EE 122 Midterm Solutions

March 14, 2006, 12:40 – 2pm

Spring 2006

Prof. Abhay Parekh
Prof. David Tse

Name:
SID:
Please put your name and SID on every page

There are 6 multiple choice questions and 4 regular questions in this exam. Answer all questions carefully. The maximum score is 110%. You have 1 hour and 20 minutes. Good luck!

Multiple Choice Questions [20%]

Circle one or more of the choices for each question in this section. There can be more than one correct choice. Each correct answer will get +1 point. Each incorrect answer will get -1 point. If you get everything correct, you will get the full 20% . If your total points is negative, you will get 0%. Anything in between is scaled appropriately.

1. Packets of two classes arrive at a switch at the rate of 10 pkt/s and 20 pkt/s. Packets of class 1 leave the switch exactly 1 s after they arrive. Packets of class 2 leave the switch exactly T s after they arrive. The average number of total packets in the switch is observed to be 50 packets.

Which of these statements are right?
a) Average packets of class 1 in switch = 10
b) Average packets of class 2 in switch = 20
c) Value of T = 2
d) Value of T = 4
ANSWER: A & C

2. The advantage(s) of the best-effort service model of the Internet is/are:

a) It makes the Internet more scalable.
b) It provides quality-of-service guarantees because the best effort is made for all applications.
c) Even UDP would work well since the network provides reliable service.
ANSWER: A

3. Which of the following statements are true?

a) Most DNS queries on the internet are resolved iteratively rather than recursively.

b) DNS is a critical infrastructure of the internet and so runs on TCP

c) HTTP 1.0 and BGP are examples of stateless protocols

d) HTTP is much faster when run over UDP

e) RTP can run over TCP or UDP

ANSWER: A & E

4. Which of the following statements is true?

a) Flow control in TCP requires feedback from the receiver.

b) Congestion control in TCP requires explicit feedback from network routers.

c) The greater the round trip time of the connection, more rapid is the linear rise of the congestion window.

ANSWER: A

5. Which statements are true?

a) Routing algorithms run at the network layer on the routers.

b) In the Internet, the transport layer header of the packet remains the same from source till destination.

c) A set of 200 nodes can have a common IP address (32 bits) prefix greater than 23 bits.

d) A TCP connection can be considered to be a virtual circuit (VC) with fixed VC number, namely the IP Address.

ANSWER B & C

6. Which of the following statements are true?

a) TCP is always better than UDP for an application except for the fact that the overhead for TCP is larger.

b) UDP error detection mechanism can detect all errors but cannot correct them.

c) The throughput of a stop-and-go protocol is almost always limited by the round trip time rather than the actual link capacity available.

d) A stop-and-go protocol does not need sequence numbers for the packets since only one packet is "in-flight" at all times.

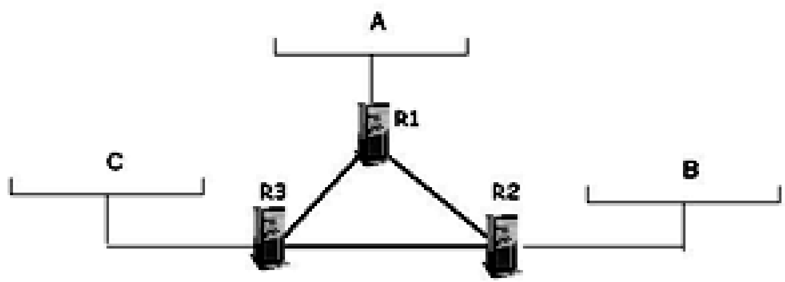
e) One advantage of Go-back-N over the selective repeat protocol is that there is no need to keep track of any state information at the receiver.

ANSWER C & E

Regular questions [80%]

Please explain your answers. Yes/no answers without explanations will get no points.

[10%] 7. Consider the network below with three routers, each connected to an ethernet network. Each router has three interfaces.



Assign IP addresses to the six subnets in the figure from the block 255.255.255/23. Ethernets A and B should have enough addresses to support 120 interfaces, and C requires enough to support 250 interfaces. Make sure to assign a sufficient number of addresses to the remaining three subnets as well.

SOLUTION

First note that the remaining 3 subnets are the networks formed by each pair of routers. So, we get the remaining 3 subnets as $\{R1,R2\}\{R1,R3\}$ $\{R2,R3.\}$ Call these subnets *D*, *E* and *F* respectively. So, each of the subnets *D*, *E* and *F* needs at least 2 IP addresses each. So, a possible allocation scheme is:

Subnet	IP address range	Number of addresses
A	255.255.255.0/25	128
B	255.255.255.128/25 − 255.255.255.0/29	128 − 8 = 120
C	255.255.254/24	256
D	255.255.255.0/31	2
E	255.255.255.2/31	2
F	255.255.255.4/30	4.

