



Computer Networks - Final Exam

December 22, 2017

Duration: 2:30 hours, closed book.

- This is a closed-book exam.
- Please write your answers on these sheets in a readable way, in English or in French.
- Please do **not** use a red pen.
- You can use extra sheets if necessary (don't forget to put your name on them).
- The total number of points is 100.
- This document contains 25 pages.
- Good luck!

Full Name (Nom et Prénom):

SCIPER No:

Division: Communication Systems Computer Science
 Other (mention it):

Year: Bachelor Year 2 Bachelor Year 3
 Other (mention it):

(answers to the questions are shown in italic and blue)

Problem 1

(15 points)

For each question, please circle a single best answer.

1. When is packet switching a better solution than connection switching?
 - (a) Always.
 - (b) When we need strict network-performance (packet loss and delay) guarantees.
 - (c) When the probability of all users sending traffic simultaneously is low. *(Correct)*
 - (d) When the probability of all users sending traffic simultaneously is high.
2. Which of the following types of delay, that packet p may experience, depends on the size of p ?
 - (a) The propagation delay.
 - (b) The transmission delay. *(Correct)*
 - (c) The queuing delay.
 - (d) None of the above.
3. Consider a switch with an initially empty buffer of infinite capacity and one outgoing link with transmission rate 500 Kbps. If groups of 1000 packets, each one of length 1 Kbits, arrive simultaneously at the switch every 1 sec, what is the maximum queuing delay experienced?
 - (a) Less than 2 seconds.
 - (b) Exactly equal to 2 seconds.
 - (c) Between 2 and 3 seconds.
 - (d) Infinite. *(Correct)*
4. All the end-hosts of IP subnet A experience low throughput when accessing web servers that are outside the subnet. What could A 's network operator do to potentially reduce these delays?
 - (a) Implement a proxy web server inside A .
 - (b) Increase the bandwidth of the outgoing link of A 's gateway IP router.
 - (c) Increase the number of connection links to A 's access ISP.
 - (d) All of the above are correct. *(Correct)*
5. Consider a user u that owns a file and wants to distribute it to a set S of other users. The total distribution time of the file is better in a peer-to-peer architecture than in a client-server architecture, if:
 - (a) The upload rate of user u is higher than the aggregate (total) download rate all the users in S .
 - (b) The upload rate of user u is lower than the aggregate download rate of all the users in S . *(Correct)*
 - (c) The number of users in S is small.
 - (d) Never, it is always slower.

6. In which case would a link-state routing algorithm and a distance-vector routing algorithm, applied to the same network, produce different results (different paths between the same pair of IP routers)?
- (a) They will always produce different results.
 - (b) If they use different link cost metrics. *(Correct)*
 - (c) If the network graph contains loops (cycles).
 - (d) If the network is particularly small.
7. An end-host wants to communicate with 3 web servers that it has never accessed before. What is the minimum number of sockets that it will need in total?
- (a) Less than 4 sockets.
 - (b) 4 sockets. *(Correct)*
 - (c) 6 sockets.
 - (d) More than 6 sockets.
8. An IP router receives a packet with destination IP address 192.168.0.50. If the router's forwarding table contains the following destination prefixes, which one will the router choose as the best match ?
- (a) 0.0.0.0/0
 - (b) 192.168.0.0/16
 - (c) 192.168.0.0/24
 - (d) 192.168.0.50/32 *(Correct)*
9. When is Go-Back-N a better solution than Selective Repeat?
- (a) The network occasionally drops a packet.
 - (b) The network frequently drops bursts (groups of back-to-back) packets. *(Correct)*
 - (c) The network never drops or corrupts packets.
 - (d) Go-Back-N can never be better than Selective Repeat.
10. What would happen if we turned the entire Internet into a single IP subnet (replaced all IP routers with link-layer switches)?
- (a) We would need significantly larger forwarding tables. *(Correct)*
 - (b) We would need significantly fewer network devices.
 - (c) Network performance (packet loss, delay, throughput) would, in general, improve.
 - (d) Nothing would really change.

11. Alice sends to Bob $K_B^+ \{m\}$, where m is a message and K_B^+ is B 's true/certified public key. Which of the following is true?
- (a) Authenticity of m is guaranteed.
 - (b) Authenticity of m is not guaranteed. (*Correct*)
 - (c) Confidentiality of m is not guaranteed.
 - (d) Both (b) and (c) are true, i.e., neither authenticity nor confidentiality are guaranteed.
12. Alice wants to send message m to Bob and wants to ensure authenticity. Assume that H is a globally known cryptographic hash function, K_B^+ is Bob's true/certified public key, K_A^- is Alice's private key, and that Alice's true/certified public key (K_A^+) is known to Bob. Alice achieves her goal by sending:
- (a) $[m \mid H(m)]$.
 - (b) $[m \mid K_B^+ \{H(m)\}]$.
 - (c) $[m \mid K_A^- \{H(m)\}]$. (*Correct*)
 - (d) None of the above.
13. Alice sends to Bob $H(m)$, where m is a message that takes values from a globally known set V , and H is a globally known cryptographic hash function. Eve is an adversary sitting on the communication channel between Alice and Bob, and she wants to recover m . Which of the following is true?
- (a) It is impossible for Eve to recover m .
 - (b) The larger the domain V is, the easier it is for Eve to recover m .
 - (c) The smaller the domain V is, the easier it is for Eve to recover m . (*Correct*)
 - (d) None of the above.
14. Alice and Bob cannot use any cryptography, but Bob knows Alice's true IP address. Persa is an adversary, located in a different IP subnet from Alice and Bob, and she is **not** on the communication channel from Alice to Bob (she cannot read or modify the messages exchanged between Alice and Bob). Which of the following is true?
- (a) If Alice and Bob communicate over UDP, then Persa can easily impersonate Alice. (*Correct*)
 - (b) If Alice and Bob communicate over TCP, then Persa can easily impersonate Alice.
 - (c) Persa can easily impersonate Alice. Whether Alice and Bob use UDP or TCP is irrelevant.
 - (d) It is impossible for Persa to impersonate Alice.
15. Alice, Bob, and Persa are end-hosts located in the same IP subnet. The IP subnet does not implement any link-layer security measures (it only supports the link-layer functionality we saw in class). Alice and Bob have already exchanged a shared secret key (not known to Persa) and use it to encrypt and authenticate all messages they exchange. Which of the following is true?
- (a) Persa can read the messages exchanged between Alice and Bob.
 - (b) Persa can impersonate either Alice or Bob.
 - (c) Persa can launch any (wo)man-in-the-middle attack between Alice and Bob.
 - (d) None of the above. (*Correct*)

