CSCI4211: Introduction to Computer Networks
Fall 2017
Homework Assignment 2

Due 11:59pm Friday November 17

Instructions:

1. Please submit your homework using the on-line electronic submission system (via Moodle) – click on the“Submit” link on the class website.

   In case you could not use the on-line electronic submission system, please hand in your homework to the instructors or TAs on the due day. Please email csci4211-help@cs.umn.edu to let us know that you have handed in a hard-copy of your homework immediately afterwards. (*Make sure that you also make and retain a copy of your homework!*)

   Please make sure that you include your name and student id in your submission, and retain a copy of your submission!

2. There are **seven** questions in total. The number of points for each question is given in parentheses. There are **131** points in total. An *estimated* time for answering each question is also given in parentheses. This is just a guideline, you may take less or more time on each problem.

3. Partial credit is possible for an answer. Please try to be as concise and make your homework as neat as possible. We *must* be able to read your handwriting in order to be able to grade your homework.

4. Enjoy!

1. TCP Flow and Congestion Control (25 points total. Approx. 25 minutes)

a. (5 points) Briefly describe how TCP flow control works.

Answer

TCP provides a flow-control service to its applications to eliminate the possibility of the sender overflowing the receivers buffer. Flow control is thus a speed-matching servicematching the rate at which the sender is sending against the rate at which the receiving application is reading buffer by sending too much data too quickly. TCP provides flow control by having the sender maintain a variable called the receive window. Informally, the receive window is used to give the sender an idea of how much free buffer space is available at the receiver.

b. [In answering the following questions, we assume that the TCP Reno version (the most commonly used version in today's Internet) is used, i.e., with the *Fast Recover/Fast Retransmit* mechanisms implemented.]

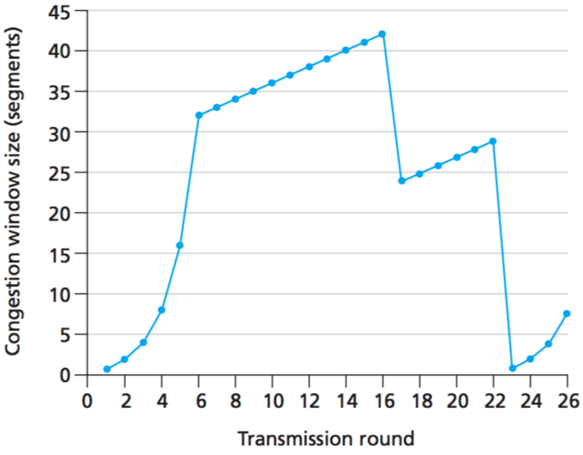
Do Problem P.40 a., b. c., d., Chapter 3 of the Textbook (*Kurose & Ross*, 7th Edition, p. 297). In case you do not have the current version of the textbook, the problem is reproduced below for you.

i. Identify the intervals of time when TCP slow start is operating.

ii. Identify the intervals of time when TCP congestion avoidance is operating.

iii. After the 16^{th} transmission round, is segment loss detected by a triple duplicate ACK or by a timeout?

iv. After the 22^{nd} transmission round,is segment loss detected by a triple duplicate ACK or by a timeout?



Answer

i. TCP slow start is operating in the intervals [1,6] and [23,26].

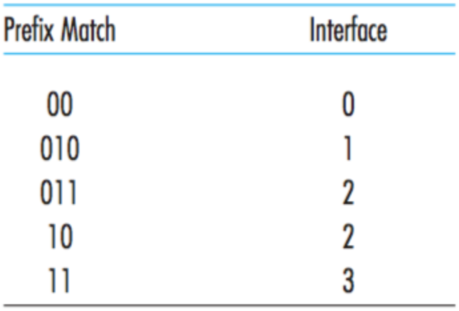
ii. TCP congestion avoidance is operating in the intervals [6,16] and [17,22]

iii. After the 16^{th} transmission round, packet loss is recognized by a triple duplicate ACK. If there was a timeout, the congestion window size would have dropped to 1.

iv. After the 22^{nd} transmission round, segment loss is detected due to timeout, and hence the congestion window size is set to 1.

2. IP Addresses and Longest Prefix Matching (10 points. Approx. 20 minutes)

a. (5 points) Do Problem P.6, Chapter 4 of the Textbook (*Kurose & Ross*, 7th Edition, p. 366). In case you do not have the current version of the textbook, the problem is reproduced below for you. Consider a datagram network using 8-bit host addresses. Suppose a router uses longest prefix matching and has the following forwarding table:



Prefix Match	Interface
00	0
010	1
011	2
10	2
11	3

For each of the four interfaces, give the associated range of destination host addresses and the number of addresses in the range.

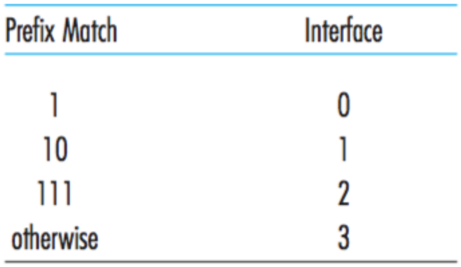
Answer

Destination Address Range	**Link Interface**
00000000 through 00111111	0
01000000 through 01011111	1
01100000 through 01111111	2
10000000 through 10111111	2
11000000 through 11111111	3

number of addresses for interface 0 $= 2^6 = 64$
number of addresses for interface 1 $= 2^5 = 32$
number of addresses for interface 2 $= 2^5 + 2^5 = 32 + 64 = 96$
number of addresses for interface 3 $= 2^6 = 64$

b. (5 points) Do Problem P.7, Chapter 4 of the Textbook (*Kurose & Ross*, 7th Edition, p. 367).

