Vern Paxson’s Paper “End-to-End Internet Packet Dynamics”, 1997/99

How often are packets dropped?

How often are packets reordered?

Why these questions?

1. Understand the Internet

“When you can measure what you are speaking about, and express it in numbers, you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meagre and unsatisfactory kind;”

- Lord Kelvin

11. Model the Internet

III. Enable more accurate evaluation through simulations

IV. Lead to a better application/systems design

How often are packets dropped?

How often are packets reordered?

How to answer these questions?

Collect lots of packet traces

Analyze the traces

Trace collection:

large number of flows

a variety of sites

many packets per flow

use TCP

Why TCP:

real-world traffic

will not overload the network

Time between measurement is Poisson distributed

PASTA Theorem. Intuitively, if we make n observations and k observations is in some state S and n-k in another state, then we can assume prob of observing S is approximately k/n.

Two traces:

N1: Dec94
N2: Nov-Dec95

use tcpdump at sender + receiver

100 kB

Size of file transfered

21

Number of sites

20800

Number of trace pairs

Part 1:

The Unexpected

Packet Reordering

1
2
5
3
4

2 reorderings

N1	N2
36%	**12%**

Percentage of connections with
at least one out-of-order delivery

N1	N2
2%	**.3%**

Percentage of data packets out-of-order

N1

.6%

N2

.1%

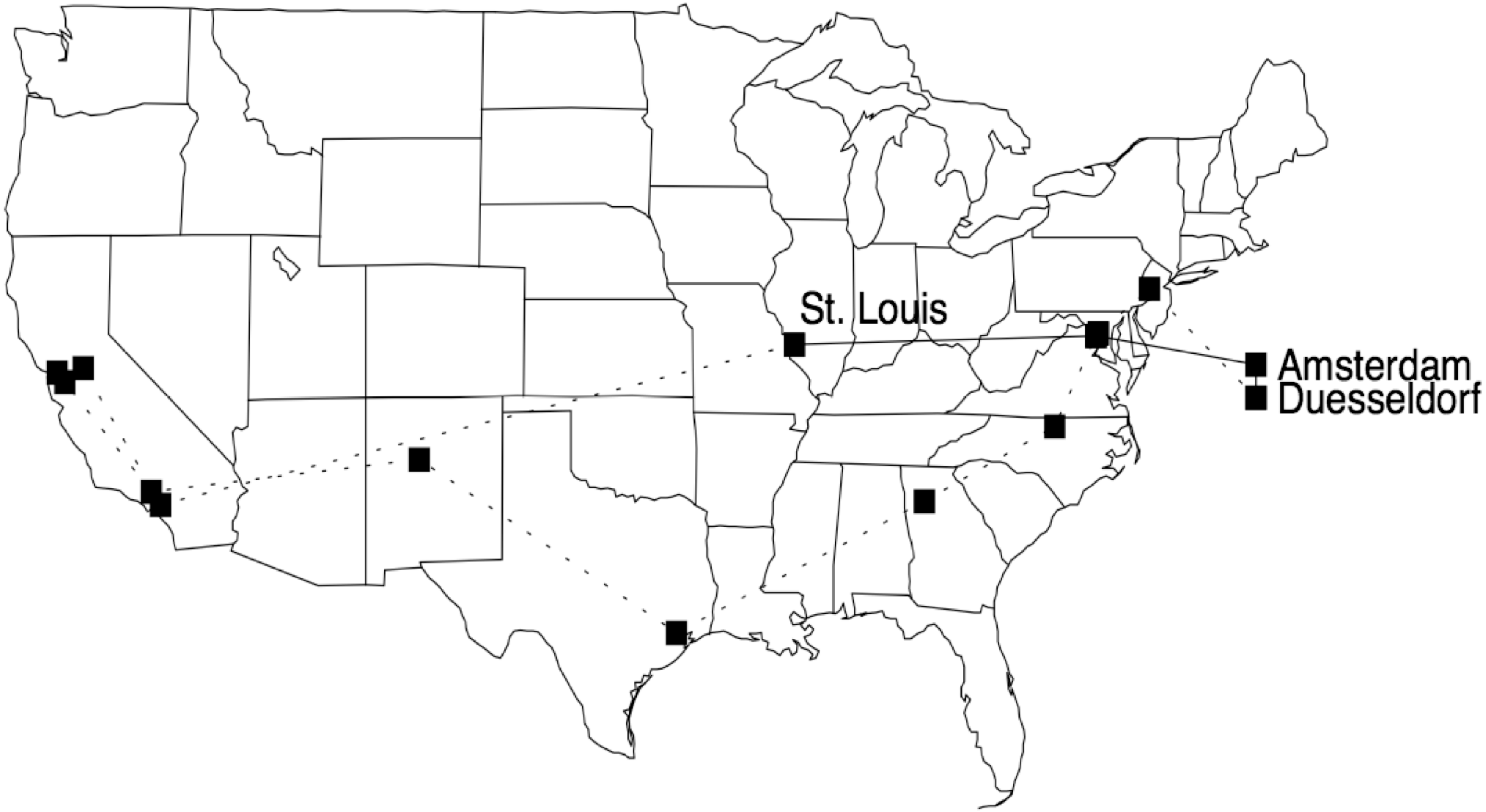
Percentage of ACK packets out-of-order

Data packets are usually sent closer together.

From	To
15%	**.2%**

Percentage of packets out-of-order
to and from U of Colorado in N1.

Route fluttering:
alternate packets
can take different
route to dest.



Taken from Paxson's PhD Thesis: Alternate routes are taken for packets from WUSTL to U Mannheim

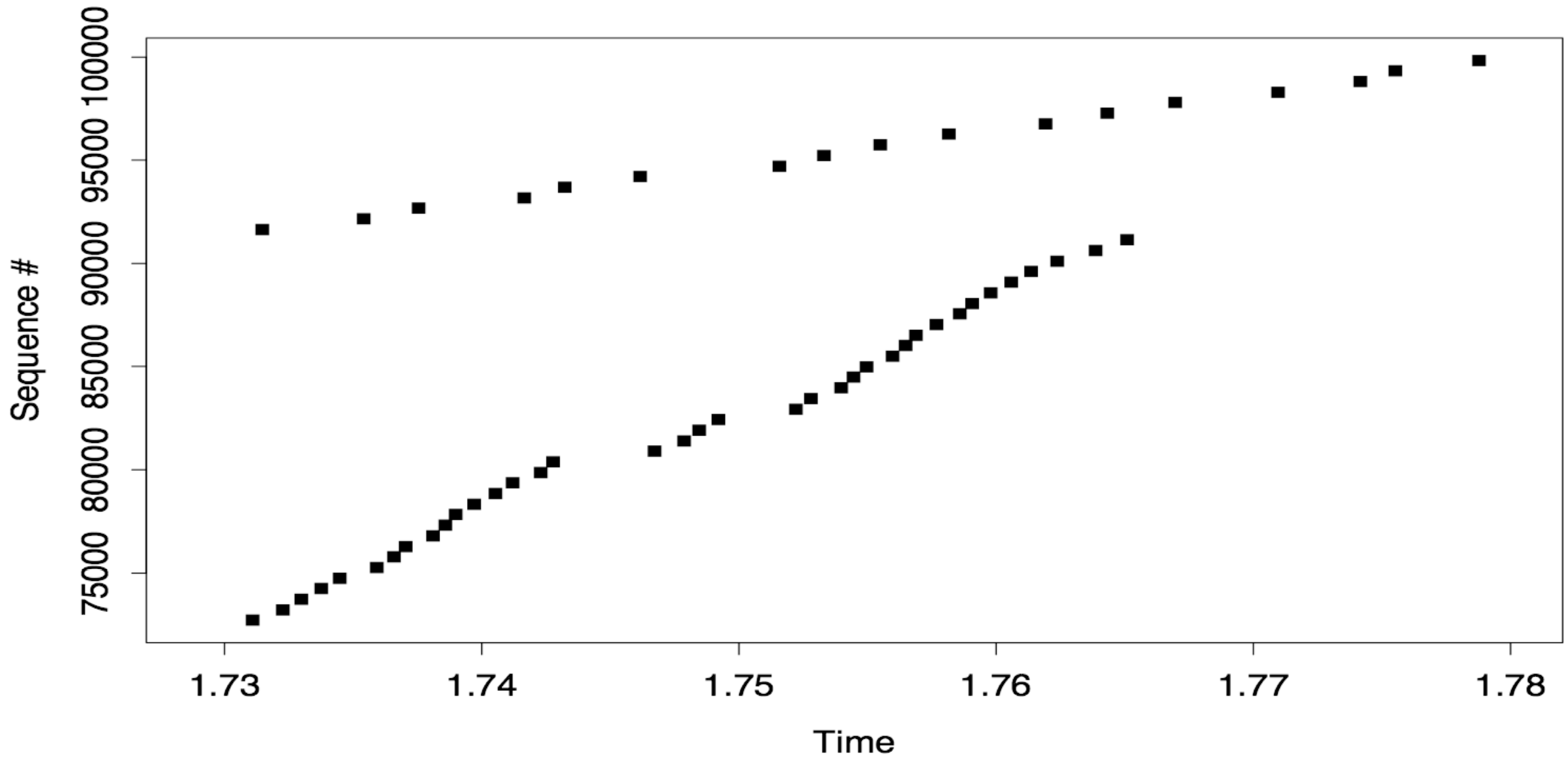


Fig 1 from the paper, showing large gap
and two slopes.