[25%]8. In this problem we analyze the case of two applications sharing a 4 Mb/s link. All packets in this problem are 1000 bits long. Application A is video traffic which arrives at the link at a constant rate of 2Mb/s. Application B is highly bursty. For simplicity, assume that all the packets of a burst arrive simultaneously. Also assume that both applications A and B last indefinitely and that propagation delays are negligible.

Suppose that the scheduler at the link is First Come First Serve, and that the scheduler serves all the packets of a burst from Application B back-to-back. Assume that the bursts from application B arrive at times 0,5,20,25,40,45 etc. and are 10Mb, 30Mb, 10Mb, 30Mb, 10Mb, 30Mb etc in size.

[15%] a) Calculate the following for Application A:

- [4%](i) The average number of packets in queue
- [4%](ii) The average packet delay
- [4%](iii) The maximum packet delay
- [3%](iv) Assume that this is the only link traversed by application A. What should be the size of the playback buffer at the receiver?

[10%] b) Assume that the scheduler is round robin, i.e. after serving a packet from one application it serves a packet from the other application if one is the buffer at that time. Calculate the same quantities as in the previous parts the question under this assumption.

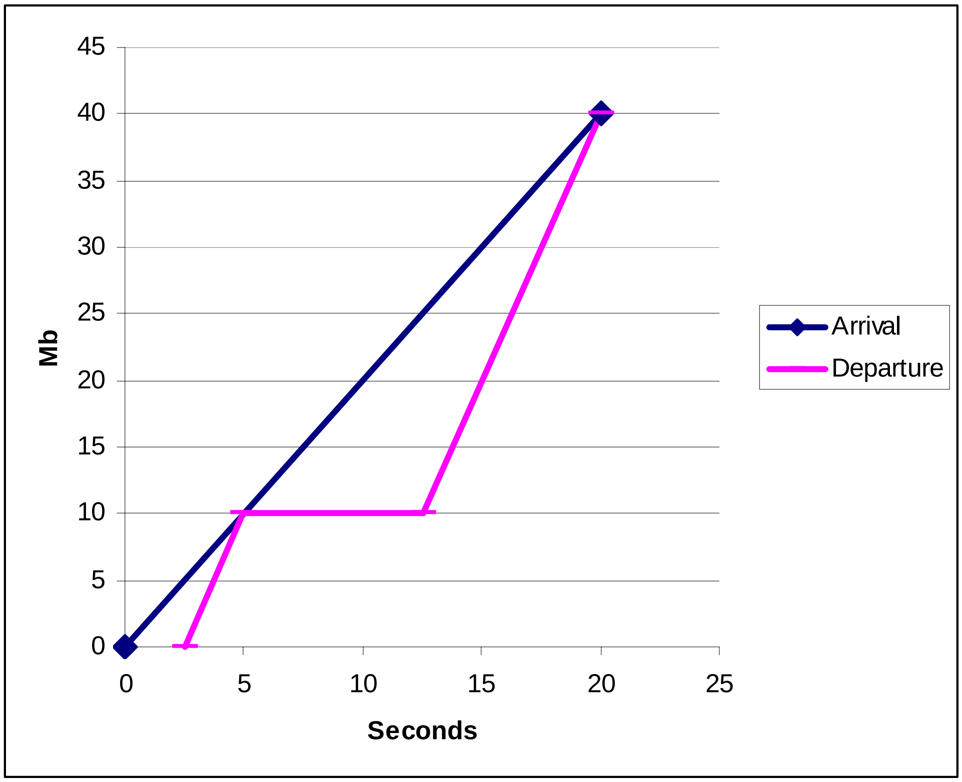
SOLUTION

In this problem it is important to carefully draw the arrival and departure curves for the Application A as we did in class in the proof of Little's Law:

Part A

The the packet sizes are small relative to the link speeds, it is ok to work in terms of bits and then convert to packets later. Then as indicated in the figure, the arrival curve is a line of slope 2Mb/s.

To calculate the departure curve note that whenever an Application B burst of size 10Mb arrives, it will take 2.5s to serve and when a burst of size 30Mb arrives it will take 7.5s to serve. At time 2.5s there are 5Mb of Application A in queue. How many will be in queue by the time next Application B burst arrives at time 5s? An additional 5Mb will arrive, and the link can serve 10Mb, so when the burst arrives there will be 0 Application A bits in queue! The link will serve the 30Mb Application B burst for 7.5s, during which 15Mb of Application A traffic will be queued. Again, calculating the number of Application A bits left when the next Application B burst arrives (at time 20s), we see that there none left. This gives enough data to draw the departure curve for Application A:



The pattern represented in the figure is periodic so we just need to consider what happens over the interval [0,20s].

