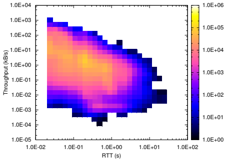
On the relationship between fundamental measurements in TCP flows

Or when is TCP not TCP?



Richard G. Clegg (richard@richardclegg.org),João Taveira Araújo, Raul Landa, Eleni Mykoniati, David Griffin, Miguel Rio,
University College London, Department of Electronic Engineering

Talk to Coseners 2013

(Prepared using LaTeX and beamer.)

Studying TCP in the wild

Mathematical starting point

Padhye et al – bandwidth (throughput) of TCP flow at equilbrium:

$$T = \frac{1}{D}\sqrt{\frac{3}{2bp}} + o(1/\sqrt{p}),$$

where D is RTT (delay), p is the probability of packet loss and b is a fixed TCP parameter.

- Result (simplified version presented) is from mathematical model with many assumptions.
- Subsequent work generalises and improves – basic inverse dependence on RTT and $\sqrt{p}$ remain fundamental.
- Beautiful mathematically, how is it statistically?
- Is TCP doing what we think it does and if not, why not.

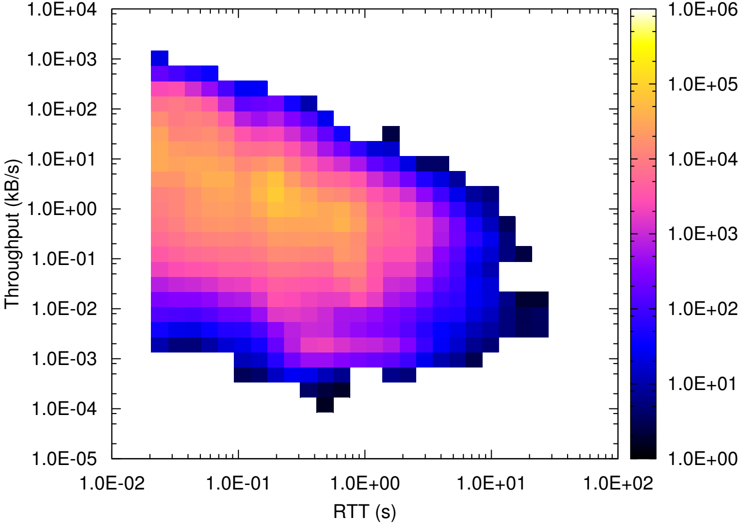
Data and analysis approach

- Basic approach – use lots of freely available packet traces.
- Test both diverse data sets and similar data sets.
- Reconstruct TCP flows – calculate RTT, loss etc. Fit formulae relating these quantities.
- Data used CAIDA (US based data) MAWI (Japanese based data):
  - CAIDA OC48 Traces (2002) — 3 hours of data: 1.4 billion packets originally 876GB of data.
  - CAIDA OC192 (2011A) — 26 minutes of data: 1.3 billion packets originally 662GB of data.
  - CAIDA OC192 (2011B) — 14 minutes of data: 0.927 billion packets, 582 GB of data.
  - CAIDA OC192 (2012) — 29 minutes of data 1.6 billion packets and 1,120 GB of data.
  - MAWI (2006–2012) — 15 minute samples once per month, 1.36 billion packets and 982 GB of data.

Fundamental relationships within TCP flows

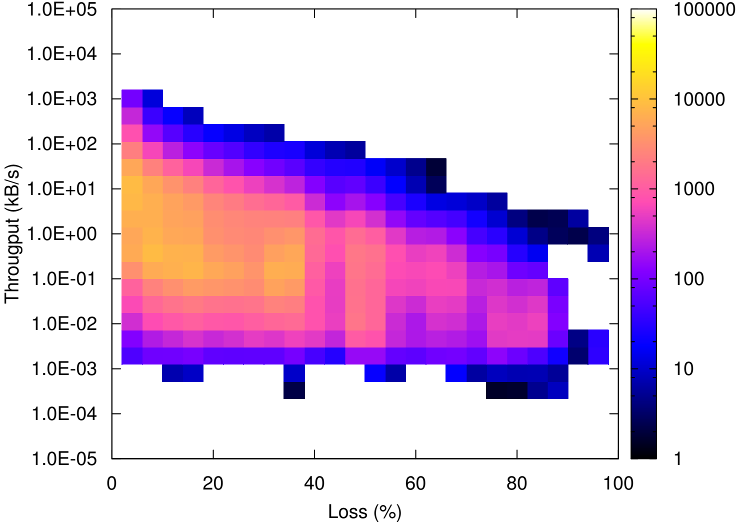
- In reality very little TCP is really TCP in the old-fashioned sense.
- TCP can be application limited (YouTube).
- TCP can be sender or receiver window limited.
- TCP can be limited by middleboxes.
- Ignoring all of this, what is the best relationship which ties network parameters to TCP performance.
- Step 1: graphically investigate the relationships in the data sets.
- Step 2: statistically fit equations which relate the parameters: throughput, loss, RTT, flow length.
- Step 3: attempt to classify flows by "cause" of delay.

