REPORT

Simulation of Link-State Routing Protocol

Vaibhav Aggarwal
Illinois Institute of Technology
CWID A20374988
vaggarwal@hawk.iit.edu
Campus : Main Campus

Dhruv Malik
Illinois Institute of Technology
CWID A20375772
dmalik@hawk.iit.edu
Campus : Main Campus

INTRODUCTION:

Link State Routing Protocol Overview

Link-state routing protocols are one of the two main classes of routing protocols used in packet switching networks for computer communications, the other being distance-vector routing protocols. Examples of link-state routing protocols include open shortest path first (OSPF) and intermediate system to intermediate system (IS-IS).

The basic concept of link-state routing is that every node constructs a map of the connectivity to the network, in the form of a graph, showing which nodes are connected to which other nodes. Each node then independently calculates the next best logical path from it to every possible destination in the network. The collection of best paths will then form the node's routing table.

Dijkistras Algorithm to Find the shortest path:

- Initially all the nodes value are set to be 999.
- The connection between the router to itself is set to be zero and if a connection does not exists with the other router then -1 is assigned.

Following are the steps carried out

1. Select the source node and then add it to the dictionary and set the shortest distance between itself and the neighbors i.e cost between the root and its adjacent nodes.Also set the shortest distances of the source node to be zero

2. Iteration : Repeat the above steps till all the routers are added to the path and select the minimum distances and update it to the current path.

3. Updating: Update the shortest distance for all remaining node that are included in the current path.

DESIGN AND DESCRIPTION:

1. Entering the Topology Matrix from the File
2. Calculation of Shortest Paths Using Djikistras Algo
3. Performing Various Operations Such as
    1. Display All shortest Paths
    2. Remove and Recover Router
    3. Remove Edge

Implementaion:

- Basic file operations functions have been performed for reading the Input file adjacency matrix.
- Implementation of the finding shortest paths is done is using Djikistras Algorithm. Which is implemented using basic data structures such as Lists,Dictionaries.
- The Djikistras Algorithm finds the shortest distance of each node to all other vertices and hence we can calculate the distance of the particular source node to the destination from the above calculated distances.
- The connection table is also built using the Djikistras algorithm keeping the record of the visited and unvisited nodes thus building the connection table.

Program Flow:

File Reading and Loading the Matrix:

- The simulator will asks for the input which contains the topology Matrix. It will read the file value in the for of Adjacency Matrix.
- If the file is not loaded or is not of the proper format there will be rror generation.
- The following Adjacency Matrix is used for processing the shortest paths. The Adjacency Matrix contains weight to the links to the other nodes.
- Value equals to -1 is assigned in the Matrix Corresponding to no connection between one router to other.
- Value of 0 in Matrix equals that there is no connection of the router with itself.

Calculation of the shortest path:

- The Topology Matrix is then passed into the Djikstras Algorithm a sone of the Parameters.
- The Djikistras Algorithm is used to calculate the shortest distance of the nodes based on the weights assigned by traversing the graph. It also keeps records of the visited and unvisited nodes.
- This involves creating of the connection table which shows the traversing of the path from the particular node to all the nodes and assigning an interface value.
- Calculation of the shortest path to a particular node can also be calculated using the Djikstras Algorithm

USER MANNUAL FOR THE PROJECT

RUNNING THE FILE:

Run the file in python 2.7 Environment

Run the file named link_state_imple.py

STEPS:

1. Enter the Network file
2. Enter a valid Source node
3. Select the options you want to perform

For Removal of the edge:

- Enter the Input Value with the comma separated. For Example if you want to remove the edge between the router 3 and 4 enter the value=3 4 space separated

For Router Removal:

- Enter the router to be removed
    1. Click on the Connection table to show the connections to various router.
    2. Click on the display shortest path to show all paths
    3. Then Recover the node to perform other operations

About the GUI package

We have used the Tkinter to link the backend of the code to the Frontend. We have used the SeaofBTCapp class to display the various frame windows for the selection options.

GUI Working

Start Page: Initial the start page is poped up allowing the user to load the File.

