EINTE Lab 1

Damian Bukowski 303828

Kacper Zdyb 303845

Task 1

In this task we are analysing the dependence of window size on the effectiveness of the protocol. First, we use the following parameters in the simulation:

- R1-R2 link delay: 70 ms
- Access links delay: 10 ms
- R1-R2 link capacity: 10 Mbps
- R1-R2 link buffer: 5 packets
- Simulation time: 300 ms
- Initialization time: 20 ms
- Segment size: 1500 [bytes] (with overhead)

Then, we calculate the theoretical optimal window size, using the formulas:

$$Optimal\ window\ size\ [bytes] = Bandwidth\ \left[\frac{bytes}{s}\right] * RTT\ [s]$$

$$RTT = 2 * (propagation\ time + 2 * delay) = 2 * (70 + 2 * 10) = 180\ [ms]$$

$$Bandwidth = 10^7 \left[\frac{bits}{s}\right] = 1250000 \left[\frac{bytes}{s}\right]$$

$$Optimal\ window\ size = 1250000\ \left[\frac{bytes}{s}\right] * 0.18\ [s] = 225000\ [bytes]$$

Since in the simulation we have to input optimal window size in segments, it is beneficial to divide the optimal window size by the segement size, in out case 1500 bytes per segment. Therefore optimal window size in segments is:

$$Optimal\ window\ size\ = \frac{225000\ [bytes]}{1500\ \left[\frac{bytes}{segment}\right]} = 150\ [segements]$$

After running the simulation several times and plotting the following graph (Figure 1) we can see that maximal value of stable throughput is for window of size 145. Therefore that should be our optimal window size for further simulations.

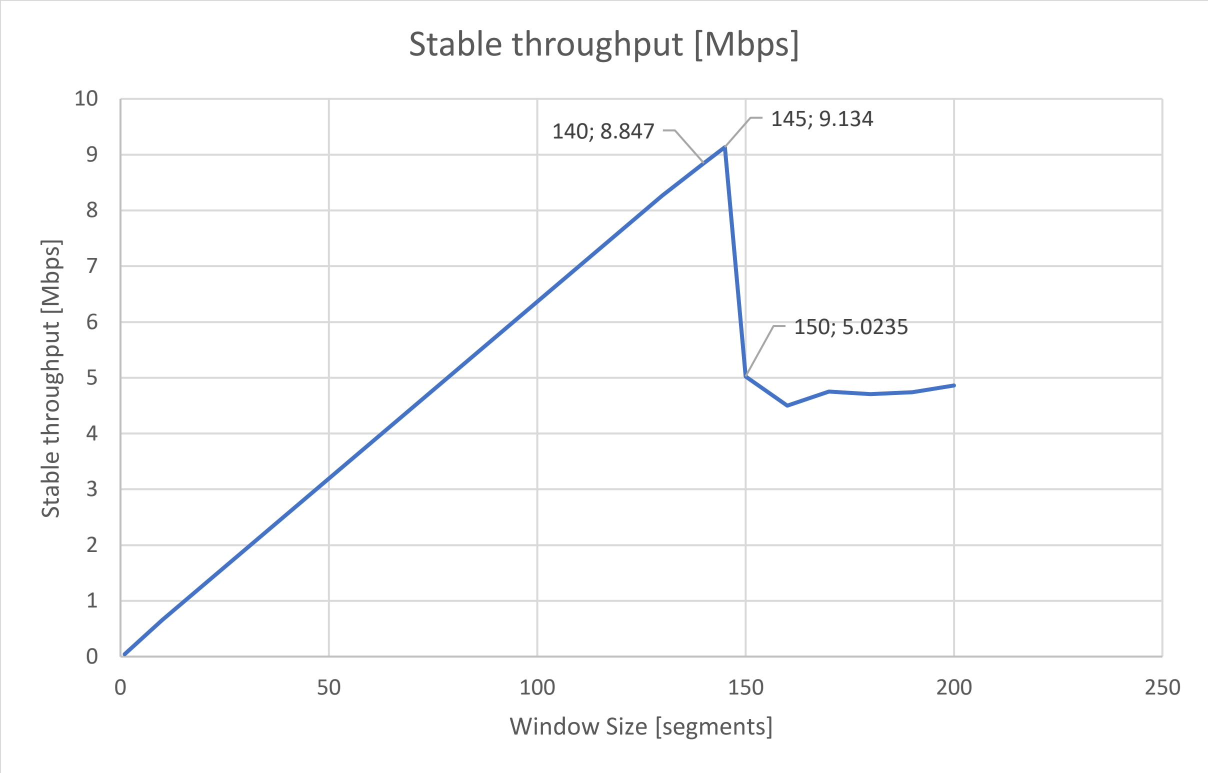

Figure 1 Stable throughput [Mbps]

In this next part of the task we are running the simulation two more times for window sizes 10% bigger and smaller than our optimal window size (in this case these sizes are 135 and 165 respectively). Plots preset themselves as follows:

