User Manual

As mentioned previously, there is two ways to use the routing program, through importing the library or running the application. As this is an user manual, I have chosen to focus on the application. The application is run by calling `python3 main.py`

From there, the user is greeted with information on all of the functions available. To view this list again the `intro_help` command can be used. To get information about a specific function, the command `help` *`command_name`* will give this information

An important function is `load_file` *`filename`*. This loads any initial nodes and connections from a predefined file. Similarly `write_file` *`filename`* can be used to store a current network's nodes and connections into a file. In both cases the structure for these files looks like:

{"nodes" : [*nodeName1, nodeName2, nodeName3, etc*] ,
"connections" : [[*nodeName1, nodeName2, distance*], [*nodeName, NodeName3, distance*], *etc*]}

Loading from a file is not necessary however, as the application will start with an empty network the user can populate. `add_node` *`nodeName`* can be used to add nodes to the network. `add_connection` *`nodeNAme1 nodeName2 distance`* can be used to add connections to existing nodes. `clear` removes all nodes and connections while `reset`, in the case of a network being loaded from a file, returns the network to it's initial state.

Connections can be modified or "downed" using the commands `update_distance` *`nodeName1 nodeName2 distance`* and `downed_link` *`nodeName1 nodeName2`* respectively. Nodes can also be removed from the network with `remove_node` *`nodeName`*. `undo` can also be used to undo back one step, to a of maximum 20 steps

To run updates, there is two options, run for a limited number of exchanges or until the network converges. `limit_run_updates` *`limit`* runs for a limited number of exchanges. It will stop if the network converges before the limit is reached. Either way, the user will be informed of the number of exchanges carried out and if the network has converged. `converge_run_updates` will run until the network converges. This is the "unsafe" version as a network may take a long time to converge. Additionally, the nodes routing tables can be traced during this, by adding an additional nodes as an arguments *e.g.* `limit_run_update 5` *`nodeName1 nodeName2 … nodeNameN`*. This also works for `converge_run_updates`

Split horizon functionality can be activated or deactivated using the commands `activate_split_horizon` and `deactivate_split_horizon` respectively. The value to count to infinity can also be changed using `set_count_to_infinity` *`value`*. By default, the count to infinity value is set to the distance of all the connections in a network plus one. It can be seen with the command `print_infinity` *`value`*

To view all nodes routing tables, the command `print_tables` is used. To view a specific node, the command `print_table` *`nodeName1`* is used. A subset of tables can be printed using `print_subset_table` *`nodeName1 nodeName2 .. nodeNameN`*

The application is exited using the command `exit`

If the user wishes to use the library instead, a sample file structure can be seen in sample.py. The functions and use are largely the same as the application. Each command is a function that the NodeHandler carries out.

2236693B – Conor Begley