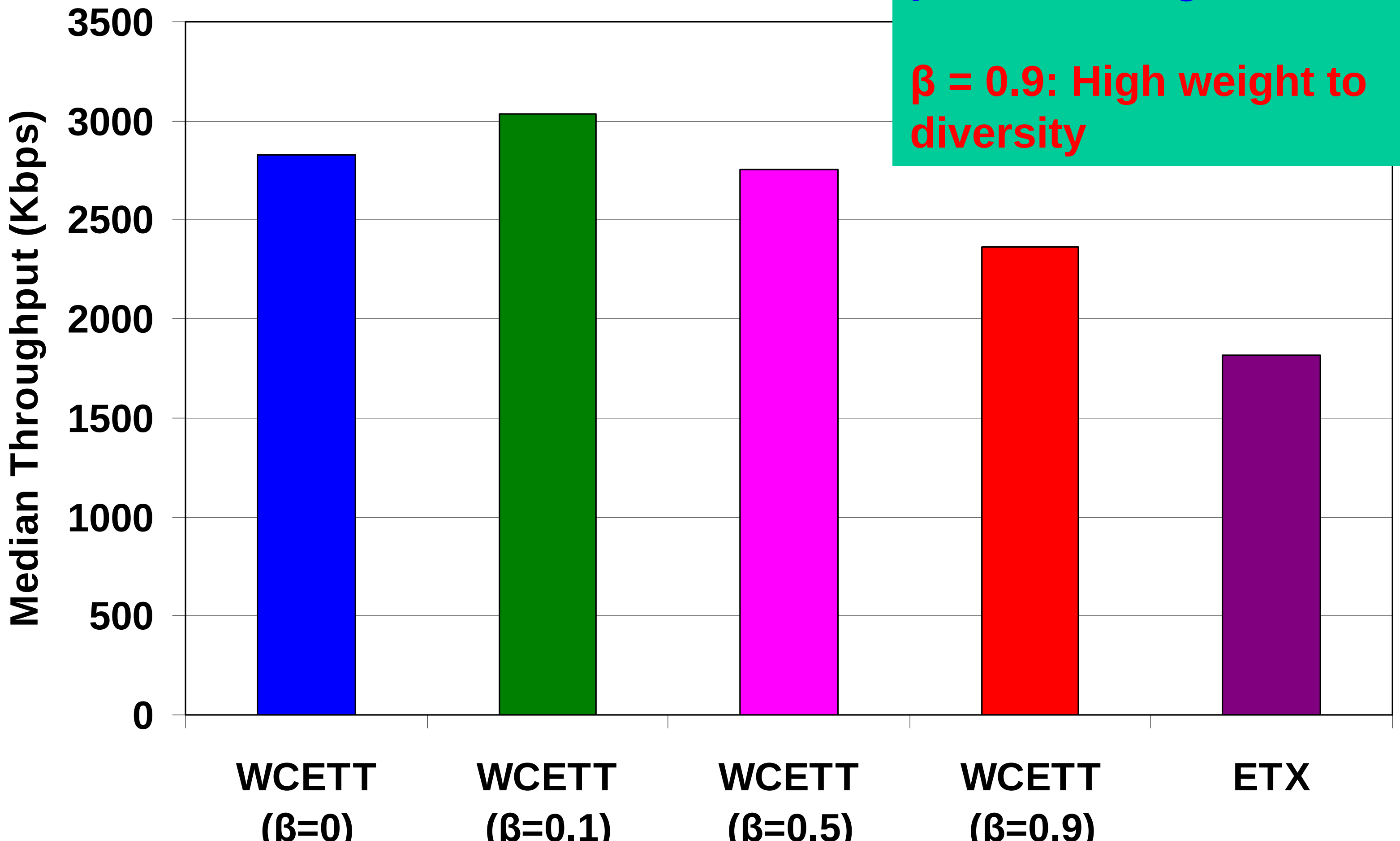


Channel diversity is important; especially for shorter paths

Performance of Two Simultaneous Flows

- 2-Minute TCP transfer between 100 randomly selected node pairs
- Two transfers active at a time
- Two radios: Netgear: 36-a, Proxim: 10-g
- Performance metric: 2 x Median throughput
- Repeat for ETX and WCETT (β = 0, 0.5, 0.9)

Two simultaneous flows



β = 0: No weight to diversity

β = 0.9: High weight to diversity

Throughput better for lower values of β

Conclusions

- Previously proposed routing metrics are inadequate in multi-radio scenario
- WCETT improves performance by judicious use of 2nd radio
  - Benefits are more prominent for shorter paths
- Optimal value of β depends on load
- Passive inference of loss rate and channel bandwidth

ExOR: Opportunistic Multi-Hop Routing for Wireless Networks

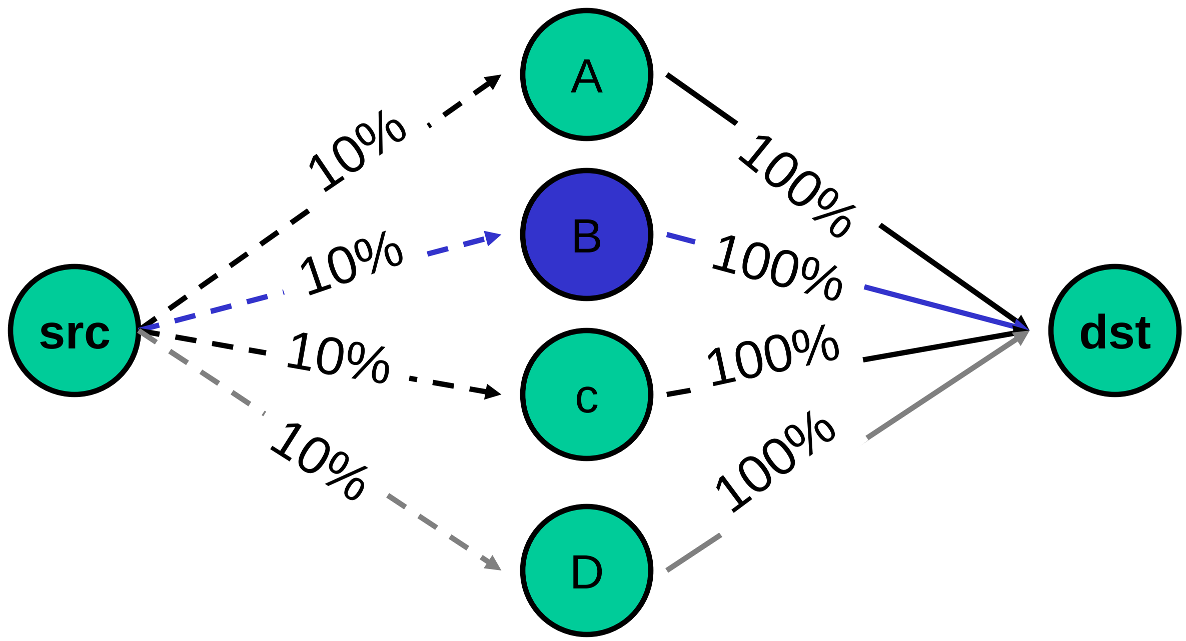
Sanjit Biswas and Robert Morris

MIT

Traditional Routing

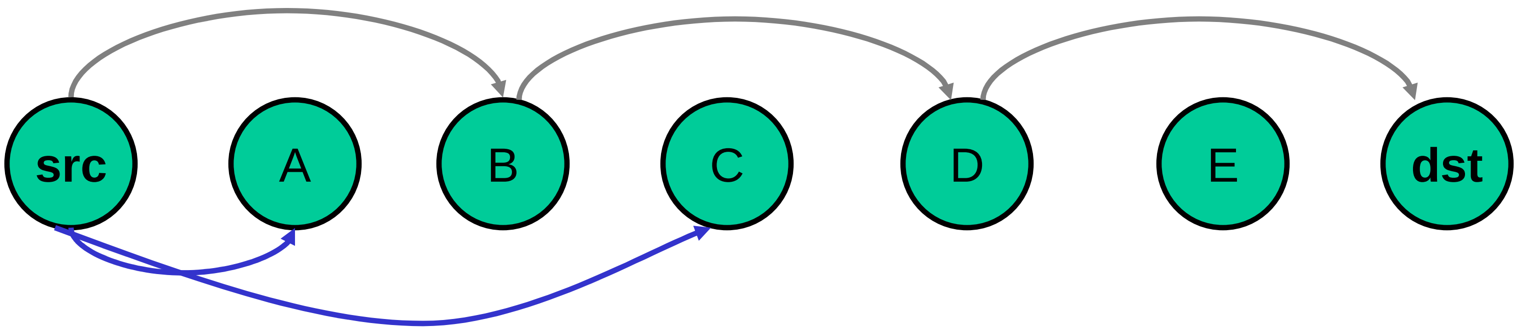
- Committed to a specific route before forwarding
- Problems: don't fully exploit path diversity
  - Unpredictable wireless medium
  - Intermittent connectivity
  - High mobility
  - Routing attacks

