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?
I. 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
II. 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 transferred
21

Number of sites
Number of trace pairs
Part I: The Unexpected
Packet Reordering
2 reorderings
Percentage of connections with at least one out-of-order delivery

N1: 36%
N2: 12%
Percentage of data packets out-of-order

NI

2%

N2

.3%
Percentage of ACK packets out-of-order

NI: 0.6%
N2: 0.1%
Data packets are usually sent closer together.
From 15%  To .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?
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.
higher BW
Waiting time with which 70% of out of order delivery would be identified.
Is needless retransmission a problem?
Good
Number of good retransmissions for every bad retransmission. 

\[ N_d = 3, \; W = 0 \]
Number of good retransmissions for every bad retransmission.

\( N_d = 2, W = 0 \)
Number of good retransmissions for every bad retransmission.

\[ N_d = 2, \ W = 20\text{ms} \]
Packet Corruption
1 in 5000

packet is corrupted
I in \(65536\) corrupted packet goes undetected using TCP checksum
Internet packet is corrupted and is undetected.
Part 2:
Bottleneck Bandwidth
Packet Pair
\( Q \times B = b \)
Problems with Packet Pair
I. 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
Collect multiple estimates, take the most freq occurrence (modes) as the bottleneck bandwidth.
Part 3: Packet Loss
2.7%  5.2%

Percentage of packets that were lost.
Percentage of loss free connections

N1: 50%
N2: 50%
Loss rate on lossy connections

N1: 5.7%
N2: 9.2%
17%  

Loss rate on connections from EU to US
Are packet losses independent?
Compute:

\[ P^u = \Pr [ \text{p lost} ] \]

\[ P^c = \Pr [ \text{p lost} \mid \text{prev pkt lost} ] \]
$P_u$ 2.8% $P_c$ 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
Percentage of retransmissions that are redundant

N1 26%  N2 28%
Type of redundant retransmission in N1.
Part 4:
Packet Delay
OTT is not well approximated as $\text{RTT}/2$
ACK Compression
$\Delta T_s$ Sending interval

$\Delta T_r$ Receiving interval
\[ \frac{\Delta T_r}{\Delta T_s} \]
\[ \xi = \frac{\Delta T_r + C_r}{\Delta T_s - C_s} \]
Compression event if $\xi < 0.75$
N1  50%  N2  60%

Percentage of connection that experiences at least one compression event.
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

$\psi_i$: expected time spent queuing behind predecessor (derived from sending time)

$\gamma_i$: diff between packet OTT and min OTT
$T_i$  \hspace{1cm} \text{time packet } i \text{ 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)