Computer Networks - Principles of Congestion Control

Joey Elsisi

October 18, 2021

1 Principles of congestion control

Defining congestion: Too many sources sending too many packets too fast for a congested link somewhere in the network to handle. This is different from flow control, which is when a sender's transmission rate is too fast for its receiver.

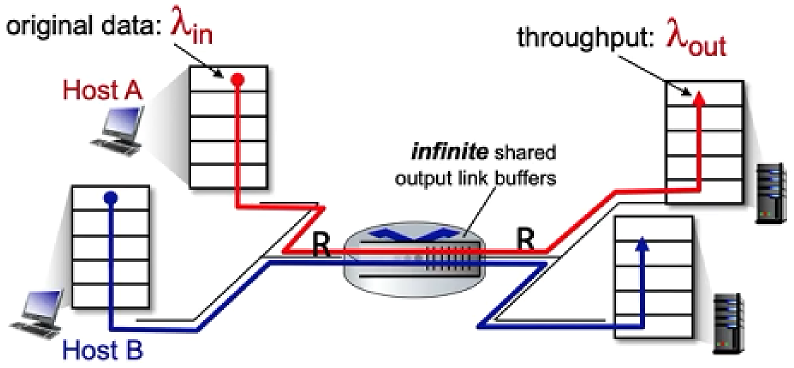
Results of congestion: long delays (queueing in router buffers), packet loss (buffer overflow at routers)

2 Causes/costs of congestion

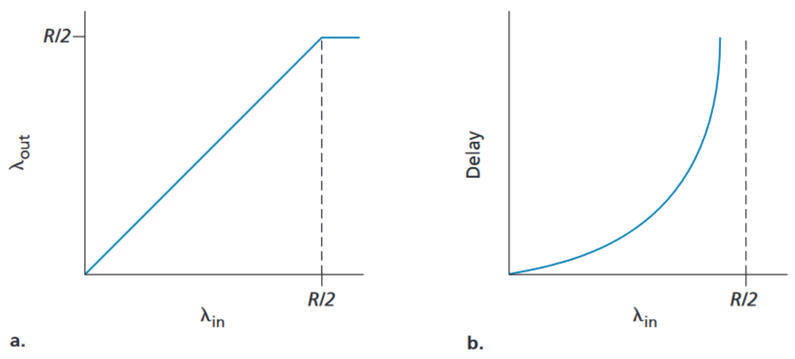
We look at 3 Scenarios

2.1 Scenario 1: Two Senders, a Router with Infinite Buffers

In this senario, Host A and Host B send data at an average rate of λ_{in} bytes/sec. Packets from Hosts A and B pass through a router and over a shared outgoing link of capacity R. Assume that the router has an infinite buffer, meaning no packets are dropped.


original data: λ_{in}
Host A
Host B
throughput: λ_{out}
infinite shared output link buffers
R
R

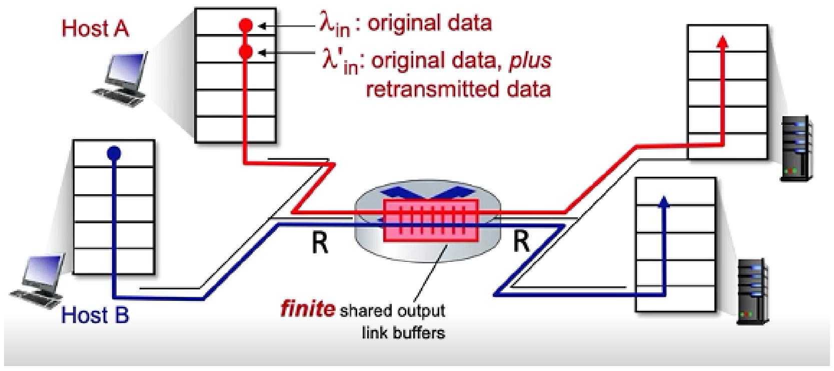
As the rate λ_{in} increases from 0 to R/2 per host, the sending rate equals the receiving rate with only a finite amount of delay, as seen in the left graph. Once the sending rate exceeds R/2, the receiving rate remains at R/2. This upper limit on throughput is a consequence of the sharing of link capacity between two connections. We see on the right graph that as the sending rate approaches R/2 per host, delay increases exponentially because of buffered packets at the router.



Takeaway: Large queuing delays are experienced as the packet-arrival rate nears the Router's link capacity.

