Part1: Execution

- First, open two terminals and cd them into the folder where the topology and the controller is, then run “sudo mn –custom topo.py –topo topo –link tc –controller remote” to build topology in Mininet, then use another terminal to run Ryu-controller typing ”sudo ryu-manager [controller] –observe-links” to start up the controller and monitors the flow passing switch 2. After the flow is added, we can ping from one host to another by ‘’ [host] ping [host]’’.
- Commands explanation

Sudo : to give authority to access some command(an abbreviation for super user do)

Mn : used to build Mininet topology

--custom : read custom classes or params from .py file(s)

--topo : network topology form(single, linear and tree)

--link tc : customized by traffic control

-- controller remote : controlled by outer controller

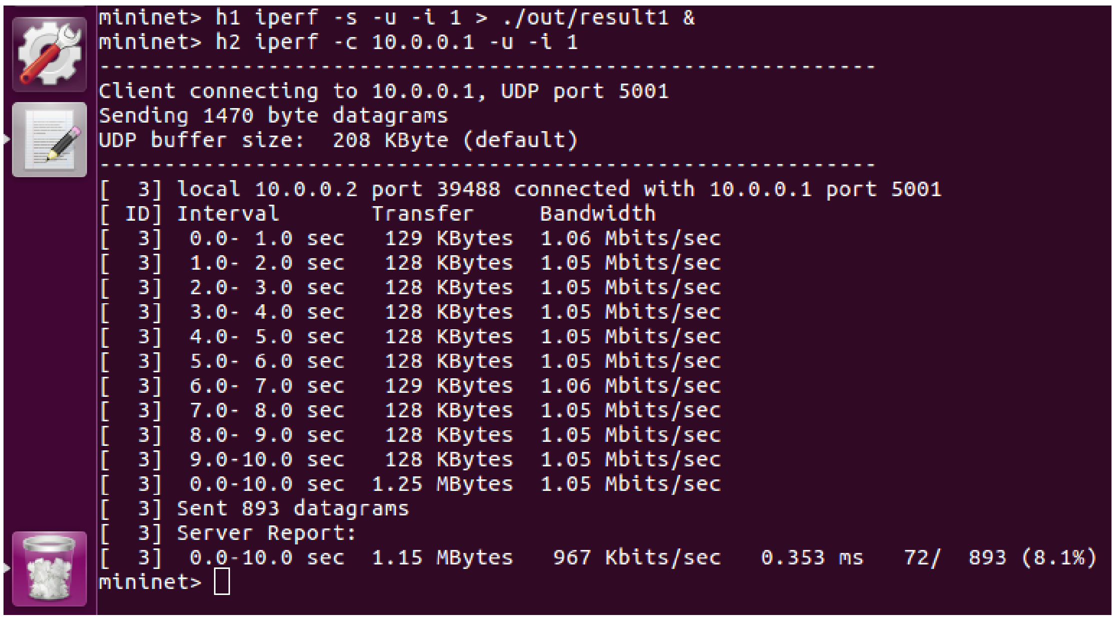
Ryu-manager : the executable for loading Ryu applications and running it

--observe links : to get links from the Mininet topology

- Screenshots

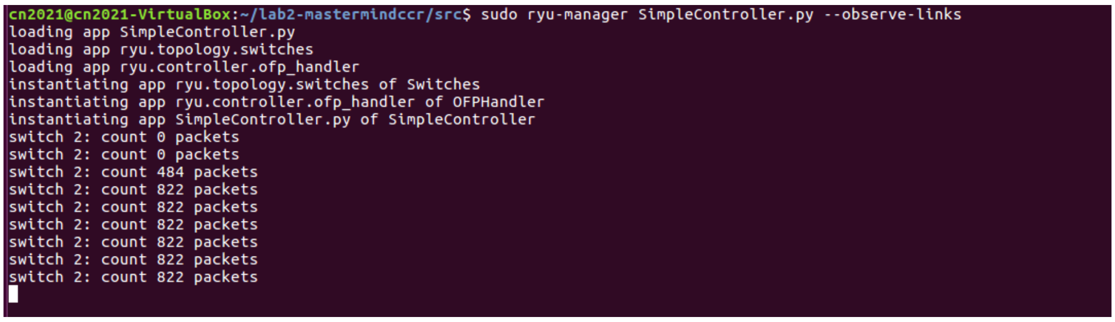
■ Step 2(SimpleController.py)