Motivating Scenario I



- *Assumes independent loss*
- Tradition routing has to follow one pre-commited route

Motivating Scenario II

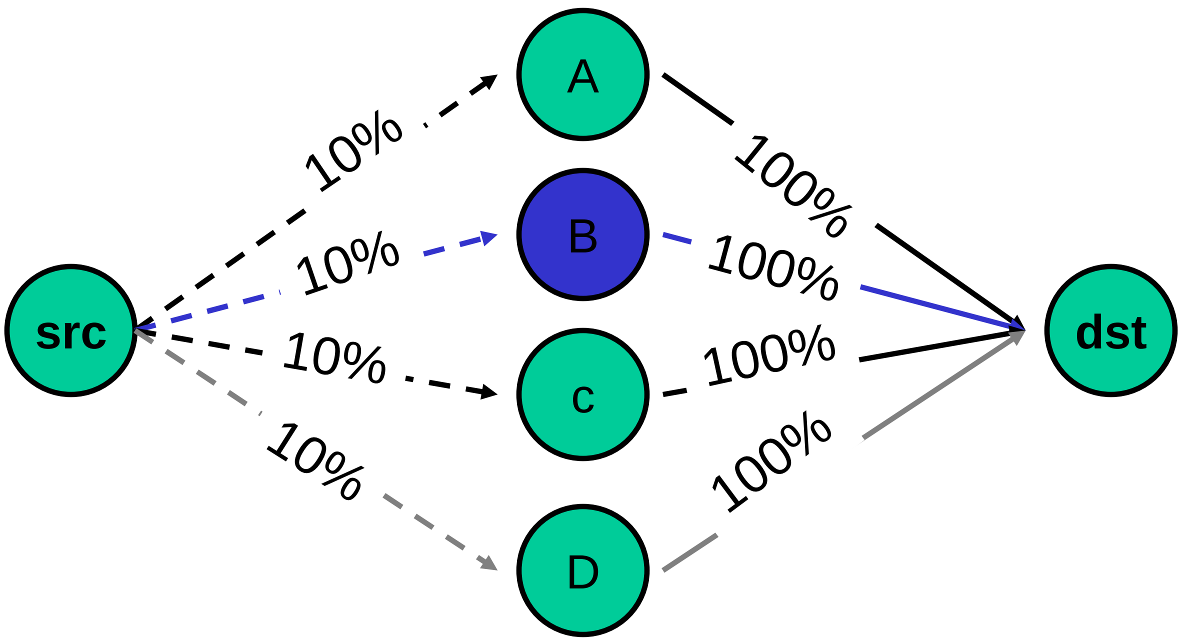


- Assumes loss rate increases gradually with distance
- Tradition routing has to make comprise between progress and loss rate

Opportunistic Routing (ExOR)

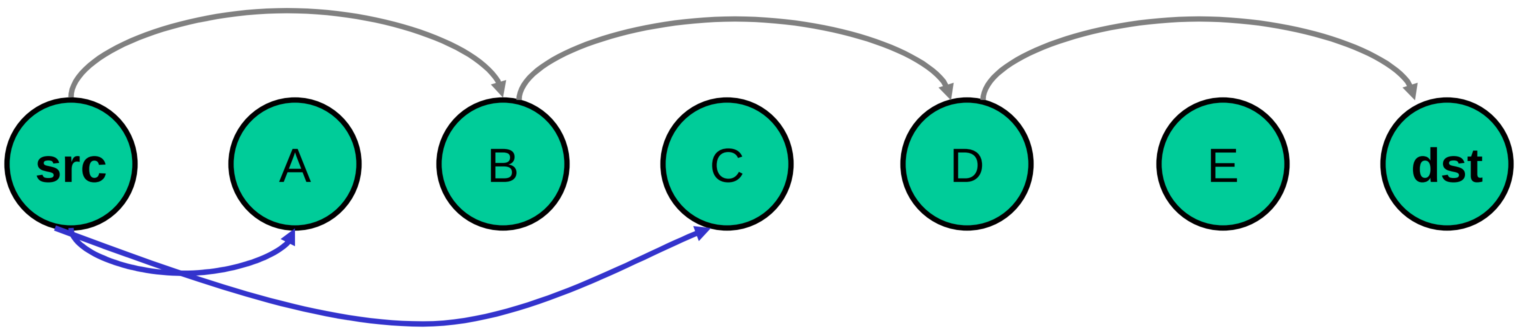
- Don't commit to a route before data forwarding
- Exploit wireless broadcast
- The source broadcasts the packet and then chooses a receiver to forward only after learning the set of nodes which actually received the packet.
- Goal: Among the nodes who receive the packet, the node closest to the destination should forward.

Motivating Scenario I



- Assumes independent loss
- Tradition routing has to follow one pre-commited route
- ExOR can take advantage of the lucky route

Motivating Scenario II



- Assumes loss rate increases gradually with distance
- Tradition routing has to make comprise between progress and loss rate
- ExOR can take advantages of transmissions that reach unexpectedly short or unexpectedly far.

Main Challenge

- How to select the node that is closest to the destination and received the packet to forward it with low overhead?

Issues to Address

- What we want: an effective protocol with low overhead
- How often should ExOR run?
  - Per packet is expensive
  - Use batches
- Who should participate the forwarding?
  - Too many participants cause large overhead
- When should each participant forward?
  - Avoid simultaneous transmission
- What should each participant forward?
  - Avoid duplicate transmission

Who should participate?

- The source chooses the participants (forwarder list) using ETX-like metric.
  - Only consider forward delivery rate
  - The source runs a simulation and selects only the nodes which transmit at least 10% of the total transmission in a batch.

- A background process collects ETX information via periodic link-state flooding.

When should each participant forward?

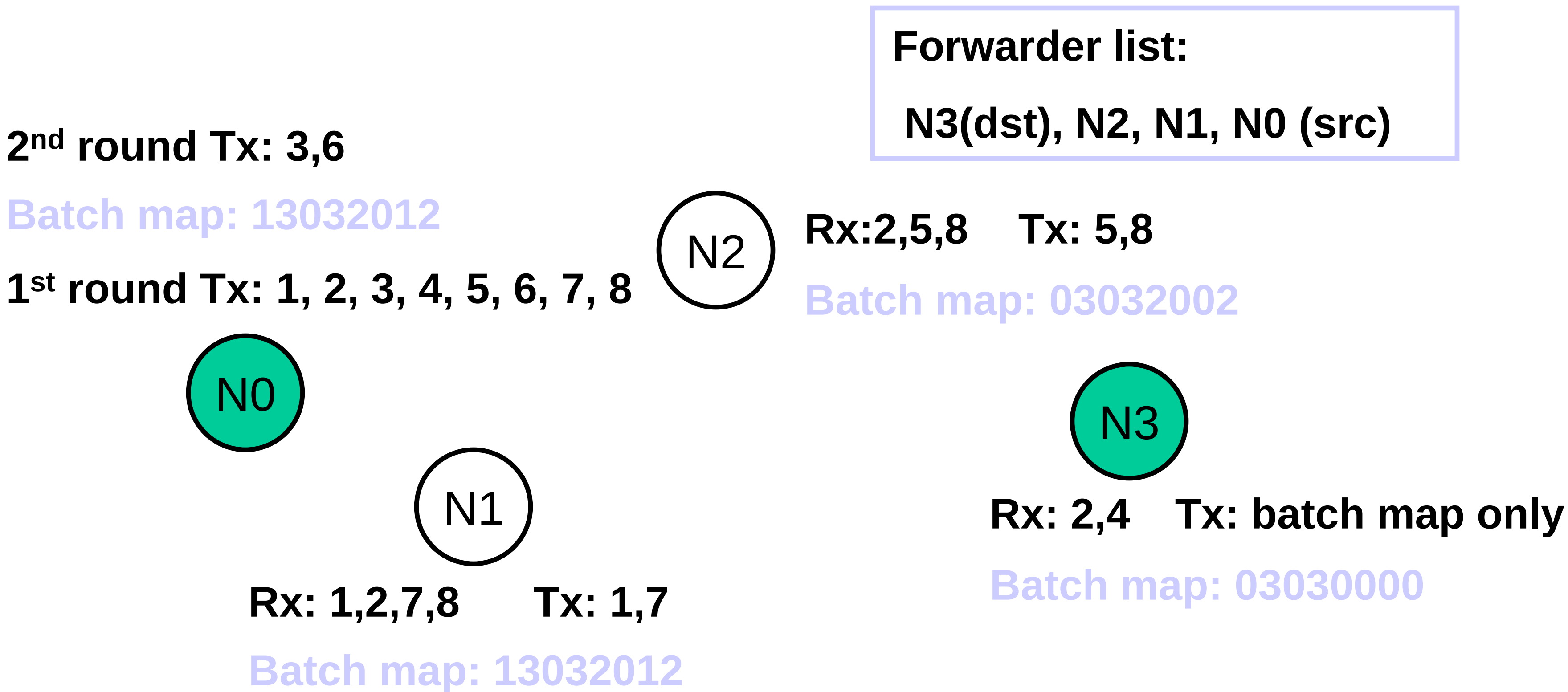
- Forwarders are prioritized by ETX-like metric to the destination
- The highest priority forwarder transmits when the batch ends
- The remaining forwarders transmit in prioritized order
- Question: How does each forwarder know it is its turn to transmit?
  - Assume other higher priority nodes send for five packet durations if not hearing anything from them

