XORs in The Air: Practical Wireless Network Coding

SIGCOMM 2006

Presented by Chung-Shih Tang

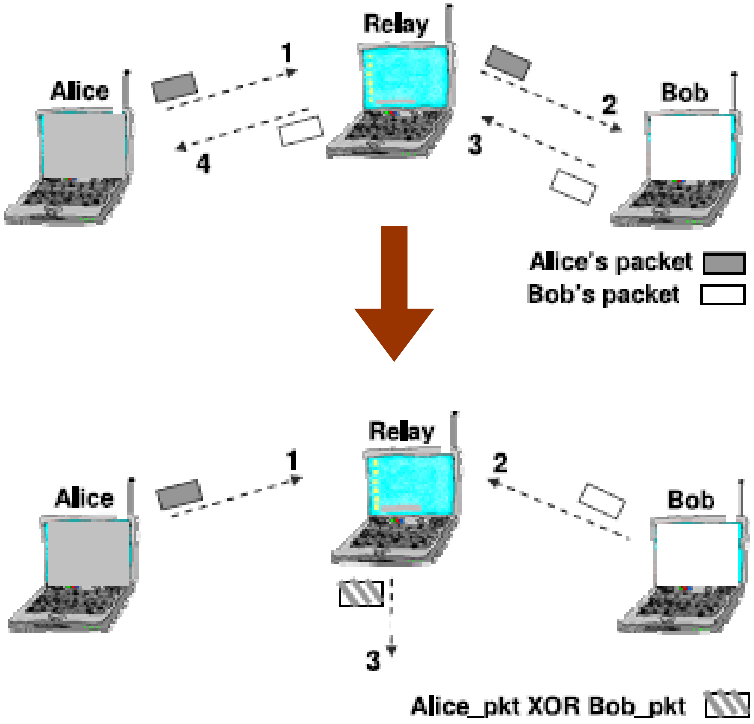
11/29/2006

Outline

- Introduction
- COPE Overview
- Coding Gain
- Design of COPE
- Implementation Details
- Experimental Results
- Discussion and Conclusion

Introduction

- This paper presents COPE, a new forwarding architecture that substantially improves the throughput of wireless networks
- COPE's design is based on
  - COPE disposes of the point-to-point abstraction and embraces the broadcast nature of wireless channel
  - COPE employs network coding to maximize transmission throughput



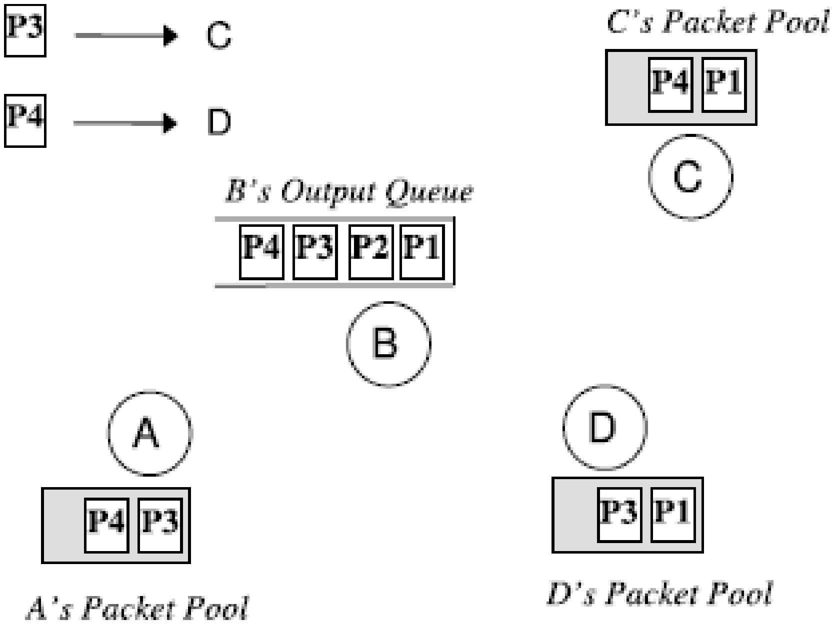
COPE Overview

- Opportunistic Listening
  - Snoop on all communications over the wireless medium and store the overheard packets for a limited time *T* (default *T* = 0.5*s*)
  - Each node broadcasts *reception reports* to tell its neighbors which packets it has stored
- Opportunistic Coding
  - Maximize the number of native packets delivered in a single transmission, while ensuring that each intended nexthop has enough information to decode its native packet
- Learning Neighbor State
  - A node cannot rely solely on reception reports, and may need to guess whether a neighbor has a particular packet

Opportunistic Coding

1. **Node B has 4 packets in its output queue**
2. **Nexthop of each packet in B's queue**
3. **Node B chooses the best coding options based on neighbor information**

