Chair of Network Architectures and Services
Department of Informatics
Technical University of Munich

Esolution
Place student sticker here

Note:

- During the attendance check a sticker containing a unique code will be put on this exam.
- This code contains a unique number that associates this exam with your registration number.
- This number is printed both next to the code and to the signature field in the attendance check list.

Advanced Computer Networking

Exam:	IN2097 / Endterm	**Date:**	Monday 28th February, 2022
Examiner:	Prof. Dr.-Ing. Georg Carle	**Time:**	11:30 – 12:45

Working instructions

- This exam consists of **14 pages** with a total of **5 problems**.
  Please make sure now that you received a complete copy of the exam.
- The total amount of achievable credits in this exam is 75 credits.
- Detaching pages from the exam is prohibited.
- Allowed resources:
  - one **analog dictionary** English ↔ native language without annotations
  - the **provided cheatsheet** without annotations (print or digital copy)
- Subproblems marked by * can be solved without results of previous subproblems.
- **Answers are only accepted if the solution approach is documented.** Give a reason for each answer unless explicitly stated otherwise in the respective subproblem.
- Do not write with red or green colors nor use pencils.
- Code of conduct:
  - I participate without the help of others and only use the allowed resources.
  - I do not share, discuss, or exchange any information related to the exam with anybody.
  - I feel in good health and I am able to participate in the exam.
  - I understood the examination policy, agree to the video supervision, and adhere to this process.

Problem 1 Quiz (17.5 credits)

The following questions cover multiple topics and can be solved independently of each other.

0 1

a)* How does ZMap close a connection on the server even though it is stateless?

- Raw sockets are used to send SYN packets
- Received SYN/ACKs are also seen by the kernel which sends a RST

0 1

b)* What is the major difference between a ZMap UDP and TCP scan?

- TCP: ZMap relies on the 3-way handshake. A server replies with a SYN/ACK if the port is open independent of the application layer protocol.
- UDP: No handshake is available. The application layer protocol needs to answer → meaningful payload is required.

0 1

c)* A server has 723 (virtual) network interface cards (NICs). You want to assign a unique ID to each NIC. What is the minimum number of bits needed to encode the IDs?

- $2^9 = 512 < 723 < 1024 = 2^{10}$

0 1

d)* Name two disadvantages if network interface cards (NICs) or switches have fixed transceivers.

- Entire NIC must be replaced in case the physical medium is changed (e.g., fiber to copper)
- Entire NIC must be replaced if the transceiver is defect, e.g., laser/LED burns out.
- Defective ports in switches cannot be replaced easily

0 1 2

e)* How does TCP BBR use windowed filters to estimate the bandwidth-delay product (BDP)?

BBR uses a min filter on the measured RTT samples.
BBR uses a max filter on the measured bandwidth samples.

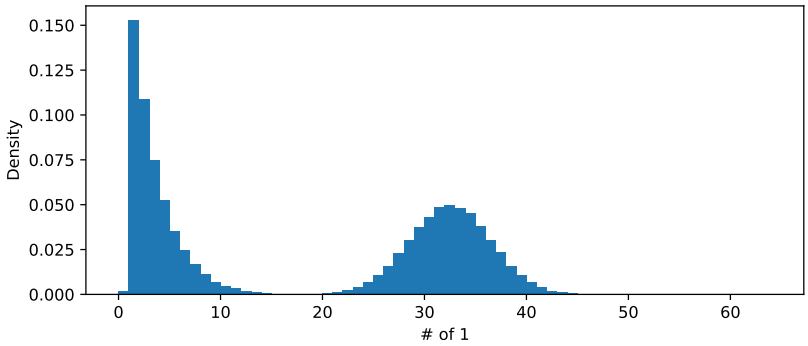

Figure 1.1: Distribution of bits set to one in the host part of a set of IPv6 addresses.

f)* Given the distribution of bits set to one in the host part of a set of IPv6 addresses shown in Figure 1.1, which device types do you expect to be represented by the addresses and why?

