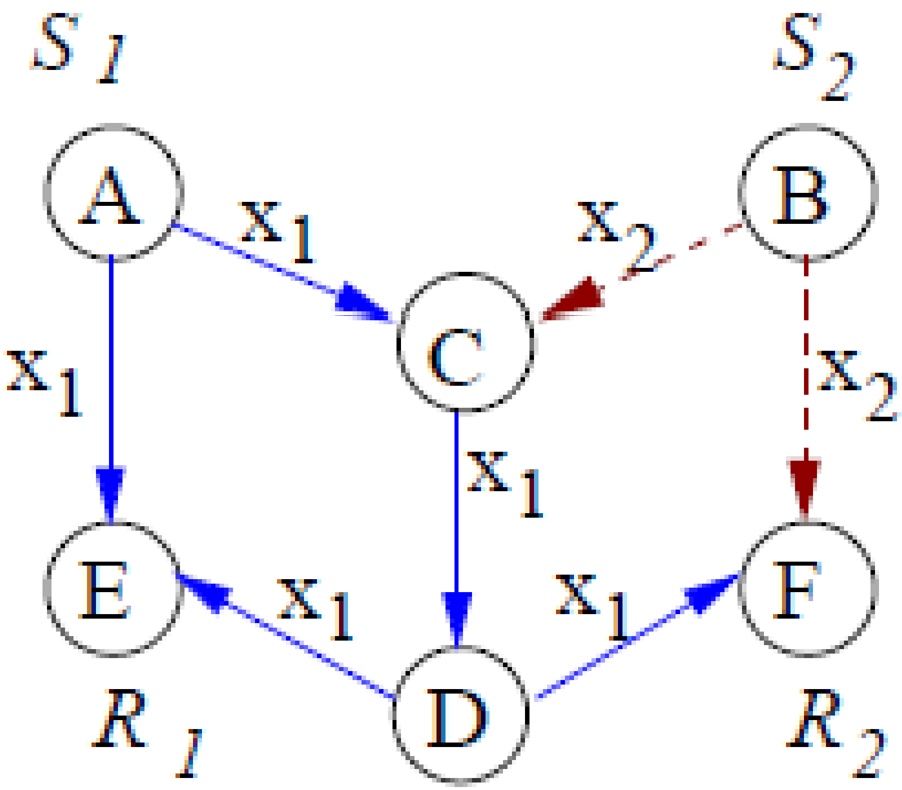
CS 5229
Advanced Compute Networks

Practical Wireless Network Coding

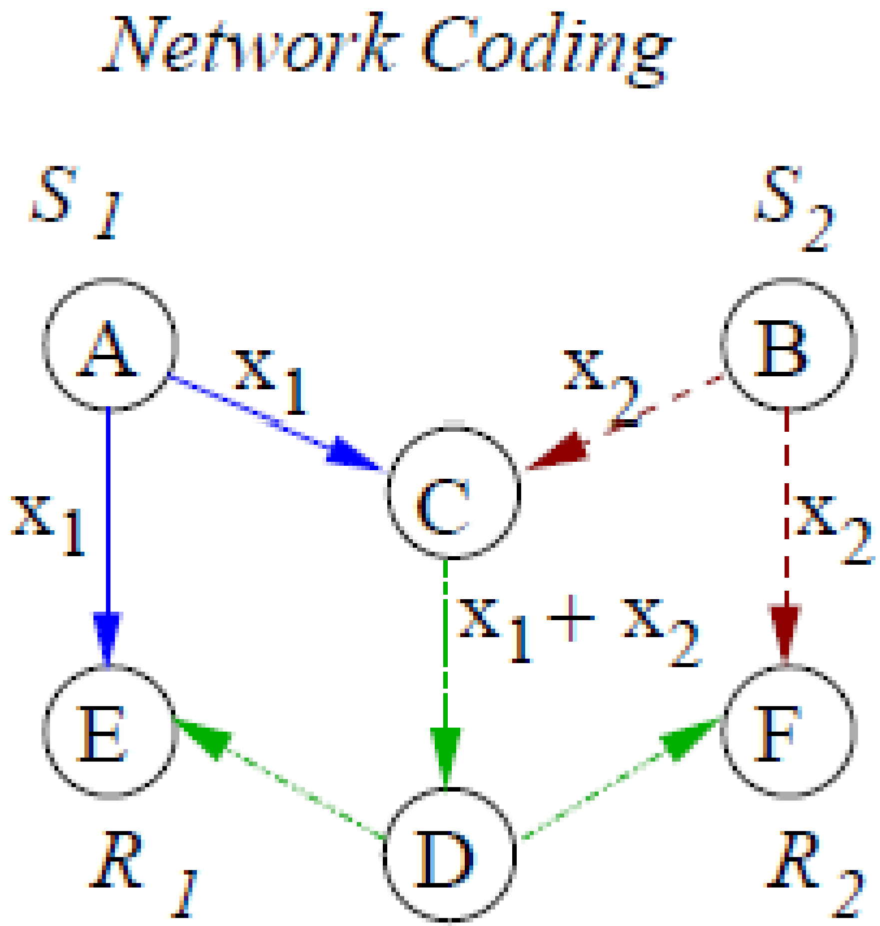
Sachin Katti, et. al., “XORs in the AIR: Practical Wireless Network Coding,” SIGCOMM 2006.