Problem 2

(45 points)

Consider the network in Figure 7, consisting of:

- End-hosts A , B and X , DNS server C , and web server D .
- IP routers R_1 , R_2 , R_3 , and R_4 .
- Link-layer switches S_1 and S_2 .
- The link costs are noted in Figure 7.
- End-hosts A , B and X use C as their local DNS server.

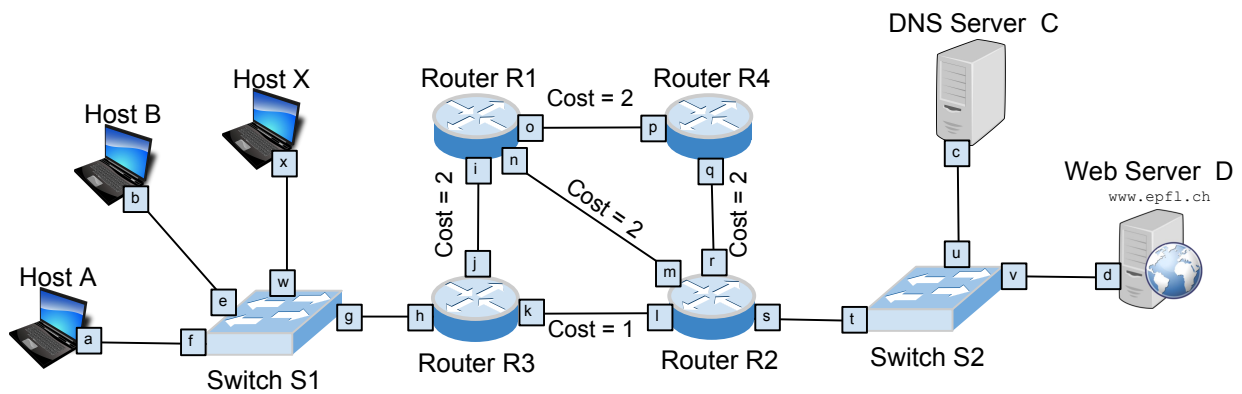


Figure 1: The Network Topology used in Problem 2

Question 1 (10 points):

Allocate an IP prefix to each IP subnet and an IP address to each network interface that needs one, following these rules:

- All IP addresses must be allocated from 8.8.8.0/24.
- Each IP subnet must be allocated the smallest possible IP prefix and must have one broadcast IP address.
- IP router (but not link-layer switch) interfaces have IP addresses.

Please explain how you compute each IP prefix and fill in Table 1 in the next page.

Subnet number	IP prefix	Interfaces and IP addresses	Broadcast IP address
<i>Example: 1</i>	10.1.1.0/24	x: 10.1.1.0 y: 10.1.1.1 z: 10.1.1.2	10.1.1.255
1	8.8.8.0/29	a : 8.8.8.0 b : 8.8.8.1 x : 8.8.8.2 h : 8.8.8.3	8.8.8.7
2	8.8.8.8/30	c : 8.8.8.8 d : 8.8.8.9 s : 8.8.8.10	8.8.8.11
3	8.8.8.12/30	o : 8.8.8.12 p : 8.8.8.13	8.8.8.15
4	8.8.8.16/30	q : 8.8.8.16 r : 8.8.8.17	8.8.8.19
5	8.8.8.20/30	k : 8.8.8.20 l : 8.8.8.21	8.8.8.23
6	8.8.8.24/30	i : 8.8.8.24 j : 8.8.8.25	8.8.8.27
7	8.8.8.28/30	n : 8.8.8.28 m : 8.8.8.29	8.8.8.31

Table 1: Allocation of IP prefixes and IP addresses for the network in Figure 7

Question 2 (5 points):

IP routers R_1 , R_2 , R_3 , and R_4 participate in a least-cost path routing algorithm. All the links between the routers are in good condition (no link has failed or is failing), and the algorithm has converged.

