Jackson Buhrer

Networks 4501

Professor Bou-Harb

5/3/25

Assignment 4 Question 4

My SDN control's design is broken down into 5 different components: The topology manager, the routing engine, the failure handler, and the CLI interface. My topology manager consists of a map that maintains the switches and hosts. Also maintains weighted links. I used this to support my add_node, remove_node, and my add and remove_link functions. My routing engine, which is the SDNController class in my code, computes Dijkstra's algorithm for all my switches. I also added dist[u] + w(u,v) == dist[v] which created an equal-cost multipath. Then we used load balancing to randomize the ECMPs. My failure handler tears down the link and then computes flow tables along the network. Finally, my CLI interface consisting of a sdn> with certain commands that inject flows and query paths. You can also visualize the path, and it downloads a picture straight to your computer. The routing algorithms I used were Dijkstra's algorithm, ECMP selection, and a flow table that I created represent the priority of the flow.

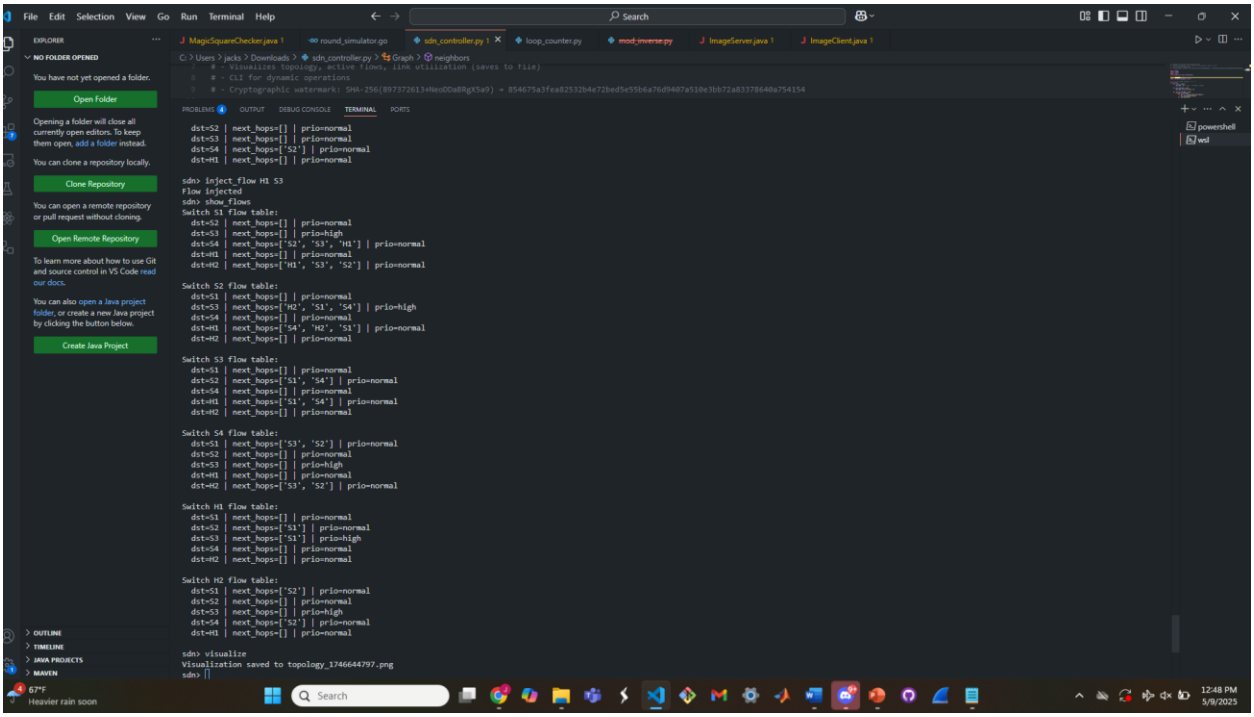
SHA-256:

