

Assignment 1

Andrew Fillmore

September 3, 2015

Problem 6

a

$$d_{prop} = m/s$$

b

$$d_{trans} = L/R$$

c

The end-to-end delay will be d_{total} . Then $d_{total} = d_{prop} + d_{trans}$

d

At time $t = d_{trans}$ the last bit of the packet will be being transmitted from Host A.

e

If $d_{prop} > d_{trans}$ then at time $t = d_{trans}$ the first bit will be at $d_{trans} \times s$ meters.

f

If $d_{prop} < d_{trans}$ then at time $t = d_{trans}$ the first bit will have arrived at Host B already.

g

If

$$d_{trans} = L/R = 120/56000 = 3/1400$$

then

$$d_{prop} = 3/1400.$$

Because

$$d_{prop} = m/s \Rightarrow m = d_{prop} \times s$$

then

$$m = (3/1400) \times 2.5 \times 10^8 = 535,714.28571428571428571429 \text{meters}$$

10

The total end-to-end delay for the packets will be

$$\frac{d_1}{s_1} + \frac{L}{R_1} + d_{proc} + \frac{d_2}{s_2} + \frac{L}{R_2} + d_{proc} + \frac{d_3}{s_3} + \frac{L}{R_3}.$$

With values the delay will be

$$\begin{aligned} & \frac{5000km}{2.5 \times 10^8 m/s} + \frac{1500b}{2Mbps} + 3ms + \frac{4000km}{2.5 \times 10^8 m/s} + \frac{1500b}{2Mbps} + 3ms + \frac{1000km}{2.5 \times 10^8 m/s} + \frac{1500b}{2Mbps} \\ &= 0.02s + 0.75ms + 3ms + 0.016s + 0.75ms + 3ms + 0.004s + 0.75ms \\ &= 48.3ms \end{aligned}$$

11

Assumption: This problem is asking for the general form, not using the specific values from the previous problem.

The end-to-end delay for this new system will be

$$\frac{L}{R} + \frac{d_1}{s_1} + \frac{d_2}{s_2} + \frac{d_3}{s_3}.$$

18

Traceroute from my host 192.168.15.105 to google.com 173.194.33.174

At 9:00pm MST on September 2 2015

```
C:\Users\Andrew Fillmore>tracert google.com
```

```
Tracing route to google.com [173.194.33.136]
```

```
over a maximum of 30 hops:
```

| | | | | |
|----|-------|-------|-------|--|
| 1 | 1 ms | 1 ms | 1 ms | 192.168.15.254 |
| 2 | * | * | * | Request timed out. |
| 3 | 1 ms | 1 ms | 1 ms | 198.60.208.4 |
| 4 | 4 ms | 4 ms | 4 ms | r11-g0-1-1600005.boi.fiberpipe.net [209.161.25.89] |
| 5 | 4 ms | 4 ms | 4 ms | r20-t4-1-2.core1.boi.fiberpipe.net [209.151.62.21] |
| 6 | 4 ms | 6 ms | 5 ms | 70.102.78.121 |
| 7 | 19 ms | 20 ms | 19 ms | six.sea01.google.com [206.81.80.17] |
| 8 | 20 ms | 20 ms | 19 ms | 66.249.94.212 |
| 9 | 20 ms | 20 ms | 20 ms | 209.85.244.61 |
| 10 | 19 ms | 19 ms | 19 ms | sea09s17-in-f8.1e100.net [173.194.33.136] |

Trace complete.

At 1:00am MST on September 3 2015

```
C:\Users\Andrew Fillmore>tracert google.com
```

```
Tracing route to google.com [173.194.33.133]  
over a maximum of 30 hops:
```

| | | | | |
|----|-------|-------|-------|--|
| 1 | 1 ms | 1 ms | 1 ms | 192.168.15.254 |
| 2 | * | * | * | Request timed out. |
| 3 | 2 ms | 1 ms | 2 ms | 198.60.208.10 |
| 4 | 4 ms | 4 ms | 4 ms | r11-g0-1-1610003.boi.fiberpipe.net [209.161.25.93] |
| 5 | 23 ms | 4 ms | 4 ms | r20-t4-1-2.core1.boi.fiberpipe.net [209.151.62.21] |
| 6 | 19 ms | 5 ms | 5 ms | 70.102.78.121 |
| 7 | 19 ms | 20 ms | 20 ms | six.sea01.google.com [206.81.80.17] |
| 8 | 19 ms | 20 ms | 20 ms | 66.249.94.212 |
| 9 | 20 ms | 20 ms | 20 ms | 209.85.244.61 |
| 10 | 20 ms | 20 ms | 20 ms | sea09s17-in-f5.1e100.net [173.194.33.133] |

