CS F303 Computer Networks: 2019-20 semester II
Assignment 3

Protocol

We have implemented an application level protocol which uses UDP sockets to transfer packets between a client and a server. The protocol uses a stop and wait method as well as added md5 checksum to provide reliability in terms of packet loss or packet corruption. It is a half duplex protocol. It has a Sender class and a Receiver class which the application will have to instantiate to create the half duplex connection. The protocol sends an ACK along with the sequence number when it recieves a packet with a particular sequence number.

Application Assumptions

As the protocol is reliable but half duplex, we assumed that the application developer will create 2 instances to make it a full duplex connection. If the a packet is lost for more than 7000 times, then we return an error. We expect the application to take the necessary steps thereafter.

Network Assumptions

We expect the network to be have less than 99.985% packet loss. If the packet loss is more than this then our protocol will most likely return an error as the same packet would have dropped for more than 7000 times.

Protocol Strategies for Network Problems

1. Packet Loss

In the Send class, the protocol keeps resending the packet until the ACK for the corresponding packet is received. But this resending goes on for at most 7000 times, after which the protocol shows an error.

In the Receive class, the protocol checks whether the sequence number of the received packet is one more than the previously received packet. If yes then it saves it in the buffer and sends an ACK with the new sequence number after unpacking else drops the packet.

2. Packet Delay

As this is a simple stop and go protocol, the delay would not affect much.
In the Send class, if the ACK packet is delayed, then the Receive class would again send the packet, and thus the Send class would again send ACK. If 2 ACKs arrive simultaneously then the one with the bigger sequence number will be registered and the next packet, if any, would be sent.