

COMP9331 Assignment Report

Student Name: Rongtao Shen

Student ID: z5178114

Please use Python 3 to test my code.

1. A brief discussion of how you have implemented the LSR protocol. Provide a list of features that you have successfully implemented. In case you have not been able to get certain features of LSR working, you should also mention that in your report.

- At first, I get the neighbor information of each router by opening the .txt file. By traversing each line in the file, I get the number of neighbors, the path cost between the router and their neighbors and the port number of the neighbors. Then I set up a dictionary which is a data structure in Python to store the neighbor information. In addition, I set up another dictionary to store all routers' information in the entire network in order to struct a topology. Then I used three threads in my program, the first one is for sending link-state packets, it can send the link-state packets every one second by using socket programming and time module in Python. The format of the packet will be mentioned in part two. The second one is for receiving link-state packets and forwarding the packets that the router received, the receiving socket will listen the address all the time. The method of dealing with node failures and restricting excessive link-state broadcasts will be mentioned in part two. The third one is for using Dijkstra's algorithm to calculate the shortest path cost between all routers and print every path to all routers in the network.
- In my program, it can achieve all the features mentioned in the assignment specifications like sending link-state packets, printing the shortest path and cost between each router, restricting link-state broadcasts, dealing with node failures and sending link state update messages when a failed router joins back the topology again.

2. Describe the data structure used to represent the network topology and the link-state packet format. Comment on how your program deals with node failures and restricts excessive link-state broadcasts.

- I use dictionary in Python to represent the network topology and the link-state packet format is a string which contains packet sequence number, packet owner's name, path cost and neighbor's port. The format is like '0 A 5000 B 6.5 5001\n0 A 5000 F 2.2 5005', the sequence number will plus one every loop.
- As for how my program deals with node failures, in receiving thread, I set up a dictionary to store the time to receive a new sequence packet from every router in network and set up a dictionary to store every router's packet sequence. At the beginning of receiving part, first check the time in the time dictionary that the last time each router package was received, if current time and the time of each router in the dictionary exceed the timeout time we set, the program need to delete this router's information in its topology and clear the sequence number list of this router.
- As for how my program restricts excessive link-state broadcasts, in receiving thread, before forwarding the packet to the router's neighbor, the program first determine whether this neighbor is the owner of this packet and determine whether this neighbor is included in packet's neighbor information, if this neighbor is the owner of this packet or this neighbor is included in packet's neighbor information, the router will not forward this packet to this neighbor. In addition, I added a sequence number to each link-state packet. Every time the router receives a link-state packet, in addition to judging the above-mentioned content, it also needs to determine whether this packet is a new sequence packet, if this packet has not been received before, the router will perform the forwarding operation and update network information. If the sequence number of this packet is received before, then it will do nothing, which could exclude interference from expired packets in the network and some unnecessary forwarding.

3. Discuss any design trade-offs considered and made. List what you consider is special about your implementation. Describe possible improvements and extensions to your program and indicate how you could realize them.

- What I consider is special about my implementation is that I used broadcast packet to replace the heartbeat packet, by recording the time to receive new sequence packet and set up timestamp to deal with node failures which I mentioned above. In addition, I added the sequence number in the packet, which could restrict link-state broadcasts and quickly sense the re-recovered router in the network.
- As for possible improvements and extensions to my program, I think I can use class and object-oriented approach to store the information of routers and their neighbors, which may make the code look clear.

4. Indicate any segments of code that you have borrowed from the Web or other books.

- For calculating the shortest path in program, I refer to the pseudocode of Dijkstra's algorithm mentioned in lecture.