ETH zürich
Institut für Technische Informatik und Kommunikationsnetze

Department ITET
August 2025

Prof. Dr. Laurent Vanbever
L. Brandner, G. Fragkouli, P. Ronchetti, L. Röllin

# Exam: Communication Networks

11 August 2025, 8:30–11:00, Room HIL G 61

General remarks:

- ▷ Write **legibly** your ETH student number (legi number) below on this front page.
- ▷ **Do not write your name** or use a stamp with your name on it.
- ▷ **TRIPLE-check that your legi number is correct!**
  You will not be graded if you make a mistake when writing your number.
- ▷ Put your **legitimation card** on the top right corner of your desk. Make sure that the side containing your name and **student number** is visible.

- ▷ Check that you have received **all task sheets** (Pages **1 − 33**).
- ▷ Do **not separate** the **task sheets**.

- ▷ Write your answers directly on the task sheets.
- ▷ **All answers fit within the allocated space and often in much less.**
- ▷ If you need more space, use the three extra sheets at the **end of the exam**. Indicate the **task** in the corresponding field.

- ▷ <u>Read each task</u> **completely before you start solving it**.
- ▷ **For the best mark, it is not required to score all points.**

- ▷ Please answer either in **English or German**.
- ▷ **Write clearly** in blue or black ink (not red) using a **pen**, not a pencil.
- ▷ **Cancel** invalid parts of your solutions **clearly**.
- ▷ At the end of the exam, **place the exam face up on the top left corner** of your desk. Then collect all your belongings and **exit the room** according to the given instructions.

Special aids:

- ▷ All written materials (vocabulary books, lecture and lab scripts, exercises, etc.) are allowed.
- ▷ Using a calculator is allowed, but the use of electronic communication tools (mobile phone, computer, etc.) is strictly forbidden.

Student legi nr.:

---

**Do not write in the table below** (used by correctors only):

| Task | Points |
|---|---|
| Ethernet & IP | /32 |
| Intra-domain routing | /28 |
| Inter-domain routing | /42 |
| Reliable transport | /30 |
| Applications | /18 |
| Total | /150 |

leave blank

leave blank

## Task 1: Ethernet & IP 32 Points

### a) Warm-Up (6 Points)

**(i)** How many links remain active when running the STP (Spanning Tree Protocol) on a fully connected network with N switches? (1 Point)

**(ii)** The MAC in MAC address stands for Medium Access Control. Why is this name not really appropriate anymore for modern ethernet? (1 Point)

**(iii)** What do the forwarding entries of a layer-2 switch contain as information? (1 Point)

**(iv)** How many IP addresses are unusable per subnet and what are they used for? (1 Point)

**(v)** Why can **only** one double colon (::) be used per IPv6 address to shorten the notation? (1 Point)

**(vi)** How can the usage of VLANs create packet fragmentations? (1 Point)

## b) DHCP and ARP (12 Points)

leave blank

Consider the following network where you just connected a new host $H_1$:

- $S_1$, $S_2$, $S_3$ and $S_4$ are layer-2 switches
- $R_x$ are layer-3 routers where $x$ is the router number
- $S_1$ is connected to a DHCP server
- $H_x$ are hosts where $x$ is the host number
- layer-2 switches do not have any MACs or IPs
- $R_1$ is the default gateway for $H_1$ and $H_2$, $R_2$ is the default gateway for $H_3$
- Host $H_x$ has MAC mac-$H_x$
- DHCP server has MAC mac-dhcp
- Router $R_x$ has MAC mac-$R_x$ on the interface connected to the layer-2 switch

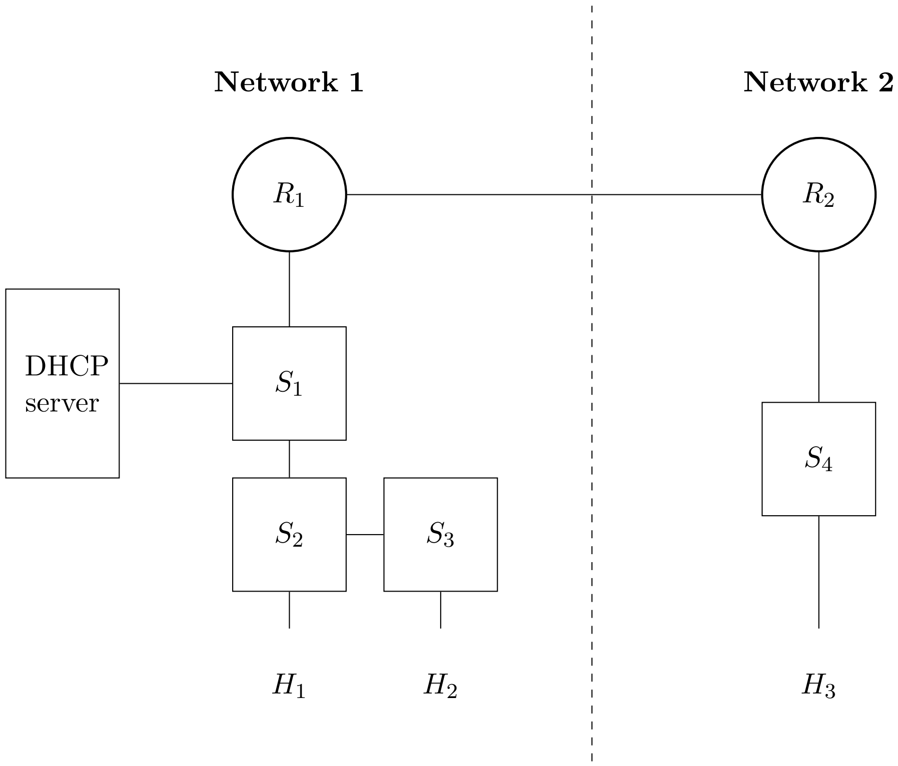



Figure 1: Two networks connected through the Internet.

