

CSC 541 Project 2 Explanation with Output Screenshots

Name: Jaspreet Singh & Susmita Patange

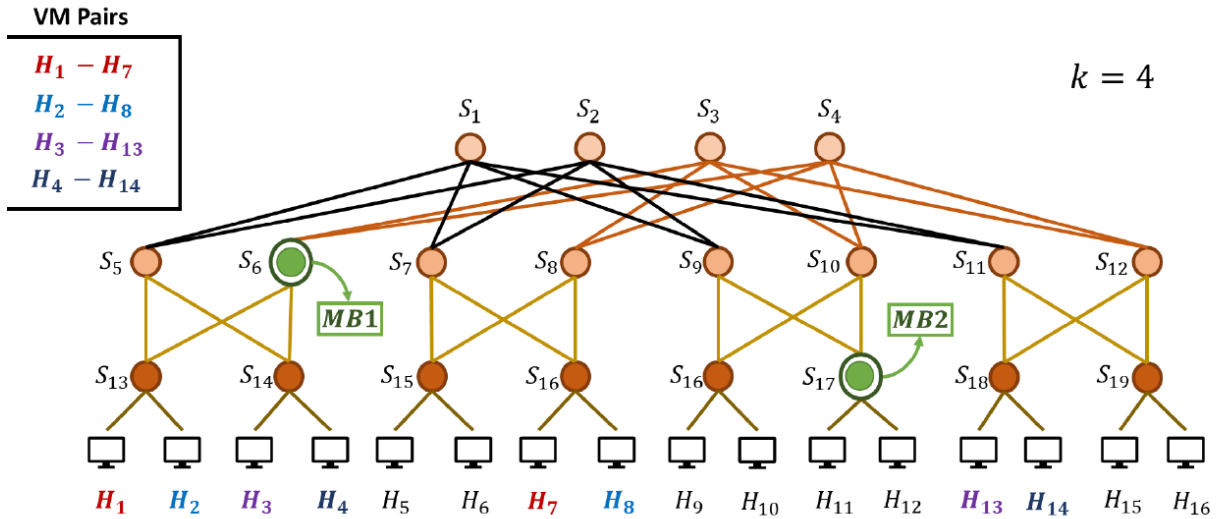
Student ID: 210555347 & 210551057

Email: fjaspreetsingh1@toromail.csudh.edu & spatange1@toromail.csudh.edu

Github: https://github.com/JaspreetToro/CSC541_Project2

S No.	MB Based	VM Based
1.	In this algorithm, for each MB instance, it is assigned κ VM pairs among all the VM pairs that give the minimum energy consumption when going through that MB instance.	For each VM pair, it is assigned to an MB instance such that it gives the minimum energy consumption for this VM pair among all the MB instances, while satisfying this MB instance's capacity.

VM Pairs & VNF Instances



MB Based - We have selected VM pair “H1-H7” and verified shortest path by making them pass each time through MB1 and MB2 as shown in above figure. Based on the above calculation shortest path between the two will be selected. In this case while routing from “H1-H7” path taken will be **[H1 > S13 > S6 > S3 > S8 > S16 > H7]**:

Implementation:

