Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich
Spring Term 2014

Operating Systems and Networks
Solutions 12

Assigned on: **22th May 2014**
Due by: **29th May 2014**

1 Congestion Control

a) *Consider the arrangement of hosts `H1` to `H8` and routers `R1` to `R7` in figure below. Show that the links of `R1` cannot become a bottleneck of the network and that for all other links, there is a traffic pattern that congests that link. Assume that all the traffic is generated exclusively by messages between two hosts; that is, routers only forward messages and are never the source nor the destination of a message.*

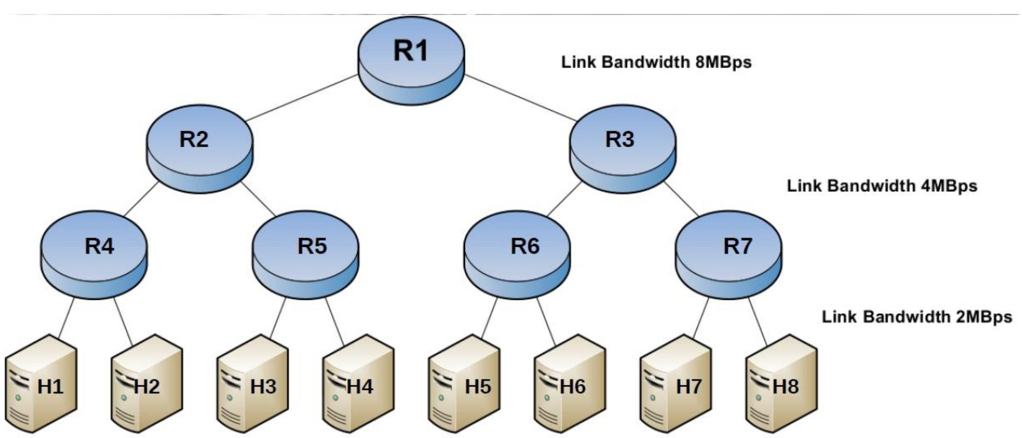

Figure 1: **Network topology for exercise**

Note: *We assume that the given bandwidth numbers correspond both to upstream and downstream link capacities, i.e. `R1` can send and receive simultaneously data to/from `R2` at a rate of 8MB/sec.*

Links of `R1` cannot become a bottleneck:

We examin the left link of `R1`:

The worst case scenario is the one where all the hosts from the left side of `R1` attempt to talk to the respective hosts in the right side of it and vice versa.

- From traffic generated by hosts `H1` to `H4` towards a host in `H5-H8`, the throughput equals to: $\sum_{i=1}^{4} x_i = 2\text{MB/sec} * 4 = 8\text{MB/sec}$.
- From traffic generated by hosts `H5` to `H8` towards a host in `H1-H4`, again, the throughput equals to 8MB / sec.

Consequently, for both directions, the link's utilization reaches it's peak (100%) but it's capacity is exaclty as the demanding throughput, so it does not become a bottleneck.

The network is symmetric, thus, same things apply for the right link of R1.

All other links can become a bottleneck:

(a) We examine the right link l of `R3`:

Consider a scenario in which hosts `H1-H6` generate traffic towards `H8`. The traffic to pass over l equals to: $\sum_{i=1}^{6} x_i$ = 2MB/sec * 6 = 12MB/sec. In this case, the demanding throuput exceedes the maximum throughput of l, which is 4MB/sec, thus the latter becomes a bottleneck.

Because of symmetry, relevant use cases can be constructed for the left link of `R3` and the two links of `R2`

(b) We examine the left link l of `R7` (connecting the H8):

Consider a scenario where hosts `H5-H7` generate traffic towards `H8`. The generated throughput equals to $\sum_{i=5}^{7} x_i$ = 2MB/sec * 3 = 6MB/sec. Again, this exceeds the maximum capacity of the link l (2MB/sec) which becomes a bottleneck. Applying the same arguments for the links between the routers `R4-R6` and the rest hosts (`H1-H7`) we easily prove the requirement.

