Part 1 – Wired Environment

client:

```
royal-07:~/Desktop/cs640-projects/lab1-sockets/src> java Iperfer -c -h 128.105.37.148 -p 2000 -t 30
sent=3504594 KB rate=934.533 Mbps
royal-07:~/Desktop/cs640-projects/lab1-sockets/src>
```

server:

```
[chuan@royal-08] (35)$ java Iperfer.java -s -p 2000
received=3504594 KB rate=933.475 Mbps
[chuan@royal-08] (36)$
```

Part 1 – Wireless Environment

We would expect greater throughput in a wired environment, because the rate of transferring data is greater in a wire compared to a wireless connection. In a wired environment, the transfer of data occurs through a physical medium. On the other hand, wireless environments are affected by factors such as obstacles, walls, etc. and are less predictable and slower.

Part 1 – Iperfer on Wireless Environment

client:

```
mininet@mininet-vm:~$ java Iperfer -c -h 10.140.191.140 -p 2000 -t 30
sent=404365 KB rate=107.831 Mbps
```

server:

```
alan@Alans-Air src % java Iperfer.java -s -p 2000
received=404365 KB rate=105.849 Mbps
alan@Alans-Air src %
```

Part 1 – Iperfer Results

The throughput of the wireless environment is slower than the wired one. As predicted, the rate of transfer is slower compared to a wired environment. As mentioned above, there are other factors that slow down the rate of transfer, one of which is the traffic of the network. The network that we used was UWNet, which is not an exclusive network, meaning that other applications are transferring data across the network.

Part 3 – Q2 Predictions

Given that we know the RTT for L1, L2, and L3. We can expect that the latency for H1 to H4 will be the sum of the latencies of L1, L2, and L3.

Additionally, we know the throughputs for L1, L2, and L3. The bandwidth from H1 to H4 would me the minimum of the 3 throughputs

H1H4 Latency = (L1 / 2) + (L2 / 2) + (L3 / 2)

= (80.192 / 2) + (20.689 / 2) + (40.549 / 2)

= 40.096 + 10.345 + 20.275 = 70.716 ms

H1H4 Bandwidth = min(L1, L2, L3)

= min(18.893, 47.275, 37.553) = 18.893 Mbps

Part 3 – Q2 Results

H1H4 RTT Results = 141.906 ms

H1H4 Latency Results = 70.953 ms

H1H4 Bandwidth Results = 18.717 Mbps

To get from H1 to H4, we take the following path: H1 -> S1 -> S2 -> S3 -> S4 -> H4

This involves taking L1, L2, and L3. So, to find the latency from H1 to H4, we take the sum of L1, L2, and L3. As for the bandwidth, we take the minimum of the bandwidths of L1, L2, L3.

Part 3 – Q3 Predictions

To test 2 pairs, we choose H1H4 and H8H9.

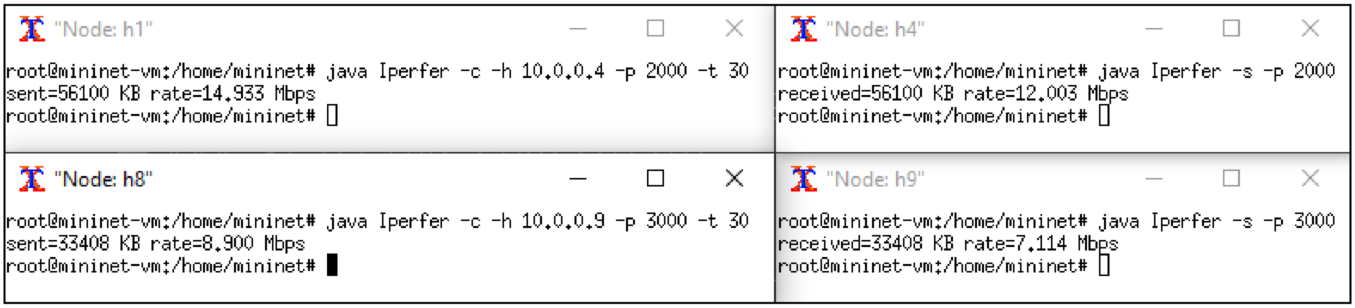
We estimate that the latency would double, and the bandwidth would be halved. This is because with two pairs of connections there are two inputs to the links and the bandwidth would have to be shared between two hosts. As a result, it would take longer to transmit the data which explains the doubling in latency.