In case you do not have the current version of the textbook, the problem is reproduced below for you. Consider a datagram network using 8-bit host addresses. Suppose a router uses longest prefix matching and has the following forwarding table:



Prefix Match	Interface
1	0
10	1
111	2
otherwise	3

For each of the four interfaces, give the associated range of destination host addresses and the number of addresses in the range.

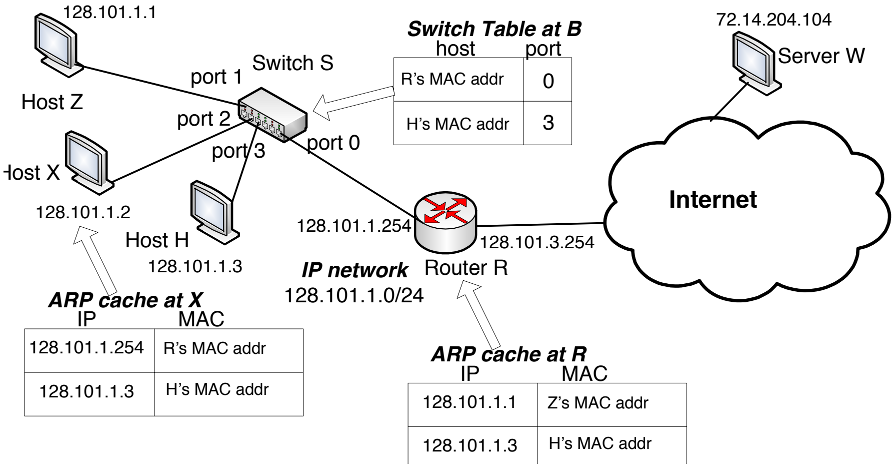
Answer

Destination Address Range	Link Interface
11000000 through 11011111	0
10000000 through 10111111	1
11100000 through 11111111	2
00000000 through 01111111	3

number of addresses for interface 0 = $2^5 = 32$
number of addresses for interface 1 = $2^6 = 64$
number of addresses for interface 2 = $2^5 = 32$
number of addresses for interface 3 = $2^7 = 128$

The first three rules covers the prefixes 1*, 10*, and 111*. Thus, the remaining prefix is 0*, which will be forwarded to interface 3. The next longest prefix is 111*, thus this range of IPs will be forwarded to interface 2. The second rule covers the prefixes 10*, which can be interpreted as the union of these two prefixes (100* and 101*). Now, among the three-bit prefixes, what is left is matched by rule 1 (due to the longest prefix matching), so rule 1 will cover the range of IPs matching the prefix 110*.

3. MAC Address, ARP, Switching and IP Forwarding (25 points. Approx. 30 minutes)



Switch Table at B

host	port
R's MAC addr	0
H's MAC addr	3

ARP cache at X

IP	MAC
128.101.1.254	R's MAC addr
128.101.1.3	H's MAC addr

ARP cache at R

IP	MAC
128.101.1.1	Z's MAC addr
128.101.1.3	H's MAC addr

Figure 1: Switched network

Consider the following switched network (Figure 1), where we have one Ethernet switch S, connecting three hosts, host Z on port 1, host X on port 2 and host H on port 3, as well as an IP router R (default router for the hosts) on port 0. These hosts lie on an IP network with the network prefix 128.101.1.0/24. The IP address for the interface of router R that is connected to switch S is 128.101.1.254. The IP addresses for hosts Z, X and H are shown in the figure. Furthermore, the *current* switch (forwarding) table at switch S and the ARP caches at host X and router R are also shown in the figure.

Answer

First of all, please note that the answers for this and other questions are very detailed so as to help you better understand the entire process the ARP protocol, switch functions, the interactions between IP and MAC layers, how IP datagram and Ethernet frames are forwarded, and so forth. Your answers do not need to be as detailed!

(a) (9 points) Suppose host X wants to send an IP datagram to host Z, and assume that host X knows the IP address of host Z (e.g., via DNS lookup).

i. (3 points) Briefly explain how host X obtains the MAC address of host Z. Describe how switch S handles the ARP request and response messages, and builds its switch table.

Answer

1. Using Z's IP address, host X looks up its ARP cache, finding no entry. It thus forms an ARP query message which asks for the MAC address of Z (using Z's

IP address), and "broadcasts" it by encapsulating the ARP query message in a broadcast Ethernet frame.

2. Switch S receives the broadcast Ethernet frame containing the ARP query message. It first attempts to locate the source MAC address in its switch table. When it finds no entry for X, it adds X into its switch table. Since this is a broadcast frame, it then forwards it to all ports except for the one it came from (ports 0, 1, and 3 NOT port 2).
3. The Ethernet layer of host Z receives the broadcast frame, decapsulates it and passes the ARP query to its ARP protocol, which learns that host X is querying for its MAC address. It thus sends an ARP response encapsulated in a unicast Ethernet frame to X. (Z also updates its own ARP cache, adding [X's MAC address, X's IP address]).
4. Upon arriving at switch S, S first attempts to lookup the source (Z's) MAC address of the ARP response in its switch table. Since there is no entry for Z, it adds Z into its switch table. It then looks up its switch table using the destination MAC address (X's MAC address) and finds it at port 2, thus it forwards the response message to port 2.

ii. (3 points) Will router R also receive the ARP request? If your answer is affirmative, what action does router R take? What about the ARP response message from host Z?