Packets in B's Queue	Next Hop
P1	A
P2	C
P3	C
P4	D



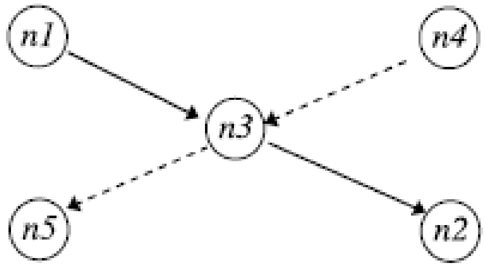
Coding Option	Is it good?
P1 + P2	Bad Coding (C can decode but A can't)
P1 + P3	Better Coding (Both A and C can decode)
P1 + P3 + P4	Best Coding (Nodes A, C, and D can decode)

Coding Gain

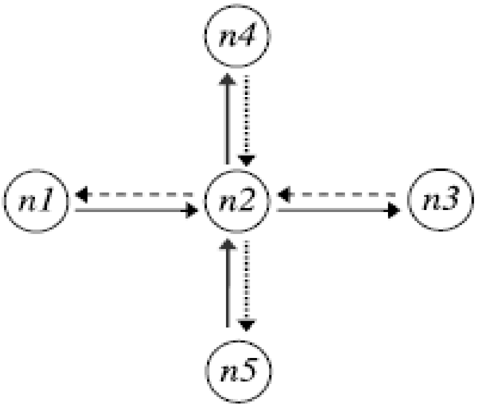
- For the same set of packets to deliver, Coding Gain =

$$\frac{\textbf{number of transmissions required by non-coding approach}}{\textbf{number of transmissions used by COPE}} \geq \mathbf{1}$$

  - For Alice-and-Bob experiment, Coding Gain = 4/3
  - For "X"-topology, COPE w/o opportunistic listening ➔ no gain COPE with opportunistic listening and guessing ➔ Gain = 4/3
  - For cross topology, assuming perfect overhearing, n_2 can XOR 4 packets in each transmission ➔ Coding Gain = 8/5



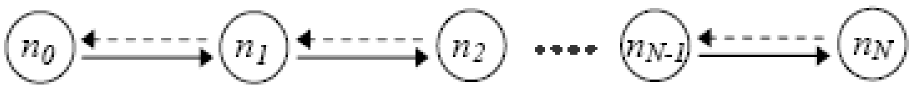
(b) "X" topology



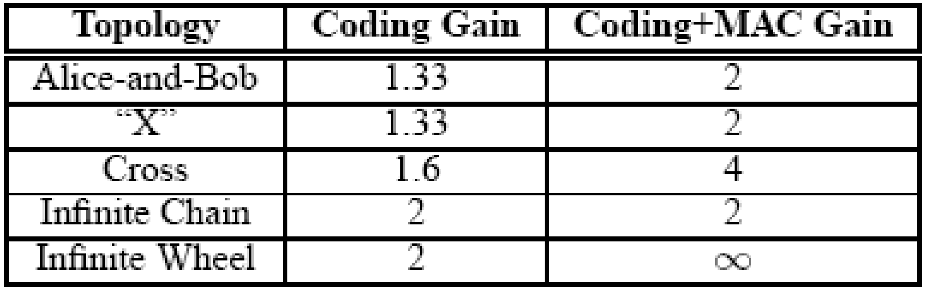
(c) Cross topology

Coding+MAC Gain

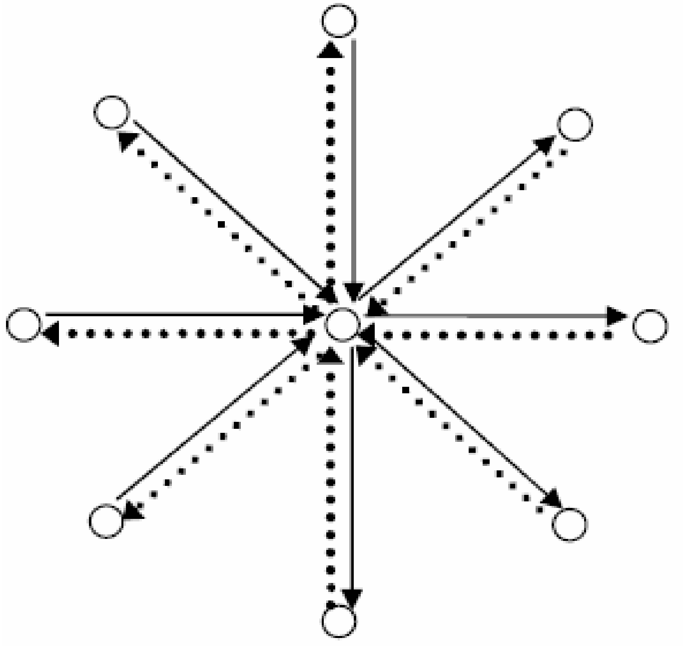
- Interaction between coding and the MAC produces a beneficial side effect, because
  - MAC divides bandwidth equally among contending nodes
- Assume all nodes continuously have some traffic to send, but are limited by their MAC-allocated bandwidth



Chain topology



Topology	Coding Gain	Coding+MAC Gain
Alice-and-Bob	1.33	2
"X"	1.33	2
Cross	1.6	4
Infinite Chain	2	2
Infinite Wheel	2	∞



