Riley Burczynski PA2
REPORT

I have read and understood the course academic integrity policy.

Time Out Scheme:

The time out schemes that I chose for my protocols were fairly straight forward. First I determined what the smallest time out would be. This came out to be 20 because a packet could take between 1-10 time units each way. Thus I knew my time out time had to be greater than 20. I then performed a few experiments on abt/gbn since they are more similar to determine the time out time that gave me more preferred results. This ended up being 30.0, meaning after 30 time units I would simply retransmit the corresponding packet in abt/gbn. Then for sr since there was more work being done in my functions I upped this time out interval to 35 time units.

SR Timer Implementation:

I implemented the multiple timers with one timer for sr by utilizing a few data structures and functions, they were as follows:

- deque<pkts> timer_packets - this deque held the packets that have currently been sent to layer 3 from A in order of the time they were sent(lowest time first)
- deque<float> timer_times - this deque held the time at which the packet at the corresponding index in timer_packets was sent
- get_sim_time() - to get the current simulation time
- Float timer_time - the default timeout for the timer, set to 35 in sr

I went with using a deque since it provided easy access to the entire list, especially the front. The idea behind my timer went like this. First any packet that is sent to layer 3 from A is pushed to the back of timer_packets and the current simulation time is pushed to the back of timer_times. If this packet is the only current packet in the medium the timer is set to the default time, else no timer is set yet for this packet. Next I had to deal with the timer when an ack was received. I did this by simply checking if a ack corresponded to any of the packets currently in timer_packets, if it did i would then remove this packet from both deques and start the timer back up for whatever packet is at the front of my timer_packets deque. It's important to note that when starting the timer back up in A_input I do not simply use the default time but rather I set the timeout time to be the remaining time for the packet in front which is the default time - (current time - time packet was sent). The final step of the timer was the timer interrupt. To accomplish this I simply took the packet at the front of timer_packets, since this is the packet corresponding with the current timer interrupt, and pushed to the back of timer_packets and timer_times with a new time sent. I then restart the timer with the current front of the timer_packets deque and set the timer using the same math described above. This scheme allows all packets to be retransmitted at the exact time they would be had we had multiple timers to work with.