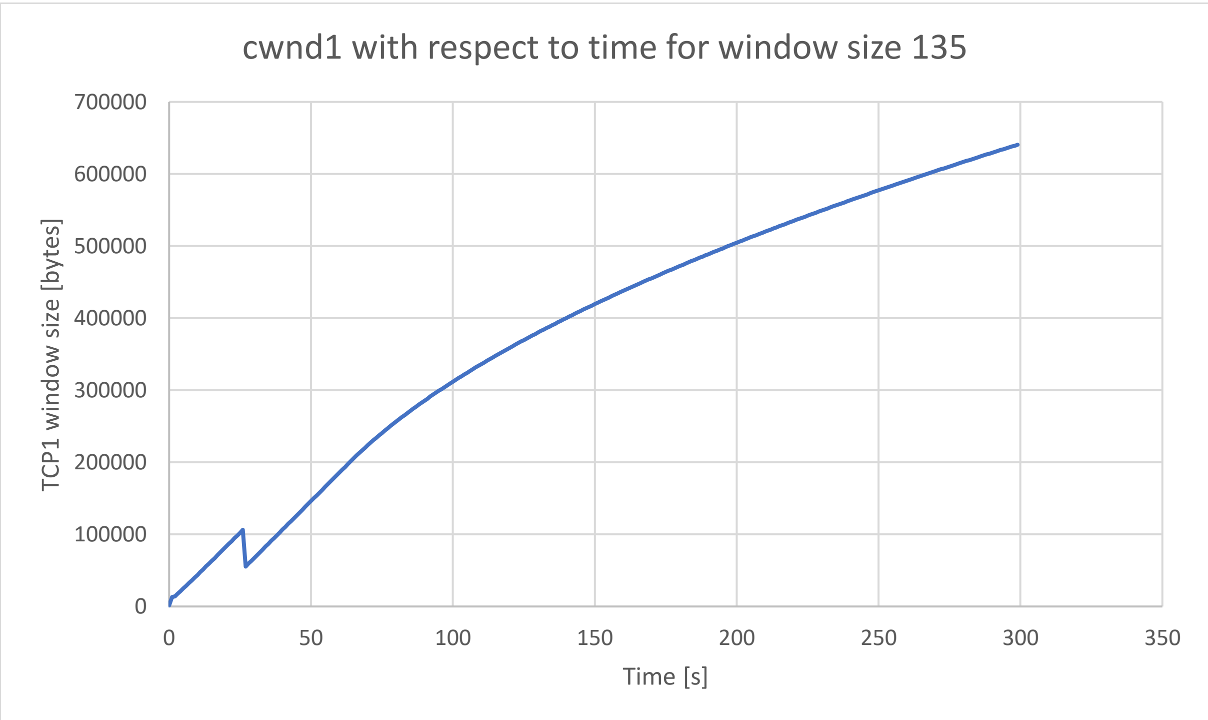

Figure 2 cwnd1 with respect to time for window size 13

```
Simulation time        300
Initialization time    20
Active sources         [1 0 0 1]
TCP Windows            [135 5000 5000]
Link delay             70ms
Link capacity          10Mb
Link buffer            5

TCP1 Average Throughput = 8.178921066666659 [Mbps]
      Stable Throughput = 8.608885714285247 [Mbps]
TCP2 Average Throughput = 0.0 [Mbps]
      Stable Throughput = 0.0 [Mbps]
TCP3 Average Throughput = 0.0 [Mbps]
      Stable Throughput = 0.0 [Mbps]
```

Figure 3 Parameters for running the simulation to achieve plot in Figure 2

Clearly the maximal value of the window is much greater than the theoretical value. Also, this plot does not resemble the sawtooth wave. From that we can deduce that there is no congestion and therefore no packet loss.

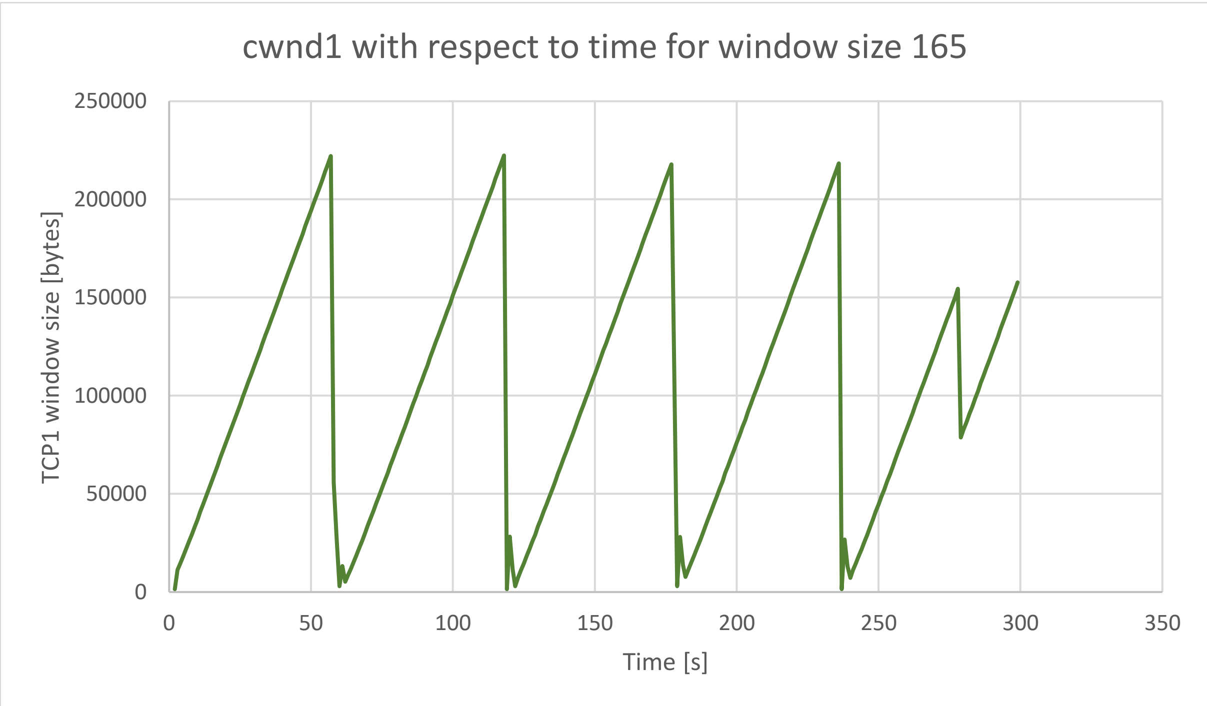

Figure 4 cwnd1 with respect to time for window size 165

```
Simulation time       300
Initialization time   20
Active sources        [1 0 0 1]
TCP Windows           [165 5000 5000]
Link delay            70ms
Link capacity         10Mb
Link buffer           5

TCP1 Average Throughput = 4.571481066666663 [Mbps]
      Stable Throughput = 4.7833118279569895 [Mbps]
TCP2 Average Throughput = 0.0 [Mbps]
      Stable Throughput = 0.0 [Mbps]
TCP3 Average Throughput = 0.0 [Mbps]
      Stable Throughput = 0.0 [Mbps]
```

Figure 5 Parameters for running the simulation to achieve plot in Figure 4

For these parameters there clearly is some packet loss, since the cwnd drops after reaching the optimal window size. On top of that, the packet loss is very big so cwnd drops almost to 0 instead of droping by half and then increasing further.