Answer

Yes, since the ARP request is encapsulated in a broadcast Ethernet frame, all devices on the same Ethernet (i.e., directly connected by Ethernet switches) will receive it – note that from the perspective of the Ethernet switch, there is no difference between a router and an end host. The Ethernet layer of router R passes the ARP request to its ARP protocol, which realizes that the ARP query is not for R, thus R discards the query. Before discarding it, however, router R first looks up its ARP cache using the source (Z's) MAC address; it finds Z's entry in its ARP cache, and it refreshes the TTL timer associated with the entry.

iii. (3 points) After host X learns the MAC address of host Z. Briefly describe how the IP datagram is delivered from host X to host Z, paying particular attention to the actions taken by switch S, and router R *if any*.

Answer

1. X constructs an IP packet where the source IP address is X's IP address and the destination IP address is Z's IP address; then X encapsulates this IP packet in an Ethernet frame, where the source MAC address is X's MAC address and the destination MAC address is Z's MAC address. X sends this Ethernet frame out.

2. Upon receiving the frame from port 2, switch S first looks up the source (X's) MAC address in its switch table. It finds X's entry and refreshes its TTL timer. Next S looks up the destination (Z's) MAC address in its switch table, and finds Z's entry. Lastly, S forwards the frame to port 1.
3. (The MAC layer of) Z receives and accepts the Ethernet frame, as from the destination MAC address, it knows that this frame is for itself. The Ethernet layer of host Z discovers that it's an IP datagram, decapsulates the datagram, and passes it to Z's IP layer.
4. Nothing happens at router R, since it will not see this frame at all.

(b) (7 points) Suppose host X sends an IP datagram to host H instead of host Z. Repeat the above questions.

Answer

1. Using H's IP address, host X looks up its ARP cache, and finds the entry for H. Host X then constructs an IP packet where the source IP address is X's IP address and the destination IP address is H's IP address. Then X encapsulates this IP packet into an Ethernet frame, where the source MAC address is X's MAC address and the destination MAC address is H's MAC address. Finally, X sends this Ethernet frame out.
2. Upon receiving the frame from port 2, switch S first looks up the source (X's) MAC address in its switch table. It finds X's entry and refreshes its TTL timer. It then looks up the destination (H's) MAC address in its switch table, and finds H's entry. It thus forwards the frame to port 3.
3. (The MAC layer of) H receives and accepts the Ethernet frame as, from the destination MAC address, it knows that this frame is for itself. The Ethernet layer of host H discovers that it's an IP datagram, decapsulates the datagram, and passes it to H's IP layer.
4. Note that again nothing happens at router R, since it will not see this frame at all.

(c) (9 points) Suppose now that host X wants to send an IP datagram to a remote server W outside the network. The IP address of server W is 72.14.204.104. Answer the following questions.

i. (3 points) Since server W's MAC address is not currently in its ARP cache, will host X issue an ARP request for server W? Briefly explain your answer.

Answer

No. This is because ARP query message (request) is only sent for a destination host that resides within the same IP network (subset) as the sending host – in such a case, an IP datagram must be delivered directly using the underlying layer-2 technology (in other words, no router is involved). By comparing its network prefix (128.101.1.0/24) with that of host W (by applying X's network mask to W's IP address), host X realizes that W resides on a different IP network, thus it should ask its default router, namely, router R, to help forward the IP datagram.

ii. (3 points) How does X know that it should forward the IP datagram to router R so that it can be delivered (via the Internet) to server W?

Answer

See the answer above.

iii. (3 points) Briefly describe how the IP datagram is delivered from host X to router R, paying in particular attention to the actions taken by switch S and router R. Moreover, please explicitly describe the source and destination IP and MAC addresses contained in the IP datagram and the encapsulating Ethernet frame.

Answer

1. In order to forward the IP datagram (destined to W) to router R, host X encapsulates the IP datagram (with X's IP address as the source IP address and W's IP address as the destination IP address) in an Ethernet frame where the source MAC address is X's and the destination MAC address is R's not W's! R's MAC address is found by looking up X's ARP cache.
2. Upon receiving the Ethernet frame, switch S looks up its switch table using the destination (R's) MAC address, finds the entry and forwards it to port 0. (It also uses the source MAC address to update its switch table entry for X.)
3. (The MAC layer of) router R receives and accepts the Ethernet frame, decapsulates it and passes it to its IP (network) layer. Using the destination IP address, router R looks up its routing table, and forwards it to an appropriate next-hop router to further deliver the frame (via the rest of the Internet) to the remote server W.

4. Virtual Circuits (15 points; 15 minutes)

a. (5 points) What is the key difference between virtual circuit and circuit switching? (one or two sentences should suffice!)

Answer

In the virtual circuit, there is no dedicated resource allocation at the call set-up time. In circuit switching, a physical path and resource are dedicated at the time of setting up connection between the source and the destination.

b. (10 points) Consider the network shown in Figure 2, where the numbers beside the links connecting hosts and routers represents the port numbers of the routers. Please write down the virtual circuit translation tables for *all the routers* after the following connections are established in the order given below. You can assume that the VCI assignment always picks the lowest unused VCI on an outgoing link.

(1) Host A connects to host K.

(2) Host B connects to host J.

(3) Host B connects to host D.

(4) Host D connects to host F.

(5) Host E connects to host L.