Show the forwarding tables of routers R_3 and R_2 .

a) Router R_3 :

Destination IP prefix	Output link
8.8.8.0/29	h
8.8.8.8/30	k
8.8.8.12/30	j
8.8.8.16/30	k
8.8.8.20/30	k
8.8.8.24/30	j
8.8.8.28/30	k

b) Router R_2 :

Destination IP prefix	Output link
8.8.8.0/29	l
8.8.8.8/30	s
8.8.8.12/30	r OR m
8.8.8.16/30	r
8.8.8.20/30	l
8.8.8.24/30	l
8.8.8.28/30	m

Question 3 (10 points):

All link-layer switches have just been rebooted, and all end-host and server caches are initially empty. Then, end-host A sends a DNS request to C .

State all the packets that are **transmitted or forwarded by all end-hosts and IP routers until A 's DNS request reaches C** . For example, if a packet follows the path $A \rightarrow R_1 \rightarrow R_2 \rightarrow D$, then you should state it 3 times: when it is transmitted by A , forwarded by R_1 , and forwarded by R_2 .

Answer by filling in Table 2. When you want to refer to the IP address of interface x , write " x ". When you want to refer to the MAC address of interface x , write " x ". If a field is not applicable, indicate that with a "-". To repeat a field from the above cell, write "".

#	Source MAC	Dest MAC	Source IP	Dst IP	Transp. prot.	Src Port	Dst Port	Application & Purpose
ex	x	y	w	v	UDP	5000	6000	HTTP GET <code>image.png</code>
1	a	broadcast	-	-	-	-	-	ARP request: h
2	h	a	-	-	-	-	-	ARP reply
3	a	h	a	c	UDP	2000	53	DNS query
4	k	broadcast	-	-	-	-	-	ARP request: l
5	l	k	-	-	-	-	-	ARP reply
6	k	l	a	c	UDP	2000	53	DNS query
7	s	broadcast	-	-	-	-	-	ARP request: c
8	c	s	-	-	-	-	-	ARP reply
9	s	c	a	c	UDP	2000	53	DNS query

Table 2: Packets transmitted or forwarded by all end-hosts and IP routers in Question 3

Question 4 (4 points):

Routers R_1 , R_2 , R_3 , and R_4 run the Bellman-Ford algorithm.

Show the final state of the Bellman-Ford tables for all the routers, **once the algorithm has converged**.

Answer by filling in the tables below. Here is an example table:

Router X :

from \ to	X	Y	Z
X	0	5(Y)	10(Y)
Y	5	0	5

10(Y) means that the path from router X to router Z has cost 10 and goes through X 's neighbor Y .

Router R_1 :

from \ to	R_1	R_2	R_3	R_4
R_1	0	2 (R_2)	2 (R_3)	2 (R_4)
R_2	2	0	1	2
R_3	2	1	0	3
R_4	2	2	3	0

Router R_2 :

from \ to	R_1	R_2	R_3	R_4
R_2	2 (R_1)	0	1 (R_3)	2 (R_4)
R_1	0	2	2	2
R_3	2	1	0	∞
R_4	2	2	∞	0

Router R_3 :

from \ to	R_1	R_2	R_3	R_4
R_3	2 (R_1)	1 (R_2)	0	3 (R_2)
R_1	0	2	2	2
R_2	2	0	1	2
R_4				

Router R_4 :

from \ to	R_1	R_2	R_3	R_4
R_4	2 (R_1)	2 (R_2)	3 (R_2)	0
R_1	0	2	2	2
R_2	2	0	1	2
R_3				

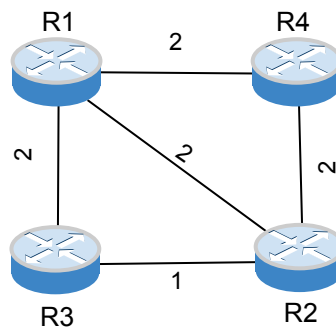


Figure 2: Part of the Network Topology used in Problem 2

Question 5 (6 points):

The link between routers R_1 and R_2 fails.

Show the Bellman-Ford tables for all the routers **from the moment the link fails and until the algorithm has re-converged**. Write clearly after how many iterations (neighbor exchanges) the algorithm re-converges.

Answer by filling in the tables below. We have provided tables for two iterations, but the algorithm may reconverge faster (in which case you just leave some tables empty).

State after the link failure:

Router R_1 :

from \ to	R_1	R_2	R_3	R_4
R_1	0	3 (R_3)	2 (R_3)	2 (R_4)
R_2				
R_3	2	1	0	3
R_4	2	2	3	0

Router R_2 :

from \ to	R_1	R_2	R_3	R_4
R_2	3 (R_3)	0	1 (R_3)	2 (R_4)
R_1				
R_3	2	1	0	∞
R_4	2	2	∞	0

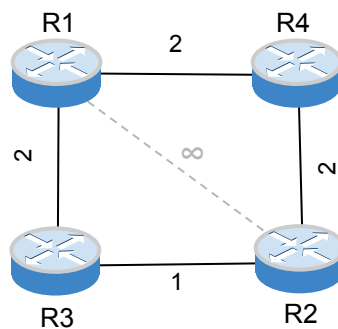


Figure 3: Part of the Network Topology used in Problem 2 after the link failure.

