CSEE 4119 Fall 2022

Homework 4 ANSWER KEY

Uploading course materials, including questions/answers from this homework, to sites such as CourseHero, Chegg or Github is academic misconduct at Columbia (see pg 10).

1. Switching and Scheduling [69 points, parts a-d]

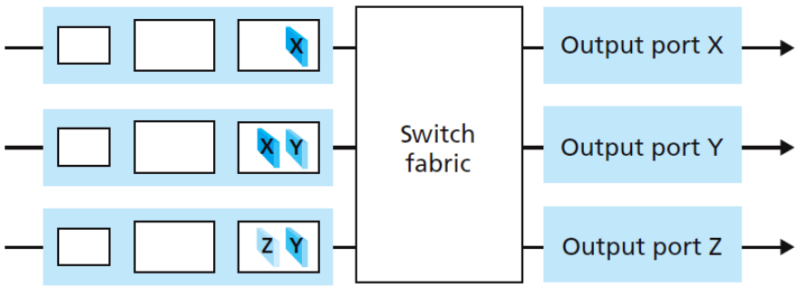
a. Suppose there are two packets arriving two different input ports of a router at exactly the same time. Assume there are no other packets anywhere in the router.

i. [2 points] If these two packets are to be forwarded to two different output ports. Is it possible to forward them at the same time when the fabric uses a *shared bus*? Explain why.
No. Only one packet can be transmitted at a time over a shared bus.
+1 correct answer
+1 correct explanation

ii. [2 points] Similar to question a, two packets are to be forwarded to different output ports. But this time, the fabric uses a *crossbar*, is it possible to forward them at the same time? Explain why.
Yes. As long as these two packets use different input buses and different output buses, they can be forwarded in parel.
+1 correct answer
+1 mentions when packets use different input buses.

iii. [2 points] If these two packets are to be forwarded to the same output port, still with a *crossbar*. Is it possible to forward them at the same time? Explain why.
No. It is impossible to forward two packets to the same output ports, no matter what methods are used by switching fabric.
+1 correct answer
+1 correct explanation

b. Consider the switch shown below. Suppose that all datagrams have the same fixed length, and the switch operates in a slotted, synchronous manner. In one time slot, a datagram can be transferred from an input to an output port. The letter labelled on the datagram means the output port it should be forwarded to.



i. [3 points] If the switch fabric uses a shared bus, what is the number of time slots needed to transfer all the packets from input ports to their output ports? Please figure out the packets forwarded in each time slot.
Five time slots. It takes one time slot for each packets.
+1 correct answer
+2 correct explanation

ii. [3 points] If the switch fabric uses a crossbar, what is the number of time slots needed? Please figure out the packets forwarded in each time slot.

Though there are several possibilities, the only three time slots are needed for all the cases.
There are two possibilities, either of them is correct.

1) In the first time slot, send X in top input queue, Y in middle input queue
The second time slot, send X in the middle, Y in the bottom.
The third time slot, send Z in the bottom.

2) In the first time slot, send X in top input queue, Y in bottom input queue
The second time slot, send Y in the middle, send Z in bottom
The third time slot, X in the middle

+1 correct answer
+2 correct explanation

c. [41 points] We will now consider switches based on interconnection networks, including some recent designs by Facebook. Facebook uses a (slight variation on a) Clos network topology to build their Six Pack switches.
As a starting point to understanding Clos topologies, here is an example Clos topology from the original 1953 paper "A Study of Non-Blocking Switching Networks":

here r=8, n=3, m=5 (24 in+outputs)
⇒ 16 8-port and 5 16-port switches
⇒ vs. 1 48-port switch

