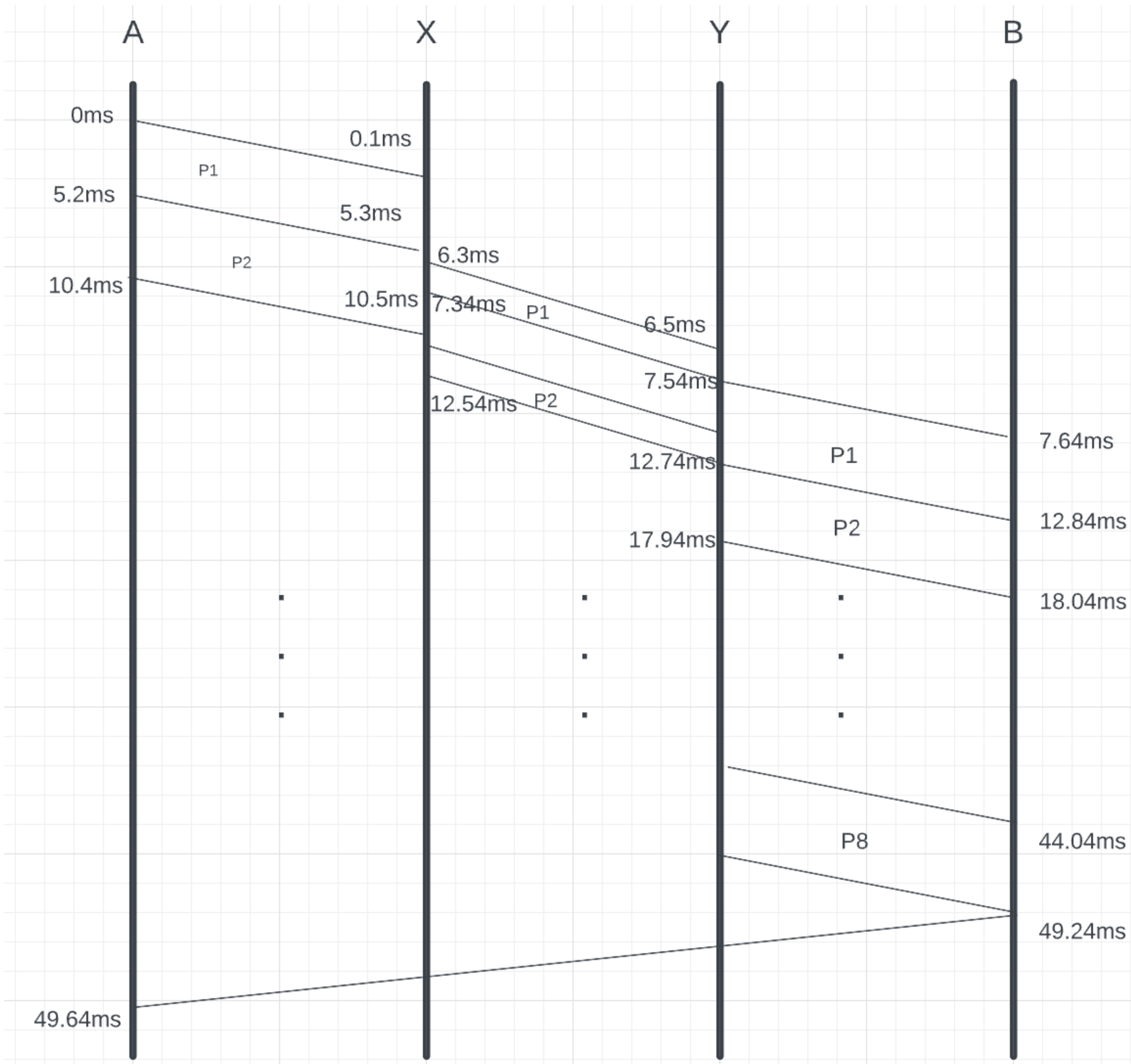
CS408 – Computer Networks – Fall 2022 - HW 1 – Due October 31, 2022, 14:30

Name: Muhammed Orhun Gale **ID**: 26754

Question 1)

a)

Answer is 49.64 ms. My work is right below the figure.



In order to compute the RTT of the model, first one must compute variable propagation and transmission delays which differ because of link capacities or length of the physical link. Also, one should highlight the number of packets too.

