UCLA CS 118 Computer Networks

Second Practice Final Exam: no solutions will be posted

Directions: Write your name on the exam. Write something for every question. You will get some points if you attempt a solution but nothing for a blank sheet of paper. Problems take long to read but can be answered concisely.

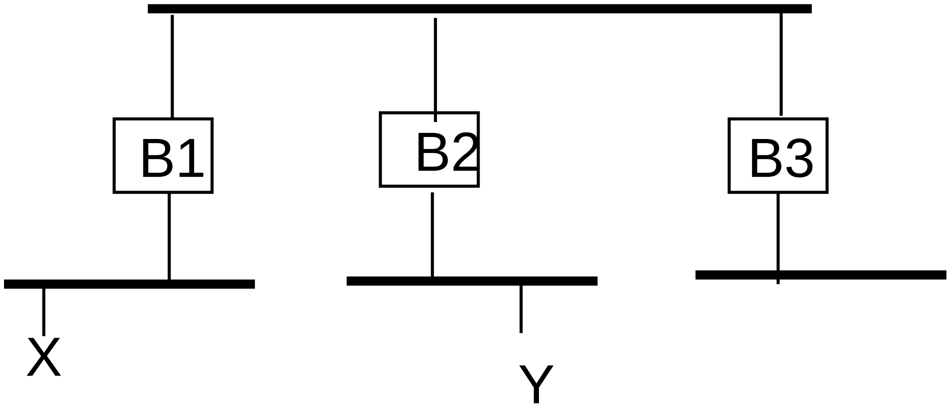
1, Fundamentals, 20 points total, 2 points each: Give short 1 line answers for each question. If you are asked to give a reason for something, give the most important reason you can think of.

- **Framing using 4-5 encoding:** Why is bit stuffing is not used today (hint: codes like 4-5 and 8-10 are so common today.)
- **Hierarchical Addressing:** Why medium size organizations are given multiple consecutive Class C addresses. Why multiple and why are they consecutive.
- **Bridges versus Routers:** Why its a good idea for bridges to be able to process packets at LAN rates while it is OK for routers to process packets at much slower rates.
- **Route Computation:** One reason why link state routing is preferable to distance vector style routing.
- **Link State Routing:** Why link state routing depends on a primitive flooding protocol to send LSPs through the network instead of using the existing routing table to send LSPs.
- **Peering:** Why ISPs peer with each other though no money is exchanged.
- **DNS:** Why is the Domain Name Service a good idea?
- **BGP:** Why BGP uses a path vector instead of a distance vector protocol
- **Fragmentation:** Why the Packet ID field in the IP header is mostly useless today.

- **Transport:** Why the sequence number space for a reliable data link on a point to point link need only be twice the window size, while transport sequence number spaces must be much larger than twice the window size.

- **TCP Congestion Control and RED:** Why routers implementing RED can drop a perfectly good packet even if there is space in the router buffers to store the packet.

2. Bridging and Pruned Multicast, 10 points: In class, we learnt about IP multicasting and how the multicast trees were "pruned" so that copies of multicast packets were not sent to LANs in which no station was listening to that multicast. We want to offer a similar "pruned multicast" service for Extended LANs by modifying bridges and endnodes slightly. The figure shows two stations X and Y. Assume that initially only X is in group G (G is a multicast data link address). Then when X sends a multicast packet to G bridge $B1$ should (ideally) not forward the packet. Later, assume that Y also joins group G. All bridges should learn this fact. Now when X sends a packet to G, bridge $B1$ should forward the packet on the upper LAN, bridge $B2$ should pick up the packet and forward it to Y's LAN, and bridge $B3$ should not forward the packet.