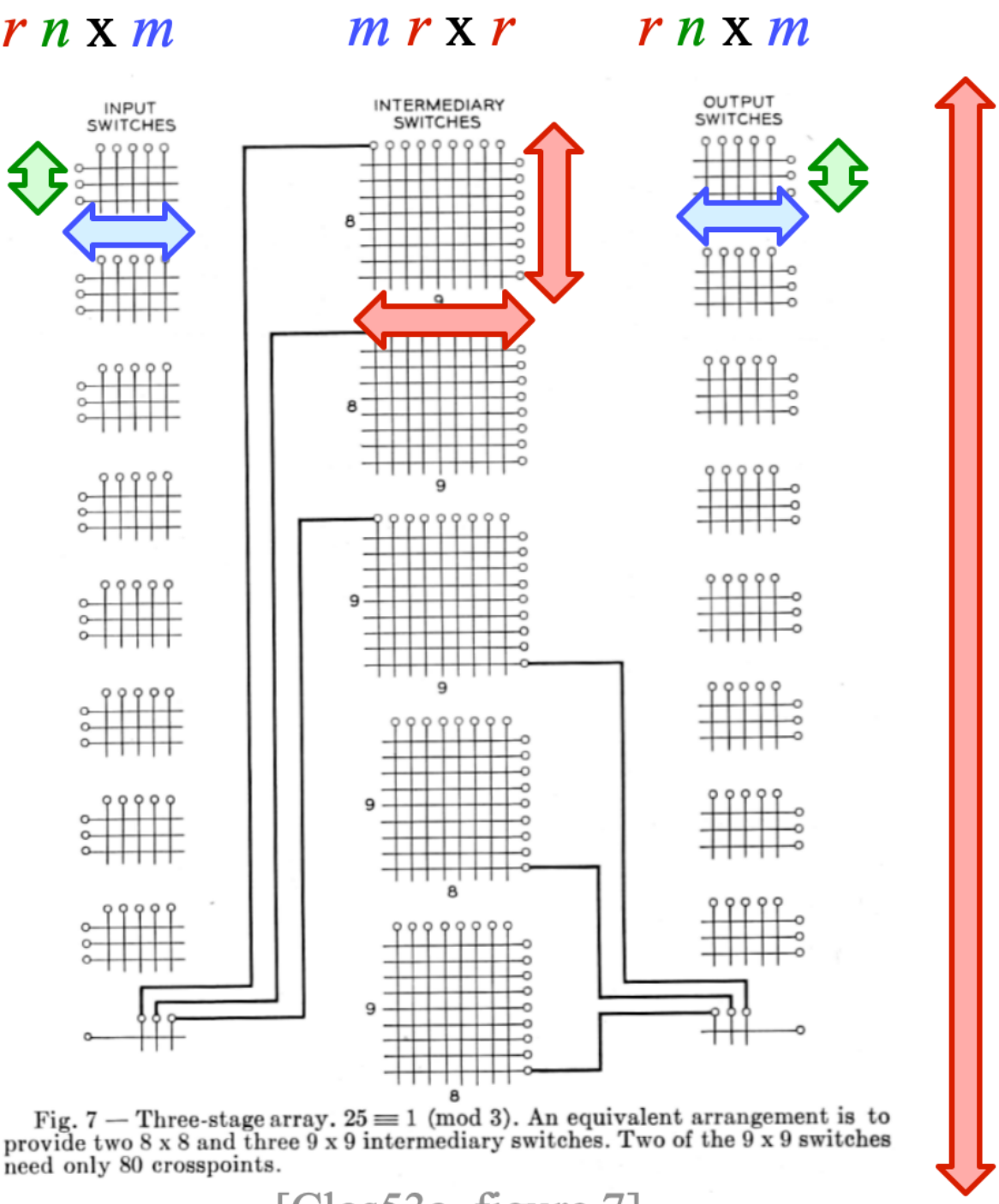


Fig. 7 — Three-stage array. $25 \equiv 1 \pmod 3$. An equivalent arrangement is to provide two 8 x 8 and three 9 x 9 intermediary switches. Two of the 9 x 9 switches need only 80 crosspoints.

[Clos53a, figure 7]

Consider a Six Pack switch that provides 64x40GE input ports and 64x40GE output ports (GE = Gbps Ethernet). [Facebook posted a decent reference article announcing their switch that you might want to look at.]

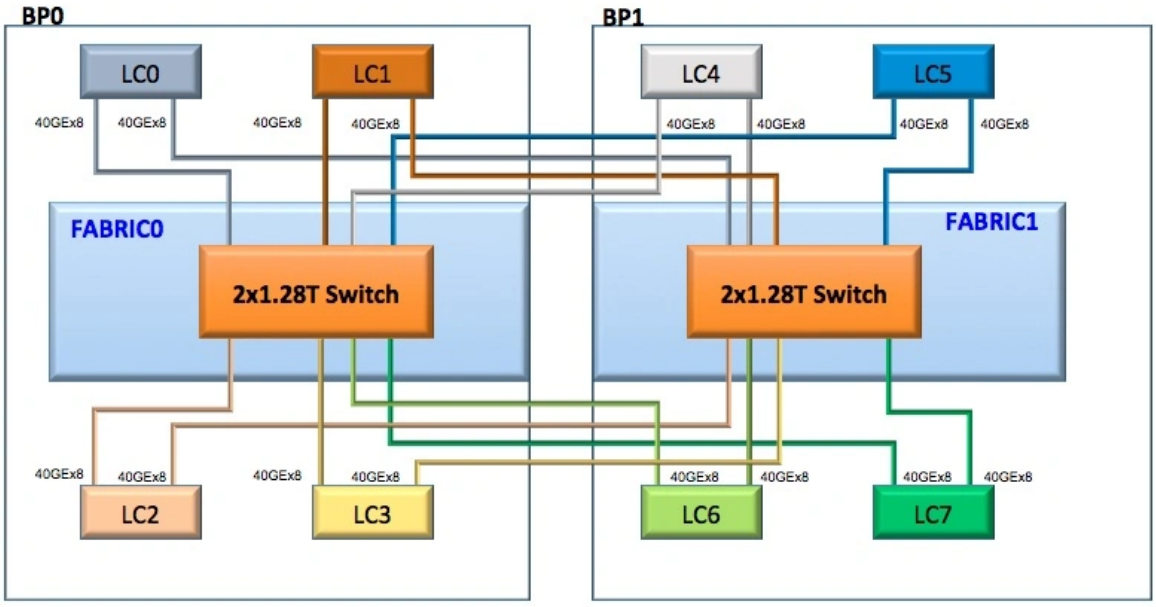
i. [3 points] What are the values of the Clos parameters n, m, and r in a Six Pack switch if it used a traditional Clos network? [Assume the same input and output switches as shown in the reference figure in Facebook's post about the switch, but the middle layer hardware does not have to be the same. Assume a three stage Clos network like the one in Clos53a Figure 7 above.]
ii. [6 points] List the switches necessary for this traditional Clos network. For example, if it would require 4 switches, each with 6 10Gbps inputs and 12 10Gbps outputs, your answer would be "4 6x12 10 GE".
iii. [3 point] Would this Six Pack switch based on a traditional Clos network be blocking or non-blocking? Why/why not? [describe the reasoning in your own words, don't just cite an answer]
iv. [3 point] What is different from a traditional Clos network about Facebook's slight variation on it?
v. [6 points] List the switches necessary for Facebook's Clos variation. For example, if it would require 4 switches, each with 6 10Gbps inputs and 12 10Gbps outputs, your answer would be "4 6x12 10 GE".
vi. [3 point] Is Facebook's variation blocking or non-blocking? Why/why not? [describe the reasoning in your own words, don't just cite an answer]
vii. [3 point] What is the main advantage of a Clos topology over a single larger switch that directly provides a 64x64 crossbar?
viii. [2 points] More recently, Facebook moved to a single ASIC (equivalent to the single larger switch providing a full crossbar mentioned in the earlier question) for their Minipack switch. What is the switching capacity of Minipack? (Describe input/output port configuration in a way similar to us telling you that Six Pack provides 64x40GE, 64x40GE ports.)
ix. [3 point] What is the primary reason Facebook moved to a single ASIC design rather than a Clos design?
x. [9 points] Design a three-stage non-blocking Clos network providing the same switching capacity as Minipack, split half and half between input and output. Use input switches that have 8x100GE inputs (and output switches that have 8x100GE outputs). You can choose how many outputs the input switches have and the size of any internal switches to fit your design.
    1. What are the parameters n, m, and r?