0
1

- The distribution combines a peak around 1-5 but also a normal distribution.
- The dataset combines devices with statically assigned addresses (e.g., routers) and devices with random host parts (privacy extension)

g)* A researcher wants to classify data packets based on their time of origin and builds a machine learning (ML) model to perform this task. Name one metric which the researcher can use to check the validity of the ML model. How can the researcher determine when the ML model is optimal?

0
1

- One out of: Accuracy, Precision, F1-score.
- In the optimal case, all these metric = 1.0

h)* Shortly explain the three different types of DNS resolvers and the typical place you can find them as discussed in the lecture.

0
1
2
3

Stub resolver Is the resolver on the local machine which converts system calls to DNS queries.

Forwarder A resolver which forwards received DNS queries to an other recursive resolver. Typically, this is the resolver on your home router (CPE)

Recursive resolver Performs the actual DNS resolution. Examples are Quad 9 (9.9.9.9) or the Google public resolver (8.8.8.8). Are usually public reachable addresses in the Internet

0
1

i)* What does DNS ECS stand for and what problem does it solve?

ECS stands for EDNS Client Subnet and it is an EDNS0 extension
The IP address (and thus the location) of the requester is not known to the authoritative nameserver if a public resolver is used. In order to support geo based load balancing on the authoritative name server DNS ECS was introduced

0
1

j)* Name two of the main components in DNS ECS (except the IP address family).

- Source prefix length (Number of relevant bits in the IP address set by the sender)
- Scope prefix length (Number of bits used in the response)
- IP address

0
1

k)* Explain the basic principle of HTTP-based load balancing.

Front-end server redirects user to back-end server using HTTP status codes

0
1

l)* Explain the basic principle of anycast-based load balancing.

Assign multiple content servers the same IP address, BGP finds best content server.

0
1

m)* Assume you operate a Content Delivery Network where the number of available cache servers changes frequently. You assign users to cache servers based on the following hash function: $u \bmod S$, where u is the user and S is the number of servers. Argue whether this is a good approach or not.

Not a good approach since majority of users are remapped each time the number of servers changes.

Problem 2 Wireshark (17 credits)

According to the OSI model network protocols are distributed to seven different layers each containing several protocols. In this problem, a frame is analyzed, referring to the involved protocols. You are given a hexdump of an Ethernet frame, starting with the Ethernet header. For simplicity, we only show the first 80 B of the frame, the omitted parts are not relevant for this problem.