**(i)** What would be the alternative to using a DHCP server to assign IP addresses?

(1 Point)

**(ii)** Host 1 ($H_1$) was only recently plugged in and has not received an IP yet but wants to send a ping to Host 2 ($H_2$). You can assume Host 1 knows the IP address of Host 2. We assume that the **DHCP server defaults to using unicast for answers**. What packets need to be exchanged **before** the first ping can be sent? Write down the packets that S2 sees in Table 1.

(4 Points)

| # | MAC | | Protocol & Purpose |
|---|---|---|---|
| | src | dst | |
| 1 | | | |
| 2 | | | |
| 3 | | | |
| 4 | | | |
| 5 | | | |

Table 1: Frames/packets seen by **S2**.

**(iii)** Host 1 now wants to communicate with Host 3 which is in a different L3 network. Why is it necessary for Host 1 to know the IP of $R_1$ which is configured as the default gateway?

(1 Point)

**(iv)** Host 1 has acquired an IP address as described above and successfully pinged both Host 2 ($H_2$) and Host 3 ($H_3$). List all the destination MACs that each switch has learned, **in the order that they were learned** in Table 2.

- We assume that all switch forwarding tables and all ARP tables were empty in the beginning. This means that router $R_1$ and $R_2$ need to do an ARP request before knowing how to reach the hosts within their L2 network.
- We assume that $R_1$ and $R_2$ know how to reach the other network.

(6 Points)

| # | **S1**<br>**dest. MAC** | **S2**<br>**dest. MAC** | **S3**<br>**dest. MAC** | **S4**<br>**dest. MAC** |
|---|---|---|---|---|
| 1 | | | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |

Table 2: MAC addresses with an entry in the forwarding tables.

**c) Network Clean Up** **(14 Points)**

leave blank

**(i)** Consider an IP router with a forwarding table composed of the 12 entries depicted on the left. Write down (on the right) an **equivalent** forwarding table by combining entries together into shorter ones such that the resulting table has the least number of entries. Your reduced forwarding table should be such that the forwarding decision made by the router for any IP packet is equivalent to the initial one. **Sort** the new forwarding table entries from **longest to shortest prefix**.

(8 Points)

| **IP prefix** | **Port** |
|---|---|
| 0.0.0.0/0 | 1 |
| 3.0.0.0/8 | 2 |
| 125.0.104.0/21 | 3 |
| 4.0.0.0/8 | 2 |
| 123.0.1.128/25 | 2 |
| 125.0.112.0/20 | 3 |
| 123.1.0.0/16 | 4 |
| 2.0.0.0/8 | 2 |
| 123.0.1.0/24 | 1 |
| 125.0.96.0/21 | 3 |
| 123.0.0.0/16 | 4 |
| 5.0.0.0/8 | 1 |

Table 3: Original entries

| **IP prefix** | **Port** |
|---|---|
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |

Table 4: Cleaned up entries.

**(ii)** In this second task we want to deactivate redundant links in our L2 network. For this the Spanning Tree Protocol (STP) is used. Assume that the network is connected as shown in Figure 2 and that the STP is configured such that Switch S1 has ID 1, Switch S2 has ID 2 and so on. Write down all the links that are **deactivated** by the STP in **alphabetical order**.

(3 Points)

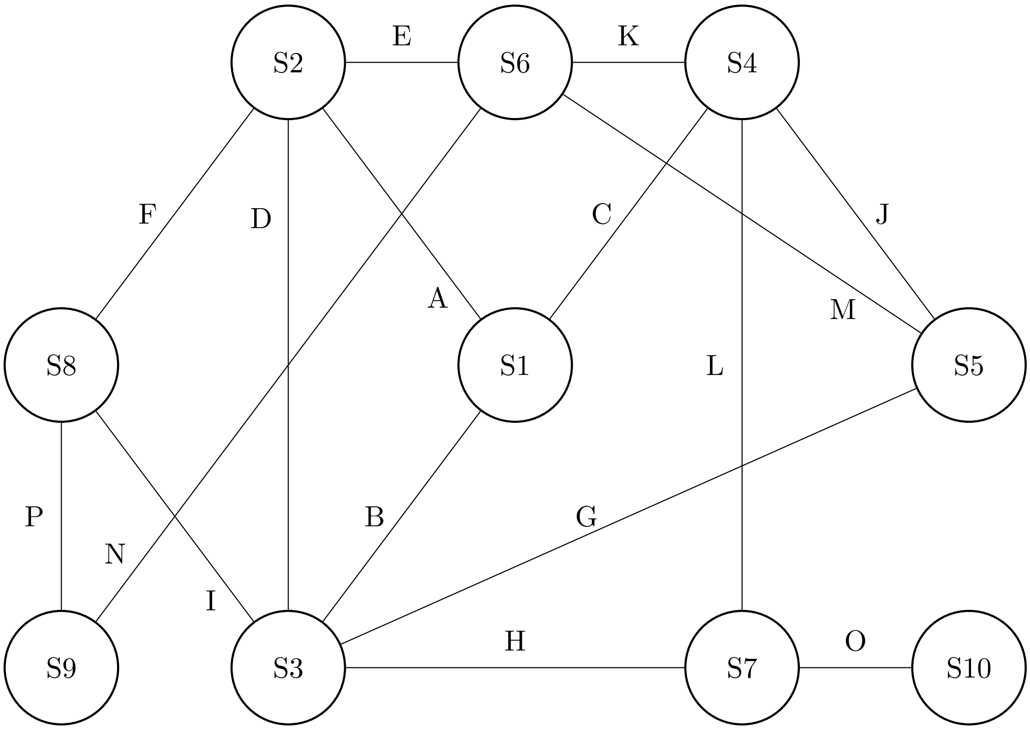



Figure 2: L2 network

**(iii)** After all the links are deactivated you find that the maximum distance from the root node to all the other nodes is not optimal. Which node should be the root node (have Switch ID 1) instead to reduce the maximum distance and what is this new distance?