What should each participant forward?

- Packets it receives yet not received by higher priority forwarders
- Question: How does a node know the set of packets received by higher priority nodes?
  - Using batch map

Batch map

- Batch map indicates, for each packet in a batch, the highest-priority node known to have received a copy of that packet.

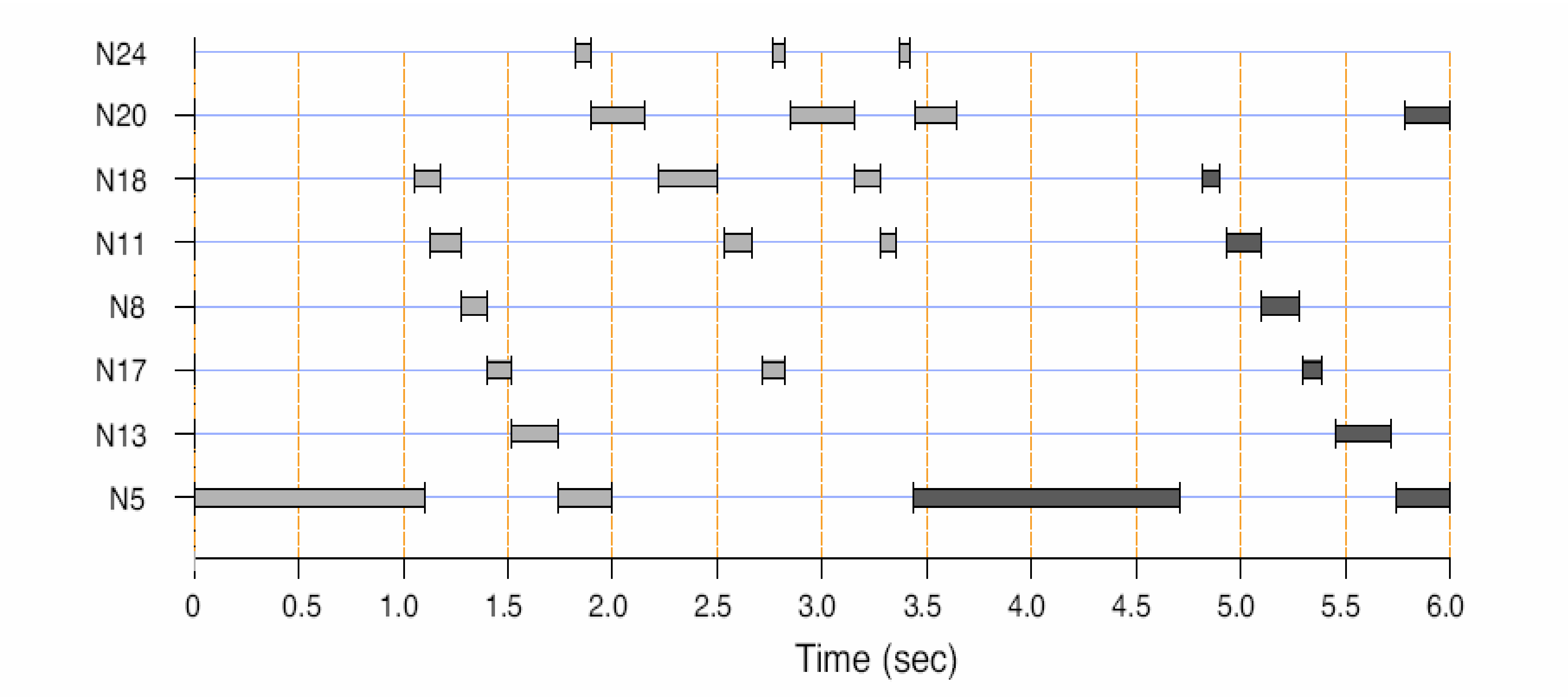


Completion

- A nodes stops sending the remaining packets in the batch if its batch map indicates over 90% of this batch has been received by higher priority nodes.

- The remaining packets transferred with traditional routing.

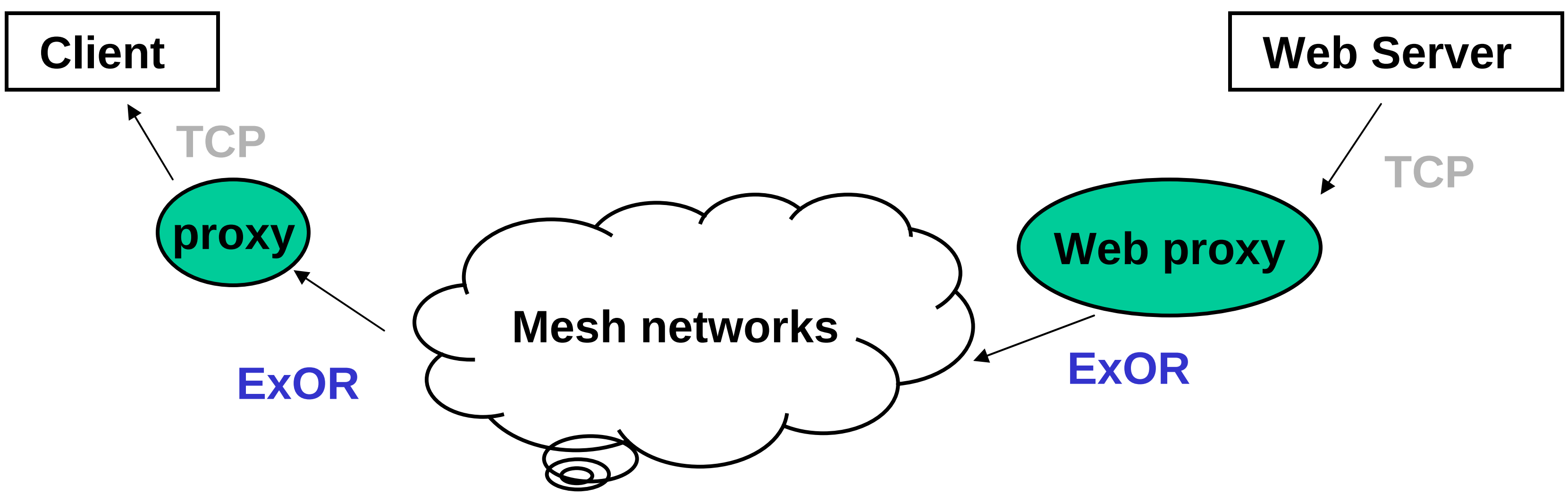
Example



Forwarder list: N24(dst), N20, N18, N11, N8, N17, N13, N5(src)