Traditional Method



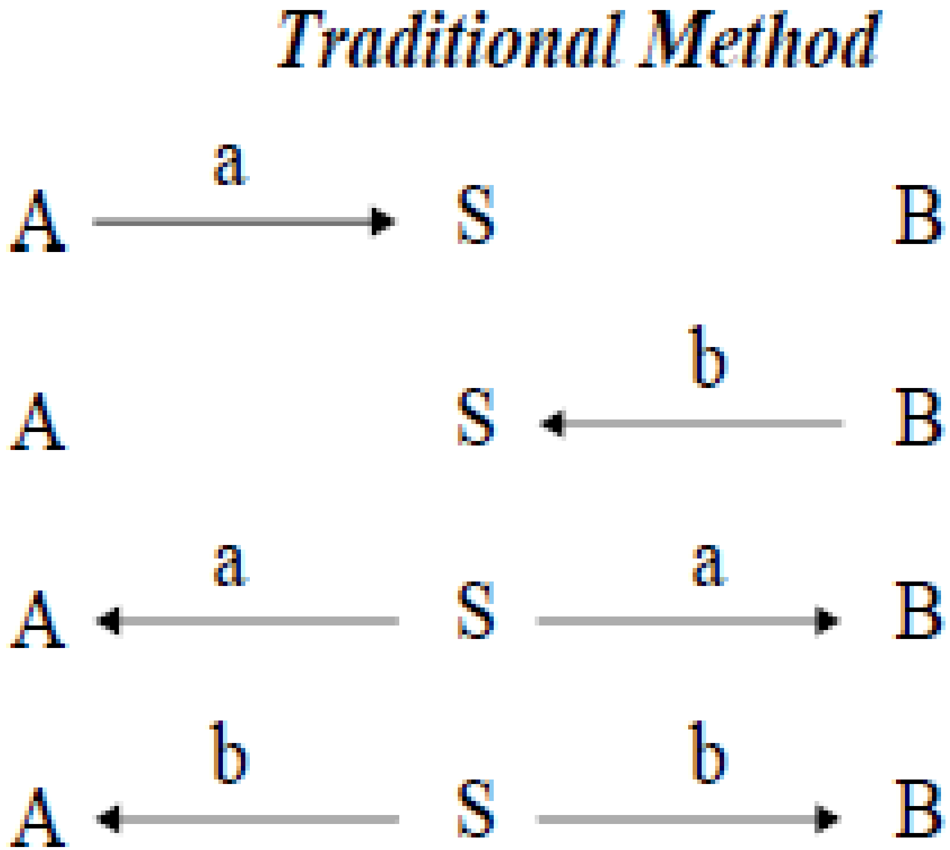
Maximum rate = 1.5

S1 & S2 send packets to R1 & R2.



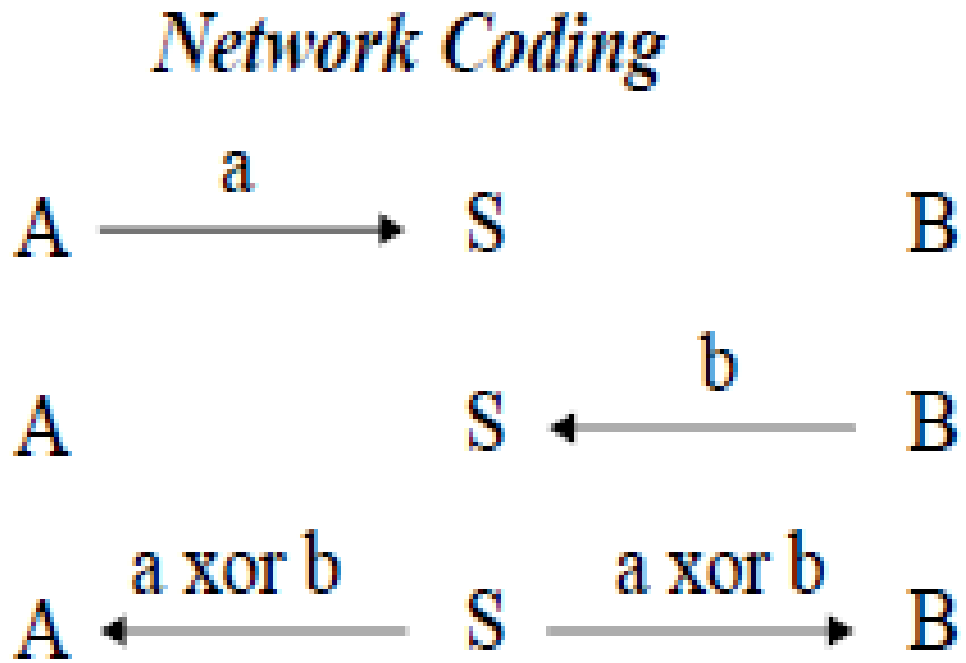
Maximum rate = 2

Wireless Transmission



Maximum Rate = 1/4 (1 pkt in 4 rounds)

Wireless Transmission



Maximum Rate = = 1/3 (1 pkt in 3 rounds)
Gain = (1/3) / (1/4) = 4/3

About Linear Network Coding

Network Coding: An Instant Primer

Christina Fragouli
EPFL - IC
christina.fragouli@epfl.ch

Jean-Yves Le Boudec
EPFL - IC
jean-yves.leboudec@epfl.ch

Jörg Widmer
DoCoMo Labs
widmer@docomolab-euro.com

Encoding

Assume that a number of original packets $M^1, ..., M^n$ are generated by one or several sources. In linear network coding, each packet in the network is associated with a sequence of coefficients $g_1, ..., g_n$ in $\mathbb{F}_{2^s}$ and is equal to $X = \sum_{i=1}^{n} g_i M^i$. The summation has to occur for every symbol position, i.e., $X_k = \sum_{i=1}^{n} g_i M_k^i$, where M_k^i and X_k is the kth symbol of M^i and X respectively.