2-2



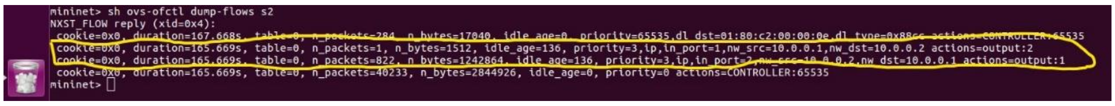
```
mininet> h1 iperf -s -u -i 1 > ./out/result1 &
mininet> h2 iperf -c 10.0.0.1 -u -i 1
------------------------------------------------------------
Client connecting to 10.0.0.1, UDP port 5001
Sending 1470 byte datagrams
UDP buffer size:  208 KByte (default)
------------------------------------------------------------
[  3] local 10.0.0.2 port 39488 connected with 10.0.0.1 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0- 1.0 sec   129 KBytes  1.06 Mbits/sec
[  3]  1.0- 2.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  2.0- 3.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  3.0- 4.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  4.0- 5.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  5.0- 6.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  6.0- 7.0 sec   129 KBytes  1.06 Mbits/sec
[  3]  7.0- 8.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  8.0- 9.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  9.0-10.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  0.0-10.0 sec  1.25 MBytes  1.05 Mbits/sec
[  3] Sent 893 datagrams
[  3] Server Report:
[  3]  0.0-10.0 sec  1.15 MBytes   967 Kbits/sec   0.353 ms   72/  893 (8.1%)
mininet> 
```

2-4



```
cn2021@cn2021-VirtualBox:~/lab2-mastermindccr/src$ sudo ryu-manager SimpleController.py --observe-links
loading app SimpleController.py
loading app ryu.topology.switches
loading app ryu.controller.ofp_handler
instantiating app ryu.topology.switches of Switches
instantiating app ryu.controller.ofp_handler of OFPHandler
instantiating app SimpleController.py of SimpleController
switch 2: count 0 packets
switch 2: count 0 packets
switch 2: count 484 packets
switch 2: count 822 packets
switch 2: count 822 packets
switch 2: count 822 packets
switch 2: count 822 packets
switch 2: count 822 packets
switch 2: count 822 packets
```

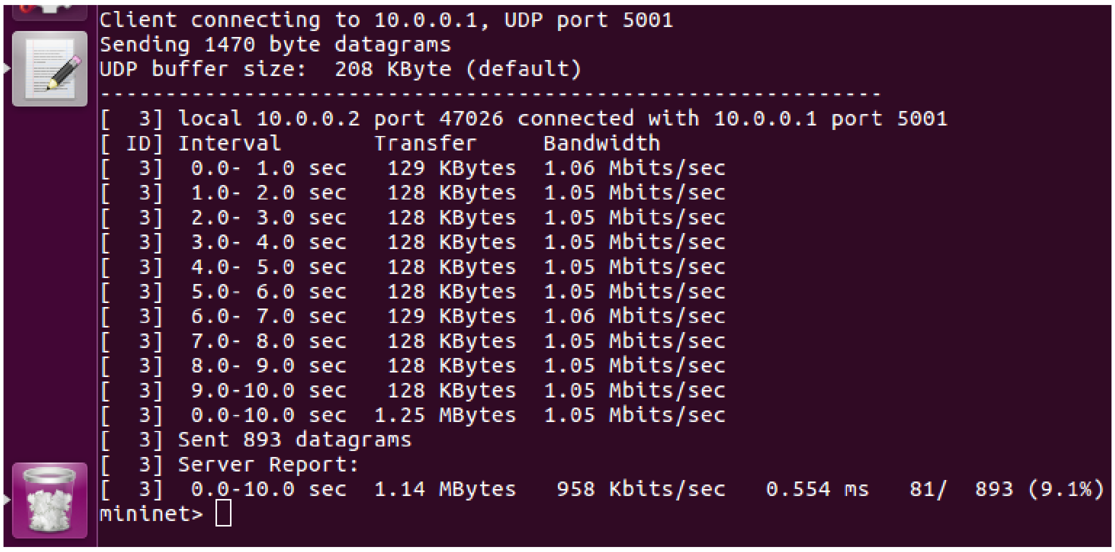
2-5



```
mininet> sh ovs-ofctl dump-flows s2
NXST_FLOW reply (xid=0x4):
 cookie=0x0, duration=167.668s, table=0, n_packets=284, n_bytes=17040, idle_age=0, priority=65535,dl_dst=01:80:c2:00:00:0e,dl_type=0x88cc actions=CONTROLLER:65535
 cookie=0x0, duration=165.669s, table=0, n_packets=1, n_bytes=1512, idle_age=136, priority=3,ip,in_port=1,nw_src=10.0.0.1,nw_dst=10.0.0.2 actions=output:2
 cookie=0x0, duration=165.669s, table=0, n_packets=822, n_bytes=1242864, idle_age=136, priority=3,ip,in_port=2,nw_src=10.0.0.2,nw_dst=10.0.0.1 actions=output:1
 cookie=0x0, duration=165.669s, table=0, n_packets=40233, n_bytes=2844926, idle_age=0, priority=0 actions=CONTROLLER:65535
mininet> 
```