Visualising correlations throughput/RTT



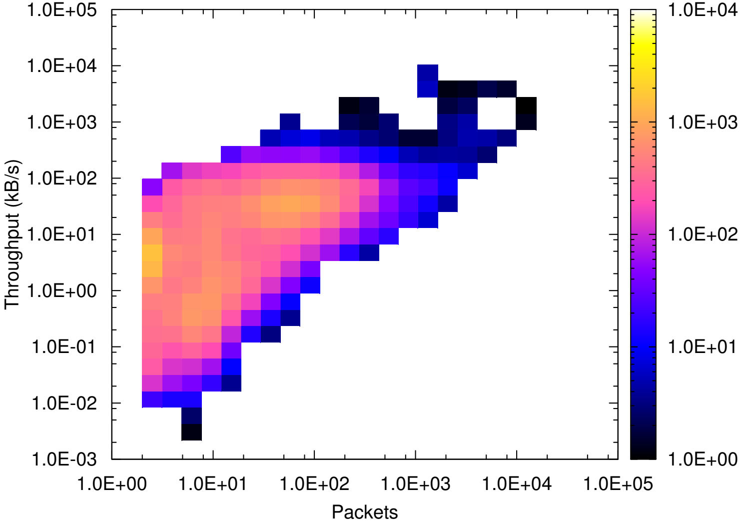
OC48 — relationship between throughput and RTT

Visualising correlations throughput/loss



MAWI — relationship between throughput and loss

Visualising correlations – throughput/packets



OC192 2012 — relationship between throughput and number of packets in flow

Fitting a Linear Model

- Use common statistical technique of linear model fitting.
- Fit log of data and use exponential transform to get $T = \beta_0 D^{\beta_1} p^{\beta_2} \varepsilon'$ where ε' is mean 1, lognormal).
- With $\beta_1 = -1$ and $\beta_2 = -0.5$ this is $T = \beta_0 / D\sqrt{p}$ (and error term).
- Throw in a lot of data from TCP flows, fit the best β_i for various models.
- Goodness of fit judged by R^2 value where $R^2 = 1$ is perfect and $R^2 = 0$ is no fit at all (amount of variance "explained" by model).
- Taking logarithms a problem for loss as sometimes $p = 0$ – use instead $\log p + p_m$ where p_m is a fitted offset parameter.
- Standard calibrate, cross-validate, test statistical methodology used.

CAIDA OC192 2012 data

Model for T	R^2	Note
$15.7D^{-0.94}(p+p_m)^{-0.563}P^{0.456}$	0.641	$p_m = 0.105$
$77.2D^{-0.975}P^{0.455}$	0.635	
$316/(D\sqrt{p+p_m})$	0.0227	$p_m = 0.105$

- Excellent fit to data.
- Loss p slightly improves model but not much.
- Best model is approx $T = k\sqrt{P}/D$ where k is constant.

Model summaries

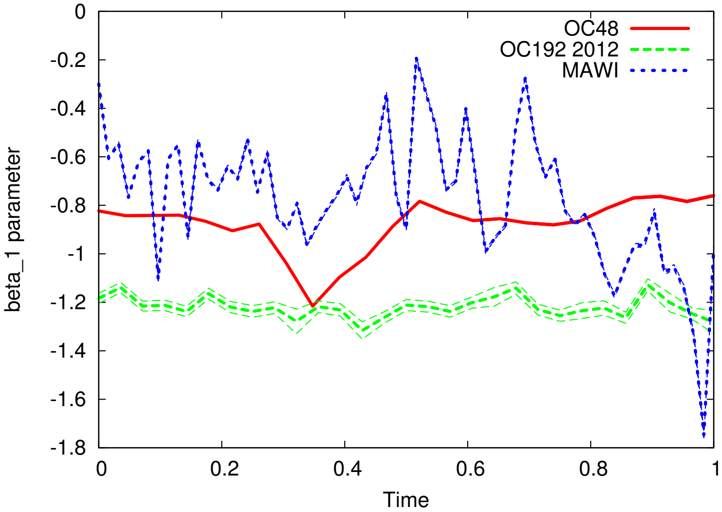
P flow length in packets, D delay (RTT), T throughput, p loss.

