Recitation 23

Last Time:

* Throughput of Stop-N-Wait

Case 1: No packet loss $\boxed{\lambda = \frac{1}{RTT}}$ pkts/s.

case 2: With packet loss; rate - ℓ

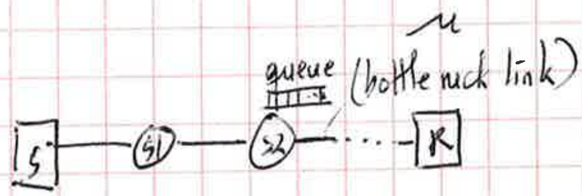
$$T = (1-\ell)RTT + \ell(RTO + T)$$

(T: expected time to send PKT and receive ACK.)

and $\boxed{\lambda = \frac{1}{T}}$ where $T = RTT + \frac{\ell}{1-\ell} RTO$

Today:

* Throughput of Sliding Window.



Case 1: No packet loss:

$$\boxed{\lambda = \min\left(\mu, \frac{W}{RTT_{min}}\right) = \begin{cases} \mu & W \geq \mu * RTT_{min} \\ \frac{W}{RTT_{min}} & W < \mu * RTT_{min} \end{cases}}$$

RTT_{min} ~ RTT in absence of queueing

W ~ Window size

$\mu * RTT_{min}$ ~ Bandwidth-delay product

μ ~ bottleneck link rate.

Case 2: With packet loss:

Suppose: Set Window Size $W > \mu * RTT_{min}$ so that bottleneck link busy!

(*) Suppose prob. of bidirectional data packet or ACK loss $= \ell$

Total *expected* # transmissions T for successful delivery of a packet & its ACK:

with prob. $(1-\ell)$, need 1 transmission

" " $\ell(1-\ell)$ " 2 transmissions

" " $\ell^2(1-\ell)$ " 3 transmissions

$\vdots$

so that

$$T = (1-\ell).1 + \ell(1-\ell).2 + \ell^2(1-\ell).3 + \ell^3(1-\ell).4 + \cdots$$

$$= (1-\ell) + (2\ell - 2\ell^2) + (3\ell^2 - 3\ell^3) + (4\ell^3 - 4\ell^4) + \cdots$$

$$= 1 + \ell + \ell^2 + \ell^3 + \cdots$$

$$= \frac{1}{1-\ell}$$

and thus

$$\lambda = \frac{1}{T} = 1 - \ell$$

(Throughput/utilization)

$$\boxed{\lambda = 1 - \ell}$$

(*) Recall: (last time): Stop-N-Wait protocol btn BOS and SFO ~ 10% utilization

For Sliding-Window, one can obtain utilization ~100% for small ℓ.

Ex 1 **Given**: Ntwk using reliable data transport protocol.

$RTT_{min} = 100ms$.

Bottleneck link bandwidth $\mu = 1$ Mbyte/s

Packet size = 1000 bytes

Assume: No packet loss.

Reqrd: **Q1**: What's the highest throughput of the stop-N-wait protocol?

Q2: To improve performance, we implement sliding window. What should W be in order to saturate the bottleneck link capacity?

Q3: What is the throughput of the sliding window protocol in Q2?

Soln: (1) SNW: (No pkt loss): $\lambda = \frac{1\ PKT}{RTT} = \frac{1\ PKT}{100ms} = 10\ PKT/s$

and since 1 PKT = 1000 bytes,

$\lambda_{SNW} = 10{,}000$ Bytes/s $= 10$ kBytes/s

(2) Make $W = \mu * RTT_{min}$ (bandwidth-delay product)

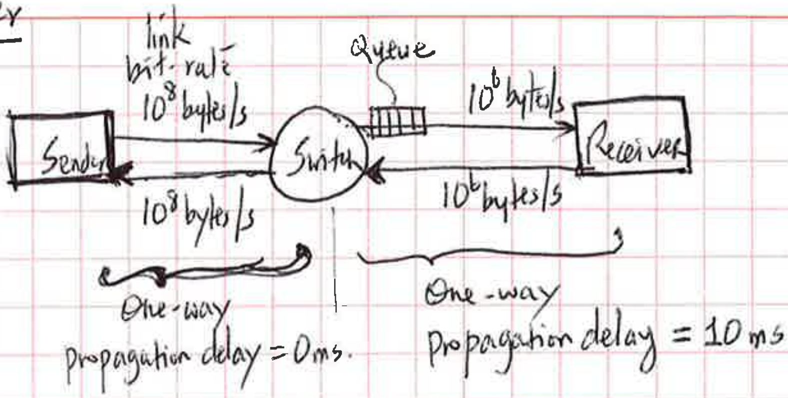
ie $W = (1 \text{Mbyte/s}) * (100\ ms) = 100$ KBytes $= 100$ Packets

(since 1 PKT = 1000 bytes).