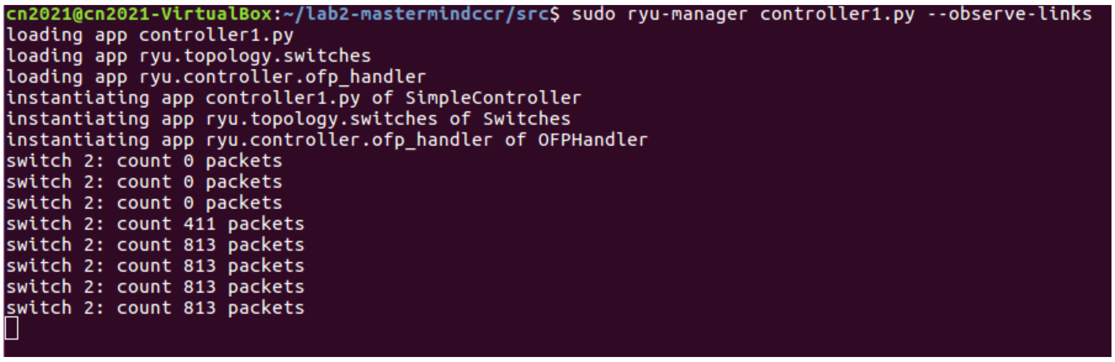
■ Step 4(Controller1.py)

4-2



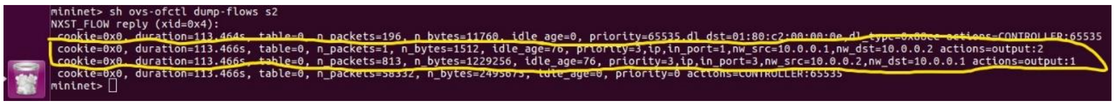
```
Client connecting to 10.0.0.1, UDP port 5001
Sending 1470 byte datagrams
UDP buffer size:  208 KByte (default)
------------------------------------------------------------
[  3] local 10.0.0.2 port 47026 connected with 10.0.0.1 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0- 1.0 sec   129 KBytes  1.06 Mbits/sec
[  3]  1.0- 2.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  2.0- 3.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  3.0- 4.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  4.0- 5.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  5.0- 6.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  6.0- 7.0 sec   129 KBytes  1.06 Mbits/sec
[  3]  7.0- 8.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  8.0- 9.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  9.0-10.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  0.0-10.0 sec  1.25 MBytes  1.05 Mbits/sec
[  3] Sent 893 datagrams
[  3] Server Report:
[  3]  0.0-10.0 sec  1.14 MBytes   958 Kbits/sec   0.554 ms   81/  893 (9.1%)
mininet> 
```

4-4



```
cn2021@cn2021-VirtualBox:~/lab2-mastermindccr/src$ sudo ryu-manager controller1.py --observe-links
loading app controller1.py
loading app ryu.topology.switches
loading app ryu.controller.ofp_handler
instantiating app controller1.py of SimpleController
instantiating app ryu.topology.switches of Switches
instantiating app ryu.controller.ofp_handler of OFPHandler
switch 2: count 0 packets
switch 2: count 0 packets
switch 2: count 0 packets
switch 2: count 411 packets
switch 2: count 813 packets
switch 2: count 813 packets
switch 2: count 813 packets
switch 2: count 813 packets
```