Decoding

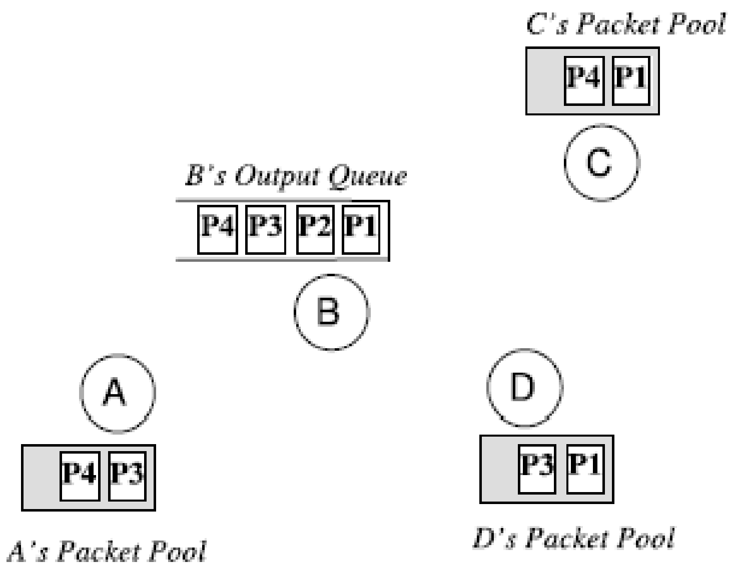
Assume a node has received the set $(g^1, X^1), ..., (g^m, X^m)$. In order to retrieve the original packets, it needs to solve the system $\{X^j = \sum_{i=1}^n g_i^j M^i\}$ (where the unknowns are M^i). This is a linear system with m equations and n unknowns. We need $m \geq n$ to have a chance of recovering all data, i.e., the number of received packets needs to be at least as large as the number of original packets. Conversely, the condition $m \geq n$ is not sufficient, as some of the combinations might be linearly dependent.

Random Linear Code

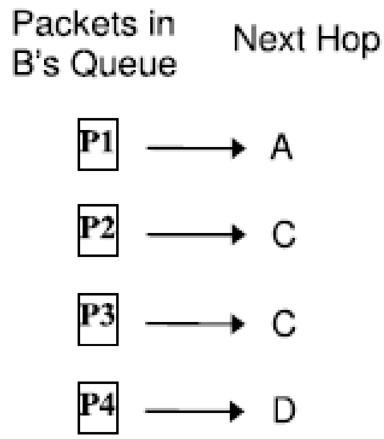
With random network coding there is a certain probability of selecting linearly dependent combinations [14]. This probability is related to the field size 2^s. Simulation results indicate that even for small field sizes (for example, $s = 8$) the probability becomes negligible [29].

COPE

- Forwarding architecture that exploits network coding to improve throughput of wireless networks
- Some highlights
  - Exploits idle listening
  - Exploits broadcast nature of wireless channel
  - Buffers packets
  - Practical implementation



(a) B can code packets it wants to send



(b) Nexthops of packets in B's queue

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)

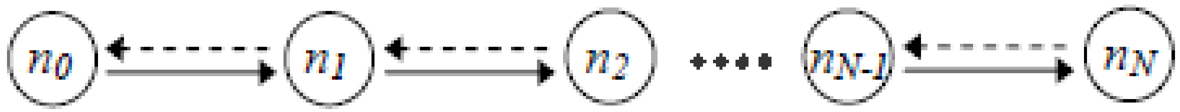
Opportunistic Listening

- Listens to wireless channel and buffer packets even if it is not the destination (for say 0.5s)
  - Why not longer?
- Node broadcasts reception report to tell its neighbors which packet it has stored
  - Piggy back to data packet
  - Special control packet