ryu-manager dijkstra_ryu_mb.py --observe-links

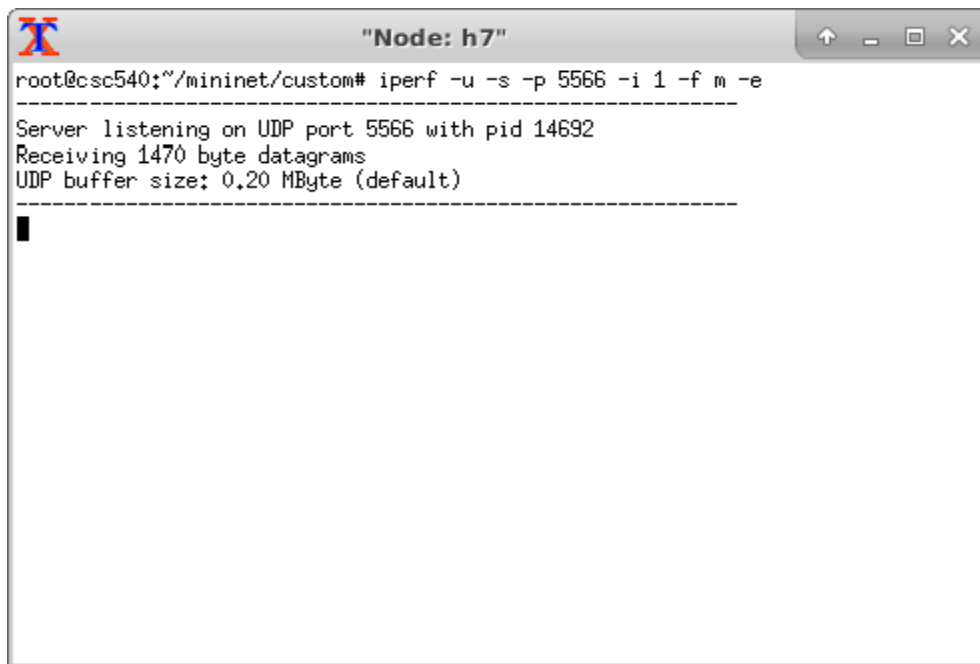
```
cscstudent@csc540: ~/local/lib/python2.7/site-packages/ryu/app
File Edit View Search Terminal Help
path1: [13, 6] [6, 3, 8, 16]
path2: [13, 5, 1, 9, 17] [17, 9, 1, 7, 16]
shortest path: [13, 6] [6, 3, 8, 16]
[(13, 3, 2), (6, 3, 1), (3, 1, 2), (8, 1, 4), (16, 2, 3)]
install_path is called
get_path is called, src= 13 dst= 6 first_port= 3 final_port= 3
get_path is called, src= 6 dst= 16 first_port= 3 final_port= 3
get_path is called, src= 13 dst= 17 first_port= 3 final_port= 3
get_path is called, src= 17 dst= 16 first_port= 3 final_port= 3
path1: [13, 6] [6, 3, 8, 16]
path2: [13, 5, 1, 9, 17] [17, 9, 1, 7, 16]
shortest path: [13, 6] [6, 3, 8, 16]
[(13, 3, 2), (6, 3, 1), (3, 1, 2), (8, 1, 4), (16, 2, 3)]
install_path is called
get_path is called, src= 13 dst= 6 first_port= 3 final_port= 3
get_path is called, src= 6 dst= 16 first_port= 3 final_port= 3
get_path is called, src= 13 dst= 17 first_port= 3 final_port= 3
get_path is called, src= 17 dst= 16 first_port= 3 final_port= 3
path1: [13, 6] [6, 3, 8, 16]
path2: [13, 5, 1, 9, 17] [17, 9, 1, 7, 16]
shortest path: [13, 6] [6, 3, 8, 16]
[(13, 3, 2), (6, 3, 1), (3, 1, 2), (8, 1, 4), (16, 2, 3)]
install_path is called
get_path is called, src= 13 dst= 6 first_port= 3 final_port= 3
get_path is called, src= 6 dst= 16 first_port= 3 final_port= 3
get_path is called, src= 13 dst= 17 first_port= 3 final_port= 3
get_path is called, src= 17 dst= 16 first_port= 3 final_port= 3
path1: [13, 6] [6, 3, 8, 16]
path2: [13, 5, 1, 9, 17] [17, 9, 1, 7, 16]
shortest path: [13, 6] [6, 3, 8, 16]
[(13, 3, 2), (6, 3, 1), (3, 1, 2), (8, 1, 4), (16, 2, 3)]
install_path is called
get_path is called, src= 13 dst= 6 first_port= 3 final_port= 3
get_path is called, src= 6 dst= 16 first_port= 3 final_port= 3
get_path is called, src= 13 dst= 17 first_port= 3 final_port= 3
get_path is called, src= 17 dst= 16 first_port= 3 final_port= 3
path1: [13, 6] [6, 3, 8, 16]
path2: [13, 5, 1, 9, 17] [17, 9, 1, 7, 16]
shortest path: [13, 6] [6, 3, 8, 16]
[(13, 3, 2), (6, 3, 1), (3, 1, 2), (8, 1, 4), (16, 2, 3)]
install_path is called
```