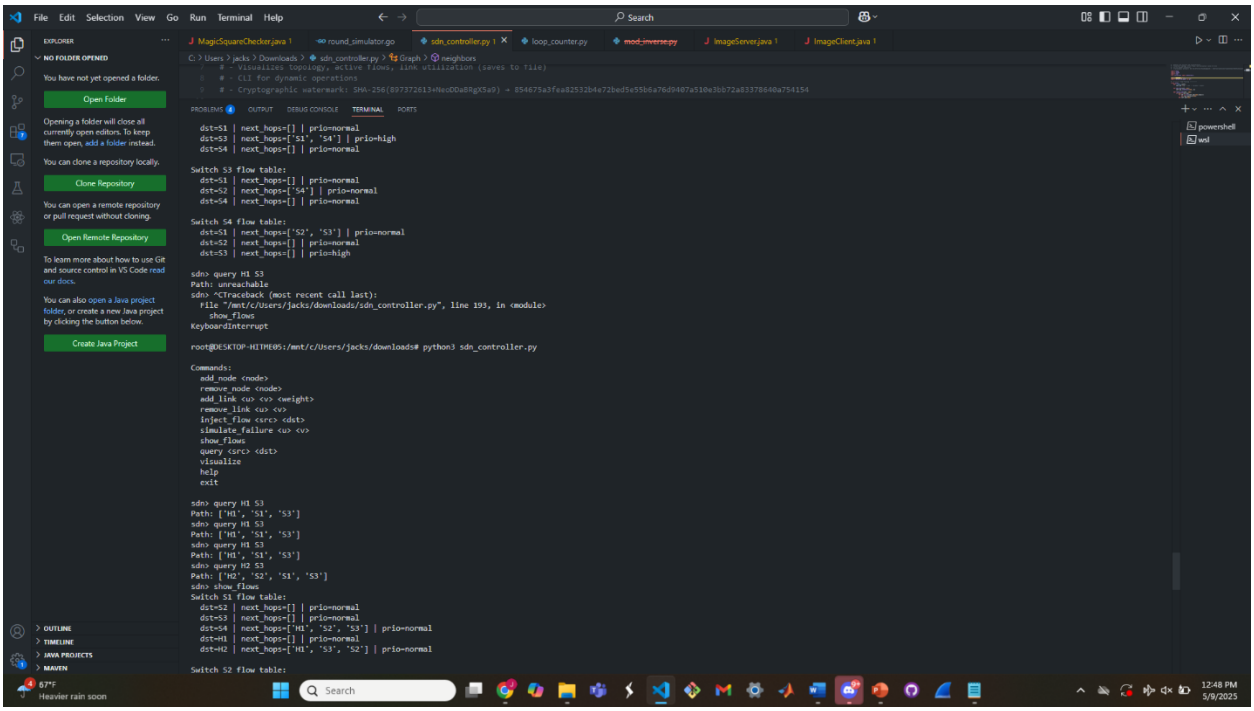
854675a3fea82532b4e72bed5e55b6a76d9407a510e3bb72a83378640a754154

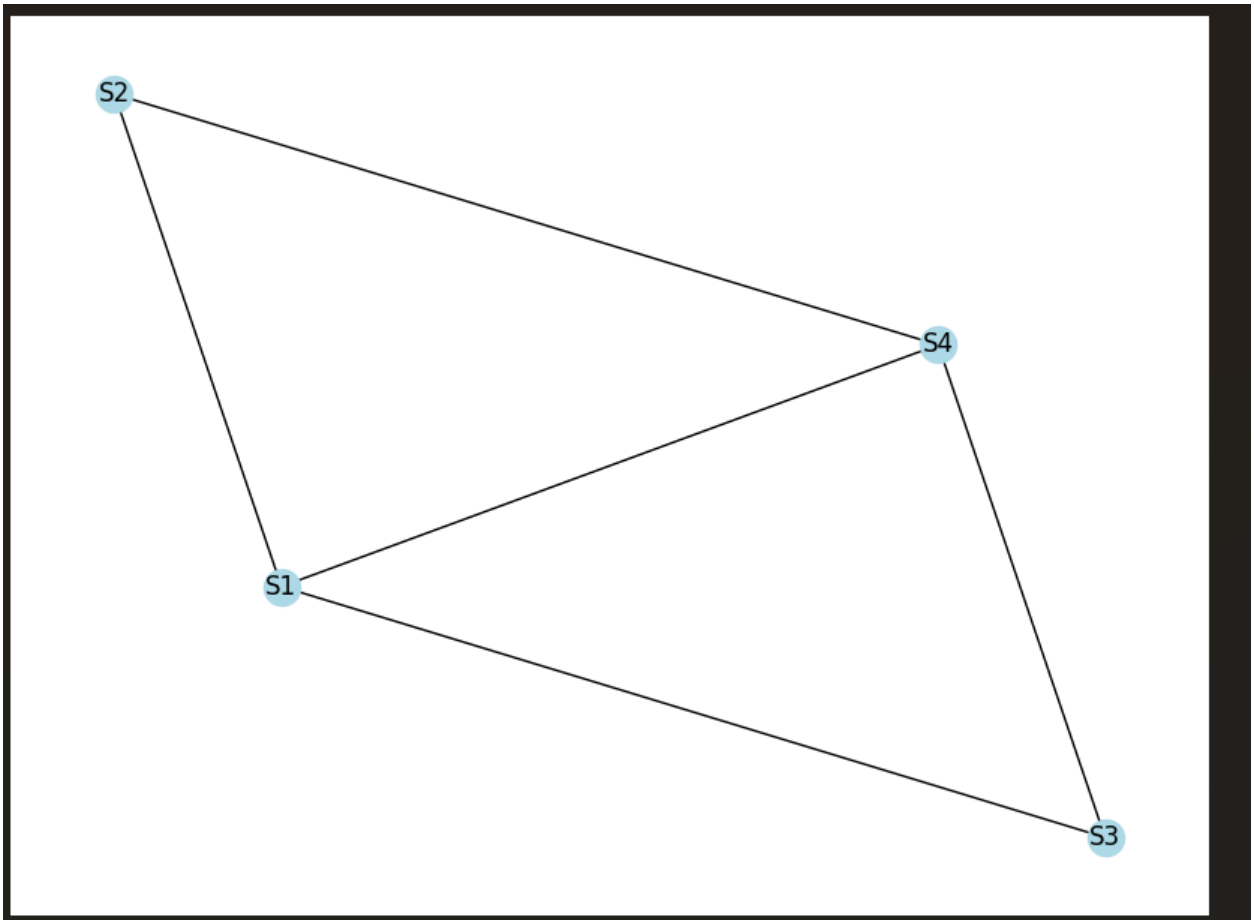
A specific challenge I encountered in this part of the assignment was to combine the load balancing for ECMP with the priority marking path, all within install_flows command. I had to evolve my code, and it finally addressed traffic distribution and QOS marking.

