Computer Networks - Final Exam

January 23, 2023

Duration: 2:15 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 50.
- This document contains 23 pages.
- Good luck!

Last Name (Nom):
First Name (Prénom):
SCIPER No:

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

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

Problem	Points achieved	Out of
1		5
2		25
3		20
Total		50

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

Problem 1 (5 points)

For each question, please circle a single best answer.

1. The larger a packet is,
   (a) the higher the transmission delay it experiences. *(Correct)*
   (b) the higher the propagation delay it experiences.
   (c) the higher the queuing delay it experiences.
   (d) the lower the processing delay it experiences.

2. The longer a link is, the higher...
   (a) the transmission delay it introduces.
   (b) the propagation delay it introduces. *(Correct)*
   (c) the queuing delay it introduces.
   (d) the throughput it enables.

3. Two packets of the same size traverse the same path (the same sequence of links and packet switches) but at different points in time. They may experience different...
   (a) transmission delays.
   (b) propagation delays.
   (c) queueing delays. *(Correct)*
   (d) nothing (they cannot experience different delays).

4. Paths P_1 and P_2 have zero queuing and processing delays. Path P_1 has a significantly higher propagation delay than path P_2. This means that the following is necessarily (always) true:
   (a) A small packet traversing P_1 experiences significantly higher end-to-end delay than a small packet traversing P_2. *(Correct)*
   (b) The maximum throughput achievable over P_1 is significantly smaller than the maximum throughput achievable over P_2.
   (c) Both of the above.
   (d) None of the above.

5. During a given time interval, Path P_1 offers higher throughput than path P_2. This means that the following is necessarily (always) true:
   (a) P_1's lowest-capacity link has a higher transmission rate than P_2's lowest-capacity link.
   (b) P_1 is experiencing less congestion than P_2 during the given time interval.
   (c) Both of the above.
   (d) None of the above. *(Correct)*

6. End-systems A and B are connected over a sequence of two links (with one packet switch between them). A is sending equally sized packets to B, and there is no other traffic on the Internet. Is it possible that A's traffic experiences queuing delay at the switch?

   (a) No, because this is a very simple topology.

   (b) No, because there is no other traffic on the Internet.

   (c) Yes, because queuing delay is always possible no matter what.

   (d) Yes, if the second link has a smaller transmission rate than the first link. *(Correct)*

7. End-system A is sending traffic to end-system B over a TCP connection. A link on the path from A to B becomes congested and starts dropping and delaying packets. How will TCP's congestion-control algorithm react?

   (a) It will send the traffic through a different path.

   (b) It will adapt the receiver window.

   (c) It will adapt the congestion window. *(Correct)*

   (d) It will end the TCP connection.

8. Alice sends messages to Bob. What information is needed to achieve confidentiality using public-key cryptography?

   (a) Alice needs to know Bob's public key. *(Correct)*

   (b) Bob needs to know Alice's public key.

   (c) They need to know each other's public key.

   (d) They don't need to know anything.

9. Alice sends messages to Bob. What information is needed to achieve authenticity and data integrity using public-key cryptography?

   (a) Alice needs to know Bob's public key.

   (b) Bob needs to know Alice's public key. *(Correct)*

   (c) They need to know each other's public key.

   (d) They don't need to know anything.

10. Which of the following is true about ”default gateway” and “proxy ARP”?

   (a) They are the exact same mechanism.

   (b) They both have the same goal: for an end-system or IP router to map an IP address to a MAC address. *(Correct)*

   (c) They both have the same goal: for a link-layer switch to map an IP address to a MAC address.

   (d) The former is used by end-systems, whereas the latter is used by IP routers.

Problem 2 **(25 points)**

Consider the Autonomous System AS0 shown in Figure 1, which includes:

- End-systems A_1, ... A_{1000} (there are 1000 of them).
- End-systems B_1, ... B_{600} (there are 600 of them).
- End-systems C_1, ... C_{100} (there are 100 of them).
- DNS server `dns.epfl.ch`.
- Web server `www.epfl.ch`.
- IP routers R_1, R_2, R_3, and R_4.
- Various link-layer switches (not explicitly shown).

The orange boxes represent network interfaces. For example, IP router R_2 has network interfaces y, z, v, and u.

Each link between IP routers shown in the figure is annotated with its routing cost, and it has the same cost in each direction.

All end-systems in AS0 use `dns.epfl.ch` as their local DNS server.

The intra-domain routing protocol of AS0 uses the Dijkstra algorithm that we saw in class.

The time-to-live (TTL) of DNS records and ARP-table entries is 24 hours.

You can find a copy of this network topology at the end of the exam (next to last page). You can detach it so that you can look at the topology while solving the problem, without having to turn the pages back and forth.