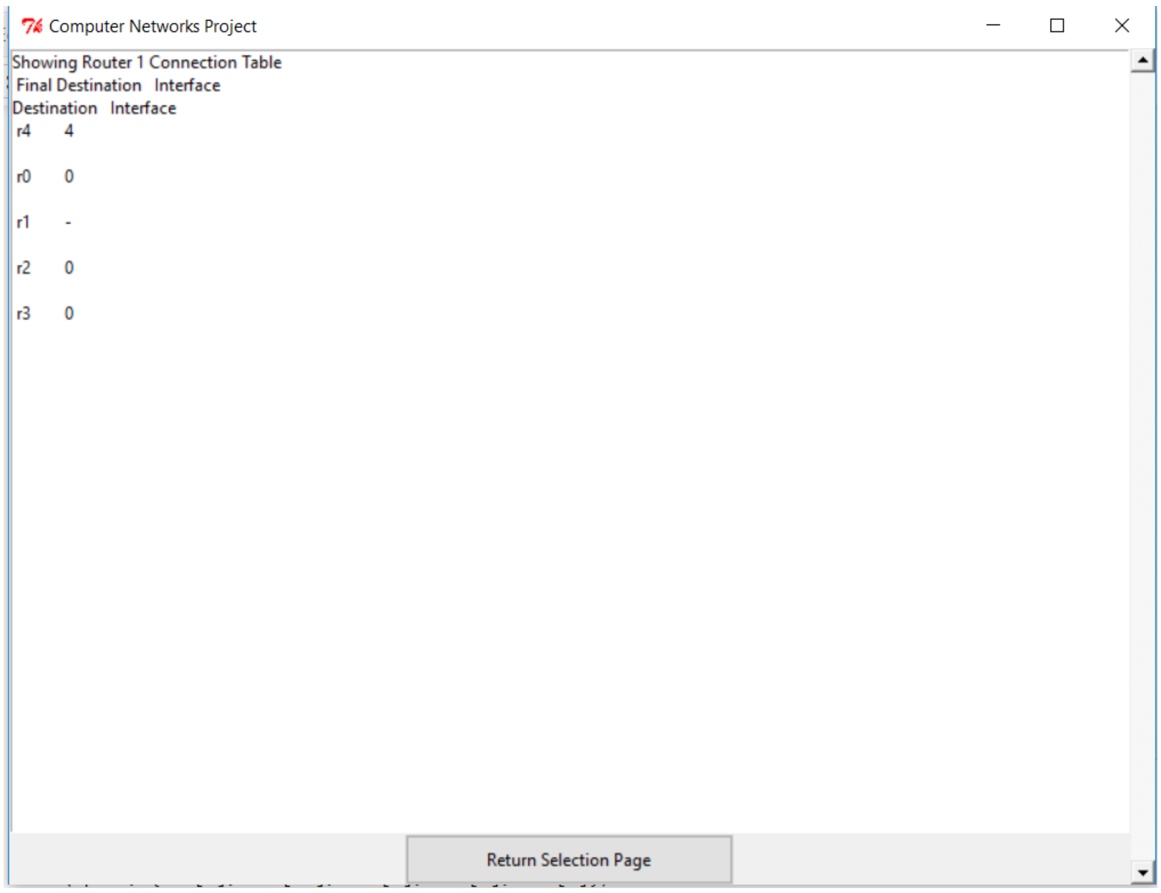
If the file is not selected or improper file Format is loaded then Error is thrown for the same.

Enter Source Node Page:

