CS740 Lab 1: Sliding window protocol in DPDK

Partho Sarthi(sarthi@wisc.edu), Ajay Joshi (joshi28@wisc.edu)

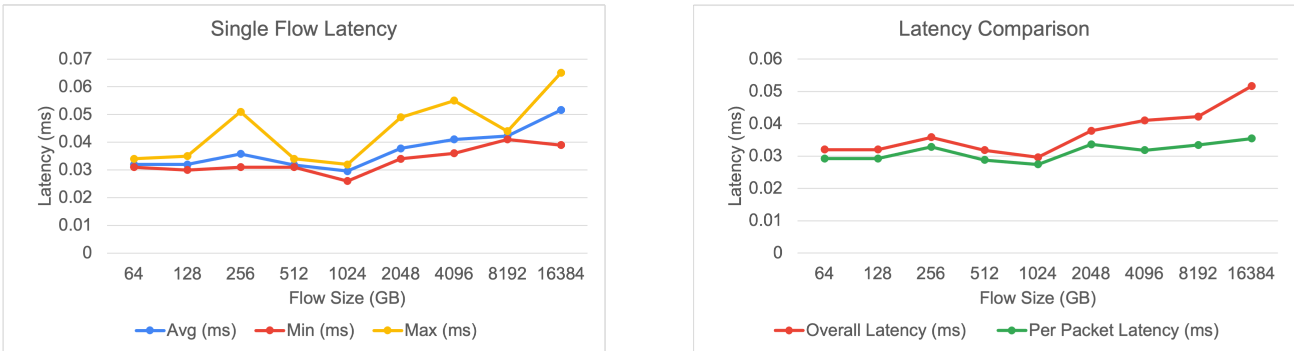
17 February 2023

1 Implementation details

- All traffic is sent/received through a single DPDK physical port and using a single queue.
- All measurements are reported as an average of 5 independent runs.
- We use a packet size of 1KB, and the maximum allowed packet size is around 1500 as Ethernet has an MTU of 1500. Packet sizes higher than 1500 failed to transmit/receive.
- We use a static TCP window size of 50 for our experiments. We assume no packets are dropped and do not handle retransmissions.

2 Single Flow Latency

Figure 1a shows the per-packet latency without considering the packet creation overhead. Figure 1b shows the overall latency for the flow. In Figure 2, we compare the variation of latency by changing the TCP window size.



(a) Single flow latency - Per-packet latency

(b) Comparison between overall and per-packet latency

Figure 1: Latency Analysis

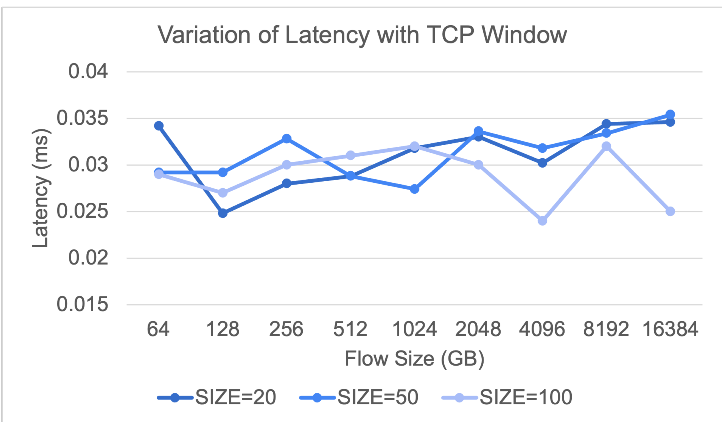
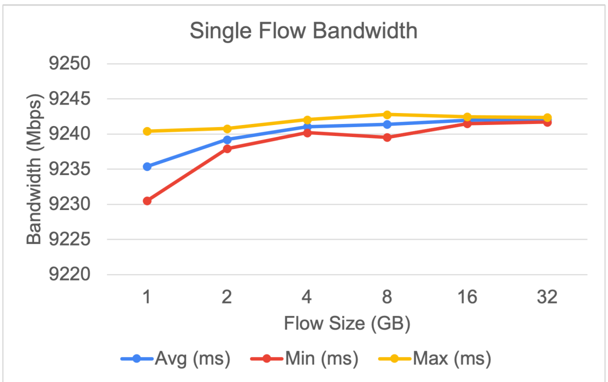


Figure 2: Variation of Latency with TCP Window

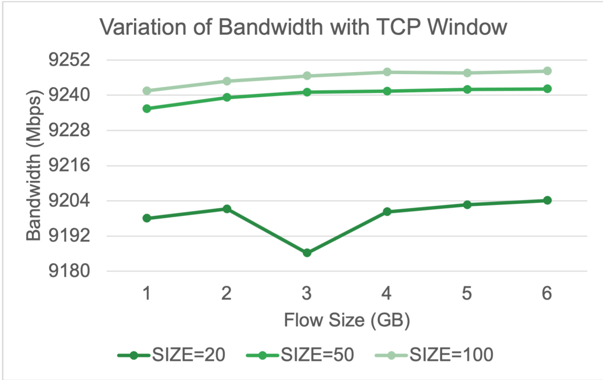
Observation: Per-packet latency remains around **30μsec** to **40μsec** for all flow sizes which is lower than the ping latency of around **150μsec**. Overall latency is not proportional to the number of packets sent due to the burst mode features of the NICs. TCP window length does not affect observations, since a small number of packets are sent.

3 Single Flow Bandwidth

Figure 3a shows the single flow bandwidth for increasing flow sizes. Figure 2 shows the variation of bandwidth by changing the TCP window size.



(a) Single flow bandwidth



(b) Variation of Bandwidth with TCP Window

Figure 3: Bandwidth Analyis

Observation: From figure 3a, we can see that bandwidth remains the same across flow sizes, which is around **9.2Gbps**. Bandwidth varies mildly with changes in TCP window size. Our bandwidth measurements are consistent with measurements done using iperf and information from lshw showing a **10Gbps** NIC.

4 Multiple Flows Bandwidth

Figure 4 shows bandwidth variation for multiple flows. We implement a round-robin strategy for scheduling multiple flows.

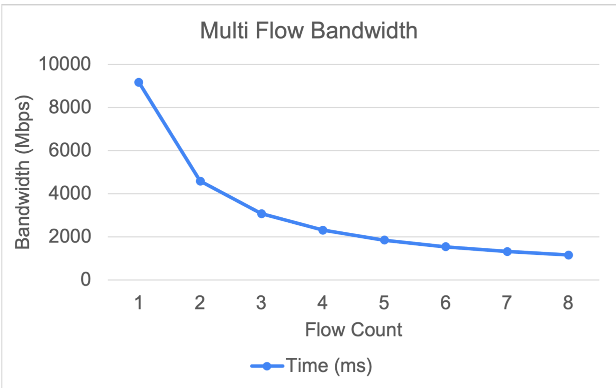


Figure 4: Multiple flow Bandwidth Analyis

Observation: Average per-flow bandwidth reduces as $\frac{Single-flow-bandwidth}{Num-flows}$ with an increase in the number of flows, showing the effect of round-robin flow scheduling.

5 Conclusion

The key observations from this project are (a) using DPDK, we are able to achieve bandwidth closer to the hardware limit (b) Per-flow bandwidth decreases as the number of flows is increased (c) TCP Window Size does not affect the network performance significantly under our assumptions.