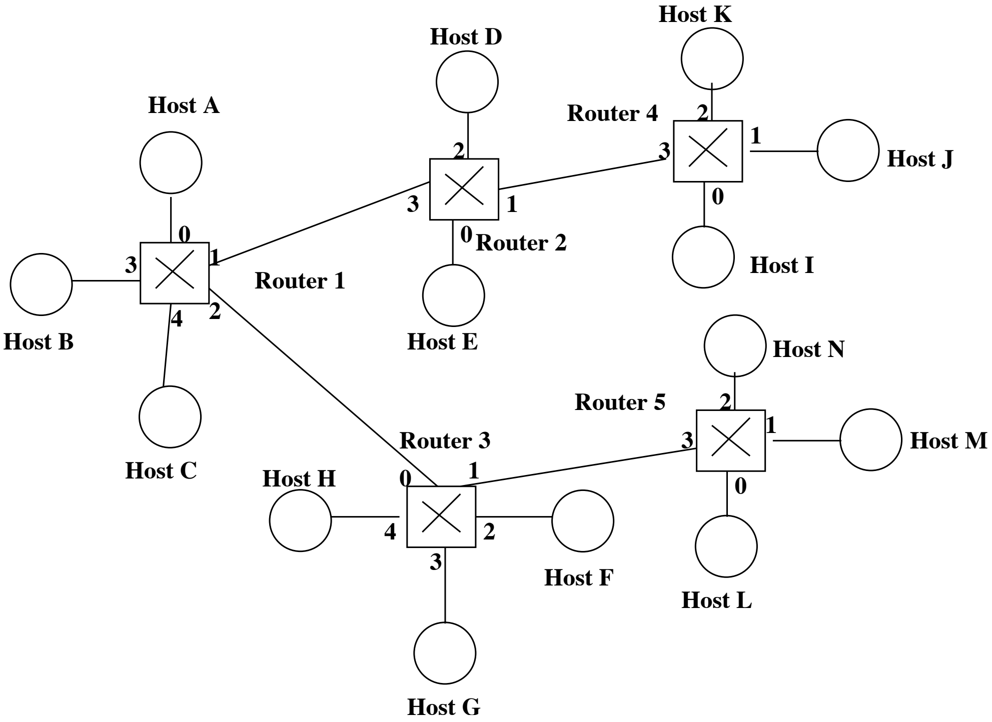


Figure 2: Figure for Question 4b.

Answer

(1) Host A connects to host K.

Router 1				Router 2				Router 3				Router 4				Router 5			
In		Out		In		Out		In		Out		In		Out		In		Out	
Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI
0	1	1	1	3	1	1	1					3	1	2	1				

(2) Host B connects to host J.

Router 1				Router 2				Router 3				Router 4				Router 5			
In		Out		In		Out		In		Out		In		Out		In		Out	
Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI
0	1	1	1	3	1	1	1					3	1	2	1				
3	1	1	2	3	2	1	2					3	2	1	1				

(3) Host B connects to host D.

Router 1				Router 2				Router 3				Router 4				Router 5			
In		Out		In		Out		In		Out		In		Out		In		Out	
Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI
0	1	1	1	3	1	1	1					3	1	2	1				
3	1	1	2	3	2	1	2					3	2	1	1				
3	2	1	3	3	3	2	1												

(4) Host D connects to host F.

Router 1				Router 2				Router 3				Router 4				Router 5			
In		Out		In		Out		In		Out		In		Out		In		Out	
Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI
0	1	1	1	3	1	1	1	0	1	2	1	3	1	2	1				
3	1	1	2	3	2	1	2					3	2	1	1				
3	2	1	3	3	3	2	1												
1	4	2	1	2	2	3	4												

(5) Host E connects to host L.

Router 1				Router 2				Router 3				Router 4				Router 5			
In		Out		In		Out		In		Out		In		Out		In		Out	
Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI	Port	VCI
0	1	1	1	3	1	1	1	0	1	2	1	3	1	2	1	3	1	0	1
3	1	1	2	3	2	1	2	0	2	1	1	3	2	1	1				
3	2	1	3	3	3	2	1												
1	4	2	1	2	2	3	4												
1	5	2	2	0	1	3	5												

5. Link State Routing (16 points; 25 minutes)

a. (10 points) Do Problem P.3 in Chapter 5 of the Textbook (*Kurose & Ross*, 7th Edition, p. 429).
In case you do not have the current version of the textbook, the problem is reproduced below for you. Consider the following network. With the indicated link costs, use Dijkstra's shortest-path algorithm to compute the shortest path from x to all network nodes. Show how the algorithm works by computing a table similar to Table 5.1.

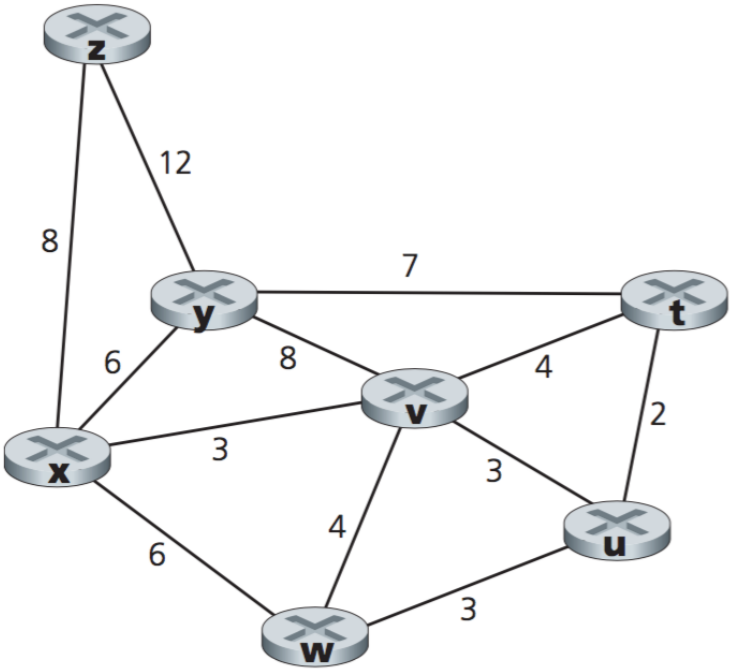


Figure 3: Figure for Question 5.