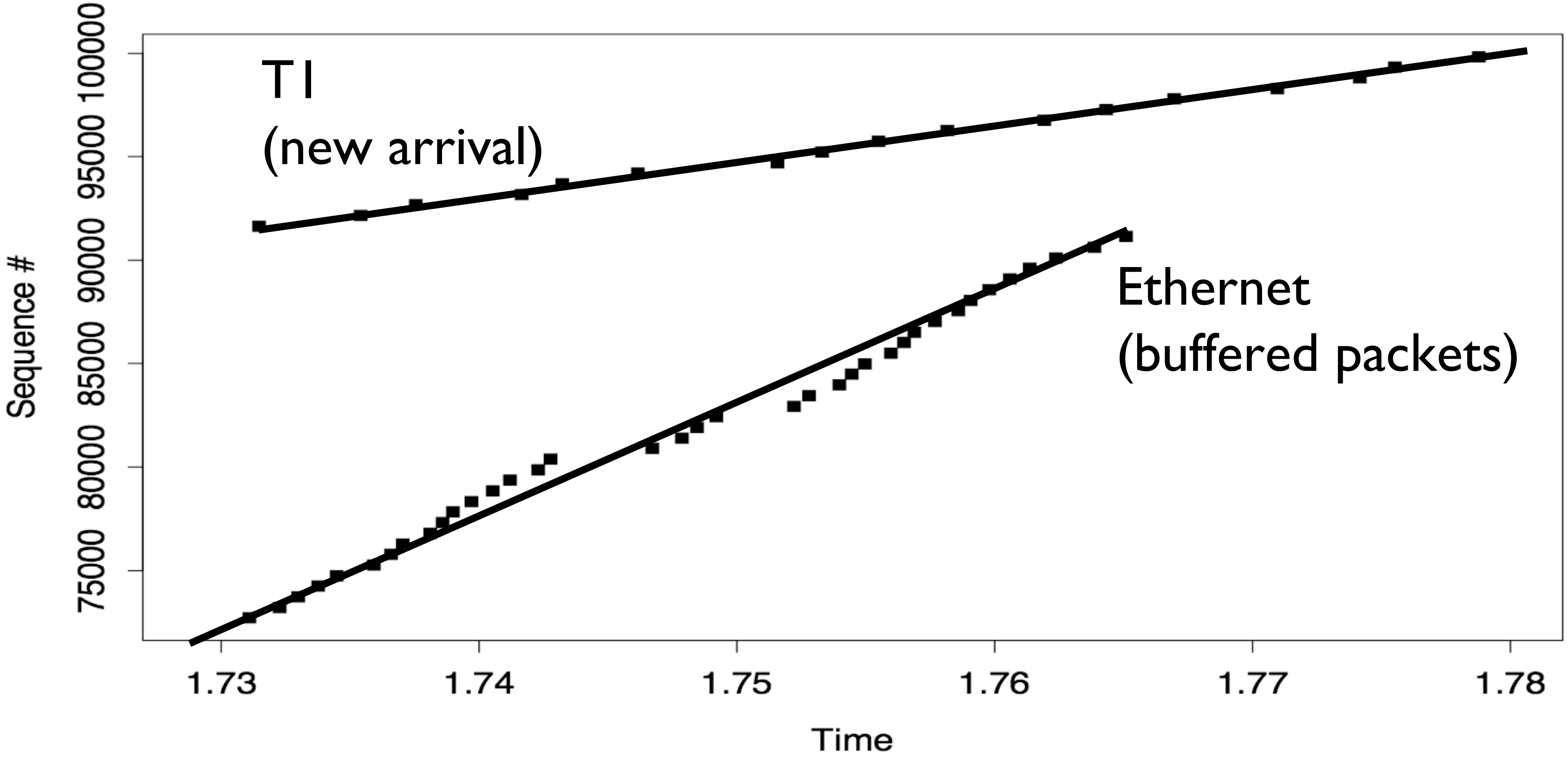
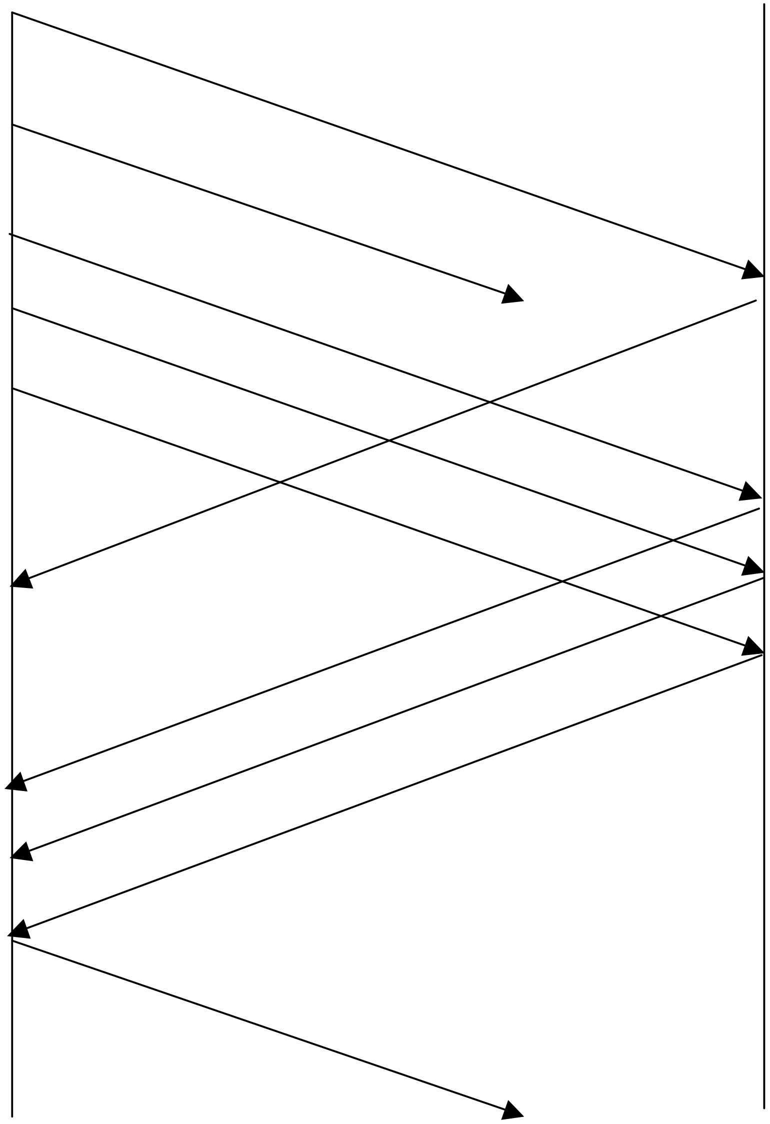
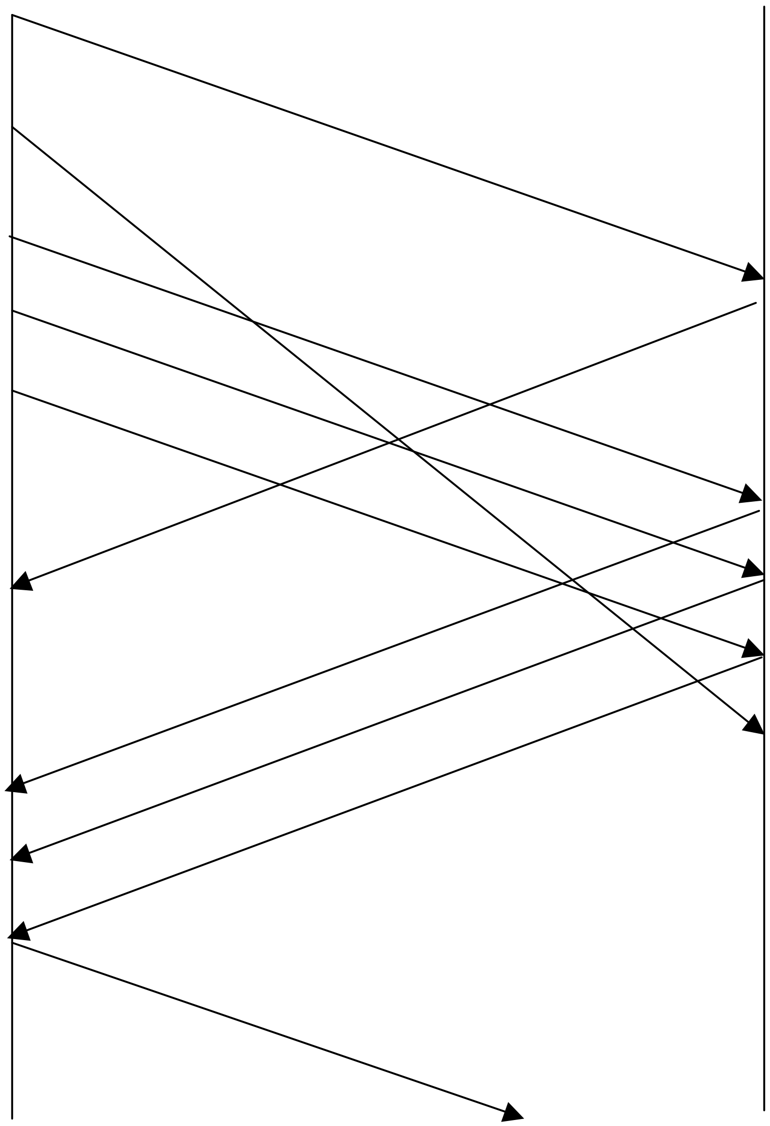


Fig 1 from the paper, showing large gap and two slopes.