Number of packets: (Total data) / (Data per packet)
10000 bytes of data to 1250 byte pieces → 10000 bytes /1250 bytes = **8 packet**

Size of one packet in terms of bits: ((Data size per packet) + (Header size)) * 8
1250 bytes + 50 bytes = 1300 bytes → 1300 bytes to bits → 1300 * 8 = **10400 bits per packet**

For propagation delay: (Length of the cable) / (speed of the signal in the given medium)

Propagation delay between A and X: 30 (km) / 300 000 (km/s) = 0.0001 s = **0.1 ms**
Propagation delay between X and Y: 60 (km) / 300 000 (km/s) = 0.0002 s = **0.2 ms**
Propagation delay between Y and B: 30 (km) / 300 000 (km/s) = 0.0001 s = **0.1 ms**

For transmission delay: (Number of bits to transfer) / (Links transmission capacity)

Transmission delay between A and X:10400 (bits) / 2 000 000 (bps) = 0.0052 s = **5.2 ms**
Transmission delay between X and Y: 10400 (bits) / 10 000 000 (bps) = 0.00104 s = **1.04 ms**
Transmission delay between Y and B: 10400 (bits) / 2 000 000 (bps) = 0.0052 s = **5.2 ms**

Processing delay is given for X to Y, not X to A so it will be considered as **1 ms** for transmissions from X to Y, not from X to A in the ACK case.

Model Explanation:
In the model, A sends the packets bit by bit through the link. While sending it does not wait for anything, it directly starts sending other packets right after sending a packet.

For X and Y to be able to start sending, they should wait for the previous nodes to send a packet fully. On the node X, there exists a processing delay for 1 ms and larger propagation delay while sending packets to Y. However, the link between X and Y is significantly faster than the links between A-X and Y-B and this situation compensates for the slow down on transmissions of packets from X to Y. **What I mean with this statement is that, even though there exists a processing delay, the X transmits the latest packet to Y earlier than A sends a new packet to X and when the new packet has come the link is ready to transmit it. Also, X transmits the latest packet to Y right before Y finish sending it to B because the transmission delays of Y to B and A to X are the same.** Therefore, there won't be any delay caused by waiting. The situation that I explained can also be observed from the figure above.

Round Trip Time: Transportation time of first packet to Y + Transmission delay of 8 packets from Y to B + propagation delay between Y and B + acknowledgement propagation delay (summation of all propagation delays) from B to A.

Transportation time of first packet to Y = 0.1 ms + 5.2 ms + 1 ms + 0.2 ms + 1.04 ms = **7.54 ms**
Transmission delay of 8 packets from Y to B = 8 * (5.2 ms) = **41.6 ms**
Propagation delay between Y and B = **0.1 ms**
Acknowledgement propagation delay from B to A = 0.1 ms + 0.2 ms + 0.1 ms = **0.4 ms**

RTT = 7.54 ms + 41.6 ms + 0.1 ms + 0.4 ms = **49.64 ms**

b) Answer is **0.95693779904**
Utilization can be found via proportioning time of the effective data transmission to time of total data transmission from that link:

Time spend for effective data transmission of one packet: (1250 bytes * 8) / 2 000 000 bps = 5 ms

Total time of effective transmission: 5 ms * 8 (number of packets) = 40 ms

Time spend for transmitting all data (including header): (1300 bytes * 8) / 2 000 0000 bps = 5.2 ms

Total time to transmit all data from A to X: 5.2 ms * 8 (number of packets) = 41.7 ms

Transmission while transmitting ACK signal from X to A: 0.1 ms

Then the utilization = 40 ms / (41.7 ms + 0.1 ms) = 40/41.8 = **0.95693779904**

Question 2)

a) Answer is **0.4459.**

For both armies to carry out the plan, army A has to carry out the plan since in order to send the last pigeon it should at least receive one of the pigeons that was sent by B and if the A receives one of the pigeons it will definitely carry out the plan.

Therefore, the probability of both armies carrying out the plan is the probability of A carrying out the plan **AND** probability of the arrival of the last pigeon that A sent to B. Also probability of successful arrival of a pigeon is 0.7 Let random variables for this situation be

Xa1 := Pigeon A1 that A sent arrives to B (First pigeon)
Xa2 := Pigeon A2 that A sent arrives to B (Last acknowledgment pigeon)
Xb1 := Pigeon B1 sent by B arrives to A (One of two pigeons)
Xb2 := Pigeon B2 sent by B arrives to A (One of two pigeons)

Then the probability of A carrying out the plan is,

Xa1 and ((Xb1' and Xb2) or (Xb1 and Xb2') or (Xb1 and Xb2)) = 0.7 * ((0.3 * 0.7) + (0.7 * 0.3) + (0.7 * 0.7)) = **0.637**
Then the probability of both armies carrying out the plan is,

0.637 * 0.7 = **0.4459**

b) Answer is **0.1911**

For actualization of the event, which is the only one army carrying out the plan, the only possibility is that **A operates the plan but B does not**. This is because for army B to operate the plan, A has to operate it. Therefore, the B should not receive the last ACK pigeon that A sent.