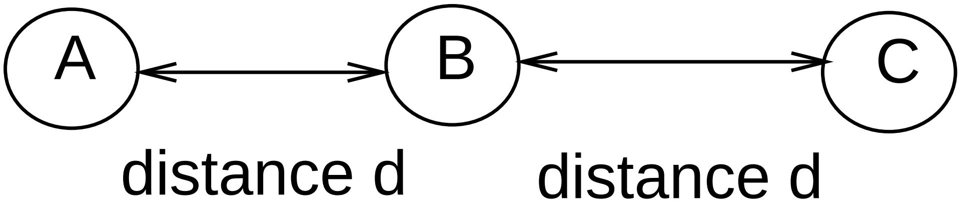
Thus bridges need to learn all the directions through which a multicast address like G is reachable. To allow bridges to learn, we modify the endnodes that participate in a group G to periodically send a multicast packet to some "All Bridges Address". However, the endnodes put G as the *source address* in such a packet.

- How would you modify the bridge learning algorithm to learn about multicast addresses like G based on the periodic updates sent by the endnodes participating in the group?

- Explain what each bridge learns in the above example. Assume that initially only X sends updates and show what each bridge learns. Then show what happens when Y joins the group and begins sending updates.
- Suppose that Y crashes and stops sending updates. X continues to send updates. Explain how bridges timeout information asssociated with multicast addresses.
- Not all stations participate in this protocol. What should a bridge do when it gets a multicast address that it has no learnt information about?

3. Mobile Stations and Radio LANs, 15 points: In this exam we will consider some of the issues concerning mobility at the Data Link, Routing, and Transport Layers. Lets start with the Data Link. Assume that mobile nodes (e.g., laptops, PDAs) communicate with fixed base stations using a radio. The radio has a fixed circular range d. A station A can communicate with station B as long as:

- B is within the range of A
- Nobody else besides A is transmitting in the range of B. (If somebody else does B will detect a collision.)



One approach to this mobile LAN is to use CSMA/CD as in Ethernet. Unfortunately, this is not efficient. Consider the 3 mobile stations in the figure. Suppose A is transmitting to B, and C decides to transmit to some other station. C may sense the channel and not hear anything because it is out of the range of A. So C begins to transmit and causes a collision at B (which receives) two signals. This is called the *hidden terminal* problem. Suppose, on the other hand, that B is transmitting to A and C wants to transmit. Then C will indeed hear B and defer transmission although it could have safely transmitted to anyone else except B. This is called the *exposed terminal* problem. We saw the hidden terminal problem in class, not the exposed terminal and we want to work out some more details

An alternate approach called MACA works as follows. When C wants to transmit it transmits a small control packet called RTS to its intended receiver. The RTS contains the length X of the data packet C wants to send. If the receiver gets the RTS, it sends another control packet back to C called a CTS, which also contains the length X of the data packet. If C gets a CTS, C transmits data. If C does not hear a CTS, C times out.

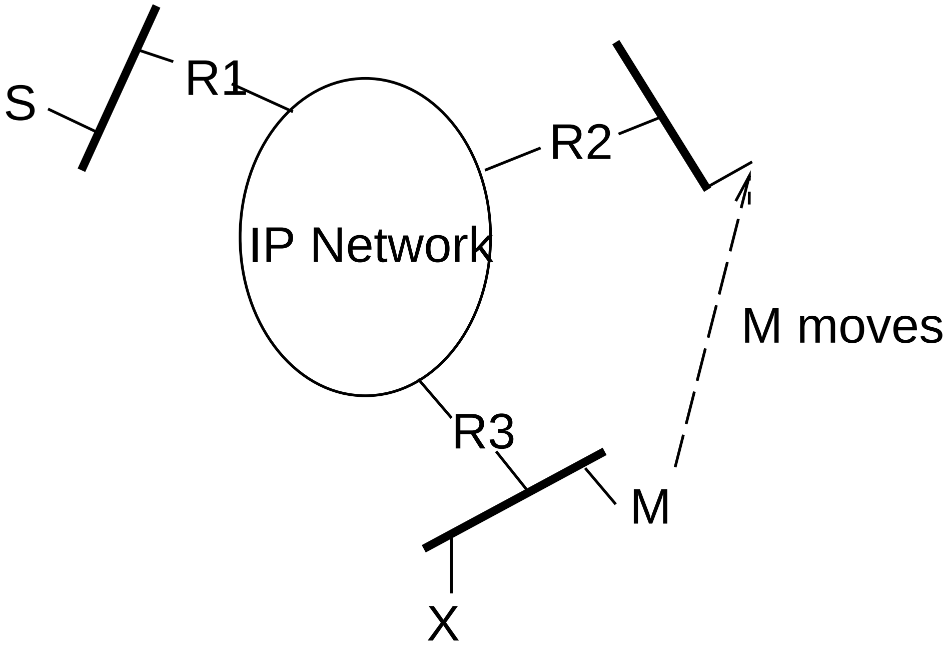
Any station D that hears a RTS waits long enough (for the RTS and the corresponding CTS to be sent back) before D attempts to transmsit. If station D hears a CTS, D waits long enough (for the CTS to finish and the corresponding data to be sent) before D attempts to transmsit. Notice that in the RTS case, D only waits for the CTS to finish and not the data; however, in the CTS case, D waits for the data to be sent as well.