Opportunistic Coding

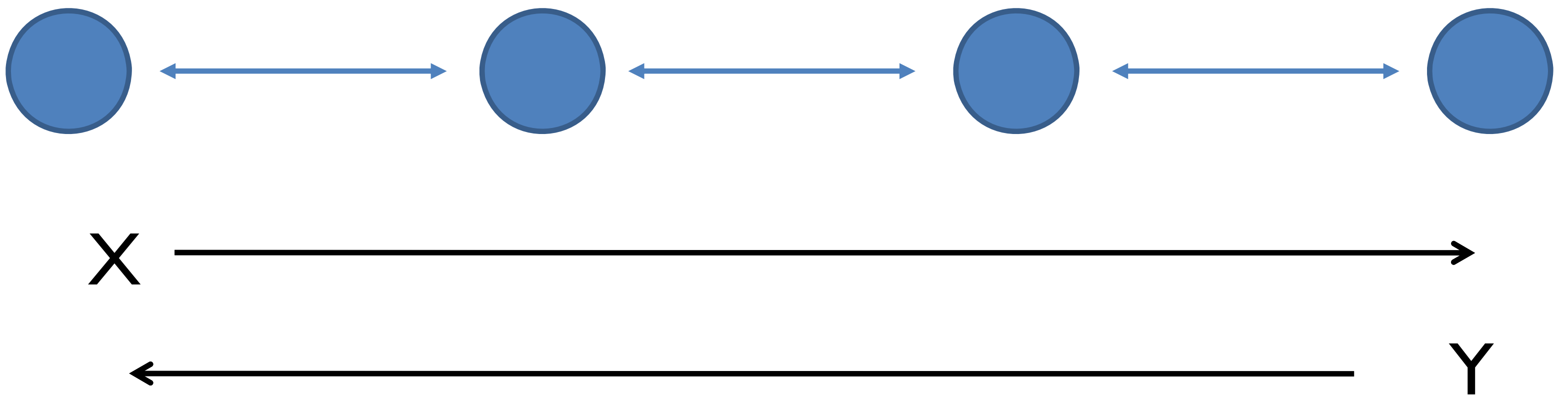
- Only XOR coding is used (not general linear code)
- Which packet should a node selects for transmission?
- Selects the packet that can be decoded by the maximum number of neighbors
- In COPE, a coded packet must be decoded by its neighbors.
  - Coded packets are not forwarded/relayed further

Coding Gains



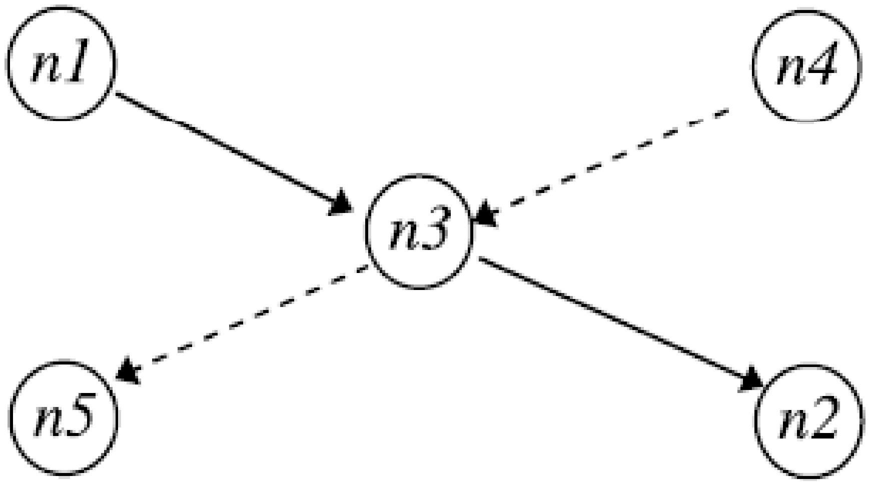
(a) Chain topology; 2 flows in reverse directions.

THEOREM 4.1. *In the absence of opportunistic listening, COPE's maximum coding gain is 2, and it is achievable.*



{Largest Seq #) Y(i-3)	X(i-1),Y(i-2)	X(i-2),Y(i-1)	X(i-3)
X(i), Y(i-3)	X(i),Y(i-2)	X(i-2), Y(i-1)	X(i-3)
Y(i-2)	X(i-1) XOR Y(i-2) X(i),Y(i-2)	X(i-1),Y(i-1)	X(i-3)
Y(i-2)	X(i),Y(i-1)	X(i-2) XOR Y(i-1) X(i-1),Y(i-1)	X(i-2)
Y(i-2)	X(i),Y(i-1)	X(i-1),Y(i)	Y(i) X(i-2)

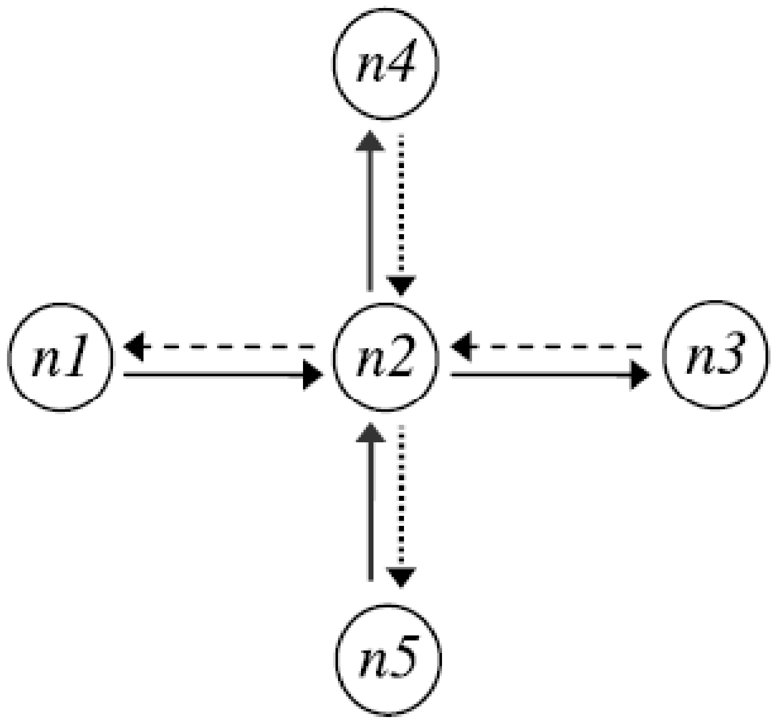
What is the coding gain?