Here is my link to the GitHub Repository containing my file and README.MD:
https://github.com/Jbuhrer/Assignment-4-Jackson-Buhrer

Here is a picture of the visualize command and my output:




File Edit Selection View Go Run Terminal Help
Search
EXPLORER
NO FOLDER OPENED
You have not yet opened a folder.
Open Folder
Opening a folder will close all currently open editors. To keep them open, add a folder instead.
You can clone a repository locally.
Clone Repository
You can open a remote repository or pull request without cloning.
Open Remote Repository
To learn more about how to use Git and source control in VS Code read our docs.
You can also open a Java project folder, or create a new Java project by clicking the button below.
Create Java Project
OUTLINE
TIMELINE
JAVA PROJECTS
MAVEN
MagicSquareChecker.java 1
round_simulator.go
sdn_controller.py 1
loop_counter.py
mod_inverse.py
ImageServer.java 1
ImageClient.java 1
C: > Users > jacks > Downloads > sdn_controller.py > Graph > neighbors
- Visualizes topology, active flows, link utilization (saves to file)
- CLI for dynamic operations
- Cryptographic watermark: SHA-256(897372613+NeoDDaBRgX5a9) = 854675a3fea82532b4e72bed5e55b6a76d9407a510e3bb72a83378640a754154
PROBLEMS 4
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
dst=S1 | next_hops=[] | prio=normal
dst=S3 | next_hops=['S1', 'S4'] | prio=high
dst=S4 | next_hops=[] | prio=normal
Switch S3 flow table:
dst=S1 | next_hops=[] | prio=normal
dst=S2 | next_hops=['S4'] | prio=normal
dst=S4 | next_hops=[] | prio=normal
Switch S4 flow table:
dst=S1 | next_hops=['S2', 'S3'] | prio=normal
dst=S2 | next_hops=[] | prio=normal
dst=S3 | next_hops=[] | prio=high
sdn> query H1 S3
Path: unreachable
sdn> ^CTraceback (most recent call last):
File "/mnt/c/Users/jacks/downloads/sdn_controller.py", line 193, in <module>
show_flows
KeyboardInterrupt
root@DESKTOP-H1TME05:/mnt/c/Users/jacks/downloads# python3 sdn_controller.py
Commands:
add_node <node>
remove_node <node>
add_link <u> <v> <weight>
remove_link <u> <v>
inject_flow <src> <dst>
simulate_failure <u> <v>
show_flows
query <src> <dst>
visualize
help
exit
sdn> query H1 S3
Path: ['H1', 'S1', 'S3']
sdn> query H1 S3
Path: ['H1', 'S1', 'S3']
sdn> query H1 S3
Path: ['H1', 'S1', 'S3']
sdn> query H2 S3
Path: ['H2', 'S2', 'S1', 'S3']
sdn> show_flows
Switch S1 flow table:
dst=S2 | next_hops=[] | prio=normal
dst=S3 | next_hops=[] | prio=normal
dst=S4 | next_hops=['H1', 'S2', 'S3'] | prio=normal
dst=H1 | next_hops=[] | prio=normal
dst=H2 | next_hops=['H1', 'S3', 'S2'] | prio=normal
Switch S2 flow table:
67°F
Heavier rain soon
Search
12:48 PM
5/9/2025


S2
S4
S1
S3