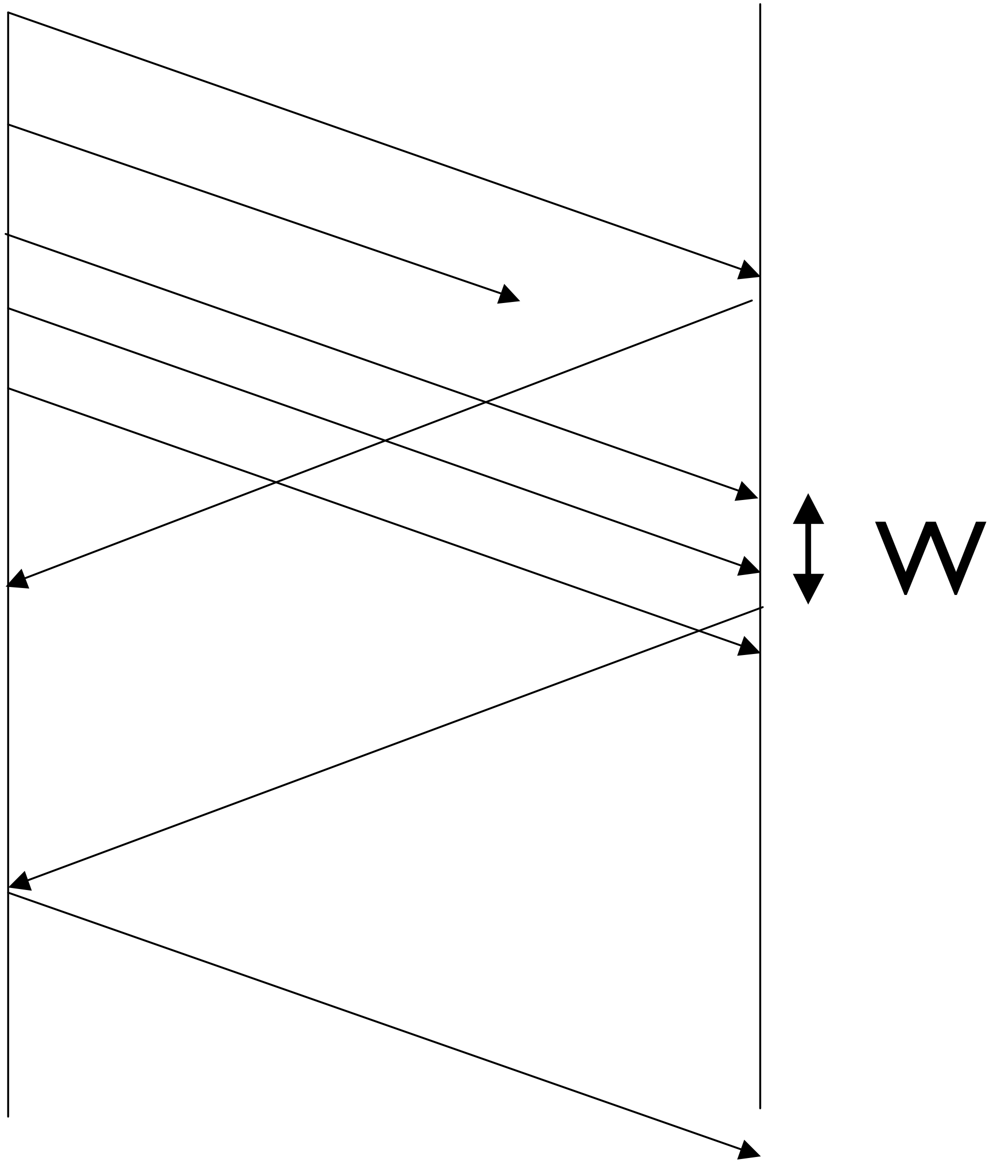
Impact of Packet Reordering

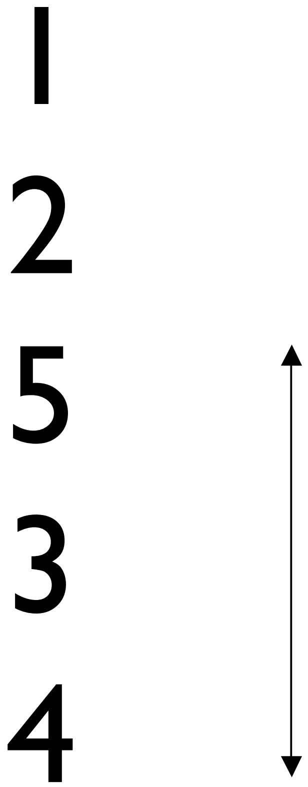




$N_d = 3$ is a conservative choice.

What if receiver
wait longer before
sending dup ack?


W



1

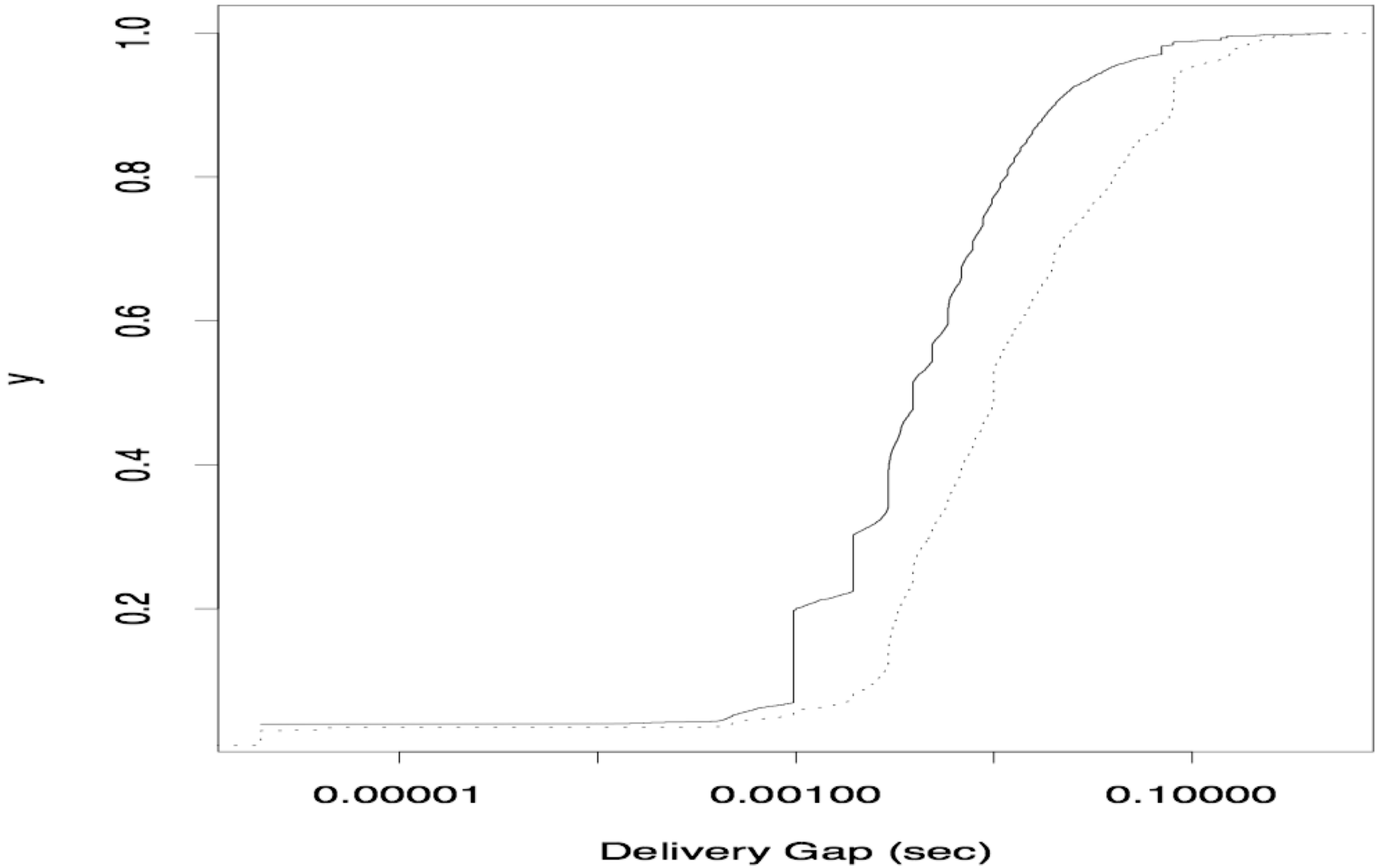
2

5

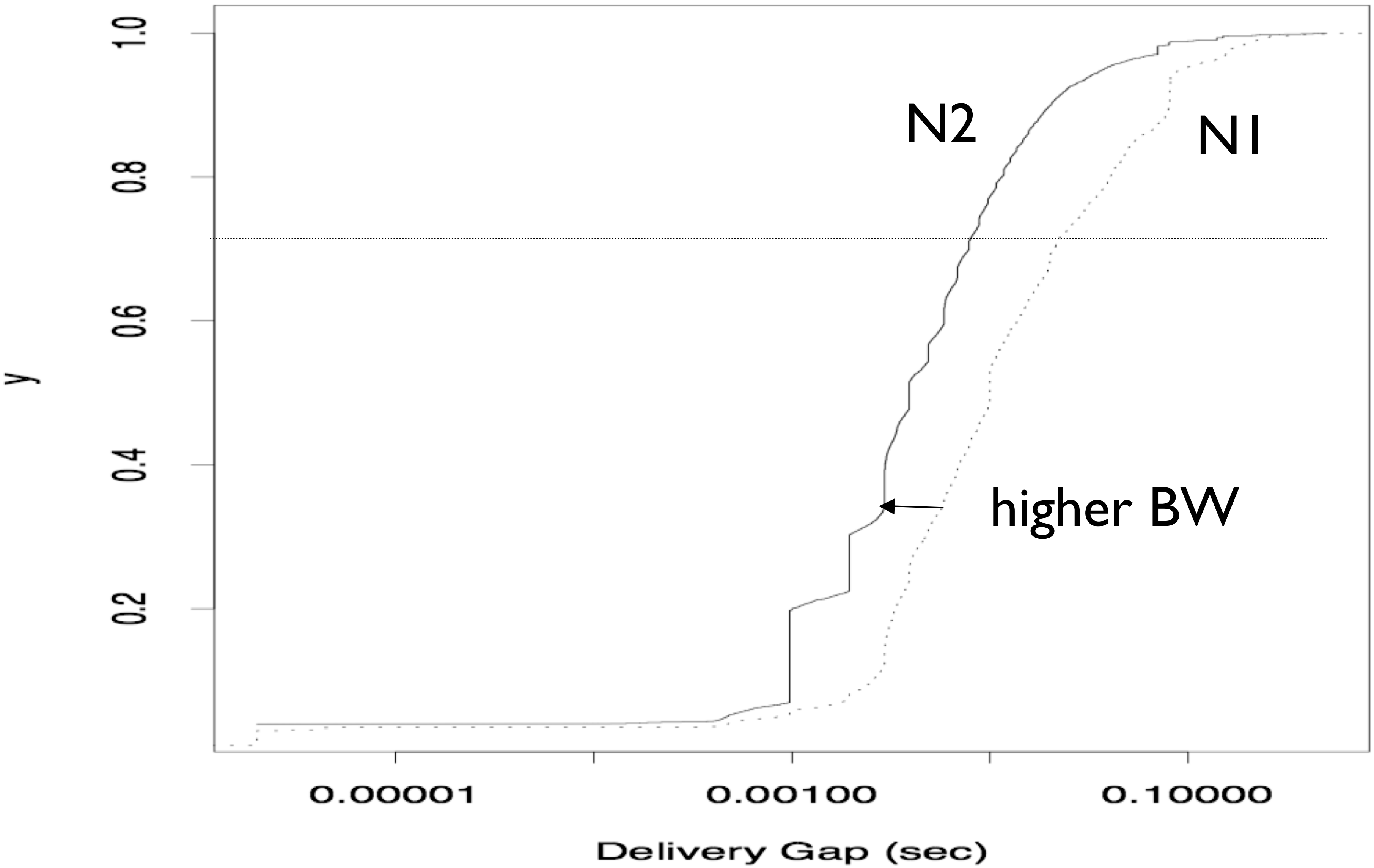
3

4

Delivery Gap:
time between receiving
an out-of-order packet and
the packet sent before it.



