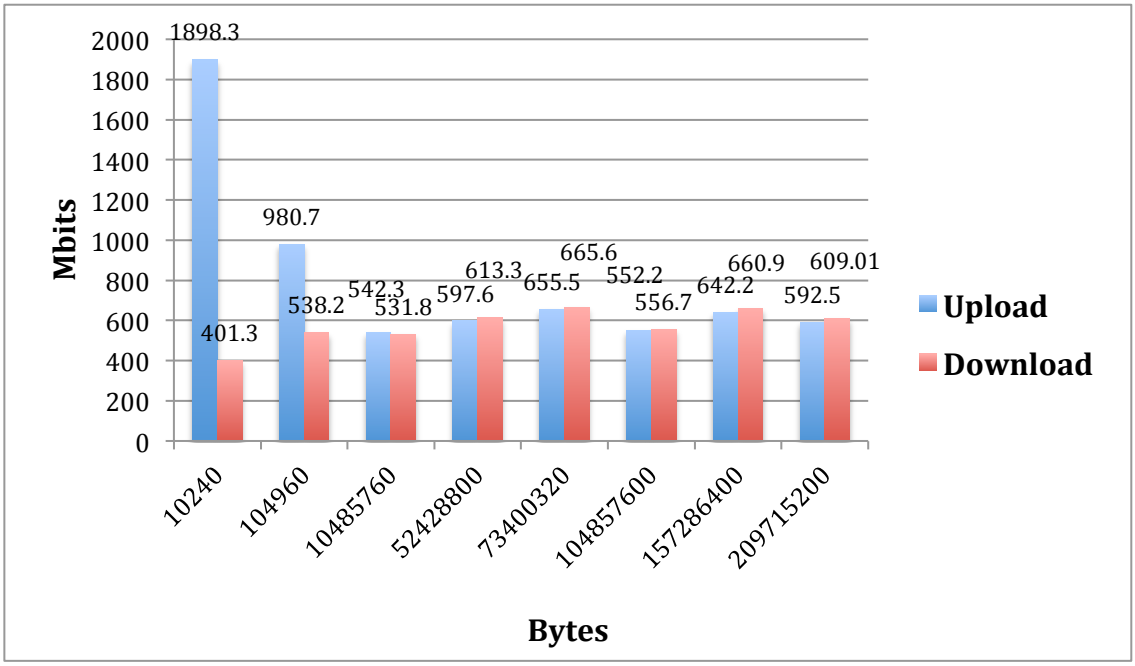
Bartlomiej Karmilowicz
ID: 37610929

CSE 589,Spring 2014

Project 1 Analysis

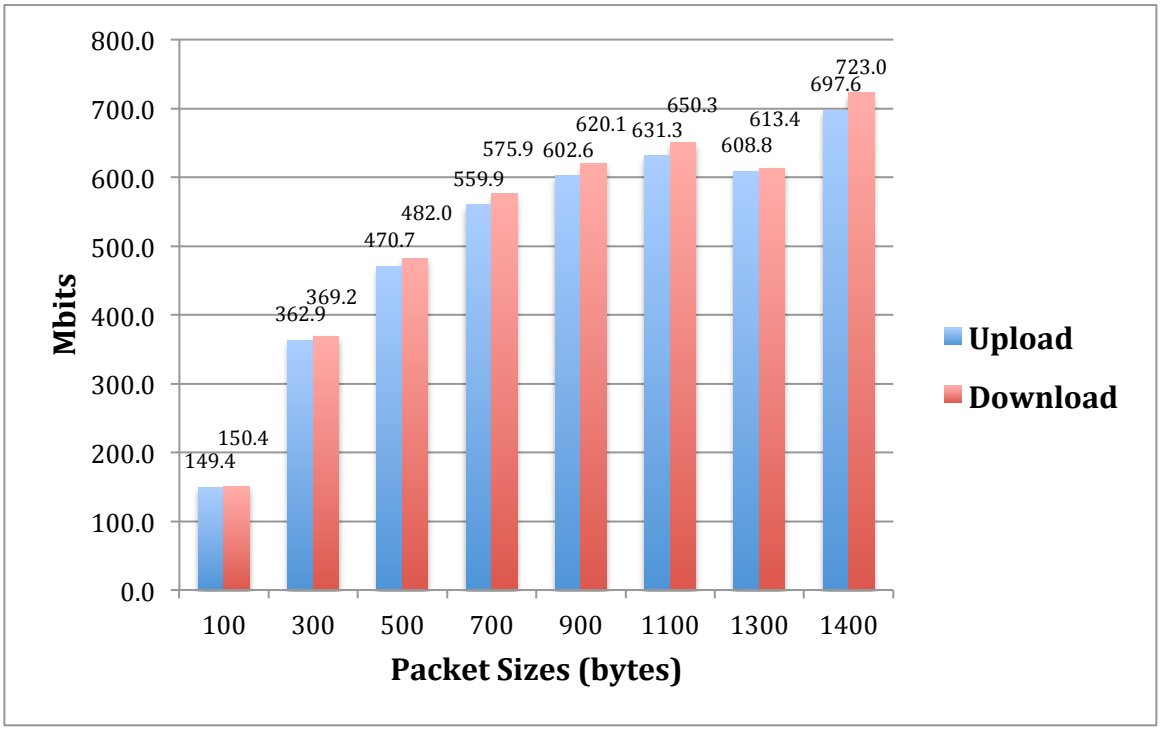
4.1 Data Rates Vs. File Size



File Sizes: 10 KB, 0.1 MB, 10 MB, 50MB, 70MB, 100MB, 150MB, 200MB
Packet Size: 1000 bytes

Observation: Based on visual inspection of graph, we can observe that as we increase file size, UPLOAD bandwidth is decreasing and DOWNLOAD is increasing. It's understandable since we have to send less packets in this case, which increases efficiency. On the other hand for large files we are pushing down the link a lot packets ,which cause the upload speed to decrease.

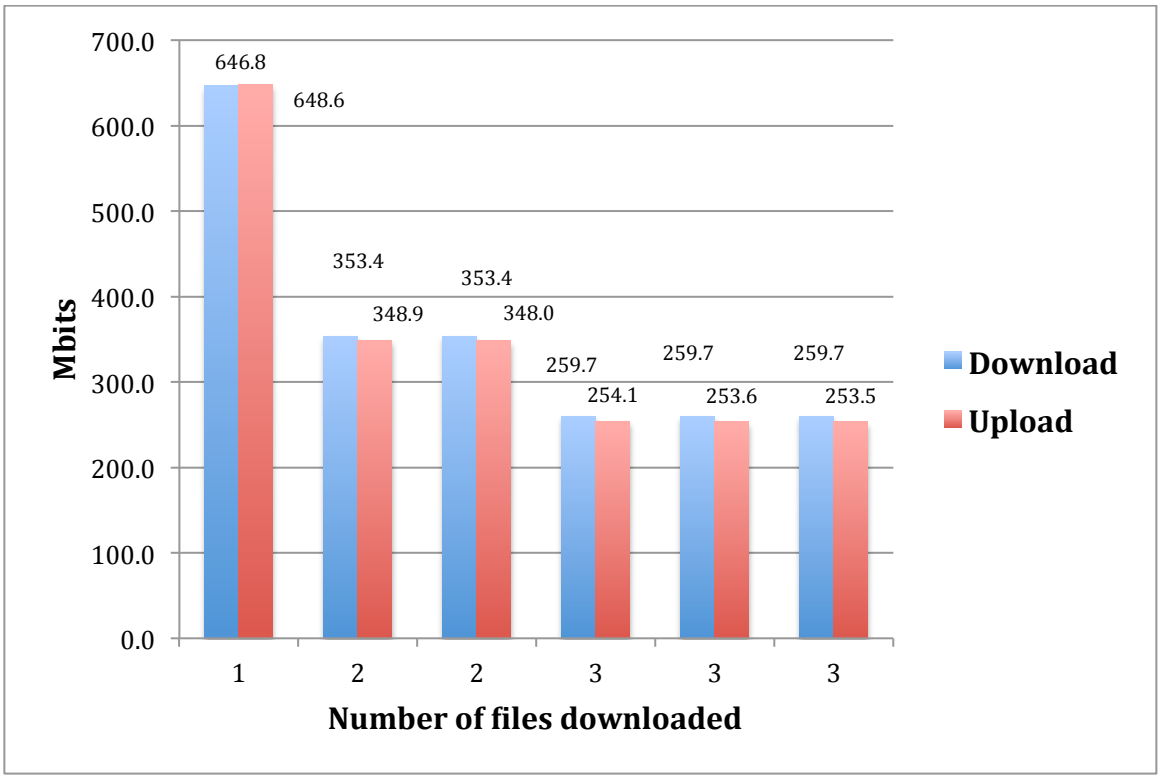
4.2 Data Rates Vs. Packet Size



File Sizes: 200 Mbytes

Observation: Based on visual inspection of graph, we can observe that as we increase packet size, tx and rx rates are increasing. We can observe significant change from 100 bytes to 300 bytes, which progresses until 900 bytes. Then the speed starts growing slower. I expected this kind of result since small packet sizes increases the number of operations (packets) we have to do to upload and download file. On the other hand if I would keep increasing packet sizes to enormous amounts I could throttle the link. This project made me understand that it is a challenging task to estimate the best packet size, which requires a lot of statistics and computations.