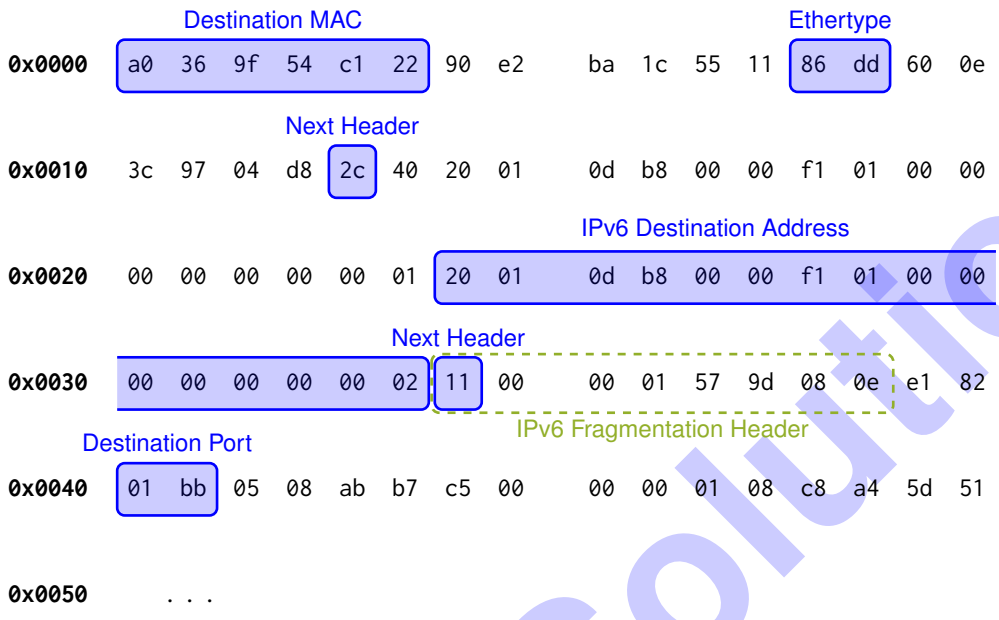

Figure 2.1: Hexdump of an Ethernet frame cut off after 80 B

For this problem you must indicate the location of the bytes in the hexdump in Figure 2.1. You can **either** mark the corresponding bytes directly in the figure **or** list the locations of the corresponding bytes using **[. . .]**. Example: the **three** bytes from position 0 to 2 can be written as `[0, 2] = 0xa0369f`.

a)* Mark the **destination MAC** address in the hexdump in Figure 2.1 **and** note the address in the common format in the solutionbox.

0 1

Destination MAC: `[0, 5]= 0xa0369f54c122`
`a0:36:9f:54:c1:22`

b)* Identify the used layer 3 protocol. Mark the corresponding bytes in the hexdump and list the protocol name.

0 1

Ethertype: `[12, 13]= 0x86dd`
`IPv6`

c) Mark the bytes, which are part of the layer 3 **destination address**. Note the address in the shortest, common notation.

0 1 2

IPv6 Destination Address: `[38, 53]= 0x20010db80000f101000000000000002`
`2001:db8:0:f101::2`

0 1 2

d) Identify the used layer 4 protocol. Mark the corresponding bytes in the hexdump and name the protocol name.

Next Header: `[20]= 0x2c` → Fragmentation Header
Next Header: `[54]= 0x11` → UDP

0 1 2 3

e) Make an educated guess for the used protocols above of TCP / UDP in the hexdump. Mark corresponding bytes in the hexdump, list the protocol names, and give a brief reason for your answer.

UDP has destination port 443 (`[64, 65]= 0x01bb`)
Probably HTTPS is used on top of QUIC, which also includes TLS.

The following problems can be answered without the results of the previous subproblems.

0 1 2 3

f)* Name **and** briefly explain two important goals (other than fixing head-of-line blocking) of QUIC, which target to fix problems with TCP / TLS.

Decrease handshake delay: merge transport layer and TLS handshake
Faster development cycles: implemented in user space not in the kernel
IP mobility: using connection IDs to separate the QUIC connection from underlying addresses

0 1

g)* What is the purpose of the QUIC spin bit?

It allows passive RTT measurements

0 1 2 3

h)* QUIC claims that it fixes the head-of-line (HoL) blocking problem of TCP. For this multiple streams are multiplexed in the same connection. In the following, three different scheduling approaches to multiplex k streams are explained:

1. Send first all data of one stream, then continue with the next stream.
2. In each QUIC packet, $\frac{1}{k}$ of the payload are from each stream.
3. Send data from one stream in a single QUIC packet, then fill the next packet with the next stream's data.

For each scheduling approach, assess if it fixes HoL blocking and briefly explain your answer.

1. No HoL blocking. Streams are not sent interleaved at all.
2. HoL blocking can happen with every packet.
3. No HoL blocking, only data of one stream is affected by a lost packet.

Problem 3 AS Relations and BGP (14 credits)

This problem investigates the autonomous system (AS) relationships in a given network and their impact on routing and traffic. $\rightarrow$ represents a customer $\rightarrow$ provider relationship, while $\leftrightarrow$ represents a peering relationship. Dashed lines are unknown policies, which need to be evaluated in Subproblem h). All ASes apply standard routing behavior. Furthermore, the following policies are applied:

- For routes with the same prefix, the AS selects the most cost-efficient route.
- For routes with the same prefix and with an equal traffic cost, the shorter route is selected.

