2E1624
Performance analysis of Communication networks

Topic 2
Congestion and error control
Congestion and error control

• Lecture material:
  – I. Kay, *Stochastic modeling*, 3.3.4

• Reading for next lecture:
Control functions in communication networks

- Protocols or control functions?
- Control functions are selected to achieve given objectives (e.g., lossless transmission)
- Protocols are realizations of a set of control functions, where
- Control functions are coupled in some sensible way.
Control functions in communication networks

fairness concept

congestion control
rate control
admission control
error control
delay control

Differentiated transm.
(congestion control)
(error control)
(admission control)
Congestion control and error control

- **Congestion control:**
  - to regulate the packet population in the network
  - to share resources (link bandwidth, buffer space)
  - *Flow control: between two users for speed matching,* sometimes means congestion control in the literature
  - The main objectives of congestion control
    - **Efficiency**
      - high utilization (from the network provider’s perspective)
      - high per flow throughput, low delay (from users’ perspective)
    - **Fairness:** fair allocation of resources

- **Error control:**
  - Retransmission of lost packets with acceptable delay
  - Per link if the loss probability is high
  - End-to-end if the loss probability is low

- **TCP:** combined congestion control and error control
Congestion control

- What happens if the incoming traffic is not restricted?
  - Bottleneck links: the offered traffic is higher than the link transmission capacity: temporarily (bursts arriving) or permanently

- What happens at bottleneck links?
  - queue sizes grow, end-to-end delays increase
  - queue space fills up, packets get dropped
  - packets are retransmitted by the applications, further increasing the load

- Congestion collapse: the network throughput decreases and delays become excessive
Congestion control techniques

• Should depend on the service requirements of the application
  – voice, video: minimum bandwidth requirement, delay sensitive
  – data: requires strict error control
• Call blocking at the network edge + rate control
  – calls blocked if resources are not available
  – the rate of accepted calls is controlled
• Packet discarding at a network node
  – at buffer overflow or earlier
  – discarding policy - fairness, service differentiation
• Packet blocking at the network edge
  – packet waits in a queue outside the network
• Packet scheduling at the network nodes
  – selective transmission of packets - fairness, service differentiation

Group work: find examples for the techniques above.
Congestion control techniques

- Should depend on the service requirements of the application
  - voice, video: minimum bandwidth requirement, delay sensitive
  - data: requires strict error control
- Call blocking at the network edge + rate control
  - calls blocked if resources are not available
  - the rate of accepted calls is controlled
- Packet discarding at a network node
  - buffer overflow or active queue management
  - discarding policy - fairness, service differentiation
- Packet blocking at the network edge
  - packet waits in a queue outside the network
- Packet scheduling at the network nodes
  - selective transmission of packets - fairness, service differentiation
End-to-end window congestion control

- Upper bound on the number of packets transmitted by A and not acknowledged by B.
- Input rate reduced if acknowledgements arrive slowly - congestion control
- Simple case: constant window size, infinite buffers
  - \( W \): window size – fix!
  - \( X \): packet transmission time (packet size/link rate [sec])
  - \( rtt \): round-trip delay (propagation, transmission and queuing delays)
  - \( r \): maximum packet transmission rate [packet/sec]

\[
r = \min \left\{ \frac{1}{X}, \frac{W}{rtt} \right\}
\]

- Rate invers proportional to round-trip delay (congestion: large queuing delays – large \( rtt \))
- Reacts to congestion in \( W \) packets transmission time – large window means slow reaction to congestion
Window control - packet delay

- Packet delay $T$?
  - Simple queuing analysis in “steady state”
  - Calculate delay at given network load
  - What is the queuing system?
    - the whole network
  - Assumption:
    - arrival intensity – constant (steady state)
    - Infinite buffers – no loss
  - Use Little-theorem

\[ N = \sum_{i=1}^{n} \text{const}_i W_i \]

\[ T = \frac{N}{\lambda} = \frac{\sum_{i=1}^{n} \text{const}_i W_i}{\lambda} \]

- For constant aggregate throughput
- Delay is proportional to the number of active sessions (or the sum of window sizes)
Window control - open questions