Allowing the user to enter the source Node

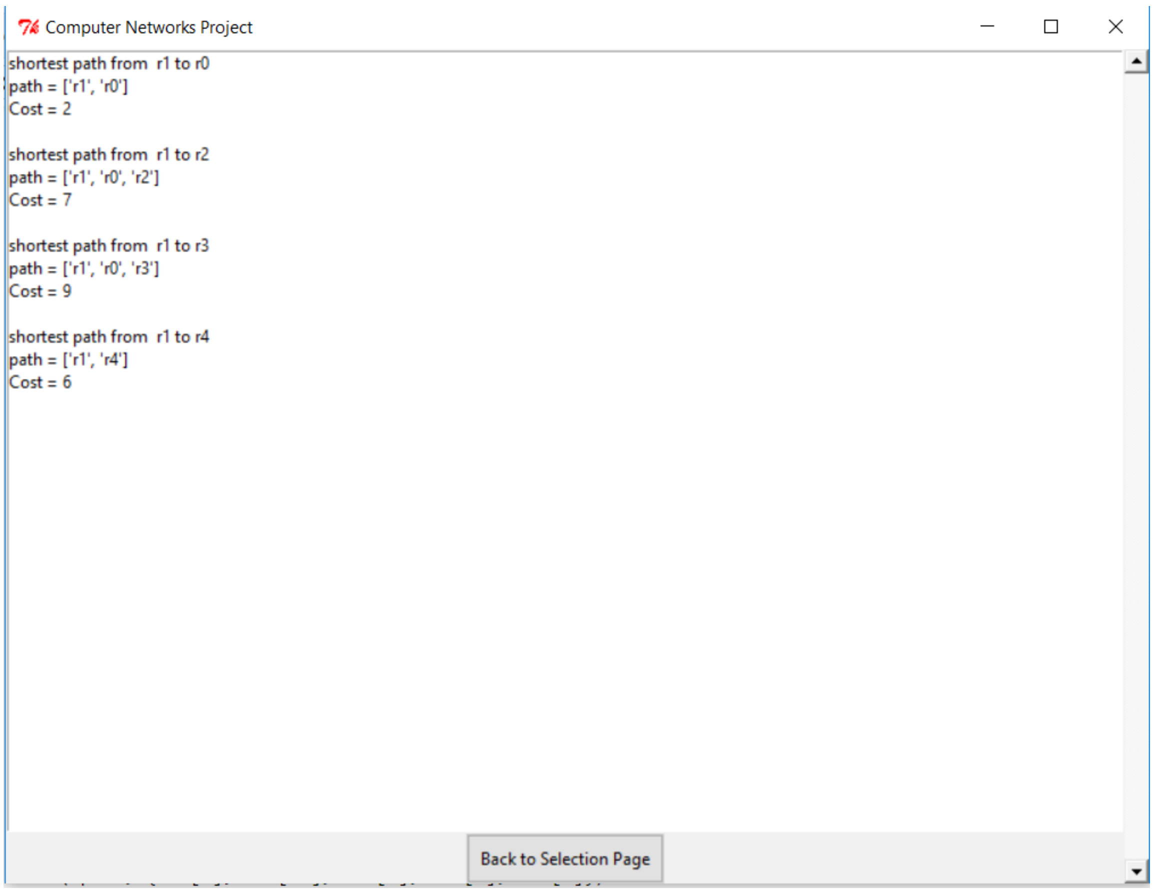
Selection Page: This Page Allows to select the user to perform Function Such as:

- **Display connection table**



```
Showing Router 1 Connection Table
 Final Destination   Interface
Destination   Interface
 r4      4

 r0      0

 r1      -

 r2      0

 r3      0
```