(b) "X" topology
2 flows intersecting at n_2.

n5 can overhear n1
n2 can overhear n4

What is the coding gain?

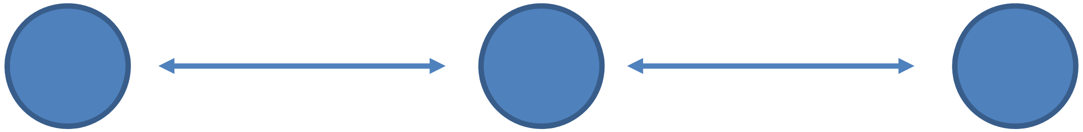


(c) Cross topology
4 flows intersecting at n_2

n1 & n3 hears n4 & n5

n4 & n5 hears n1 and n3

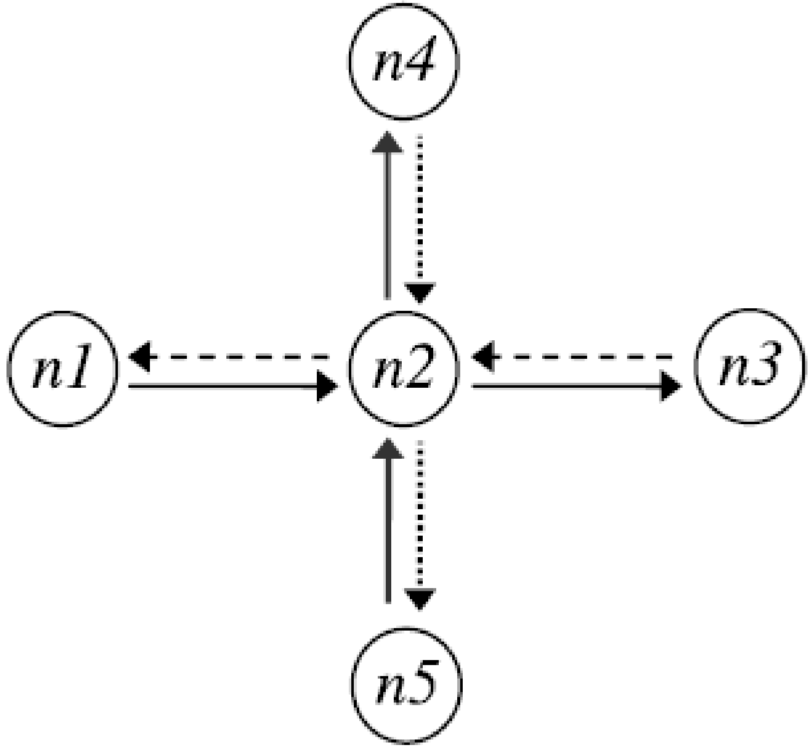
Coding + MAC



With only coding, gain is 4/3.

But MAC protocol shares bandwidth equally, what is the gain?

What is the coding + MAC gain?



(c) Cross topology

4 flows intersecting at n_2

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	∞

Table 2—**Theoretical gains for a few basic topologies.**

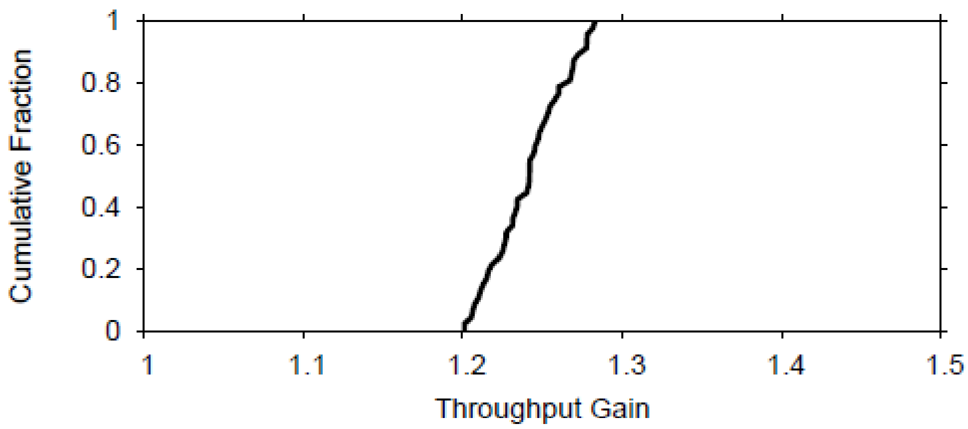
What kind of gain can we expect in practice?