Wheel topology

Packet Coding Algorithm

- Principle of never delay packets
- Prefer to XOR-ing packets of similar lengths
- Never code together packets headed to the same nexthop
- Maintain virtual queues for the searching of appropriate packets to code
- Limit reordering packets from the same flow
- Ensure each neighbor to whom a packet is headed has a high probability of decoding
- Each node maintains the following data structures
  - **A FIFO queue called *output queue***
  - **Two *per-neighbor virtual queues,* one for small packets, the other for large packets**
  - **A hash table, *packet info,* keyed on packet-id**

Pseudo-Broadcast

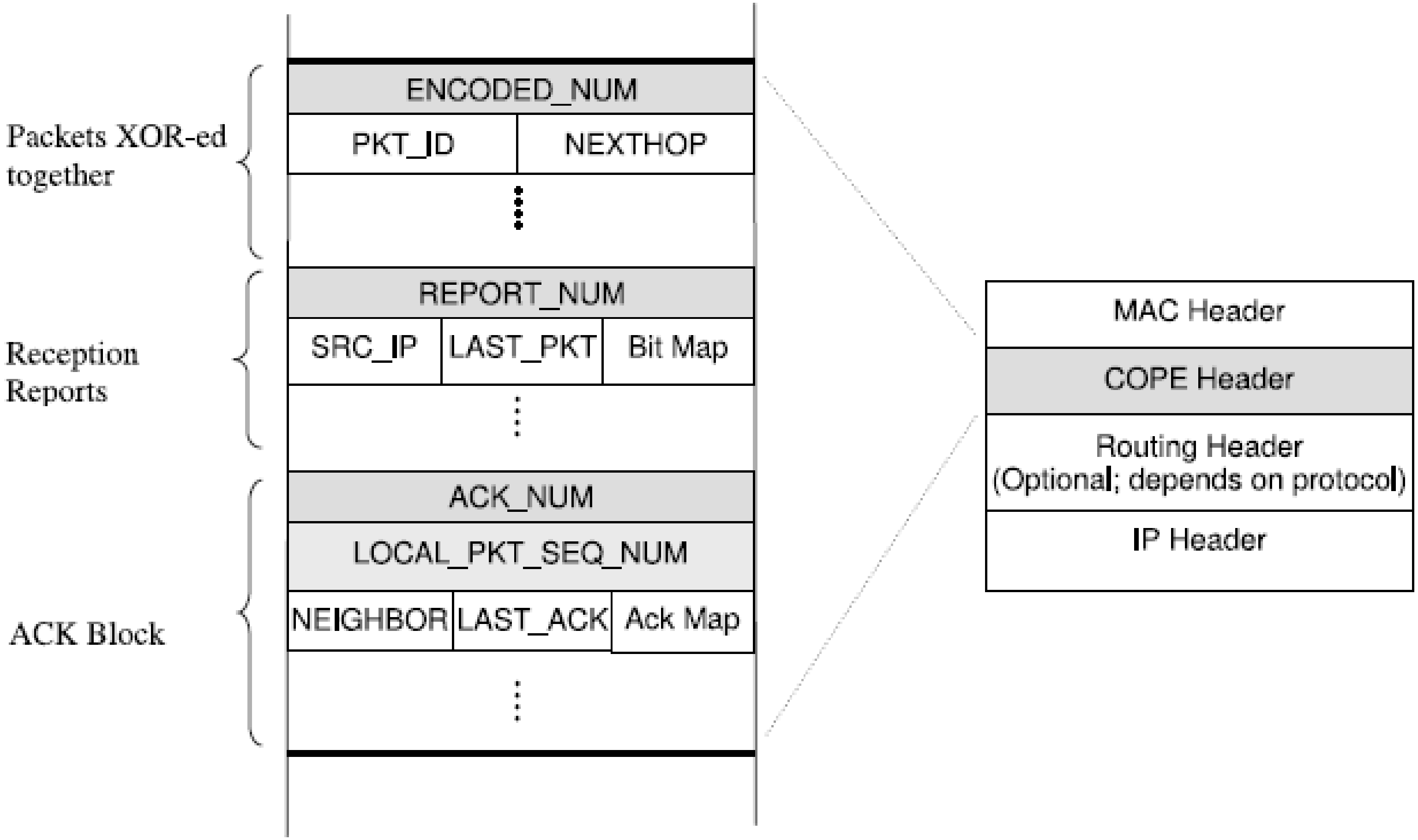
- The broadcast mode of 802.11 MAC can not be used by COPE because of two reasons
  - Poor reliability
  - Lack of backoff
- The solution is pseudo-broadcast
  - Link-layer destination field is set to the MAC of one of the intended recipients
  - An XOR-header is added after link-layer header, listing all next hops of the packet
  - All nodes are set in promiscuous mode to overhear packets
  - When a node receives a packet with other’s MAC, it checks XOR header to see if it is a nexthop. If so, process further, else store as an opportunistic packet

