CS 5450 Report: Project 2 Go-Back-N

By Mengran Wang (mw866@cornell.edu) and Ruiheng Wang (rw533@cornell.edu)

Introduction

In this project, a Go-Back-N network protocol is implemented in C using the socket library. This report aims to briefly explain the design and implementation challenges.

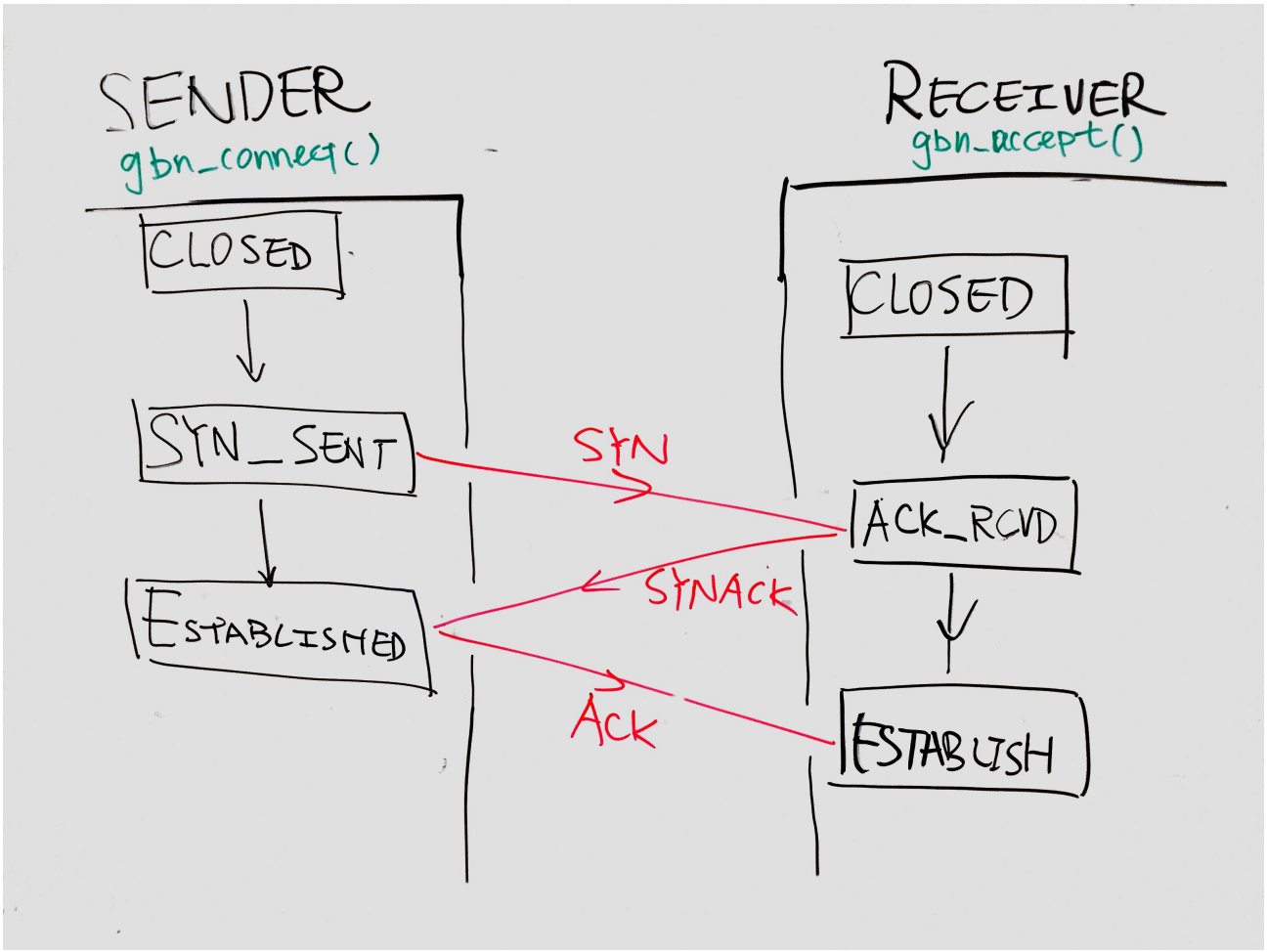
A general description of the protocol

The prototol consists of the followoing three stages: (a) connection setup, (b) data transmission, and (c) connection teardown.

Connection Setup

First, the connection is set up with a 3-way handshake between the sender and receiver as shown in the diagram below.