List the switches necessary for this traditional Clos network at each stage of the topology.
Answer/notes:

i. The 6 pack figure shows 4 input switches, each with 2 sets of 40GEx8 links to the aggregation switches (so a total of 16x40GE). Since the article mentions that the switches have a configuration that exposes "16x40GE ports to the front and 640G (16x40GE) to the back," this suggests that they each expose 16x40GE input. So the Clos parameters are r=4, n=16, m=16.

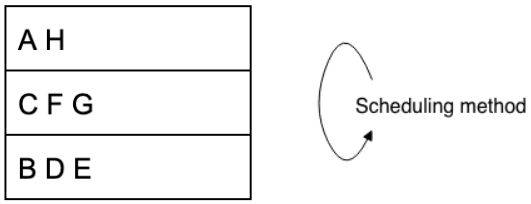
ii. A traditional Clos topology would then require 4 16x16 40GE input switches, 16 4x4 40GE intermediate switches, and 4 16x16 40GE output switches.
iii. Blocking: Assume that there is a free port on the input of an ingress switch, and this has to be connected to a free port on a particular egress switch. In the worst case, n-1=15 other connections are active on the ingress switch in question, and n-1=15 other connections are active on the output switch in question. Assume, also in the worst case, that each of these calls pass through a different middle-stage switch. Hence in the worst case, 2n - 2 of the middle stage switches are unable to carry the new call (because either their connection to the input switch is in use, or their connection to the output switch is in use). Therefore, to ensure strict-sense nonblocking operation, another middle stage switch is required, making a total of 2n - 1=31, which is more than the number of intermediate switches that we have.
iv. Instead of using 16 4x4 intermediate switches, each with one connection to each input switch and one connection to each output switch, Facebook uses 4 16x16 intermediate switches, each with 4 connections to each input switch and 4 connections to each output switch.
v. 12 16x16 40GE switches, 4 input, 4 intermediate, 4 output.
vi. Blocking: Assume that there is a free port on the input of an ingress switch, and this has to be connected to a free port on a particular egress switch. In the worst case, n-1=15 other connections are active on the ingress switch in question, and n-1=15 other connections are active on the output switch in question. There is only 1 available link from input to intermediate, and 1 available link from intermediate to output. In the worst case, they may not be on the same intermediate switch.
Non-blocking: Probably when Facebook says that it is non-blocking they mean rearrangeably non-blocking: "If m ≥ n, the Clos network is rearrangeably nonblocking, meaning that an unused input on an ingress switch can always be connected to an unused output on an egress switch, but for this to take place, existing calls may have to be rearranged by assigning them to different centre stage switches in the Clos network." From Wikipedia: https://en.wikipedia.org/wiki/Clos_network (thanks to student Jingxin Jiang for this reasoning) So perhaps they mean that Six Pack can move existing flows to make room for the new one, making it such that the 1 available link on the input switch and the 1 available link on the output switch share a fabric card in common.
vii. It's generally cheaper.
viii. 128 x 100GE
ix. Less power
x. We'll design it for 64 x 100GE input and 64 x 100GE output. That means we have 8 input switches with 8x100GE in, so r=8, n=8. To make it non-blocking, we need m >= 2n-1 (see the argument in (iii)), so m>=15. So for example we can use 8 8x15 100GE input switches, 15 8x8 100GE intermediate switches, and 8 8x15 100GE output switches.

Six pack:



"It is a full mesh non-blocking two-stage switch that includes 12 independent switching elements. Each independent element can switch 1.28Tbps. We have two configurations: One configuration exposes 16x40GE ports to the front and 640G (16x40GE) to the back, and the other is used for aggregation and exposes all 1.28T to the back." From https://engineering.fb.com/2015/02/11/production-engineering/introducing-6-pack-the-first-open-hardware-modular-switch/

d. Now let's delve deep in packet scheduling methods. Consider the input queues displayed below. All the packets are classified into three classes. A and H are in class 1, C, F, G belong to class 2, B, D, E belong to class 3. The arrival time of all packets and the length of each packet are shown in the table below. Suppose it will take 1 unit of time to transmit a packet for each unit of length. Suppose that a packet's transmission will not be interrupted to begin transmission of another packet (i.e. non-preemptive).



Packet	Arrival Time	Length
A	0	8
B	5	6
C	5	10
D	8	9
E	8	8
F	10	6
G	11	10
H	20	8

i. [4 points] If Round Robin is used, please figure out the transmission order of packets, and give your explanation. (Suppose Round Robin starts from class 1, and iterate in the order of 1->2->3->1...)
A C B H F D G E
In round-robin, we take one packet in turn from each non-empty queue at the next output time.
+2 correct answer (+1 if there is a minor mistake)
+2 correct explanation (+1 if there is a minor mistake)

ii. [12 points] If Weighted Fair Queuing is used, we set the weight of both class 1 and class 2 to be 1, and the weight of class 3 to be 2, then what is the order of the transmission order of packets? Please write down in detail how you get your answer. Notice Weighted Fair Queueing is not fully discussed in the textbook, so you can learn how it works by yourself. (Hint: In weighted fair queuing, packets are selected to be forwarded according to the virtual departure time).
ABDCEHFG or BADCEHFG

+2 correct answer (+1 only for minor calculation error)
+1 only correct answer for simplified equation

+2 correct explanation
+1 only correct explanation for simplified equation

You can find the answer using virtual finish time/ bit by bit round robin/GPS-fluid-model. You can refer to the table and explanation below to figure out how it really works.