(3 Points)

leave blank

leave blank

## Task 2: Intra-domain routing 28 Points

### a) Warm-Up (4 Points)

**(i)** Consider two paths, $p_1$ and $p_2$, between two routers. Each path has a bottleneck capacity of $b$. Considering that we (perfectly) load-balance a **single** flow over the two paths, what could be the reason for the total achieved throughput to be $< 2b$? (2 Points)

**(ii)** In a network with strictly positive link costs, running a link-state protocol (LSP) versus a distance-vector protocol (DVP) results in different final shortest paths between routers A and Z (e.g., path A→C→Z with LSP versus A→B→D→Z with DVP). Both protocols use the same link costs and there are no errors (e.g., in link-state advertisements, distance vectors, computations).

What could be a reason for the two protocols resulting in different final shortest paths? (2 Points)

**b) Convergence** **(7 Points)**

leave blank

Consider a network of five routers, shown in Figure 3. Each link is annotated with its cost. Link costs are bi-directional.

All routers run either a distance-vector or a link-state protocol (see below). For either of the two protocols, shortest-path ties are broken by a router selecting the path that goes through the neighbor with the smallest identifier in the alphabet.

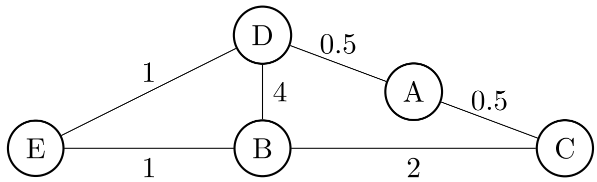



Figure 3: The network for Task 2b.

**(i)** Routers run a distance-vector protocol *with poisoned reverse*. The network has converged.

Fill in the tables below with the distance vectors (DVs) sent periodically by each of the routers A, B, C, D to E. For routers that do not send DVs to E, you must leave the corresponding DVs empty. We have already completed the first entry in the DV sent from D to E, i.e., "{dest: A, cost: 0.5}": this means that D tells E that D's shortest path to A has a cost of 0.5. (3 Points)

DV sent from **A** to E

| **dest** | **cost** |
|---|---|
| A | |
| B | |
| C | |
| D | |
| E | |

DV sent from **B** to E

| **dest** | **cost** |
|---|---|
| A | |
| B | |
| C | |
| D | |
| E | |

DV sent from **C** to E

| **dest** | **cost** |
|---|---|
| A | |
| B | |
| C | |
| D | |
| E | |

DV sent from **D** to E

| **dest** | **cost** |
|---|---|
| A | 0.5 |
| B | |
| C | |
| D | |
| E | |

**(ii)** Routers run a link-state protocol. The network has converged.

Routers A, B, C and D transit traffic for an IP destination connected to E. The network administrator wants to take down link D – E, on purpose, for maintenance reasons. To avoid transient issues, the administrator wants to move away all traffic from link D – E before taking it down and this, without creating any transient loops.

What is the minimum sequence of increased costs we must set on link D – E to ensure that no packet destined to E is dropped? Assume that whenever we set a new link cost, we wait for the network to converge before we set the link cost again or finally take the link down. Write each new link cost on a different line, in chronological order. (You may not need all the lines.) We have already added a final line where we take the link down.

(4 Points)

#1: we set the link D – E cost to ______________________________

#2: we set the link D – E cost to ______________________________

#3: we set the link D – E cost to ______________________________

#4: we set the link D – E cost to ______________________________

we take link D – E down

**c) Reverse Dijkstra with Help from Traceroute** **(9 Points)**

leave blank

Consider a network consisting of seven nodes: A, B, C, D, E, F, G. You do not know which links exist in this network. You aim to find some of those missing links and link costs using the following information:

- There is at most one link between any pair of nodes;
- Each link has a cost $> 0$, which is equal to the link propagation delay in milliseconds. Link costs are bi-directional;
- There is only one shortest path between any pair of nodes;
- Table 5 shows the *partial* output of Dijkstra's algorithm (some cells are empty) as performed in the lecture starting from node **A**. For each iteration, the table indicates the costs of the shortest paths found so far;
- Figure 4 shows the outputs of traceroutes ran from A to E, and G to B after all nodes have finished running Dijkstra and installed the shortest paths. Assume that traceroute traffic experiences only propagation delays. The IP address of node A is `a.a.a.a`, node B is `b.b.b.b`, etc.

| # | **A** | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 101 | 3 | 4 | | $\infty$ | $\infty$ |
| 2 | 0 | 101 | 3 | 4 | 2 | $\infty$ | 12 |
| 3 | 0 | 100 | 3 | 4 | 2 | $\infty$ | 12 |
| 4 | 0 | 100 | 3 | 4 | 2 | 16 | 12 |
| 5 | 0 | 100 | 3 | 4 | 2 | | 12 |
| 6 | 0 | | 3 | 4 | 2 | | 12 |
| 7 | 0 | | 3 | 4 | 2 | | 12 |

Table 5: For each iteration (1 to 7), the table shows the costs of the shortest paths found by Dijkstra's algorithm performed on node **A** toward all nodes.

```
traceroute from A (a.a.a.a) to E (e.e.e.e), 30 hops max, 60 byte packets
1  E (e.e.e.e) 4ms 4ms 4ms

traceroute from G (g.g.g.g) to B (b.b.b.b), 30 hops max, 60 byte packets
1  F (f.f.f.f) 2ms 2ms 2ms
2  B (b.b.b.b) 4ms 4ms 4ms
```

Figure 4: Traceroute outputs from A to E, and G to B, ran after all nodes have finished running Dijkstra and installed the shortest paths. Traceroute traffic experiences only propagation delays.