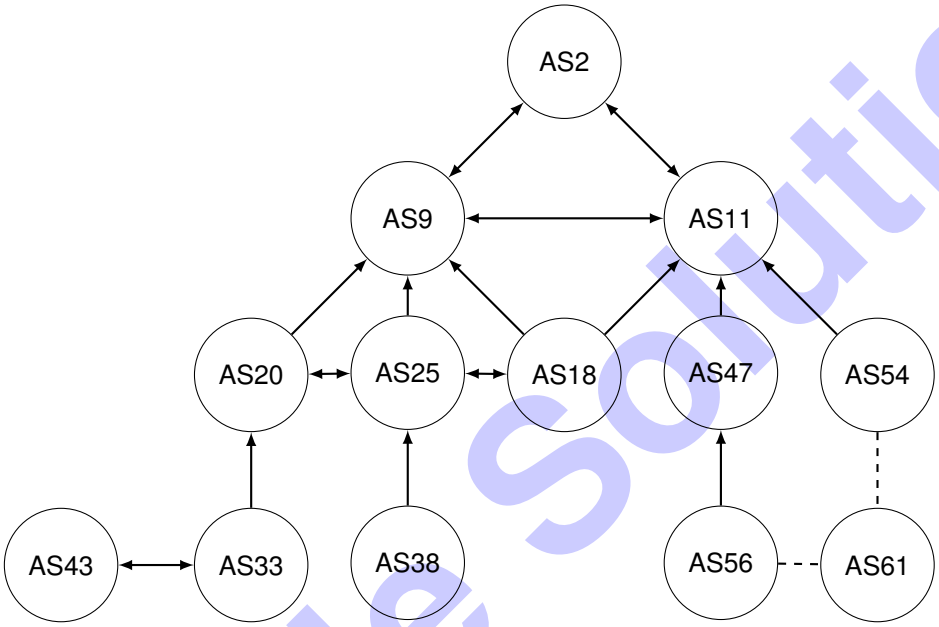

Figure 3.1: AS Network

a)* If one removes AS38 and AS43, is the first step of the k-core algorithm finished?

0 1

- Nodes of degree k have to be removed recursively. (Recursively is not required but a simple explanation)
- During the first step, AS33 has to be removed as well.

b)* Of which degree is the core of the network?

0 1

- The core is 1. During the step with k=2 all nodes would be removed.

c)* In routing, what does FIB and RIB stand for and what is the difference?

0 1 2

- FIB=Forwarding Information Base, RIB=Routing Information Base
- The RIB contains all information a router has gathered from its neighbors.
- The FIB contains a single entry for each destination.

0 1

d)* Briefly explain the term **Tier-1 provider** based on the lecture and name all Tier-1 providers in the given network.

- Tier-1: Default-Free-Zone, only peerings, no providers
- AS2 and AS9 and AS11

0 1

e)* Briefly explain the term **stub network** based on the lecture and name all stub networks in the given network.

- Stub ASes have exactly one BGP relationship (with their provider)
- AS38

AS18 owns the prefix 10.0.0.0/22 and announces the prefix to the network.

0 1

f)* How is traffic from AS33 routed towards the announced prefix?

- AS25 does not announce the route towards AS20. It would generate traffic on a cost free link but AS25 needs to pay for its uplink.
- AS33 -> AS20 -> AS9 -> AS18

0 1

g)* Can AS43 reach the prefix?

- No AS33 would not announce the route towards AS43.
- Without any additional agreement AS43 would generate traffic on a cost free link but AS33 needs to pay for its uplink.

0 1 2 3

h)* Given that AS61 is not a provider and has to send traffic towards the prefix 10.0.0.0/22 via AS56, which type of relationships does AS61 have with AS56 and AS54? Explain your answer.