Coding Algorithm

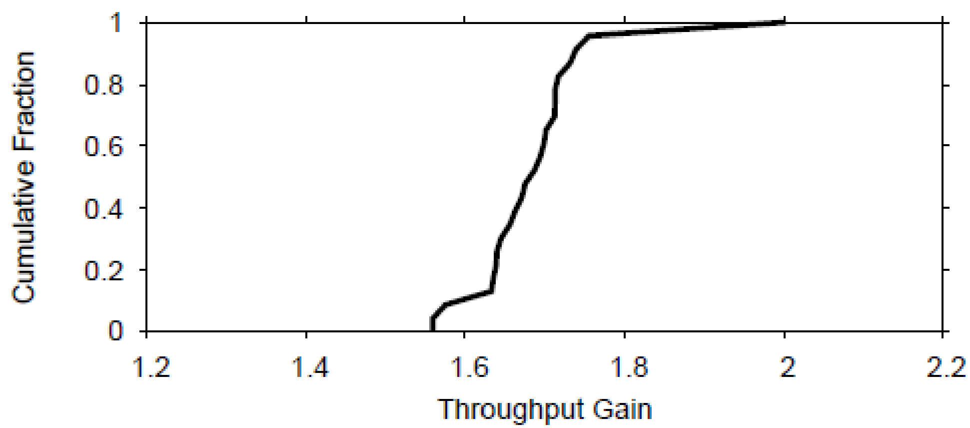
- Schedule from head of queue, does not delay packet to increase coding opportunity
- Coding/XOR packets of similar length
- Never code together packets headed to the same next hop
- When reception reports have not been received, use estimate, including delivery probability

Pseudo-Broadcast

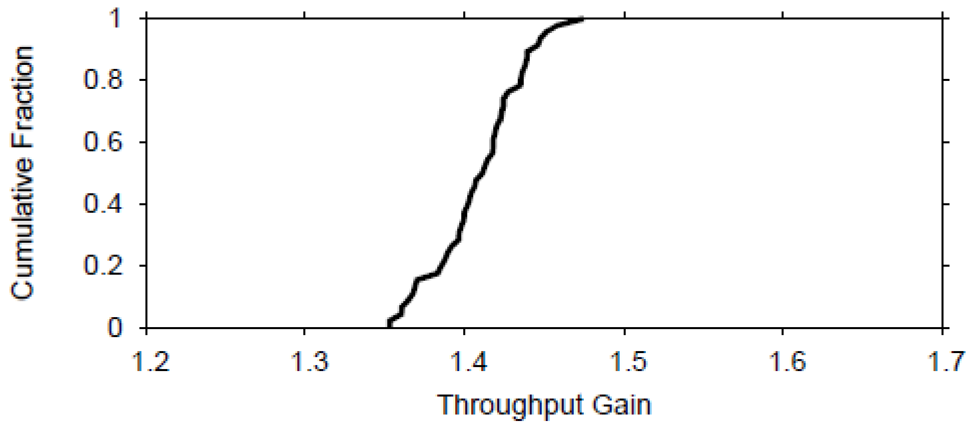
- Broadcast packets are not reliable
  - No (link layer) acknowledgement
  - Difficult to retransmit and add backoff
- Use pseudo-broadcast
  - Use unicast, sends to one receiver
  - Use extended header to list other receivers
  - Only the named receiver acknowledges
  - Does not resolve reliability issue



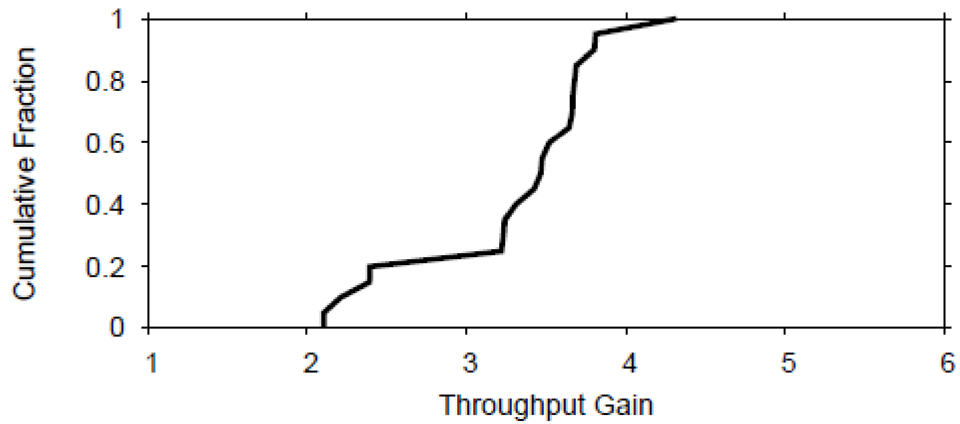
(a) TCP gain in the Alice-and-Bob topology