(3) When $W = \mu * RTT_{min}$, $\lambda_{SW} = \mu = 1$ MBytes/s

Comparing throughput of SNW to that of SW protocol, 10 KBytes/s vs 1 MByte/s

Ex2: Consider



Given: Packet_size = 1000 bytes (W)

ACK_size = 40 bytes

Sender Window Size = 10 packets

No other traffic; no packet loss; no processing delay.

Q: At what approximate rate (in PKTS/s) will the protocol deliver a multi-gigabyte file from the cender to the receiver?

Soln: Since no pkt loss,

$$\boxed{\lambda = \min\left(\mu, \frac{W}{RTT_{min}}\right)}$$

$\mu = 10^6$ bytes/s $= 10^3$ PKTS/s

$W = 1000$ bytes.

Need RTT_{min}.

RTT = Propagation delays (PKT and ACK) + Transmission delays (PKT and ACK) + ~~Processing delays~~ $\to 0$, where Transmission delay = $\frac{\text{PKT\_Size}}{\text{link rate}}$ (bottleneck link).

(negligible (small)) negligible (small)

$= 20\text{ms} + \frac{1000 \text{ bytes}}{10^6 \text{ bytes/s}} = \underline{21 \text{ ms}}$

So that $\frac{W}{RTT_{min}} = \frac{10 \text{ PKT}}{21 \text{ ms}} = \underline{476 \text{ PKTS/s}}$.

Thus, $\lambda = \min\left(\mu, \frac{W}{RTT_{min}}\right) = \min\left(10^3, 476\right)$ PKTS/s $= \underline{476 \text{ PKTS/s}}$

Ex3 Consider a reliable transport connection using sliding window protocol.

Given: $RTT_{min} = 0.1s$ (RTT_{min} is RTT in the absence of queueing delay).

PKT_size = 1000 bytes

No other traffic, no pkt loss.

Questions:

Q1: If bottleneck link rate $\mu = 100$ packets/s and window size $W = 8$ packets, what is the throughput?

Q2: If bottleneck link rate remains the same ($\mu = 100$ packets/s), but window size W increases to **16** Packets, what is the throughput?

Q3: What is the smallest window size for which the connection's RTT exceeds RTT_{min}?

Q4: Suppose we set $W = \mu * RTT_{min}$. If we use an 8-bit field for the sequence number in each packet, what is the smallest value of the bottleneck link bandwidth (μ) that will cause the protocol to stop working correctly?

Solns:

1. Since no PKT loss, $\lambda = \min\left(\mu, \frac{W}{RTT_{min}}\right)$ where $\mu = 100$ PKTS/s

$$\frac{W}{RTT_{min}} = \frac{8 \text{ PKTS}}{0.1s} = 80 \text{ PKTS/s}$$

Then $\lambda = \min(100, 80)$ PKTS/s $= \underline{80 \text{ PKTS/s}}$

Alternatively, since $\mu * RTT_{min} = 100 \frac{PKTS}{s} * 0.1s = 10 \text{ PKTS} \geq W$ (8 PKTS)

$$\lambda = \frac{W}{RTT_{min}} = \frac{8}{0.1} = 80 \text{ PKTS/s}$$

(2) Again, since $\mu * RTT_{min} = 10 < \overset{(W)}{16}$,

$$\lambda = \mu = \underline{100 \text{ PKTs/s}}.$$

OR $\lambda = \min\left(100, \frac{16}{0.1}\right) = \underline{100}$ as above.

(3) When $W > \mu * RTT_{min}$, there is queueing delay in front of the bottleneck link, and RTT exceeds RTT_{min}.

$$\mu * RTT_{min} = (100 \text{ PKT/s})(0.1 \text{ s}) = 10 \text{ PKTS}$$

So the smallest W for which RTT exceeds RTT_{min} is 11 PKTS

(4) When $W > 2^8 = 256$, the sequence number field wraps around, and the protocol stops working correctly.

i.e. the sequence numbers are no longer unique!

So for $W = 256$ Packets, protocol stops working correctly.

$$\mu = \frac{W}{RTT_{min}} = \frac{256 \text{ PKTS}}{0.1 \text{ s}} = 2560 \text{ PKTs/s} = 2560 * 1000 \text{ Bytes/s}$$

$$= 2.56 \text{ MBytes/s}.$$

Thus, the smallest value of the bottleneck link bandwidth for which the protocol stops working correctly is 2.56 MBytes/s