In this sense, the probability of this event is A carrying out the plan **AND** B does not receive the last ACK pigeon.

On the part a of question 2, I found that the probability of A carrying out the plan is **0.637**. Lets describe the probability of A carrying out the plan with random variable Xaok and receiving the ACK pigeon that A sent to B as Xa2. Then the probability of only one army carrying out the plan is,
Xaok and Xa2' = 0.637 * 0.3 = **0.1911**

Question 3) Answer is **"1 packet" and "throughput value 0.9615"**

Throughput of a network model is calculated via finding effective capacity of the link. Effective capacity is transmitted effective data bits (not control, header bits etc.) per unit of time. In this model, there exists a packet header which decreases the effective capacity of the link; however, the number of packets that would be sent is a variable. Let's say the number of packets is "**p**". In order to find this,

The total time to send the all data:

(5000 bits + 200*p bits) / (1 000 000 bps) = ((5000 + 200*p) * 10^-6) seconds

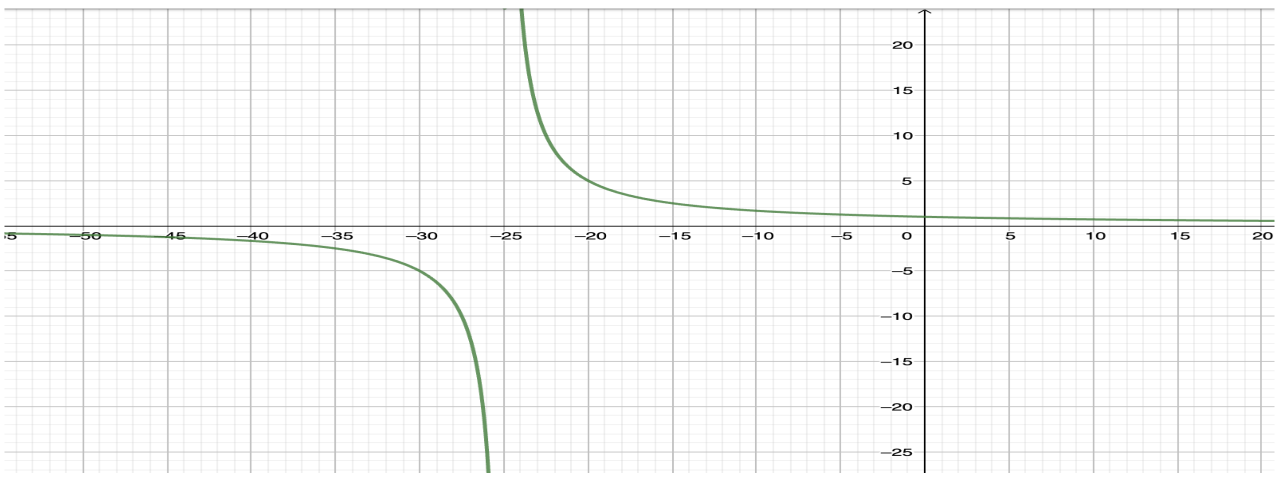
The effective capacity of the model is that:
(Effective data that is transmitted) / (Total transmission time)
= 5000 bits / ((5000 + 200*p) * 10^-6) seconds
= (25*10^6) / (25 + p) = (25*10^6) * (p + 25)^-1 bps (bits per second)

In order to find the optimal value, one should take the derivative of the above function and must equalize it to 0 in order to determine the local maxima which would maximize the throughput. The derivative is: -(25*10^6) * (p + 25)^-2.

Derivative of the function is always negative therefore it is a decreasing function. In this sense, one should select the smallest possible packet number to maximize the throughput. Since the smallest possible value for the number of packets is 1, **the number of packets must be 1.**

To find the throughput value at 1, basically input the value 1 to the (25*10^6) * (p + 25)^-1 equation. Then the answer is: **25/26 * 10^6 bits per second which is equal to 0.9615 Mbps.**

Function:



Function

Derivative: