Computer Networks HW2

Mitchell Meier

October 4, 2020

1. (1) The minimum time needed to distribute this file from a server to all clients in a client-server model is D_{cs}

$$D_{cs} = max(\frac{NF}{u_s}, \frac{F}{d_{min}})$$

With N equal to the number of clients (7), F is equal to the file size (4 Gigabits), u_s is equal to the upload rate of the server (94 Megabits per second), and d_{min} is equal to the lowest download rate out of all clients (c_1 in this case, 15 Megabits per second). Using these numerical values, the solution to D_{cs} can be simplified to

$$D_{cs} = max(297.87, 266.67)$$

Which means the minimum time needed to distrubute F to all clients will be 297.87 seconds

(2) The server, s, is the cause for the final minimum time. The server is guaranteed to take longer to upload the file F to each individual client than the slowest client is to download that file

(3) The minimum time needed to distribute this file from a server to all clients in a peer to peer model is D_{P2P}

$$D_{P2P} = max(\frac{F}{u_s}, \frac{F}{d_{min}}, \frac{NF}{u_s + \sum\limits_{i=1}^{N} u_i})$$

With N equal to the number of clients (7), F is equal to the file size (4 Gigabits), u_s is equal to the upload rate of the server (94 Megabits per second), d_{min} is equal to the lowest download rate out of all clients (c_1 in this case, 15 Megabits per second), and $\sum_{i=1}^{N} u_i$ is equal to the combined upload bandwith of all clients in the system (135 Megabits per seond). Using these numerical values, the solution to D_{P2P} can be simplified to

$$D_{P2P} = max(42.55, 266.67, 122.27)$$

Which means the minimum time needed to distrubute F to all clients will be 266.67 seconds

(4) The client, c_1, is the cause for the final minimum time. The client is guaranteed (assuming all clients are utilizing their full upload speed) to take longer to download the file F than it will take the other clients to upload/share the file with each other

2. Yes, it is possible for an application to utilize reliable data transfer even while not using TCP. An application developer can use UDP and then implement reliable data transfer into the application layer. A developer may do this to avoid some of TCP's built in features, such as congestion control

3.

4. Sequence numbers help the reciever categorize arriving packets and determine if they contain new or retransmitted data. Sequence numbers also support reordering and provide insight on potential dropped packets

5. Timers help detect lost packets. By detecting if the ACK for a transmitted packet is not recieved within the time specified by the timer, the packet is projected to have been lost, and the packet is transmitted again

6. (a)