Hop-by-hop ACKs and Retransmissions

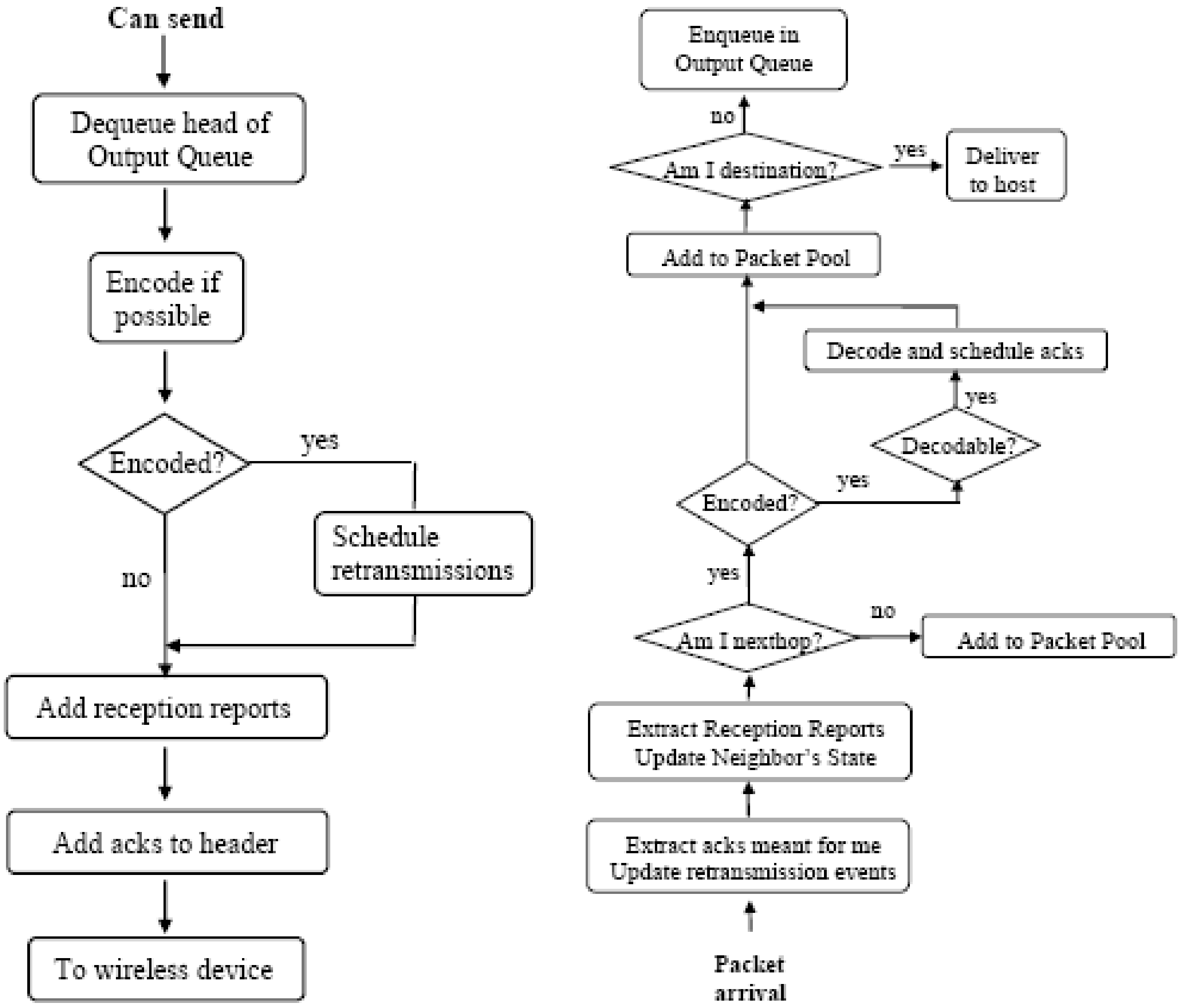
- Why hop-by-hop acks?
  - The sender gets synchronous acks to the encoded packets only from the nexthop that is set as the link-layer destination. There is still a probability of loss to the other nexthops
  - COPE may opportunistically guess a nexthop can decode the XOR-ed packet, when it actually does not
- COPE uses local retransmission to address loss problem
  - For non-coded packets, simply use 802.11 synchronous acks
  - For coded packets, using asynchronous acks and retransmission
    - When a node sends an encoded packet, it schedules a retransmission event for each native packet in the encoded packet
    - If any of these packets is not acked within T_a, the packet is inserted at the head of output queue and retransmitted