Using the aforementioned information:

**(i)** Fill in the empty cells in Table 5. Write your answers directly in Table 5. (3 Points)

**(ii)** Find the minimum number of missing links. Write below these links and the corresponding costs. For example, write "X – Y (5)" to indicate that there is a link between nodes X and Y of cost 5 ms. Use a different line for each link. (You may not need all the lines.) (6 Points)

link #1: ______

#2: ______

#3: ______

#4: ______

#5: ______

#6: ______

#7: ______

#8: ______

#9: ______

#10: ______

**d) Correct Shortest Paths with Incorrect Link Costs** **(8 Points)**



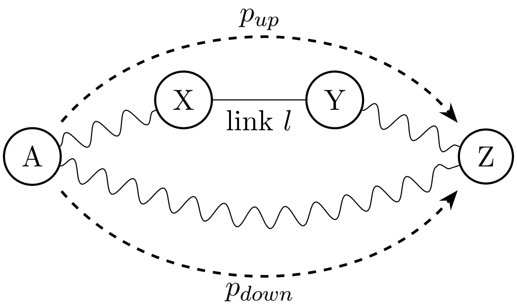



Figure 5: The network for Task 2d.

You are the operator of the network shown in Figure 5 where:

- four routers of the network (A, Z, X, Y) are shown;
- there is at most one link between any pair of routers; each link has a cost $> 0$;
- there are exactly two paths from A to Z: $p_{up}$ and $p_{down}$:
  - path $p_{up}$ passes through link $l$;
  - path $p_{down}$ does *not* pass through link $l$;
  - the zig-zag lines between routers indicate other routers/links of the network.

All routers run a link-state protocol twice:

- the first time with wrong cost $x > 0$ for link $l$. This results in router A computing:
  - total cost $c_{p_{up}}$ for path $p_{up}$;
  - total cost $c_{p_{down}}$ for path $p_{down}$;
  - path $p$ ($p_{up}$ or $p_{down}$, see below) as the only (i.e., no ties) shortest path to Z.
- the second time with correct cost $x' > 0$ for link $l$. This results in router A computing:
  - path $p'$ as the only shortest path to Z.

In both cases, routers use the correct link costs for all other links.

You want to know under what conditions $p = p'$ as, in this case, A correctly computes the shortest path to Z even when link $l$'s cost is wrong. In each of the following cases, write any extra condition on $c_{p_{up}}$ and $c_{p_{down}}$ that is required to meet goal $p = p'$. If no extra condition is required, write "Goal already met." If the goal cannot be met, write "Goal impossible."

**(i)** You also know: $x' = x - 10$ and $p = p_{up}$. What extra condition is required? (2 Points)

Condition: __________

**(ii)** You also know: $x' = x - 10$ and $p = p_{down}$. What extra condition is required? (2 Points)

Condition: __________

**(iii)** You also know: $x' = x + 10$ and $p = p_{up}$. What extra condition is required? (2 Points)

Condition: __________

**(iv)** You also know: $x' = x + 10$ and $p = p_{down}$. What extra condition is required? (2 Points)

Condition: __________

leave blank

leave blank

## Task 3: Inter-domain routing **42 Points**

### a) Vacation Trouble (9 Points)

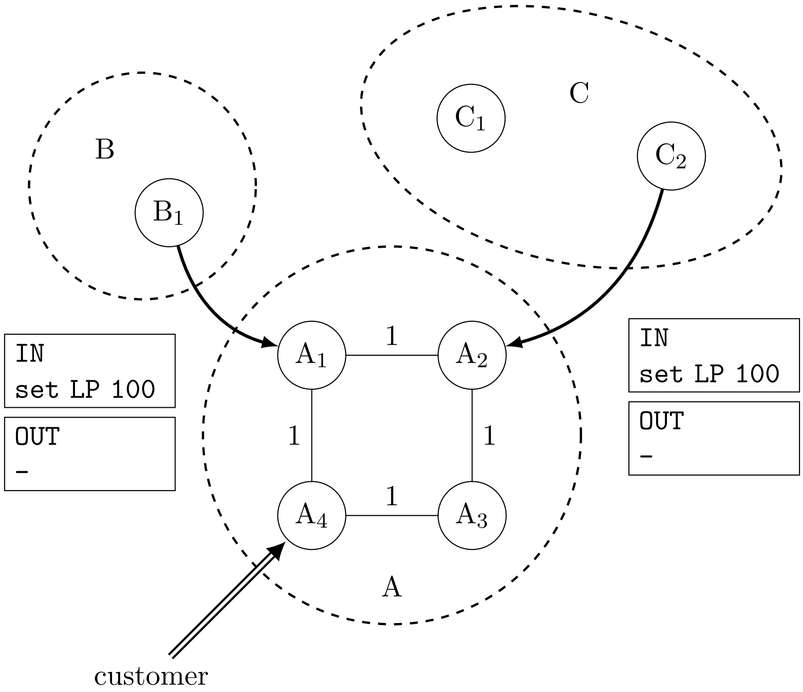



Figure 6: A BGP network comprised of 3 Autonomous Systems

You have recently started working as a network engineer for a company that manages AS $A$. AS $A$ has 4 internal routers, interconnected by 4 internal links (each with an IGP weight of 1) as shown in Figure 6. The internal routers form an iBGP **full-mesh** (not shown in the figure). AS $A$ is also connected to two neighboring ASes $B$ and $C$ via eBGP sessions.

$B$ and $C$ are provider ASes; they both advertise the same prefix $p$ to AS $A$. Their advertisements go through the route maps pictured in the figure, for now, they both set the local preference (`LP`) to `100` for incoming advertisements.

In this task, all changes introduced in each subtask will be implemented incrementally.