To test 3 pairs, we use H1H4, H8H9, and H7H10.

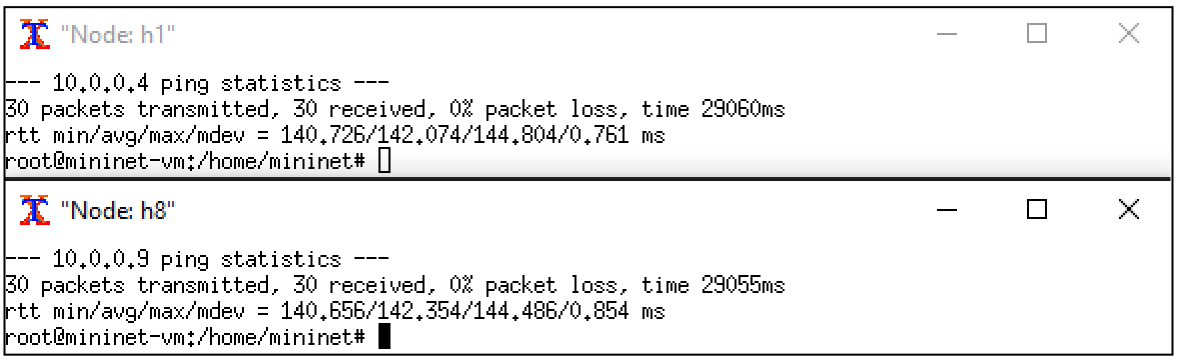
We estimate that the bandwidth would be much lower as it must be shared with more connections now and latency would also take longer as now there are more hosts communicating with each other.

Part 3 – Q3 Results

Two pairs: H1H4 and H8H9

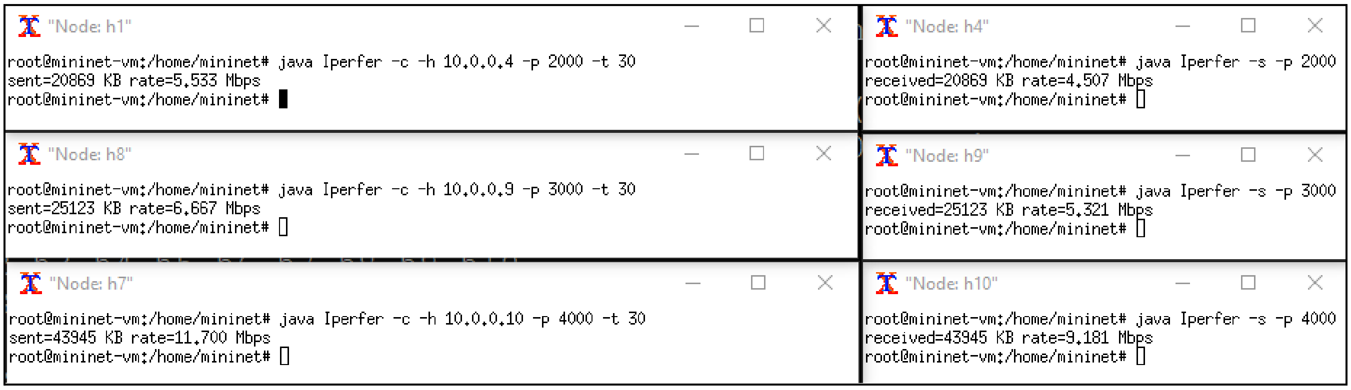


The bandwidth for the two pairs is lower than the bandwidth for one pair. Their bandwidth is not equal though which could be due to the multiplexer doing internal work to assign bandwidth.