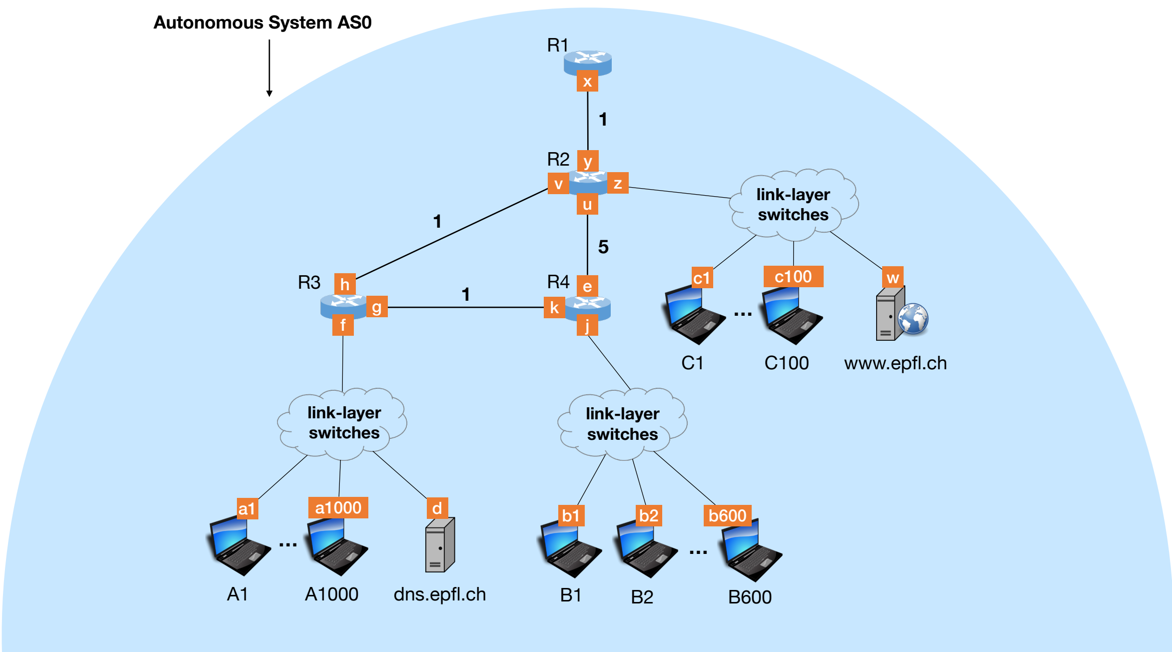

Figure 1: Network topology for Problem 2.

Question 1 (5 points):

Allocate an IP prefix to each IP subnet of AS0 following these rules:

- All IP prefixes must be allocated from $5.0.0.0/8$.
- Each IP subnet must be allocated the smallest possible IP prefix.
- Assume one IP address per end-system and per-IP-router interface (but not for link-layer switches).
- Assume one broadcast IP address per IP subnet.
- You do not need to assume a network address per IP subnet (but it's not a mistake if you do).

Explain in one or two sentences how you compute each IP prefix.

In decreasing order of size, we have 5 IP subnets: A (contains end-systems A_x), B, C, R_{12} (subnet between R_1 and R_2), R_{23}, R_{34}, and R_{24}. We will allocate IP prefixes in this order, but there are many possible solutions.

- IP subnet A needs 1001 addresses for end-systems, one for interface f, and one broadcast address. To assign 1003 addresses we need 10 bits ($2^{10} = 1024$) and thus the mask size is $32 - 10 = 22$bits. We can therefore assign the following address range:

  0000 0101.0000 0000.0000 00xx.xxxx xxxx

  which is equivalent to:

  5.0.0.0/22

- IP subnet B needs 600 addresses for end-systems, one for interface j, and one broadcast address. To assign 602 addresses we need 10 bits and thus the mask size is $32 - 10 = 22$bits. Continuing from where the previous range ends, we have:

  0000 0101.0000 0000.0000 01xx.xxxx xxxx

  which is equivalent to:

  5.0.4.0/22

- IP subnet C needs 101 addresses for end-systems, one for interface z, and one broadcast address. To assign 103 addresses we need 7 bits and thus the mask size is $32 - 7 = 25$bits. Continuing from where the previous range ends, we have:

  0000 0101.0000 0000.0000 1000.0xxx xxxx

  which is equivalent to:

  5.0.8.0/25

- IP subnet R_{12} needs 2 addresses for interfaces x and y and one broadcast address. To assign 3 addresses we need 2 bits and thus the mask size is $32 - 2 = 30$bits. Continuing from where the previous range ends, we have:

  0000 0101.0000 0000.0000 1000.1000 00xx

  which is equivalent to:

  5.0.8.128/30

- IP subnet R_{23} needs 2 addresses for interfaces v and h and one broadcast address. To assign 3 addresses we need 2 bits and thus the mask size is $32 - 2 = 30$bits. Continuing from where the previous range ends, we have:

  0000 0101.0000 0000.0000 1000.1000 01xx

  which is equivalent to:

  5.0.8.132/30

- IP subnet R_{34} needs 2 addresses for interfaces g and k and one broadcast address. To assign 3 addresses we need 2 bits and thus the mask size is $32 - 2 = 30$bits. Continuing from where the previous range ends, we have:

  0000 0101.0000 0000.0000 1000.1000 10xx

  which is equivalent to:

  5.0.8.136/30

- IP subnet R_{24} needs 2 addresses for interfaces w and e and one broadcast address. To assign 3 addresses we need 2 bits and thus the mask size is $32 - 2 = 30$bits. Continuing from where the previous range ends, we have:

  0000 0101.0000 0000.0000 1000.1000 11xx

  which is equivalent to:

  5.0.8.140/30

Question 2 (8 points):

All link-layer switches have just been rebooted, and all end-system caches/ARP tables are initially empty. All routers have populated their forwarding tables according to the intra-domain routing protocol.