**(i)** For each internal router, indicate the path (sequence of routers) it selects to reach $p$ (2 Points)

$A_1 \rightarrow$ ____________________

$A_2 \rightarrow$ ____________________

$A_3 \rightarrow$ ____________________

$A_4 \rightarrow$ ____________________

**(ii)** A customer AS, connected to AS $A$ at router $A_4$, is complaining that AS $B$ is slowing their outgoing traffic bound for $p$. Your manager is on vacation and you don't have access to the route maps, you can only configure the IGP weights. How do you ensure that your customer's traffic bound for $p$ is sent through $C$ rather than $B$? (1 Point)

**(iii)** The operators of $B$ have noticed that there is less traffic flowing their way. How could they trick your AS into sending them more traffic again? (1 Point)

**(iv)** Your manager has finally replied to your emails and you can now modify the routemaps. How do you ensure that your customer's traffic bound for $p$ is sent through $C$ rather than $B$? (1 Point)

**(v)** You tear down the $B_1 - A_1$ link. In case the $C_2 - A_2$ link fails, you have added a new eBGP session between the router $C_1$ and $A_1$ as a backup. Set up the `IN` route map on $C_1 - A_1$ such that your traffic bound for `p` is sent through $C_2$ as long as the link is up, but switched to $C_1$ in case of a failure. (1 Point)

**(vi)** You now set out to steer the return traffic entering from $C$ to your customer through $C_2$. Describe two strategies to do that and how you would configure the route-maps on $C_1 - A_1$ and $C_2 - A_2$, if need be. Also indicate whether each strategy guarantees the return traffic will enter via $C_2$ or just influences $C$ in picking it. (3 Points)

Strategy:

Strategy:

**b) Relationship Investigation** **(16 Points)**

leave blank

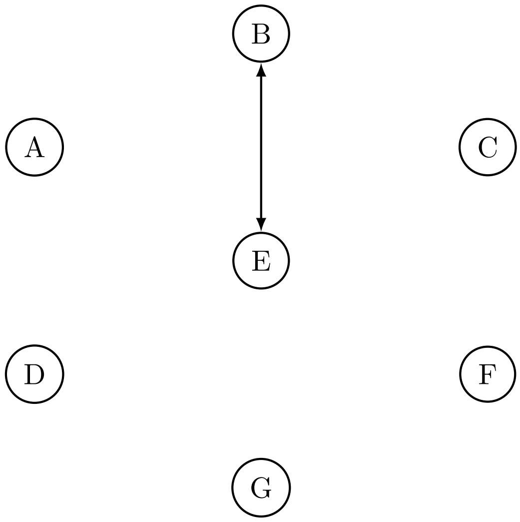



Figure 7: A small Internet topology made up of 7 ASes

Consider a small Internet topology made up of 7 Autonomous Systems (ASes). Each AS in this Internet consists of a single BGP router and applies the default selection and exportation BGP policies based on their customers, peers and providers.

The only connection you are aware of is the one between ASes $B$ and $E$. You also know that theirs is a peer/peer relationship.

**(i)** You are given the **full** routing tables of the ASes $D$ and $C$ in Figure 8. AS $D$, for example, knows two paths to $A$: one to $A$ directly, and $B$ $A$. Draw a connection between all ASes that are directly connected using Figure 7 above. (2 Points)

D

| Dst | Path |
|---|---|
| A | A; BA |
| B | B |
| C | BEC |
| E | BE |
| G | AG; BEG |

C

| Dst | Path |
|---|---|
| B | EB |
| D | EBD |
| E | E |
| F | F |
| G | FG; EG |

Figure 8: Routing Tables of ASes $D$ and $C$.

**(ii)** While it is interesting to know the direct connections between the ASes, it is also very interesting to know their business relationships. Using the information in Figure 8, infer the type of certain connections between the ASes.

Start by identifying the type of relationships of the edges $E-C$, $B-C$ and $E-G$. Complete the table below by filling in the type of relationship (none, cust/prov, prov/cust or peer/peer) and provide a short justification.

Pay attention to correctly specify the relationships. For example, $X$-$Y$, cust/prov means that $X$ is the customer and $Y$ the provider; none means that there is no edge between the two ASes. (4 Points)

| Link | Relationship | Justification |
|---|---|---|
| $E-C$ | | |
| $B-C$ | | |
| $E-G$ | | |

**(iii)** Building on the answer you gave in the previous question, continue identifying the types of relationships of these edges:

(8 Points)

| Link | Relationship | Justification |
|---|---|---|
| $B-D$ | | |
| $A-B$ | | |
| $A-D$ | | |

**(iv)** In the real world, is there a guarantee that the traffic will follow the routes as presented in Figure 8? Why?

(2 Points)

**c) Bouncing Routes** **(17 Points)**

leave blank

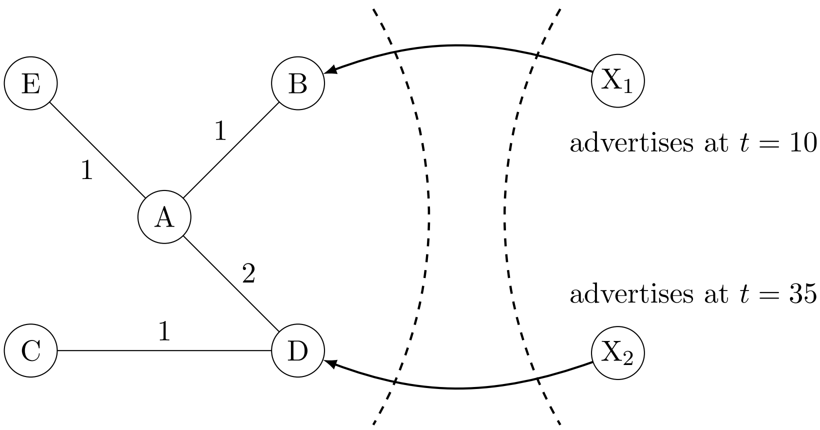



Figure 9: Your small AS and your provider AS

After successfully completing the CommNet exam, you are immediately hired by a small Internet Service Provider (ISP) that manages a very small AS. The AS is depicted in Figure 9, along with the physical connections between the internal routers. Every internal connection incurs a **delay** of 10 [ms] and the internal links have IGP costs associated to them as pictured.
Your AS has five internal routers, two of which border with your only neighboring AS, which is your provider. Your provider only advertises a single prefix.

