gRPC vs regular sockets
performance comparison ——

We note here first that due to the limited scale of our simple messaging service and the small number of threads allotted for both server and client, we do not see qualitative differences in performance.

This being said, our implementations of both methods have their own advantages and drawbacks. For one, our non-gRPC service limits received chunks of data to 1024 bytes. This limit exists so the client socket will always know when to stop reading, but this makes sending larger messages more complicated. On the other hand, our gRPC implementation allows for much more data to be sent, but incurs an extra overhead (the "request" and "context" data structures sent with every request contribute 64 and 48 bytes of overhead each, respectively).

A final comment on our gRPC performance — because we do not implement bi-directional streaming, the client continuously sends requests to the server to ask for any new material to read (e.g. a new message). In smaller devices, this continuous extra use of another thread may take up too much memory/CPU resources.