CSS432 A

Documentation – Homework 3

Sliding Window

Johnathan Hewit

# **Design/Algorithm**

## Stop-and-Wait

The client function, clientStopWait, is a function that operates as a simple two-way communication using UDP protocol. It begins by receiving its parameters from the driver class, hw3.cpp, and borrows some attributes from classes to create/manage the socket and set a timer from UdpSocket and Timer, respectively. The function starts a timer initially and sends a message or packet to the server. It then waits for an acknowledgement (ACK) from the server, up to 1500 usec. If it does not receive an ACK from the server, it resends the original message and proceeds in that fashion until all messages have been sent.

The server function, serverReliable, is a function that operates as a simple two-way communication using UDP protocol. It begins by receiving its parameters from the driver class, hw3.cpp, and borrows some attributes from classes to create/manage the socket from UdpSocket. The server checks if the message has arrived and if the message includes the sequence number it’s expecting. If it does, the server responds with an ACK to the client. If it doesn’t, it allows the client to timeout in order to retransmit the correct message.

## Sliding Window

The client function, clientSlidingWindow, is a function that operates as a simple two-way communication using UDP protocol. It begins by receiving its parameters from the driver class, hw3.cpp, and borrows some attributes from classes to create/manage the socket and set a timer from UdpSocket and Timer, respectively. The function proceeds through each sequence number or message of the number dictated by the driver. Next, it sends those unacknowledged (unACKed) messages to a client in succession up until the end of its window, which is also receives as a parameter dictated by the driver. Once all messages in the window are sent, the client starts a timer and awaits the ACKs from the server. If the server responds before the timer ends, the client checks the content to see if the ACK it receives matches the last sequence number it sent. If it does, the client proceeds to the next sliding window. If it does not, or the timer runs out, the client retransmits the last ACKd message from the server.

The server function, serverEarlyTrans, is a function that operates as a simple two-way communication using UDP protocol. It begins by receiving its parameters from the driver class, hw3.cpp, and borrows some attributes from classes to create/manage the socket from UdpSocket. The server proceeds through each sequence number or message of the number dictated by the driver and waits for an incoming message from the client. In the test case 3 version of the assignment, the server receives the message and checks to see if it matches the expected sequence number (in-order). If it does, the server looks to see if there window is full and if so, it sends the ACK; if not, it waits for the remainder of the frames to be checked. If the sequence number received does not match the expected in-order number, the server allows the client to timeout and retransmit the last ACK’d message. In the test case 4 version, the only addition is adding a 0 – 10% chance of dropping an incoming packet by causing a timeout.