(i) **To find the average number of bits in queue, just add up the areas of the two triangles and divide by 20. That yields 6.25 Mb which is 6250 packets.**
(ii) **To find the average packet delay, we merely divide the areas of the triangles by the number of packets which have arrived (remember the proof of Little's Law). Since 40,000 packets would have arrived in the interval [0,20], we get 3.125s**
(iii) **This is just the maximum horizontal distance between the two curves which 7.5s**
(iv) **Since the maximum delay of a packet is 7.5s the receiver needs to buffer 7.5*2Mb/s packets, i.e. 15Mb.**

Part B

In the round robin case Application A always gets a service rate of at least 2Mb/s (i.e. half of the link capacity) whenever it has packets in queue. Thus, it will always get service immediately upon arrival or after waiting for one Application B's packets to have completed service.

(i) **The average number of packets in the queue is 1 but an answer of 0 is acceptable**
(ii) **The average packet delay is one packet transmission time or 0.25 ms**

(iii) **0.25ms**
(iv) **1000 bits**

[35%] 9. Suppose two TCP connections A and B are sharing a link of capacity 10 Mbps. They both have the same round trip time (RTT) of 100 ms and the same segment size of 1 Kbits. They both follow an additive increase and multiplicative decrease congestion control algorithm, increasing the window size by one segment every RTT and halving the window size whenever they detect loss. You can ignore the slow start phase in answering the questions below.

[8%] a) Suppose initially flow A is sending at rate of 4 Mbps while B is sending at a rate of 2 Mbps. By drawing a carefully labeled 2-D plot of the evolution of their rates with time or otherwise, argue that in the long run they will each get the same share of the link capacity. Hence, TCP is fair.

[8%] b) What is the long run throughput each connection gets? (You can assume, for this part only, that loss is immediately detected when the aggregate transmission rate of the two connections exceed the link capacity.) How does this performance compare to that under an explicit rate congestion control algorithm where the available bandwidth is explicitly allocated to the connections?

[6%] c) Suppose the window size is reduced by one third instead of halved every time there is a packet loss. Is the congestion control algorithm still fair? Explain.

[6%] d) Bob has taken EECS 122 and decides that a multiplicative decrease is too drastic, resulting in too much loss in throughput. Instead, he suggests that the decrease should be additive as well, decreasing the window size by 1 segment size every round trip time until there is no loss. Is this algorithm fair, with the initial rates as in part (a)? Justify your answer.

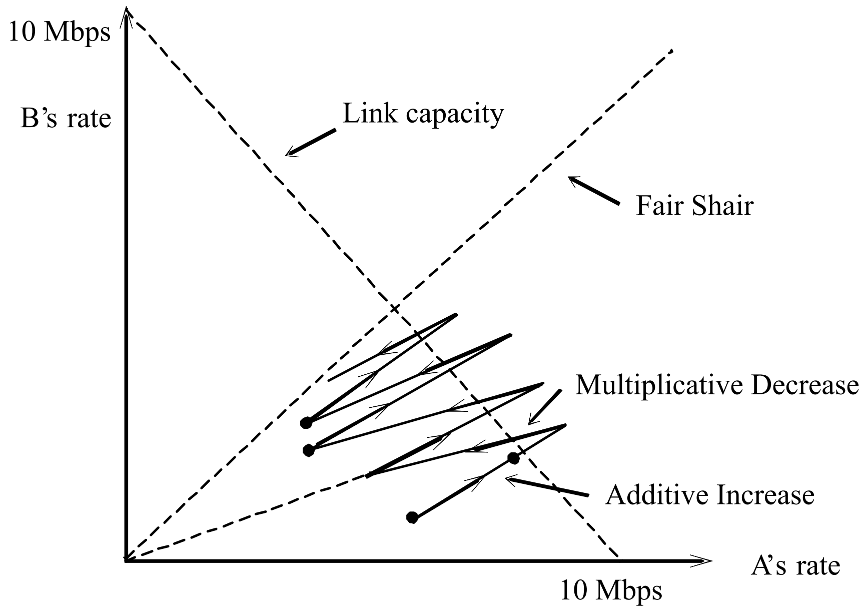
[7%] e) Consider again the basic additive increase multiplicative decrease algorithm, but now the RTT of A is 200 ms and the RTT of B is 100ms. What is the ratio of the long run throughputs of the two connections? Is TCP still fair? Justify your answer.