ExOR with TCP

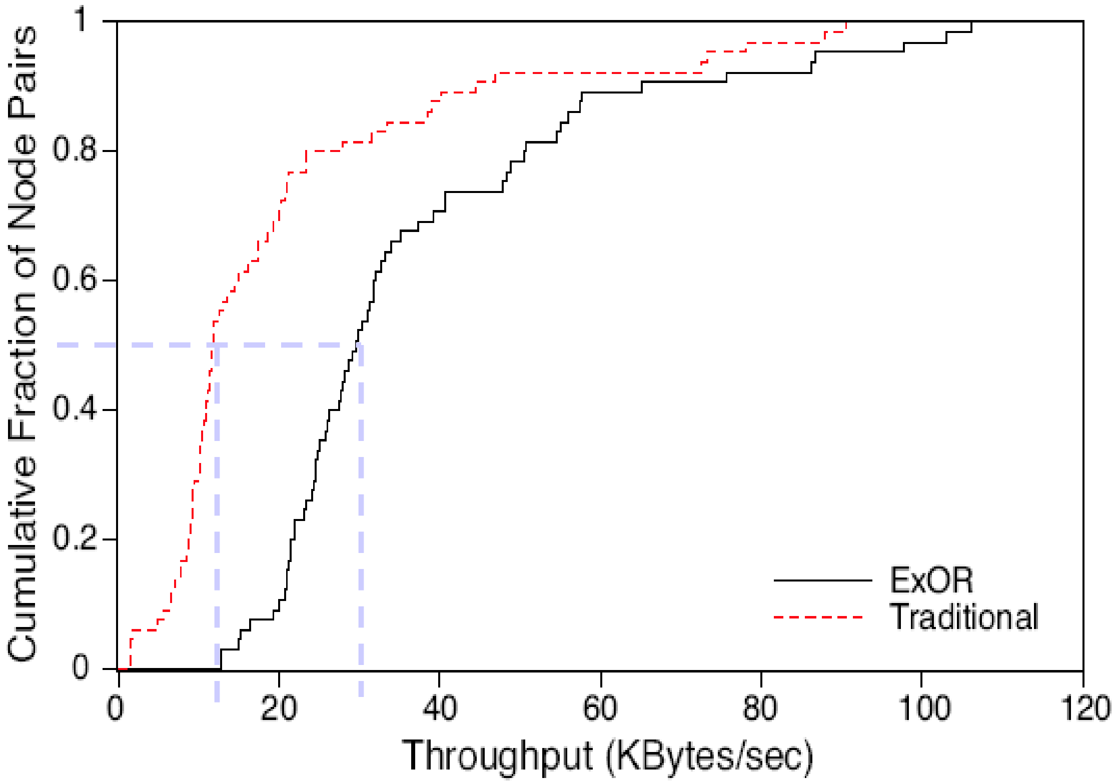
- ExOR creates lots of packet reordering
- ExOR increase end-to-end delay
- Solution: Split web proxy



Evaluation – Network Description

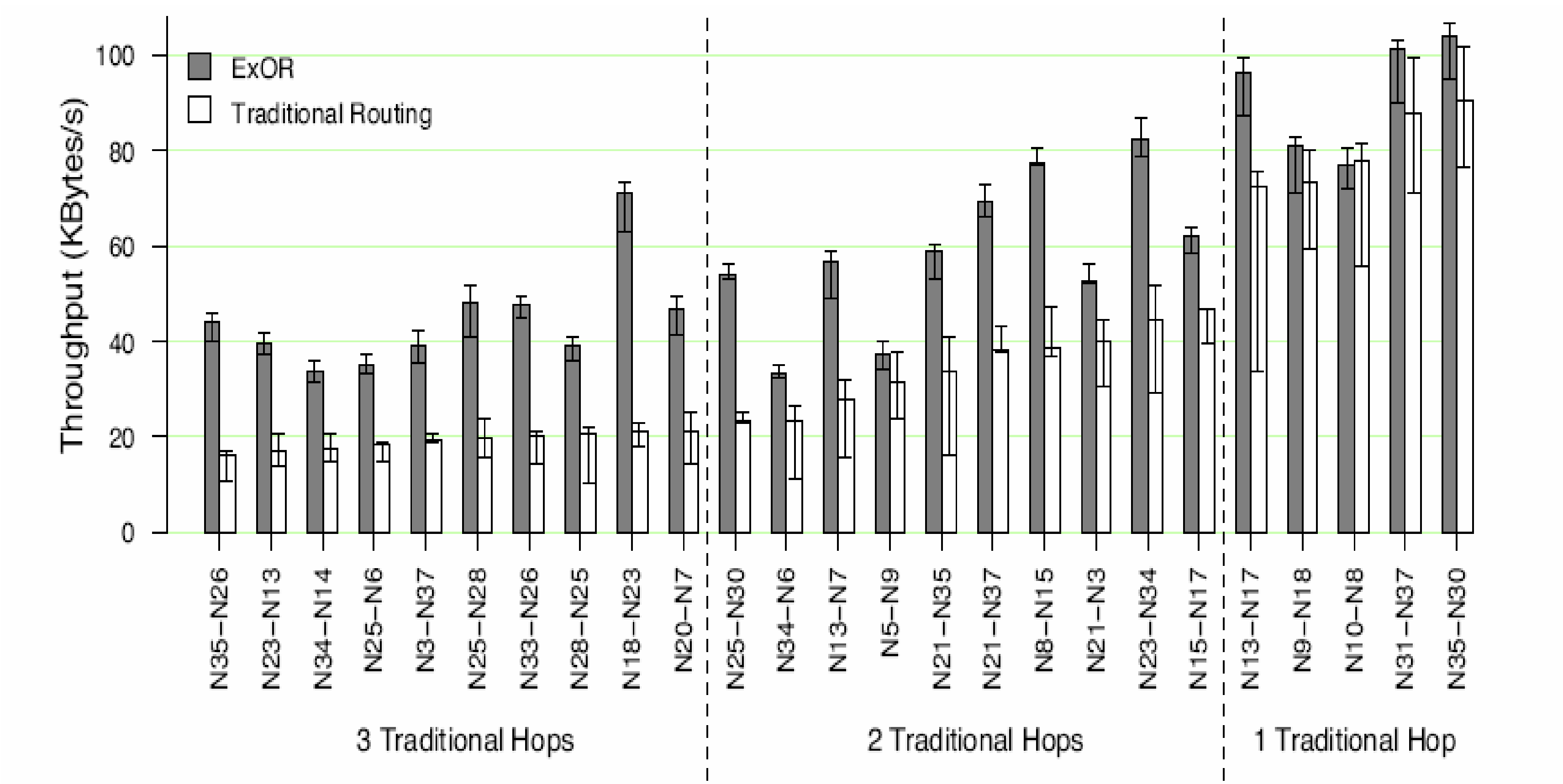
- Performed on Roofnet, an outdoor roof-top 802.11 networks
- 38 nodes distributed over six square kilometers
- 65 Node pairs
- 1.0MByte file transfer
- 1 Mbit/s 802.11 bit rate
- 1 KByte packets

