

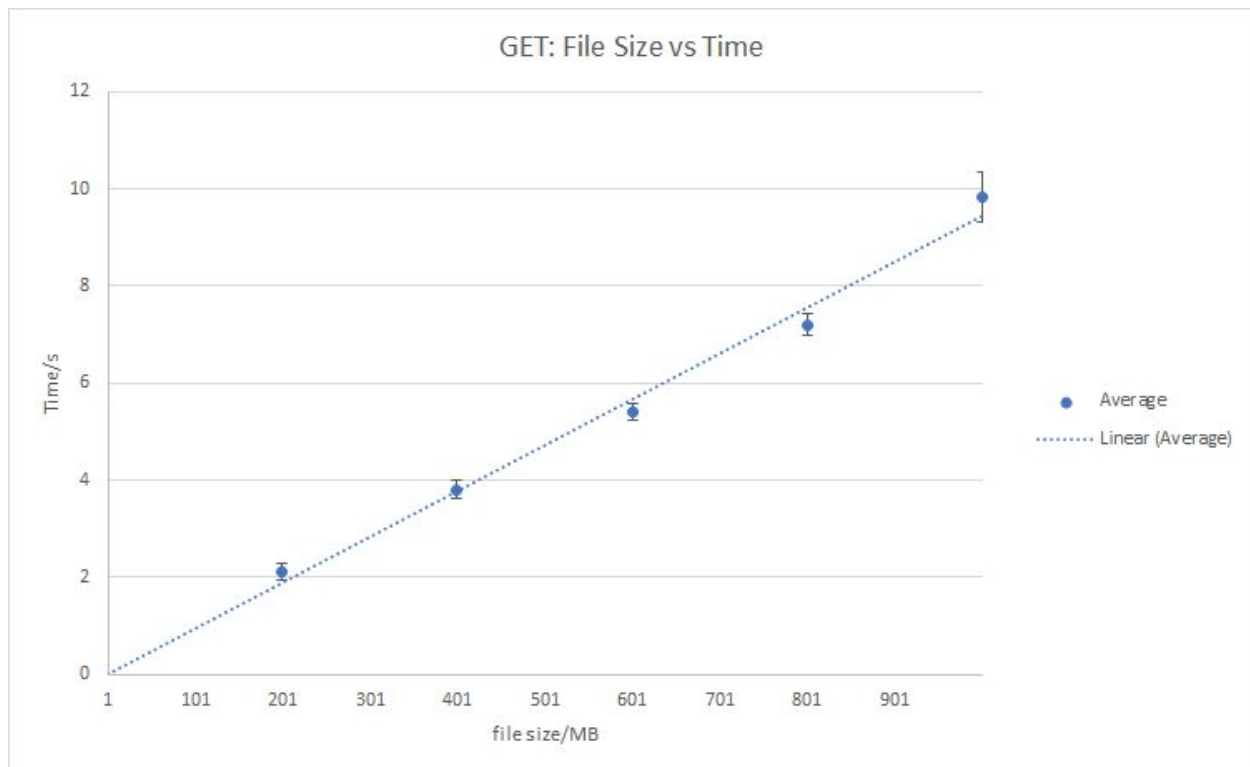
CS 425 Mp2 report (Group 10)

Design

Our Simple Distributed File System (SDFS) works by electing a single master that handles and queues all file operations. The master is elected by picking the node with the smallest IP address from a sorted list of all node IP addresses. As a result, any node can become the master. To ensure consistency, we constantly gossip our file dictionary (a dictionary containing a list of files along with the IPs of the nodes that contain them) alongside our membership list. Hence if a master fails, a new master can be elected easily with its file information kept up-to-date. When a file is inserted, to decide which nodes get a replica, we hash the file name to get an index unique to it corresponding to the list of sorted node ips. We then take four nodes around that index value. To allow upto three failures, we replicate each file four times to ensure that at least one copy is present if three failures occur. Lastly, if a failure does occur, for each file in the failed node, we replicate that file in the node with the lowest IP address that does not already contain the file. This ensures that 4 replicas of each file are always maintained in the system. All message requests in our system are sent using UDP for speed but all file transfers are performed using TCP for reliability.