After the first exchange:

Router R_1 :

from \ to	R_1	R_2	R_3	R_4
R_1	0	3 (R_3)	2 (R_3)	2 (R_4)
R_2				
R_3	2	1	0	3
R_4	2	2	3	0

Router R_2 :

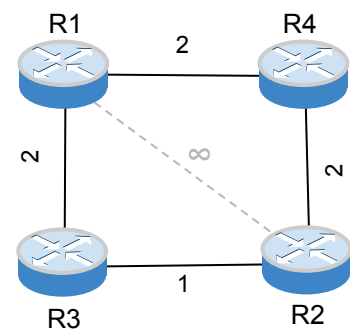
from \ to	R_1	R_2	R_3	R_4
R_2	3 (R_3)	0	1 (R_3)	2 (R_4)
R_1				
R_3	2	1	0	∞
R_4	2	2	∞	0

Router R_3 :

from \ to	R_1	R_2	R_3	R_4
R_3	2 (R_1)	1 (R_2)	0	3 (R_2)
R_1	0	∞	2	2
R_2	∞	0	1	2
R_4				

Router R_4 :

from \ to	R_1	R_2	R_3	R_4
R_4	2 (R_1)	2 (R_2)	3 (R_2)	0
R_1	0	3	2	2
R_2	3	0	1	2
R_3				



After the second exchange:

Router R_1 :

from \ to	R_1	R_2	R_3	R_4
R_1				
R_2				
R_3				
R_4				

Router R_2 :

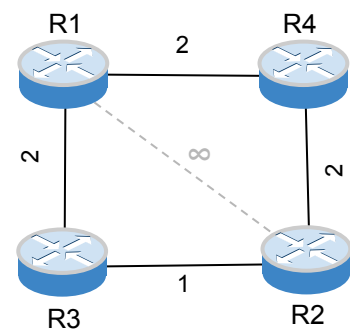
from \ to	R_1	R_2	R_3	R_4
R_2				
R_1				
R_3				
R_4				

Router R_3 :

from \ to	R_1	R_2	R_3	R_4
R_3				
R_1				
R_2				
R_4				

Router R_4 :

from \ to	R_1	R_2	R_3	R_4
R_4				
R_1				
R_2				
R_3				



Question 5 (5 points):

Suppose there is a firewall between S_1 and R_3 , which allows the minimum amount of traffic necessary such that all end-hosts attached to S_1 can access web pages from web server D (and no other end-hosts can access any other service).

Fill the firewall table (use as many rows and columns as necessary), clearly showing the fields and content of its access control rules.

Action	Protocol	Source IP	Dest. IP	Source Port	Dest. Port
allow	UDP	8.8.8.0/29	c	>1023	53
allow	UDP	c	8.8.8.0/29	53	>1023
allow	TCP	8.8.8.0/29	d	>1023	80
allow	TCP	d	8.8.8.0/29	80	>1023
deny	all	all	all	all	all

Problem 3

(40 points)

Assume the following for all the questions in this problem:

- The maximum segment size is $MSS = 1$ byte.
- The round trip time (RTT) is the same in both directions.
- Each TCP receiver sends an ACK every time it receives a data segment.
- Each TCP sender's retransmission timeout is fixed and equal to $2 \times RTT$.
- Fast-retransmit is disabled (TCP senders do not fast-retransmit).

When you complete the diagrams in Questions 1, 2, and 3, the following information should be visible:

- All the segments (including the ACKs) exchanged between the communicating end-hosts.
- The sequence numbers of all data segments.
- The acknowledgment numbers of all ACKs.
- The state of the TCP sender's congestion-control algorithm.
- The status of the TCP sender's congestion window and its size (`cwnd`) in bytes.
- The value of the TCP sender's congestion threshold (`ssthresh`) in bytes.
- If your answer includes any dropped segments or ACKs, mark them clearly.
- If your answer includes any timeouts, mark them clearly and indicate the duration of each timeout and the sequence number of the data segment that timed out.

Question 1 (10 points):

Alice has already established a persistent TCP connection to Bob, but has not sent any data over it.

At time T_0 , Alice starts sending, over this connection, a file of size 13 bytes. The sequence number of the first segment she sends (the one that carries the first byte of the file) is 1.

At time T_1 , Alice has sent 8 bytes out of the 13 bytes of the file, Bob has acknowledged receiving all of them, and the following is true for the TCP congestion-control algorithm running in Alice's transport layer:

- The congestion window is $cwnd = 2$ bytes.
- The congestion threshold is $ssthresh = 2$ bytes.

After time T_1 , none of the segments transmitted by Alice or Bob are dropped.

Show what happens **from time** T_1 and until the file transfer completes, by completing the diagram in Figure 4 in the next page.