Evaluation - Throughput

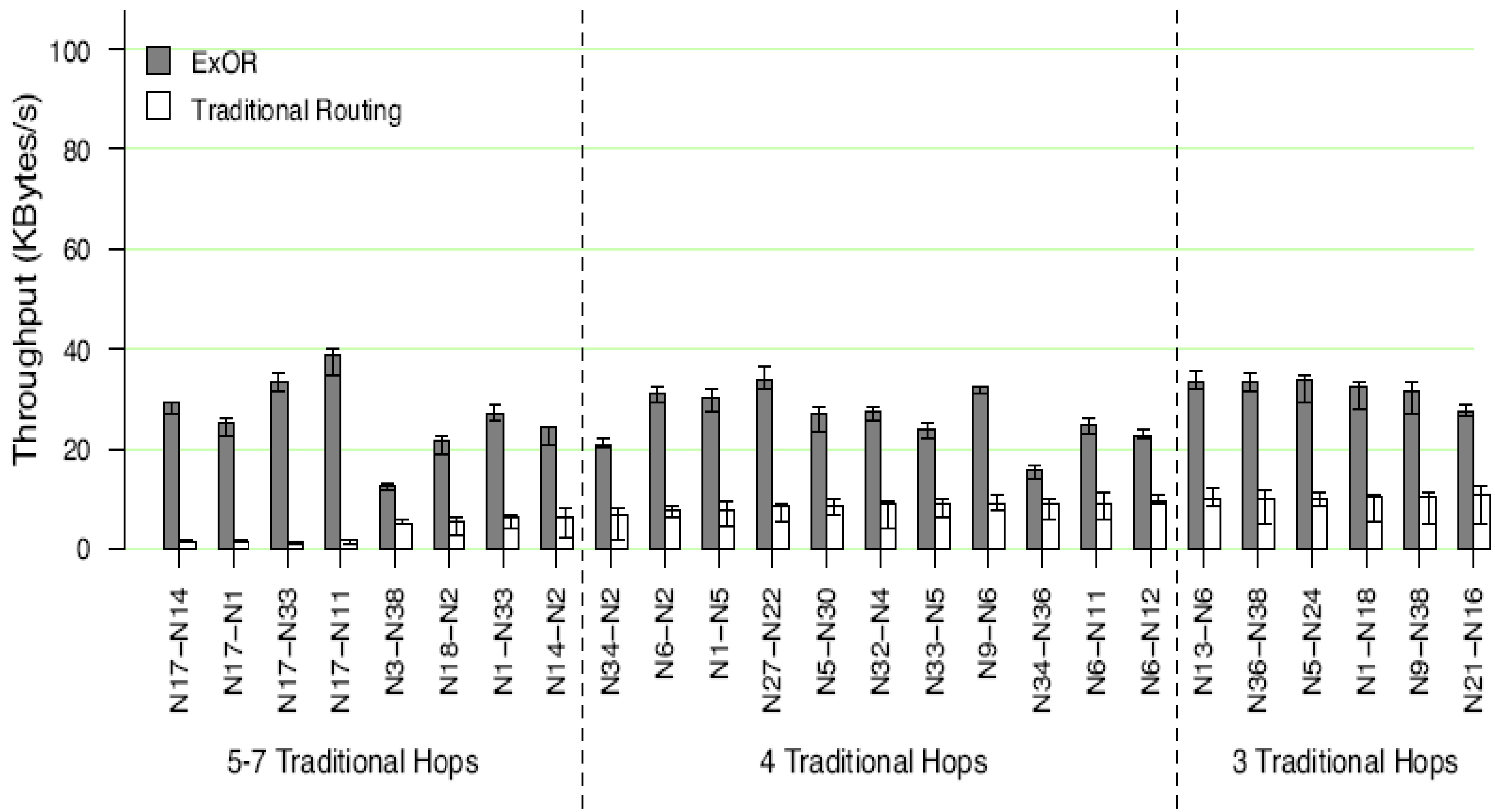


Median throughputs: 2X overall improvement

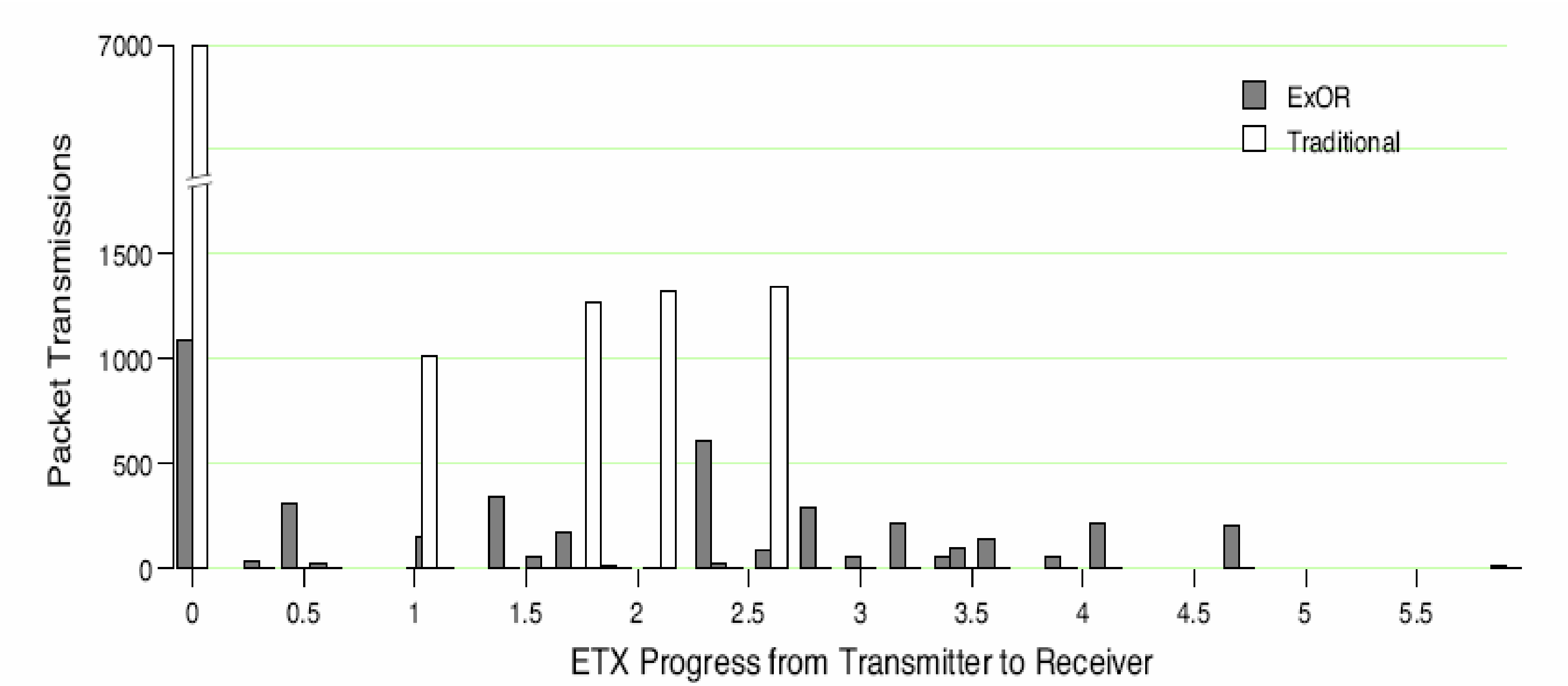
Evaluation - 25 Highest Throughput Pairs



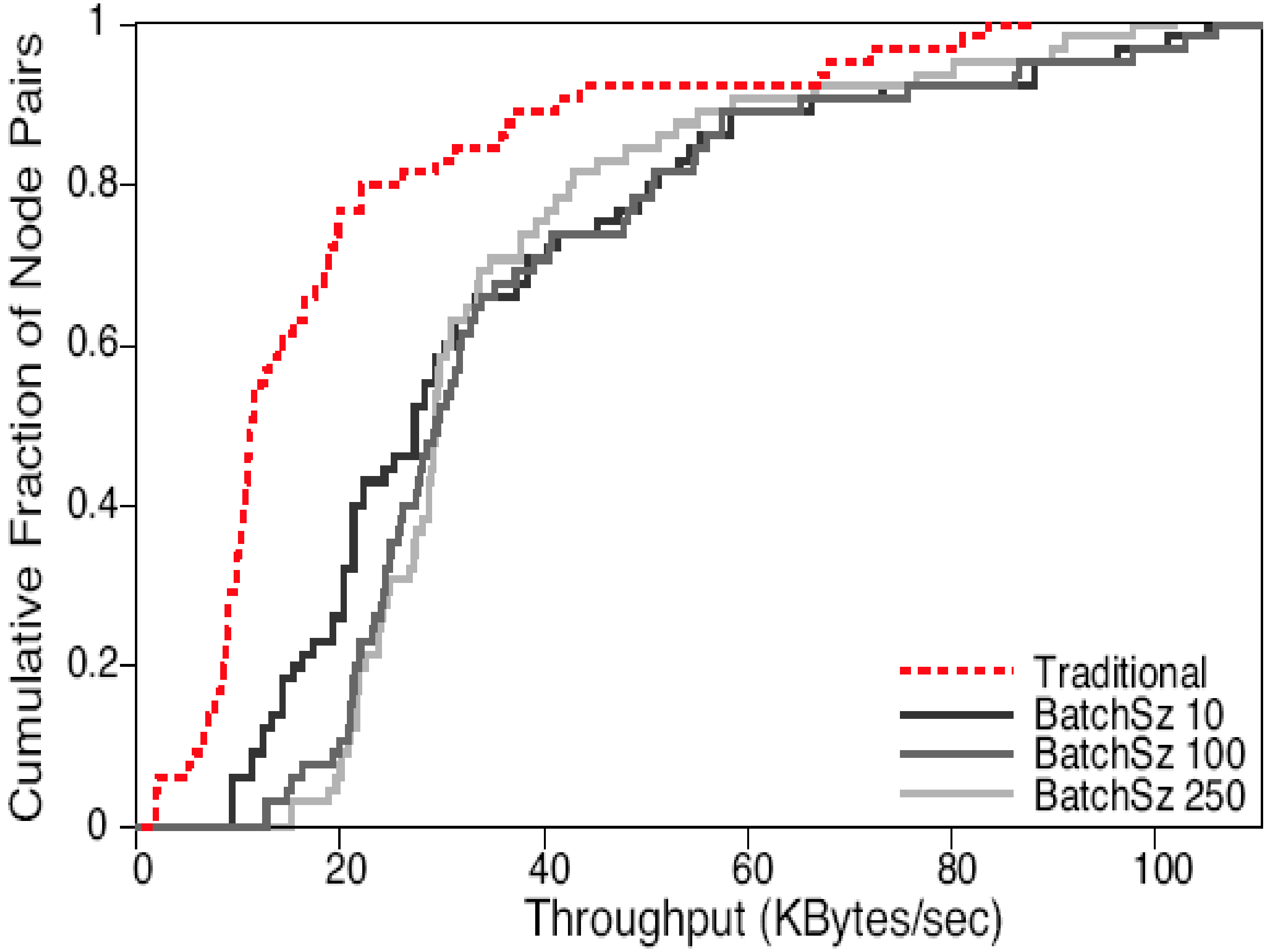
Evaluation - 25 Lowest Throughput Pairs



Evaluation – Distance per Transmission



Evaluation – Batch Size



Comments?