Here are some useful links:
https://en.wikipedia.org/wiki/Weighted_fair_queueing
https://intronetworks.cs.luc.edu/current/html/queuing.html

Packet	Arrival Time	Length	Finish time	Output order
A	0	8	17	1
B	5	6	17	2
C	5	10	44	4
D	8	9	34	3
E	8	8	50	5
F	10	6	55	7
G	11	10	65	8
H	20	8	51	6

Here is an overview of how the fluid model works for the first 20 units of time (emulating a situation in which multiple flows can be sent at the same time):

at 0-5, A will take all the bandwidth
at 5-17, A will take ¼, C will take ¼, B will take ½, so for A, it will need 3/ ¼ = 12 units of time to finish transmitting. As a result, the finish time of A is 5+12=17. For B 6 / ½ =12 units of time necessary as well, so B will also finish transmitting at 17.
at 17-20, D will take ⅔ and C will take ⅓ of the bandwidth.
at 20 H will arrive and start transmitting, making the new bandwidth allocation ¼ for H, ¼ for C and ½ for D.

2. Internet Protocols [25 points, parts a-n]

a. [8 points] Given a subnet with IPv4 address whose prefix is 127.0.8.0/23. Design address assignment for each subnet of incoming users. You are not allowed to waste addresses. You need to provide the smallest number of addresses and lowest unassigned addresses for each subnet.
   i. [2 points] Design address assignment for the first subnet with 12 users. (Please explain by describing how many addresses are in the subnet total, and how many are free for allocation)
   Answer:
   Total network prefix: 127.0.8.0/23

01111111 00000000 0000100X XXXXXXXX
Subnet 1: 16 interfaces
01111111 00000000 00001000 0000XXXX
14 addresses are free for allocation (The first one and last one is reserved by default)
127.0.8.0/28
+1 for correct answer
+1 for explanation

ii. [2 points] Design address assignment for the second subnet with 61 users. (Please explain by describing how many addresses are in the subnet total, and how many are free for allocation)
Answer:
Subnet 2: 64 interfaces
01111111 00000000 00001000 01XXXXXX
62 addresses are free for allocation (The first one and the last one is reserved by default)
127.0.8.64/26
+1 for correct answer
+1 for explanation

iii. [2 points] Design address assignment for the third subnet with 61 users. (Please explain by describing how many addresses are in the subnet total, and how many are free for allocation)
Answer:
Subnet 3: 64 interfaces
01111111 00000000 00001000 10XXXXXX
62 addresses are free for allocation (The first one and last one is reserved by default)
127.0.8.128/26
+1 for correct answer
+1 for explanation

iv. [2 points] Design address assignment for the last subnet with 64 users. (Please explain by describing how many addresses are in the subnet total, and how many are free for allocation.
Answer:
Subnet 4: 128 interfaces considering the first and last address in subnet.
01111111 00000000 00001001 0XXXXXXX
126 addresses are free for allocation (The first one and last one is reserved by default)
127.0.9.0/25
+1 for correct answer
+1 for explanation

b. [1 point] Other than the length of the source and destination address, what is one **key** change / difference from IPv4 to IPv6? (any difference in just the name of fields or the size of fields should not be considered a key difference)
Answer:
- IPv6 header is fixed-length and IPv4 is variable-length.
  Reason: To streamline IPv6 header processing on the router.
- IPv6 removed options field in IPv4.
  Reason: The header can stay fixed length in order to streamline header processing.
- There is no checksum field in IPv6 header.
  Reason: IPv4 had TTL field that gets decremented at each intermediate router, each router has to recompute the checksum field. Removing the checksum field in IPv6 makes header processing less costly.
- IPv6 does not allow fragmentation and reassembly by the routers.
  Reason: Fragmentation and reassembly are time-consuming and removing it speeds up IP packet forwarding in the network

+1 for each reason and +1 for corresponding explanation

c. [1 point] Why was this change made?
Refer to corresponding answer in b
+1 for each reason and +1 for corresponding explanation

d. [1 point] Give another **key** change or difference from IPv4 to IPv6 headers (other than the one you answered previously and other than the change in address length or superficial changes in the lengths or names of fields).

e. [1 point] Why was this change made?
Refer to corresponding answer in b
+1 for each reason and +1 for corresponding explanation

f. [2 points] Transitioning from IPv4 to IPv6 has been a challenging problem in practice. There have been multiple proposals to transition (or help transition) from IPv4 to IPv6. One approach could be to have a world-wide "flag-day". On (or by) a designated date called flag-day, every machine on the Internet is assigned a new IPv6 address, all routing tables are recomputed, and IPv4 is no longer used. Is this "flag-day" proposal feasible for the transition to IPv6? Explain why or why not.

