PART1 predictions

THe wired connection would have a much faster throughput than the wireless connection.

Part1-Wired Environment

Server

```
[krumbach@royal-29] (20)$ java Iperfer -s -p 3000
rate= 10.217 Mbps
received=12795 KB
[krumbach@royal-29] (21)$
```

Client

```
[krumbach@royal-28] (9)$ java Iperfer -c -h 128.105.37.169 -p 3000 -t 10
rate= 10.236 Mbps
Sent=12795 KB
[krumbach@royal-28] (10)$
```

Part 1 - Wireless Environment

Server

```
gagekrumbach@Gages-MBP src % java Iperfer -s -p 3000
rate= 9.405 Mbps
received=3609 KB
gagekrumbach@Gages-MBP src %
```

Client

```
/home/abhiram/cs_640/Lab1/src>java Iperfer -c -h 192.168.1.89 -p 3000 -t 3
rate= 9.624 Mbps
Sent=3609 KB
```

Part 1 - Iperfer Results

The screenshots are above. The throughput was faster for the wired connection. However, it is not as much as we thought it would be. The wireless connection was done using the local router instead of UWNet. We aso only ran the client for 3 seconds. These could be reasons why there isn’t as much of a difference as we thought would be.

Part 3 - Q2 Predictions

I Believe the latency will be a sum of the latency of the three links as observed in Question1. The Throughput will be an average of the three throughputs of the three links.

Part 3 - Q2 Results

Average RTT - 141.842 ms
Throughput - 12.088 Mbps

The Average Rtt is about the same as what is predicted. This makes sense because The distance between the two hosts is the sum of the intermediary Links. The Throughput is higher than the predicted value(11.65 Mbs was predicted). This is probably because the overhead is lower in one connection than three individual ones.

Part 3 - Q3 Predictions

With two pairs of hosts, I believe that the latency will not change because the latency is taken when the host is using the link so it won't be affected if it gets blocked by a busy line a few times. With three hosts we will see the same latency results. For the rate of transfer, I expect the rate to go down with more hosts. The rates should all be about the same as well because statistically each client should be blocked by a busy link about the same amount of times as any other host.

Part3 - Q3 Results

I was correct in thinking that the latency would be similar among all the host pairs. There was a ~2 ms difference with two pairs and ~3 ms difference with three pairs. That speed is marginal enough to assume it wasn't affected by the increase in pairs. As for the rate, I was also correct in thinking that the speed would decrease with the increase of host pairs. There was about a 6 Mnps decrease for two of the pairs, but only a ~2 Mbps decrease for one of the pairs. I believe this is due to the fact that I couldn't run all the client/server pairs at once and the pair that had the smallest difference was the pair I ran first in the terminal. This gave it some time alone with the link which would have affected the average.

Two Pairs:

- **h1 -> h4:**
  - rtt avg = 144.372 ms
  - Latency = 72.186 ms
  - **Client h1:**
    - rate= 8.885 Mbps
    - Sent=33318 KB
  - **Server h4:**
    - rate= 8.566 Mbps
    - received=33318 KB
- **h7 -> h9:**
  - rtt avg = 142.886 ms
  - Latency = 71.443 ms
  - **Client h7:**
    - rate= 8.685 Mbps
    - Sent=32567 KB
  - **Server h9:**
    - rate= 8.436 Mbps
    - received=32567 KB

Three Pairs:

- **h1 -> h4:**
  - rtt avg = 145.452 ms
  - Latency = 72.726 ms
  - **Client h1:**
    - rate= 2.139 Mbps
    - Sent=8020 KB
  - **Server h4:**
    - rate= 2.124 Mbps
    - received=8020 KB
- **h7 -> h9:**
  - rtt avg = 144.001 ms
  - Latency = 72.0005 ms
  - **Client h7:**
    - rate= 2.466 Mbps
    - Sent=9247 KB
  - **Server h9:**
    - rate= 2.367 Mbps
    - received=9247 KB
- **h8 -> h10:**
  - rtt avg = 142.619 ms
  - Latency = 71.3095 ms
  - **Client h8:**
    - rate= 7.315 Mbps

        - Sent=27433 KB
    - **Server h10:**
        - rate= 6.739 Mbps
        - received=27433 KB

Part 3 - Q4 Predictions

If you map out the paths for the two pairs of hosts, you can see an intersection point at link 2. This connection is shared between the two pairs and will slow down the throughput and increase the latency. The latency will be longer because a packet could start transferring through unshared links but will eventually hit the busy link 2. Therefore the latency will be greater compared to Q3 where the hosts were all leaving the same router. The throughput will be slower due to the same reasoning.

Part3 - Q4 Results

The throughput was actually around the same as Q3 two pairs which I did not expect. This makes sense because there were the same number of links and only one colliding link for both scenarios. Therefore the throughput should be similar because the same bottlenecks are present in both tests. The latency for h1->h4 was similar to those in Q3, but the latency for h5->h6 was faster. I believe this could be due timing when running the ping commands and because there is a shared link between the two transfers. This shared link will be busy some of the time, making one of the two pairs have a longer rtt. Because the clients don't start at the same router, the packets will send but get stopped half way through at the shared link. This will slow the rtt.