Task 1 conclusions

Optimal window size (145 segments) is just a bit shy of the theoretical one (150 segments). Choosing many window sizes and then checking for which the throughput is the biggest seems to be a very efficient way of finding a good window size.

Task 2

We chose to first run the simulations with the same parameters as in the previous task, except this time the window size is contant 5000 segments. We will change the buffer, starting from 5 packets. As can be seen from Figure 6, the throughput is around 70%. We can do better. For link buffer equal to 65 packets the throughput is around 90% (as can be observed on Figure 9) which is good enough for our needs. Therefore that is the link buffer that we will be using later on.

Task 2.1 Buffer size equal to 5 packets

```
Simulation time       300
Initialization time   20
Active sources        [1 0 0 1]
TCP Windows           [5000 5000 5000]
Link delay            70ms
Link capacity         10Mb
Link buffer           5

TCP1 Average Throughput = 6.5898010666666664 [Mbps]
      Stable Throughput = 6.9535483870967738 [Mbps]
TCP2 Average Throughput = 0.0 [Mbps]
      Stable Throughput = 0.0 [Mbps]
TCP3 Average Throughput = 0.0 [Mbps]
      Stable Throughput = 0.0 [Mbps]
```

Figure 6 Simulation results for link buffer size equal to 5 packets

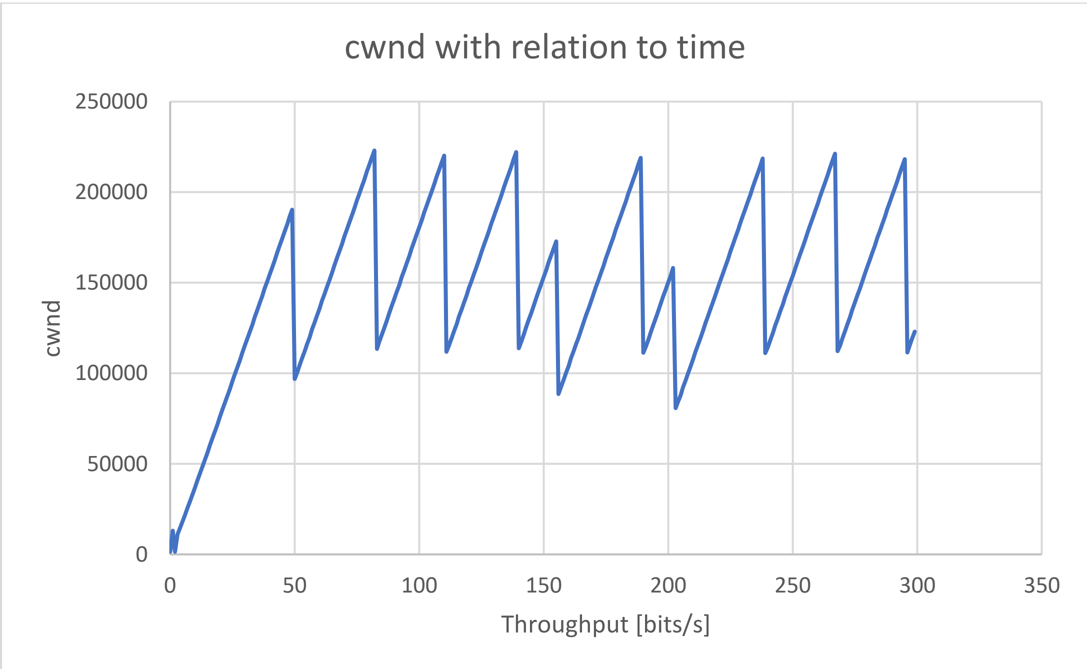

Figure 7 Plot for link buffer size equal to 5 packets

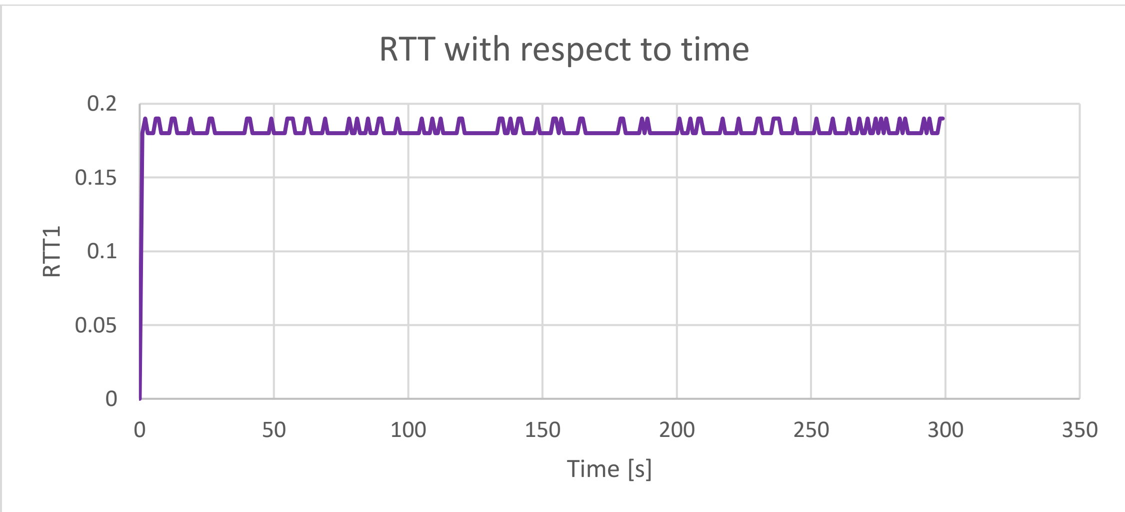

Figure 8 Plot for link buffer size equal to 5 packets

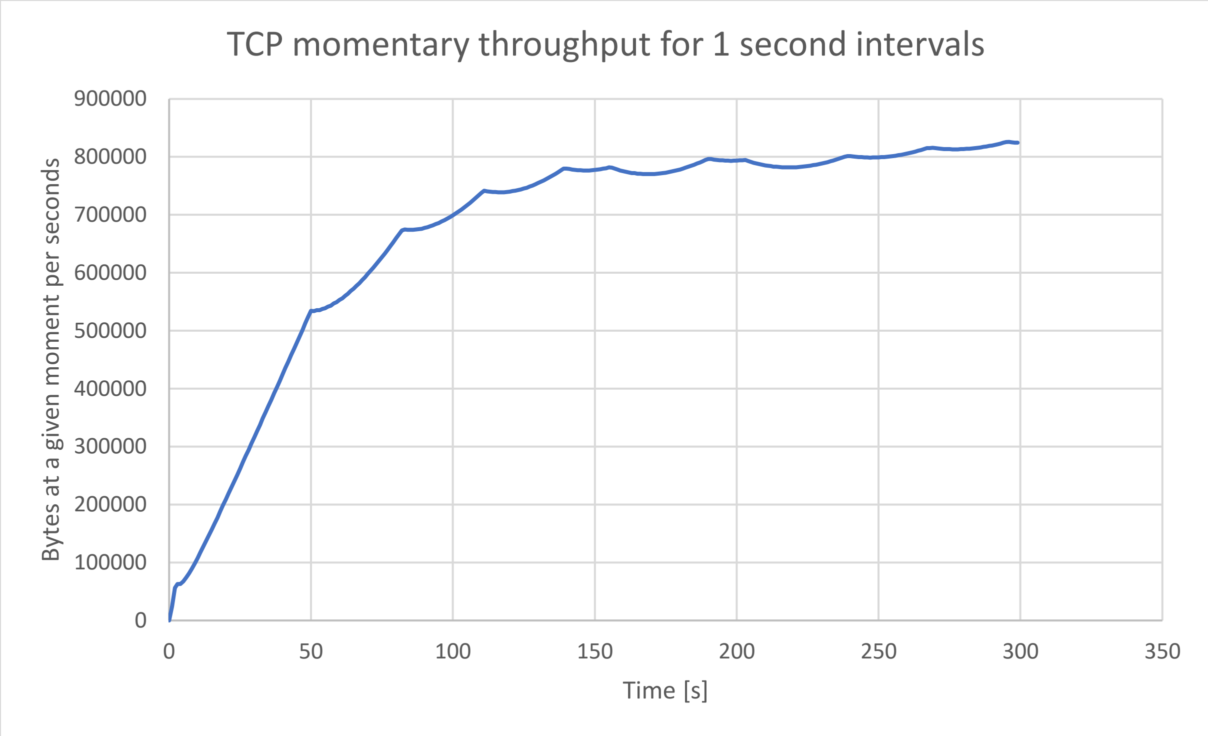

Figure 9 TCP momentary throughput for 1 second intervals

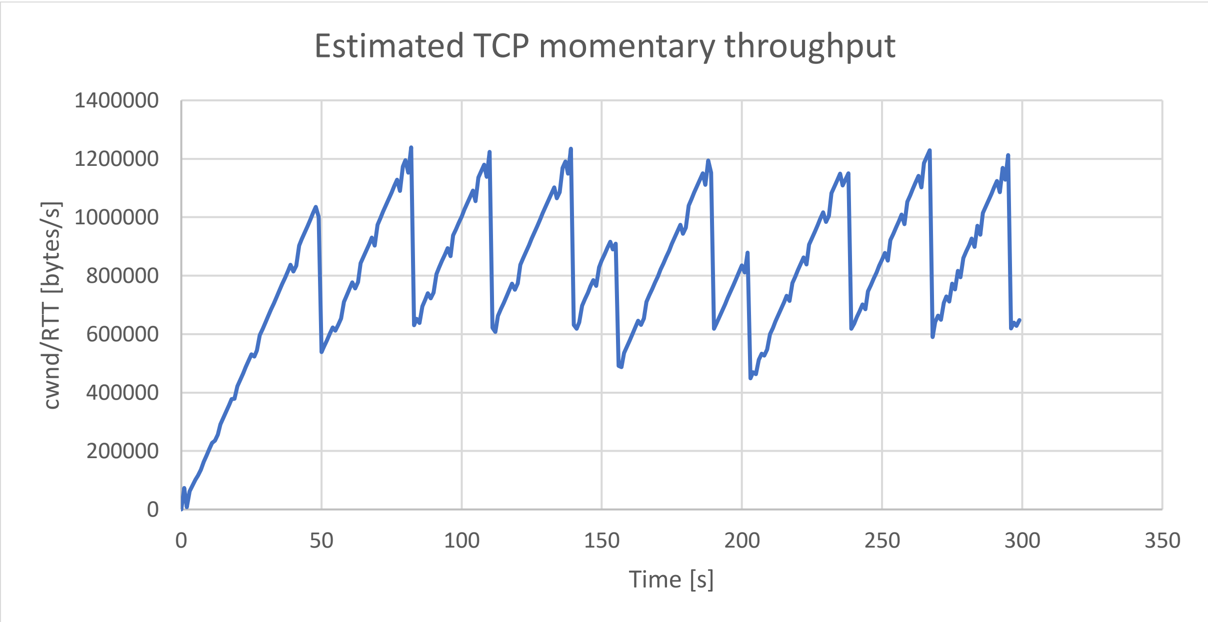

Figure 10 Estimated TCP momentary throughput

$$Rate \approx \frac{cwnd}{RTT} \left[\frac{bytes}{s}\right]$$

From above plots we can see that Figure 9 is a bit wonky. That is because we decided to chose the probing time to be 1 second in the simulator and then devide number of transfered bytes in a given moment by the time that has elapsed. Since throughput is defined by how many bytes have been travelled in the given time, we decided that this would be an appropriate way of approximating it. As for figure 10, we simply applied the above formula and plotted the results. We don't need to calculate mean cwnd nor RTT since our interval is already 1 second, thus simulation results are already sort of an average. Same rules apply to plots in Task 2.2.