Taken from Paxson's PhD Thesis: CDF for delivery gap between reordered packets.


N2
N1
higher BW
y
1.0
0.8
0.6
0.4
0.2
0.00001
0.00100
0.10000
Delivery Gap (sec)

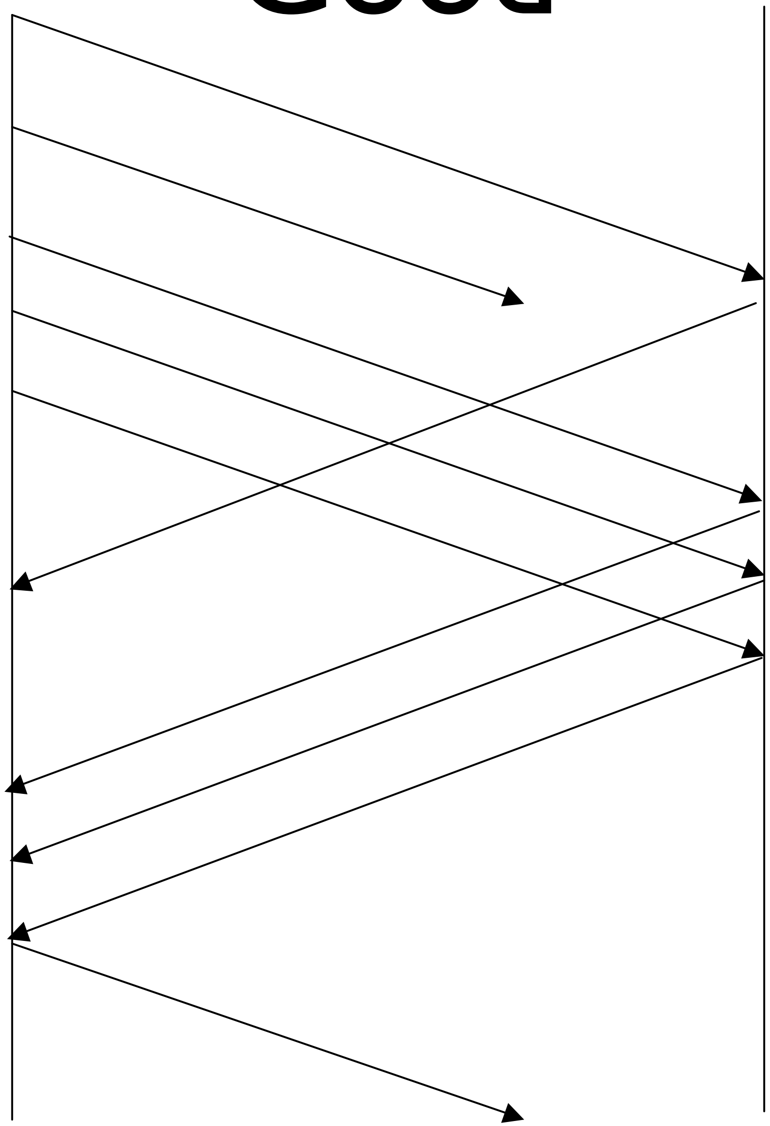
N1 N2

20ms **8ms**

Waiting time with which 70% of
out of order delivery would be identified.

Is needless retransmission a problem?

Good



N1	N2
22	**300**

Number of good retransmissions
for every bad retransmission.
$N_d = 3$, $W = 0$

N1	N2
~7	**100**

Number of good retransmissions for every bad retransmission.

$N_d = 2, W = 0$

N1	N2
15	**300**

Number of good retransmissions
for every bad retransmission.
$N_d = 2$, $W = 20$ms

Packet Corruption

1 in **5000**

packet is corrupted

1 in **65536**

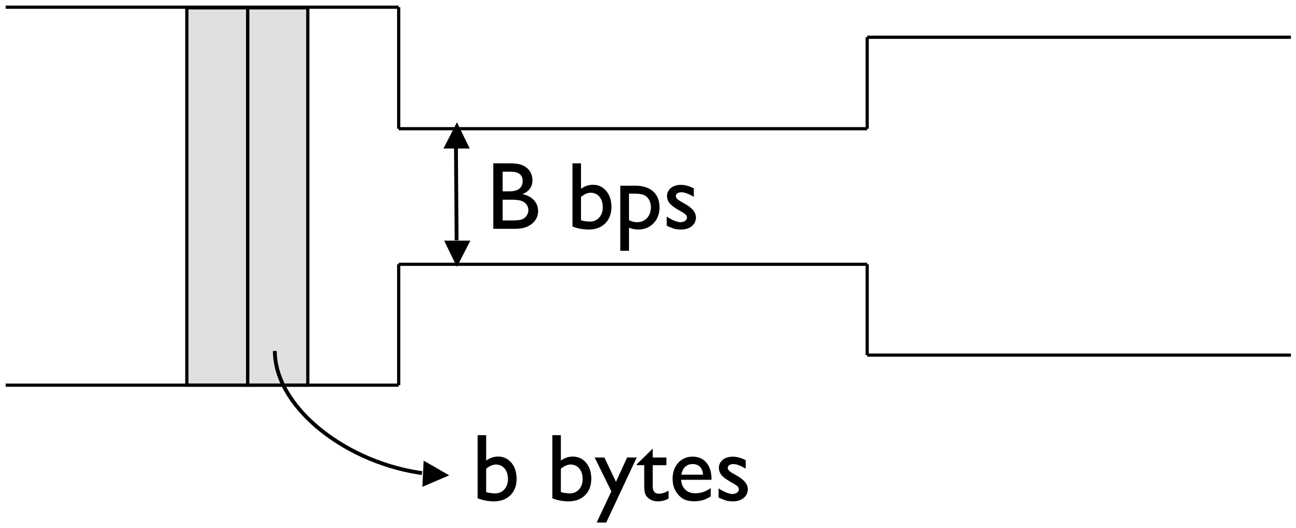
corrupted packet goes undetected
using TCP checksum

1 in **300**million

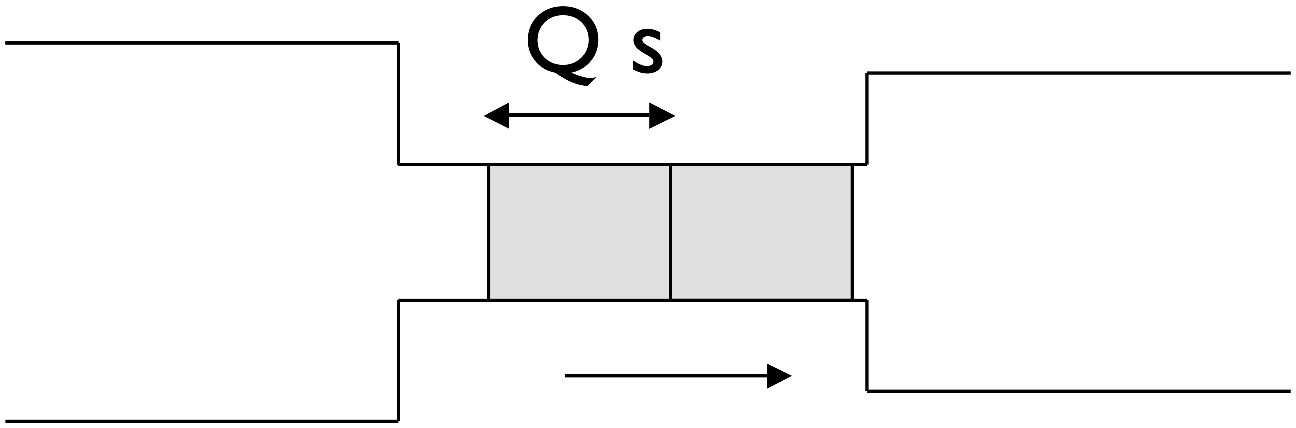
Internet packet is corrupted
and is undetected.