(1 point) No.
(1 point) Flag-day is not administratively feasible to do it. (Answer any two out of four points below would give full credit. Otherwise, 0.5pt is granted for each correct answer)

g. First of all, every network administrator has to agree to configure their machines with assigned IPv6 address.
h. Second, there's also the cost of buying and replacing IPv6 hardware and routers.
i. Third, no changes can happen simultaneously and instantaneously, so it takes time to adapt the changes.
j. Lastly, overhauling the existing infrastructure within a short period of time is more likely introduce security loopholes due to time constraint.

k. [1 point] When TCP/IP was adopted, there was a flag day. What IETF RFC defined the flag day? (Just give the # of the RFC)
RFC801

l. [1 point] When was the TCP/IP flag day?
Jan 1, 1983

m. [1 point] What protocol was in use before the TCP/IP flag day?
(Network Control Program) NCP

n. [1 point] Why was a TCP/IP flag day possible, when an IPv6 flag day was not?
The NCP to TCP/IP transition plan was possible because they were both deployed on **ARPANET, which at that time was very small** (only ~400 hosts) compared to modern Internet. Only 400 hosts connected to ARPANET was affected so it was feasible to complete the transition by a deadline (the NCP-TCP flag day). Related, whereas **today Internet hosts are administered by a very large number of entities**, many commercial businesses, the ARPANET then was under much more unified control.
(1 point) mentioned that ARPANET is small compared to today's huge-scale Internet.

o. [1 point] Dual stack approach and tunneling are two practical approaches to transition from IPv4 to IPv6. Explain dual stack in one or two sentences.
Dual Stack Approach: deploy or upgrade routers to support both IPv4 and IPv6 (dual stacks), so that they can translate between IPv4 addresses and IPv6 addresses.

p. [1 point] Explain IPv4/IPv6 tunneling in one or two sentences.
Tunneling: A dual stack router encapsulates IPv6 packets inside IPv4 packets to send across a region that of IPv4-only routers.

q. [2 points] Both dual stack and tunneling are needed to transition from IPv4 to IPv6. Why is dual stack alone not a solution? Why is tunneling alone not a solution? Explain each with one or two sentences.
Dual Stack: similar to the flag day problem, universal simultaneous upgrade to dual stack approach is not administratively feasible to do it. Deploying new routers that can translate from IPv4 to IPv6 (and vice versa) is costly and slow.
Tunneling: to form the tunnel end points, we need nodes that implemented both dual stack and tunneling.

r. [3 points] With a transition to IPv6, is it still necessary to minimize the number of forwarding rules per router using strategies such as longest prefix matching? Explain why/why not.
Yes it is still necessary, more forwarding rules requires more memory which is an expensive resource. IPv6 solves the address-exhaustion problem, which is orthogonal to this limitation.

3. OSPF and BGP [15 points]

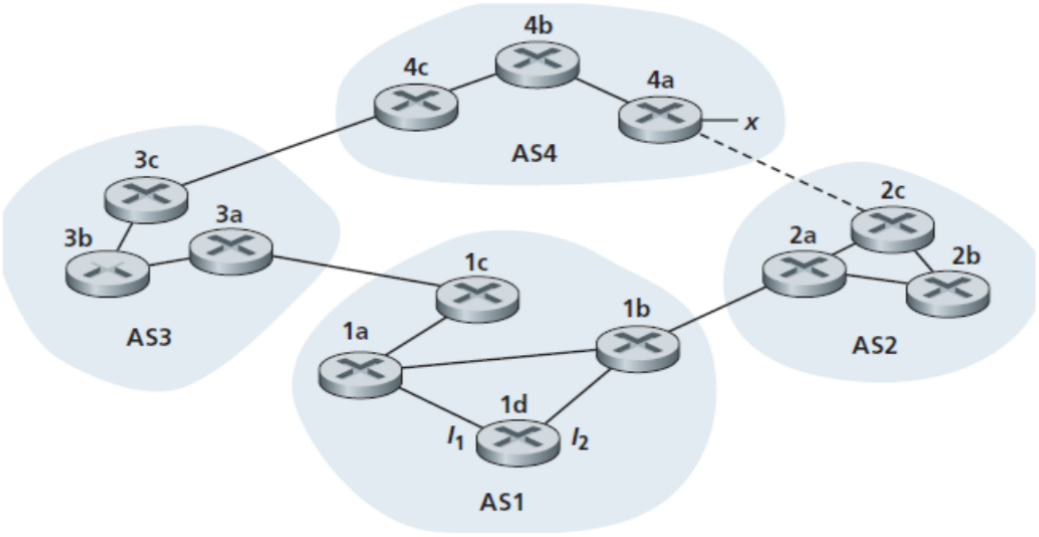


Figure 1. Network for Q1

Consider the network in Figure 1. Suppose all AS's are running OSPF for their intra-AS routing, and eBGP and iBGP are used for inter-AS routing. Initially suppose there is no physical link between AS2 and AS4. Once router 1d learns about x it will put an entry (x,I) in its forwarding table. Assume that each link has the same weight.

a. [2 points] Router 1d learns about x from which routing protocol?
   iBGP

b. [2 points] Will *I* be equal to I_1 or I_2 for this entry? Why?
   I_1
   I equals to I1 because there is no link between AS2 and AS4 so that data should be forwarded from 1d to 1c and the least cost path is the path via 1a.

c. [2 points] Now assume there is a physical link between AS2 and AS4 as shown by the dotted line. Router 1d learns that x is accessible both via AS3 and AS2. Will *I* be equal to I_1 or I_2 for this entry if AS1 assigns a higher local preference to routes learned from AS3 than to routes learned from AS2? Why?
   I_1
   Local preference has the highest priority when determining path and because of that in this case the path through AS3 will be prefered.

d. [2 points] Would the answer for the previous question change if the local preference for AS2 and AS3 were the same?
   Yes, in this case I will be set to I_2 because the AS-PATH length is equal and it should consider the the closest NEXT-HOP router which is 1b.

e. [2 points] Continue to assume that the local preferences for AS2 and AS3 are the same. Now assume there is another AS, called AS5, with 1 router, which lies between AS2 and AS4. Will *I* be equal to I_1 or I_2 for this entry? Why?
I_1
I will be set to I_1 as this path has a shorter AS-PATH length

f. [2 points] We refer to a protocol like OSPF, which is used to exchange routing information for destinations within a single AS, as an Interior Gateway Protocol (IGP). What is one important reason an AS commonly needs to use iBGP over an IGP, instead of using iBGP as an IGP?
+2 for mentioning ANY one of the following reasons,
- iBGP can't solve IP/router-level **loops** whereas IGPs like OSPF can.
- iBGP doesn't support **metrics** but IGPs like OSPF can.
- iBGP can't tell you how to get to **next hop** but IGPs like OSPF can. iBGP assumes the route already can reach the nexthop, whereas OSPF builds up that state.