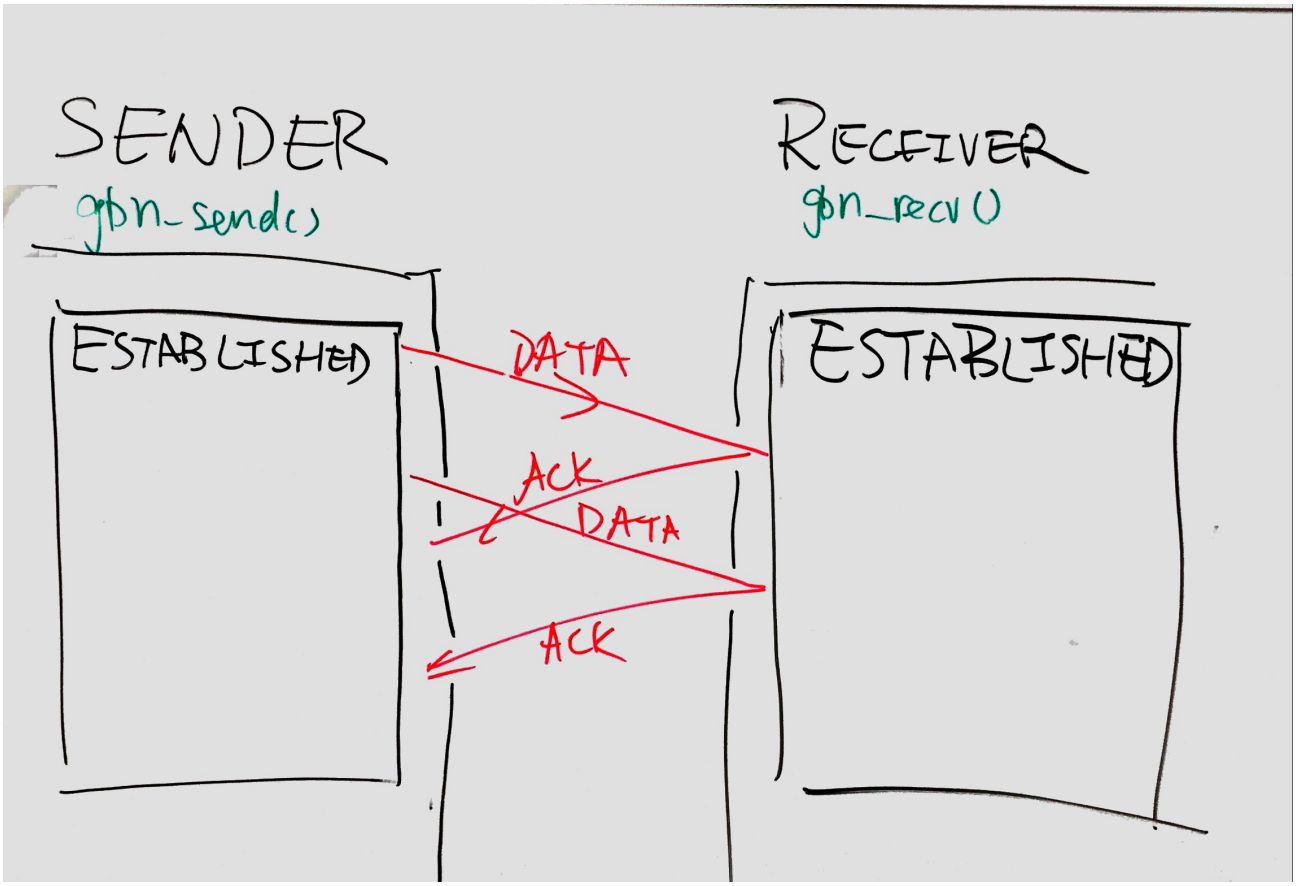
It is implemented on the **sender** in `gbn_connect()` and on the **receiver** in `gbn_accept()`.



Data Transmission

Once the connection is established, the data is sent in DATA packets by the **sender** in `gbn_send()`. Each DATA packet is acknowleged by the **receiver** by sending an ACK packet in `gnb_recv()`.

