Morgan Trembley

CSE 160

Project 2 Description

LinkState:

My LinkState interface first initializes a graph filled with values of ‘255’ unless ( i == j) because all nodes are on that diagonal, in which case it is filled with ‘0’. A periodic timer is started that fires once a minute unless the neighbor table is updated, then it fires immediately after updating. Then, for each node, it references its’ neighbor table and fills neighbors in as ‘1’ (default cost for now). Then, it floods its’ neighbor table to all other nodes. The Flood interface passes the LSA payload and source to the receiving node and it fills in the appropriate row of its graph. The SP function runs if there is an update to the neighbor table or after 5 cycles of graph updates to lower the odds of dropped packets/timing issues. When SP runs, it adds the neighbors of the original node to a list of potential shortest paths, it picks the shortest path, adds it to a confirmed SP list, then does the same thing for the chosen node. After adding the new node’s neighbors to the list along with their cost (including the cost for the first traversal), the list is consulted and the lowest cost path is chosen again until the confirmed shortest path table is full. The lists include the destination node, the cost to get there, and the node the current node needs to send to for the shortest path. When this is complete, it is ready to forward ping requests. Currently, The graph is rebuilt once a minute, the neighbor Discovery interface fires every 30 seconds. If a change occurs in the neighbor table, it informs LinkState and it fires the graph update and after 5 graph updates, the shortest path calculation runs and recreates the routing table to reflect the update to the routing table.

IP:

My IP interface waits for a call with arguments of destination and the payload. It builds a packet then checks the destination. If the destination is the current node, it passes the packet back to the caller. If it isn’t, it consults the routing table (confirmed) from LinkState to find out who to send the packet to. If the value in the table is 0, it posts a warning that the node is unreachable and ends. If not, it calls the LinkLayer to send the packet to the next node in the path and this process repeats until the packet reaches the destination and passes the payload back to the caller.

MyPing:

The MyPing interface is called by the default ping command in Node in place of the original implementation. It takes in a destination and payload, then calls the IP interface to forward the payload to the destination. Once the payload is returned, it is displayed in the transport channel.

*Sometimes ping doesn’t successfully complete and output the payload. I think this is a timing issue but have not been able to find the cause. It is rare but happens occasionally.