We saw that the "count-to-infinity" problem can cause temporary routing loops for a distance vector protocol when link costs change. However, assuming stable/static link costs, Bellman-Ford style distance vector should not create loops.

g. [1 points] How does Bellman-Ford avoid creating routing loops?
+1 for mentioning shortest path
+1 for mentioning the fact that shortest path should never contain loops.
Given stable (non-negative) link costs, a shortest path should never contain a loop.

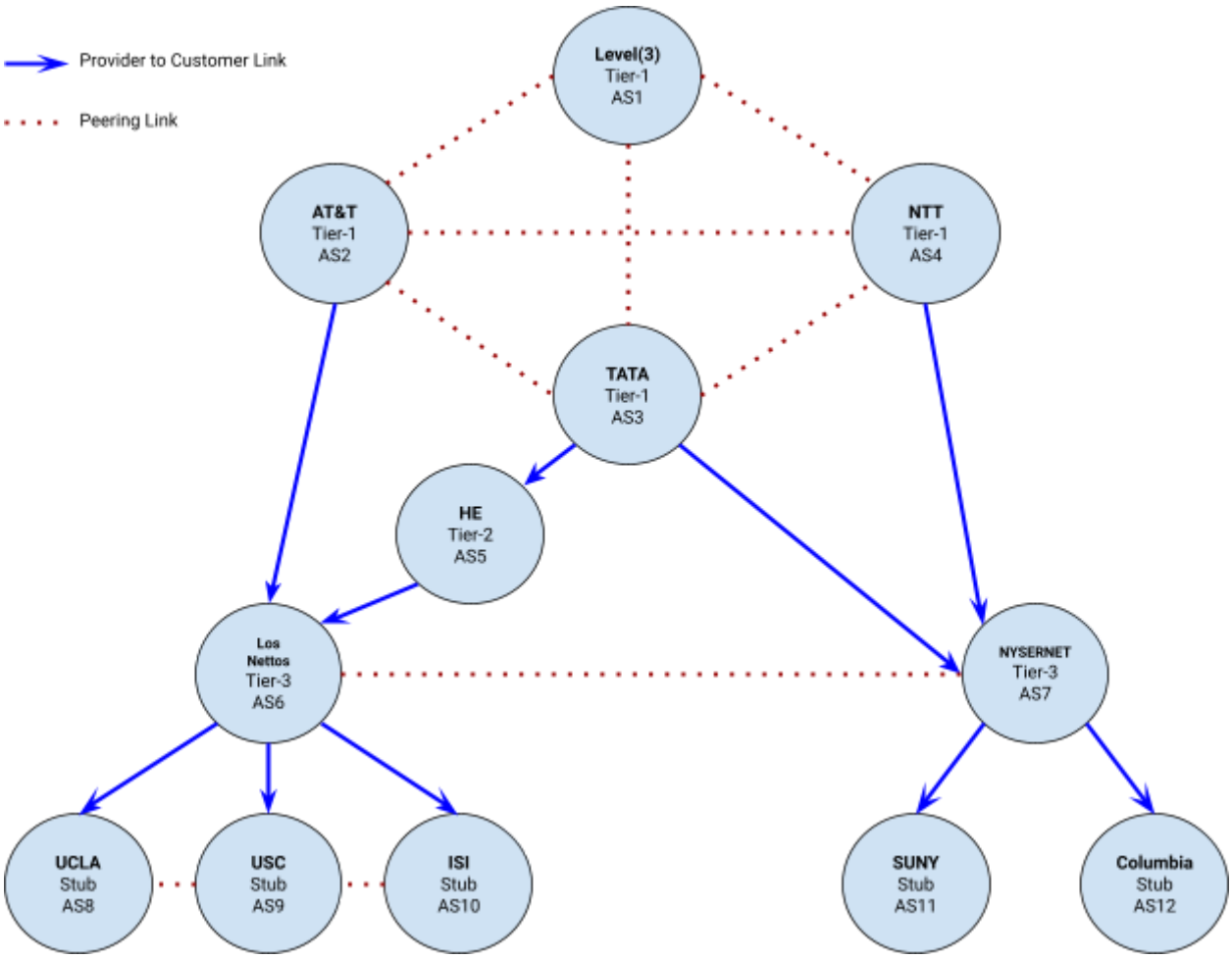
h. [2 points] eBGP uses a different mechanism from Bellman-Ford to avoid creating loops. Why can't it use the same approach as Bellman-Ford?
2/2: Bellman-Ford relies on (a) announcements carrying global costs and (b) every node using a policy that minimizes these global costs. For full credit, an answer must indicate that BGP is missing at least one of these properties (it is missing both) and that the local policy of BGP therefore requires more information to avoid loops (since distance alone won't be enough to avoid loops if ASes aren't trying to minimize distance). Each node can use arbitrary policy, so there is both no notion of global cost and no universal optimization.

1/2: Announcements do not have distances in them. This isn't quite right, because AS path length is could be used as a distance metric. However, BGP has to allow autonomous decisions so cannot rely on a universal optimization and so cannot force ASes to minimize path length--path length is not a global cost.