SOLUTION

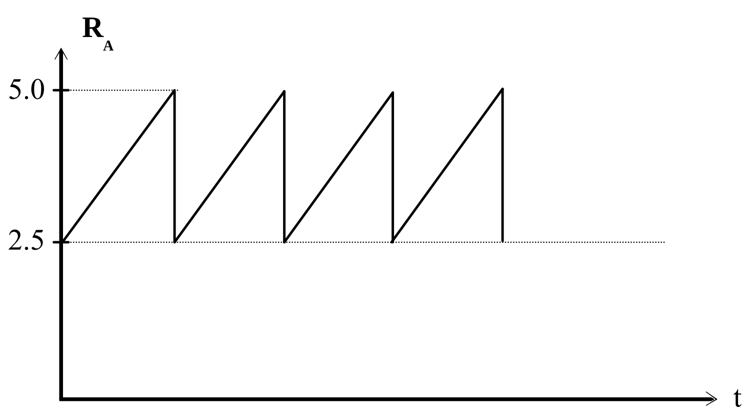
9 a)

In the addition increase phase, the gap between the transmission rates of A and B remains unchanged. Every time this is loss the gap is reduced by $\frac{1}{2}$ **So often N such**

losses event, the gap is reduced by $\left(\frac{1}{2}\right)^N$. As $N \to \infty$, the gap goes to zero. This is also shown in the following figure:



9 b)

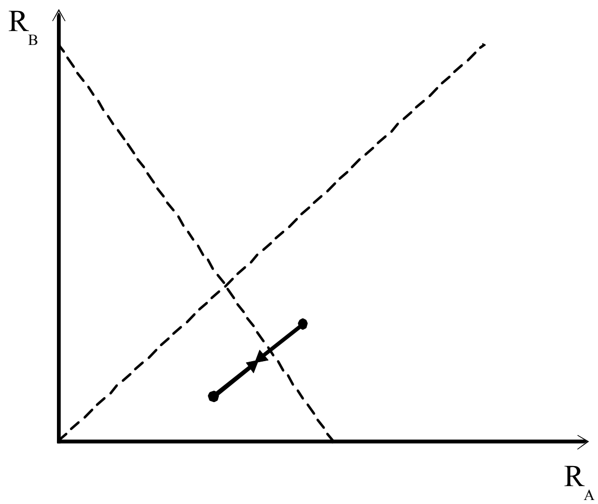


$$\text{Average rate} = \frac{5+2.5}{2} = 3.75Mbps$$

9 c) Yes, the argument in (a) does not depend on the factor of $\frac{1}{2}$; any constant factor $\alpha < 1$ would work.

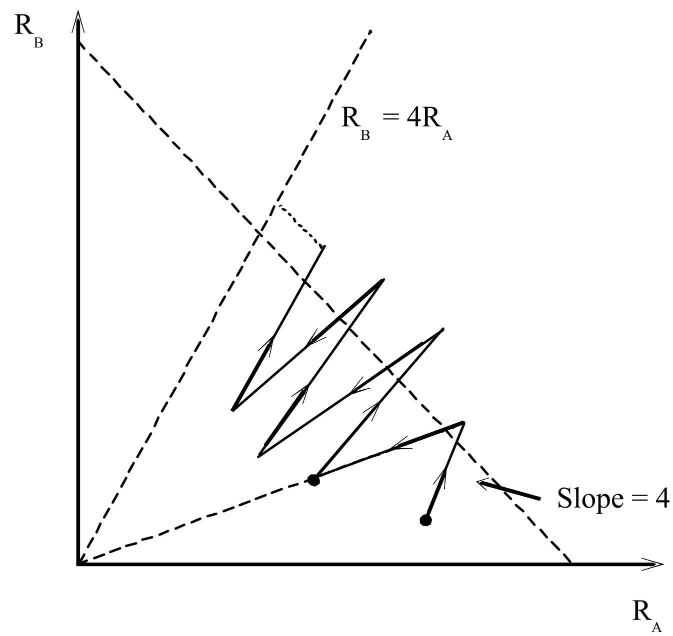
9 d)

No, the gap between the users' rates remains constant.

