Assignment 1: Sockets, Mininet, & Performance

Group members: Shang-Yen Yeh, Yifan Hong

Part 1

Testing Iperfer on Wired Environment:

- Client Screenshot:

```
[syeh@royal-06] (4)$ java Iperfer -c -h 128.105.37.145 -p 10000 -t 30
sent=3530380 KB rate=941.0 Mbps
```

- Server Screenshot:

```
[yifanh@royal-05] (8)$ make
javac -g Iperfer.java
[yifanh@royal-05] (9)$ ls
IperfClient.class  Iperfer.class  IperfServer.class  Makefile
IperfClient.java   Iperfer.java   IperfServer.java
[yifanh@royal-05] (10)$ ifconfig
enp2s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 128.105.37.145  netmask 255.255.255.128  broadcast 128.105.37.255
        inet6 fe80::fabc:12ff:fe93:2458  prefixlen 64  scopeid 0x20<link>
        ether f8:bc:12:93:24:58  txqueuelen 1000  (Ethernet)
        RX packets 14376139  bytes 15209073810 (15.2 GB)
        RX errors 0  dropped 125  overruns 0  frame 0
        TX packets 9791467  bytes 4904265621 (4.9 GB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 1985389  bytes 1029972569 (1.0 GB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1985389  bytes 1029972569 (1.0 GB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[yifanh@royal-05] (11)$ java Iperfer -s -p 10000
received=3530380 KB rate=941.0 Mbps
```

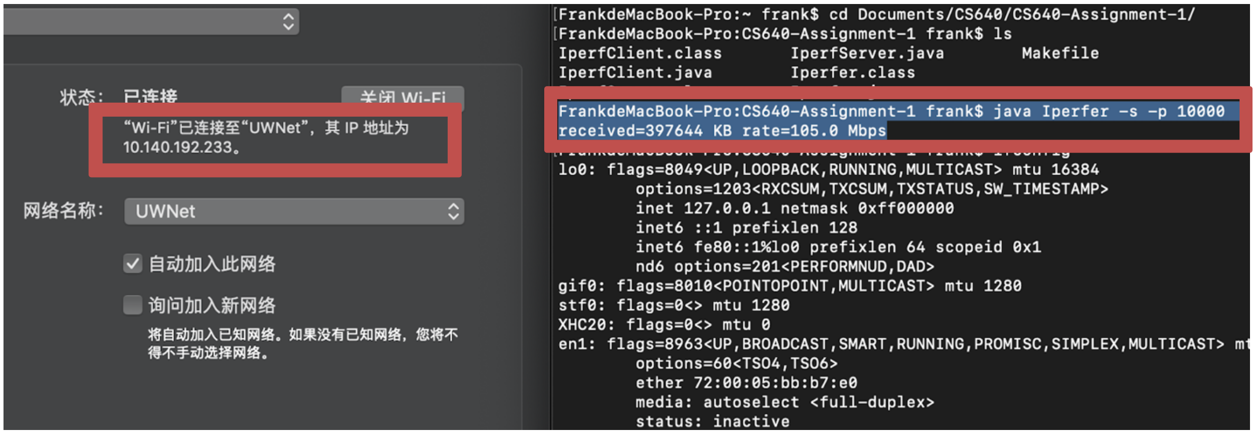
Wireless Environment:

- Under which environment - wired or wireless, would you expect greater throughput? Wired environment. (Assume that wired and wireless environments are both connected to the same network). Due to power limitation and limitation and the signal redundancy, the capacity in wireless is meant to be smaller than the wired environment.

- Client Screenshot:

```
yeshangyande-MacBook-Pro:CS640-Assignment-1 sandersyeh$ java Iperfer -c -h 10.140.192.233 -p 10000 -t 30
sent=397644 KB rate=105.0 Mbps
```

- Server Screenshot



- Did it match the prediction you made at the start of this section? In either case, explain your results:
  It matched our prediction.
  Explanation: 1. Wireless network works on 802.11 technology with a limitation of maximum supported rate.
  2. CSMA/CA mechanism will also reduce the throughput.
  3. Signal interference cannot be ignored.

Testing on Mininet:

- Latency:

```
mininet> h1 ping 10.0.0.2 -c 5
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=20.4 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=20.0 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=20.0 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=20.0 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=20.0 ms

--- 10.0.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 20.077/20.161/20.469/0.178 ms
mininet> h2 ping 10.0.0.3 -c 5
PING 10.0.0.3 (10.0.0.3) 56(84) bytes of data.
64 bytes from 10.0.0.3: icmp_seq=1 ttl=64 time=32.3 ms
64 bytes from 10.0.0.3: icmp_seq=2 ttl=64 time=31.9 ms
64 bytes from 10.0.0.3: icmp_seq=3 ttl=64 time=30.3 ms
64 bytes from 10.0.0.3: icmp_seq=4 ttl=64 time=30.0 ms
64 bytes from 10.0.0.3: icmp_seq=5 ttl=64 time=30.1 ms

--- 10.0.0.3 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 30.075/30.960/32.315/0.972 ms
mininet> h3 ping 10.0.0.4 -c 5
PING 10.0.0.4 (10.0.0.4) 56(84) bytes of data.
64 bytes from 10.0.0.4: icmp_seq=1 ttl=64 time=41.8 ms
64 bytes from 10.0.0.4: icmp_seq=2 ttl=64 time=41.6 ms
64 bytes from 10.0.0.4: icmp_seq=3 ttl=64 time=40.3 ms
64 bytes from 10.0.0.4: icmp_seq=4 ttl=64 time=40.0 ms
64 bytes from 10.0.0.4: icmp_seq=5 ttl=64 time=40.0 ms

--- 10.0.0.4 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 40.065/40.803/41.891/0.847 ms
```