Comments

- Pros
  - Takes advantage of the probabilistic reception to increase the throughput
  - Does not require changes in the MAC layer
  - Can cope well with unreliable wireless medium and mobility
- Cons
  - Do not support multiple flows or TCP traffic
  - Hard to scale to a large network
  - Overhead in packet header (batch info)
  - Batches increase delay

Announcements

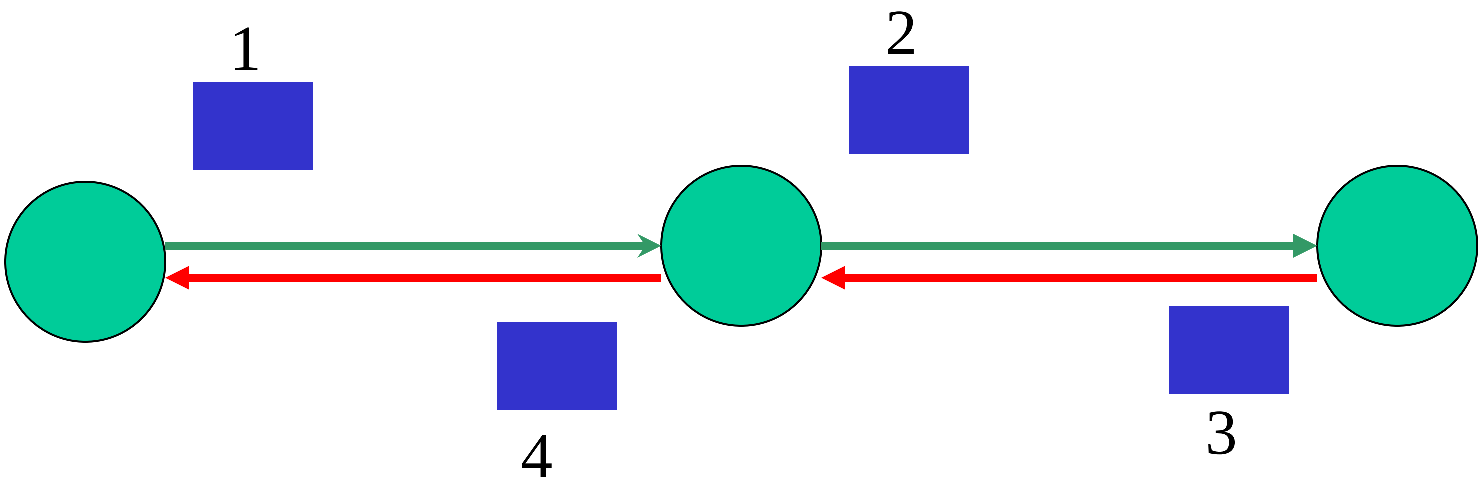
- Mid-term next Monday @ WRW (W. R. Woolrich Lab) 102
- Mid-term
  - Basic material: physical layer, 802.11, Routing, Mobile IP, TCP ($\geq$ 30%)
  - Advanced topics: papers
- Office hour
  - After class today
  - Friday 3:30 – 4:30pm

XORs in the Air: Practical Wireless Network Coding

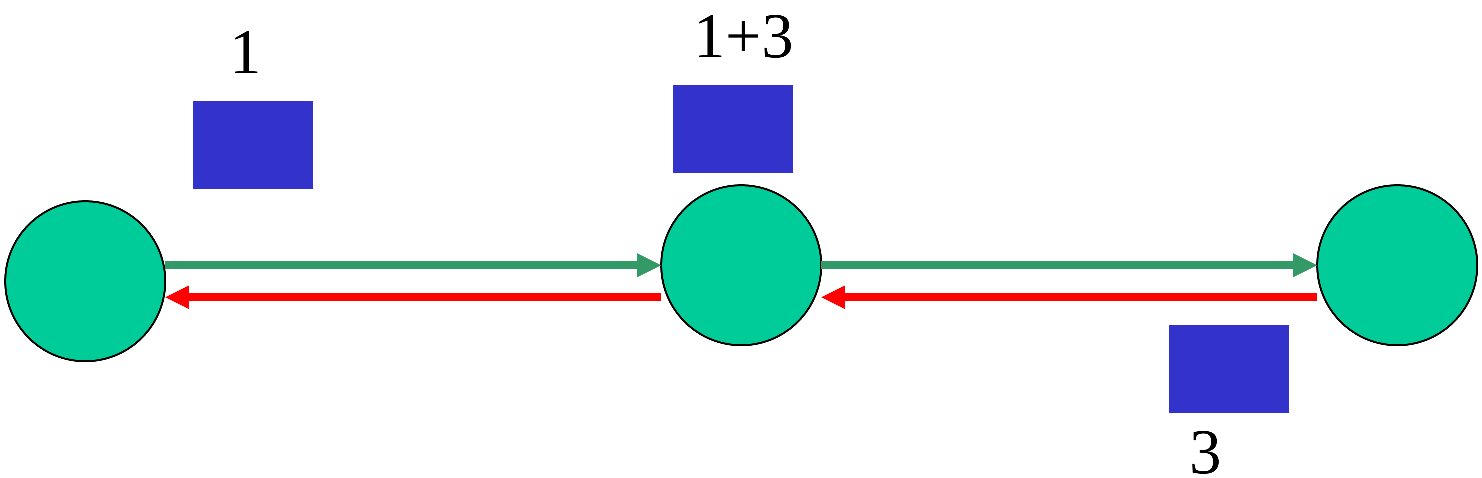
S. Katti, H. Rahul, W. Hu, D. Katabi,
M. Medard, J. Crowcroft

MIT & University of Cambridge

Can we use 3 transmissions to send traffic?



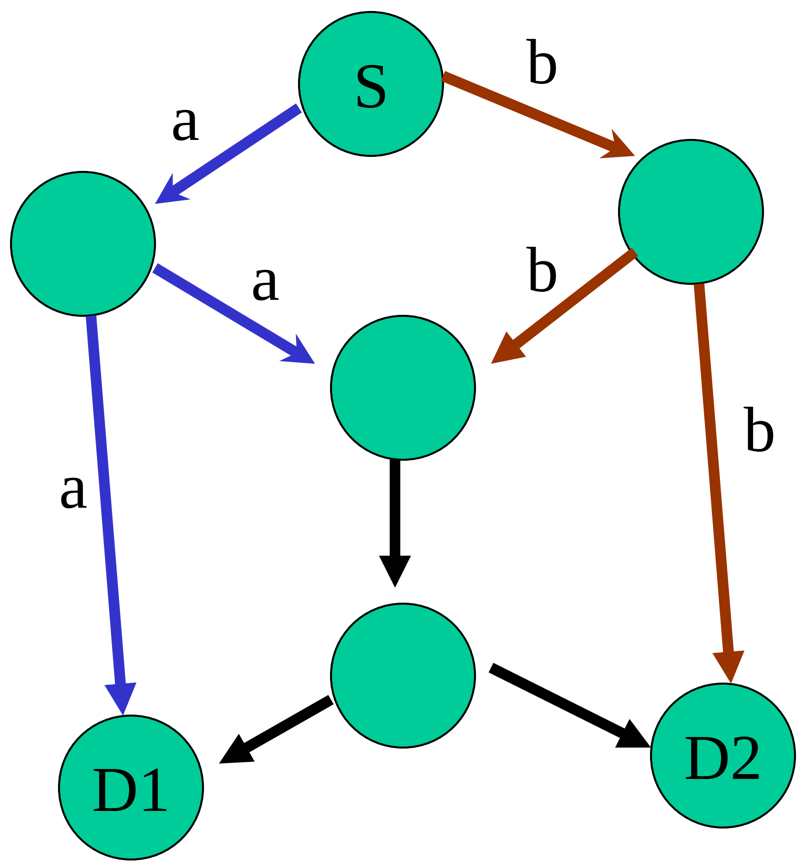
Can we use 3 transmissions to send traffic?



