Computer Networks and Applications (COMP3331)

Programming Assignment:

Routing Performance Analysis

Nicholas Ho z3422527

Steven Ru z3418434

# Data Structures

We have chosen to represent the topology with an undirected graph, which is made up of 3 Java classes:

### Node

This represents each of the routers in the network.

Contains 3 fields:

* **String** name: The name of the node (A or B or C etc)
* **Double** minDistance: Keeps track of the minimum distance this particular node is from the start Node. Required for Dijkstra’s Algorithm.
* **Node** prev: Keeps track of the previous Node object that was visited before the current Node was visited.

### Edge

This represents each of the links in the network. Since the graph is an undirected graph, we represent each undirected Edge with 2 directed Edges. Any operations on such an undirected Edge will affect both Edges running in either direction.

Contains 5 fields:

* **Node** from: Contains one of the two Nodes at either end of the Edge.
* **Node** to: Contains one of the two Nodes at either end of the Edge.
* **Int** propagationDelay: Propagation delay is an argument taken in from topology.txt.
* **Double** numSimulCircuits: numSimulCircuits is an argument taken in from topology.txt.
* **ArrayList<VirtualCircuit>** circuits: This is the list of Virtual Circuits that are currently using this particular Edge as a Link.

### UndirectedGraph

This represents the entire network.

Contains 2 fields:

* **ArrayList<Node>** nodes: Contains the list of all nodes or routers that are involved in this network.
* **ArrayList<Edge>** edges: Contains the list of all edges or links that are involved in this network.

# Comparison of Performance Metrics

# Analysis of Results

# Evaluation of VPN