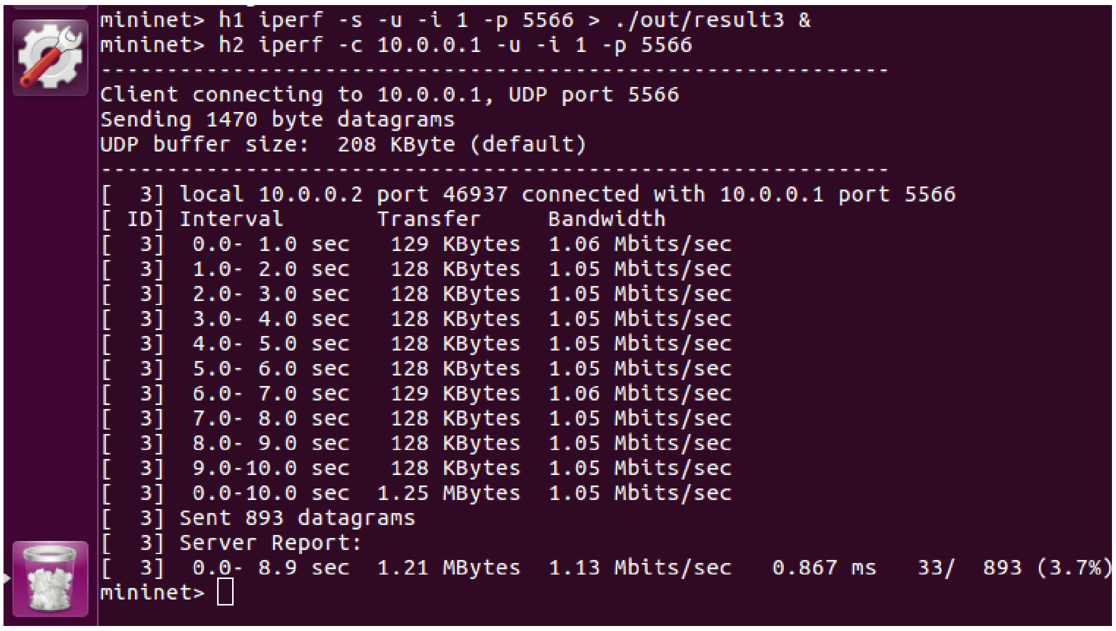
4-5



```
mininet> sh ovs-ofctl dump-flows s2
NXST_FLOW reply (xid=0x4):
 cookie=0x0, duration=113.464s, table=0, n_packets=196, n_bytes=11760, idle_age=0, priority=65535,dl_dst=01:80:c2:00:00:0e,dl_type=0x88cc actions=CONTROLLER:65535
 cookie=0x0, duration=113.466s, table=0, n_packets=1, n_bytes=1512, idle_age=76, priority=3,ip,in_port=1,nw_src=10.0.0.1,nw_dst=10.0.0.2 actions=output:2
 cookie=0x0, duration=113.466s, table=0, n_packets=813, n_bytes=1229256, idle_age=76, priority=3,ip,in_port=3,nw_src=10.0.0.2,nw_dst=10.0.0.1 actions=output:1
 cookie=0x0, duration=113.466s, table=0, n_packets=58332, n_bytes=2495673, idle_age=0, priority=0 actions=CONTROLLER:65535
mininet> 
```