Implementation Details

- COPE Header

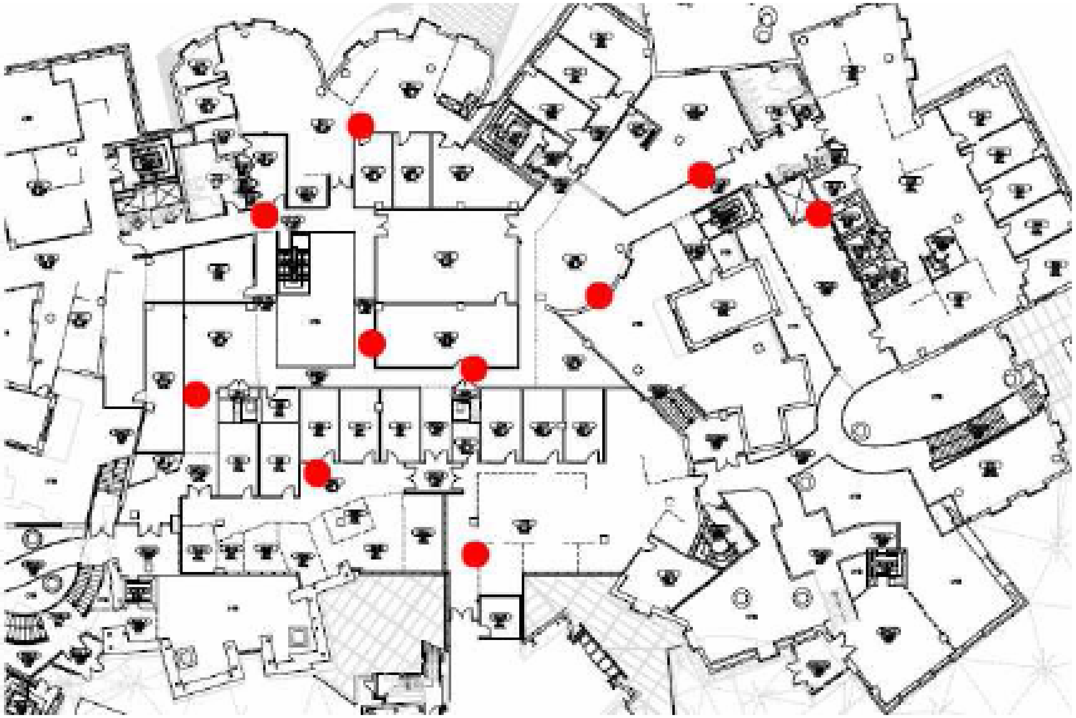


Flow Chart for COPE Implementation



Experimental Results

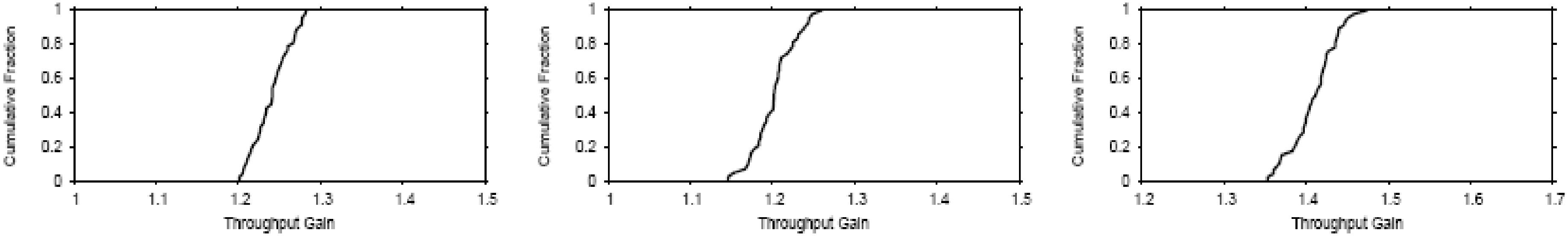
- Testbed: 20-node spanning two floors
  - Path: 1~6 hops
  - Loss rate: 0 ~ 30%
  - Run on 802.11a with a bit-rate of 6Mb/s



- Evaluation Metrics
  - Network Throughput
  - Throughput Gain

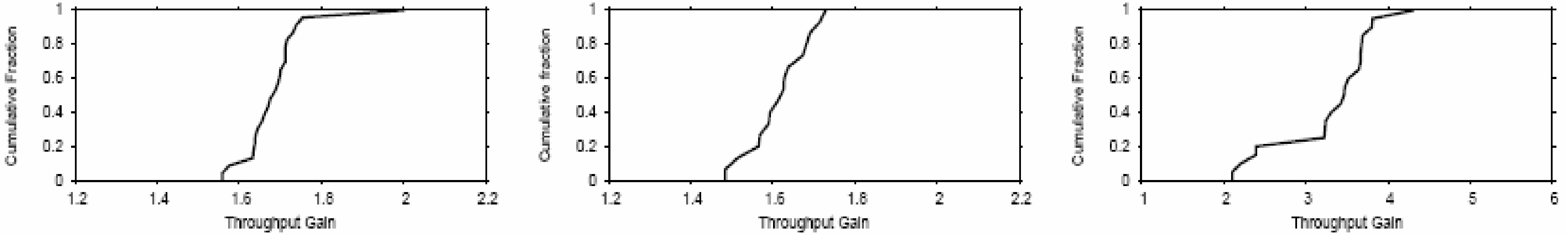
COPE in Gadget Topologies

- For long-lived TCP flows
  - When the traffic exercises congestion control, the throughput gain corresponds to the coding gain



(a) TCP gain in the Alice-and-Bob topology (b) TCP gain in the X-topology (c) TCP gain in the cross topology

