End-to-End Available Bandwidth: Measurement Methodology, Dynamics, and Relation with TCP Throughput

Manish Jain
Constantinos Dovrolis
SIGCOMM 2002

Presented by:
Honggang Zhang
Talk Overview

- Capacity and available bandwidth (avail-bw)
- Avail-bw estimation methodology (SLoPS) and tool (Pathload)
- Verification of Pathload
- Using Pathload to examine avail-bw variability
Capacity and Available bandwidth

- Path capacity $C$: maximum possible end-to-end throughput. It is defined as $C = \min_{i=0,...,H} \{C_i\}$, where $C_i$ is capacity of link $i$.

  **Narrow Link**: the link with minimum capacity

- Avail-bw: spare capacity in the path. Maximum end-to-end throughput given cross traffic load. It is a time-varying metric, defined as average over certain time interval.

  **Tight Link**: the link with minimum available bandwidth.

[Diagram of network flow with capacities and available bandwidths]
Definition of avail-bw

- $u_i$ : average utilization of link $i$ in a time interval of length $\tau$ ($0 \leq u_i \leq 1$)
- **Avail-bw of link $i$:** $A_i = C_i \cdot (1-u_i)$

**End-to-end avail-bw:** $A = \min_{i=0 \ldots H} A_i = \min_{i=0 \ldots H} C_i(1-u_i)$

- Time interval length $\tau$ : averaging timescale

- **Avail-bw is limited by tight-link**
Previous work on avail-bw estimation

- Measure throughput of bulk TCP transfer
  - A bulk TCP’s throughput is not avail-bw.
  - TCP saturates path (i.e., intrusive measurements)
- Carter & Crovella: dispersion of long packets trains (cprobe)
- Ribeiro et al.: estimation technique for single-queue paths
- Melander et al.: attempt to estimate capacity & avail-bw of every link in path
Self-loading Periodic Streams (SLoPS)

- **SND** sends a periodic UDP packet stream of rate $R$
- Stream characteristics: $K$ packets, size $L$, period $T$, rate $R=L/T$
- **RCV** Measured One-Way Delay (OWD): $D^K = T^{RCV}_{\text{arrive}} - T^{SND}_{\text{send}}$
- OWD variation: $\Delta D^k = D^{k+1} - D^k$ (independent of clock offset)
- With a stationary & fluid model for the cross traffic, and FIFO queues:
  - If $R > A = \min A_i$, then $\Delta D^k > 0$ for $k=1,\ldots,K-1$
  - Else, $\Delta D^k = 0$ for $k=1,\ldots,K-1$
Illustration of basic idea

• Periodic stream: K packets, period T, packet size L, rate: \( R = \frac{L}{T} \)
Increasing delay trend: R > A

- Path: Univ-Oregon to Univ-Delaware (12-hops)
- A=73Mbps (MRTG), R=96Mbps (K=100 packets, T=100μs, L=1200B)
Non-increasing delay trend: $R < A$

- Path: Univ-Oregon to Univ-Delaware (12-hops)
- $A=74$Mbps (MRTG), $R=37$Mbps ($K=100$ packets, $T=100\mu s$, $L=462$B)
Iterative rate adjustment to measure $A$

- **Source**: send $n$-th periodic stream with rate $R(n)$
- **Receiver**: measure delays $D^k$ for $k=1\ldots K$
- **Receiver**: check for increasing delay trend, notify source
- **Source**:
  - If delays show increasing trend ($R(n) > A$), $R_{\text{max}} = R(n)$;
  - If delays show non-increasing trend ($R(n) < A$), $R_{\text{min}} = R(n)$;
  - $R(n+1) = (R_{\text{max}} + R_{\text{min}})/2$;

- **Exit when** $R_{\text{max}} - R_{\text{min}} \leq \omega$ ($\omega$: estimate resolution)
Rate-adjustment Algorithm

In actual implementation: a fleet of N streams sent out at time $n$ to infer if $R(n)>A$, $R(n)<A$, or $R(n) \approx A$. Then, the iterative algorithm determines the rate $R(n+1)$ of the next fleet.