The user of end-system A_1 visits web page `www.epfl.ch`, which contains no embedded objects (e.g., no images). Immediately after A_1's user views `www.epfl.ch`, the user of end-system B_1 visits the same web page.

State all the packets that are **received, forwarded, or transmitted by router R_3 as a result of $\underline{B_1}$'s actions and until B_1's user can view the web page**. For example, if router R_3 receives and forwards an IP packet, you should state that packet twice: once to state that R_3 received it, and once to state that R_3 forwarded it.

Answer by filling in Table 1. To denote the IP address or the MAC address of interface x, write "x". If a field is not applicable, write "–". To repeat a field from the above cell, write ".". To illustrate the format, we have provided a hypothetical example entry (thee first entry in the table).

#	Source MAC	Dest MAC	Source IP	Dst IP	Transp. prot.	Src Port	Dst Port	Application & Purpose
1	k	broadcast	-	-	-	-	-	ARP request for g's MAC
2	g	k	-	-	-	-	-	ARP reply
3	k	g	b_1	d	UDP	2000	53	DNS request for w's IP
4	f	broadcast	-	-	-	-	-	ARP request for d's MAC
5	d	f	-	-	-	-	-	ARP reply
6	f	d	b_1	d	UDP	2000	53	DNS request for w's IP
7	d	f	d	b_1	UDP	53	2000	DNS reply
8	g	k	d	b_1	UDP	53	2000	DNS reply
9	k	g	b_1	w	TCP	4000	80	TCP SYN
10	h	v	b_1	w	TCP	4000	80	TCP SYN
11	v	h	w	b_1	TCP	80	4000	TCP SYN ACK
12	g	k	w	b_1	TCP	80	4000	TCP SYN ACK
13	k	g	b_1	w	TCP	4000	80	HTTP GET index
14	h	v	b_1	w	TCP	4000	80	HTTP GET index
15	v	h	w	b_1	TCP	80	4000	HTTP OK
16	g	k	w	b_1	TCP	80	4000	HTTP OK

Table 1: Packets received, forwarded, or transmitted by router R_3 in Question 2.

Question 3 (4 points):

(a) Show the entries of R_3's forwarding table that match packets addressed to AS0.

Destination IP prefix	Output interface
5.0.0.0/22	f
5.0.4.0/22	g
5.0.8.0/25	h
5.0.8.128/30	h
5.0.8.132/30	h
5.0.8.136/30	g
5.0.8.140/30	h

Table 2: R_3's forwarding table for Question 3(a).

(b) Suppose the link between R_2 and R_3 is cut (and cannot be repaired). Show the entries of R_3's forwarding table that match packets addressed to AS0 after the intra-domain routing protocol reconverges. Show only the entries that change after the cut.

Destination IP prefix	Output interface
5.0.8.0/25	g
5.0.8.128/30	g
5.0.8.132/30	g
5.0.8.140/30	g

Table 3: R_3's forwarding table for Question 3(b).

(c) Suppose that, after the link between R_2 and R_3 is cut, the link between R_1 and R_2 is also cut (and cannot be repaired). Show the entries of R_3's forwarding table that match packets addressed to AS0 after the intra-domain routing protocol reconverges. Show only the entries that change after the (second) cut.

Destination IP prefix	Output interface
5.0.8.128/30	–

Table 4: R_3's forwarding table for Question 3(b).

Question 4 (4 points):

Ignore the events of Question 3 (pretend they didn't happen).

Suppose the IP subnet behind interface j of IP router R_4 is a private IP subnet (i.e., it uses private IP addresses), and R_4 acts as a Network Address Translation (NAT) gateway (for both TCP and UDP traffic).

(a) How does this change your answer to Question 2? Write down only the modified lines; use the '#' column to refer to the designated row in Table 1. If a field between the original and modified line remains the same, leave it empty. To illustrate the format, we have provided a hypothetical example entry (the first entry in the table).

#	Source MAC	Dest MAC	Source IP	Dst IP	Transp. prot.	Src Port	Dst Port	Application & Purpose
3			j			500		
6			j			500		
7				j			500	
8				j			500	
9			j			600		
10			j			600		
11				j			600	
12				j			600	
13			j			600		
14			j			600		
15				j			600	
16				j			600	

Table 5: Packets received, forwarded, or transmitted by router R_3 in Question 4.

(b) What state (information for ongoing communications) does R_4 need to keep to operate as a NAT gateway? Show its state at the end of the events of Question 2 (i.e., after B_1's user has received all the packets needed to view the web page).

It needs to keep a mapping between a packet's original (private) source IP address and original source port number to the packet's modified source port number.