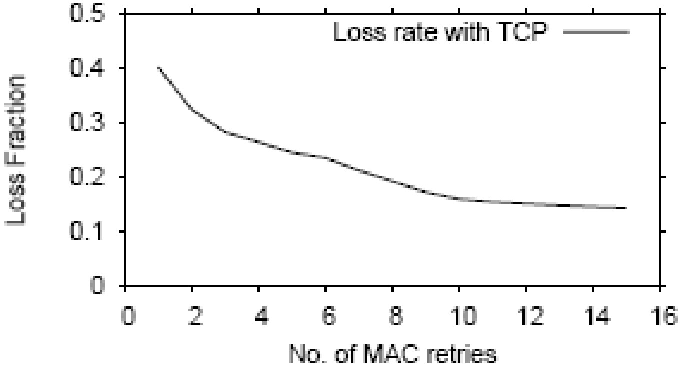
- For UDP flows
  - UDP gains reflect the Coding+MAC gains

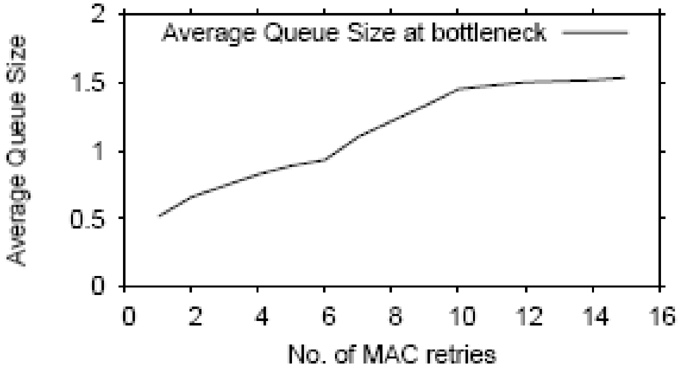


(a) UDP gain in the Alice-and-Bob topology (b) UDP gain in the X-topology (c) UDP gain in the cross topology

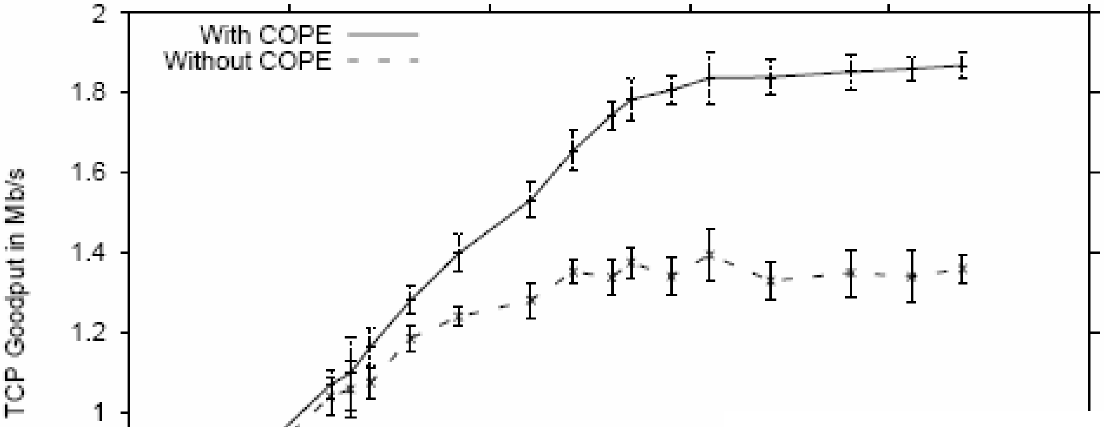
COPE in an Ad Hoc Network

- TCP does not show any significant improvement with coding in the testbed
  - A number of nodes send packets to the bottleneck nodes, but not within carrier sense range of each other → hidden terminal
  - This creates collision-related losses even with the maximum number of MAC retries
  - The bottleneck node never see enough traffic for coding