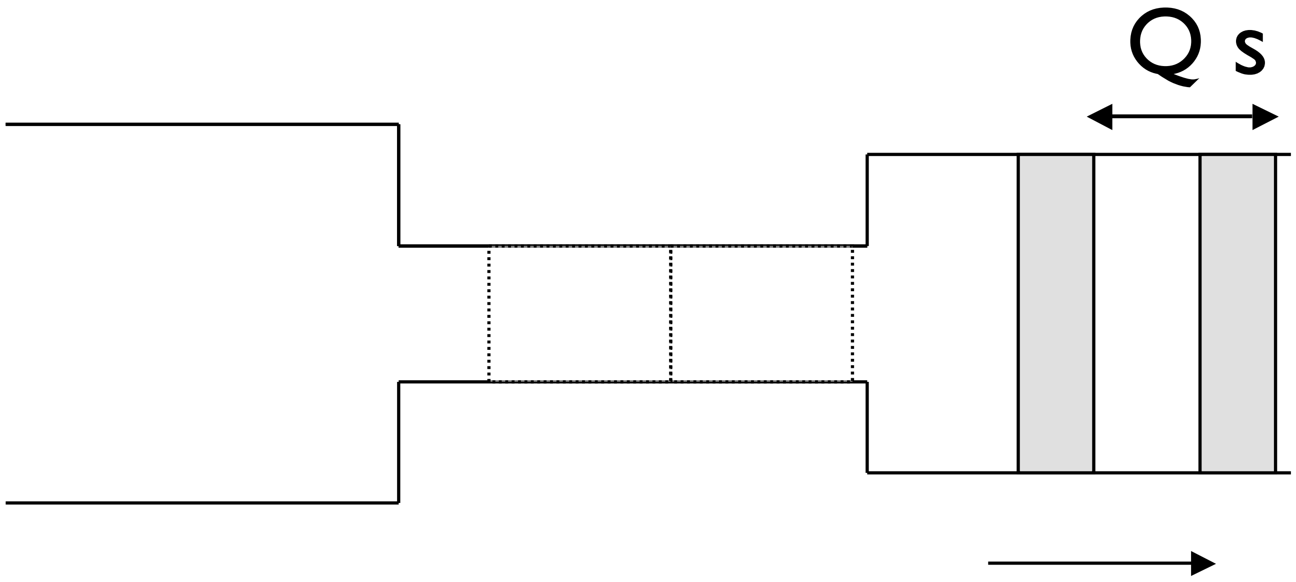
Part 2:
Bottleneck Bandwidth

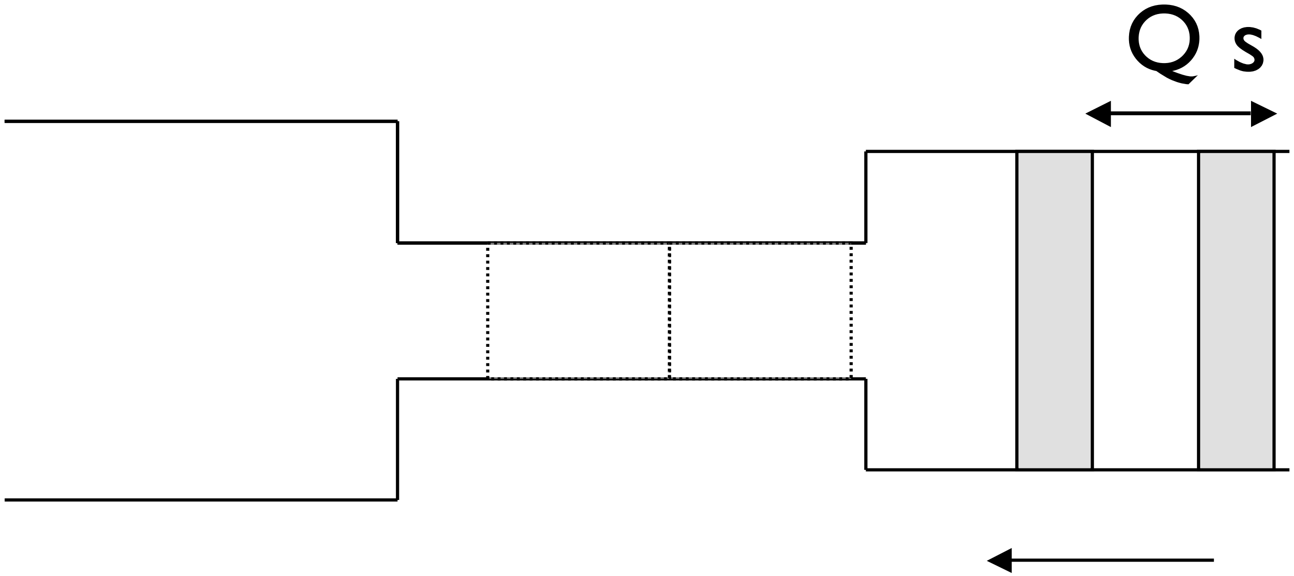
Packet Pair

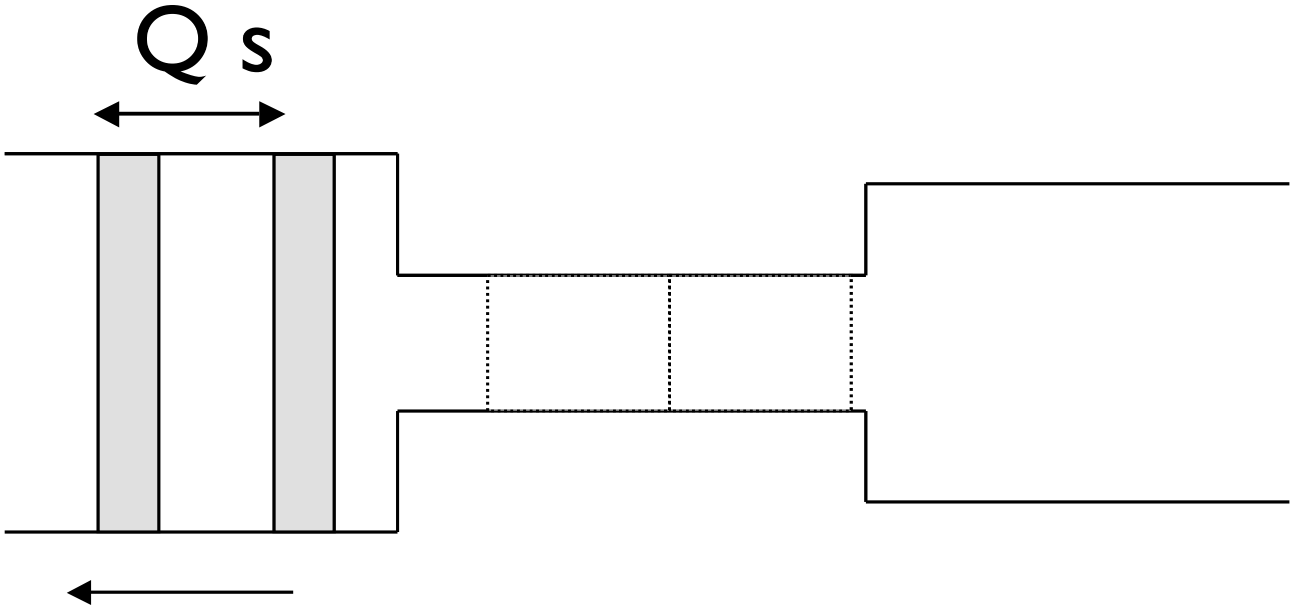

B bps
b bytes

$$Q \times B = b$$




Q s


Q s


Q s

Problems with Packet Pair

1. Asymmetric Link

2. ACK Compression

3. Out of order delivery

4. Clock resolution

5. Changing bottleneck bandwidth

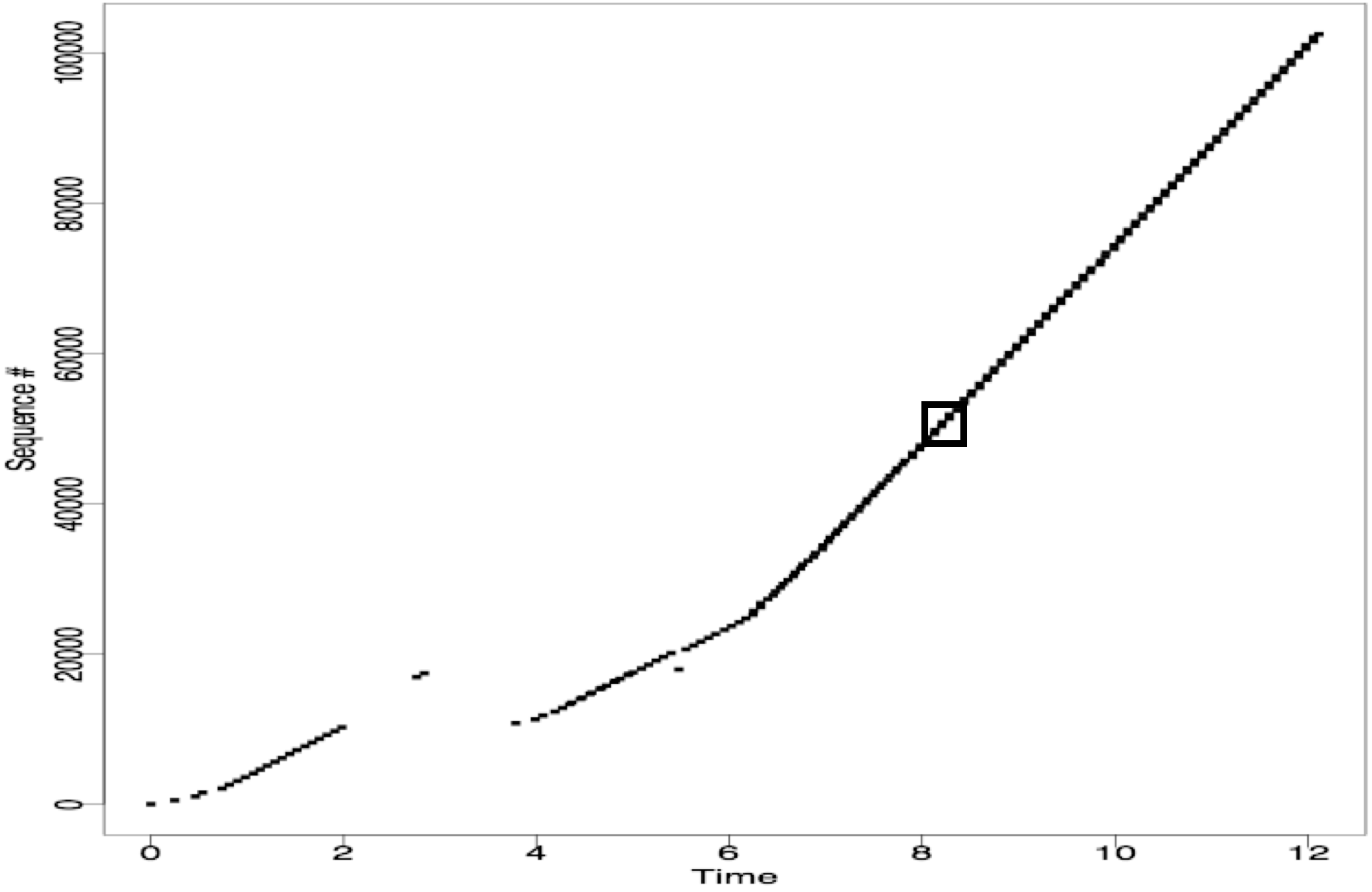


Fig 2 from the paper, showing changing bandwidth.