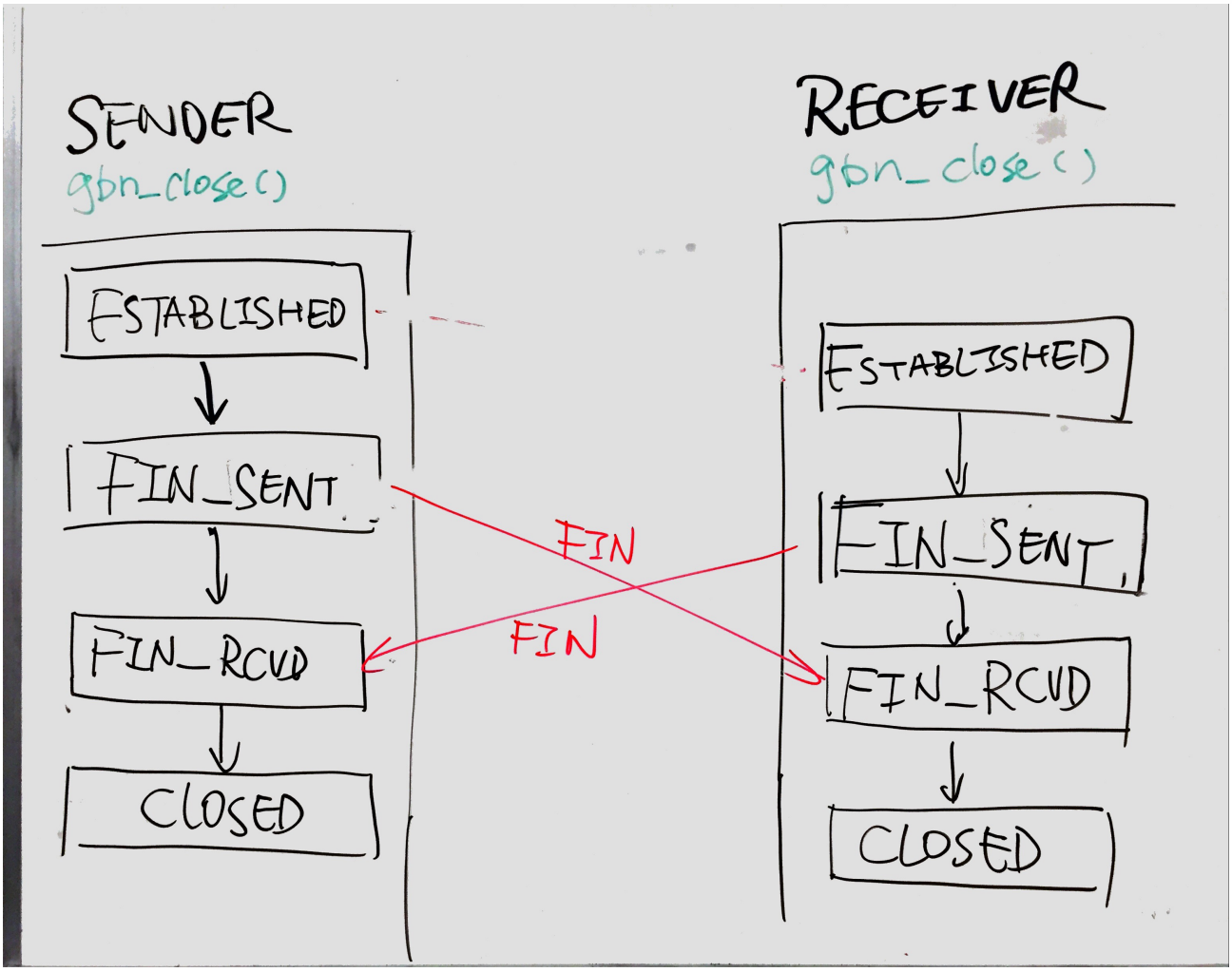
The design is illustrated in the diagram below.