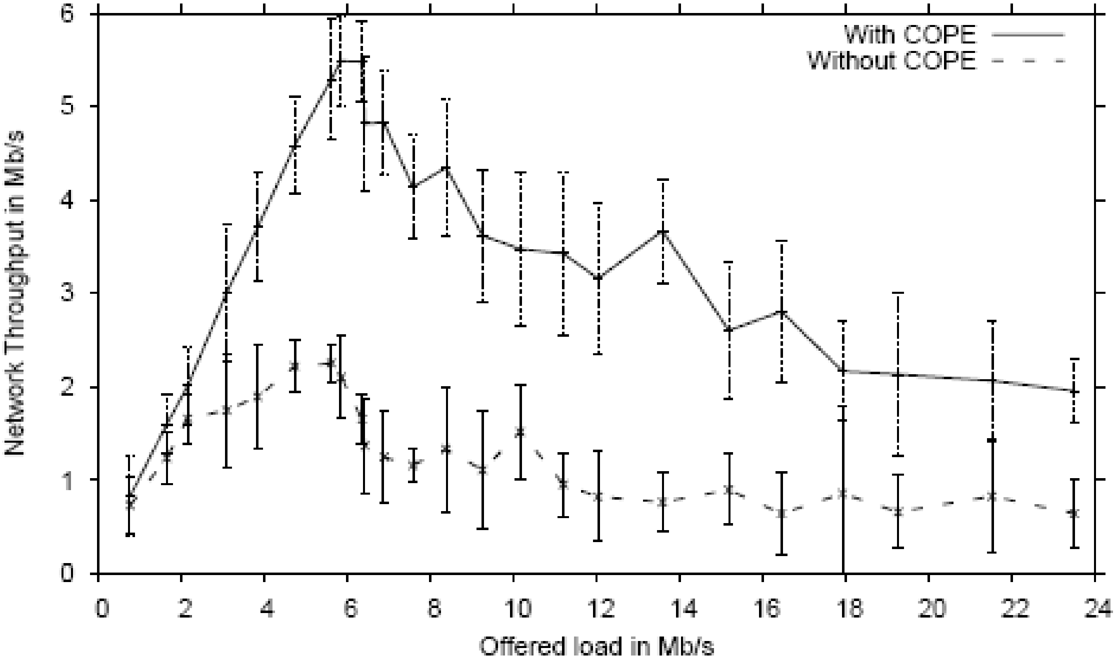




COPE in an Ad Hoc Network

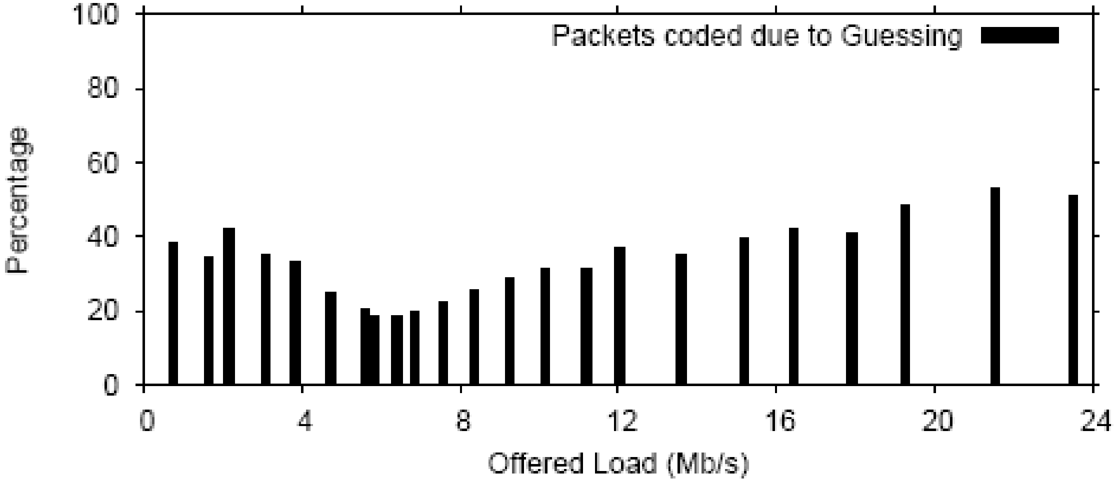


- Compress the testbed topology to eliminate the collision-related losses
- COPE provides 38% increase in TCP goodput over no coding

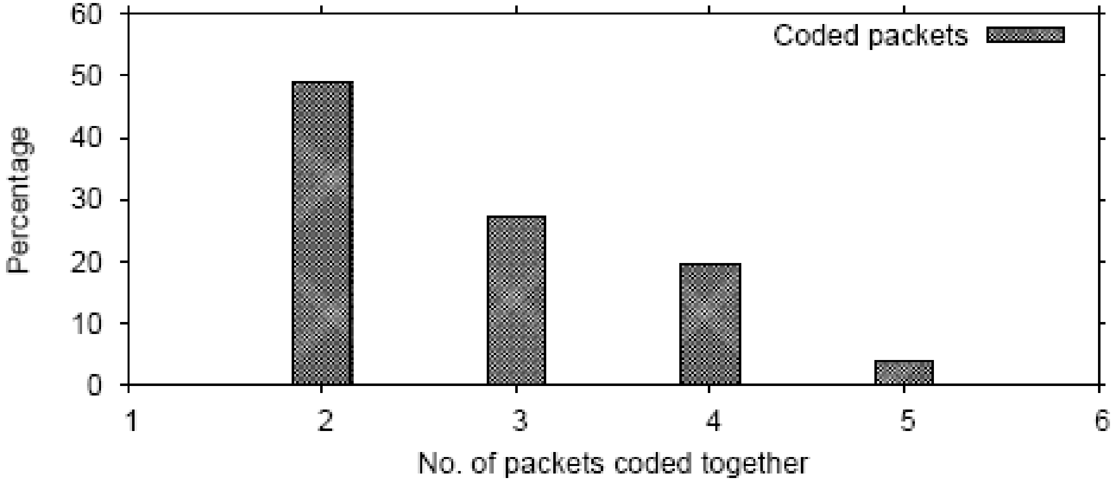


- For large scale testbed experiments with UDP
- COPE can provide 3-4x increase in the throughput of wireless Ad hoc networks