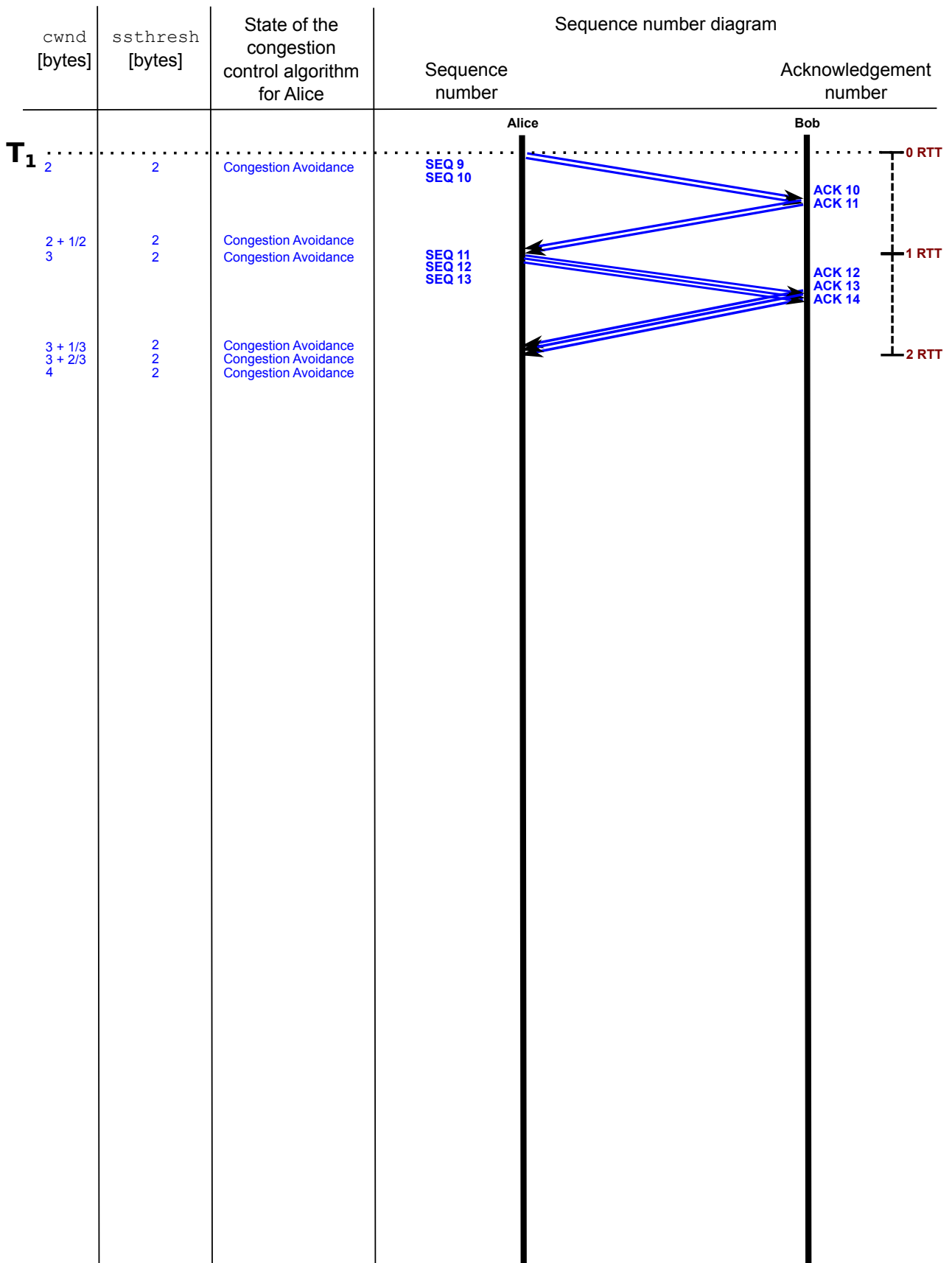


Figure 4: Sequence diagram to be completed for Question 1

Since Alice was already in the congestion control phase the congestion control increments by 1. See sequence diagram in Figure 4.

Question 2 (10 points):

Think of a sequence of events **from time T_0 until time T_1** , that could have led Alice's TCP congestion-control algorithm to the state it was at time T_1 (i.e., Alice has sent 8 bytes out of the 13 bytes of the file, Bob has acknowledged all of them, the congestion window size is $cwnd=2$ bytes and the congestion threshold is $ssthresh=2$ bytes).

Describe your scenario by completing the diagram in Figure 5 in the next page.