- AS61 -> AS56: Customer -> Provider
- AS61 -> AS54: Peering
- If both links would be peering links, AS61 could not reach AS18.
- If both would be customer-> provider relationships the path via AS54 would be shorter.

0 1 2

i)* AS54 wants to hijack the prefix from AS18 and announces 10.0.0.0/21. Which other ASes are impacted by this announcement? Is this a successful hijack?

- No other AS is impacted
- The announced prefix is less specific and does not take precedence in the routing table of any AS

Problem 4 P4 (13 credits)

This problem investigates a Software-Defined Network (SDN) powered by P4. The source code of a P4 switch program is given in Listing 1.

```
1  header eth_t { bit<48> dstAddr;
                  bit<48> srcAddr;
3                 bit<16> etherType; }

5  struct metadata { /* unused */ }

7  struct headers  { eth_t  eth; }

9  parser ParserImpl(packet_in packet, out headers hdr, inout metadata meta, inout standard_metadata_t
       standard_metadata) {

11     state parse_eth { packet.extract(hdr.eth);
                         transition select(hdr.eth.etherType) { default: accept; }}
13     state start     { transition parse_eth; }

15 }

17 control DeparserImpl(packet_out packet, in headers hdr) {
       apply { packet.emit(hdr.eth); }
19 }

21 control Pipeline(inout headers hdr, inout metadata meta, inout standard_metadata_t standard_metadata) {

23     action my_drop()               { mark_to_drop(standard_metadata); }
       action set_egress(bit<9> port) { standard_metadata.egress_spec = port; }
25     action set_default_egress()    { standard_metadata.egress_spec = 2; }
       action set_brdcast()           { \* __INSERT_P4_CODE_HERE__ *\ }
27
       table brdcast { actions = { set_brdcast; NoAction }
29                     key =     { standard_metadata.ingress_port: exact; }
                       default_action = set_brdcast(); }
31
       table forward { actions = { set_egress; my_drop; set_default_egress; }
33                     key =     { standard_metadata.ingress_port: exact; }
                       default_action = set_default_egress(); }
35
       apply { brdcast.apply();
37             forward.apply(); }

39 }

41 V1Switch(ParserImpl(), Pipeline(), DeparserImpl()) main;
```

Listing 1: Simple P4 program

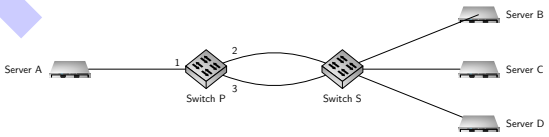

Figure 4.1: Network topology

For the following problems, the network topology is given in Figure 4.1. Switch P is a P4 switch running the P4 program of Listing 1 with Ports 1-3. Switch S is a regular, non-programmable switch. Switches P and S are connected via two links.
The network administrator of the network in Figure 4.1 wants to distribute all frames from Server A to Servers B, C, and D. Servers B, C, and D are configured to record each frame they receive. To distribute the frames, the two Switches P and S are used. Switch P uses the P4 program in Listing 1 that prepares frames for broadcasting, Switch S then performs the actual broadcast.

a)* Complete the `set_brdcast()` action in Listing 1.

```
hdr.eth.dstAddr= 0xff ff ff ff ff ff;
```

0
1
2

The network administrator notices that both switches become overloaded within seconds after connecting all devices. A closer investigation uncovers that the servers do not create enough traffic to overload the network.

0 1 2

b)* Explain the reason for the overload in the network.

- Broadcasts are cycled in an infinite loop between the two switches
- Handling of the broadcasts leaves little room for processing regular traffic and overloads the switches (broadcast storm)

0 1 2 3

c) List possible solutions how this kind of Problem (cf. Subproblem b)) can be solved in general (not using an SDN-capable hardware but regular switches and routers). Name and briefly explain one solution for Layer 1, one solution for Layer 2, and one solution for Layer 3.

