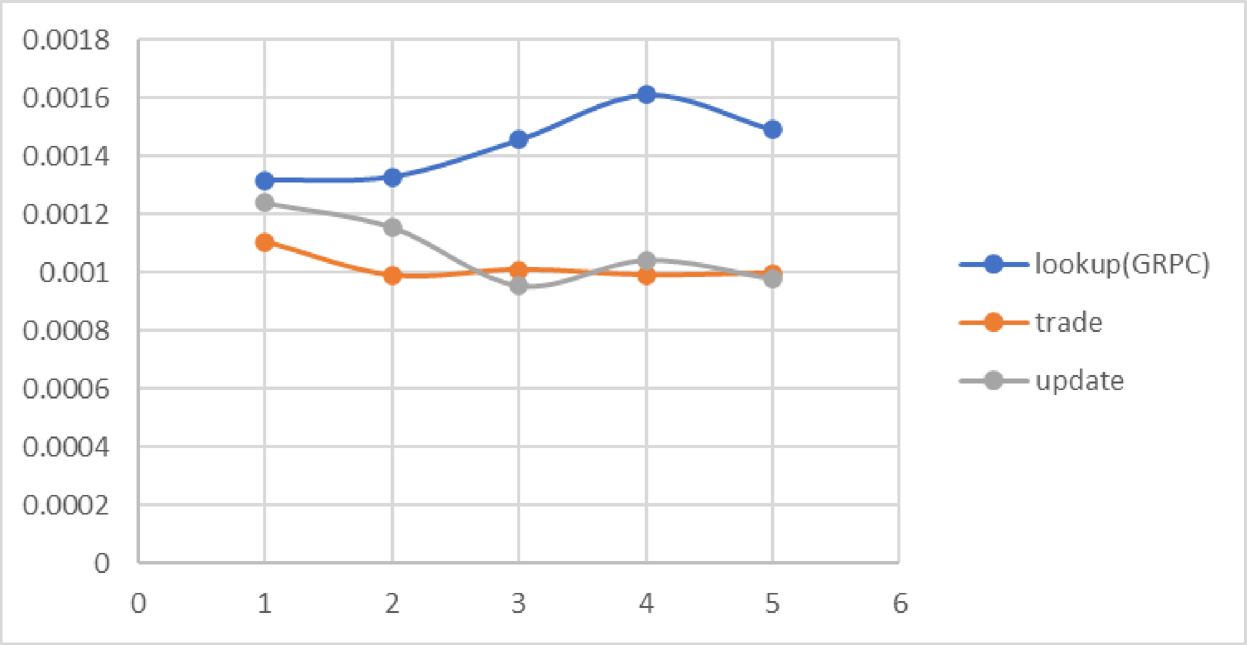

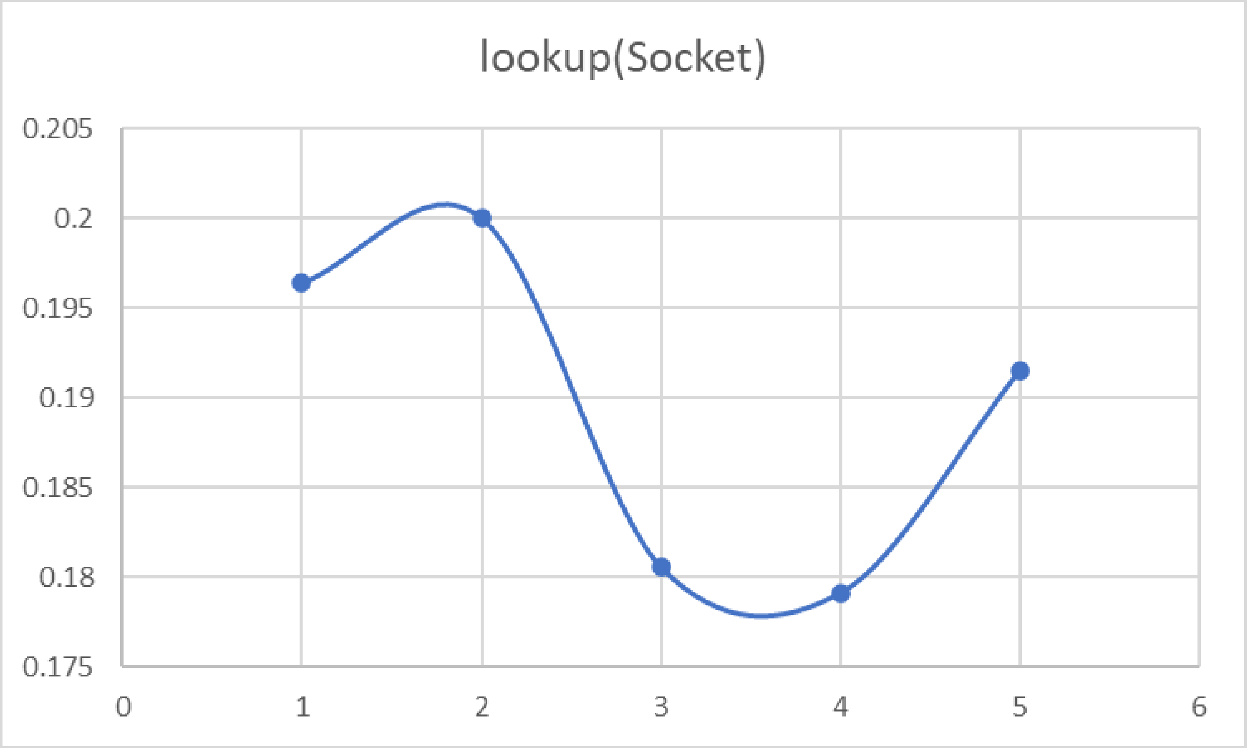

Answers to the questions

1. How does the latency of Lookup compare across part 1 and part 2? Is one more efficient than the other?

Latency of part1 is bigger than part2. It's obvious that gRPC is more efficient than socket.

2. How does the latency change as the number of clients (load) is varied? Does a load increase impact response time?

As the number of clients increases, the latency of Lookup tends to be bigger and then smaller, while update has the opposite results and the latency for trade tends to stay still.

3. How does the latency of lookup compare to trade? You can measure latency of each by having clients only issue lookup requests and only issue trade requests and measure the latency of each separately. Does synchronization pay a role in your design and impact performance of each? While you are not expected to do so, use of read-write locks should ideally cause lookup requests (with read locks) to be be faster than trade requests (which need write locks). Your design may differ, and you should observe if your measurements show any differences for lookup and trade based on your design.

Latency of lookup is bigger than trade.

4. In part 1, what happens when the number of clients is larger than the size of the static thread pool? Does the response time increase due to request waiting?

When the number of clients is larger than the size of the static thread pool, the rest clients need to wait. Yes