Answer

The shortest path from x to all network nodes.

Step	N'	D(t),p(t)	D(u),p(u)	D(v),p(v)	D(w),p(w)	D(y),p(y)	D(z),p(z)
0	x	∞	∞	**3,x**	6,x	6,x	8,x
1	xv	7,v	**6,v**		6,x	6,x	8,x
2	xvu	7,v			**6,x**	6,x	8,x
3	xvuw	7,v				**6,x**	8,x
4	xvuwy	**7,v**					8,x
5	xvuwyt						**8,x**
6	xvuwytz						

b. (6 points) Do Problem P.4 in Chapter 5 of the Textbook (*Kurose & Ross*, 7th Edition, p. 429).
In case you do not have the current version of the textbook, the problem is reproduced below for you. Consider the network shown in Figure 3. Using Dijkstra's algorithm, and showing your work using a table similar to Table 5.1, do the following:

(a) Compute the shortest path from t to all network nodes.

(b) Compute the shortest path from u to all network nodes.

(c) Compute the shortest path from v to all network nodes.

(d) Compute the shortest path from w to all network nodes.

(e) Compute the shortest path from y to all network nodes.

(f) Compute the shortest path from z to all network nodes.

Answer

(a) The shortest path from t to all network nodes.

Step	N'	D(x),p(x)	D(u),p(u)	D(v),p(v)	D(w),p(w)	D(y),p(y)	D(z),p(z)
0	t	∞	**2,t**	4,t	∞	7,t	∞
1	tu	∞		**4,t**	5,u	7,t	∞
2	tuv	7,v			**5,u**	7,t	∞
3	tuvw	**7,v**				7,t	∞
4	tuvwx					**7,t**	15,x
5	tuvwxy						**15,x**
6	tuvwxyz						

(b) The shortest path from u to all network nodes.

Step	N'	D(x),p(x)	D(t),p(t)	D(v),p(v)	D(w),p(w)	D(y),p(y)	D(z),p(z)
0	u	∞	**2,u**	3,u	3,u	∞	∞
1	ut	∞		**3,u**	3,u	9,t	∞
2	utv	6,v			**3,u**	9,t	∞
3	utvw	**6,v**				9,t	∞
4	utvwx					**9,t**	14,x
5	utvwxy						**14,x**
6	utvwxyz						

(c) The shortest path from v to all network nodes.

Step	N'	D(x),p(x)	D(u),p(u)	D(t),p(t)	D(w),p(w)	D(y),p(y)	D(z),p(z)
0	v	**3,v**	3,v	4,v	4,v	8,v	∞
1	vx		**3,v**	4,v	4,v	8,v	11,x
2	vxu			**4,v**	4,v	8,v	11,x
3	vxut				**4,v**	8,v	11,x
4	vxutw					**8,v**	11,x
5	vxutwy						**11,x**
6	vxutwyz						

(d) The shortest path from w to all network nodes.

Step	N'	D(x),p(x)	D(u),p(u)	D(v),p(v)	D(t),p(t)	D(y),p(y)	D(z),p(z)
0	w	6,w	**3,w**	4,w	∞	∞	∞
1	wu	6,w		**4,w**	5,u	∞	∞
2	wuv	6,w			**5,u**	12,v	∞
3	wuvt	**6,w**				12,v	∞
4	wuvtx					**12,v**	14,x
5	wuvtxy						**14,x**
6	wuvtxyz						

(e) The shortest path from y to all network nodes.

