HW #3 Due Wed Oct 29 (with solutions)

All problems equally weighted

1. ODMRP and flooding

It is well known that in an ad hoc network when the mobility is very high, it is better to use “flooding” (ie, universal broadcast) instead of multicast over a tree/mesh. Explain how ODMRP could be modified to operate in a flood mode when mobility is very high (Hint: first, figure out how the source can determine that it is in a high mobility regime…)

Answer: in ODMRP the destinations/sources periodically refresh their paths to the members. To cope with high mobility, a node will increase its refresh rate as a function of the change in local connectivity it observes – for example, if half of the neighbors have changed in one minute, the node will refresh the membership at least every minute. When the refresh interval goes below a certain threshold, the O/H becomes excessive. The source can then request the “flood”. For example, the source sends out the next m-cast message with a “flood” header.

2. TCP ACK compression on wireless multihop

Consider an ad hoc 802.11 network. The collision avoidance mode (RTS/CTS) is used. We have shown that TCP ACKs compete with TCP data packets and cause a reduction in throughput. Do you think it would make sense to implement a “receiver only” modification to TCP which extends the “delayed ACK” feature of the conventional TCP? Namely, the destination waits until the full window of packets has been received before it returns a “cumulative” ACK. This way, we avoid the conflict between data packets and ACKs. Discuss performance for large N. For simplicity assume that ACK tx time is the same as Data pkt tx time.

Answer: with 802.11 and N hops the optimal window is approximately N/6 (if data pkt size = ACK pkt size). Thus, for large N, the throughput asymptotically converges to C/6, where C is link capacity. If we remove the ACKs in the opposite direction, we can maintain one outstanding packet every three hops. So, throughput increases to C/3. However, the flow must be suspended when the N/3 packets are all delivered to destination until the ACK is returned. Assuming the tx time of a packet is T, the total cycle including full window delivery to destination and ACK return is then 2NT + NT = 3 NT. Throughput is thus given by C/3 times the fraction of cycle during which the source transmits, ie:

\[ \text{Thr} = \frac{C}{3} \times \left\lfloor \frac{NT}{3NT} \right\rfloor = \frac{C}{9} \]

So, the effective throughput with “ACK compression” is worse than C/6.

Comment: for the ACK compression scheme to yield throughput improvement, the source must output more that two full windows before it stops to wait for the cumulative ACK.
3. Improving ad hoc TCP fairness

In class we have discussed two techniques to improve TCP fairness: a physical and MAC layer modification called \{Active Node Estimation + Beam Forming\} and a network layer technique called NRED. Can you briefly discuss the pros and cons of the two schemes? Specifically, which scheme is easier to implement? Which scheme reacts more promptly to capture?

**Answer:**
The ANE+BF is definitely more difficult to implement (you must change the MAC standards); it however intervenes more promptly, preventing capture.

4. 802.11 and Bluetooth scatternets

In discussing TCP unfairness in 802.11, we pointed out several possible causes including different radio ranges (tx, interference and carrier sensing) and the interference between data packets and ACKs. Are these causes present also in Bluetooth?

**Answer:** in Bluetooth there is no sensing (it is time division); moreover the ACKs do not compete with data since the channel is “full duplex”. Thus, the above causes are not present.

5. Random loss vs congestion loss – impact on video streaming

Consider a video streaming session over an ad hoc path. The source can adapt the stream rate by adjusting the resolution (higher resolution = higher rate). It can also increase/reduce pkt length and include more or less forward error correction coding. The adaptation is of the type: Additive Increase Multiplicative Decrease (AIMD). The destination monitors the packet loss rate and the jitter of the packets as they are received. It feeds back congestion and loss information to source

(a) explain how can the destination discriminates between random packet loss (as caused by jamming by a microwave oven) from congestion loss
(b) will the source react differently if it knows the loss was random or congestion related?
(c) What is the potential problem if the destination cannot discriminate?

**Answer:** The receiver declares random loss if the jitter is next to zero. Else, congestion! The source reduces rate in congestion. It reduces pkt length and increases FEC in random error cases. If there is no discrimination, the source will eventually reduce the rate to zero in case of random errors