#	Original IP	Original Port	Modified Port
1	b_1	2000	500
2	b_1	4000	600

Table 6: State of NAT gateway R_4 in Question 5.

Question 5 (4 points):

Considering the conditions of **Question 4**:

(a) Does the fact that R_4 is a NAT gateway restrict the number of parallel (simultaneous) TCP connections that B_1 can establish with other end-systems located in the same IP subnet? Justify your answer.

No. Connections inside the same subnet will not go through the NAT (no address translation is required), so it does not impose any restriction.

(b) Does the fact that R_4 is a NAT gateway restrict the number of parallel (simultaneous) TCP connections that B_1 can establish with other end-systems located in different IP subnets? Justify your answer.

Yes. NAT uses port numbers to map between the private and public sides, which limits the number of connections that can be established to the maximum size of the port field (2^{16}).

Problem 3 (20 points)

Assume the following for all the questions in this problem:

- Alice and Bob communicate using TCP at the transport layer.
- Fast Retransmit/Fast Recovery are DISABLED.
- The maximum segment size is `MSS` $= 1$ byte.
- The TCP timeout is 2 RTT, where RTT is the sender's estimate of the round trip time from sender to receiver. Both RTT and the TCP timeout stay fixed throughout the TCP connection.
- Transmission delays are negligible.
- A TCP receiver sends an ACK every time it receives a data segment.
- Alice and Bob do not use the Secure Sockets Layer (SSL) or any other security mechanism.

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

- All the segments (including the ACKs) exchanged between the communicating end-systems.
- The sequence numbers of all data segments sent from Alice to Bob.
- The acknowledgment numbers of all ACKs sent from Bob to Alice.
- The state of Alice's congestion-control algorithm.
- The size of Alice's congestion window (`cwnd`) in bytes.
- The value of Alice's congestion threshold (`ssthresh`) in bytes.
- Any dropped segments.
- If your answer includes any timeouts, mark them clearly (on the side where the timeout occurs) and indicate the sequence number of the data segment that timed out.

Question 1 (5 points):

A process running on Alice's computer has established (sometime in the past) a TCP connection with an online-store process running on Bob's computer and has sent to it some number of bytes. Bob's process does not send any data to Alice's process throughout this question. Bob always advertizes a receiver window of 1000 bytes to Alice.

At time t_0 Alice's congestion window is 4 bytes, and no packet loss has occurred since the establishment of the TCP connection. At that moment, Alice's process produces 12 bytes to send to Bob's process. Each byte produced by Alice's process carries a distinct query (command). The next sequence number (that Bob is expecting) is 1.

Alice's 3rd segment (counting from time t_0) is lost. No other segment is lost, corrupted, unpredictably delayed, or reordered.

Show all the segments sent by Alice and Bob starting at time t_0. Use the diagram in Figure 2 on the next page. You can find a copy of the same diagram at the end of the exam, to first do a draft, if you wish.

cwnd [bytes]	ssthresh [bytes]	State of the congestion control algorithm for Alice
4	∞	Slow Start
6	∞	Slow Start
1	3	Slow Start
2	3	Slow Start
3	3	Congestion Avoidance
3	3	Congestion Avoidance