- Layer 1: remove one cable / disable a port on a switch to break up the loop
- Layer 2: use the spanning tree protocol to create a loop-free network
- Layer 3: use a hop count / ttl to discard older packets to avoid overloading the network

0 1 2 3

d)* The administrator does not rely on the general solutions mentioned in Subproblem c), but wants the servers to reach each other using the correct table entries for the given P4 program. **Hint:** The tables below may contain more rows than the number of rules actually required to perform the described task.

Brdcast Table

Match field(s)	Key	Action	Action data
std_metadata.ingress_port	P.2	NoAction	-

Forward Table

Match field(s)	Key	Action	Action data
std_metadata.ingress_port	P.2	set_egress	port=P.1
std_metadata.ingress_port	P.3	my_drop	-

e)* Name the four P4 targets presented in the lecture.

0 1 2

Software, NPU, FPGA, ASIC

f) A researcher wants to modify the physical layer of a P4 device. Which P4 target(s) allow(s) this kind of modification?

0 1

- Software, NPU and ASICs can only use the physical capabilities available on the device, therefore, it cannot be modified without creating a new piece of hardware supporting the modification.
- FPGA allows modifying the networking hardware itself, i.e. also the physical layer.
- **Alternative solution:** Any target supporting switchable transceivers allows physical layer modification.

Problem 5 Network Calculus (13.5 credits)

This problem investigates performance bounds in networks using Network Calculus.
Consider the following network topology with flow and server descriptions. Flow f_1 traverses Servers s_1, s_2, and s_3. Flow f_2 traverses Servers s_1 and s_2. Assume each server handles flows according to strict priority scheduling with preemption. Assume Flow f_1 has a low priority and Flow f_2 has a high priority. In the following, we want to apply the Separate Flow Analysis to compute an end-to-end delay bound for Flow f_1.
Hint: Use the following formula to calculate a left-over service curve: $\beta^{l.o.} = \left[\beta_{R,T} - \gamma_{r,b}\right]^+ = \beta_{R-r,\frac{b+R\cdot T}{R-r}}$. An output arrival curve is given by $\alpha^\star = \alpha_{r,b+r\cdot T}$.

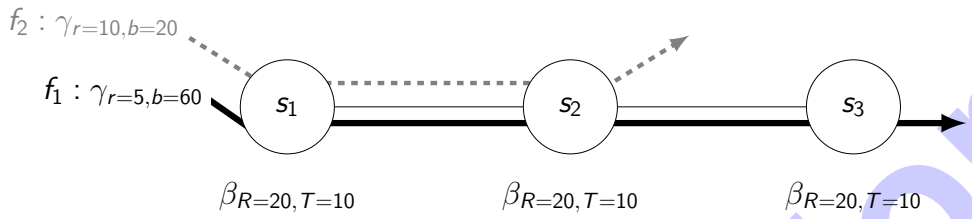

Figure 5.1: Network topology with flow and server specifications

0 1 2 3 4

a)* Perform the first step (left-over service curves) of the Separate Flow Analysis.

Left-over service curve at s_1:

$$\beta_{s_1}^{l.o.} = \left[\beta_{s1} - \gamma_{f_2}\right]^+ = \beta_{R=R_{s_1}-r_{f_2},T=\frac{b_{f_2}+R_{s_1}\cdot T_{s_1}}{R_{s_1}-r_{f_2}}} = \beta_{R=20-10,T=\frac{20+20\cdot 10}{20-10}} = \beta_{R=10,T=22}$$

Left-over service curve at s_2:

$$\gamma_{f_2}^\star = \gamma_{r_{f_2},b_{f_2}+r_{f_2}\cdot T_{s_1}} = \gamma_{10,20+10\cdot 10} = \gamma_{10,120}$$

$$\beta_{s_2}^{l.o.} = \left[\beta_{s2} - \gamma_{f_2}^\star\right]^+ = \beta_{R=R_{s_2}-r_{f_2}^\star,T=\frac{b_{f_2}^\star+R_{s_2}\cdot T_{s_2}}{R_{s_2}-r_{f_2}^\star}} = \beta_{R=R_{s_2}-r_{f_2},T=\frac{b_{f_2}+r_{f_2}\cdot T_{s_1}+R_{s_2}\cdot T_{s_2}}{R_{s_2}-r_{f_2}}} = \beta_{R=20-10,T=\frac{20+10\cdot 10+20\cdot 10}{20-10}} = \beta_{R=10,T=32}$$