Data	Model for T	R^2
OC48	$29.7 D^{-0.89} P^{0.354}$	0.35
OC192 2011A	$4.62 D^{-0.698} P^{0.41}$	0.448
OC192 2012B	$156 D^{-0.981} P^{0.386}$	0.611
OC192 2012	$77.2 D^{-0.975} P^{0.455}$	0.635
MAWI	$1.65 D^{-0.711} P^{0.558}$	0.261
MAWI w loss	$0.15 D^{-0.664} (p + p_m)^{-0.416} P^{0.635}$ (*)	0.282

(*) $p_m = 0.0132$

- Summary – mostly surprisingly good fit from simple model
- Best model approximately 1/RTT and square root of number of packets
- RTT and length predicts throughput well – loss rarely useful

Parameter dynamism

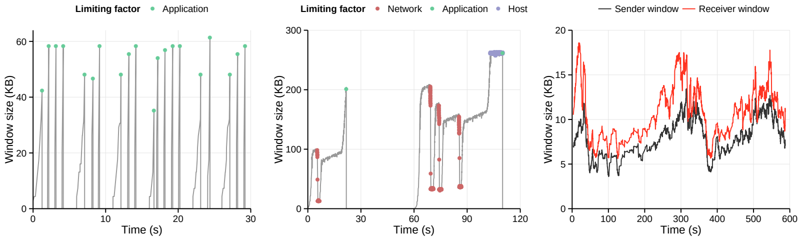


Evolution of β_1 parameter in model $T = \beta_0 D^{\beta_1}$ across normalised time

Why doesn't TCP fit the Padhye model?

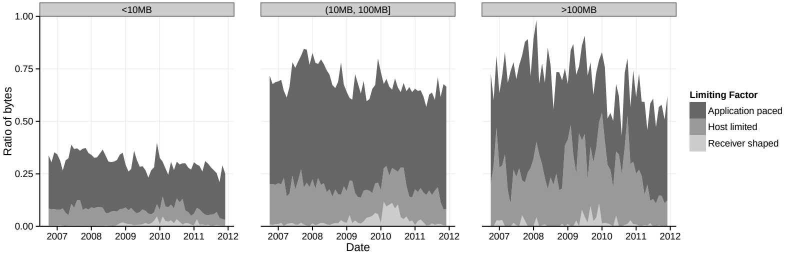
- Main reason I am knowingly misapplying the model (not steady state, loss should be average over all flows etc).
- However, other mechanisms interfere with TCP behaviour:
  1. Application paced – think youtube. Sender limits flow by only sending limited amounts to throttle their bandwidth.
  2. Host limited – sender or receiver have limited maximum window size.
  3. Receiver shaped – receiver or middlebox manipulates advertised window size.
- These mechanisms were investigated by classifying flows according to the type of limit on bandwidth.

Classifying TCP



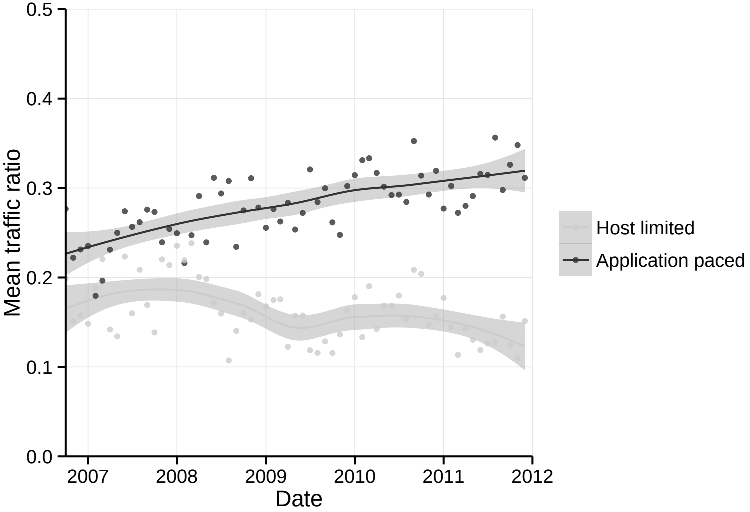
- Application paced – look for flights with pauses between them.
- Host limited – look for hard “ceiling” in window size.
- Receiver shaped – look for correlation between sender + receiver window when no loss observed.

MAWI data by limitation type



Note that for smaller flows it may be simpler harder to identify a limitation the default assumption is none.

MAWI data by limitation type



Application limiting is a growing trend.

Conclusions and further work

- Very very simple models of TCP throughput are often surprisingly good.
- If you know the RTT it would be relatively simple to produce a good estimate of flow completion time in real time.
- Length of flow is very important to throughput as is delay.
- Packet loss does not have as significant an impact – even though it was often high.
- Most TCP (in the MAWI data) is not what we think it is.
- The majority of TCP flows are not limited by loss or delay.
- TCP is not doing what we tell people it is – it has been repurposed deliberately or accidentally.

Questions?

?