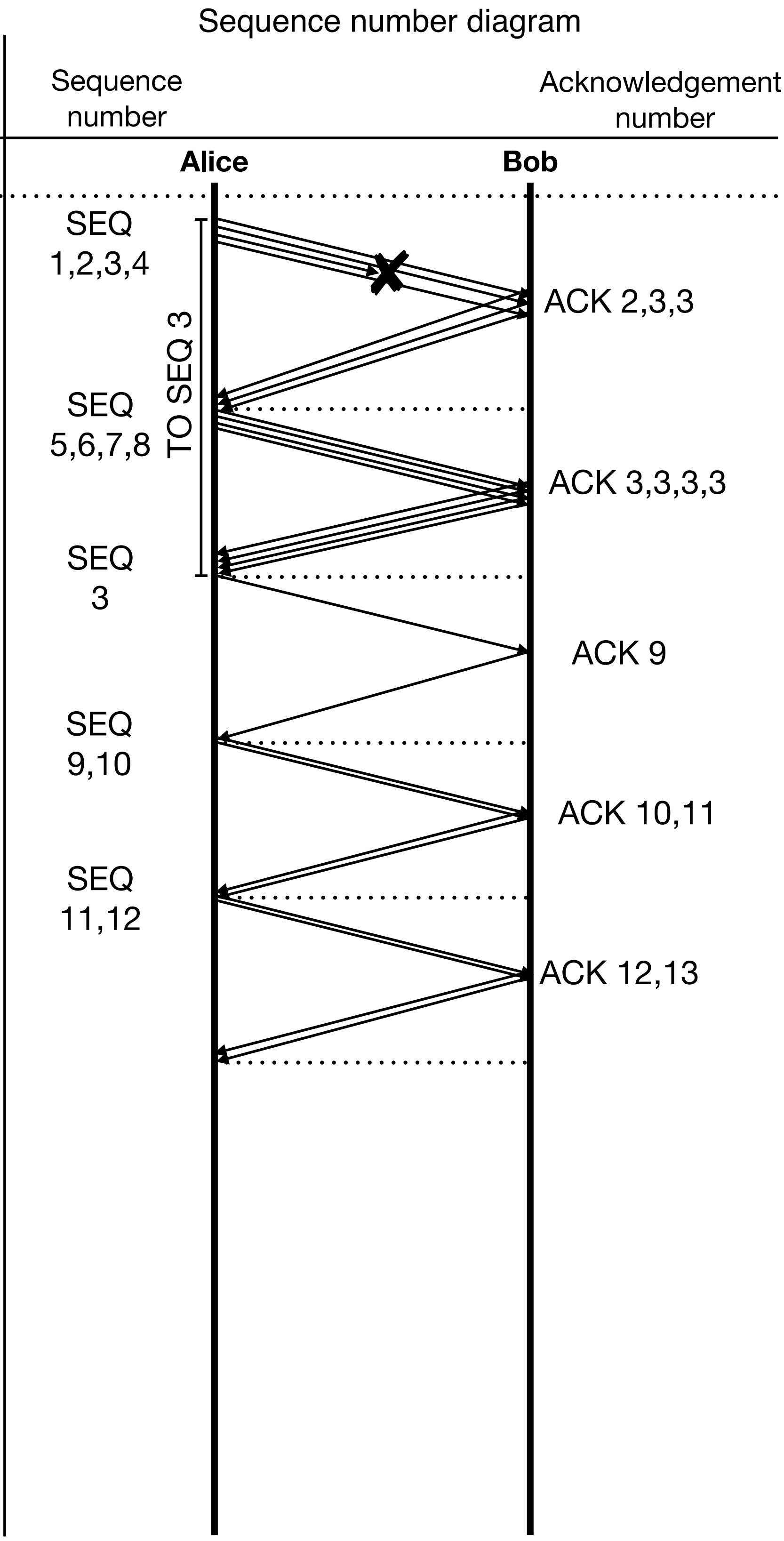

Figure 2: Sequence diagram to be completed for Question 1.

Question 2 (5 points):

Answer again Question 1 but under the following changed condition:

No segment is lost (not even Alice's 3rd segment), corrupted, or unpredictably delayed. However, starting at time t_0, a malicious switch on the path from Alice to Bob wants Bob's process to execute the queries produced by Alice's process in a different order than the one in which she produced and sent them. To this end, the malicious switch swaps each pair of segments sent by Alice, causing Bob to receive Alice's 2nd segment, then her 1st segment, then her 4th segment, then her 3rd segment, and so on.

Show all the segments sent by Alice and Bob starting at time t_0. Use the diagram in Figure 3 on the next page. You can find a copy of the same diagram at the end of the exam, to first do a draft, if you wish.