Task 2.2 Buffer size equal to 65 packets

```
Simulation time       300
Initialization time   20
Active sources        [1 0 0 1]
TCP Windows           [5000 5000 5000]
Link delay            70ms
Link capacity         10Mb
Link buffer           65

TCP1 Average Throughput = 8.734121066666669 [Mbps]
      Stable Throughput = 8.978107526881720l [Mbps]
TCP2 Average Throughput = 0.0 [Mbps]
      Stable Throughput = 0.0 [Mbps]
TCP3 Average Throughput = 0.0 [Mbps]
      Stable Throughput = 0.0 [Mbps]
```

Figure 11 Simulation results for buffer size equal to 65 packets

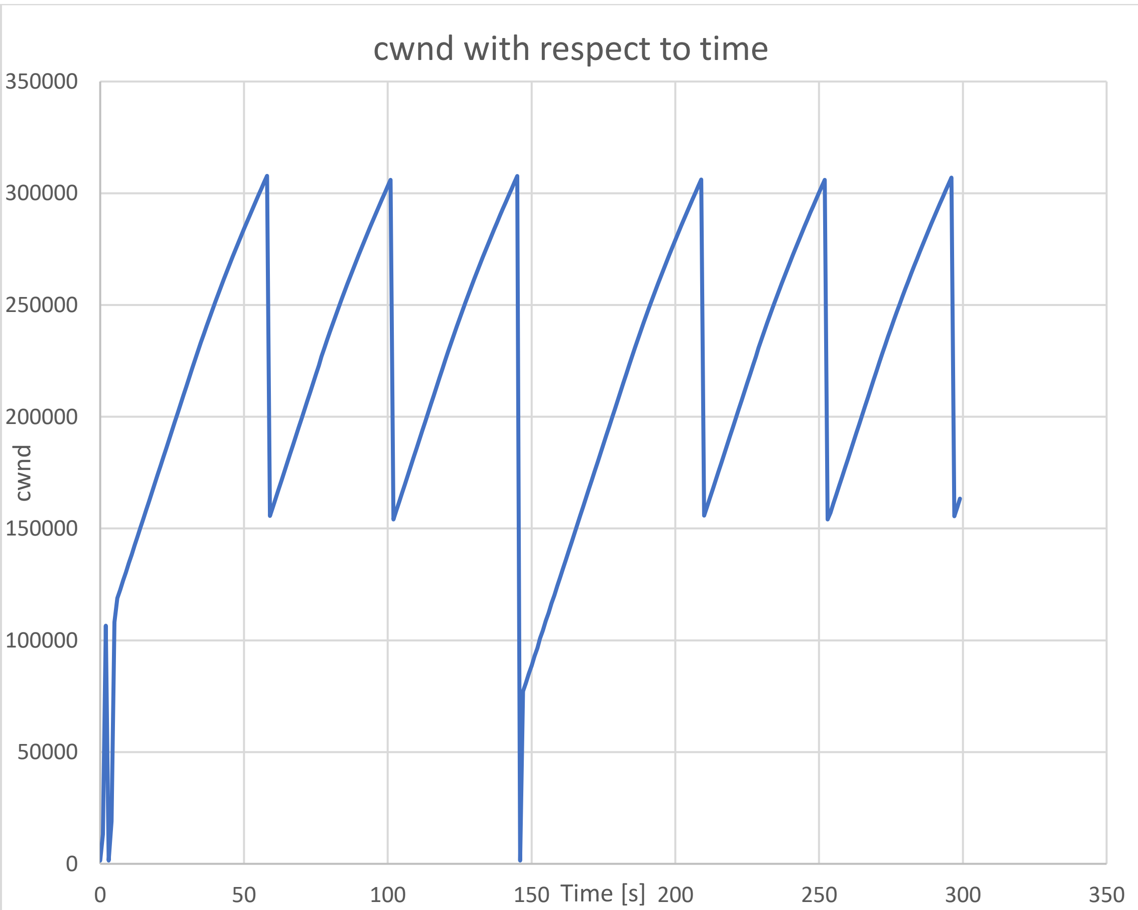

Figure 12 Plot for buffer size equal to 65 packets

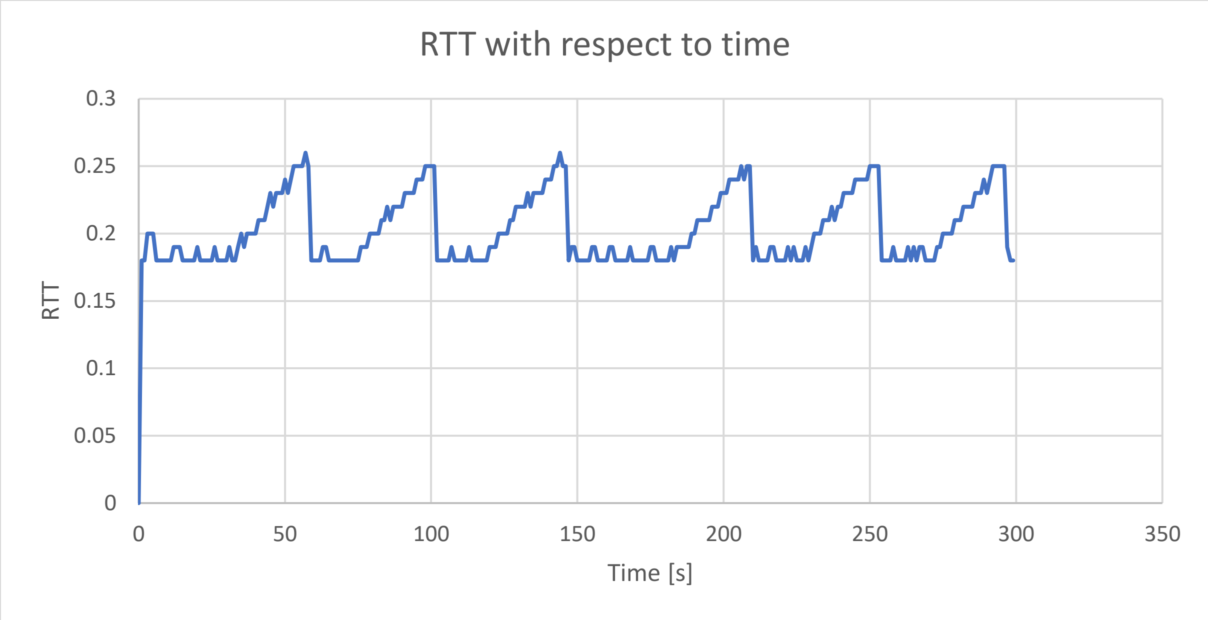

Figure 13 Plot for buffer size equal to 65 packets

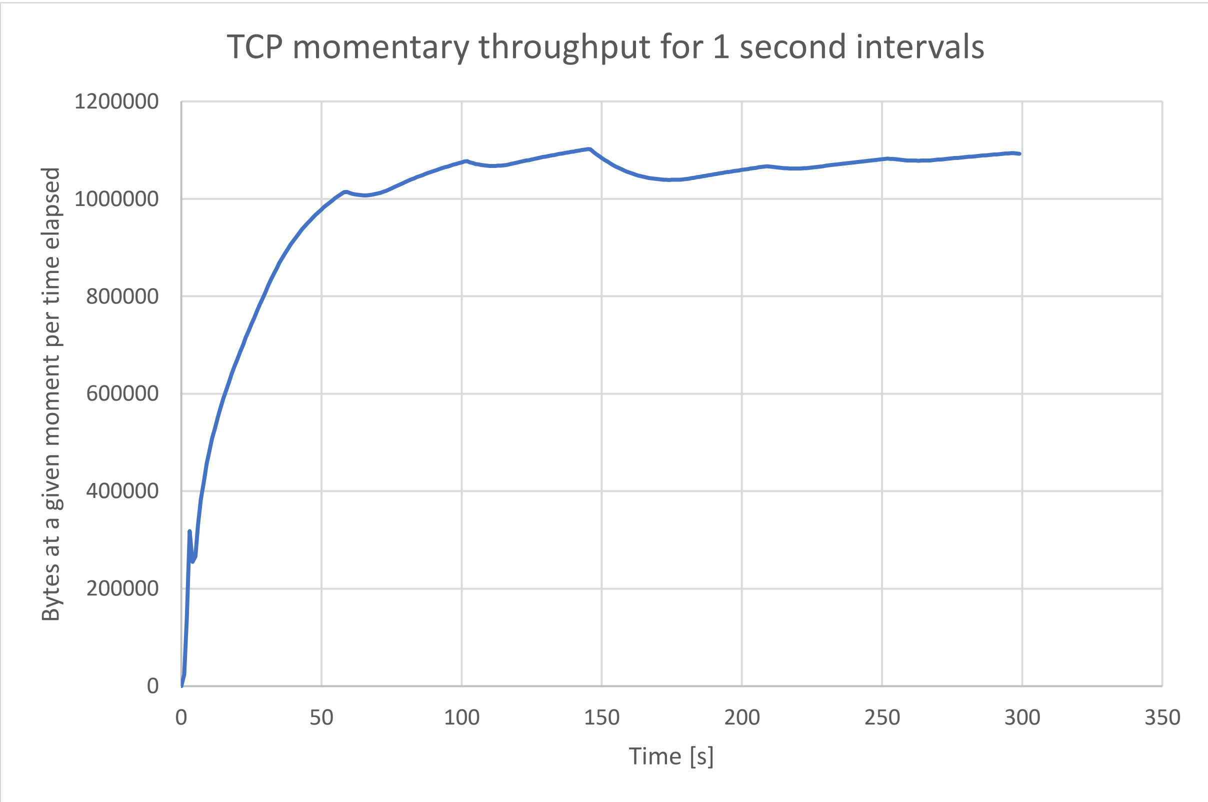

Figure 14 TCP momentary throughput for 1 second intervals

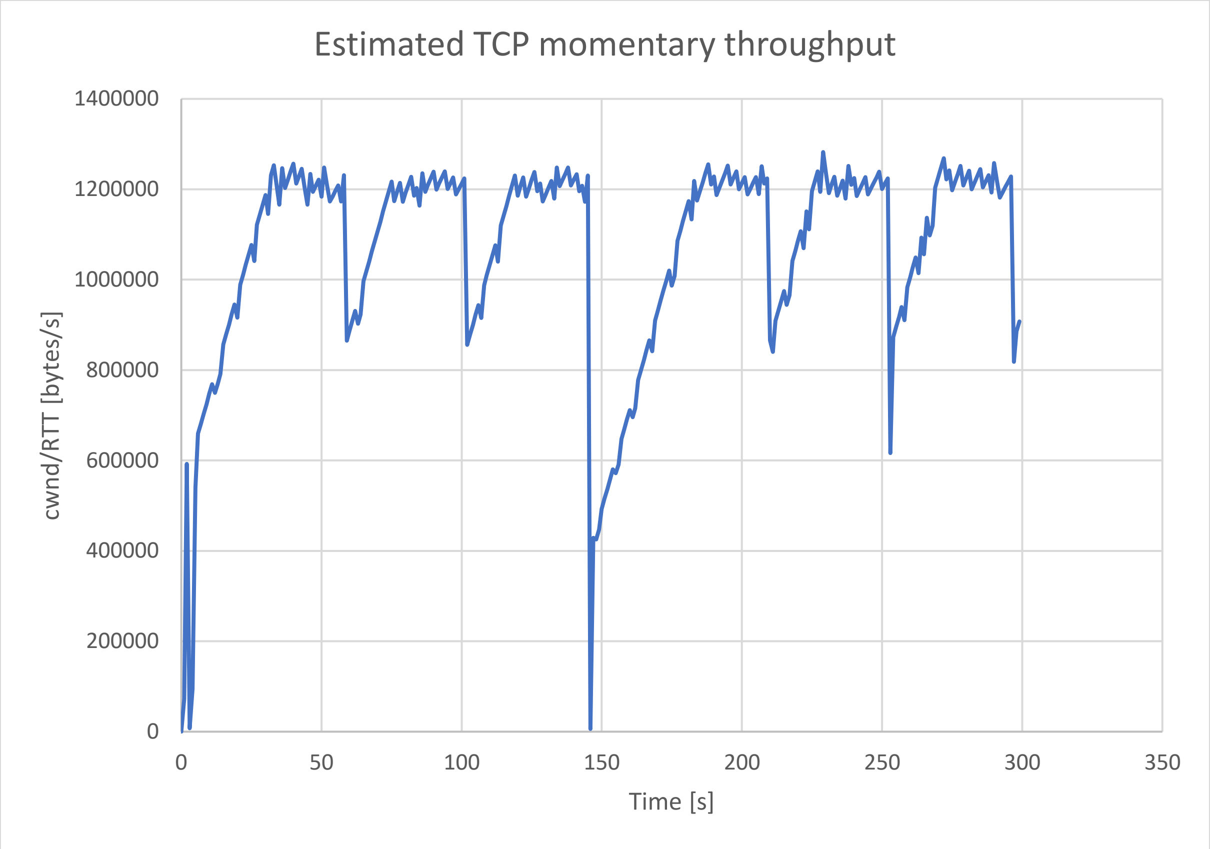

Figure 15 Estimated TCP momentary throughput

Task 2 Conclusions

When comparing how the protocol behaves with different buffer sizes we can see that for buffer size equal to 5 packets the RTT is lower, but varies a lot more than in the other case. We believe that a more desirable behavior is that of the second case, since the RTT difference is slight, about 5 miliseconds, but is spikes much less frequently which makes the conncetion more stable. When comparing the window sizes we can see that it reaches greater values in the second case, meaning that the throughput usage is greater (as can be seen on the console output) which is good. Also, in the first case the window size drops more often, making the data transfer more unstable. Lastly, if we compare the momentary throughputs, we can see that for the 1 second interval the maximal throughput is reached faster i nthe second case, which is really good because big amounts of data can be transfered faster. As for the estimated momentary throughputs, in the second case the data transfered is atleat around 900000 bytes per seond in most cases, which is more than in the first case. If we combine that information with the fact that window size is greater in the second case and it doesn't drop as much, we can conclude that the second case produces less packet loss, which is obviously very good.