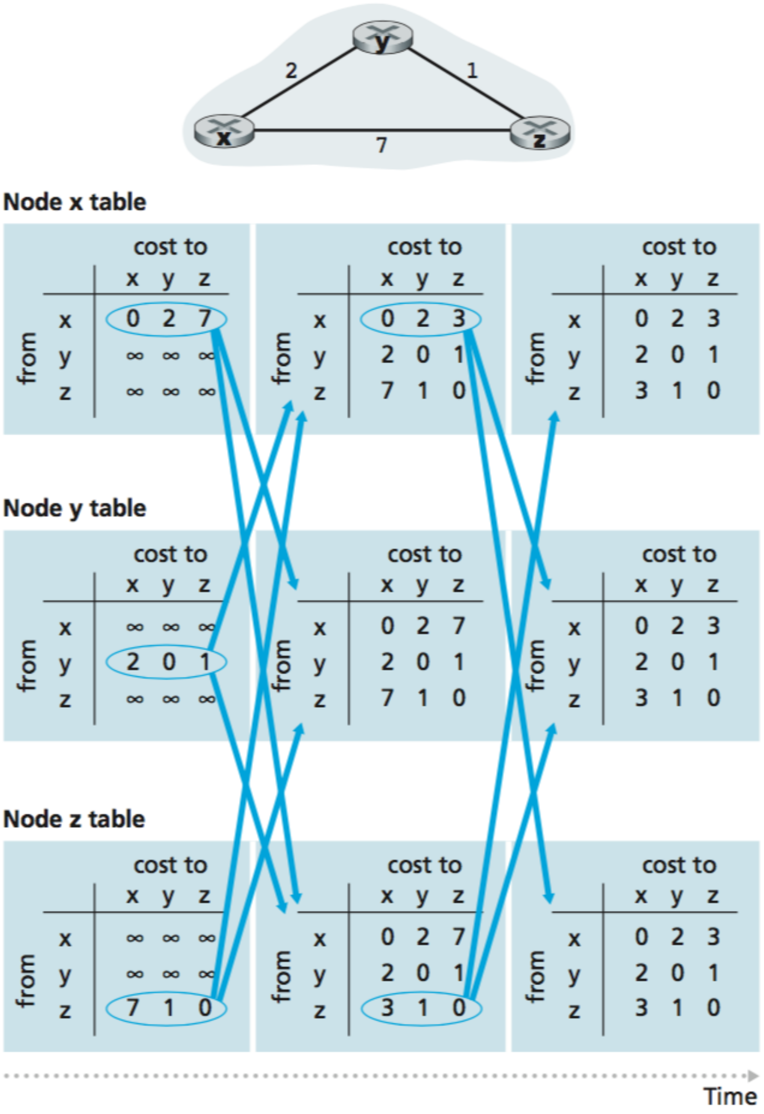
Step	N'	D(x),p(x)	D(u),p(u)	D(v),p(v)	D(w),p(w)	D(t),p(t)	D(z),p(z)
0	y	**6,y**	∞	8,y	∞	7,y	12,y
1	yx		∞	8,y	12,x	**7,y**	12,y
2	yxt		9,t	**8,y**	12,x		12,y
3	yxtv		**9,t**		12,x		12,y
4	yxtvu				**12,x**		12,y
5	yxtvuw						**12,y**
6	yxtvuwz						

(f) The shortest path from z to all network nodes.

Step	N'	D(x),p(x)	D(u),p(u)	D(v),p(v)	D(w),p(w)	D(y),p(y)	D(t),p(t)
0	z	**8,z**	∞	∞	∞	12,z	∞
1	zx		∞	**11,x**	14,x	12,z	∞
2	zxv		14,v		14,x	**12,z**	15,v
3	zxvy		**14,v**		14,x		15,v
4	zxvyu				**14,x**		15,y
5	zxvyuw						**15,y**
6	zxvyuwt						

6. Distance Vector Routing (20 points; 30 minutes)

a. (10 points) Do Problem P.8 in Chapter 5 of the Textbook (*Kurose & Ross*, 7th Edition, pp. 430.
In case you do not have the current version of the textbook, the problem is reproduced below for you. Consider the three-node topology shown in Figure 5.6. Rather than having the link costs shown in Figure 5.6, the link costs are $c(x, y) = 3$, $c(y, z) = 6$, $c(z, x) = 4$. Compute the distance tables after the initialization step and after each iteration of a synchronous version of the distance-vector algorithm (as we did in our earlier discussion of Figure 5.6)



Node x table

from	x	y	z
x	0	2	7
y	∞	∞	∞
z	∞	∞	∞

from	x	y	z
x	0	2	3
y	2	0	1
z	7	1	0

from	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

Node y table

from	x	y	z
x	∞	∞	∞
y	2	0	1
z	∞	∞	∞

from	x	y	z
x	0	2	7
y	2	0	1
z	7	1	0

from	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

Node z table

from	x	y	z
x	∞	∞	∞
y	∞	∞	∞
z	7	1	0

from	x	y	z
x	0	2	7
y	2	0	1
z	3	1	0

from	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

Figure 4: Figure 5.6 Distance-vector (DV) algorithm in operation

Answer

Node x table

		cost to		
		x	y	z
from	x	0	3	4
	y	∞	∞	∞
	z	∞	∞	∞

		cost to		
		x	y	z
from	x	0	3	4
	y	3	0	6
	z	4	6	0

Node y table

		cost to		
		x	y	z
from	x	∞	∞	∞
	y	3	0	6
	z	∞	∞	∞

		cost to		
		x	y	z
from	x	0	3	4
	y	3	0	6
	z	4	6	0

Node z table

		cost to		
		x	y	z
from	x	∞	∞	∞
	y	∞	∞	∞
	z	4	6	0

		cost to		
		x	y	z
from	x	0	3	4
	y	3	0	6
	z	4	6	0

b. (5 points) What is the “count-to-infinity” problem?