Left-over service curve at s_3:

$$\beta_{s_3}^{l.o.} = \beta_{s3} = \beta_{R=20,T=10}$$

0 1 2

b) Perform the second step (end-to-end left-over service curve) of the Separate Flow Analysis.

End-to-end left-over service curve:

$$\beta_{e2e}^{l.o.} = \beta_{s_1}^{l.o.} \otimes \beta_{s_2}^{l.o.} \otimes \beta_{s_3}^{l.o.} = \beta_{R=min(R_{s_1}^{l.o.},R_{s_2}^{l.o.},R_{s_3}^{l.o.}),T=T_{s_1}^{l.o.}+T_{s_2}^{l.o.}+T_{s_3}^{l.o.}} = \beta_{R=min(10,10,20),T=22+32+10} = \beta_{R=10,T=64}$$

0 1

c)* Perform the third step (calculate the delay bound) of the Separate Flow Analysis. If you have no solution to b), assume an end-to-end left-over service curve of $\beta_{R=6,T=7}$.

End-to-end delay bound: $d = T_{e2e}^{l.o.} + \frac{b_{f_1}}{R_{e2e}^{l.o.}} = 64 + \frac{60}{10} = 70$
(Alternative: $d = 7 + \frac{60}{6} = 17$)

d)* Calculate the end-to-end delay bound of Flow f_2 using the Separate Flow Analysis.

0 1 2 3

Left-over service curves:

$\beta_{s_1}^{l.o.} = \beta_{R=20,T=10}$
$\beta_{s_2}^{l.o.} = \beta_{R=20,T=10}$

End-to-end left-over service curve:

$\beta_{e2e}^{l.o.} = \beta_{s_1}^{l.o.} \otimes \beta_{s_2}^{l.o.} = \beta_{R=min(20,20),T=10+10} = \beta_{R=20,T=20}$

End-to-end delay bound:

$d = T_{e2e}^{l.o.} + \frac{b_{f_2}}{R_{e2e}^{l.o.}} = 20 + \frac{20}{20} = 21$

e) Consider the following scenario for the network in Figure 5.1:

- The rate of all servers (s_1, s_2, and s_3) is set to $R = 12$.

Argue what the influence on the end-to-end delay bound of Flow f_1 is.

0 1

The delay bound will be infinity since the rate of the end-to-end left-over service curve is less than the rate of the flow f_1.Therefore, the maximum horizontal distance is infinite.

f)* Consider a token-bucket constrained flow, traversing a single rate-latency server. The corresponding arrival- and service curves are shown in Figure 5.2. Determine the backlog bound.

0 1

Visual determination: 60

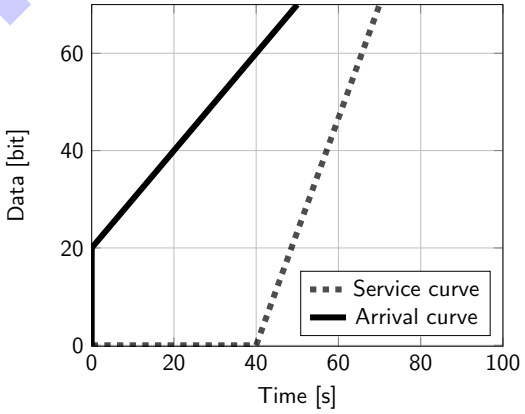

Figure 5.2: Arrival- and service curve

Additional space for solutions–clearly mark the (sub)problem your answers are related to and strike out invalid solutions.