Network Coding

- Three types
  - Store and forward
  - Source coding (FEC, compression)
  - Network coding
- Goal: increase the amount of information that is transported

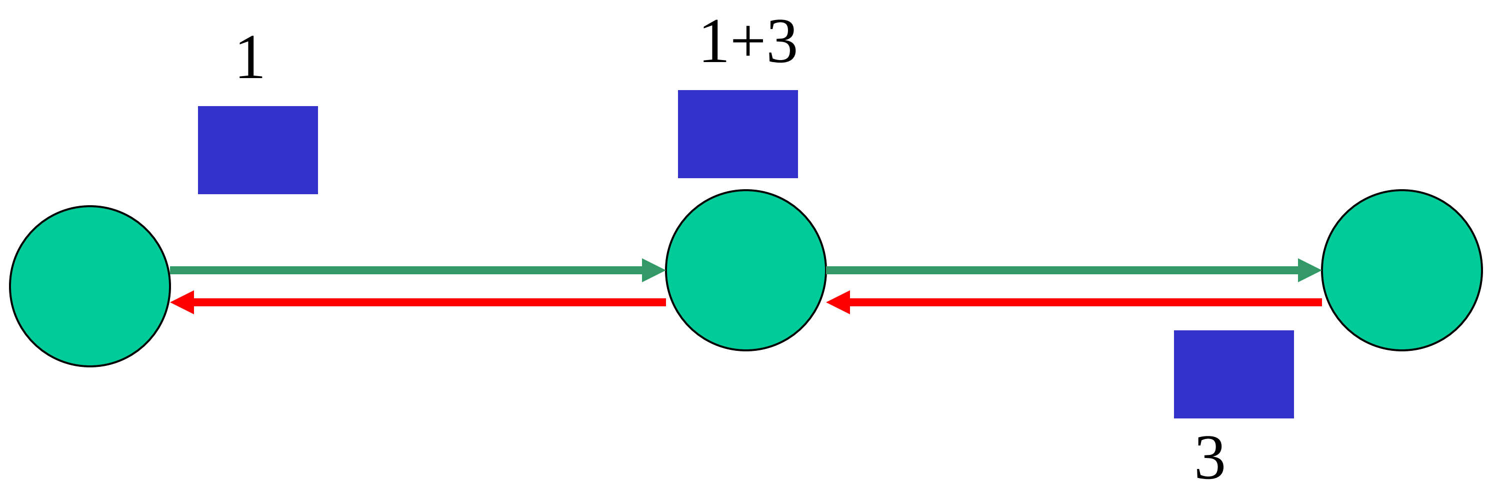
Network Coding in Wireline Networks



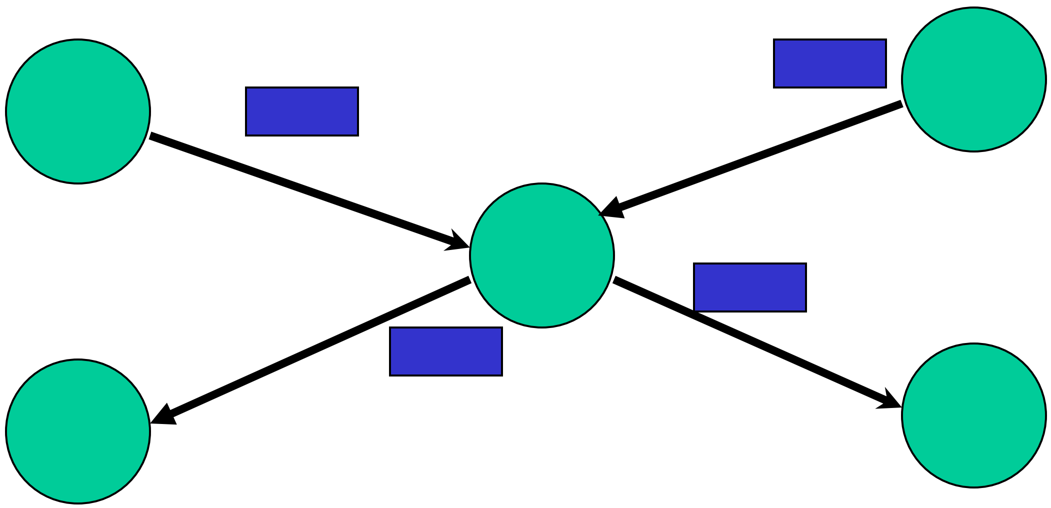
Networking coding is beneficial for multicast in wireline networks.
Networking coding has little benefit for unicast in wireline networks.

Coding Gain

- Coding gain = 4/3

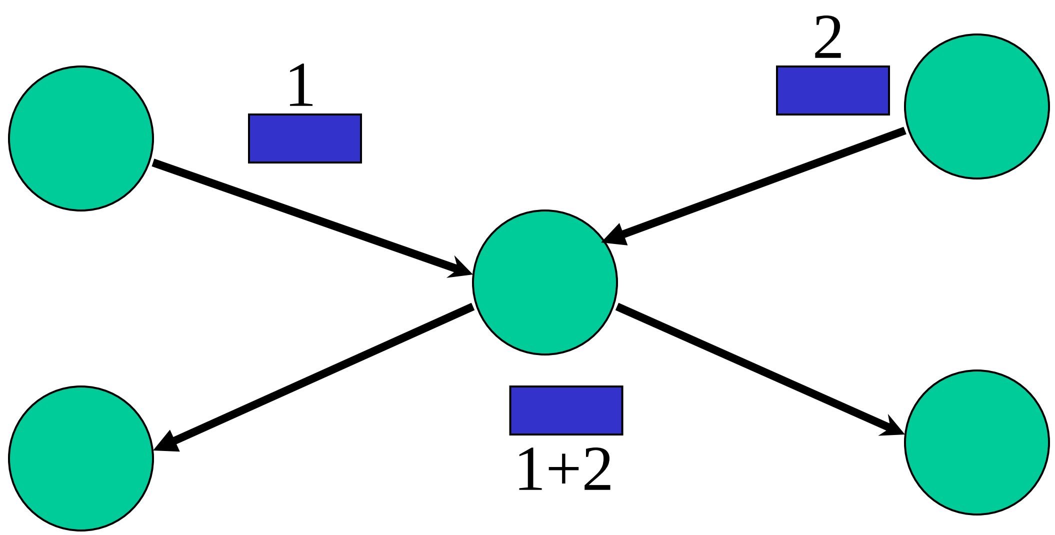


Coding Gain: more examples



Without opportunistic listening, no coding gain.