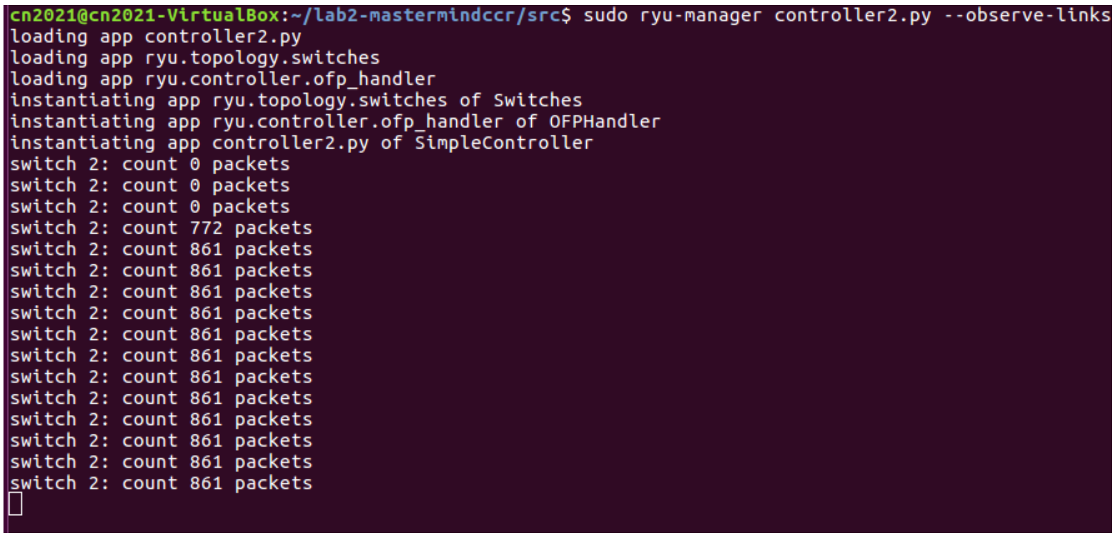
- Step 6

6-2



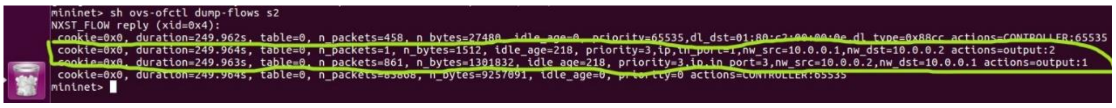
```
mininet> h1 iperf -s -u -i 1 -p 5566 > ./out/result3 &
mininet> h2 iperf -c 10.0.0.1 -u -i 1 -p 5566
------------------------------------------------------------
Client connecting to 10.0.0.1, UDP port 5566
Sending 1470 byte datagrams
UDP buffer size:  208 KByte (default)
------------------------------------------------------------
[  3] local 10.0.0.2 port 46937 connected with 10.0.0.1 port 5566
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0- 1.0 sec   129 KBytes  1.06 Mbits/sec
[  3]  1.0- 2.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  2.0- 3.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  3.0- 4.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  4.0- 5.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  5.0- 6.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  6.0- 7.0 sec   129 KBytes  1.06 Mbits/sec
[  3]  7.0- 8.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  8.0- 9.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  9.0-10.0 sec   128 KBytes  1.05 Mbits/sec
[  3]  0.0-10.0 sec  1.25 MBytes  1.05 Mbits/sec
[  3] Sent 893 datagrams
[  3] Server Report:
[  3]  0.0- 8.9 sec  1.21 MBytes  1.13 Mbits/sec   0.867 ms   33/  893 (3.7%)
mininet> 
```

6-4



```
cn2021@cn2021-VirtualBox:~/lab2-mastermindccr/src$ sudo ryu-manager controller2.py --observe-links
loading app controller2.py
loading app ryu.topology.switches
loading app ryu.controller.ofp_handler
instantiating app ryu.topology.switches of Switches
instantiating app ryu.controller.ofp_handler of OFPHandler
instantiating app controller2.py of SimpleController
switch 2: count 0 packets
switch 2: count 0 packets
switch 2: count 0 packets
switch 2: count 772 packets
switch 2: count 861 packets
switch 2: count 861 packets
switch 2: count 861 packets
switch 2: count 861 packets
switch 2: count 861 packets
switch 2: count 861 packets
switch 2: count 861 packets
switch 2: count 861 packets
switch 2: count 861 packets
switch 2: count 861 packets
switch 2: count 861 packets
switch 2: count 861 packets
```