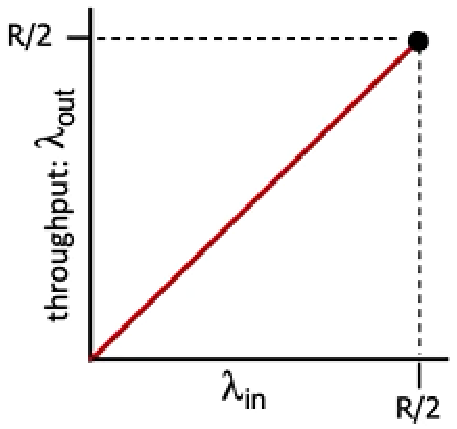
2.2 Scenario 2: Two Senders and a Router with Finite Buffers

We modify the first scenario in 2 ways. First, the router buffer is finite, meaning packets will be dropped when arriving to an already-full buffer. Second, we assume that the transport layer will retransmit dropped packets to guarantee delivery. Because packets can be retransmitted, we must now be more careful with our use of the term sending rate. The rate at which the transport layer sends segments (containing original data and retransmitted data) into the network will be denoted $\lambda_{in}^{'}$.

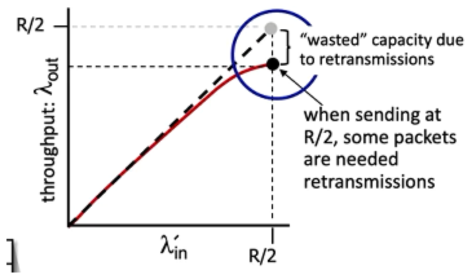

Host A
λin : original data
λ'in: original data, plus retransmitted data
R
R
Host B
finite shared output link buffers

The performance realized under scenario 2 will now depend strongly on how retransmission is performed. Consider these 3 scenarios:

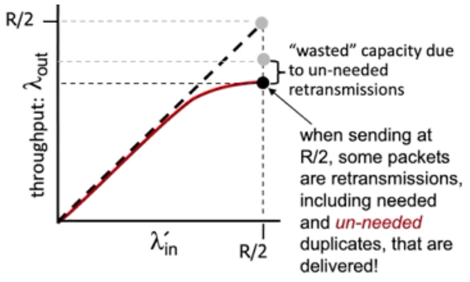
perfect knowledge: Sender sends only when router has free buffer space. In this case, no loss would occur, λ_{in} would be equal to λ_{in}, and the throughput of the connection would be equal to λ_{in}. From a throughput standpoint, performance is ideal—everything that is sent is received.



some perfect knowledge: The slightly more realistic case where the sender retransmits only when a packet is known for certain to be lost. We see in the graph below that although the receiver receives R/2 data, a fraction of the data is retransmitted, which we consider wasted capacity.



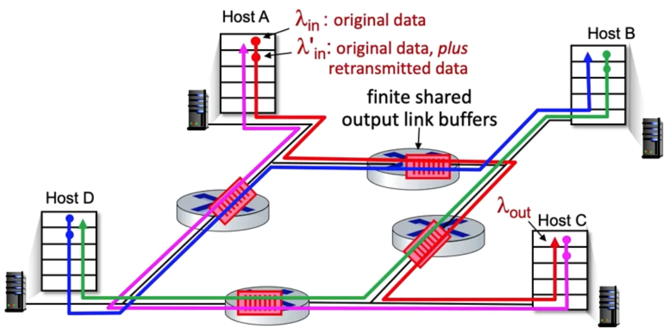
duplicate data: We also need to consider the case when the sender may time out prematurely and retransmit a packet that has been delayed in the buffer but not yet lost. Both the original data packet and the retransmission may reach the receiver, wasting the routers forwarding capacity. We update our graph.



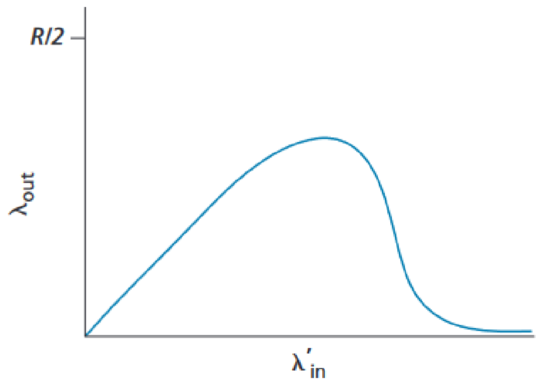
Takeaway: Congestion leads to both needed and unneeded retransmissions, further weakening throughput.

2.3 Scenario 3: Four Senders, Routers with Finite Buffers, and Multihop Paths

In our final scenario, four hosts transmit packets, each over two-hop paths.


Host A
λ_{in} : original data
λ'_{in}: original data, *plus* retransmitted data
Host B
finite shared output link buffers
Host D
λ_{out}
Host C

The A–C connection shares router R2 with the B–D connection. Because A-C must first go through R1 to reach R2, it's maximum arrival rate at R2 is R (the outgoing rate of the R1). Since B-D's first link is R2, it can send to R2 at a much higher rate, completely filling R2's buffer for high values of λ_{in}. This produced the performance graph below for finite buffers and multihop paths. We see that throughput approaches 0 for large values of λ_{in}.



Takeaway: when a packet is dropped along a path, the work done at each of the upstream links to forward that packet to the point at which it is dropped ends up having been wasted.

3 Handling congestion

See Ethan Hanover's notes. For graphs, see Ch. 3.7.1 in the textbook.