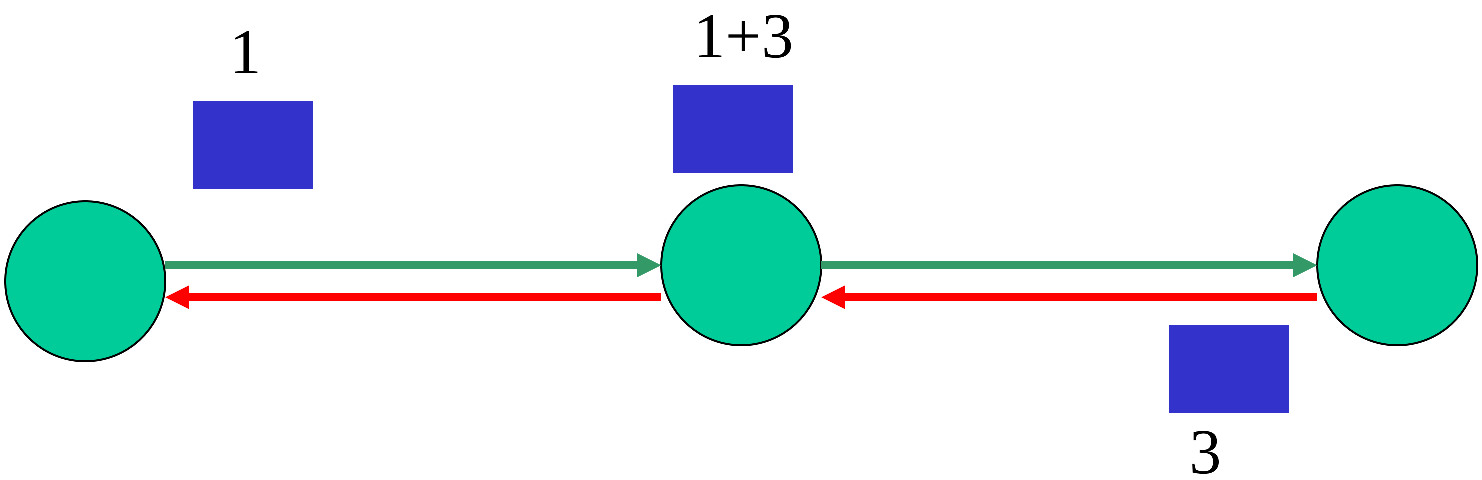
Coding Gain: more examples



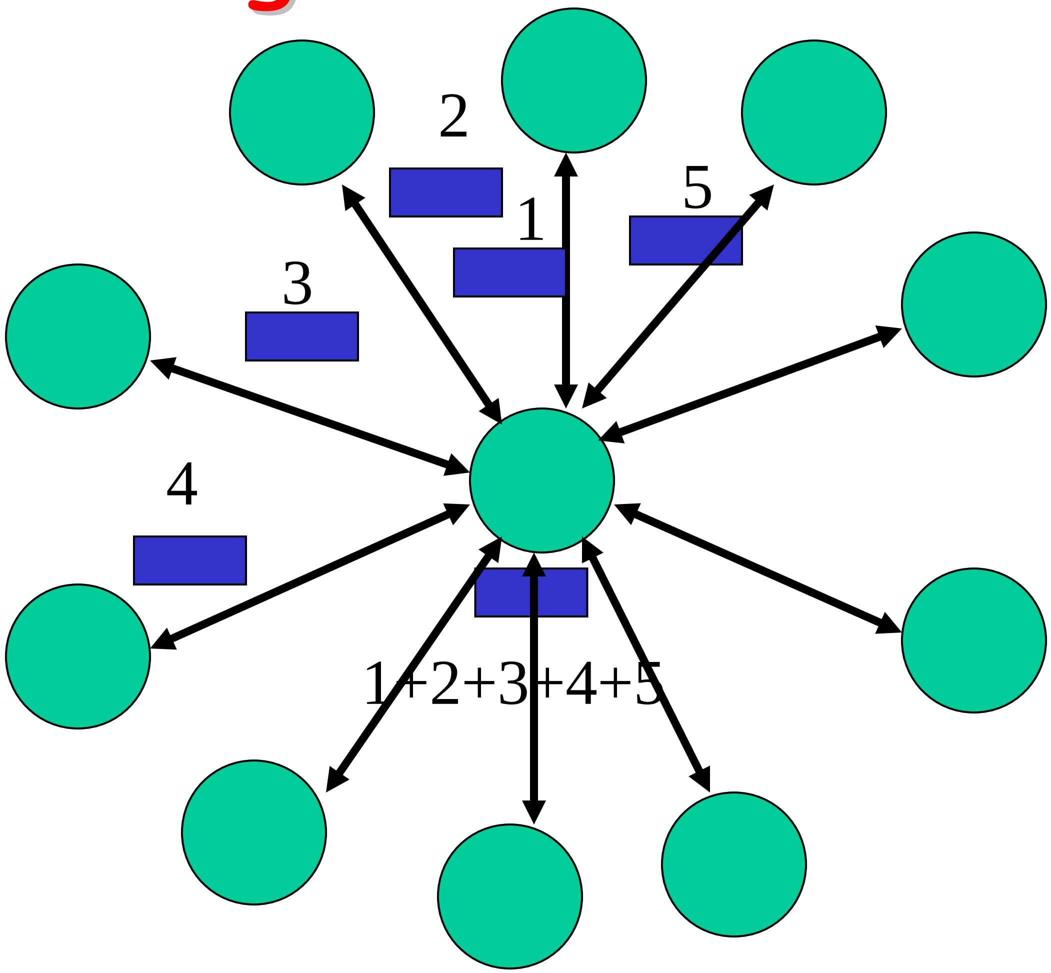
With opportunistic listening, coding gain = 4/3.

Throughput Improvement

- UDP throughput improvement ~ a factor 2 > 4/3 coding gain



Coding Gain: more examples



With opportunistic listening, coding gain=(1+N)/2N.
With opportunistic listening, coding gain + MAC gain → ∞

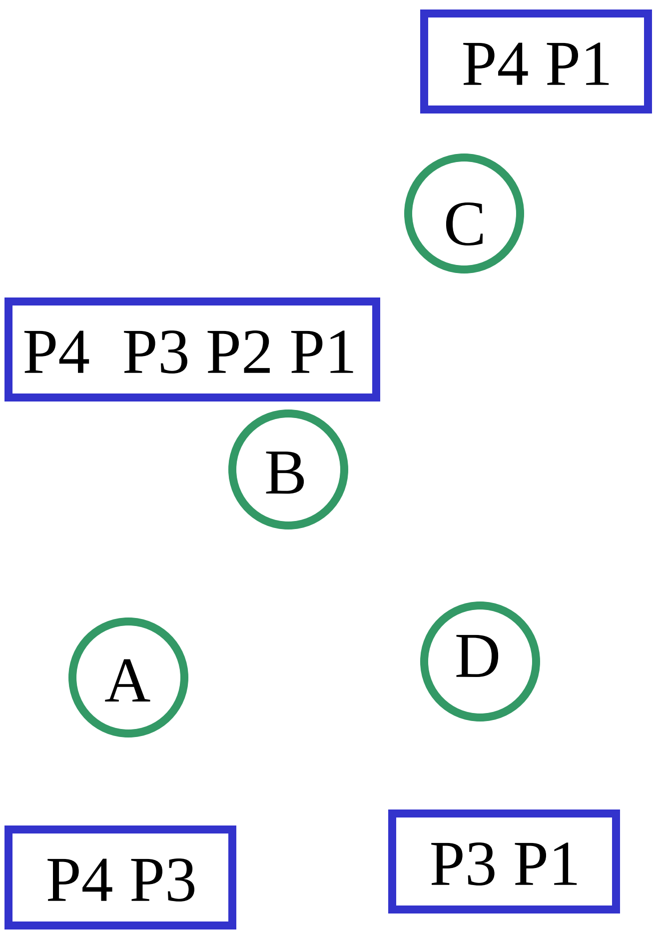
Properties

- In the absence of opportunistic listening, COPE's maximum coding gain is 2 and it is achievable.
- In the absence of opportunistic listening, COPE's maximum coding gain + MAC gain is 2 and it is achievable.
- In the presence of opportunistic listening, COPE's maximum coding+MAC gain is unbounded.

COPE Overview

- Layer between IP and MAC
- Techniques
  - Opportunistic listening (promiscuous mode)
  - Opportunistic coding
  - Learning neighbor state

Opportunistic Coding



B's queue	Next hop
P1	A
P2	C
P3	C
P4	D

Coding	Is it good?
P1+P2	Bad (only C can decode)
P1+P3	Better coding (Both A and C can decode)
P1+P3+P4	Best coding (A, C, D can decode)

Packet Coding Algorithm

- When to send?
  - Option 1: delay packets till enough packets to code with
  - Option 2: never delaying packets -- when there's a transmission opportunity, send packet right away
- Which packets to use for XOR?
  - Prefer XOR-ing packets of similar lengths
  - Never code together packets headed to the same next hop
  - Limit packet re-ordering
  - XORing a packet as long as all its nexthops can decode it with a high enough probability

Packet Decoding

- Where to decode?
  - Decode at each intermediate hop
- How to decode?
  - Upon receiving a packet encoded with n native packets
    - find n-1 native packets from its queue
    - XOR these n-1 native packets with the received packet to extract the new packet

Pseudo Broadcast

- Each packet is destined for multiple nexthops
  - Broadcast
    - Natural for multiple receivers
  - Unicast
    - Cheap ACK wo/ contention
    - Link layer retransmissions
    - More effective backoff
    - Take advantage of multiple rates
  - Unicast + hop-by-hop ACKs/retx
    - Unicast alone is insufficient

Prevent Packet Reordering

- Packet reordering due to async acks degrade TCP performance
- Ordering agent
  - Deliver in-sequence packets immediately
  - Order the packets until the gap in seq. no is filled or timer expires

Summary of Results

- Improve UDP throughput by a factor of 3-4
- Improve TCP by
  - wo/ hidden terminal: up to 38% improvement
  - w/ hidden terminal and high loss: little improvement
- Improvement is largest when uplink to downlink has similar traffic

Reasons for Lower Improvement in TCP

- COPE introduces packet re-ordering
- Router queue is small → smaller coding opportunity
  - TCP congestion window does not sufficiently open up due to wireless losses
- TCP doesn't provide fair allocation across different flows

Lessons

- Both COPE and ExOR dispose the point-to-point wireless link abstraction
  - Leverage broadcast nature of wireless medium to its advantage
- Network coding has a great potential in wireless network