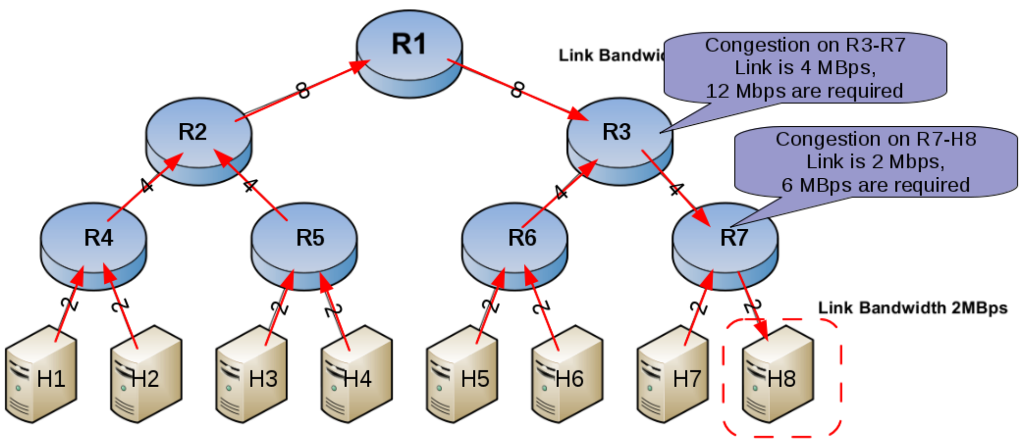

Figure 2: **Two cases where congestion can occur in lower level routers**

b) *Assume that TCP implements an extension that allows window sizes much larger than 64 KB. Suppose that you are using this extended TCP over a 1-Gbps link with a latency of 100 ms to transfer a 10-MB file, and the TCP receive window is 1 MB. If TCP sends 1-KB packets (assuming no congestion and no lost packets):*

(a) *How many RTTs occur until the first transmission with a window size of 1MB is done?*

Using slow start, the sending rate is increased exponentially rather than linearly. The window begins with a size of 1 KB and increases exponentially, doubling every time. $1 = 2^0, 2 = 2^1, ...1024 = 2^{10}$ Therefore, it takes 10 RTTs for the window size to ramp up to 1 MB (the 11th send uses a 1 MB window).

(b) *How many RTTs does it take to send the file?*

In the first RTT, 1 KB is sent, in the second, 2 KB is sent for a total of 3 KB. On round n, the amount of data sent during this round is given by the equation:

$$data_n = \begin{cases} 2^{n-1} \quad \text{KB} & \text{if } n < 11 \\ 1024 \quad \text{KB} & \text{if } n \geq 11 \end{cases}$$

In n iterations, the total data sent is:

$$total\_data_n = \begin{cases} \Sigma_{i=0}^{n}(2^{i-1}) \quad \text{KB} & \text{if } n < 11 \\ 1024 * (n-10) + \Sigma_{i=0}^{10}(2^{i-1}) \quad \text{KB} & \text{if } n \geq 11 \end{cases}$$

Which can be rewritten as:

$$total\_data_n = \begin{cases} \Sigma_{i=0}^{n}(2^{i-1}) \quad \text{KB} & \text{if } n < 11 \\ 1024 * (n-10) + 1024 \quad \text{KB} & \text{if } n \geq 11 \end{cases}$$

After calculus analysis, we find that in 19 RTTs, a total of 10,239 KBytes (this is equal to 10 MBs - 1 KB, assuming that 1024 KB = 1 MB) have been sent and in 20 RTTs, a total of 11,263 KBytes have been sent. So after 20 RTTs the file would be sent completely.

(c) *If the time to send the file is given by the number of required RTTs multiplied by the link latency, what is the effective throughput for the transfer? What percentage of the link bandwidth is utilized?*

Time to send the file is:

$$time = 20RTTs * 2 * 100ms = 4.0sec$$

The effective throughput is given by:

$$\frac{10MB}{4.0sec} = 2.5MBs = 20Mbs$$

Percentage of bandwidth utilized is:

$$\frac{20Mbs}{1Gbs} = 2\%$$

2 TCP congestion control

A TCP connection uses the slow start algorithm with a threshold of 8KB for congestion control. The maximum segment size should be 1KB and the receive's window is 16KB. After the 8th, the 11th, and the 17th transmission, timeouts are occurring, which are interpreted as network overload. Sketch the size of congestion window and the threshold into the following diagram.

Figure below plots the size of congestion window and the threshold.

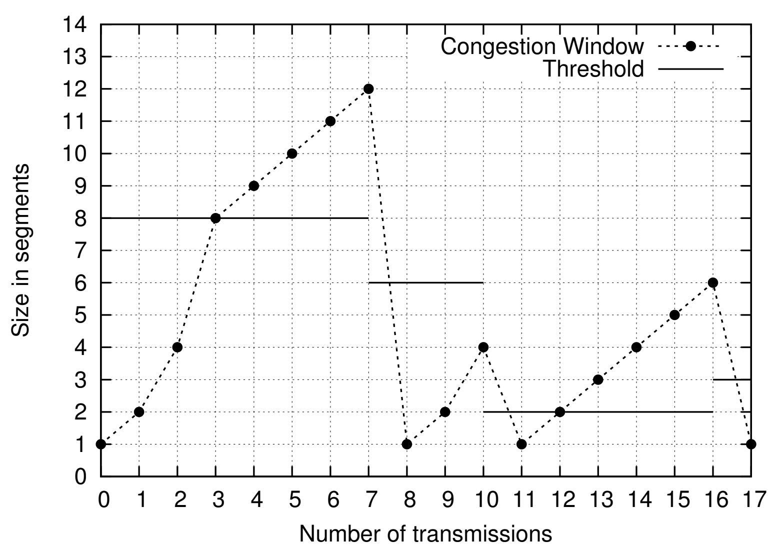

Figure 3: **TCP slow start and congestion avoidance**