sudo mn --custom fattree4.py --mac --controller=remote --topo mytopo --switch
ovsk,stp=1,protocols=OpenFlow13 --arp

```
cscstudent@csc540: ~/mininet/custom
File Edit View Search Terminal Help
cscstudent@csc540:~/mininet/custom$ sudo mn --custom fattree4.py --mac --controller=remote --topo mytopo --switch ovsk,stp=1,protocols=OpenFlow13 --arp
*** Creating network
*** Adding controller
Connecting to remote controller at 127.0.0.1:6653
*** Adding hosts:
h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11 h12 h13 h14 h15 h16
*** Adding switches:
s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15 s16 s17 s18 s19 s20
*** Adding links:
(s1, s5) (s1, s7) (s1, s9) (s1, s11) (s2, s5) (s2, s7) (s2, s9) (s2, s11) (s3, s6) (s3, s8) (s3, s10) (s3, s12) (s4, s6) (s4, s8) (s4, s10) (s4, s12) (s5, s13)
(s5, s14) (s6, s13) (s6, s14) (s7, s15) (s7, s16) (s8, s15) (s8, s16) (s9, s17) (s9, s18) (s10, s17) (s10, s18) (s11, s19) (s11, s20) (s12, s19) (s12, s20)
(s13, h1) (s13, h2) (s14, h3) (s14, h4) (s15, h5) (s15, h6) (s16, h7) (s16, h8) (s17, h9) (s17, h10) (s18, h11) (s18, h12) (s19, h13) (s19, h14) (s20, h15) (s
20, h16)
*** Configuring hosts
h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11 h12 h13 h14 h15 h16
*** Starting controller
c0
*** Starting 20 switches
s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15 s16 s17 s18 s19 s20 ...
*** Starting CLI:
mininet> h1 ping h7
PING 10.0.0.7 (10.0.0.7) 56(84) bytes of data.
```

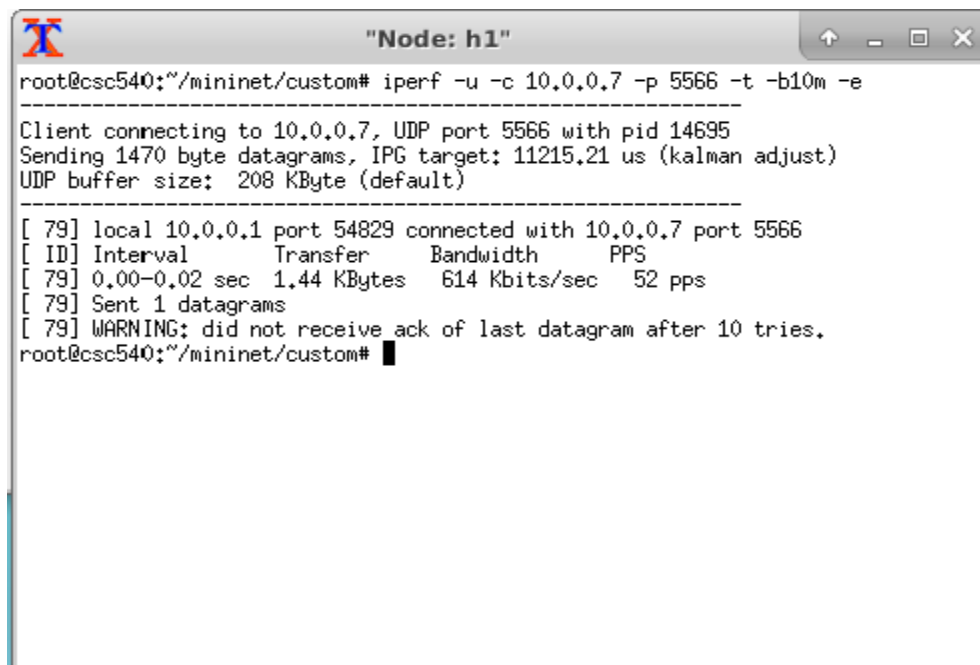
xterm h1 h7

iperf -u -s -p 5566 -i 1 -f m -e



```
root@csc540:~/mininet/custom# iperf -u -s -p 5566 -i 1 -f m -e
-----
Server listening on UDP port 5566 with pid 14692
Receiving 1470 byte datagrams
UDP buffer size: 0.20 MByte (default)
-----
█
```

iperf -u -c 10.0.0.7 -p 5566 -t 300 -b10m -e



```
root@csc540:~/mininet/custom# iperf -u -c 10.0.0.7 -p 5566 -t -b10m -e
-----
Client connecting to 10.0.0.7, UDP port 5566 with pid 14695
Sending 1470 byte datagrams, IPG target: 11215.21 us (kalman adjust)
UDP buffer size: 208 KByte (default)
-----
[ 79] local 10.0.0.1 port 54829 connected with 10.0.0.7 port 5566
[ ID] Interval      Transfer    Bandwidth   PPS
[ 79] 0.00-0.02 sec  1.44 KBytes  614 Kbits/sec  52 pps
[ 79] Sent 1 datagrams
[ 79] WARNING: did not receive ack of last datagram after 10 tries.
root@csc540:~/mininet/custom# █
```