6-5

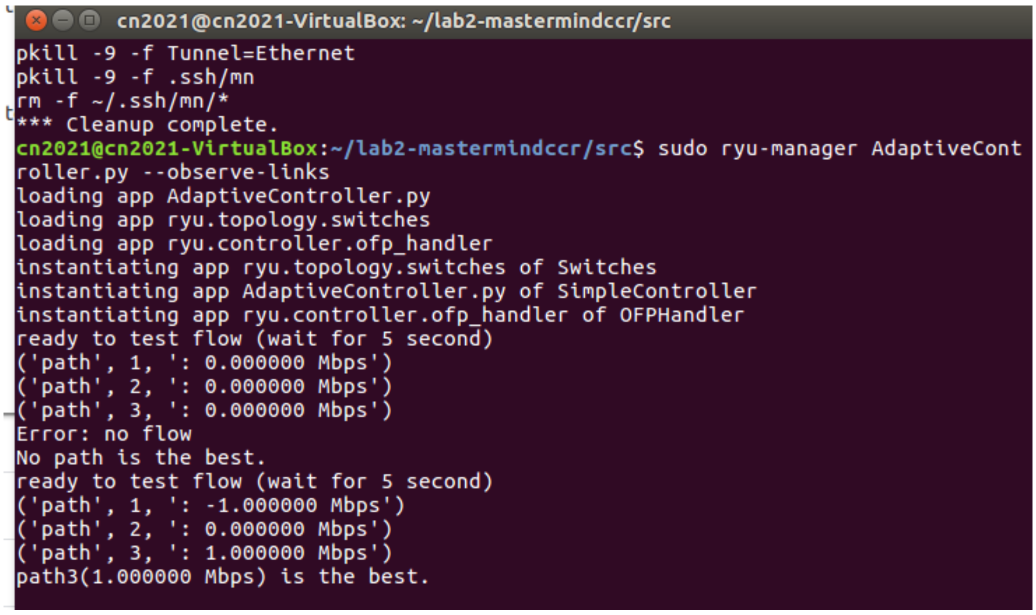


```
mininet> sh ovs-ofctl dump-flows s2
NXST_FLOW reply (xid=0x4):
 cookie=0x0, duration=249.962s, table=0, n_packets=458, n_bytes=27480, idle_age=0, priority=65535,dl_dst=01:80:c2:00:00:0e,dl_type=0x88cc actions=CONTROLLER:65535
 cookie=0x0, duration=249.964s, table=0, n_packets=1, n_bytes=1512, idle_age=218, priority=3,ip,in_port=1,nw_src=10.0.0.1,nw_dst=10.0.0.2 actions=output:2
 cookie=0x0, duration=249.963s, table=0, n_packets=861, n_bytes=1301832, idle_age=218, priority=3,ip,in_port=3,nw_src=10.0.0.2,nw_dst=10.0.0.1 actions=output:1
 cookie=0x0, duration=249.964s, table=0, n_packets=83868, n_bytes=9257091, idle_age=0, priority=0 actions=CONTROLLER:65535
mininet> 
```

Part 3 : Problems encountered

I think the hardest part in this lab is task 6. As a beginner to python as well as unfamiliar to Mininet and Ryu framework, despite doing lots of research, I can't implement my idea into the code and the output result is not as expected. My idea is to use a timer to take turns put on the path flow, measure the bandwidth and then dump the flow using hard_timeout. However, I don't know why the output bandwidth has some negative value and the others are also incorrect.

Screenshot:



```
cn2021@cn2021-VirtualBox: ~/lab2-mastermindccr/src
pkill -9 -f Tunnel=Ethernet
pkill -9 -f .ssh/mn
rm -f ~/.ssh/mn/*
*** Cleanup complete.
cn2021@cn2021-VirtualBox:~/lab2-mastermindccr/src$ sudo ryu-manager AdaptiveCont
roller.py --observe-links
loading app AdaptiveController.py
loading app ryu.topology.switches
loading app ryu.controller.ofp_handler
instantiating app ryu.topology.switches of Switches
instantiating app AdaptiveController.py of SimpleController
instantiating app ryu.controller.ofp_handler of OFPHandler
ready to test flow (wait for 5 second)
('path', 1, ': 0.000000 Mbps')
('path', 2, ': 0.000000 Mbps')
('path', 3, ': 0.000000 Mbps')
Error: no flow
No path is the best.
ready to test flow (wait for 5 second)
('path', 1, ': -1.000000 Mbps')
('path', 2, ': 0.000000 Mbps')
('path', 3, ': 1.000000 Mbps')
path3(1.000000 Mbps) is the best.
```

Discussion

1. Packet-in is the packet that sent into the switch, while packet-out is the packet that sent out from the switch

2. Table miss means there’s no forwarding rule in the flow table for the corresponding flow entry. The priority of table miss entry could be regarded as 0.
3. Adding after the declaration of class means that SimpleController class is a Ryu application that must be inherited from ryu.base.app_manager.RyuApp.
4. Datapath is a class to describe an OpenFlow switch in the topology connected to this controller.
5. Because we assigned the host with IPv4 address and eth type 0x0800 stands for IPv4.
6. The forwarding rule of controller2.py is better as it is configured with the minimum loss rate of 3%, while SimpleController.py has a loss rate of 6% and controller1.py has a loss rate of 9%.