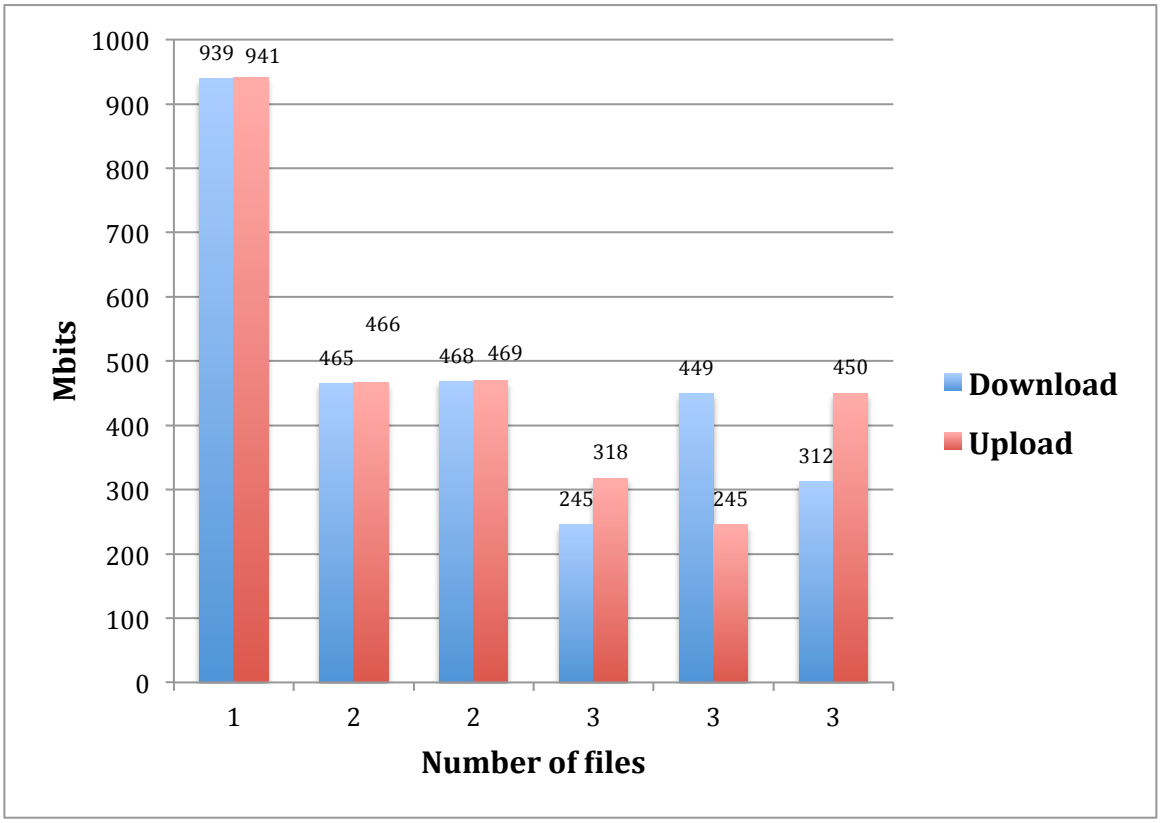
4.2 Data Rates Vs. Load Variations



File Size: 70 Mbytes

Observation: Based on visual inspection of graph, we can observe that as we increase the number of files being downloaded, tx and rx rates are significantly decreasing by a factor of 1/(number of files). I expected this kind of behavior since there is a fixed amount of bandwidth allocated for each file.

4.2.1 Data Rates Vs. Load Variations – iPerf



Observation: Based on experiments executed on iPerf measurement tool I recorded values presented above. It was a quite challenging task to execute those in the same point of time (especially for 3 files) but It's really easy to observe that the bandwidth is also divided between number of files. The only difference between my system results and iPerf results is the amount of available bandwidth. We can observe that In my case only 650 Mbits were utilized, whereas iPerf had 939 Mbits. My understanding of this difference is the way that iPerf is pushing the packets. My program may generate a lot of overhead, whereas iPerf is using the most efficient way to do it.