(a) UDP gain in the Alice-and-Bob topology



(c) TCP gain in the cross topology



(c) UDP gain in the cross topology

TCP Performance in Testbed

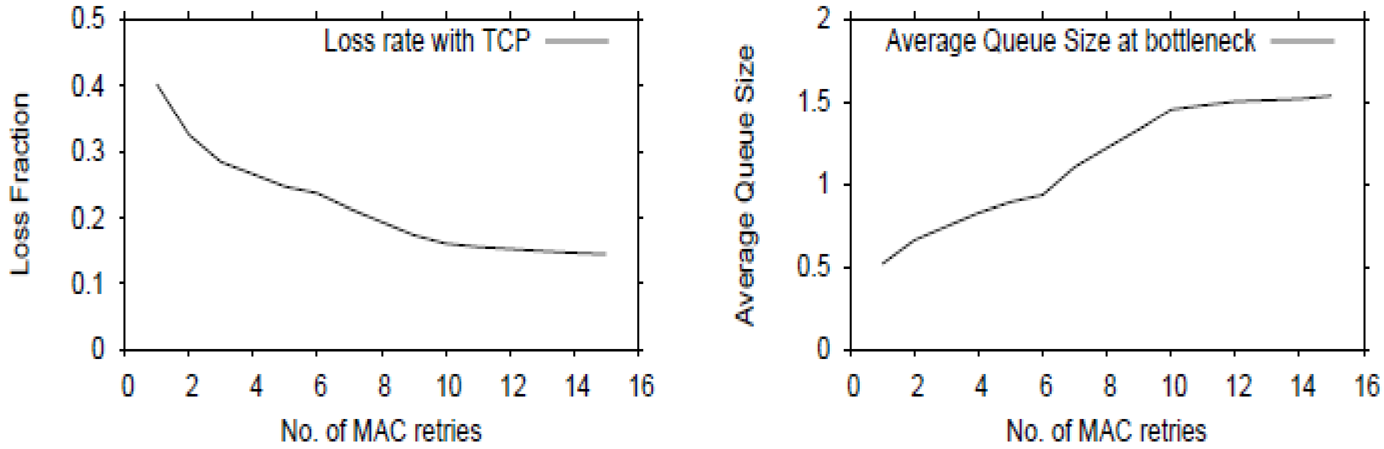
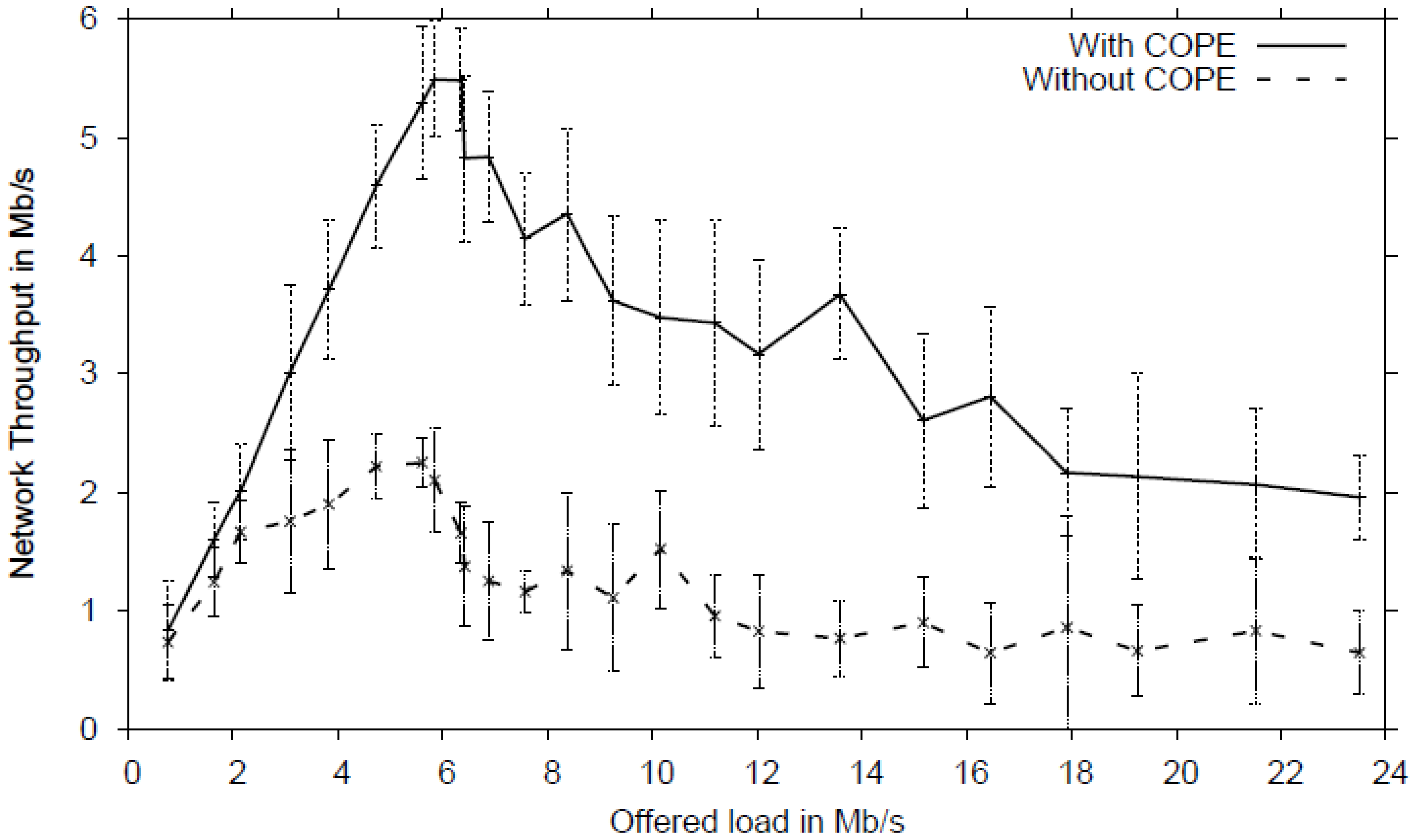
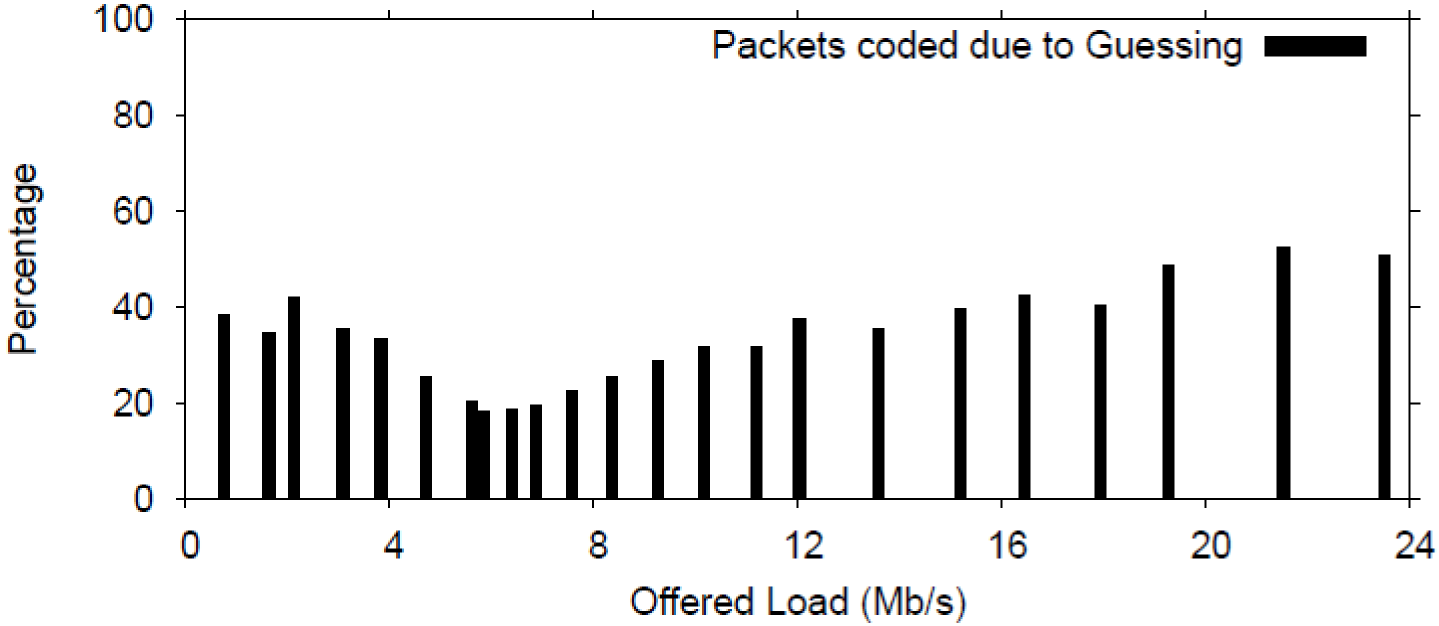


