**Average Response Time and Throughput**

Below table shows the performance metrics which were measured as a part of the assignment. We need to understand that the performance metrics will depend on the system performance, how efficient is the concurrent code and optimizations. These tests were conducted locally and not on cloud instances.

For these tests File I/O was turned off, database was maintained in memory. In the next section we can see Session Latency tests where File I/O was kept on.

Table

Description automatically generated

Chart, line chart, scatter chart

Description automatically generated

To measure Average Response Time, Buyers/Sellers were spawned, and time taken to complete 10 operations per Buyer/Seller was calculated. We can see that the latency increases linearly with increase in number of concurrent instances. This is probably because of how Go language schedules go routines parallelly on available threads. With increasing number of Goroutines, they get less and less amount of scheduled time. This increases latency and time of completion.

Chart, line chart, scatter chart

Description automatically generated

In the above image we can observe that the number of operations per second increases as we increase the number of Buyers/Sellers instances, each making 1000API calls. This is just a function of how many CPU threads are utilized concurrently and at what capacity. At 100 Buyers/Sellers I observed a CPU utilization of 75% over 16 threads.

The number of operations per second can further be increased by taking a closer look at server algorithms and parallelizing efficiently.

With File I/O these numbers take a huge hit, about 1/9th the number of operations. This is again dependent on the Disk speed and other factors.

**Session Latency**

Sessions are defined in the README file.

Sessions were created to simulate real world interactive usage but automated to measure time taken to finish a session.

Table

Description automatically generated Table

Description automatically generated

We can see that the Client Buyer sessions upto 200 sessions concurrently completed without any issue with a linear scaling in session latency. On the other hand Client seller sessions started finishing with error after 70 concurrent sessions.

These were probably caused due to File I/O being turned on and the nested loop operations. Latency numbers without File I/O are future work.