1/2: BGP enables arbitrary policy. This alone is not enough for full credit, as no AS is going to have a policy that allows loops. It needs to be coupled with the fact that B-F wouldn't provide enough information to couple policy with loop avoidance.

i. [2 points] iBGP uses a different mechanism from eBGP to avoid creating loops. How does iBGP avoid creating routing loops?
+1 for mentioning iBGP sets up full mesh.
+1 for mentioning routers do not redistribute/announce iBGP routes to other iBGP routers.
iBGP sets up a full mesh, where every router connects to every other router directly, and routers do not redistribute iBGP routes to other iBGP routers (each router only redistributes the routes it learns via eBGP).

4. BGP Routing Policy [33 points]



This question focuses on how routes propagate between networks across different types of connections.

Topology overview:

- There are four types of networks (tier 1,2,3 and stub).
- There are two types of interconnections (provider / customer and peering).
    - For provider / customer links, the arrow points from the provider to the customer.
- Each AS advertises a single /8 prefix, with the first octet equal to the network's ASN, similar to in the class project.
    - For instance, AS1 advertises 1.0.0.0/8 and AS7 advertises 7.0.0.0/8.
- Unlike the class project, each AS only has a single router (no internal topology).
- Each AS's router receives routes advertised by neighbors and selects a best path for each destination (BGP) using policy based routing. All ASes use both *no-valley routing* and *prefer-customer routing*.
- When two paths exist to a destination, an AS will use LOCAL_PREF and the AS_PATH length to choose between them. Assume that the LOCAL_PREF is already set by each AS to achieve *prefer-customer routing* (below).

No-valley routing and prefer-customer routing *(similar to what you'll implement in project 2, stage C)***:**

- No-valley routing specifies that you should not advertise routes learned from one provider/peer to another provider/peer, and prefer-customer routing specifies that if you can reach a destination through multiple AS-paths, you should always prefer the paths that go through your customers first, then peers, then providers.
- If you have paths through multiple neighbors of the same class, prefer-customer does not specify which to choose. In such a case, the route with the shorter AS_PATH will be preferred. If two routes are of the same class and have equal AS_PATH lengths, then they are considered equal for this assignment.

a. (2 points) Will TATA be willing to carry traffic from Los Nettos destined to NYSERNET? Why or why not?
   Yes--Tata will carry ANY traffic to Nysernet because Nysernet is its customer
   +1 Yes.
   +1 Reasonable explanations.
b. (2 points) Will Los Nettos be willing to carry traffic from NYSERNET destined to TATA? Why or why not?
   No--neither will pay it.
   +1 Yes.
   +1 Reasonable explanations.
c. (2 points) Will NYSERNET be willing to carry traffic from TATA destined to NTT? Why or why not?

No--neither will pay it.
+1 Yes.
+1 Reasonable explanations.

d. (2 points) Will TATA be willing to carry traffic from NYSERNET destined to NTT? Why or why not?
Yes--NYSERNET will pay it.
+1 Yes.
+1 Reasonable explanations.

e. (4 points) Suppose NYSERNET prefers to route all traffic destined to Level(3) via TATA if a route via TATA exists, and only use NTT as a backup. What BGP configuration in routers is needed to achieve this goal?
NYSERNET should configure its routers to assign higher local pref to routes from TATA than routes from NTT.
+2 mention assign local pref
+2 mention TATA's local pref is higher than NTT

f. (4 points) Suppose NYSERNET prefers to receive traffic via TATA rather than NTT, in cases in which the sender does not have a strong preference, but will let the sender choose if the sender has a strong preference. What BGP configuration is needed to achieve this goal? Be specific about how BGP should be configured in each relevant router, referring to the routers by name as needed.
NYSERNET should "prepend" its announcements to NTT, setting the AS path to have multiple instances of AS7 (or NYSERNET). This will cause that route to be longer than the one via TATA, and so it will lose out on decisions based on AS path length.
+2 mention setting AS path to have multiple instances of NYSERNET (or AS7)
+2 mention make the route via NTT longer than the one via TATA

g. (11 points) AS1 has a route to every prefix in the Internet, but the route that it uses to reach each destination depends on (1) the routes it receives from neighbors and (2) its application of the BGP route selection process. Both of these aspects are controlled by the *no-valley routing* and *prefer-customer routing* policies (see above).

Fill out the table below with the AS path of the route that AS1 will use to reach each prefix as a comma separated string. AS paths can be formed by appending each AS encountered on the selected route from AS1 to the AS advertising the prefix. Do not include AS1 in the path. If there are multiple equivalent paths, list all of the paths (separated by the word "OR").

For instance, if the route goes from AS1 -> 3 -> 5 -> 6 -> 7, the AS path would be 3,5,6,7

Prefix	**AS Path(s) of Best Route from AS1**
1.0.0.0/8	

2.0.0.0/8	
3.0.0.0/8	
4.0.0.0/8	
5.0.0.0/8	
6.0.0.0/8	
7.0.0.0/8	
8.0.0.0/8	
9.0.0.0/8	
10.0.0.0/8	
11.0.0.0/8	
12.0.0.0/8	

Prefix	**AS Path(s) of Best Route from AS1**
1.0.0.0/8	
2.0.0.0/8	2
3.0.0.0/8	3
4.0.0.0/8	4
5.0.0.0/8	3,5
6.0.0.0/8	2,6
7.0.0.0/8	3,7 OR 4,7
8.0.0.0/8	2,6,8
9.0.0.0/8	2,6,9
10.0.0.0/8	2,6,10
11.0.0.0/8	3,7,11 OR 4,7,11
12.0.0.0/8	3,7,12 OR 4,7,12