Task 3

In this task we check how the TCP behaves (and how it changes the bandwidth) when there are many contesting connections. In the first part we check how the bandwidth usage varies when we change the RTT, having all 3 access links with the same delay. Next we change the access link delays to vary between each connection and check the how bandwidth is distrubuted.

Task 3.1 Equal access link delay

Console outputs for each case:

```
Simulation time      300
Initialization time  20
Active sources       [1 1 1 1]
TCP Windows          [5000 5000 5000]
Link delay           3ms
Link capacity        10Mb
Link buffer          65

TCP1 Average Throughput = 3.5116810666666667 [Mbps]
      Stable Throughput = 3.4950107526881724 [Mbps]
TCP2 Average Throughput = 3.0763210666666669 [Mbps]
      Stable Throughput = 3.1275698924731183 [Mbps]
TCP3 Average Throughput = 3.2936010666666671 [Mbps]
      Stable Throughput = 3.262064516129032 [Mbps]
```

Figure 16 Console output for case 1

```
Simulation time      300
Initialization time  20
Active sources       [1 1 1 1]
TCP Windows          [5000 5000 5000]
Link delay           10ms
Link capacity        10Mb
Link buffer          65

TCP1 Average Throughput = 3.1544010666666669 [Mbps]
      Stable Throughput = 3.1431827956989249 [Mbps]
TCP2 Average Throughput = 3.562161066666667 [Mbps]
      Stable Throughput = 3.5932043010752688 [Mbps]
TCP3 Average Throughput = 3.1500410666666667 [Mbps]
      Stable Throughput = 3.1428387096774193 [Mbps]
```

Figure 17 Console output for case 2

```
Simulation time      300
Initialization time  20
Active sources       [1 1 1 1]
TCP Windows          [5000 5000 5000]
Link delay           15ms
Link capacity        10Mb
Link buffer          65

TCP1 Average Throughput = 3.3769210666666667 [Mbps]
      Stable Throughput = 3.4129462365591396 [Mbps]
TCP2 Average Throughput = 3.0883610666666668 [Mbps]