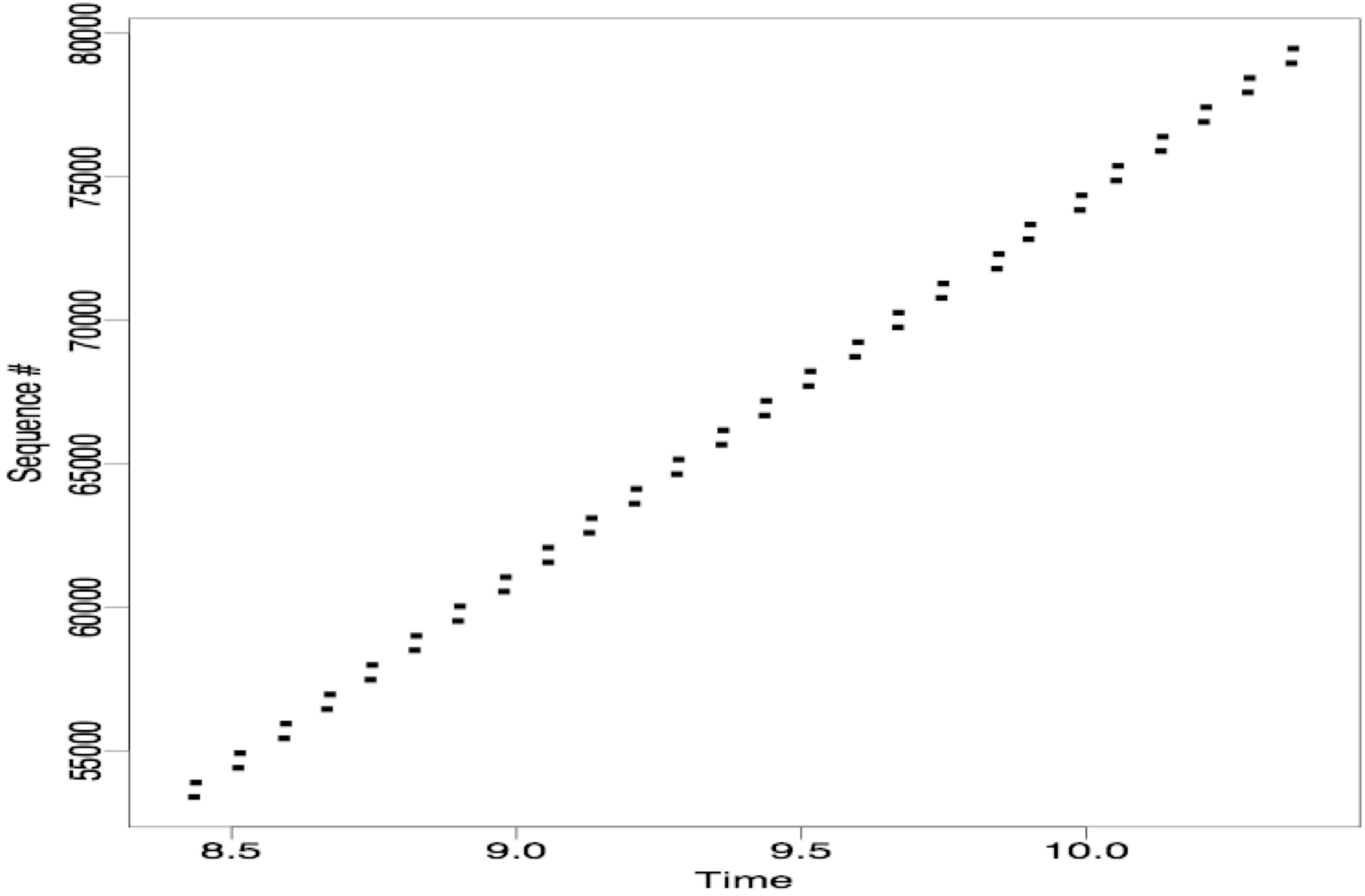


Fig 3 from the paper, showing multi-channel links.

6. Multi-channel Links

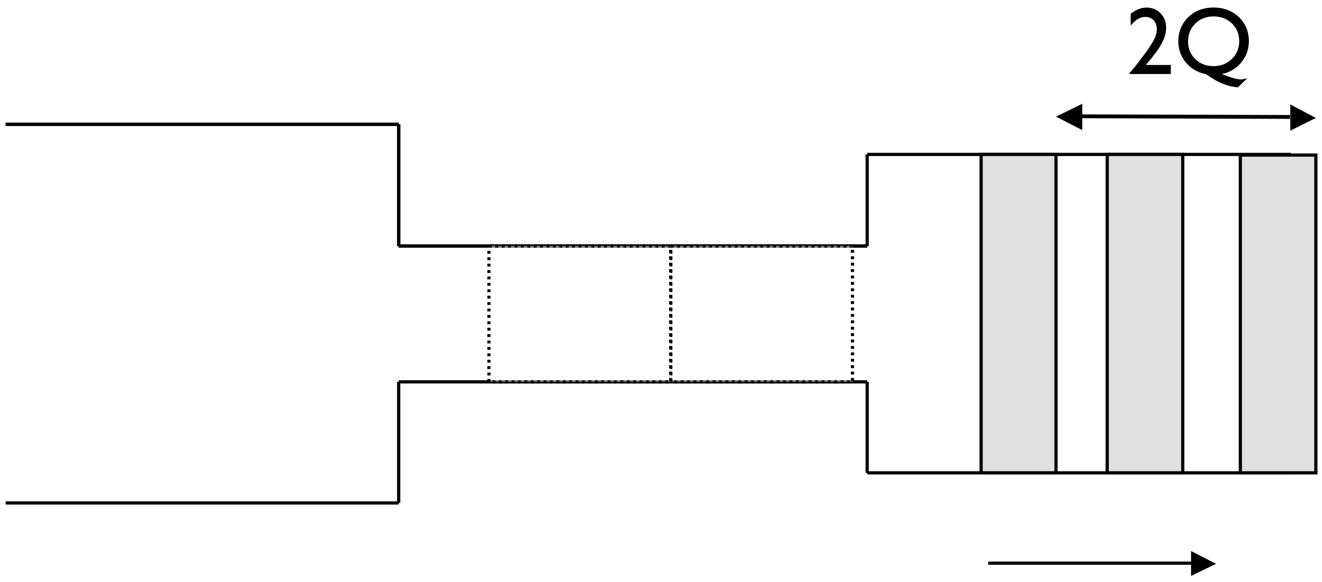
Asymmetric links
ACK compression
Out-of-order delivery
Clock resolution
Changes in bottleneck bandwidth
Multi-channel links

Measure at receiver:

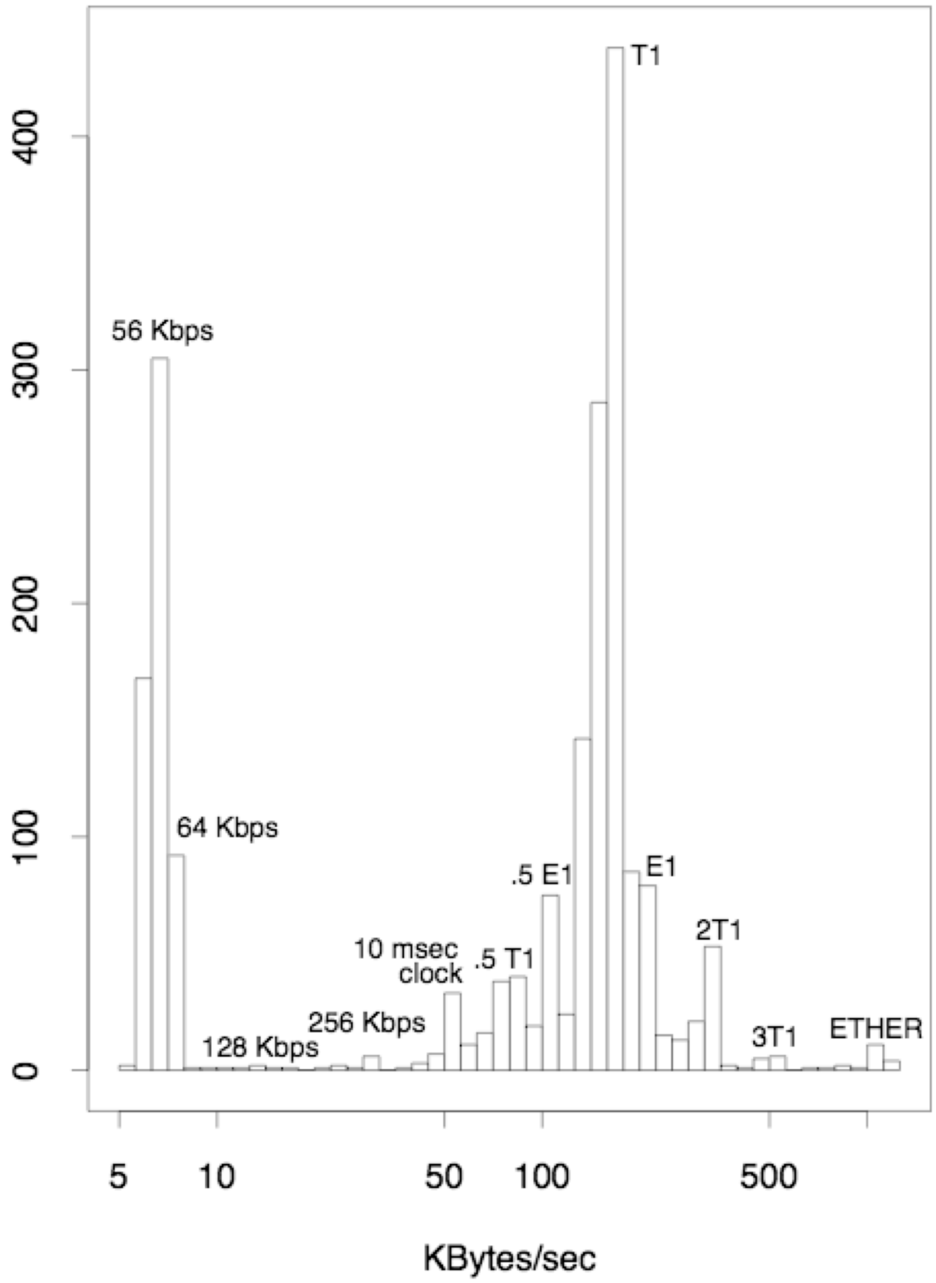
Asymmetric links
ACK compression

Packet bunch:

Out-of-order delivery
Clock resolution
Changes in bottleneck bandwidth
Multi-channel links


2Q

Collect multiple estimates, take the most freq occurrence (modes) as the bottleneck bandwidth.



400
300
200
100
0
56 Kbps
64 Kbps
128 Kbps
256 Kbps
10 msec
clock
.5 T1
.5 E1
T1
E1
2T1
3T1
ETHER
5
10
50
100
500
KBytes/sec

Part 3:

Packet Loss

N1	N2
2.7%	**5.2%**

Percentage of packets that were lost.

N1

50%

N2

50%

Percentage of loss free connections

N1	N2
5.7%	**9.2%**

Loss rate on lossy connections

17%

Loss rate on connections from EU to US

Are packet losses independent?