sudo apt-get install gnuplot-qt

//installing gnuplot library

```
root@csc540:~/mininet/custom# sudo apt-get install gnuplot-qt
Reading package lists... Done
Building dependency tree
Reading state information... Done
gnuplot-qt is already the newest version (5.2.2+dfsg1-2ubuntu1).
The following packages were automatically installed and are no longer required:
  linux-headers-4.15.0-55 linux-headers-4.15.0-55-generic
  linux-image-4.15.0-55-generic linux-modules-4.15.0-55-generic
  linux-modules-extra-4.15.0-55-generic
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 111 not upgraded.
root@csc540:~/mininet/custom#
```

```
root@csc540:~/mininet/custom# gnuplot

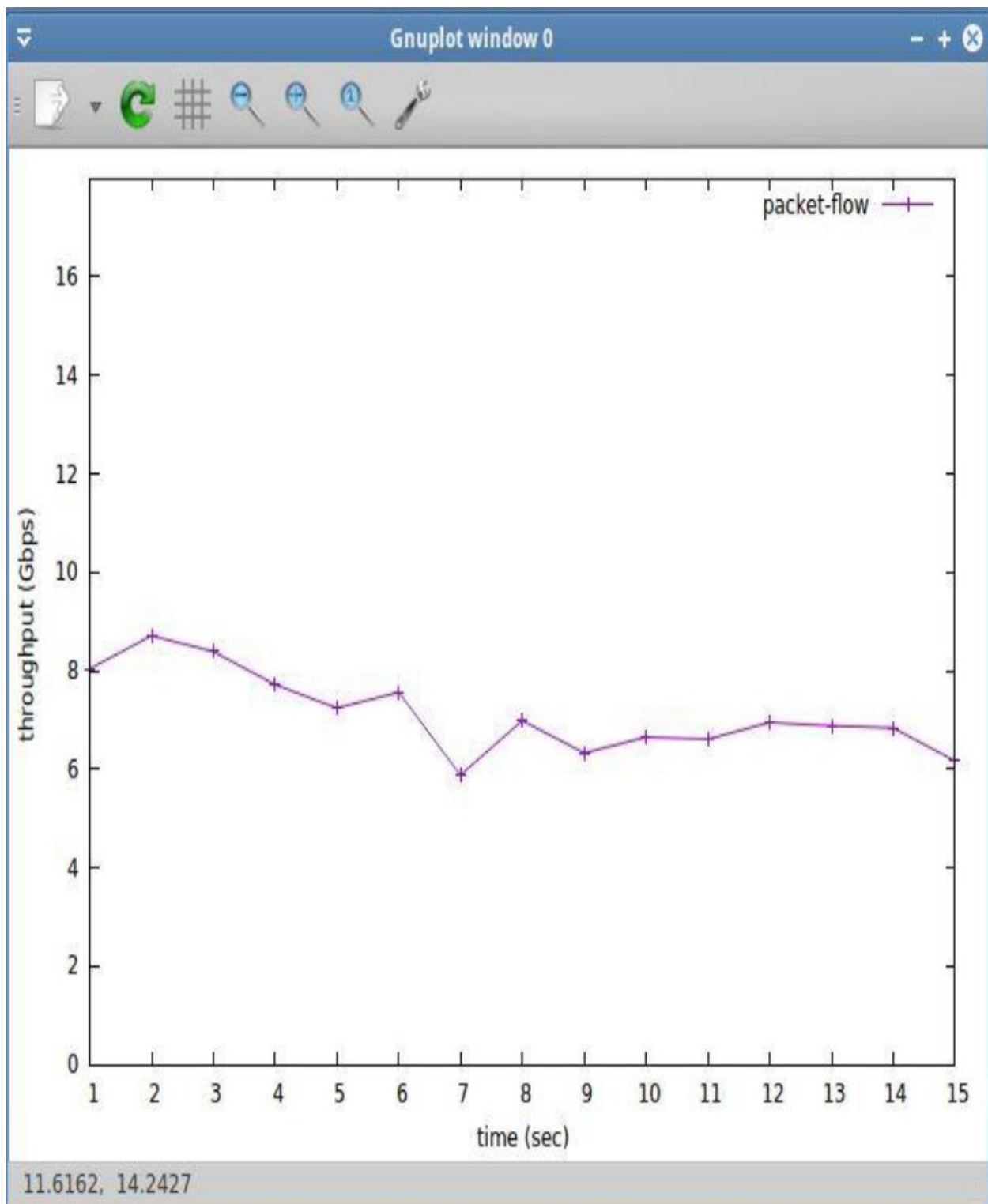
  G N U P L O T
  Version 5.2 patchlevel 2   last modified 2017-11-01

  Copyright (C) 1986-1993, 1998, 2004, 2007-2017
  Thomas Williams, Colin Kelley and many others

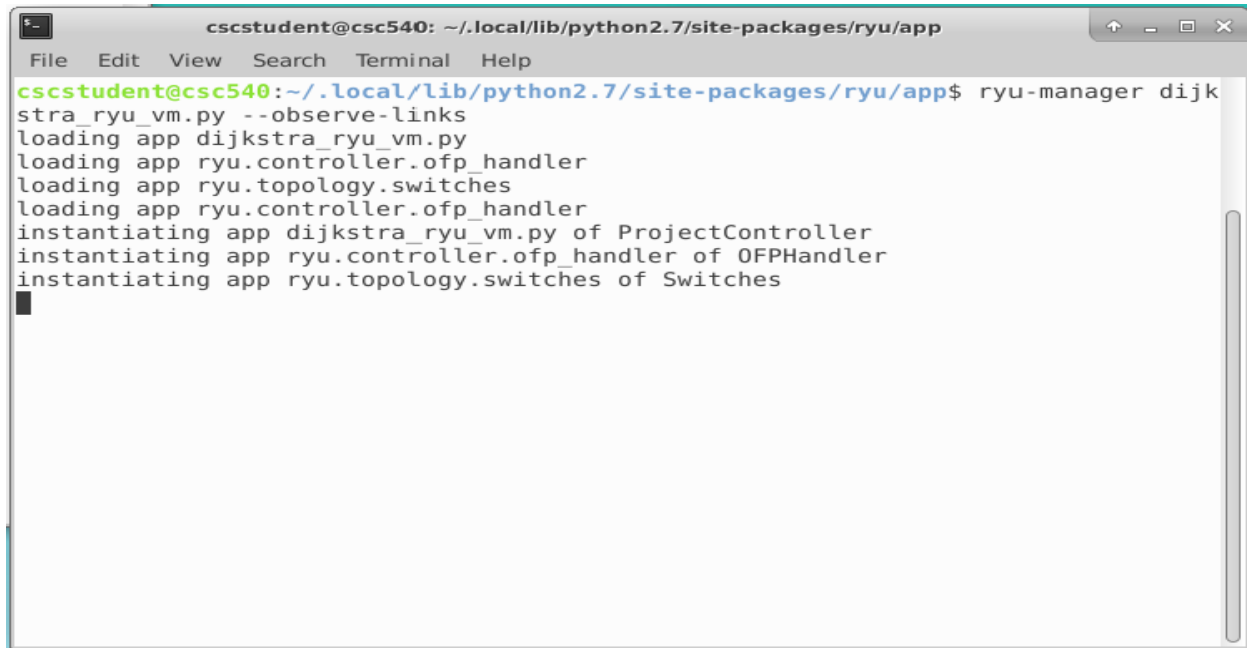
  gnuplot home:      http://www.gnuplot.info
  faq, bugs, etc:    type "help FAQ"
  immediate help:    type "help" (plot window: hit 'h')

Terminal type is now 'qt'
gnuplot>
```