      Stable Throughput = 3.1089892473118277 [Mbps]
TCP3 Average Throughput = 3.3787210666666669 [Mbps]
      Stable Throughput = 3.3559999999999999 [Mbps]
```

Figure 18 Console output for case 3

```
Simulation time      300
Initialization time  20
Active sources       [1 1 1 1]
TCP Windows          [5000 5000 5000]
Link delay           20ms
Link capacity        10Mb
Link buffer          65

TCP1 Average Throughput = 3.62060106666667 [Mbps]
      Stable Throughput = 3.6448172043010749 [Mbps]
TCP2 Average Throughput = 3.3751210666666669 [Mbps]
      Stable Throughput = 3.3838279569892475 [Mbps]
TCP3 Average Throughput = 2.8058410666666669 [Mbps]
      Stable Throughput = 2.8342365591397849 [Mbps]
```

Figure 19 Console output for case 4

```
Simulation time      300
Initialization time  20
Active sources       [1 1 1 1]
TCP Windows          [5000 5000 5000]
Link delay           35ms
Link capacity        10Mb
Link buffer          65

TCP1 Average Throughput = 3.275961066666667 [Mbps]
      Stable Throughput = 3.3264516129032256 [Mbps]
TCP2 Average Throughput = 3.052601066666667 [Mbps]
      Stable Throughput = 3.0563440860215052 [Mbps]
TCP3 Average Throughput = 3.3348810666666671 [Mbps]
      Stable Throughput = 3.3915698924731186 [Mbps]
```

Figure 20 Console output for case 5

```
Simulation time      300
Initialization time  20
Active sources       [1 1 1 1]
TCP Windows          [5000 5000 5000]
Link delay           50ms
Link capacity        10Mb
Link buffer          65

TCP1 Average Throughput = 2.9067610666666668 [Mbps]
      Stable Throughput = 2.9418924731182794 [Mbps]
TCP2 Average Throughput = 3.258401066666667 [Mbps]
      Stable Throughput = 3.3376344086021508 [Mbps]
TCP3 Average Throughput = 3.2624410666666668 [Mbps]
      Stable Throughput = 3.4153548387096775 [Mbps]
```

Figure 21 Console output for case 6

Table containing parameters required to run the simulation:

Case number	RTT [ms]	Link access delay	Router-to-router delay
1	10	1	3
2	40	5	10
3	70	10	15
4	100	15	20
5	150	20	35
6	200	25	50