Trace complete.

a

b

c

There appears to be a company called Boise Fiberpipe, which can be inferred from host names ending in `.boi.fiberpipe.net`. Another company is represented by `six.sea01.google.com`. It may be reasonable to assume that this is a ISP that has a direct connection to Google's servers. Although it may be the case that this "sea" company is directly owned by Google as it also seems to be hosting the end address.

d

```
Traceroute from www.telstra.net(Australia) to google.com 173.194.33.174
```

| | | | |
|---|--|-----------|-----------|
| 1 | gigabitethernet3-3.exi2.melbourne.telstra.net (203.50.77.53) | | |
| | 0.375 ms | 0.206 ms | 0.243 ms |
| 2 | bundle-ether3-100.win-core10.melbourne.telstra.net (203.50.80.129) | | |
| | 0.989 ms | 1.354 ms | 2.117 ms |
| 3 | bundle-ether12.ken-core10.sydney.telstra.net (203.50.11.122) | | |
| | 14.611 ms | 14.096 ms | 14.983 ms |
| 4 | bundle-ether1.ken-edge901.sydney.telstra.net (203.50.11.95) | | |
| | 13.235 ms | 13.225 ms | 13.233 ms |
| 5 | goo1154859.lnk.telstra.net (139.130.213.54) | | |
| | 12.984 ms | 12.974 ms | 12.985 ms |
| 6 | 209.85.242.124 (209.85.242.124) | | |
| | 14.109 ms | 24.595 ms | 14.106 ms |
| 7 | 72.14.238.186 (72.14.238.186) | | |
| | 150.155 ms | | |

```

8  64.233.174.205 (64.233.174.205)
   159.146 ms
9  72.14.232.63 (72.14.232.63)
   194.758 ms
10 209.85.254.19 (209.85.254.19)
    177.015 ms  176.881 ms  176.886 ms
11 66.249.94.200 (66.249.94.200)
    178.389 ms  178.878 ms  178.640 ms
12 209.85.244.65 (209.85.244.65)
    177.389 ms  177.379 ms  177.387 ms
13 sea09s18-in-f14.1e100.net (173.194.33.174)
    177.012 ms  176.874 ms  176.883 ms

```

Telstra seems to be the major Australian company in this trace. It also follows the same pattern as before, where the major ISP has a direct connection to Google. Something interesting to note here however is that the connection transfers to Google before making the overseas jump, meaning that Google has connections that cross the Pacific Ocean.

23

a

The inter-arrival time will be $\frac{L}{R_s} + \frac{L}{R_c}$ seconds. This is assuming the router is using store-forwarding. If the router is not using store-forwarding then the inter-arrival time will be $2\frac{L}{R_s}$ seconds.

b

If the router is using store-forwarding then there will always be a delay while waiting for the entire packet to arrive. Even if the router is not using store-forwarding the second packet will be queued for $\frac{L}{R_c} - \frac{L}{R_s}$ seconds. This is the difference between the time each link takes to transmit a packet. Therefore T must be $\frac{L}{R_c} - \frac{L}{R_s}$ to ensure no queuing.

24

40 terabytes is 3.2×10^{14} bits. It will take 3.2×10^6 seconds to transmit which is equivalent to 37 days 5 minutes 20 seconds. Therefore it would be better to send the data by FedEx.

“Never underestimate the bandwidth of a station wagon full of tapes hurtling down the highway.” - Andrew S. Tanenbaum

31

a

It will take 4 seconds to move the message from the source host to the first packet switch. Assuming that there are 2 switches between the source and the the host, it will take 12 seconds to move the message from source host to destination host.

b

It will take 5ms to move the first packet from source host to the first switch. The second packet will be fully received by the first switch at 10ms.

c

It will take 4 seconds plus a pseudo- delay of 10 ms to fully transmit the entire message. That is 4.01 seconds. This is nearly one-third of the non-segmented message. This makes sense as segmenting will allow for pipelining and more throughput.

d

One reason to use message segmentation is that all the segments don't need to be sent down the same paths. They can be spread out so that they don't cause congestion in one part of the internet. Also, segmentation means that even if some of the packets are lost, most of the data will most likely be received by the client.

e

A drawback of message segmentation is that it requires a lot more setup than non-segmented messages. Each packet must contain all the information about where it needs to go. Also the packets will need to be reassembled, which will be much harder if the packets arrive out of order.