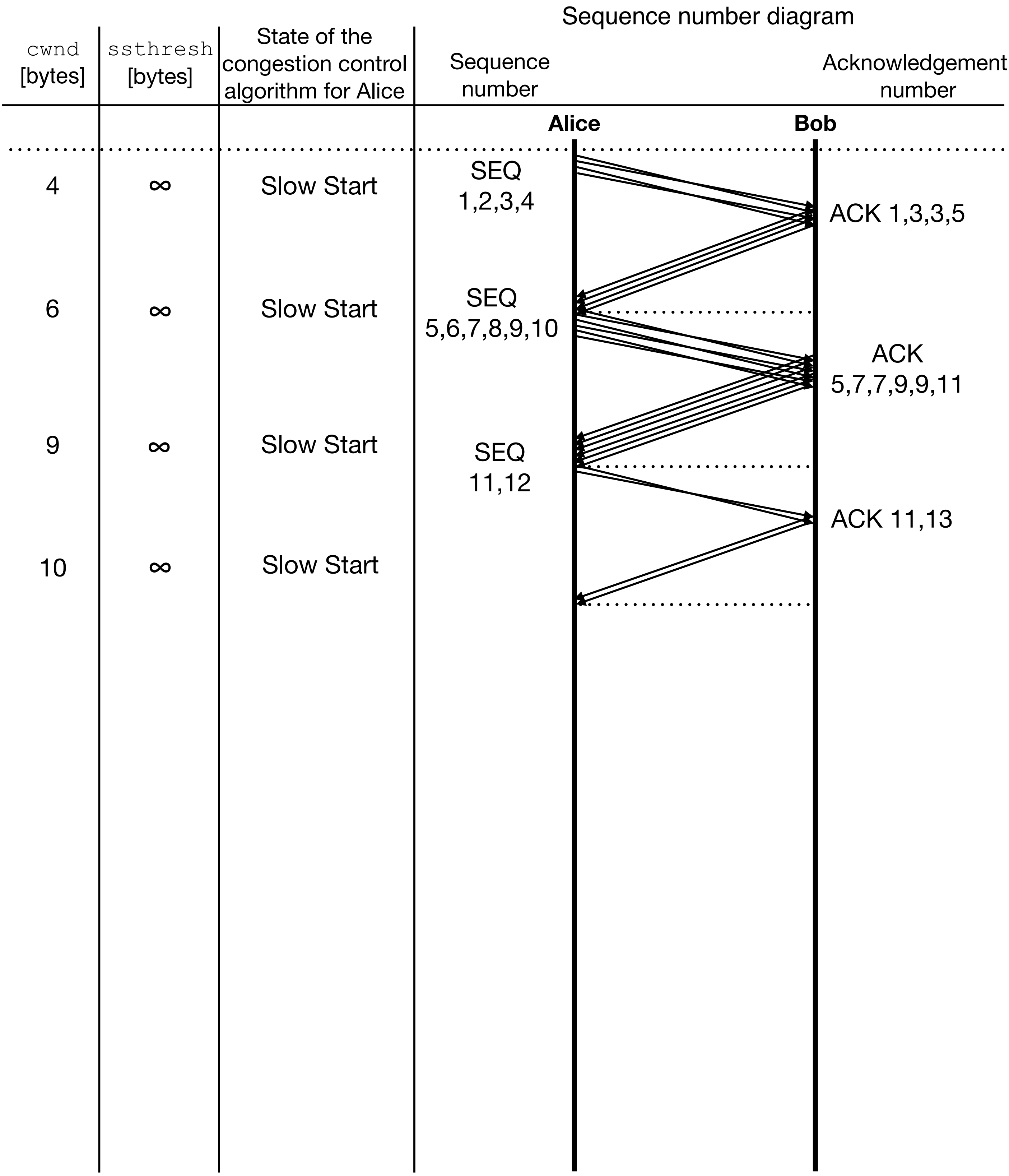

Figure 3: Sequence diagram to be completed for Question 2.

Question 3 (4 points):

Consider again the events of Question 2.

Does the malicious switch succeed in its attack? I.e., does it manage to convince Bob that Alice produced and sent her queries in a different order than she actually did? If yes, explain why. If not, explain what exactly the malicious switch should have done to succeed in its attack.

The attack does not succeed because TCP delivers segments to the application in order.
To carry out the attack, the switch also has to swap the sequence numbers and recalculate the checksum (both are fields of the TCP header).

Question 4 (6 points):

Ignore the previous questions. Consider the following topology, where R denotes transmission rate:

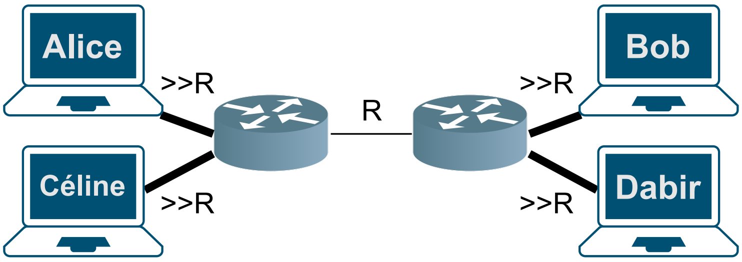

Figure 4: Network topology for Problem 3 question 4.

At time t_0, a process in Alice's computer and a process in Bob's computer establishes a TCP connection between them. From that point on, Alice's process produces a random amount of bytes for Bob's process, then stays silent for a random amount of time, then again produces a random amount of bytes for Bob's process, and so on; this continues indefinitely. Alice and Bob use the first TCP congestion-control algorithm that we saw in class (TCP-Tahoe, which does not do Fast Retransmit).

At time $t_1 > t_0$, a process in Céline's computer and a process in Dabir's computer establish a TCP connection between them. Once the connection is established, Céline's process produces an infinite amount of bytes for Dabir's process.

(a) Suppose Céline and Dabir use the same TCP congestion-control algorithm as Alice and Bob. Does the Céline-Dabir communication affect the throughput of the Alice-Bob communication? Justify your answer.

Yes. Both connections share a common bottleneck link, so the connection Céline-Dabir will decrease the bandwidth available to Alice-Bob.

(b) Suppose Céline and Dabir have the following goal for their communication: maximize their throughput, as long as they cause zero packet loss to the Alice-Bob communication.

Propose a congestion-control algorithm, to be used by Céline and Dabir, that achieves this goal. In particular, answer the following questions:

- What event(s) should cause this algorithm to increase its congestion window?
- What event(s) should cause this algorithm to decrease its congestion window?
- By how much would you suggest to increase or decrease the congestion window at each event?

Justify your answers.

Céline needs to determine that Alice started sending as soon as possible to backoff and wait for the communication to end without causing loss, so she uses RTT increases as signals of congestion instead of loss. While sending, Céline monitors the RTT and as soon as she sees an increase, she backs off aggresively so that Alice does not suffer any loss. To determine when Alice stops sending, Céline needs to send packets one at a time until she sees the RTT decrease again; then, she can start sending at full rate again, repeating the cycle.