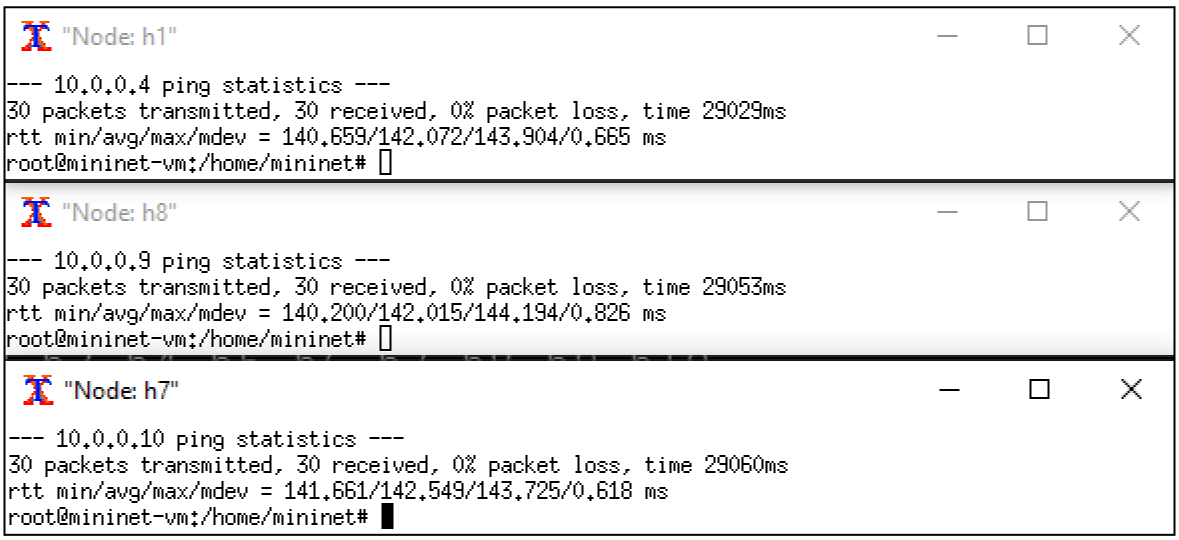


Latency was not doubled and remained the same as one pair would be. This is because the amount of data being transferred in the link has no impact on the latency of the link. The amount of time it takes to traverse the link is always the same.

Three pairs: H1H4 and H8H9 and H7H10



The bandwidth of the 3 pairs are not equal and are lower than the bandwidth of a 1 pair scenario.



The latencies remained the same for each pair. No effect on latency. The same as above, this is because the amount of data being transferred in the link has no impact on the latency of the link. The amount of time it takes to traverse the link is always the same.

Part 3 – Q4 Predictions

H1 to H4 Latency: we just add the latencies of L1, L2, and L3 which is 40 + 10 + 20 = 70 ms

H1 to H4 Throughput: we take the minimum of L1, L2, and L3 which is min(18, 47, 37) = 18 Mbps

H5 to H6 Latency: we just add the latencies of L2, L4, and L5 which is 10 + 15 + 15 = 40 ms

H5 to H6 Throughput: we take the minimum of L2, L4, and L5 which is min(47, 28, 28) = 28 Mbps

Part 3 – Q4 Results

H1 to H4 Latency: 70.825 ms

```
--- 10.0.0.4 ping statistics ---
30 packets transmitted, 30 received, 0% packet loss, time 29053ms
rtt min/avg/max/mdev = 140.172/141.649/143.593/0.878 ms
```

H1 to H4 Throughput: 18.660 Mbps

```
mininet@mininet-vm:~$ cat throughput_h1-h4.txt
sent=80451 KB rate=21.433 Mbps
received=80451 KB rate=18.660 Mbps
```

H5 to H6 Latency: 40.972 ms

```
--- 10.0.0.6 ping statistics ---
30 packets transmitted, 30 received, 0% packet loss, time 29026ms
rtt min/avg/max/mdev = 81.079/81.944/83.403/0.651 ms
```

H5 to H6 Throughput: 28.044 Mbps

```
mininet@mininet-vm:~$ cat throughput_h5-h6.txt
sent=114941 KB rate=30.633 Mbps
received=114941 KB rate=28.044 Mbps
```