Computer Science 118: Computer Network Fundamentals

Spring 2014

Professor: Ciaran McGoldrick

Authors:

Jeffrey Tai – 504147859

Brian Chang – 304151550

Mark Matney – 504152097

I. Implementation

A. Overview

The basic overview of the project will cover from the beginning at initialization to the end when the message is sent from the first to last router following the respective least cost path. We begin by initializing our distance vectors from the sample file provided and finding all neighbors of each router. The router output files are set up, followed by binding each socket to a specific port. The “starting” router sends its DV to its first neighbor, and from there, each router will be ready to receive a DV. Upon receipt, it will update its own DV, if necessary, and log the change in the routing output text file. Each router will then send its updated DV to all of its direct neighbors. These DVs continue to propagate across the network until the network has reached a stable state in which each router has the following information about all other routers: the name of the router, shortest cost, outgoing port number, and destination port number. Once the stable state is reached, all DVs are written to their respective router output files.

When killing a router, we put the changes to the sample.txt file and re-initialize the neighbor matrix and the DVs. For all routers that were previously were neighbors of the dropped router, the cost is changed to INT\_MAX. When the routers are sending their DVs, no DVs will be sent to or sent from the dropped router. In the end, the tables will once again converge.

B. Structures

Two structures are used, one for the router and another for the packet. The router structure represents the distance vector so it contains the index of the respective router and following information about the other routers: the character name, cost, outgoing port number, and the destination port number.

C. Message Structure

The structure of the packet contains a flag character to determine if it is a control or data packet, a message, the original source node, the destination node, and the arrival and forwarding port of each router visited along the path that the packet takes.

D. Routing/Forwarding

The user is given the option to forward a packet once all the initial tables have converged. The source router and destination router are taken as input, which are in the form of letters, such as ‘a’ or ‘b’. Both lower case and upper case letters are supported. The information is then placed into the packet structure and starting from the source node, it continues to the respective outgoing port of the specified destination.

As the message is passed through each router given the shortest cost path, the arrival of a data packet is recorded in the routing output text file with the arrival port number and the forwarding port number. When the packet reaches its destination, all previously visited routers along the path are written to the destination router’s output file.

II. Challenges

We were presented with two major challenges as we did this project. We had issues with having a router send its distance vector, in the form of a buffer, to multiple sockets, which would be its neighbors. We learned how the select() function of UDP sockets was similar to a scheduling algorithm in which it would choose which socket should have data received from. Through trial and error, it was obvious that each test to see if a socket file descriptor was set had to be an if statement rather than an if-else statement.

For testing purposes, we originally had Router A begin by sending its first neighbor, Router B, its DV, and the while loop would continue from there. However, we came across an obstacle in which Router A successfully sent its DV to Router B, and in turn, Router B would send its DV to all of its neighbors. Each of those neighbors would properly update their own DVs but from there, they wouldn’t send their DVs to their respective neighbors. Eventually we found that there was a small issue when converting the DV into a buffer that caused the routers not to send to their neighbors.