Figure 10—End-to-end loss rate and average queue size at the bottlenecks for the TCP flows in the testbed. Loss rates are as high as 14% even after 15 MAC retries; TCP therefore performs poorly. The queues at the bottlenecks almost never build up resulting in very few coding opportunities and virtually no gains.

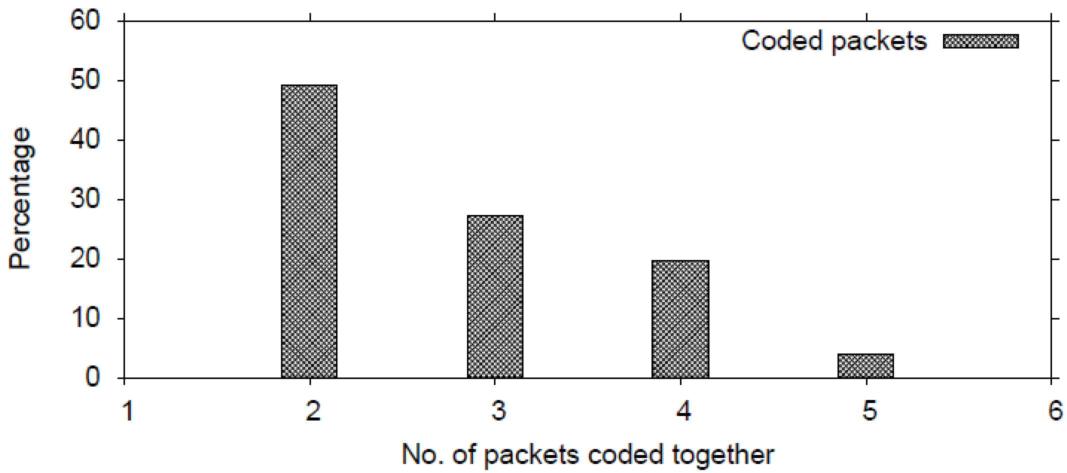
Performance with UDP



UDP - Details



Percentage
100
80
60
40
20
0
Packets coded due to Guessing
0
4
8
12
16
20
24
Offered Load (Mb/s)



Percentage
60
50
40
30
20
10
0
Coded packets
1
2
3
4
5
6
No. of packets coded together