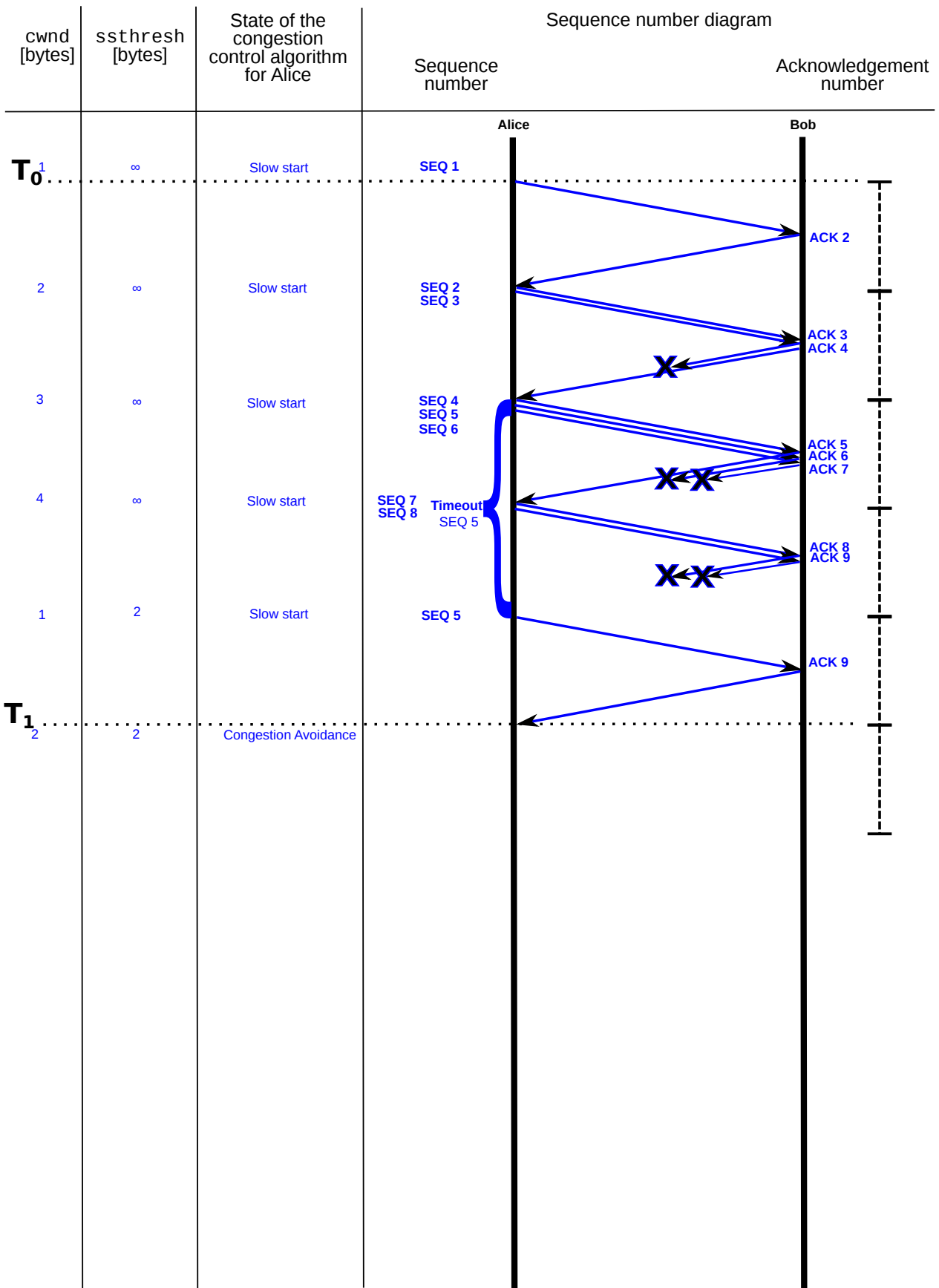


Figure 5: Sequence diagram to be completed for Question 2

See sequence diagram in Figure 5.

Question 3 (10 points):

Cecilia has already established a persistent TCP connection to Dario, but has not sent any data over it.

At time T_0 , Cecilia starts sending, over this connection, a file of size 7 bytes. The sequence number of the first segment she sends (the one that carries the first byte of the file) is 1.

The communication channel from Cecilia to Dario consists of two network paths: one through IP routers R_1 , R_2 , and R_3 , and one through IP routers R_1 , R_4 , and R_3 , as shown in Figure 6.