Coding Efficiency

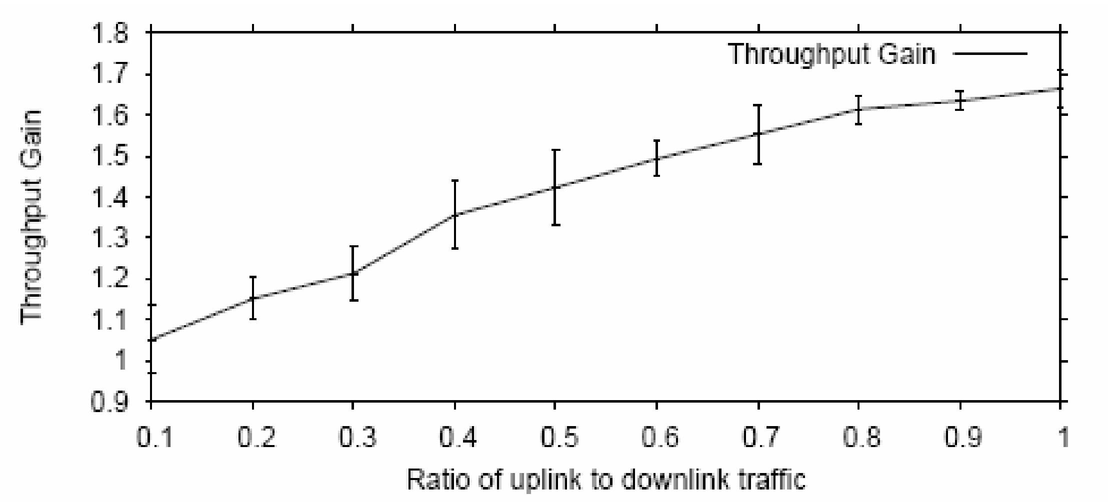


- How much of the coding is due to guessing, as opposed to reception reports?

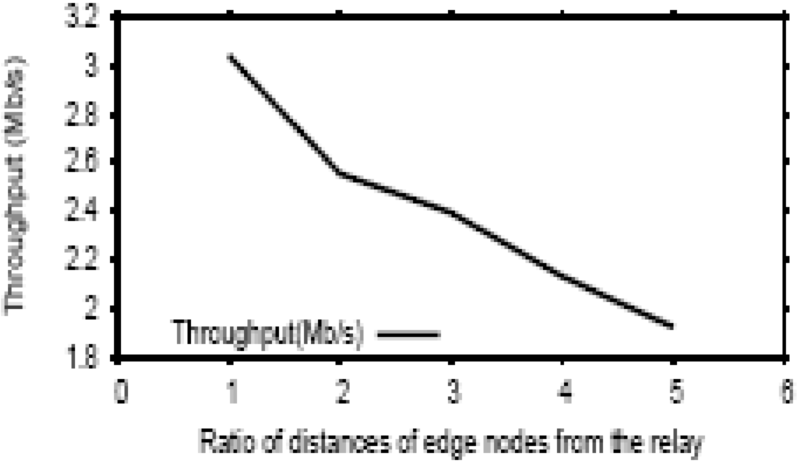


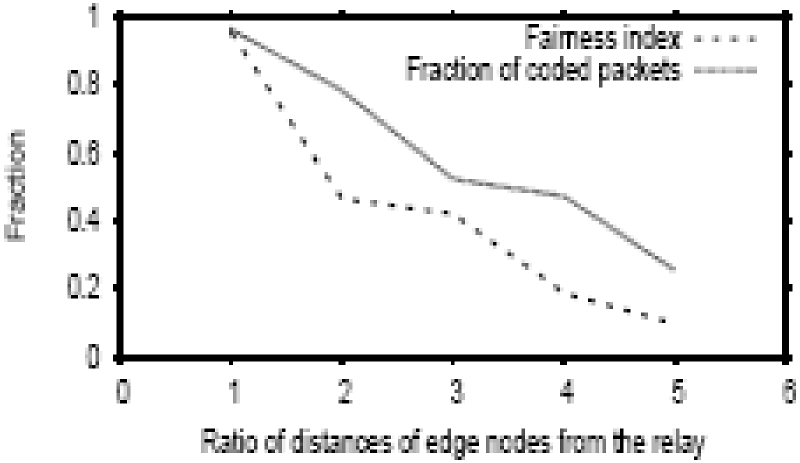
- How many packets are getting coded together?

COPE in a Mesh Access Network



- Nodes are divided into 4 sets, each communicates with Internet via a specific node as gateway
- As uplink traffic increases, gain increases to 70%





Discussion and Conclusion

- Summary
  - For congested wireless medium and traffic of many random UDP flows, COPE delivers a 3-4*x* increase in the throughput of the wireless testbed.
  - When the traffic does not exercise congestion control, COPE’s improvement exceeds the expected coding gain and agrees with the Coding+MAC gain.
  - For a mesh network connected to the Internet, the improvement varies depending on the ratio of download traffic to upload traffic at the gateway, and ranges from 5% to 70%.
  - Hidden terminals create a high loss rate that cannot be masked even with the maximum number of 802.11 retransmissions.

Discussion and Conclusion

- COPE can be used in multi-hop wireless networks
  - **Memory**: COPE’s nodes need to store recently heard packets for future decoding
  - **Omni-directional antenna**: for opportunistic listening
  - **Power requirements**: current design of COPE does not optimize power usage and assumes the nodes are not energy limited
- The idea of COPE may be applicable beyond WiFi mesh networks
  - COPE can conceptually work with a variety of MAC protocols including WiMAX and TDMA
  - COPE may be modified to address the needs of sensor networks