Scratch Paper

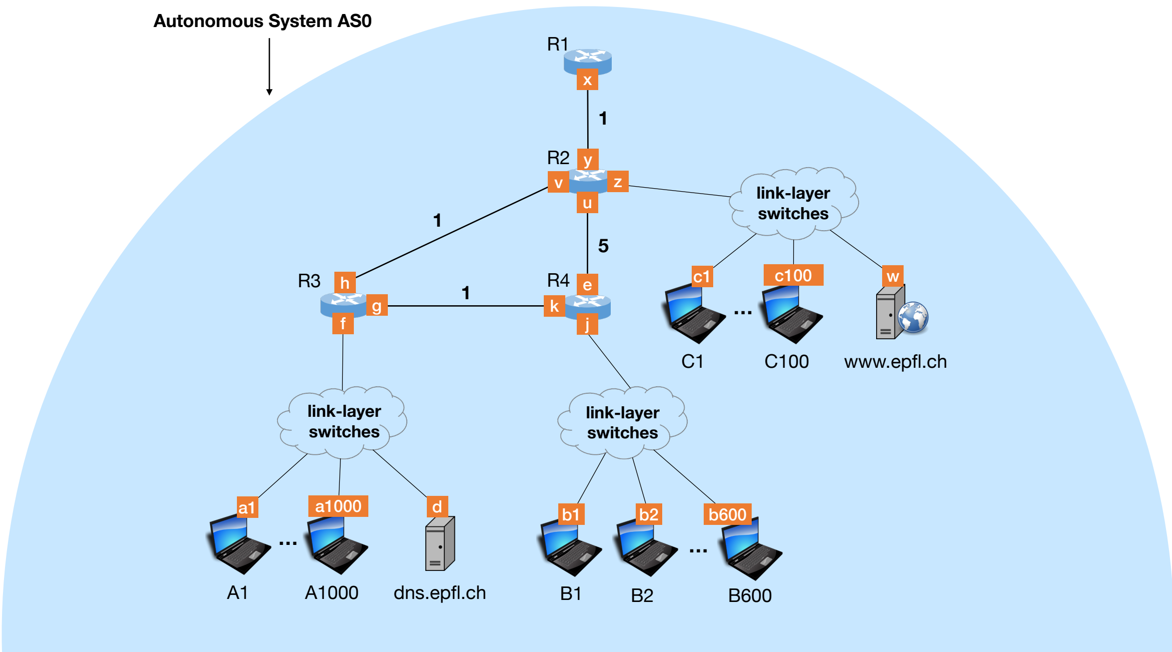

Figure 5: The Network Topology used in Problem 2.