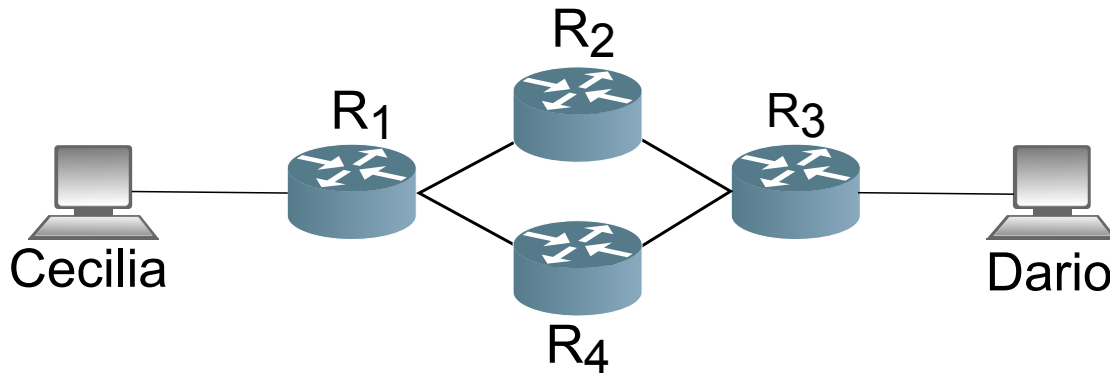


Figure 6: Topology for Questions 3 and 4

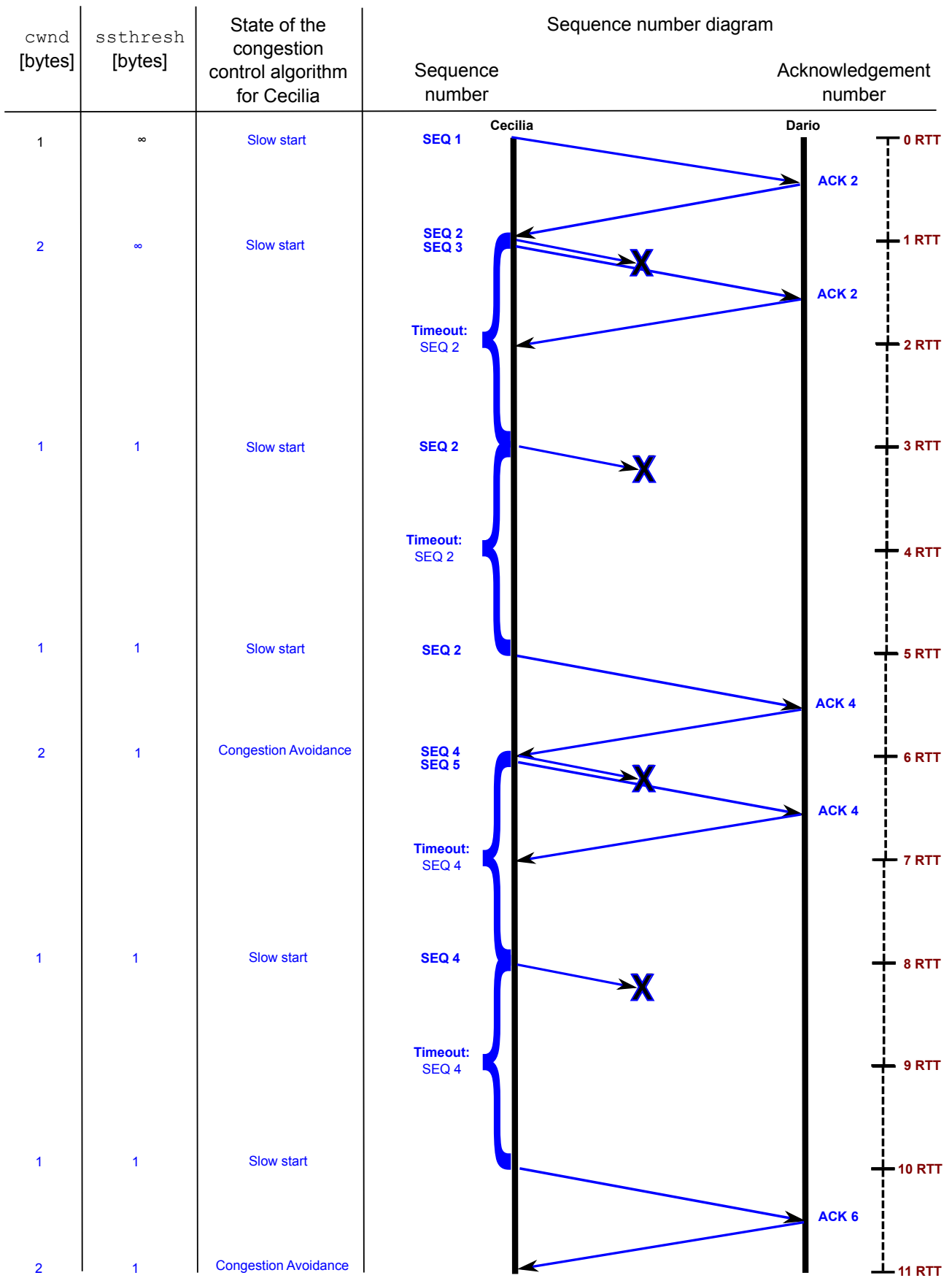
IP router R_1 forwards the packets it receives from Cecilia to Dario through alternating paths: the first packet it receives through R_2 , the next packet through R_4 , the next packet through R_2 , the next packet through R_4 , and so on.

The two paths have different packet-drop properties: the path through R_2 never drops any packet, while the path through R_4 drops all the packets.

All the ACKs sent by Dario reach Cecilia successfully.

Show what happens from time T_0 and until the file transfer completes, by completing the diagram in Figure 6 in the next page.

See sequence diagram in Figure 6.