- One Stream $V=KT$
- Interval $\Delta$ between streams: $\max \{ RTT, 9V \}$
- N streams in a fleet at a single iterative step
  - $N_{\text{default}} = 12$

Measurement Latency? Time scale?
K_{\text{default}}=100$, if $L=800$B, $T=100\mu$sec, a stream lasts 10msec.
Using default parameters, if $A \approx 100$Mbps, $\Delta=100$ms, the tool takes 15 seconds to converge.
Rate-adjustment Algorithm, Grey-region, and avail-bw variability

- Measurement stream rate can fall into avail-bw variation range

Pathload reports grey-region boundaries \([G_{\text{min}}, G_{\text{max}}]\)

- Relative width of grey-region: quantify avail-bw variability
Detection of increasing trend in a single stream

Pairwise Comparison Test (PCT)

\[
PCT = \frac{\sum_{k=2}^{K} I(D^k > D^{k-1})}{K - 1}
\]

\[0 \leq PCT \leq 1\]

Pairwise Difference Test (PDT)

\[
PDT = \frac{D^K - D^1}{\sum_{k=2}^{K} |D^k - D^{k-1}|}
\]

\[-1 \leq PDT \leq 1\]
Experiment verification

- From Univ-Oregon to Univ-Delaware
- Tight link: U-Oregon GigaPoP link (C=155Mbps)
- Compare Pathload estimate (average of consecutive runs for 5 mins) with 5-min average avail-bw from MRTG readings.

![U-Oregon to U-Delaware availability graph](chart.png)
Pathload: latency and intrusiveness

• For RTT=100msec and A≈100Mbps, Pathload takes approx 15 seconds to converge

• Pathload does not cause:
  – Significant reduction in avail-bw (less than 10%)
  – Significant increase in queuing delays

• It is not intrusive: does not cause significant increases in network utilization, delays, or losses.

• To achieve non-intrusiveness:
  – Short measurement streams (K=100)
  – Introduce delay (‘silence period’) between streams
Avail-bw variability versus traffic load

- Relative variation index: \( \rho = \frac{R_{\text{max}} - R_{\text{min}}}{(R_{\text{max}} + R_{\text{min}})/2} \)

![Graph showing CDF of \( \rho \).](image)

C=12.4Mbps.

110 runs.

- Heavier tight link utilization leads to higher avail-bw variability
Avail-bw variability versus stream length

CDF of $\rho$.

$C=12.4\text{Mbps}$.

The stream duration for $R=A(=4.5\text{Mbps})$, $L=200\text{B}$, $T=356\mu\text{s}$ is:

- $18\text{ms}$ for $K=50$, $36\text{ms}$ for $K=100$, $180\text{ms}$ for $K=500$

- Longer probing streams observe lower avail-bw variability
- But also, longer streams can be more intrusive
Future directions

- Sensitivity analysis for several Pathload parameters
- Apply avail-bw estimation in high-throughput TCP bulk transfers
- Apply avail-bw in overlay network routing optimizations
- Pathload is currently available at www.pathrate.org
Comments

• What can we take away from this paper?
  – Using binary search to find out the avail-bw by sending out probing packets.
  – More … ?

• What I like about this paper?
  – Basic idea is simple and easy to be implemented.
    Looking at the trend of delays of a periodic stream.
  – Algorithm is well designed.
  – Actual experiments to verify methodology.
  – Pathload is used to estimate the variability of avail-bw.
  – More …. ?
Questions

• Not intrusive?
  – Only gives a single experiment. Difficult to justify.
  – How about if lots of users are using pathloads?

• Almost all parameters are empirical.
  – Could be difficult to tune them under different scenarios.
  – Difficult to draw general conclusions.

• Difficult to predict converge time.
  – In their reported experiments, converge time for a single fleet of streams is [10, 30] seconds.

• Works well when there is only one tight link.

• Stationary assumption.

• More…. ?
Some interesting research problems?
Thank you!