

COL334 - Assignment 2 Report

Harshit Mawandia - 2020CS10348

I have attached files for both the first and second part to the zip.

Analysis:

1. For the first part:

Time Taken for transfer of one of the chunk is 0.0160 seconds

Here is a [link](#) to detail of each captured packet by wireshark with RTT

For Average time calculation:

Total Time taken for complete transfer of file to N clients : 9.83840 seconds

Total Number of chunks exchanged captured by wireshark = 668

Average time taken by each chunk = 0.0148 seconds

For the second part:

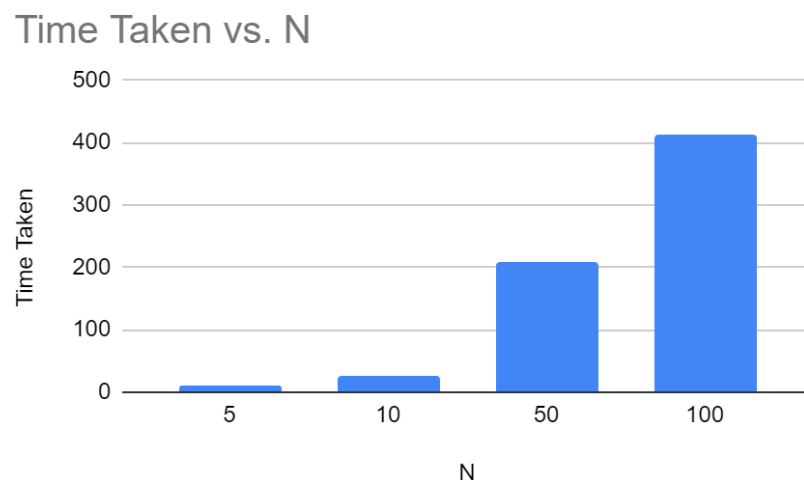
Time Taken for transfer of one chunk is 0.018 seconds

The average time for each chunk is around 0.0173 seconds

The RTT in second part is higher

Yes, this was expected as for UDP requests, connection has to be initiated each time while in TCP request connection has to only be made once.

2. The average RTT for some chunks which had been lost due to timeout is greater than the rest.
3. Time Taken for transfer for different N:

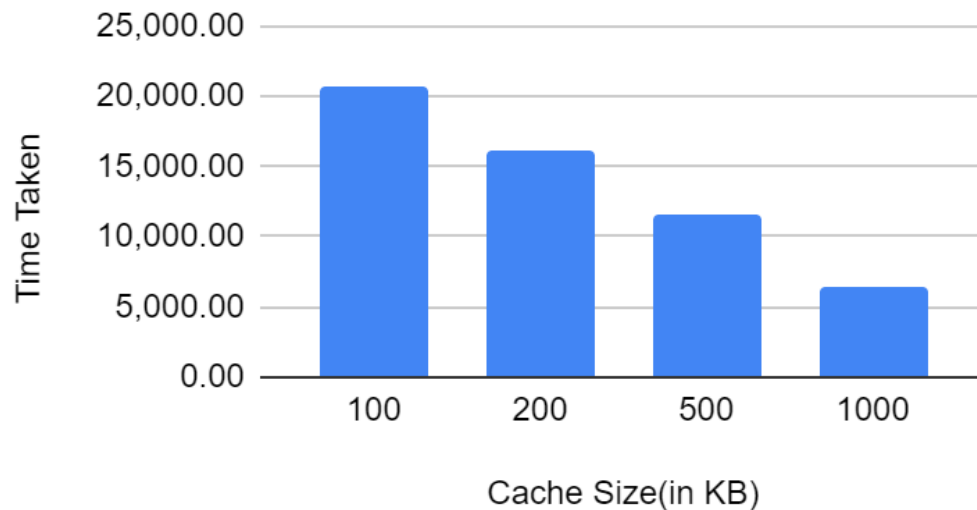


The time taken increases with N.

Yes, this trend was expected as with N the number of chunks needed to transfer increases by more than the file size.

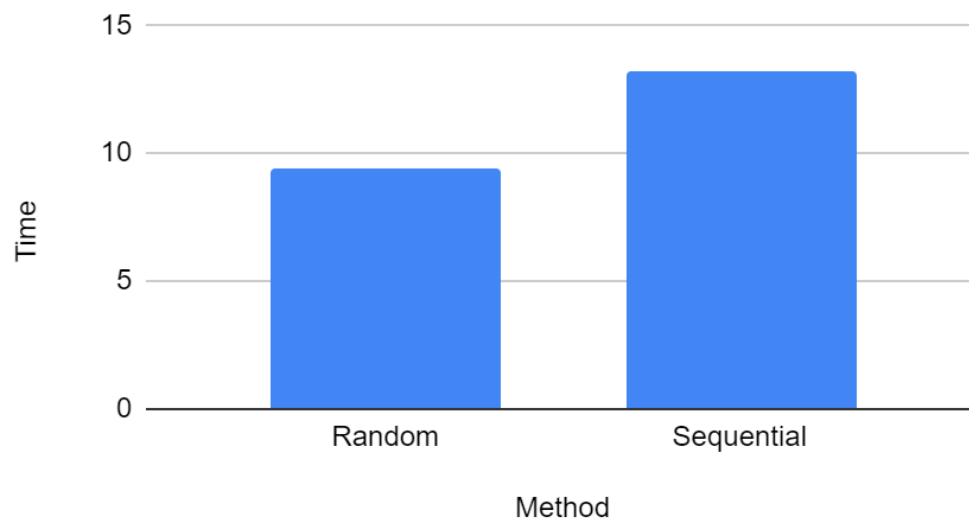
4. As the time taken for A2_large_file.txt was too long to be simulated, I used a file of size 5 MB. The simulation for that resulted in the following result for N=100

Time Taken vs. Cache Size(in KB)



5. Sequential method takes more time than random as during sequential method as there are multiple request for file transfers to a single client, which adds up in the buffer and are handled sequentially.

Time vs. Method



Food For Thought:

1. The advantage this network has over Traditional is that the server does not need to store the entire file in its server. This has great advantages when there are thousands of files worth gigabytes of data as it saves a lot of cost for the server with a small cost of time borne by the clients. This increases scalability.
2. The advantages of PSP over P2P are
 - a. The burden is not on the client to ensure that at least someone has the data
 - b. Client does not have to find another client to fetch data from, that job still remains with the server which has more reliability.
 - c. The presence of a cache decreases the responsibility on the clients.
3. I would initially add the name of the file, or maybe index the files and add the first 2 bytes of each chunk as the index as an integer to identify the files. The other methods can remain intact, also I'll make a dictionary which keeps track of the files with index on the client side as well.