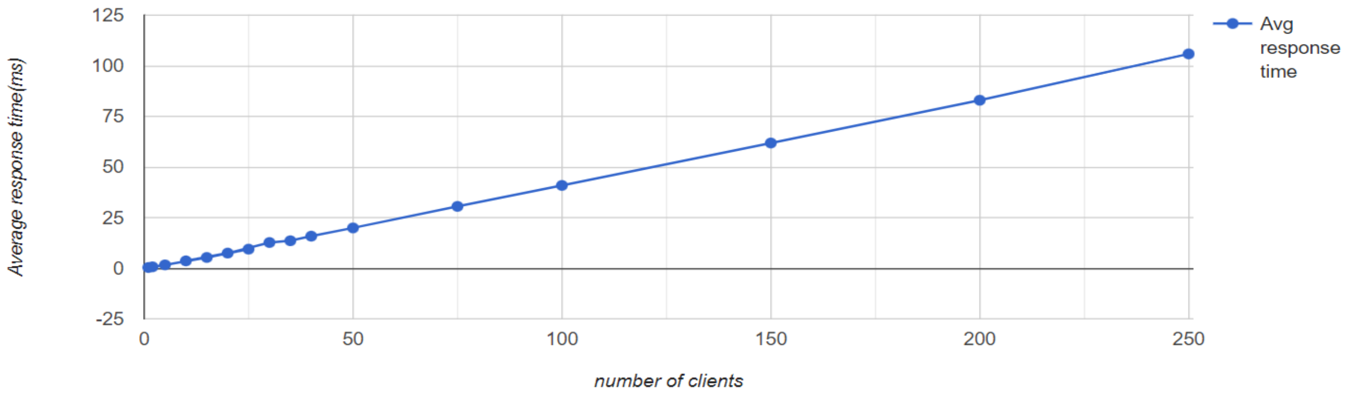
Performance Report

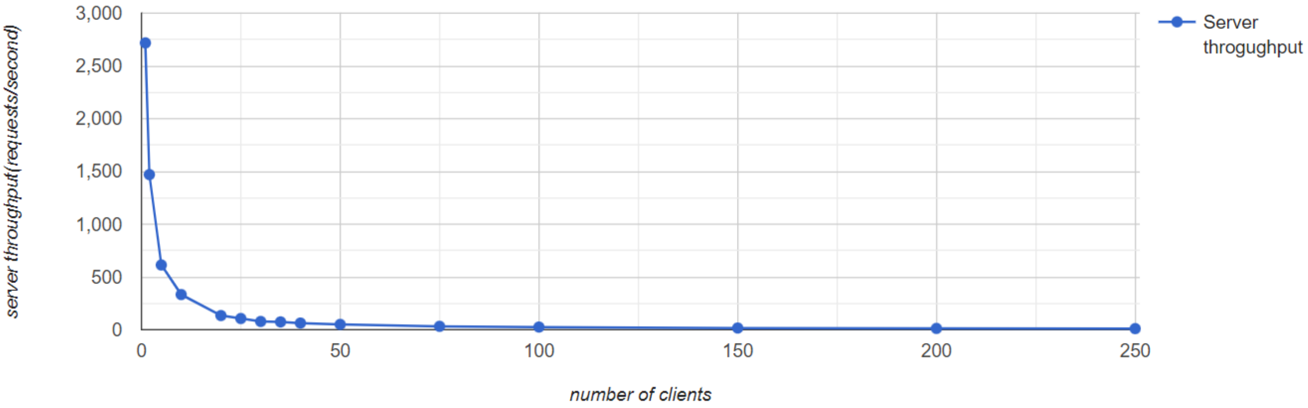
Nikhil Barhate and Sriranga Kalkunte Ramaswamy

CSCI 5673 Distributed Systems programming assignment 1

Performance figures

	Number of buyer instances	Number of seller instances	Response time (ms)	Throughput (requests / sec)
1	1	1	0.368	2718
2	10	10	3.576	280
3	100	100	40.894	25





Explanation

1. Average response time: We measured the response time for each request and averaged it over a 1000 API calls. The Graph indicates that the average response time increases linearly with the increase in the number of client instances. The linear increase in response time indicates that the server's processing capacity is not able to keep up with the increase in the number of clients, resulting in longer waiting times for clients.

2. Average Server throughput: Throughput refers to the amount of work that a system can handle per unit of time. In this case, the average server throughput is the average number of requests the server is able to handle in a given time period. As we can see from the graph and the table, the server throughput reduces exponentially with the increase in number of client instances. The exponential decrease in performance suggests that the server is struggling to process client requests with the increased number of clients.

Since TCP/IP sockets need to hold a connection until the request is processed by the server and the response is reliably transmitted to the client, other clients' requests have to wait in the server queue for a long time. Large numbers of clients lead to high contention for the one TCP/IP port that the server is listening on. These findings suggest that using TCP/IP sockets may not be able to scale effectively as the number of clients increases. Running multiple instances of the server could mitigate this issue to a certain extent.