Hu Huiwen A021114H

EE4204 Lab Report

Requirements:

Measure the **message transfer time** and **throughput** for various sizes of data-units and compare it with the stop-and-wait protocol where the batch size is always fixed to be 1.

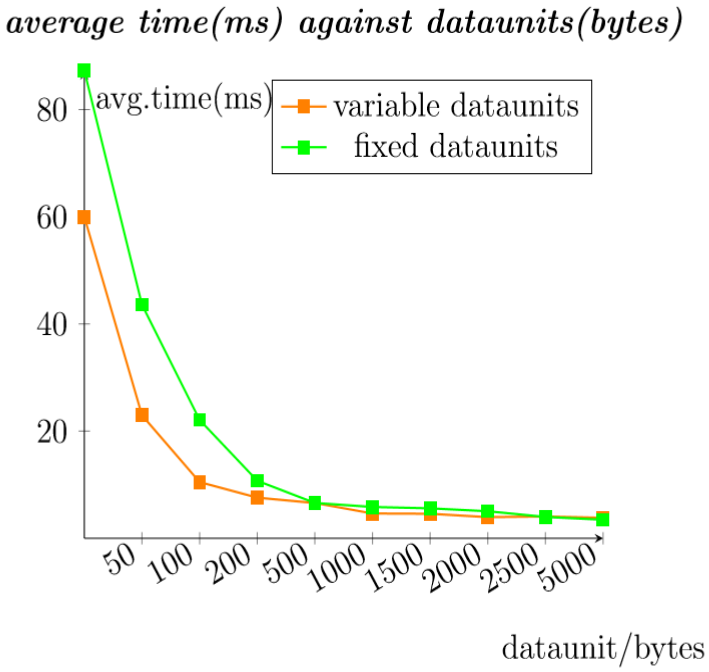
Repeat the experiment several times and plot the average values in a report with a brief description of results, assumptions mad.

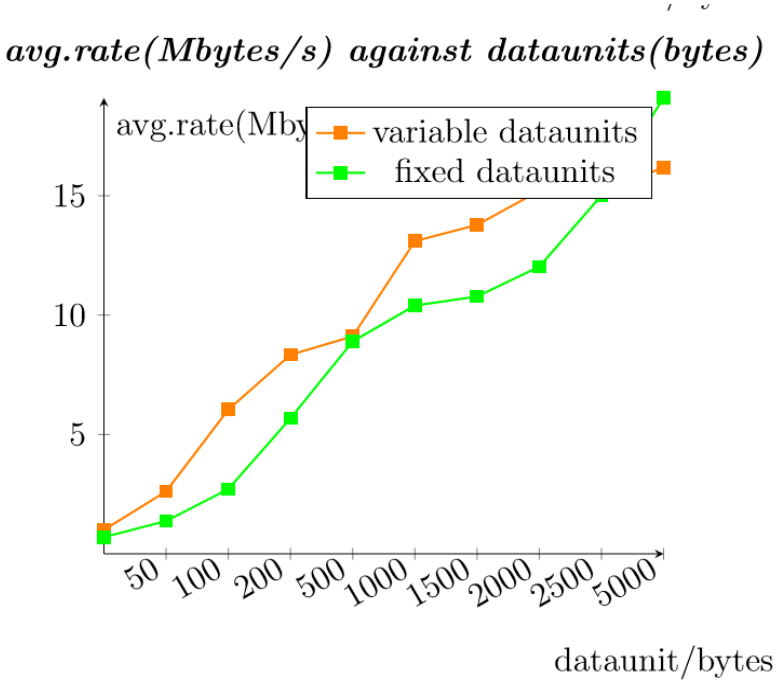
Assumptions:

The results obtained are done on the same computer by making the client send to the IP address of the server. It is assumed that the performance obtained in this way is similar to that between different computers.

Results:

Different dataunits (i.e. variable DATALEN) from 20 to 5000 were experimented with in order. The results are as follows:





1.From the graph, we can see that both UDP and Stop and Wait protocols generally follows the same trend, increases in average data rate and decreases in average transfer time when the size of DATALEN is increased from 20 to 5000. This is because when DATALEN increases, less time is spent on waiting for the ACK.

2.The average data rate of UDP is always higher than that of Stop and Wait. This is because UDP spends less time waiting for the ACK.

3.However, when DATALEN is increased, the differences between the two protocols become less. This is because since fewer packets are exchanged, fewer acks are transmitted and received, so the differences adding is less.

4. When DATALEN is bigger than 3000 bytes, the stop and wait protocol has better performance, this could be due to the way the transmission is implemented, maybe is because the stop and wait protocol need not track the state due to the same batch sizes, while the variable DU method uses a counter to track the state. What's more, the variable DU method fails several times after sending the file the first time, so I cannot send continuously without restarting the server, maybe it breaks out of the blocking recvfrom() call. While the fixed DU can transferring continuously.