- How to select the window size?
  - Small window: OK packet delay, but low throughput
  - Large window: OK throughput, but high packet delay
- Dynamic window sizes are necessary to follow the network load
  - small window if the network is congested
  - large window if the network is low loaded
  - congestion is controlled by the window size – not by the RTT
Congestion control

- How to select the window size?
  - Small window: OK packet delay, but low throughput
  - Large window: OK throughput, but high packet delay

- Dynamic window sizes are necessary to follow the network load
  - Congestion is controlled by the window size – not by the RTT

- TCP
  - Performance with dynamic window sizes - comes now
  - More realistic models for TCP performance - next lecture
TCP congestion control

- Window based congestion control
- Dynamic window size:
  - decrease window size if congestion is detected (e.g., by packet loss)
  - increase window size if current rate does not cause losses (no congestion)
- How to increase and decrease the window sizes?
  - Additive-increase, multiplicative decrease (AIMD)
  - Efficient and fair: Chiu and Jain, 1989

- Probing - increase phase in each rtt:
  \[ w_{i+1} = w_i + b, 0 < b << w_{\text{max}} \]

- Congestion - decrease phase:
  \[ w_{i+1} = aw_i, 0 < a < 1 \]

- TCP: \( a=0.5, b=1 \)

- TCP additional phases: slow start, fast recovery – not considered now
Analysis – AIDM model

• How does the throughput depend on the loss rate?

• Assumptions – for a very simplified case
  – constant background traffic, loss at the same window size
  – loss due to congestion only
  – transients negligible (long flows)
  ⇒ we model a static congestion avoidance phase
  – constant round trip time (rtt)
  – rtt >> transmission time
  – constant packet size L
  – low loss probability
  – (to simplify calculations)
TCP throughput vs. packet loss

- Statement: throughput in steady state varies as the inverse-square-root of the loss rate

\[ T_0 = \frac{W_m}{2} \text{rtt}, \]

\[ N = \int_0^{T_0} \frac{W(t)}{\text{rtt}} \, dt = \frac{1}{\text{rtt}} \frac{W_m/2 + W_m}{2} T_0 = \frac{3}{8} W_m^2 \]

\[ p = \frac{1}{N}, \quad N = \frac{1}{p} \]

\[ W_m = \sqrt{\frac{8}{3}} \sqrt{N} = 2 \sqrt{\frac{2}{3}} \frac{1}{\sqrt{p}} \]

\[ Th = \frac{N-1}{T_0} L = \frac{\left( \frac{1}{p} - 1 \right) L}{2 \sqrt{\frac{2}{3}} \frac{1}{\text{rtt}} \frac{1}{\sqrt{p}}} = \frac{3}{2} \frac{L}{\text{rtt}} \left( \frac{1-p}{p} \right) \sqrt{p} \approx \frac{3}{2} \frac{L}{\text{rtt}} \frac{1}{\sqrt{p}} \]

\( T_0: \) cycle (s)
\( W_m: \) max window (packets)
\( \text{rtt}: \) round trip time (s)
\( N: \) packets transmitted in one cycle
\( p: \) packet loss probability
\( L: \) packet length (bit)
\( Th: \) throughput (bit/s)
TCP throughput vs. packet loss

- Statement: throughput in steady state varies as the inverse-square-root of the loss rate

\[ Th \approx \sqrt{\frac{3}{2}} \frac{L}{rtt} \frac{1}{\sqrt{p}} \]
Reading for next lecture

TCP throughput – random loss due to congestion

  - read sections 1, 2-2.1, 3, 4, 5

- Home assignment:
  - Explain how TCP Reno works

- Presentation
  - Explain briefly how TCP Reno works (including slow start, fast retransmit, etc.)
  - list and explain the assumptions used in model 2.1 – why are these assumptions necessary and how are they valid? How does the model differ from the one presented on the lecture?
  - explain the analysis in 2.1 and related performance figures from section 4
Project topics related to TCP

- Research areas where some of the assumptions are released or discussed:
  - TCP performance in high bandwidth-delay product networks
  - Loss does not mean congestion: TCP over wireless links
  - Short sessions: TCP slow start can not be neglected (mice and elephant problem)
  - TCP – misbehavior: synchronization of sessions – are they synchronized? Does it lead to low utilization?
  - Loss distribution among sessions: active queue management for TCP
Error control