**(i)** Your iBGP topology is, at first, designed as a full-mesh. How many iBGP sessions do we have? If we had $N$ routers, how would the number of sessions scale wrt. $N$? (1 Point)

**(ii)** At $t = 10$, router $B$ receives an advertisement from $X_1$. At $t = 35$, router $D$ receives the **same** advertisement from $X_2$. Fill out the missing information in Table 6 below. For every exchanged message, specify the time the message is **delivered**, the router that sent the BGP message, the router that received it as well as the *effect* of the message on the routing table of the receiver. By *effect* we mean how the **egress** changes; it could be none if no change occurs. Computation is assumed to be instantaneous. (4 Points)

| $t_d$ [ms] | Source | Destination | Egress |
|---|---|---|---|
| 10 | $X_1$ | $B$ | $B \rightarrow$ self |
| 20 | $B$ | | |
| | | | |
| | | | |
| 35 | $X_2$ | $D$ | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

Table 6: Sequence of BGP messages. The first row states that at 10ms a message from $X_1$ to $B$ causes $B$ to select itself as the egress

**(iii)** The CEO of your company has big plans for your small ISP. She wants to make sure that the network can scale to a large number of nodes, but she is worried about the total number of iBGP sessions. Your routers are cheap and they won't be able to handle too many of them!

You are thinking of introducing special *reflector* routers, which are allowed to share routes they have **selected** internally. You come up with the following rules

0. A non-reflector router forwards advertisements normally
1. If a *reflector* selects a route learned from another *reflector*, they share it *only* with certain designated routers (their *clients*).
2. If a *reflector* selects a route learned from its *client*, they reflect it on *all* their sessions (both other *clients* and *reflectors*).
3. If a *reflector* selects a route learned from an *external* BGP (eBGP) peer, they reflect it on *all* their sessions.

As shown in Figure 10, we select the routers $A$ and $D$ to be *reflector* routers.
Router $C$ is $D$'s *client*. Routers $B$ and $E$ are $A$'s *clients*.

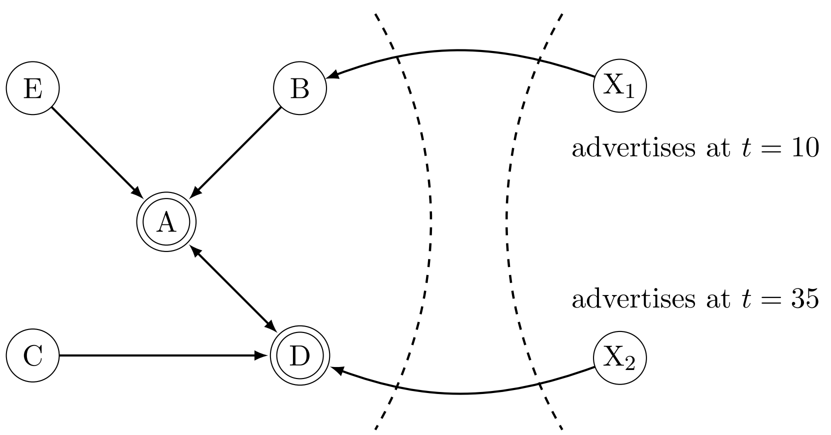



Figure 10: Same physical links as in Figure 9, now with special iBGP sessions

How many iBGP sessions do we have now? (1 Point)

---

**(iv)** How many iBGP sessions would we have if we had N nodes and 2 *reflector* nodes (each *client* node has a single *reflector*)? (2 Points)

---

---

**(v)** Consider what would happen if you had this iBGP topology instead of a full mesh. The eBGP sessions are the same as before.

The *Rule* column of Table 7 indicates which of the rules above caused the message to be sent. The rest are the same as those in Table 6. Fill out the missing information in the table below. (7 Points)

| $t_d$ [ms] | Source | Destination | Egress | Rule |
|---|---|---|---|---|
| 10 | $X_1$ | $B$ | $B \to$ self | - |
| 20 | $B$ | | | 0 |
| | | | | 2 |
| | | | | |
| 35 | $X_2$ | $D$ | | - |
| | | | | |
| | | | | |
| | | | | |

Table 7: Sequence of BGP messages

**(vi)** What kind of improvement do we see? There are two positive aspects that should be apparent from your answer to the previous questions. (2 Points)

1: ________________________________________

2: ________________________________________

leave blank

leave blank

## Task 4: Reliable transport — 30 Points

### a) Warm-Up — (6 Points)

**(i)** Assume that TCP always uses the initial sequence number 1234. Give an advantage and disadvantage of this design. (1 Point)

Advantage: ____________________________________

____________________________________

Disadvantage: ____________________________________

____________________________________

**(ii)** Is it possible for an application running over UDP to send a reliable byte stream similar to TCP? Briefly explain what mechanisms the application would need to implement. (1 Point)

____________________________________

____________________________________

____________________________________

**(iii)** UDP is said to give finer-grained control to the application compared to TCP. Explain why the application has less control over the data and the time it is sent when using TCP. (2 Points)

Data: ____________________________________

____________________________________

Time: ____________________________________

____________________________________

**(iv)** Provide an example of a well-known port along with its application. Is it possible to deploy this application on an ephemeral port too? Justify your answer. (2 Points)

Port: __________________ Application: __________________

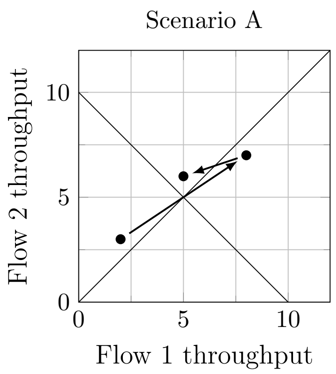
Explanation: ____________________________________

____________________________________

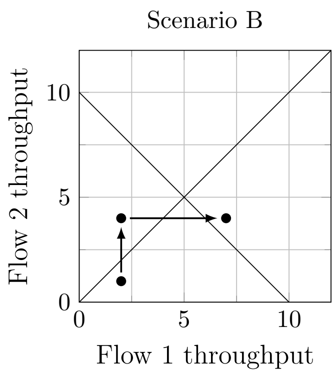
**b) Efficiency and Fairness** **(11 Points)**

leave blank

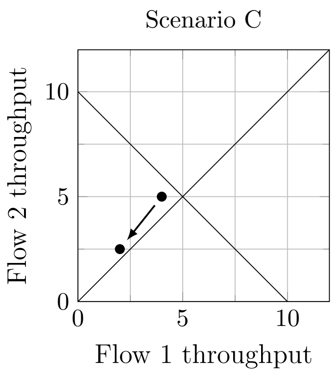
**(i)** Consider two independent flows sharing a common link with a capacity of $C = 10$ units, each running a congestion control (CC) algorithm with constant parameters $(\alpha_1, \beta_1)$ and $(\alpha_2, \beta_2)$. At random time points, we observe both flows' throughput and produce scatter graphs to evaluate their interaction. Below are multiple such system trajectory plots. For each plot, indicate whether the shown trajectory can be possibly observed for the independent flows with **AIMD** CC algorithms. *Justify your answers.* (4 Points)



