Morgan Trembley

Networks Project 1

19 September 2020

Neighbor Discovery:

Once started, a periodic timer of 30 seconds sets off a neighbor discovery routine. I tried to mimic the layers that were suggested without using a different packet type (ND has a R/Q and seq header and LL has src and dest) so ND fills in the packet's seq, TTL as Q or R, protocol as ping, and a payload of "Hello?". Then it is sent to the LinkLayer interface along with the node ID and destination as broadcast and fills in the source and destination of the packet. Then simplesend is called to broadcast the packet. When a node receives a packet, it checks the protocol. If it is ping, it is unpacked and sent back to the neighbor discovery interface along with the original source. If there is a 'Q' in TTL we change it to 'R', change the protocol to ping reply, change the payload to "Hello Neighbor!" then send it back to the LinkLayer using the current node id as source and original source as the destination. The LinkLayer sends, receives, checks the protocol for ping reply and sends it to the right neighbor discovery command where the reply source is added to the destinations, neighbor list along with the stats required for this assignment. After, we update the neighbor table, checking that current neighbor's link quality is at least 33% and TTL has not fallen below 0. If either fails, the neighbor flag is changed to 0. printNeighbors shows each node and its neighbors in the format NODE -> NEIGHBOR and is currently triggered at the end of the fired() event.

Flooding:

I hijacked the ping python command to test my flooding protocol. Now, when ping is called, the source is the node source for flood, the destination is the target destination for flooding and the payload is the flooded payload. I opted for the wired flooding strategy utilizing the neighbor table from NeighborDiscovery to find a node's neighbors. When Flood() is called, a package is built using the parameters above, the node checks the neighbor table for neighbors, eliminates the source if there is one, checks the TTL and seq to make sure it isn't sending the packet more than once or after It dies, then uses the LinkLayer to send the packet to all valid neighbors. If the packet is received by the target destination the payload is output in the flooding debug channel and is not sent on. If the receiving node is not the destination, the TTL is checked then sent back to be repackaged and sent to the receiving nodes neighbors. The other neighbors will continue to send packets until all other accessible neighbors have the packet though.