Table containing bandwidth for each connection and the overall bandwidth usage:

TCP1 Stable throughput	TCP2 Stable throughput	TCP3 Stable throughput	TCP1 Bandwidth usage [%]
3.495	3.128	3.262	98.85
3.143	3.593	3.143	98.79
3.413	3.109	3.356	98.78
3.645	3.384	2.834	98.63
3.326	3.0564	3.392	97.744
2.941	3.258	3.415	96.14

Task 3.2 Varying access link delay

Here are the console outputs for this part of the task:

```
Simulation time      300
Initialization time  20
Active sources       [1 1 1 1]
TCP Windows          [5000 5000 5000]
Link delay           5ms
Link capacity        10Mb
Link buffer          65

TCP1 Average Throughput = 4.5559210666666665 [Mbps]
      Stable Throughput = 4.5329892473118276 [Mbps]
TCP2 Average Throughput = 3.1464810666666669 [Mbps]
      Stable Throughput = 3.1503655913978492 [Mbps]
TCP3 Average Throughput = 2.1658410666666668 [Mbps]
      Stable Throughput = 2.196817204301075 [Mbps]
```

Figure 22 Console output for case 1

```
Simulation time      300
Initialization time  20
Active sources       [1 1 1 1]
TCP Windows          [5000 5000 5000]
Link delay           5ms
Link capacity        10Mb
Link buffer          65

TCP1 Average Throughput = 3.850961066666667 [Mbps]
      Stable Throughput = 3.8720860215053765 [Mbps]
TCP2 Average Throughput = 3.314561066666667 [Mbps]
      Stable Throughput = 3.3033978494623657 [Mbps]
TCP3 Average Throughput = 2.6910010666666668 [Mbps]
      Stable Throughput = 2.7033978494623656 [Mbps]
```

Figure 23 Console output for case 2

```
Simulation time      300
Initialization time  20
Active sources       [1 1 1 1]
TCP Windows          [5000 5000 5000]
Link delay           5ms
Link capacity        10Mb
Link buffer          65

TCP1 Average Throughput = 3.047921066666667 [Mbps]
      Stable Throughput = 3.0449462365591398 [Mbps]
TCP2 Average Throughput = 3.339161066666667 [Mbps]
      Stable Throughput = 3.4324301075268817 [Mbps]
TCP3 Average Throughput = 3.478801066666667 [Mbps]
      Stable Throughput = 3.3999139784946237 [Mbps]
```

Figure 24 Console output for case 3

```
Simulation time      300
Initialization time  20
Active sources       [1 1 1 1]
TCP Windows          [5000 5000 5000]
Link delay           5ms
Link capacity        10Mb
Link buffer          65

TCP1 Average Throughput = 2.579241066666667 [Mbps]
      Stable Throughput = 2.5493763440860215 [Mbps]
TCP2 Average Throughput = 4.251601066666668 [Mbps]
      Stable Throughput = 4.2627526881720437 [Mbps]
TCP3 Average Throughput = 3.044041066666669 [Mbps]
      Stable Throughput = 3.0674838709677421 [Mbps]
```

Figure 25 Console output for case 4

```
Simulation time       300
Initialization time   20
Active sources        [1 1 1 1]
TCP Windows           [5000 5000 5000]
Link delay            5ms
Link capacity         10Mb
Link buffer           65

TCP1 Average Throughput = 4.816361066666665 [Mbps]
      Stable Throughput = 5.0471397849462365 [Mbps]
TCP2 Average Throughput = 2.442001066666667 [Mbps]
      Stable Throughput = 2.5811182795698926 [Mbps]
TCP3 Average Throughput = 1.0239610666666668 [Mbps]
      Stable Throughput = 1.0754838709677419 [Mbps]
```

Figure 26 Console output for case 5

```
Simulation time       300
Initialization time   20
Active sources        [1 1 1 1]
TCP Windows           [5000 5000 5000]
Link delay            5ms
Link capacity         10Mb
Link buffer           65

TCP1 Average Throughput = 6.234401066666666 [Mbps]
      Stable Throughput = 6.2405591397849465 [Mbps]
TCP2 Average Throughput = 3.5809210666666669 [Mbps]
      Stable Throughput = 3.5834838709677421 [Mbps]
TCP3 Average Throughput = 0.05196106666666667 [Mbps]
      Stable Throughput = 0.054967741935483871 [Mbps]
```

Figure 27 Console output for case 6

Table containing parameters for each case:

Case number	Router-to-Router Delay	Access delays 1	Access delays 2	Access delays 3
1	5	5	10	15
2	5	10	10	15
3	5	10	10	5
4	5	1	3	5
5	5	50	100	150
6	5	5	10	500

Tables containing RTT and Bandwidth:

RTT1	RTT2	RTT3
30	50	70
50	50	70
50	50	30
14	22	30
210	410	610
30	50	2010

TCP1 Throughput	TCP2 Throughput	TCP3 Throughput	TCP Bandwidth usage [%]
4.532	3.1503	2.197	98.793
3.872	3.3034	2.7034	98.788
3.0449	3.432	3.4	98.769
2.549	4.263	3.0674	98.794
5.0471	2.581	1.0754	87.035
6.2406	3.583	0.05497	98.7857

Task 3 Conclusions

In the first part of the task (3.1) we can conclude that the bandwidth share is similar. The bandwidth usage is very high, above 96% in all cases. Also, the bandwidth usage grows smaller when the RTT increases. That is due to the fact that increase in RTT causes the buffer to drop some packets, therefore less information can be transmitted, freeing some bandwidth. In the second part of the task (3.2) we can see that the lower the delay on a connection, the more bandwidth that connection will receive. Also, cases 5 and 6 are especially interesting. In case 5 we tested how the bandwidth usage will change with very high RTT on all connections. As expected bandwidth usage drops drastically due to the packet loss discusted before. What is interesting, is that bandwidth usage in case 6 is again very high, but RTT on link 3 is also very high. We can see that since RTTs on links 1 and 2 is so low in comparison to link 3, the protocol assigns the whole bandwidth to these 2 connections and almost nothing to the last connection. That probably means that on the 3rd connection close to no data is transfered.