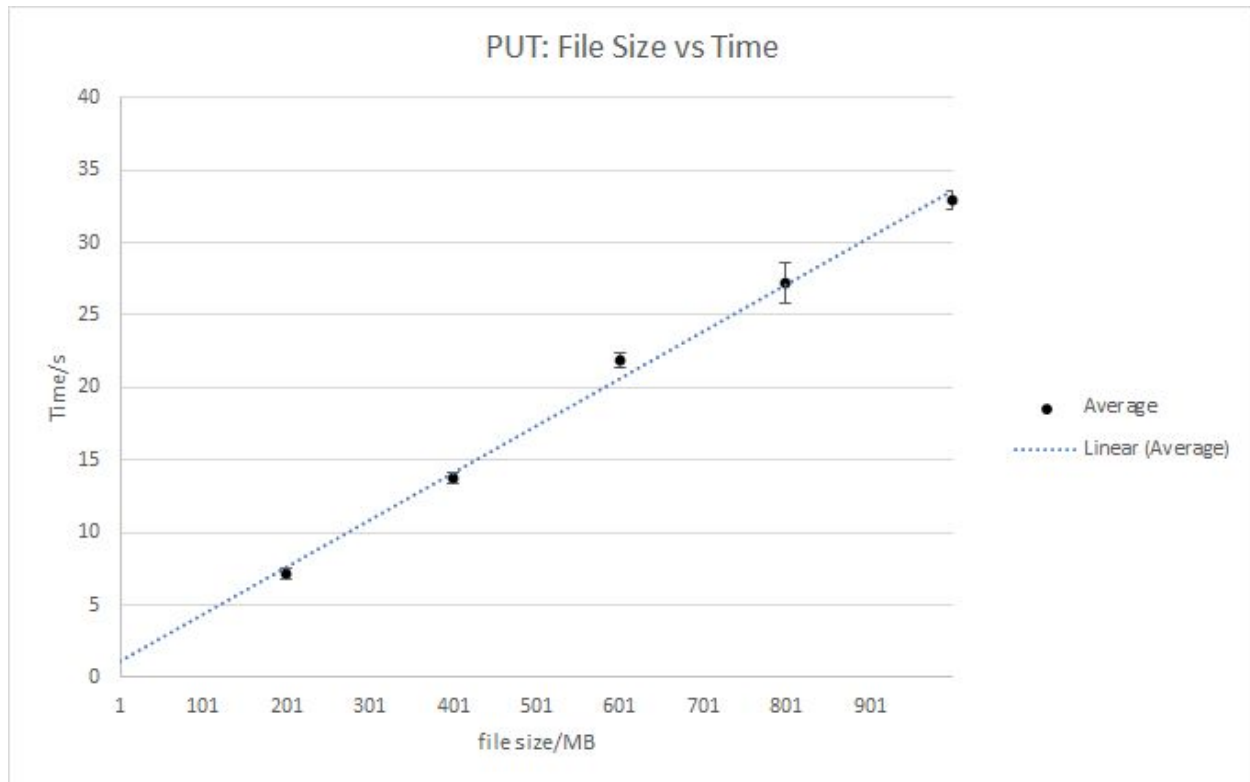
Graphs

1. Time to get a file vs file size



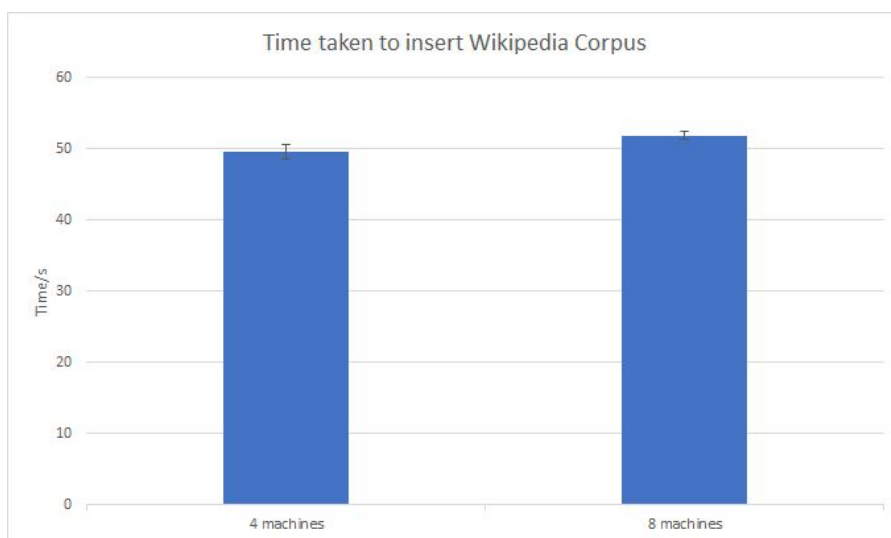
We see that as the file size increases (from 1MB to 1GB), the average time taken to get a file linearly increases. This is expected since the larger the file size, the more shards of the file we create to send.

2. Time to put a file vs file size



Similar to graph1, as file size increases, the average time taken to put a file in the SDFS increases linearly. We see that the time to put a file is greater for each file size as compared to the time to get a file. Each put request is completed after 4 replicas are transferred into the sdfs. Hence the time taken for a 'put' request is about four times bigger than the time taken for a 'get' request.

3. Time to store the entire English Wikipedia corpus into SDFS



We used the zip version of the English wikipedia corpus. We see that the time taken to store it into an SDFS with 4 machines is almost equal to the time taken to store it into an SDFS with 8 machines (within the error bound as depicted by the error bars). This is expected because the only difference in computation times arises

while sorting the list of IP addresses to elect the master node.