Compute:

P_u = Pr [p lost]

P_c = Pr [p lost | prev pkt lost]

P^u P^c

2.8% **49%**

Loss rate for “queued data pkt” on N1

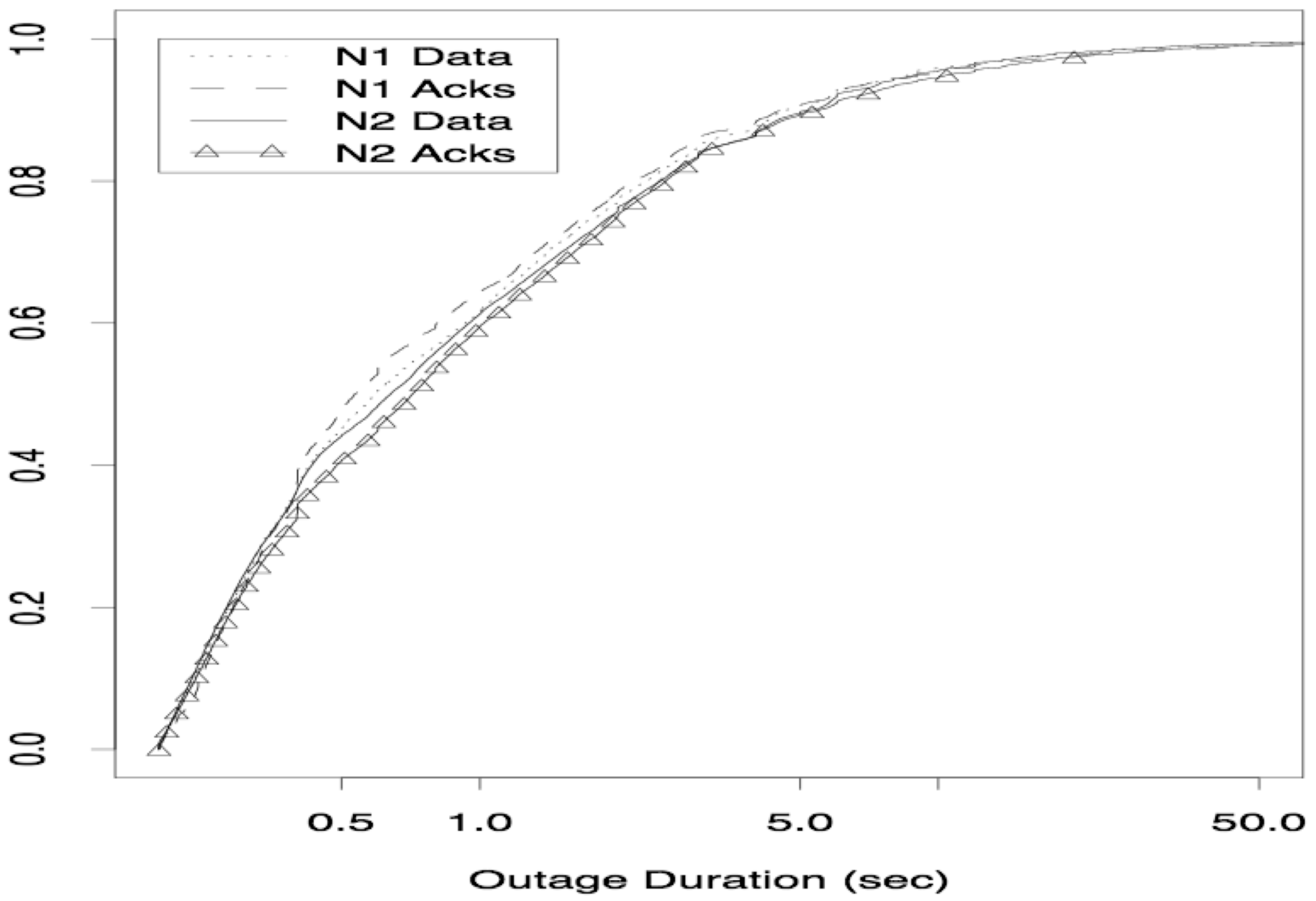
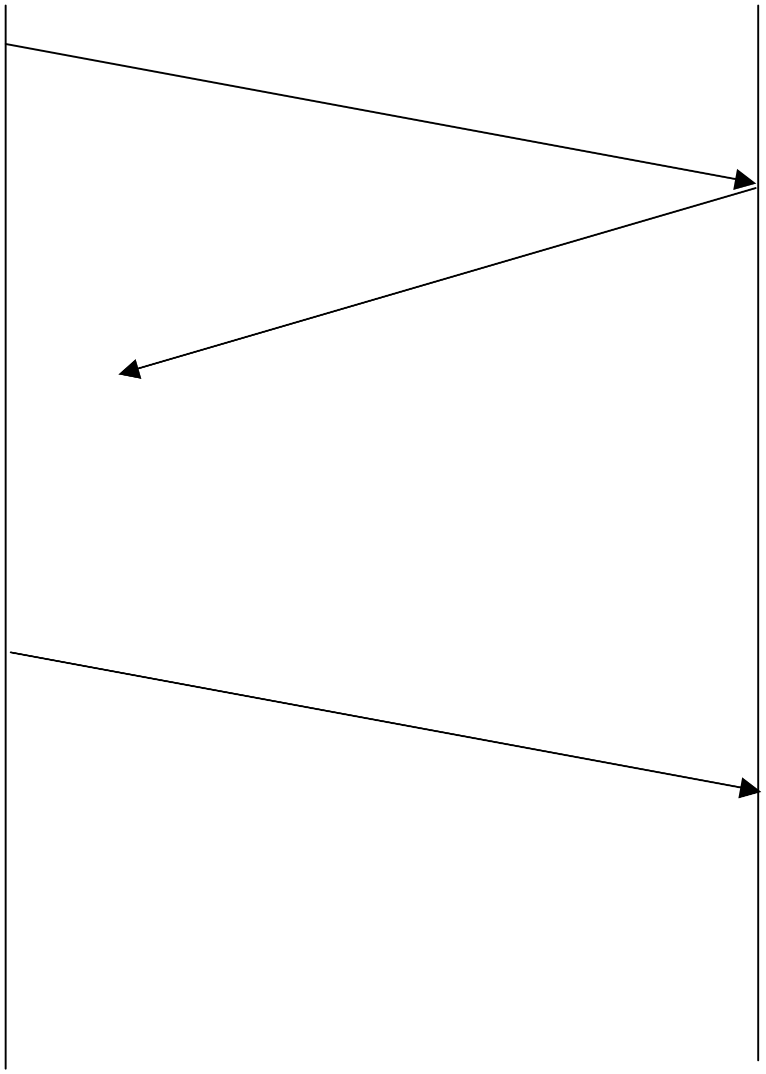


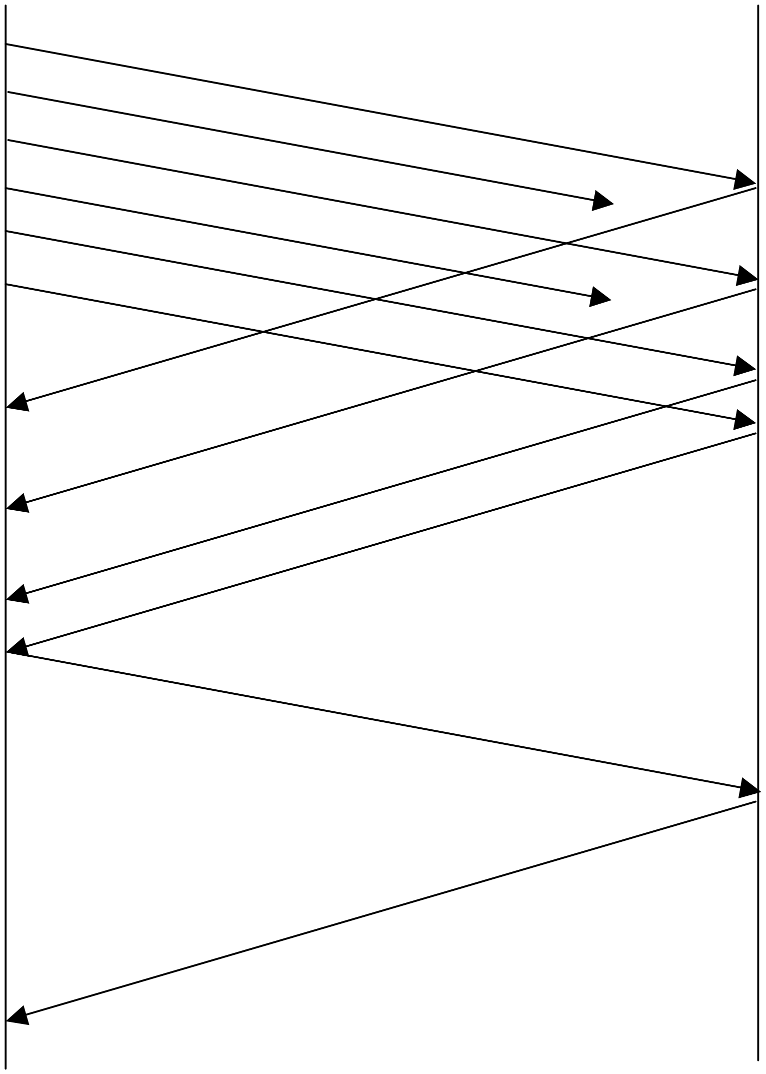
Fig 6 from the paper, showing outage duration.

Are retransmission redundant?