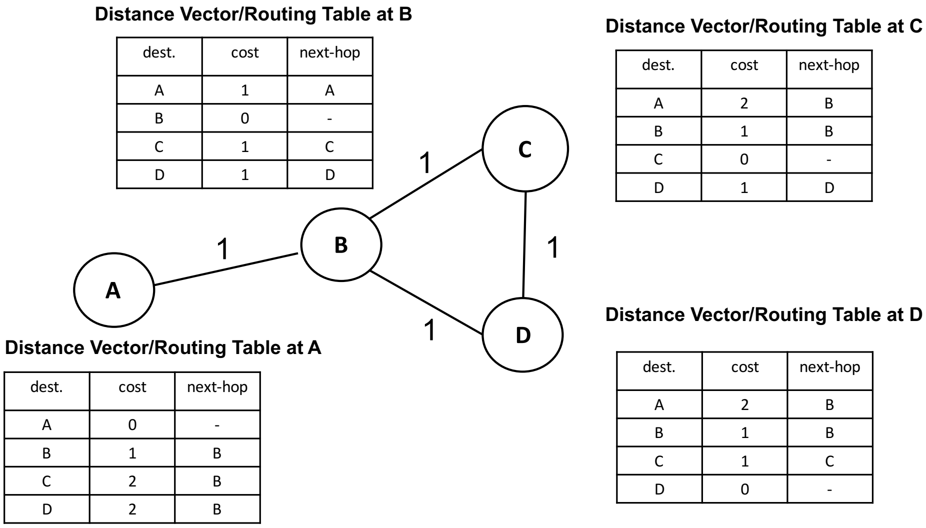
Answer

The count-to-infinity problem refers to a problem of distance vector routing. The problem means that it takes a long time for a distance vector routing algorithm to converge when there is a link cost increase. For example, consider a network of three nodes x, y, and z. Suppose initially the link costs are c(x,y)=4, c(x,z)=50, and c(y,z)=1. The result of distance-vector routing algorithm says that z's path to x is $z \rightarrow y \rightarrow x$ and the cost is 5(=4+1). When the cost of link (x,y) increases from 4 to 60, it will take 44 iterations of running the distance-vector routing algorithm for node z to realize that its new least-cost path to x is via its direct link to x, and hence y will also realize its least-cost path to x is via z.

c. (5 points) Provide an example to show that why the "poisoned reverse" is a hack, in other words, it does not always work!

Answer

Consider the network in Figure 5, where the cost of the links are given above the corresponding links. The network uses distance vector routing algorithm to compute routing tables. The distance vector/routing tables at routers A, B, C, and D have been computed for you, as shown in the figure.



Distance Vector/Routing Table at B

dest.	cost	next-hop
A	1	A
B	0	-
C	1	C
D	1	D

Distance Vector/Routing Table at C

dest.	cost	next-hop
A	2	B
B	1	B
C	0	-
D	1	D

Distance Vector/Routing Table at D

dest.	cost	next-hop
A	2	B
B	1	B
C	1	C
D	0	-

Distance Vector/Routing Table at A

dest.	cost	next-hop
A	0	-
B	1	B
C	2	B
D	2	B

Figure 5: Figure for Question 6-c: *Split-horizon* with *poisonous reverse.*

Suppose now the link between A and B goes down. Nodes B, C, and D will get into an infinite loop trying to reach node A, which is now disconnected from the three nodes.

The *split-horizon* hack is a way of fixing the count-to-infinity problem. It sets a router to never advertise the cost of a destination to a neighbor, *if it uses the neighbor as the next-hop to reach the destination.* Unfortunately, the *split-horizon* hack won't fix this problem thats why is called a hack; it can only fix the "county-to-infinity" problem when the loop only involves two nodes. When the loop involves three or more nodes, it breaks down.

Note that the *split-horizon* with *poisonous reverse* hack adds a "poisonous reverse" as follows: if X routes to Z via Y, then X tells Y that its distance to Z is infinity (instead of just not telling it anything) adding poisonous reverse speeds up the convergence.

Assuming that the split horizon hack with poisonous reverse is used, then after the next round of routing information is exchanged among routers B, C and D, the updated distance vector/routing tables (with respect to the destination A) at routers B, C and D look like the following:

Router	**Destination**	**Cost**	**Next-Hop**
B:	A	∞	-
C:	A	3	D
D:	A	3	C

You can notice that in the next round, B, C, and D would still get into a loop trying to reach node A by using each other as their next hop and incrementing their cost till infinity.

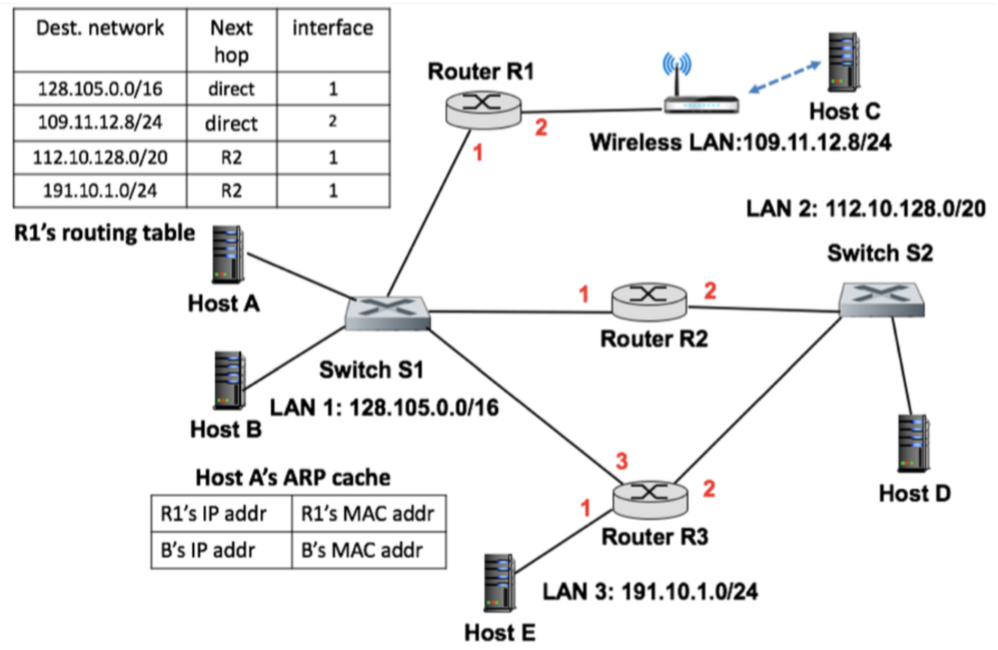
Unfortunately, the *split-horizon* with *poisonous reverse* hack will not fix this problem also. Just as in the case of the *split-horizon* hack, it breaks down when the loop involves three or more nodes.