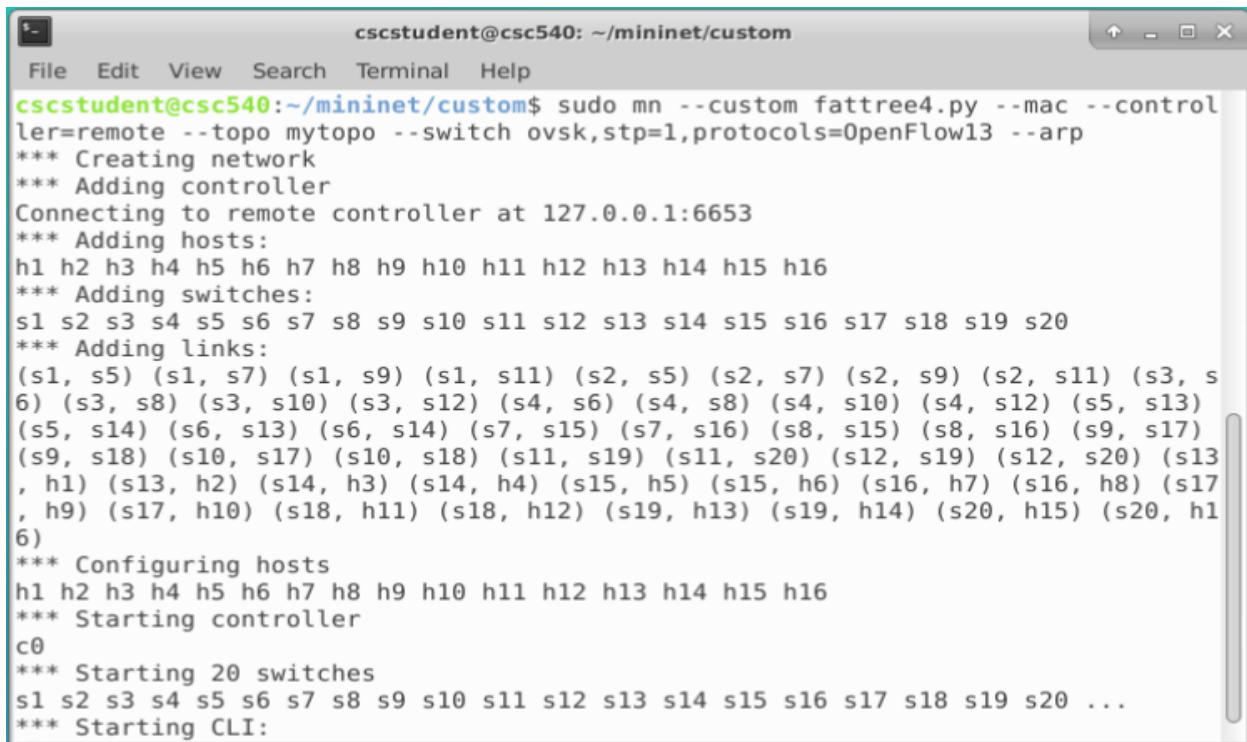
Results:



