# 1 Understanding network protocols using ns3

## 1.2 Exercise 2: Play with FTP flow (for sliding window, etc)

1. For the given link data rate, what is the maximum throughput the ftp flow can achieve?
   1. 7739.2Kbps
2. What should be the theoretically calculated window size (in bytes) that will achieve this throughput
   1. (7739.2Kbps)\*(RTT) = (7739.2Kbps)\*(2\*10ms) = 19812.35 bytes
3. What is experimentally seen minimum window size at which this throughput was obtained?
   1. ~19300bytes
4. How does the maximum achieved throughput compare with the raw data rate of the link?
   1. Close to but less than raw data rate (8Mbps).
5. How does changing link delay affect all the above? Write down the parameters and measurements of the experiments you did to study the impact of link delay.
   1. Increasing link delay will increase minimum window size and observed throughput (though still <= 8Mbps).
   2. Parameters to explore: Window size (line 74), link delay (line 99)

## 1.3 Exercise 3: Only CBR Flow

“CBR” stands for Constant Bit Rate. This is the kind of traffic generated by non-coded voice flows (voice is sampled at a fixed rate, a packet assembled of the digitized voice, and it is sent off). This type of traffic is time-sensitive but can do with a few losses hence RDT is not used for such flows. In the Internet stack, they are sent over UDP, which has no ACKs etc.

We’ll study this flow to see how it behaves in contrast to a flow under sliding window control.

Study the 3rd file provided. CBRonly.cc . Copy this to the scratch folder, and run waf in the higher directory as

./waf --run CBRonly

Understand the code (very similar to FTP simulator code). Play with the CBR flow parameters and check the throughput and delays

Give the following answers

1. Enter 3 different throughput values (in increasing order), for 3 different sets of flow parameters
   1. Throughput v/s (Link rate, link delay, CBR data rate)
      1. 231.073 Kbps (8Mbps, 10ms, 224Kbps)
      2. 435.011 Kbps (448Kbps, 10ms, 448Kbps)
      3. 461.296 Kbps (8Mbps, 10ms, 448Kbps)

Note: window size does not affect CBR throughput

1. What is the maximum throughput this flow can achieve?
   * 1. 461.296Kbps (if CBR data rate set to 448Kbps)
     2. 7768.12Kbps (if CBR data rate increased)
2. How does this compare with the raw data rate of the bottleneck link?
   * 1. Much lower than (8Mbps). Note here the CBR flow has a input throughput of 448Kbps. The observed throughput should be close to this. It is more than 448Kbps since the Flow Monitor measures throughput at the Network Layer which would also include UDP overheads (that is more bytes transferred than what CBR sent in the same amount of time)
     2. Here, the physical link is the bottleneck. Although CBR > 8Mbps, the observed throughput has to be < 8Mbps

## 1.4 Exercise 4: Study impact of CBR on Sliding window flow

1. CBR and FTP Flow Baseline:

To ensure FTP does not run out of data to send, set **max\_bytes = 0**

**Only CBR flow:** ~10% of Link throughput

Flow ID: 1

Src Addr: 10.1.1.1 ----- Dst Addr: 10.1.1.2

Tx Bytes 527040

Tx Packets 976

Rx Bytes 525960

Rx Packet 974

Input Load 824.82 Kbps

Observed Throughput 823.08 Kbps

Mean delay 0.010542

Mean jitter 0

**CBR and FTP:** baseline: receive buffer/window size = 19300 bytes, cbr data rate = 800Kbps

====================== Flow monitor statistics ======================

Flow ID: 1 (FTP)

Src Addr: 10.1.1.1 ----- Dst Addr: 10.1.1.2

Tx Bytes 4441272

Tx Packets 7555

Rx Bytes 4428336

Rx Packet 7533

Input Load 6939.61 Kbps

Observed Throughput 6920.38 Kbps

Mean delay 0.0134461

Mean jitter 0.000592226

Flow ID: 2 (CBR)

Src Addr: 10.1.1.1 ----- Dst Addr: 10.1.1.2

Tx Bytes 527040

Tx Packets 976

Rx Bytes 525960

Rx Packet 974

Input Load 824.82 Kbps

Observed Throughput 822.698 Kbps

Mean delay 0.0133501

Mean jitter 0.000290869

1. Impact of high CBR on FTP

both the throughputs on one graph and both the delays on another graph.

Since in this topology, we have used only 2 nodes, one would have to distinguish FTP flow from CBR flow by looking at the throughputs (CBR should have around 800 Kbps throughput at 10% data rate). A more robust experiment would be one where we use a 4 node topology with 2 source nodes (1FTP, 1CBR ), both connected to a router (n2) which is in turn connected to the destination node (n3).