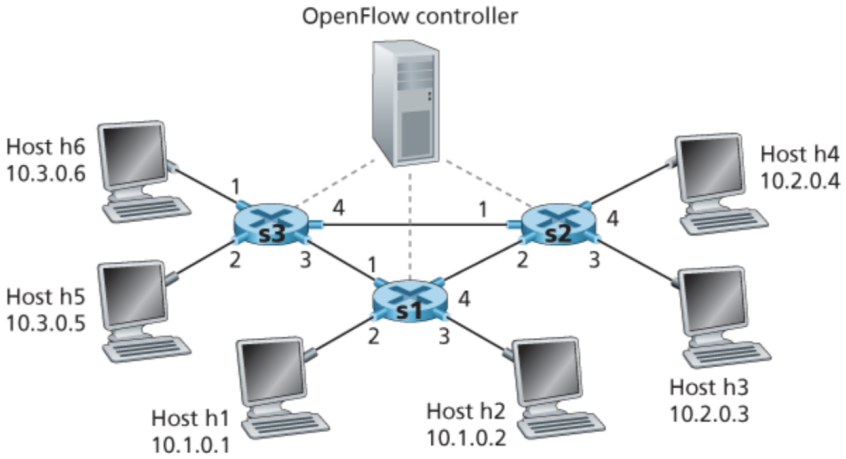
+1 for each correct row

h. (2 points) Does AS10 receive a route to 5.0.0.0/8 from AS6? Why or why not?
Yes--AS10 is a customer of AS6 and will pay AS6. So AS6 will still advertise routes learnt from its provider (AS5) to its customer (AS10).
+1 Yes.
+1 For mentioning AS10 is a customer of AS6 or other reasonable explanations.

i. (2 points) Does AS5 receive a route to 7.0.0.0/8 from AS6? Why or why not?
No--AS5 is AS6's provider and AS7 is AS6's peer. According to no-valley routing, AS6 will not advertise 7.0.0.0/8 to AS5.
+1 No.
+1 For mentioning no-valley routing and correctly identifying their relationships, or other reasonable explanations.

j. (2 points) Does AS6 receive a route to 4.0.0.0/8 from AS7? Why or why not?
No--AS6 is AS7's peer and AS4 is AS7's provieder. According to no-valley routing, AS7 will not advertise 4.0.0.0/8 to AS6.
+1 No.
+1 For mentioning no-valley routing and correctly identifying their relationships, or other reasonable explanations.

5. Generalized Forwarding and OpenFlow [12 points]



Consider the OpenFlow match-plus-action network shown above (figure 4.30 from textbook chapter 4.4.3). There are three packet switches marked as s1-s3, six hosts marked as h1-h6 and one OpenFlow controller.

a. [2 points] Design S2's flow table so that it achieves the following goals:

i. Datagrams arrived on port 4 and destined to h5 or h6 should be forwarded to port 1.
ii. Datagrams arrived on port 3 and destined to h1 or h2 should be forwarded to port 2.

S2 Flow Table	
Match	**Action**
…...	…...

Answer:

S2 Flow Table	
Match	**Action**
Ingress Port = 4; IP Dst = 10.3.*.*	Forward (1)
Ingress Port = 3; IP Dst = 10.1.*.*	Forward (2)

+1 If achieve design goal i)
+1 if achieve design goal ii)

b. [1 point] Based on the design in (a), what will happen if host 4 sends a datagram to host 3?

Answer:

When S2 receives this datagram, it cannot find a matching entry in the flow table. It can either drop it or send it to the remote controller for processing.
+1 For correct answer

c. [3 points] OpenFlow's match-plus-action is a more general paradigm in which matches can be made over multiple header fields associated with different protocols at different protocol stack layers. Suppose we want to use S2 as a firewall so that:
   i. Packets coming from h1 and h2 can be delivered to h3 or h4 (if they don't violate other rules) while packets coming from h5 and h6 will be blocked.
   ii. Only packets sent to port 80 will be delivered.

Note: You don't need to write down flow table entries for other behaviors of S2.

S2 Flow Table	
Match	**Action**

…..	…..

d. [2 points] Suppose due to a business concern, the network administrator wants to make sure that all packets at S2 destined to 10.3.*.* are forwarded to port 2 so installs one entry into S2's flow table:

S2 Flow Table	
Match	**Action**
IP Dst = 10.3.*.*	Forward (2)

Additionally, h3, which is directly connected to S2, prefers to route all of its packets via S3. Therefore, the administrator installs another entry into S2's flow table:

S2 Flow Table	
Match	**Action**
IP Dst = 10.3.*.*	Forward (2)
IP Src = 10.2.0.3	Forward (1)

Now if h3 sends a packet to h5 or h6, this packet will match two entries in S2's flow table. Suppose h3 is absolutely vital, so the administrator will try to meet h3's demand regardless of business concerns. If we can't modify the Match condition in S2's flow table, how can we help accomplish this goal at S2? Hint: There are other components in OpenFlow's flow table entry.

Answer:

We can make use of flow table entry's priority attribute. By assigning higher priority to the second entry, we can make sure h3's demand is always met.

+2 For correct answer

e. [4 points] A key difference between SDN and traditional networking is infrastructure - SDN is software-based and centralized, while traditional networking is hardware-based.

i. What is an advantage of doing routing in software?

Acceptable advantages of doing routing in software are including, but not limited to: Software is easier/faster to update (than hardware); Easier to debug (allows for abstraction); Greater flexibility.

+ 2 for correct answer

ii. What is an advantage of having a centralized perspective of routing?

Acceptable advantages of centralized perspective of routing are including, but not limited to: Easier to do network management; Full view of the network allows for better/more efficient decisions than localized ones; Better for traffic engineering (e.g. split traffic, special routes).

+ 2 for correct answer