Cross-Layer Modeling

Performance and Capacity Planning

Transport Layer

- Internet protocol suit has layering architecture
- Transport layer performs end-to-end control
- Provides distributed control of networking infrastructure
- Data delivery control, flow control, congestion avoidance, etc.

Transport Layer

- TCP has important assumptions accepted by default for 'wire epoch'
- Data loss mean congestion, since it happens due to the routers tail or AQM dropping (AIMD)
- Large RTT deviation means heavy congestion of the end-to-end path (Back of and Slow Start)
- Data are delivered or lost but almost never corrupted
- Data disorder and delay is preferable, data loss should be recovered

Transport Layer

- Transport layer decide WHICH data and WHEN to inject in the network
- TCP (more then 90% of all flows), UDP
- Transport layer form key characteristics of data flow

Transport Layer

- Wire channels are:
  Reliable (loss rate 10^-16)
  High bandwidth
  Stability
- Wireless channels are:
  Unreliable (loss rate 10^-2 – 10^-6)
  Low bandwidth
  Instability (Outage)

Transport Layer

- Reality of wireless links
- Data loss **may** signal congestion
- Significant change in RTT **may** signal congestion
- Channel performance varies during connection lifetime
- For multimedia application loss is better then delay and out of order delivery

Transport Layer

- Data sent by transport protocol experience features of lower layers
- TCP recognizes lower level events and interpret them as a feedback
- Model of transport layer performance is cross-layer

TCP performance state-of-art

- TCP is sophisticated instance
- TCP realizes several algorithms (Slow Start, Congestion Avoidance, Fast Retransmit and Recovery, Exponential back off etc.)
- TCP performance is a function of many parameters (segm. loss probability, receiver adv. window, e2e path bottleneck throughput and workload, RTT distribution etc.)

TCP performance state-of-art

- 'Root square' low formulae used in IETF documents.
- Simple formula (S. Floyd) Only Congestion Avoidance Algorithm. No back off and timeouts.

$$T \approx \frac{C}{\sqrt{p}}$$

TCP performance state-of-art

- Another 'root square' formula which counts timeouts

$$T = \frac{MSS}{RTT\sqrt{1.33p} + RTOp(1+32p)\min\left[1,3\sqrt{.75p}\right]}$$

For simple formula

$$C = 0.93\frac{MSS}{RTT}$$

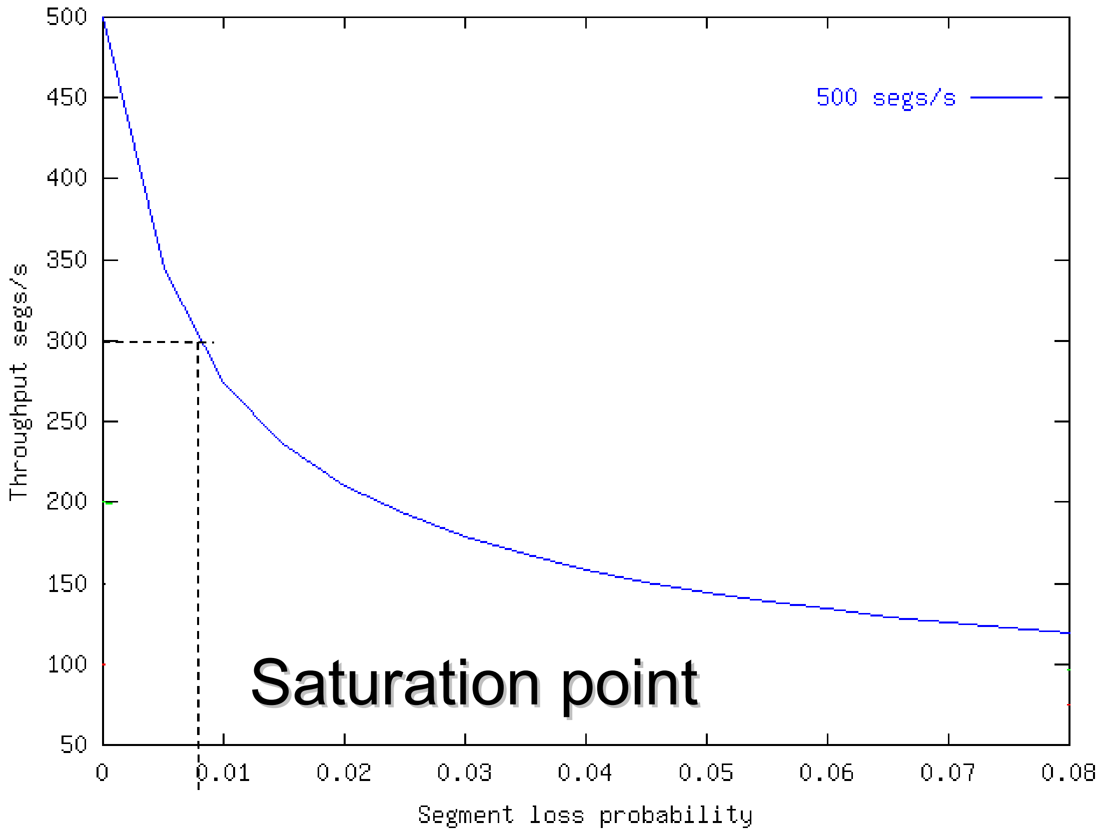
TCP performance state-of-art

- ‘Root square’ lows break down as segment loss probability *p* approaches to zero
- Researches are topical
- For other models see ACM and IEEE publications

Capacity planning at transport

- Define input parameters
- Use formulae and get throughput forecast
- Use the forecast for capacity planning process

Capacity planning at transport



Capacity planning example

- Designer invents new wireless link which losses 1% of IP packets
- This link layer is used in the internet on a path which otherwise had RTT of 80 msec
- Designer is interested in understanding performance of TCP over this link
- WLAN gives 2 Mbit/s

Capacity planning example

- Define MSS=1000 bytes (remove 40 bytes for TCP/IP headers)
- Define RTT=120 msec (80 msec for internet part, plus 20 msec each way for the new wireless links)
- Define p=0.01

Capacity planning example

- Simple formula gives

$$T=\frac{0,93\cdot 1000\cdot 8\text{bits}}{0,120\text{sec}\cdot\sqrt{0,01}}=620\text{kbit/s}$$

- More complex one provides

$$T=402.9\text{kbit/s}$$

Capacity planning example

- Equation of TCP performance are expressed in TCP segments
- Sub-network designers use bit-error ratio
- If channel bit-error are independent
  p = 1 - ([1 – BER]^[FRAME_SIZE*8])
- Here we assume FRAME_SIZE is in bytes and "^" represents exponentiation. It includes the user data and all headers (TCP,IP and subnetwork)

Capacity planning at transport

- TCP needs minimal delay, variation and loss probability
- These parameters may conflict
- Retransmission (ARQ) and/or forward error correction (FEC) to trade off delay, delay variance, and packet loss.
- While ARQ increases delay variance, FEC does not. However, FEC often increases mean delay, even on good channels where ARQ are not needed and would not increase either the delay or the delay variance.

Capacity planning at transport

- Bandwidth asymmetry may reduce TCP performance
- Data segment are clocked by returning acks
- Slow return of the acks directly impact performance
- Reduce number of acks in return path
- Change packet forwarding strategy

Capacity planning at transport

- Reordering may affect TCP performance, needlessly decreasing performance
- Could be tolerated since in some cases reduce latency and improve link efficiency and reliability
- Avoid reordering as far as possible