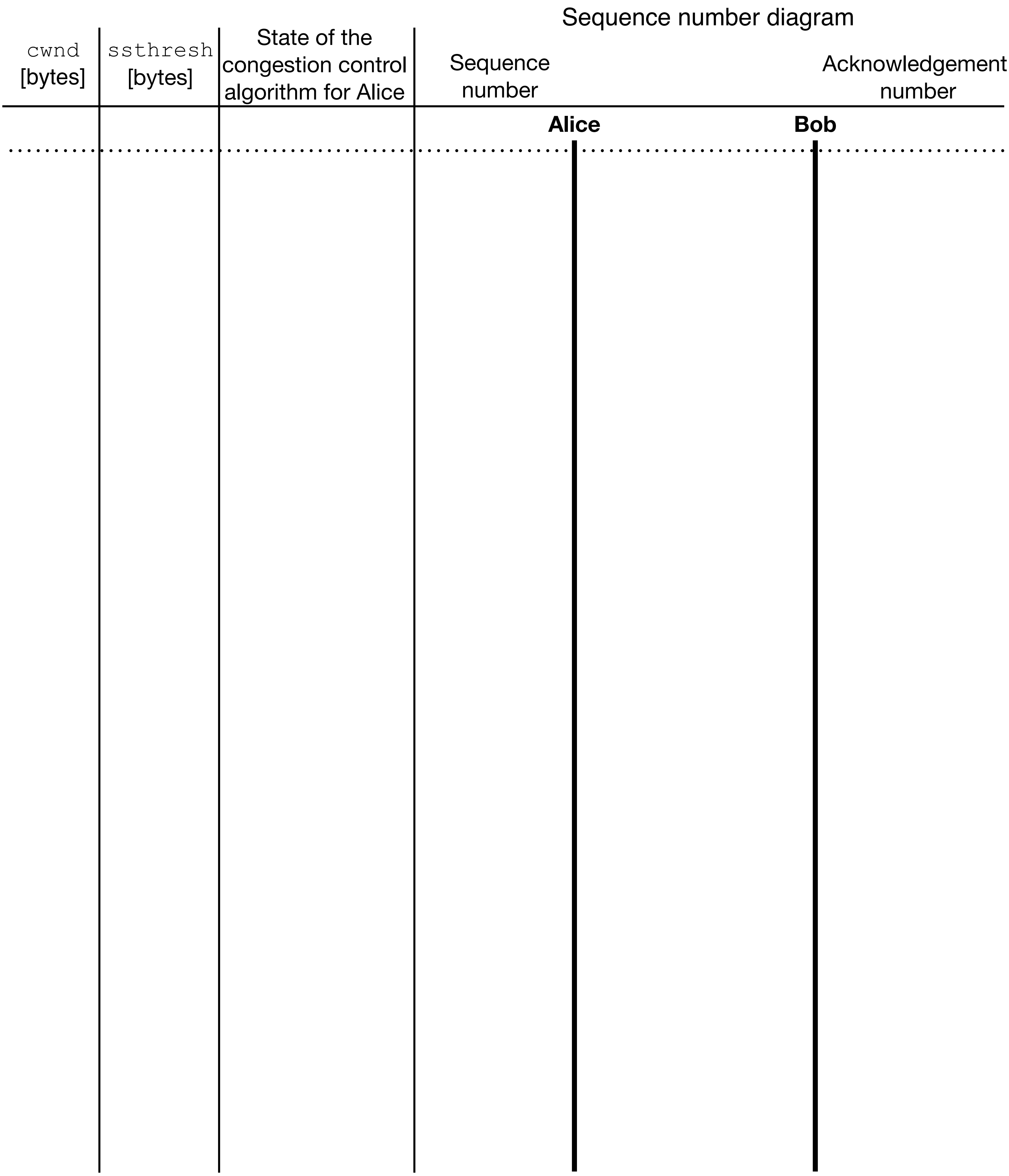

Figure 6: Copy of the sequence diagram to be completed for Problem 3, Question 1.

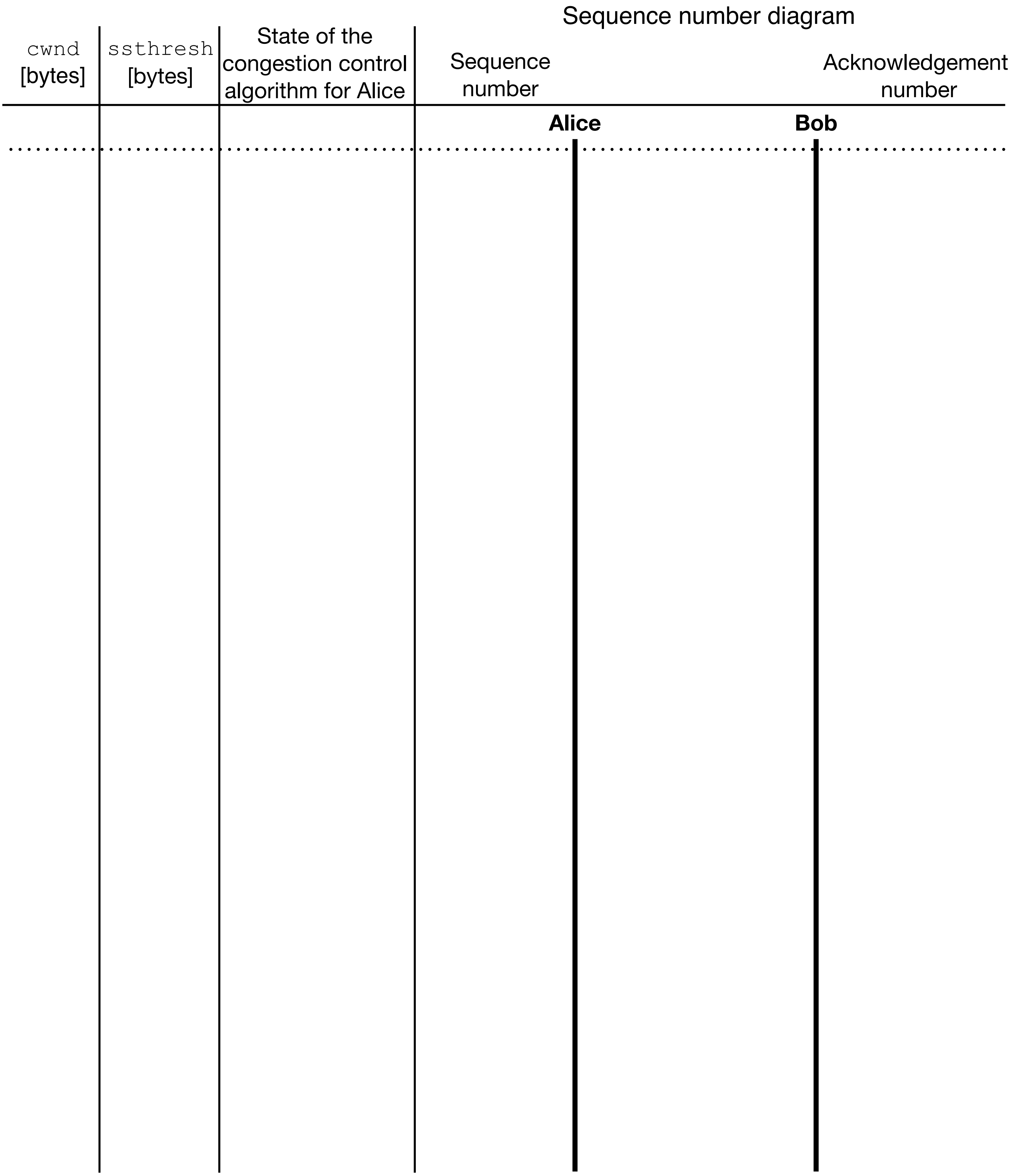

Figure 7: Copy of the sequence diagram to be completed for Problem 3, Question 2.