**// n0**

**// \**

**// \**

**// n2 -------------------------n3**

**// /**

**// /**

**// n1**

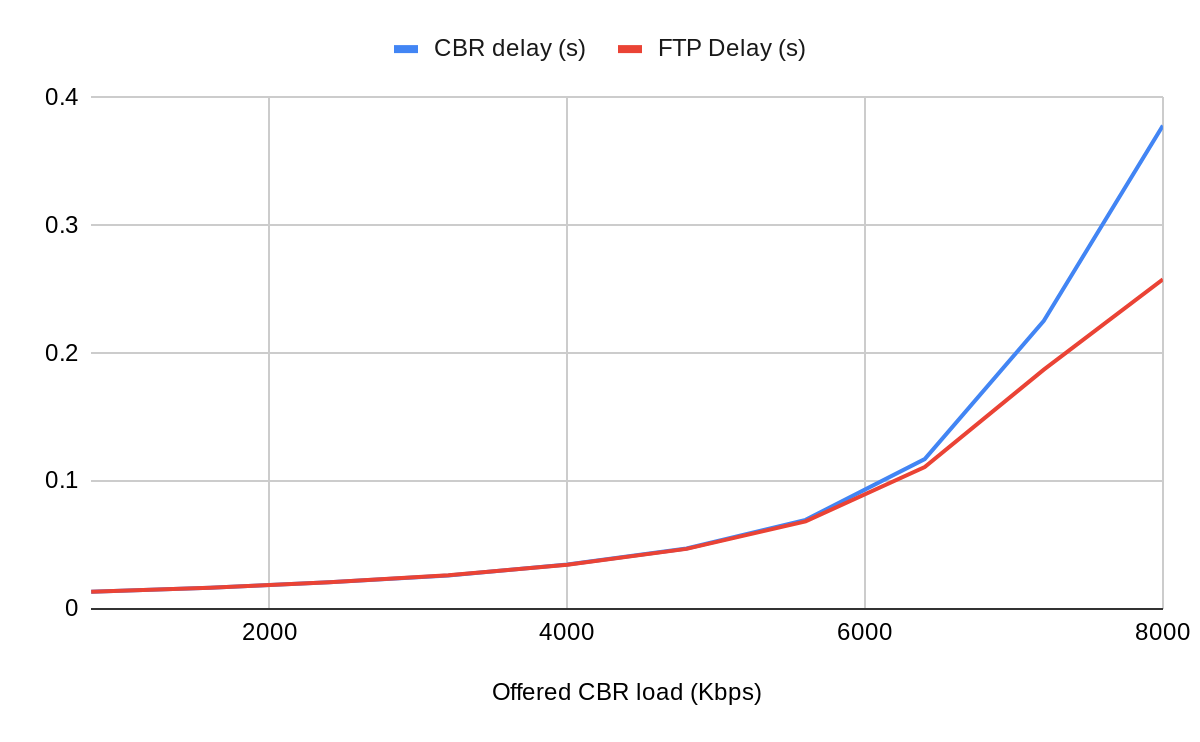
Note: Don’t worry if you cannot figure out how to convert a 2 node topology into a 4 node topology. It isn’t necessary for this lab.

* 1. Tabulate the four metrics above as a function of this data rate and also plot them (best to use a spreadsheet). Copy-paste the table below:.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **~ CBR offered load (% link speed)** | **CBR offered load (Kbps)** | **CBR throughput (Kbps) (includes UDP overheads)** | **CBR delay (s)** | **FTP throughput (Kbps)** | **FTP Delay (s)** |
| 10 | 800 | 822.698 | 0.0133501 | 6920.38 | 0.0134461 |
| 20 | 1600 | 1643.4 | 0.0165751 | 6102.65 | 0.0166246 |
| 30 | 2400 | 2462.31 | 0.0208712 | 5286.85 | 0.0207731 |
| 40 | 3200 | 3279.31 | 0.026234 | 4472.8 | 0.0264127 |
| 50 | 4000 | 4091.49 | 0.0346988 | 3663.76 | 0.0344942 |
| 60 | 4800 | 4897.46 | 0.0473347 | 2861.37 | 0.046987 |
| 70 | 5600 | 5685.21 | 0.0696749 | 2075.86 | 0.0684871 |
| 80 | 6400 | 6423.12 | 0.11725 | 1341.17 | 0.11096 |
| 90 | 7200 | 6961 | 0.225464 | 807.765 | 0.187312 |
| 100 | 8000 | 7230.62 | 0.378271 | 538.97 | 0.257877 |

* 1. Plot both the throughputs on one graph and both the delays on another graph as a function of the data rate. Copy-paste the plots below:





Write down conclusions regarding how

The increase affects CBR’s own throughput and delay (and why)?

More CBR input load indicates more CBR packets to be sent per FTP packet per unit time. Thus, CBR packets will be sent more often and CBR throughput increases. However, each packet will have to wait more (mean CBR delay increases) since the CBR throughput will not grow in proportion (FTP packets still need to be sent). Delays shoot up due to high congestion as CBR flow approaches 100%. CBR throughput saturates.

How the increase affects FTP’s throughput and delay (and why)?

More CBR input load indicates more CBR packets to be sent per FTP packet per unit time. Thus, FTP packets will be sent less often and FTP throughput decreases. Since the input load is same (packets per unit time), this would lead to each packet to wait more before it is scheduled to be delivered on the link. Thus mean FTP delay increases. Delays shoot up due to high congestion as CBR flow approaches 100%. FTP throughput saturates at a low value since it has to wait for ACKs as well, and congestion delays them significantly.