- Why do the hidden terminal and exposed terminal problems not occur in the Ethernet?
- How does MACA avoid the hidden terminal example described above? Explain briefly the sequence of events when C wants to transmit.
- How does MACA avoid the exposed terminal example described above? Explain briefly the sequence of events when C wants to transmit.
- Suggest a reasonable retransmission strategy for a station that times out without receiving a CTS? (Your strategy should ensure that repeated collisions between stations will eventually sort themselves out.)

4. Mobility and Routing, 15 points: We now consider the problem of routing to a mobile station that has moved. In the figure below, assume that mobile station M is "homed" on the LAN that contains $R3$ and X. M has an IP address that contains the net number corresponding to its home LAN. Sometime later, M moves to the LAN that contains $R2$. Stations S and X may still send packets to M at the old address, even after M has moved. We want to explore how to modify IP routing so that these packets can be delivered to M at its new location.

Lets assume that M (before or after moving) tells its home router $R3$ of the net number of its new LAN. We assume that M does not get a new host ID at its new LAN because it would take too long to have one assigned. Assume that M has a unique Data Link ID (e.g., Ethernet address) D_M that it carries to the new LAN.

- Assume X wants to speak to M. Since it thinks its on the same LAN as M, X may send an ARP for M. What should router $R3$ do in response?
- Suppose that $R3$ has a packet destined for M that was sent by either S or X. How does $R3$ send the packet to M?
- Assume that the packet for M gets to $R2$. How does $R2$ deliver the packet to M (remember that $R2$ cannot do an ARP because M does not have a host ID on this new LAN. However, you can add info to the packet that can help $R2$.)



- The indirect route can result in sending a large number of extra packets when speaking to M. Can you suggest some way of eventually avoiding the indirect route?
- When M eventually moves back to its home LAN, how do you ensure that eventually everyone sends packets to M's home LAN.

5, Mobility and Transport, 15 points: The mobile station M in the previous question may take a long time to move, but may be in the middle of a transport connection with S. Suppose the time to move can vary from a few seconds to a few hours depending on where M is going to move. Assume that a lot of data has been transferred in the transport connection already, and M does not want to waste this work by tearing down the connection (and then restarting the connection after M moves.)

- Hugh Hopeful suggests that M just use the regular flow control mechanisms in the transport protocol to stop S from sending data while M is moving. What mechanism is High referring to?
- Peter Protocol points out that Hugh's mechanism will not work if the move time is significantly greater than the network round trip delay. Why?
- Peter Protocol therefore suggests that M send a FREEZE message to S before M moves. The FREEZE message contains a conservative bound on the time for the move, and must be acked before M moves. What should S do on receiving a FREEZE message?

- After a few bugs, Peter Protocol realizes that the FREEZE messages have to be numbered with a sequence number. Can you suggest why?