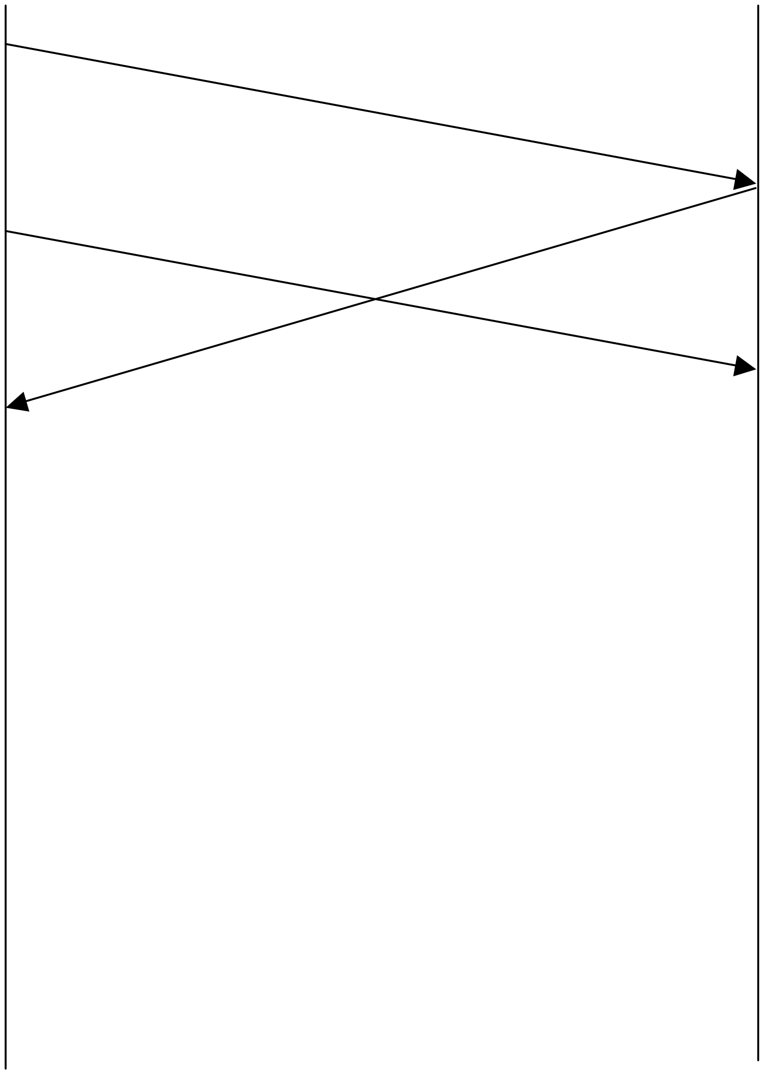
Unavoidable



Coarse Feedback



Bad RTO



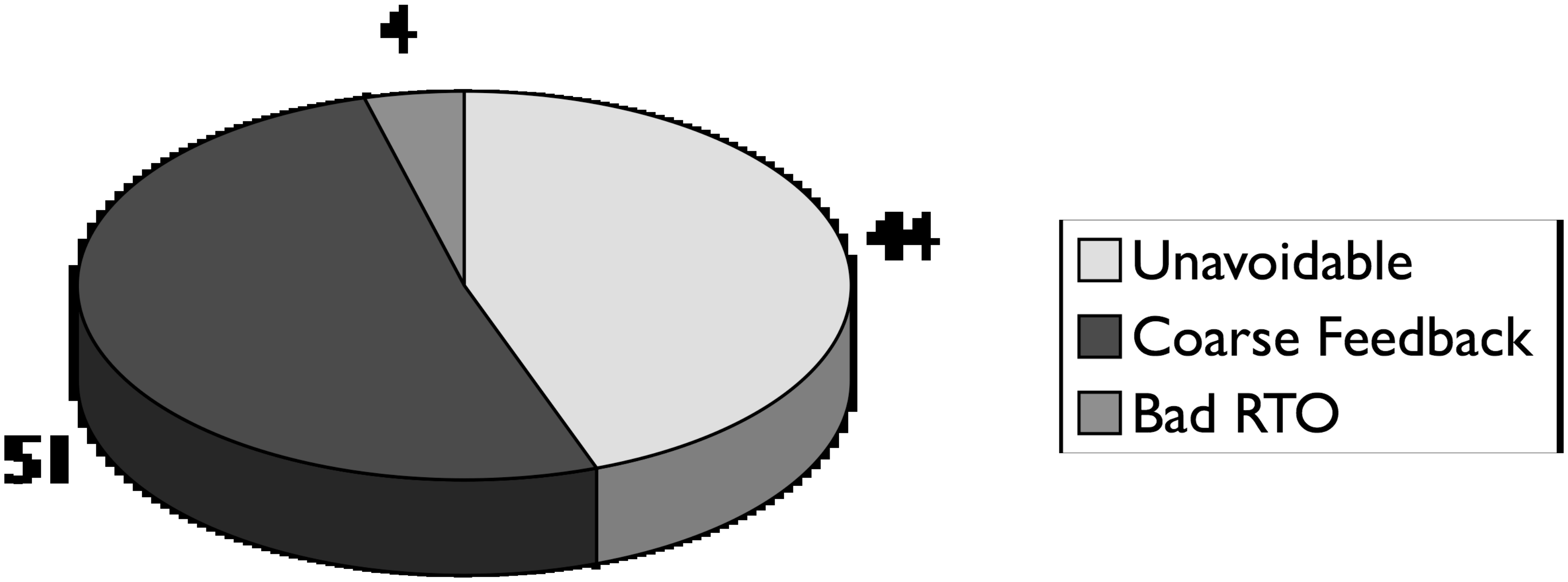
N1

26%

N2

28%

Percentage of retransmissions that are redundant



Type of redundant retransmission in N1.

Part 4:

Packet Delay

OTT is not well approximated as RTT/2

ACK Compression

ΔT_s Sending interval

ΔT_r Receiving interval

$$\frac{\Delta T_r}{\Delta T_s}$$

$$\xi = \frac{\Delta T_r + C_r}{\Delta T_s - C_s}$$

Compression

event if $\xi < .75$

N1	N2
50%	**60%**

Percentage of connection that experiences at least one compression event.

N1

50%

N2

60%

Percentage of connection that experiences at least one compression event.

2

Average number of events per connection.

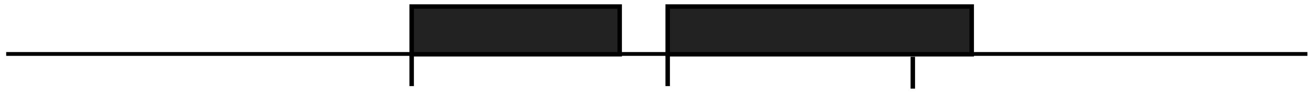
Estimating Available Bandwidth

Q_b: time to transit the bottleneck

ψ_i: expected time spent queuing behind predecessor (derived from sending time)

γ_i: diff between packet OTT and min OTT

T_i time packet *i* is sent



$$\psi_1 = 0$$

$$\psi_i = max\{(T_{i-1} + \psi_{i-1} + Q_b - T_i), 0\}$$

$$\beta = \frac{\sum_i(\psi_i + Q_b)}{\sum_i(\gamma_i + Q_b)}$$

$\beta = 1$ means all bandwidth is available.

$\beta = 0$ means none of the bandwidth is available.

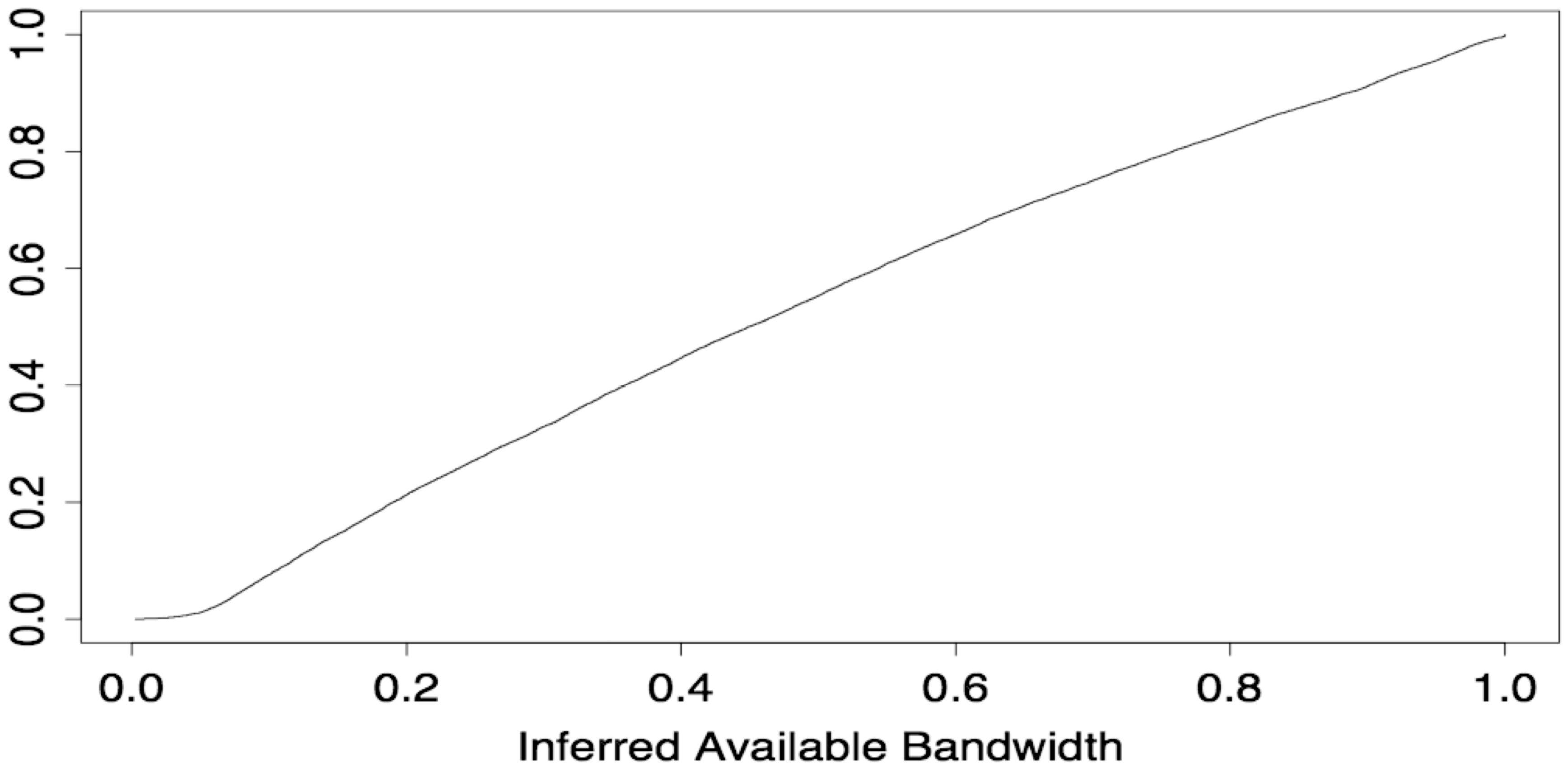


Fig 10 from the paper, showing distribution of available bandwidth.

All numbers in the paper is not important (the Internet has changed!).

Measurement is difficult but useful

Many new techniques needed (e.g to measure bottleneck bandwidth)

We can improve current design (e.g. TCP if we know more about reordering)

We can identify problem (e.g. packet corruption)

We can better model the behavior (e.g. bursty packet loss)

We can infer many info
from just a packet trace
(e.g. available bandwidth)