There is another *hack* that can potentially address this problem using the so-called *hold-down timer*. Again it's a hack and can break down depending on the size of the loop and the value of the timer used. If you are interested, look it up by yourself.

So you ask: Is there a real *solution* to the "count-to-infinity" problem? Yes, there is. But it is very convoluted, and is implemented in the Cisco EIGRP protocol. If you are interested in a pointer to the solution, please ask Prof. Zhang!

7. (*Optional* Bonus Question) IP Forwarding, Default Router and ICMP Redirect (20 points. Approx. 30 minutes)

Refer to Figure 6 and answer the following questions as concisely as you can. In answering these questions, you'll need to google and look up relevant information on *ICMP Redirect*, e.g., on *wikipedia*.



Dest. network	Next hop	interface
128.105.0.0/16	direct	1
109.11.12.8/24	direct	2
112.10.128.0/20	R2	1
191.10.1.0/24	R2	1

R1's routing table

Host A's ARP cache

R1's IP addr	R1's MAC addr
B's IP addr	B's MAC addr

Figure 6: Figure for Bonus Question 7: IP Forwarding, Default Router and ICMP Re-Direct.

a. (4 points) Consider the following scenario: host A wants to send an IP packet to host D, which is connected to LAN 2 (IP network prefix: 112.10.128.0/20). Since host D is *not* on the same IP network as host A. Hence host A will send it to its default gateway router R1. Describe the actions taken by router R1 (and subsequently at router R2) at *both* the network layer and data link layer for sending this IP packet to host D. You can assume that R1 has the MAC address of R2, and R2 has the MAC address of host D.

Answer

R1 matches D's IP address against its routing table, and decides that the packet has to be forwarded to R2 on interface 1. R1 then looks up R2's IP address and finds that it has the same network prefix as R1 does and is thus on the same network. R1 knows R2's MAC address, therefore it sends a frame destined to R2's MAC address using a unicast frame. R1 will then send an ICMP redirect message to host A telling it that it should contact R2 directly if it needs to contact the network prefix 112.10.128.0/20. R2 receives the frame from R1, strips out the data link header and finds that the destination is host D. From R2's routing table, it knows that it should forward the packet directly on interface 2. R2 checks its ARP table and already has D's MAC address. Finally, it adds its own header information with the whole packet destined for D's MAC address. Note: An ICMP redirect is an error message sent by a router to the sender of an IP packet. Redirects are used when a router believes a packet is being routed sub optimally and it would like to inform the sending host that it should forward subsequent packets to that same destination through a different gateway.

b. (4 points) Continue the scenario in problem **7.a**: suppose host A wants to send *another* IP packet to host D. Will host A still send it to router R1? Or will it send it to router R2? (Hint: if you did not mention `ICMP Redirect` message in your answer to problem **7.a**, your answer is *not complete].*) Since host A does not have router R2's MAC address. Describe how host A obtains R2's MAC address using the ARP protocol.

Answer

If host A needs to send another packet to host D, it will not send it to router R1 again if it received the ICMP redirect message from R1. It will thus send the packet to R2. Since host A does not know R2's MAC address but knows R2's IP address (through the ICMP redirect message from R1), it will broadcast an ARP request message with the destination MAC address filled with all 1s (FF:FF:FF:FF:FF:FF). The destination IP address is that of R2, and the source IP and MAC addresss belong to A. This packet will be received by every host on LAN1, but only R2 will accept it (the network layer of R2 will find that the destination IP address matches its own IP address). R2 will then send an ARP reply message with destination MAC address A and source MAC address R2. Only A will receive this packet since it is destined for it. A will then know R2's MAC address and update its ARP table.

c. (4 points) Consider yet another scenario: suppose host A now wants to send an IP packet to host E, which is connected to LAN 3 (IP network prefix: 191.10.1.0/24). Host A will send it to its default gateway router R1, which will send it to router R2, based on its current routing table. Now let's assume that, based on the routing table at router R2 (not shown), it will send the packet to router R3 via its interface 1, namely, back to the same interface where it comes from. Will router R2 send an `ICMP Redirect` message to *router R1*, telling it to use R3 for destination network prefix 191.10.1.0/24? (Hint: `ICMP` messages are *always* sent to the source IP address of an IP packet.) More basically, can the *network layer* at router R2 tell whether router R1 or host A sends the packet to it? Briefly explain your answer.

Answer

R2 will send an ICMP redirect message to host A, and NOT to router R1. This is because ICMP is only sent from router to host or vice versa. R2 will still send the current packet to R3, so that A does not have to resend the packet.

d. (4 points) Continue the scenario in problem **7.c**: how will router R1 eventually knows that it can reach destination network prefix 191.10.1.0/24 directly via router R3 instead of router R2? Namely, changing the next hop from R2 to R3?

Answer

This is done through network routing protocols, such as link state or distance vector. Eventually, R1 will have the optimal route to R3.

e. (4 points) What might have happened that caused R1 to have a "non-optimal" route entry that has router R2 as the next-hop to the destination network prefix 191.10.1.0/24 instead of router R3 in the first place?

Answer

One reason is that Router R3 went down and just came back up. Another reason is that the link S1-R3 went down and was fixed. Yet another reason was that R2 was advertising a shorter route to R3 than R3 was, but this changed.