VM Based - We have selected VM pair “H1-H7” and assigned MB1 to this pair. In any case “H1-H7” will be routed through MB1 i.e. switch 6. This is also applicable to VM pair “H2-H8” via MB1 i.e. switch 6. Similarly, “H3-H13” VM pair which is routed through MB2 i.e. Switch 17. Below are the screenshots for the same:



```
cscstudent@csc540: ~/.local/lib/python2.7/site-packages/ryu/app
File Edit View Search Terminal Help
cscstudent@csc540:~/.local/lib/python2.7/site-packages/ryu/app$ ryu-manager dijkstra_ryu_vm.py --observe-links
loading app dijkstra_ryu_vm.py
loading app ryu.controller.ofp_handler
loading app ryu.topology.switches
loading app ryu.controller.ofp_handler
instantiating app dijkstra_ryu_vm.py of ProjectController
instantiating app ryu.controller.ofp_handler of OFPHandler
instantiating app ryu.topology.switches of Switches
```



```
cscstudent@csc540: ~/mininet/custom
File Edit View Search Terminal Help
cscstudent@csc540:~/mininet/custom$ sudo mn --custom fattree4.py --mac --controller=remote --topo mytopo --switch ovsk,stp=1,protocols=OpenFlow13 --arp
*** Creating network
*** Adding controller
Connecting to remote controller at 127.0.0.1:6653
*** Adding hosts:
h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11 h12 h13 h14 h15 h16
*** Adding switches:
s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15 s16 s17 s18 s19 s20
*** Adding links:
(s1, s5) (s1, s7) (s1, s9) (s1, s11) (s2, s5) (s2, s7) (s2, s9) (s2, s11) (s3, s6) (s3, s8) (s3, s10) (s3, s12) (s4, s6) (s4, s8) (s4, s10) (s4, s12) (s5, s13) (s5, s14) (s6, s13) (s6, s14) (s7, s15) (s7, s16) (s8, s15) (s8, s16) (s9, s17) (s9, s18) (s10, s17) (s10, s18) (s11, s19) (s11, s20) (s12, s19) (s12, s20) (s13, h1) (s13, h2) (s14, h3) (s14, h4) (s15, h5) (s15, h6) (s16, h7) (s16, h8) (s17, h9) (s17, h10) (s18, h11) (s18, h12) (s19, h13) (s19, h14) (s20, h15) (s20, h16)
*** Configuring hosts
h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11 h12 h13 h14 h15 h16
*** Starting controller
c0
*** Starting 20 switches
s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15 s16 s17 s18 s19 s20 ...
*** Starting CLI:
```

```
cscstudent@csc540: ~/mininet/custom
File Edit View Search Terminal Help

*** Adding controller
Connecting to remote controller at 127.0.0.1:6653
*** Adding hosts:
h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11 h12 h13 h14 h15 h16
*** Adding switches:
s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15 s16 s17 s18 s19 s20
*** Adding links:
(s1, s5) (s1, s7) (s1, s9) (s1, s11) (s2, s5) (s2, s7) (s2, s9) (s2, s11) (s3, s
6) (s3, s8) (s3, s10) (s3, s12) (s4, s6) (s4, s8) (s4, s10) (s4, s12) (s5, s13)
(s5, s14) (s6, s13) (s6, s14) (s7, s15) (s7, s16) (s8, s15) (s8, s16) (s9, s17)
(s9, s18) (s10, s17) (s10, s18) (s11, s19) (s11, s20) (s12, s19) (s12, s20) (s13
, h1) (s13, h2) (s14, h3) (s14, h4) (s15, h5) (s15, h6) (s16, h7) (s16, h8) (s17
, h9) (s17, h10) (s18, h11) (s18, h12) (s19, h13) (s19, h14) (s20, h15) (s20, h1
6)
*** Configuring hosts
h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11 h12 h13 h14 h15 h16
*** Starting controller
c0
*** Starting 20 switches
s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15 s16 s17 s18 s19 s20 ...
*** Starting CLI:
mininet> h1 ping h7
PING 10.0.0.7 (10.0.0.7) 56(84) bytes of data.
```

```
cscstudent@csc540: ~/.local/lib/python2.7/site-packages/ryu/app
File Edit View Search Terminal Help

get_path is called, src= 13 dst= 6 first_port= 3 final_port= 3
get_path is called, src= 6 dst= 16 first_port= 3 final_port= 3
path: [13, 6, 3, 8, 16]
[(13, 3, 2), (6, 3, 1), (3, 1, 2), (8, 1, 4), (16, 2, 3)]
install_path is called
get_path is called, src= 13 dst= 6 first_port= 3 final_port= 3
get_path is called, src= 6 dst= 16 first_port= 3 final_port= 3
path: [13, 6, 3, 8, 16]
[(13, 3, 2), (6, 3, 1), (3, 1, 2), (8, 1, 4), (16, 2, 3)]
install_path is called
get_path is called, src= 13 dst= 6 first_port= 3 final_port= 3
get_path is called, src= 6 dst= 16 first_port= 3 final_port= 3
path: [13, 6, 3, 8, 16]
[(13, 3, 2), (6, 3, 1), (3, 1, 2), (8, 1, 4), (16, 2, 3)]
install_path is called
get_path is called, src= 13 dst= 6 first_port= 3 final_port= 3
get_path is called, src= 6 dst= 16 first_port= 3 final_port= 3
path: [13, 6, 3, 8, 16]
[(13, 3, 2), (6, 3, 1), (3, 1, 2), (8, 1, 4), (16, 2, 3)]
```