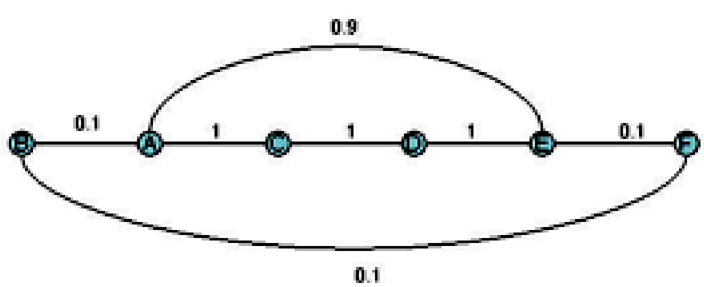


9 e)

The window of A increases twice as slowly as B's. The rate of A is equal to $\frac{W}{RTT}$, so the rate of A increases $\frac{1}{4}$ as slowly as rate the of B. Asymptotically, throughput of A is $\frac{1}{4}$ that of B. No, TCP is not fair.



[20%] 10. In this problem we will compute distance vectors using the Bellman Ford algorithm for the six node graph given below:



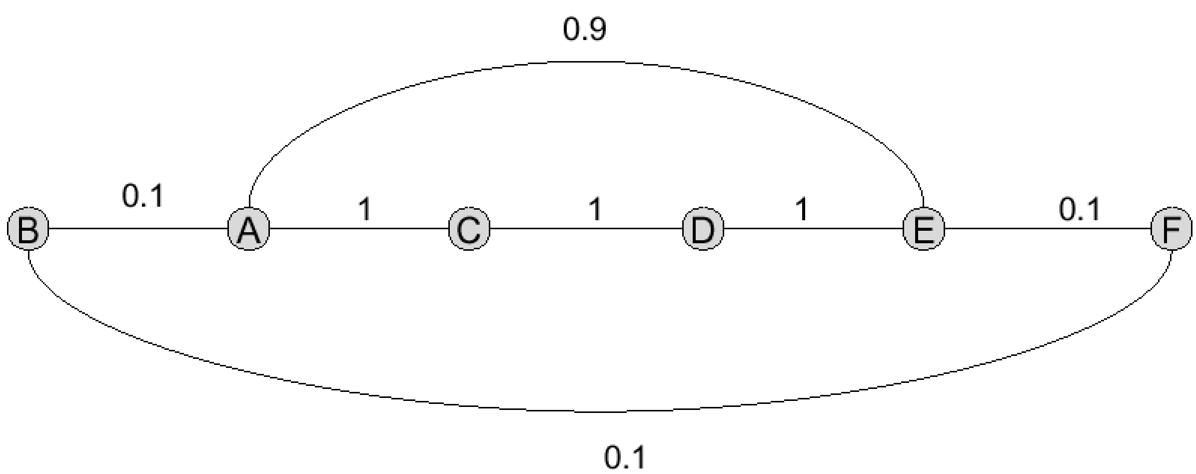
(*) Assume that time is slotted (t=1,2,3...) and that a node sends its distance vector estimates to its neighbors at the beginning of each slot. A distance vector estimate sent at the beginning of a slot arrives by the end of that slot. All distance estimates are computed using the most recently available estimates.

What are node A's distance vectors at the beginning of the time slots 1,2,3,4:
[10%] (a) For the problem as stated in (*)

[10%] (b) For the problem as stated in (*) except that link BF is so congested that it takes two timeslots to traverse, i.e. a distance vector from either B or F sent over the link at the beginning of time slot j, is only received at the other end of the link at the end time slot j+1. All of the nodes and the rest of the links behave as in (*).

SOLUTION

In the six node network shown below the nodes compute shortest path distances using the Bellman Ford distance vector method.



(a) The problem is as stated in (*)

In iteration h, node A knows h-hop shortest distances to each of the other nodes.

	A	B	C	D	E	F
1	0	0.1	1	∞	0.9	∞
2	0	0.1	1	1.9	0.9	0.2
3	0	0.1	1	1.9	0.3	0.2
4	0	0.1	1	1.3	0.3	0.2

(b) The problem is as stated in (*) except that the link (B, F) has become so congested that it takes **two timeslots** to traverse, i.e. a distance vector from either B or F sent over the link at the beginning of time slot j, is only received at the other end of the link at the end time slot j+1. All of the nodes and the rest of the links behave as in (*).

The estimates received by A from B in time slots 1 and 2 do not reflect any information sent by F. However, since node A does hear from B in time slots 1 and 2, it can correctly compute the shortest path to F. At the end of time slot 2, node B receives the time slot 1 update of node F. Thus, node A can correctly compute the shortest path to E for the estimate of time slot 4, but it will take one more iteration to correctly compute the shortest path to D.

	A	B	C	D	E	F
1	0	0.1	1	∞	0.9	∞
2	0	0.1	1	1.9	0.9	0.2
3	0	0.1	1	1.9	0.9	0.2
4	0	0.1	1	1.9	0.3	0.2