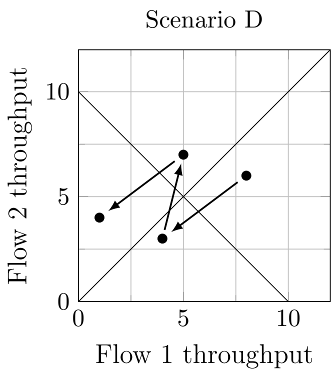

☐ Possible with AIMD
☐ Not possible with AIMD




☐ Possible with AIMD
☐ Not possible with AIMD




☐ Possible with AIMD
☐ Not possible with AIMD




☐ Possible with AIMD
☐ Not possible with AIMD

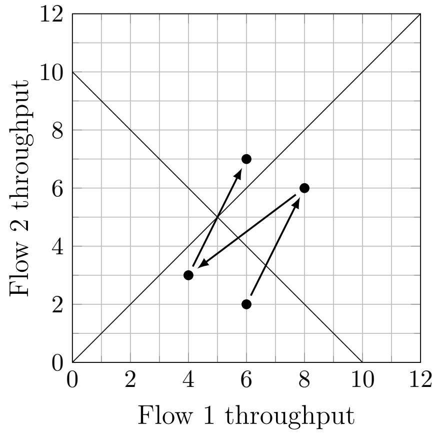
Justification for A: __________________________________________________

Justification for B: __________________________________________________

Justification for C: __________________________________________________

Justification for D: __________________________________________________

Now consider the following diagram, which shows the throughput of two independent flows running CC algorithms sharing a link with capacity of 10 units.




**(ii)** Is the increasing behaviour of each CC algorithm additive or multiplicative? *Justify your answer.* (1 Point)

**(iii)** Indicate the next point $(x_1, x_2)$ in the system's trajectory, after $(6, 7)$, in the cases that the congestion control algorithm's decreasing behaviour is 1) additive, and 2) multiplicative for both flows. Will the system converge to a fair allocation? *Justify your answer.* (6 Points)

Next point assuming additive increase:

Will the system converge to a fair state?

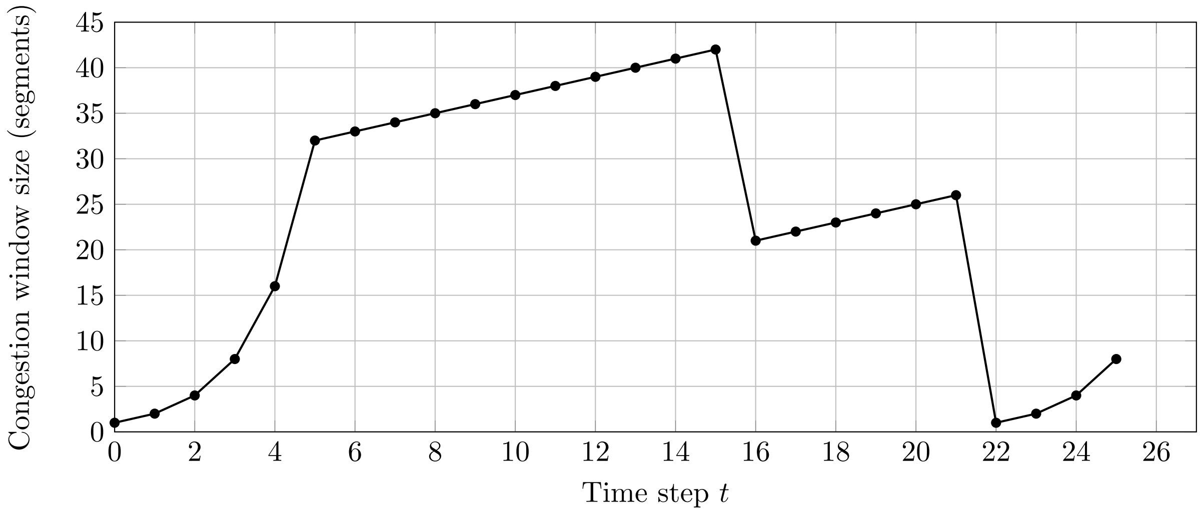
Next point assuming multiplicative increase:

Will the system converge to a fair state?

**c) TCP Congestion Control** **(13 Points)**

Consider the following diagram depicting the evolution of the congestion window size (`cwnd`) for a flow (**Flow 1**) using TCP's congestion control algorithm.

leave blank




Based on the diagram, answer the following tasks:

**(i)** Indicate the *sequence* of values that the Slow Start threshold (`ssthresh`) takes, considering the value at time $t = 0$ equals 32. (2 Points)

32, ______

**(ii)** What event caused the `cwnd` to drop at timestep $t = 16$? (1 Point)

Event: ______

Justification: ______

**(iii)** Explain whether the event at timestep 22 necessitates that the network dropped a packet. (1 Point)

**(iv)** During which timestep is the 40th segment sent? (1 Point)

**(v)** Now, consider a second flow, **Flow 2**, which runs the same CC algorithm as Flow 1, with the same parameters. However, Flow 2 experiences **double the RTT** (Round Trip Time) of Flow 1. Given that

- The Round Trip Time of **Flow 1** is constant at $\texttt{RTT}_1$=100ms
- Flow 1's congestion window evolution is as in the previous figure
- **Only one** congestion event is detected for Flow 2 in the examined timeframe, and its occurence can be inferred from the filled cells in the table below