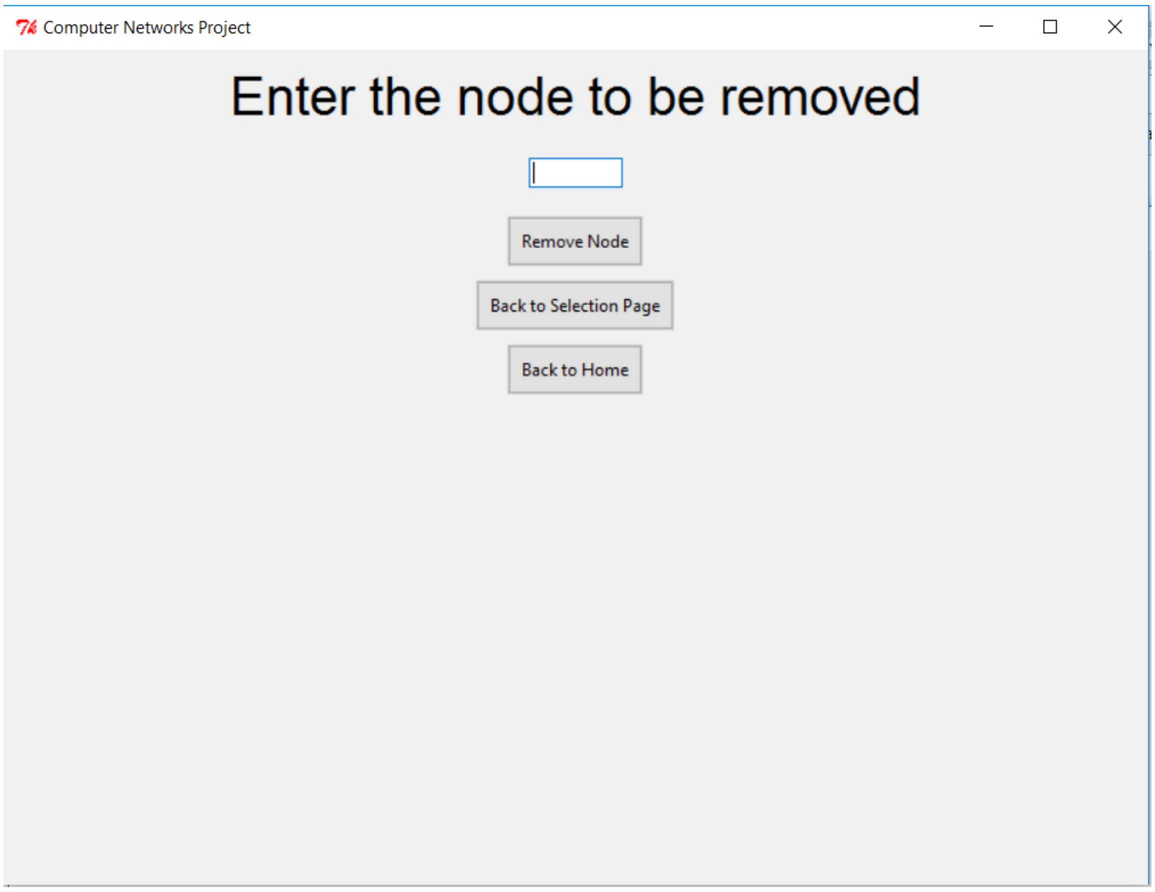
- **Display shortest Paths**



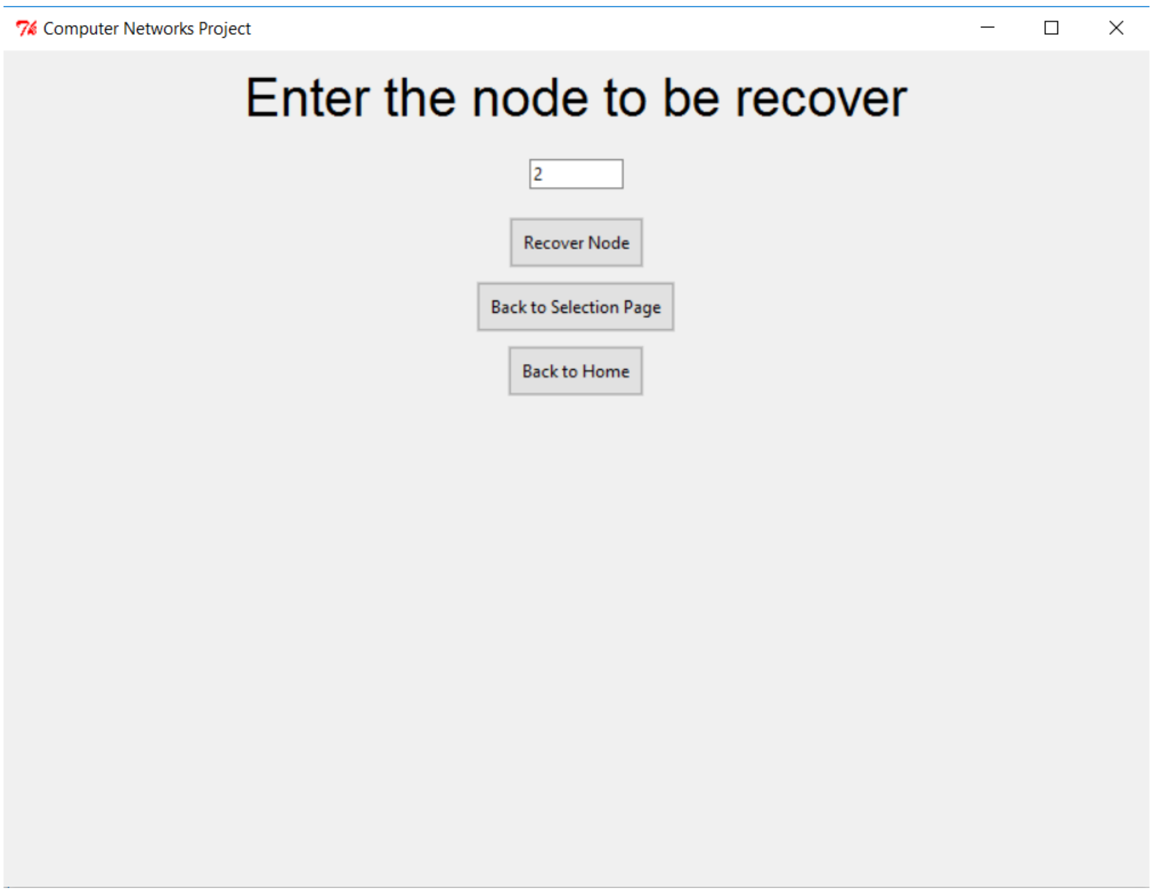
Computer Networks Project

```
shortest path from  r1 to r0
path = ['r1', 'r0']
Cost = 2

shortest path from  r1 to r2
path = ['r1', 'r0', 'r2']
Cost = 7

shortest path from  r1 to r3
path = ['r1', 'r0', 'r3']
Cost = 9

shortest path from  r1 to r4
path = ['r1', 'r4']
Cost = 6
```

- **Remove a Router**



- **Recover a router**



For each of the above the Functions a class is Created to perform to corresponding functionality and to show the output to the user.

Low Level Design Document

Software Requirements:

- The simulator we have build is using Python 2.7 programming language and wil work on windows 7/8
- We have used following Python Packages :
    1. Matplotlib: This package is used for plotting the graphs in python and we have used pyplot(x,y) function for plotting the graphs.
    2. Numpy: This package is used for processing the given data in arrays format. It allows performing easy operations such as retrieving the values at particular node, deleting updating at a particular index.

Following are the functions created to perform the low level requirements.

- def get_the_network(self): This function is created to allow the user to load the file and to create the Adjacency matrix for further processing

.

- def get_source(self, source): The next step is to allow the user to enter the source node.
- def display_connection_table(self): this function displays the connection table of the source node entered by the user.

<u>High Level Design (HLD) Document</u>

The document will specify the requirements and the high-level design of link state routing simulator.

- Integrate the Backend code with the frontend code. This calls for the creation of the GUI.
- The GUI which allows user to work on different frame windows allowing to choose the selection option to perform various operations.
- The GUI is created using Tkinter package and using the inbuilt functions of it such as
    1. Message Box: It is used to display the error messages in a box.
    2. FileDialog: It allows user to search the file in the computer and loading it in the computer.

Implementing the node remove:

def remove_node(self,new_node): This function takes the argument as the node to be removed. It modifies the current Adjacency Matrix by deleting the column and row corresponding to the entered removed node.Then, it computes the shortest distances and the connection table to the corresponding new Matrix.

Implementing the Recover node:

def recover_node(self,new_node): This function allows the user to enter the node that is being deleted to be recovered. It adds the initial list of values of the corresponding matrix back to the matrix at the router number position and then updates the corresponding Topology Matrix.Thus the new shortest paths and the connection table are being calculated with the new values of the Matrix.