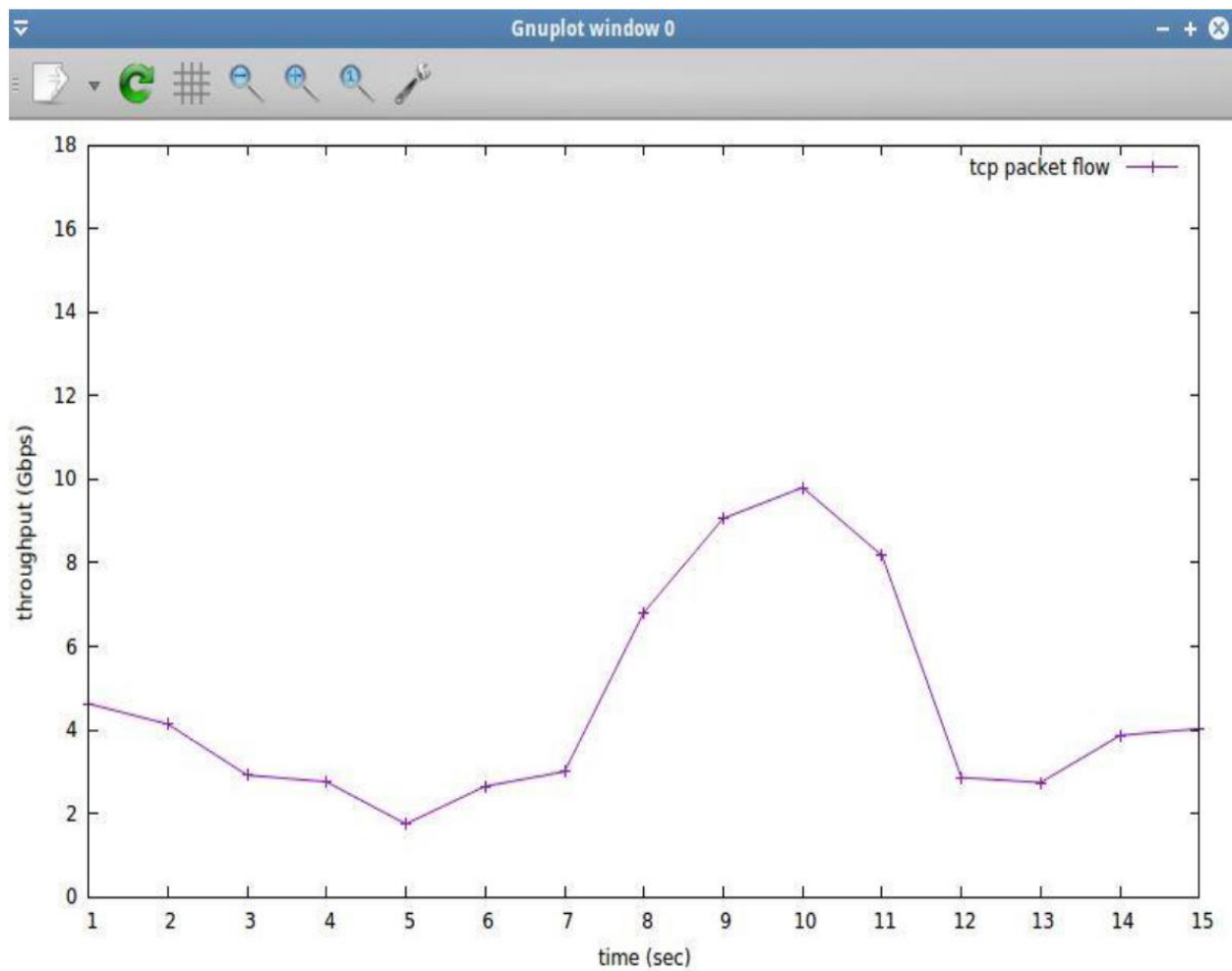


```
cscstudent@csc540: ~/mininet/custom
File Edit View Search Terminal Help
s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15 s16 s17 s18 s19 s20
*** Adding links:
(s1, s5) (s1, s7) (s1, s9) (s1, s11) (s2, s5) (s2, s7) (s2, s9) (s2, s11) (s3, s
6) (s3, s8) (s3, s10) (s3, s12) (s4, s6) (s4, s8) (s4, s10) (s4, s12) (s5, s13)
(s5, s14) (s6, s13) (s6, s14) (s7, s15) (s7, s16) (s8, s15) (s8, s16) (s9, s17)
(s9, s18) (s10, s17) (s10, s18) (s11, s19) (s11, s20) (s12, s19) (s12, s20) (s13
, h1) (s13, h2) (s14, h3) (s14, h4) (s15, h5) (s15, h6) (s16, h7) (s16, h8) (s17
, h9) (s17, h10) (s18, h11) (s18, h12) (s19, h13) (s19, h14) (s20, h15) (s20, h1
6)
*** Configuring hosts
h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11 h12 h13 h14 h15 h16
*** Starting controller
c0
*** Starting 20 switches
s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15 s16 s17 s18 s19 s20 ...
*** Starting CLI:
mininet> h1 ping h7
PING 10.0.0.7 (10.0.0.7) 56(84) bytes of data.
^X^C
--- 10.0.0.7 ping statistics ---
243 packets transmitted, 0 received, 100% packet loss, time 247810ms

mininet> xterm h1 h7
mininet>
```

```
"Node: h7"
root@csc540:~/mininet/custom# iperf -u -s -p 5566 -i 1 -f m -e
-----
Server listening on UDP port 5566 with pid 29361
Receiving 1470 byte datagrams
UDP buffer size: 0.20 MByte (default)
-----
█
```

```
"Node: h1"
root@csc540:~/mininet/custom# iperf -u -c 10.0.0.7 -p 5566 -t 300 -b10m -e
-----
Client connecting to 10.0.0.7, UDP port 5566 with pid 32431
Sending 1470 byte datagrams, IPG target: 1176.00 us (kalman adjust)
UDP buffer size: 208 KByte (default)
-----
[ 79] local 10.0.0.1 port 33663 connected with 10.0.0.7 port 5566
[ ID] Interval      Transfer      Bandwidth      PPS
[ 79] 0.00-300.00 sec  358 MBytes  10.0 Mbits/sec  850 pps
[ 79] Sent 255097 datagrams
[ 79] WARNING: did not receive ack of last datagram after 10 tries.
root@csc540:~/mininet/custom# █
```

Contribution:

Susmita: Research on MB Based, VM Based Algorithms.

Jaspreet Singh: Implementation of MB Based and VM Based Algorithms.

References:

- [1] http://csie.nqu.edu.tw/smallko/sdn/dijkstra_ryu.htm
- [2] http://csie.nqu.edu.tw/smallko/sdn/iperf_mininet.htm