Fill in the empty cells in the table below with the correct values for the congestion windows, slow start thresholds, and timestamps. Does the TCP congestion control algorithm achieve fair bandwidth utilization? Briefly explain why or why not. (8 Points)

(*Notice that the documented timestamps are not uniformly spaced!*)

| **time (s)** | 0.0 | 0.4 | 1.0 | | | | |
|---|---|---|---|---|---|---|---|
| **Flow 1 cwnd** | 1 | 16 | 37 | | | | |
| **Flow 2 cwnd** | 1 | | | 34 | 17 | 18 | 19 |
| **Flow 2 ssthresh** | 32 | | | | | | |

Does CC achieve fair allocation? ______________________________

______________________________

leave blank

leave blank

## Task 5: Applications **18 Points**

### a) Warm-Up **(6 Points)**

Consider the following URL:

`https://a.b.c.d.internet-routing-rocks.ch`

**(i)** What is the **minimum** number of levels of the DNS hierarchy that could be involved to resolve this domain name by your local resolver, assuming iterative resolution and empty caches? Briefly explain. (2 Points)

**(ii)** What is the **maximum** number of levels of the DNS hierarchy that could be involved to resolve this domain name by your local resolver, assuming iterative resolution and empty caches? Briefly explain. (2 Points)

**(iii)** Suppose `a.b.c.d.internet-routing-rocks.ch` is actually a CNAME for `nsg.ethz.ch`.

How does the presence of this CNAME affect the DNS resolution process? List the additional queries that your local resolver would need to perform upon receiving the CNAME for `a.b.c.d.internet-routing-rocks.ch`, assuming iterative resolution and assuming your local resolver has cached the records while resolving `a.b.c.d.internet-routing-rocks.ch`. For each query, indicate: the query type, queried name, and the server being queried (describe the server in English, e.g., "the DNS server responsible for a.b.c"). (2 Points)

**b) DNS operations** **(6 Points)**

leave blank

You operate the (authoritative) DNS zone for `internet-routing-rocks.ch` with the following entries:

| **Name** | **Type** | **Data** | **TTL** |
|---|---|---|---|
| `a.b.c.d.internet-routing-rocks.ch.` | CNAME | `nsg.ethz.ch.` | 600 s |
| `nsg.ethz.ch.` | A | `101.100.99.98` | 120 s |

Here is a timeline of events (all times in seconds) leading to DNS queries:

| $t$ | **Event** |
|---|---|
| 0 | Local resolver $\mathbf{R_1}$ receives its *first* query for `a.b.c.d.internet-routing-rocks.ch`. |
| 60 | Local resolver $\mathbf{R_2}$ receives its *first* query for `a.b.c.d.internet-routing-rocks.ch`. |
| 90 | A-record changed to `1.2.3.4`; TTL unchanged. |
| 150 | Clients behind $\mathbf{R_1}$ and $\mathbf{R_2}$ query `a.b.c.d.internet-routing-rocks.ch`. |
| 250 | Clients behind $\mathbf{R_1}$ and $\mathbf{R_2}$ query `a.b.c.d.internet-routing-rocks.ch`. |
| 601 | Clients behind $\mathbf{R_1}$ and $\mathbf{R_2}$ query `a.b.c.d.internet-routing-rocks.ch`. |

**(i) Resolver caches.** For each of the six lookups triggered by each of the last three events in the above table (three to each of $\mathbf{R_1}$ and $\mathbf{R_2}$), write in the table below the value that the resolver can deliver *directly from its own cache*: `101.100.99.98`, `1.2.3.4`, or *MISS* (cache lacks valid entry). (3 Points)

| **Resolver** | $t = 150\,\text{s}$ | $t = 250\,\text{s}$ | $t = 601\,\text{s}$ |
|---|---|---|---|
| $\mathbf{R_1}$ | | | |
| $\mathbf{R_2}$ | | | |

**(ii) Resolver queries.** For each of the six lookups above, list which name(s) the resolver issues request for, if any, and/or which name(s) is returned from the cache. (3 Points)

150 s, $R_1$: ____________________

150 s, $R_2$: ____________________

250 s, $R_1$: ____________________

250 s, $R_2$: ____________________

601 s, $R_1$: ____________________

601 s, $R_2$: ____________________

**c) ETH–*flix*** **(6 Points)**

leave blank

You are part of a group of ETHZ students who want to create a new video streaming service: "ETH–*flix*". After following the lecture, you feel inspired by Netflix's infrastructure and would like to replicate some of it, albeit with a smaller budget.

You managed to secure the funding for deploying three Points-of-Presence (PoPs), with two servers each. All the servers can accommodate the entire video catalogue and are equivalent, except maybe in terms of location. For each PoP, the two servers are in the same location.

| **PoP** | **Server #1** | **Server #2** | **IPv4 prefix** |
|---|---|---|---|
| Zürich | `zrh01.ethflix.ch` | `zrh02.ethflix.ch` | `100.0.0.0/24` |
| Basel | `bsl01.ethflix.ch` | `bsl02.ethflix.ch` | `100.0.1.0/24` |
| Lugano | `lug01.ethflix.ch` | `lug02.ethflix.ch` | `100.0.2.0/24` |

**(i)** Explain how you intend to redirect clients to the various PoPs so that clients fetch content from the closest servers available. Also explain how you could load balance the incoming requests amongst the servers within a PoP. (3 Points)

How do you map clients to the closest PoP: ____________________

How do you load balance incoming requests within a PoP: ____________________

**(ii)** Explain how you could enable a fast failover between servers of the same PoP should one of them fail. The failover must avoid any connection drops on the user's side. Explain how your solution addresses these requirements. (3 Points)

## Extra Sheet 1

In case you need more space, use the following pages. Make sure to always indicate the task to which the answer belongs (e.g., *3 d) (ii)*).

Task: ______________

Task: ______________

**Extra Sheet 2**

Task: ________________

Task: ________________

**Extra Sheet 3**

Task: ______________

Task: ______________