  - According to the screenshot, when h1 run ping to h2, the average RTT is 20.161ms, so the single way propagation delay is around 10.08ms. Since the

packets are small in this case, the latency is dominated by propagation delay, which matches the expected latency for L1 with only a little deviation
  - The same conclusion can also apply to L2 and L3 in our test, the propagation delay is 15.48 ms and 20.4 ms respectively.
- Throughput
  - L1:
    - `sent=81748 KB rate=21.0 Mbps`
    - `received=81748 KB rate=19.0 Mbps`
    - The average from server and client is 20.0 Mbps, which meets the expected throughput.
  - L2:
    - `sent=40879 KB rate=10.0 Mbps`
    - `received=40879 KB rate=9.0 Mbps`
    - The average from server and client is 9.5 Mbps, which meets the expected throughput with little deviation.
  - L3:
    - `sent=79684 KB rate=19.0 Mbps`
    - `received=79684 KB rate=18.0 Mbps`
    - The average from server and client is 18.5 Mbps, which meets the expected throughput with little deviation.

Part 3

Q1 Link Latency and Throughput:

- Latency: see txt files latency_L1.txt - latency_L5.txt
- Throughput: see txt files throughput_L1.txt - throughput_L5.txt

	L1	L2	L3	L4	L5
avg RTT/ms	80	20	40	30	30
throughput/Mbps	20	50	40	30	30

Q2 Path Latency and Throughput:

- Prediction: RTT = RTT_L1+RTT_L2+RTT_L3 = 140ms
  Throughput = min{Throughput_L1, Throughput_L2, Throughput_L3}
  = 19.0 Mbps
- Latency: see txt files latency_Q2.txt, avg RTT = 140.442ms
- Throughput: see txt files throughput_Q2.txt, throughput = 18.0Mbps
- Explanation:
  - Average RTT: In this case, since N(numbers of bit) is small and C(bandwidth) is vary large, D(propagation delay= distance/speed of light) matters a lot. Total latency = Total distance/light speed = The sum of each link's delay.
  - Measured throughput: Since the throughput for L1, L2, and L3 are different, the total throughput should be decided by the smallest throughput. Even if there are other links with greater throughput, the smallest throughput can't be bigger, so the smallest throughput will be the limitation for this path.

Q3 Effects of Multiplexing

- Prediction: RTT_2pairs = RTT_3pairs =140ms
  Throughput_2pairs = min{Throughput_L1, Throughput_L2, Throughput_L3}/2
  = 10.0 Mbps
  Throughput_3pairs = min{Throughput_L1, Throughput_L2, Throughput_L3}/3
  = 6.7 Mbps

- Latency: avg RTT_2pairs_1 = 140.630ms, avg RTT_2pairs_2 = 140.710ms
  avg RTT_3pairs_1 = 140.767ms, avg RTT_3pairs_2 = 140.699ms
  avg RTT_3pairs_3 = 140.705ms
- Throughput: Throughput_2pairs_1 = 12.5 Mbps Throughput_2pairs_2 = 7.0 Mbps
  Throughput_3pairs_1 = 8.5 Mbps Throughput_3pairs_2 = 4.0 Mbps
  Throughput_3pairs_3 = 6.5 Mbps
- Explanation:
  - Average RTT: Even if bandwidth decline, D is still the main factor. Thus, the latency under multiplexing is the same as Q2.
  - Measured throughput: In this case, different connections share same links, switches will divide bandwidth equally when multiplexing. Thus, the throughput in Q3 equals the throughput in Q2 divided by numbers of pairs. However, the throughput of the first pair is always slightly bigger than others in both cases, it is because we start the Iperfer manually and the time difference always exists. Hence, there will be a little amount of time that the first pair of hosts occupy the whole bandwidth only by itself and make its throughput bigger.

Q4 Effects of Latency

- Prediction: RTT_h1-h4 = RTT_L1+RTT_L2+RTT_L3 = 140ms
  RTT_h4-h5 = RTT_L4+RTT_L2+RTT_L5 = 80ms
  Throughput_h1-h4 = 20 Mbps
  Throughput_h4-h5 = 30 Mbps
- Latency: avg RTT_h1-h4 = 140.571ms
  avg RTT_h4-h5 = 80.544ms,
- Throughput: Throughput_h1-h4 = 18.0 Mbps
  Throughput_h4-h5 = 29.0 Mbps
- Explanation:
  - Average RTT: The result of latency is according with explanation in Q2 and Q3.
  - Measured throughput: Without connection h1-h4, throughput between h4 and h5 should be 30Mbps. Without connection h4-h5, throughput between h1 and h2 should be 20Mbps.Notice that the throughput of L2 (the link two connections share) is 50Mbps, equals to 20Mbps+30Mbps. Thus, in this case, throughput_h1-h4 is 20 Mbps and throughput_h4-h5 is 30 Mbps, even if they communicate simultaneously.