- Packet loss due to congestion or link failure
- Lost packets have to be retransmitted
  - ACK: positive acknowledgement
  - NACK: negative acknowledgement

- Evaluation of error control schemes based on positive acknowledgement
  - Go-back-N
  - Selective Repeat

- Evaluate the effective throughput as a function of the packet loss probability
  - Go-back-1 (GB1)
  - Go-back-N (GBN)
  - Selective Repeat (SS)
Error control

• Assumptions:
  – Packets numbered, sequence number used for loss detection
  – Fixed packet size
  – Constant rtt (rtt=1)
  – Packet size<<rtt
  – Independent packet (or ACK) loss, p (Bernoulli loss model)
  – Retransmission timeout $T_0$
Error control

- **Go-Back-1**
  - Each packet acknowledged
  - New packet transmission when ACK received
  - Retransmission after $rtt + T_0$
  - Cycle: sequence of transmissions until loss: $K$
  - Cycle time: $U$
  - Packets received in one cycle: $R$
  - Throughput $= \frac{E[R]}{E[U]}$

$$
\begin{align*}
P(K = k) &= (1 - p)^{k-1} p, \quad E[K] = \frac{1}{p} \\
R &= K - 1, \quad E[R] = E[K] - 1 = \frac{1}{p} - 1 = \frac{1 - p}{p} \\
U &= K \cdot rtt + T_0 = K + T_0, \quad E[U] = E[K] + T_0 = \frac{1 + pT_0}{p} \\
Th &= \frac{E[R]}{E[U]} = \frac{1 - p}{1 + pT_0}
\end{align*}
$$
Error control

- Go-Back-N
  - N packet transmitted after ACK (N: window size)
  - All N packets retransmitted in case of loss
  - Probability of retransmission = \((1-(1-p)^N)\)

\[
p_N = 1 - (1 - p)^N
\]

\[
P(K = k) = (1 - p_N)^{k-1} p_N, \quad E[K] = \frac{1}{p_N} = \frac{1}{1 - (1 - p)^N}
\]

\[
R = N(K - 1), \quad E[R] = NE[K - 1] = N \frac{(1-p)^N}{1 - (1-p)^N}
\]

\[
E[U] = E[K] + T_0 = \frac{1}{1 - (1 - p)^N} + T_0
\]

\[
Th = \frac{E[R]}{E[U]} = \frac{N(1-p)^N}{1 + (1 - (1 - p)^N)T_0}
\]
Error control

- Selective Repeat
  - N packet transmitted after ACK (N: window size)
  - Only the lost packet retransmitted in case of loss
  - Probability of retransmission = $p_N = 1 - (1 - p)^N$
  - Assumption: probability of multiple loss is very small

\[
p_N = 1 - (1 - p)^N
\]

\[
P(K = k) = (1 - p_N)^{k-1} p_N, \quad E[K] = \frac{1}{p_N} = \frac{1}{1 - (1 - p)^N}
\]

\[
R = NK - 1, \quad E[R] = NE[K] - 1 = N \frac{1}{1 - (1 - p)^N} - 1
\]

\[
E[U] = E[K] + T_0 = \frac{1}{1 - (1 - p)^N} + T_0
\]

\[
T_h = \frac{E[R]}{E[U]} = \frac{N - 1 + (1 - p)^N}{1 + (1 - (1 - p)^N)T_0}
\]
For lecture 4: Consider the expressions we have derived for the throughput of Go-Back-N (GBN) and Selective Repeat (SR). Evaluate the performance of these error control schemes based on the equations.

a) Draw the graphs showing the throughput as a function of the loss probability for GBN and SR for $p=0...0.1$ and $N=1,10,40$. Explain the results briefly.

b) Consider GBN. Draw the graphs showing the throughput as a function of window size ($N$) for different loss probability values and fixed $T_0$. Why is there a maximum throughput?

c) Consider a more realistic version of GBN, where after a packet loss only those packets scheduled for transmission in the same round but after the lost one must be retransmitted. Find the throughput of this system and show how it improves compared to the one of the basic GBN.