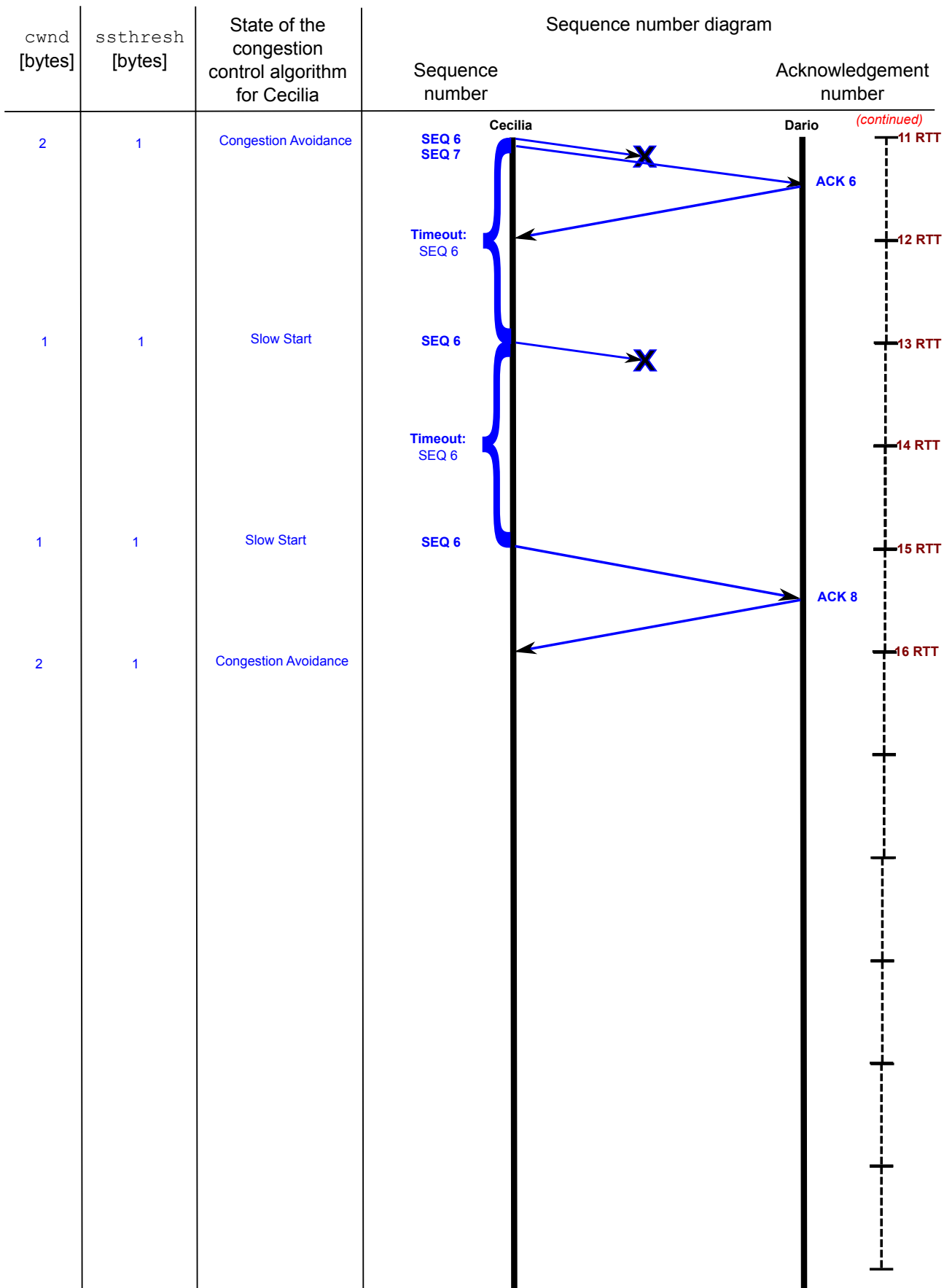


Figure 6: Sequence diagram to be completed for Question 3

Question 4 (10 points):

The TCP congestion-control algorithm was designed under the assumption that there exists one network path between each TCP sender and receiver. The goal of the algorithm is to make the sender transmit at the maximum rate supported by the network path without introducing congestion.

However, when there are multiple network paths between a TCP sender and receiver with different packet-drop rates, the TCP congestion-control algorithm fails to make the sender transmit at the maximum aggregate rate supported by all the network paths.

(a) Why does the algorithm fail in this way? Why does the presence of multiple network paths with different packet-drop rates affect the outcome of the algorithm?

TCP congestion control assumes a single path and reacts to loss; i.e. it uses one single congestion window and one single threshold (no matter how many network paths are actually used), in order to adapt its sending rate to the underlying congestion.

If there are multiple paths with different loss rates, then the higher loss rate of one path will affect (and may reduce) the sending rate over another path that has lower loss rate. In this way, the algorithm gets confused and cannot transmit at the maximum aggregate rate supported by all the network paths.

(b) Would it be possible to change the algorithm such that it does not fail in this way (i.e., it makes the TCP sender transmit at the maximum aggregate rate supported by all the network paths)?

The only way of transmitting at the maximum aggregate rate supported by all paths is to keep different state (i.e. different cwnd and ssthresh) and run a different TCP congestion control algorithm for each network path that is used.

However, TCP's congestion control algorithm runs within the transport layer, which knows nothing about network paths. Also, TCP cannot choose a specific path, it can only hold back the sending rate. The path is up to the network layer to be chosen.

An idea could be to give the necessary routing/path information to TCP. Unfortunately, such an attempt would violate the layering design, and as we have learned, breaking layering would not be good for other reasons (like flexibility and modularity).

*Hence, the answer to this question is **no**.*

Scratch Paper

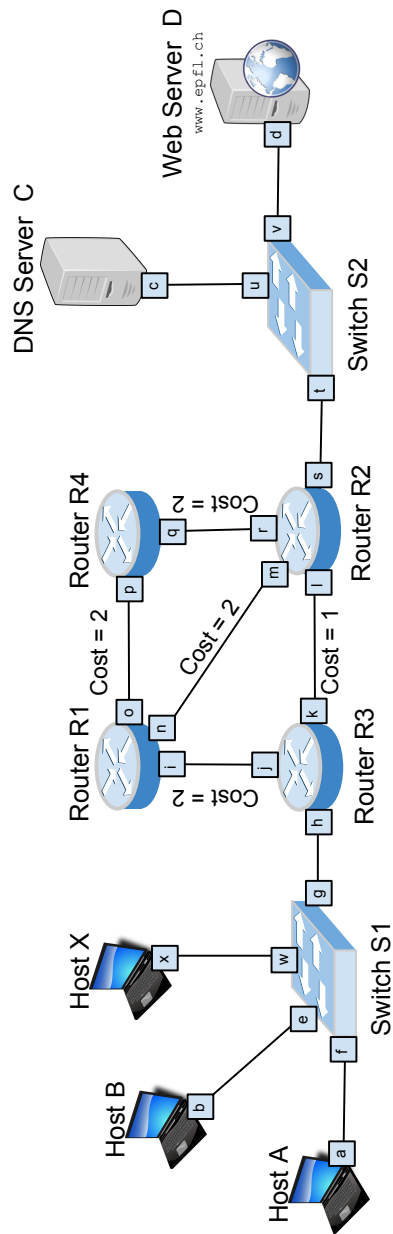


Figure 7: The Network Topology used in Problem 2