Connection Teardown

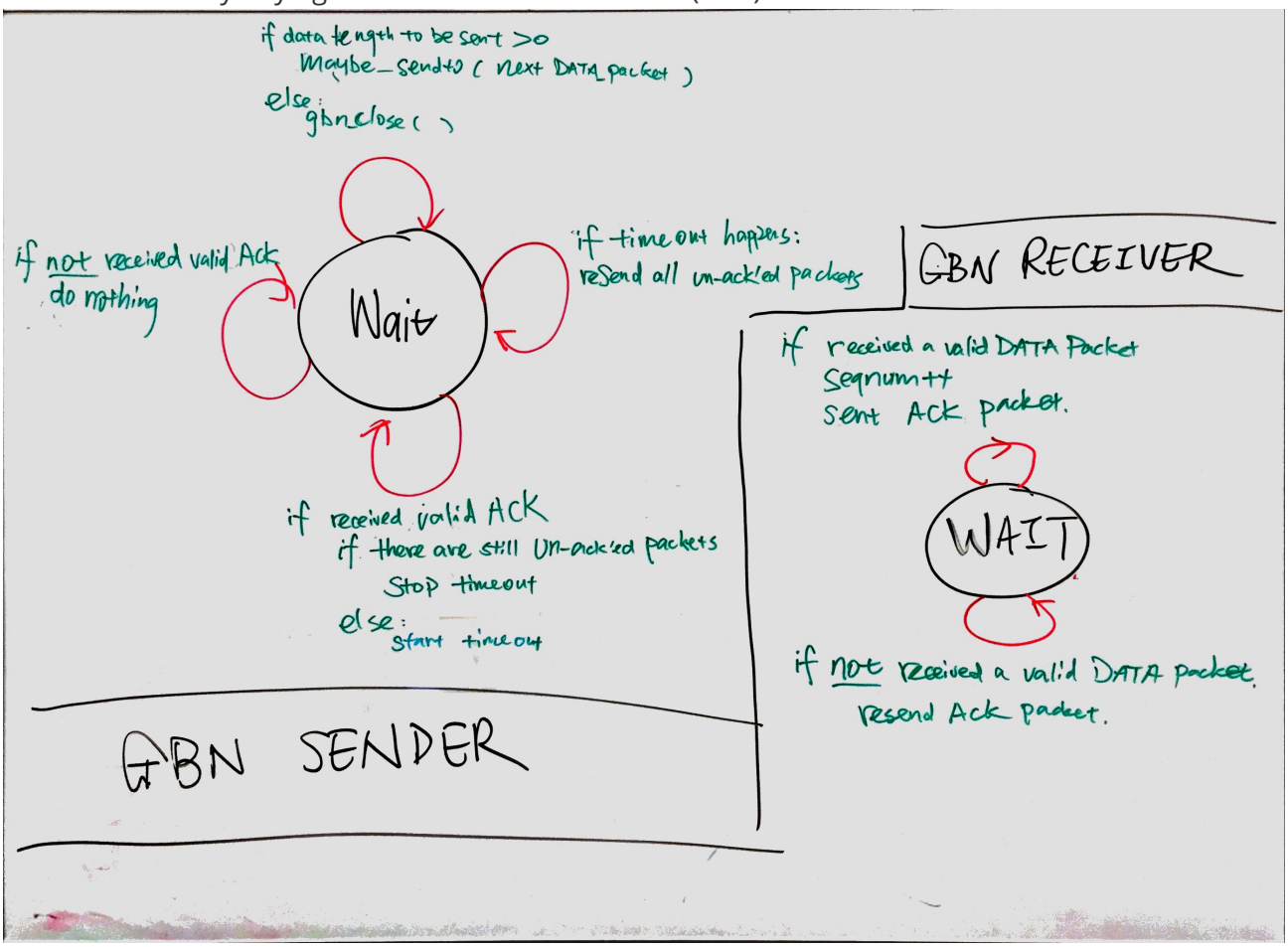
One the data transmission is completed, the connection is torn down by both sender and receiver by sending FIN packets in `gbn_close()`.

The design is illustrated in the diagram below.



A paragraph about the tricky parts of the implementation

The most tricky part of the implementation is figuring out the logic for the the `gbn_sender()` and `gbn_receiver()`.
We tackled this by relying on the Finite State Machine (FSM) as below:



Known Issue

There are timeout issues when sending and receiving ACK during data transmission using `gun_send()` and `gbn_recv()`. Despite hours spent, we were unable to resolve this issue. We tested that 3-way hand shake works successfully. Then we tested with `sendto()` function and it works as well. However, after changing with maybe_sendto() function the timeout issue occurs. We thought it possible occurs from `gnn_receive().`

Output from `sender`:

```
$ ./sender 127.0.0.1 9999 README.md
FUNCTION: gbn_socket()... Create socket.... socket_descriptor: 4
FUNCTION: gbn_connect()  4...
STATE: SYN_SENT
SUCCESS: Sent SYN.
SUCCESS: Received SYNACK...
type: 1 seqnum:73checksum(received)65206checksum(calculated)65206
SUCCESS: Received valid SYN_ACK!
```

```
FUNCTION: gbn_send() 4...
STATE: ESTABLISHED
INFO: DATA length of 429 packets left to be sent...
SUCCESS: Sent DATA packet (73)...
type: 2 73seqnum: 46584 checksum(received): 46584       checksum(calculated):
SUCCESS: Received ACK packet.
SUCCESS: Received valid SYNACK packet.
SUCCESS: Sent SYNACK.
ERROR: Unable to receive ACK!
ERROR: Timeout when receiving ACK.
STATE: ESTABLISHED
INFO: DATA length of 429 packets left to be sent...
SUCCESS: Sent DATA packet (73)...
type: 2 73seqnum: 46584 checksum(received): 46584       checksum(calculated):
SUCCESS: Received ACK packet.
ERROR: Unable to receive ACK!
ERROR: Timeout when receiving ACK.
STATE: ESTABLISHED
INFO: DATA length of 429 packets left to be sent...
SUCCESS: Sent DATA packet (73)...
type: 2 73seqnum: 46584 checksum(received): 46584       checksum(calculated):
SUCCESS: Received ACK packet.
ERROR: Unable to receive ACK!
ERROR: Timeout when receiving ACK.
STATE: ESTABLISHED
INFO: DATA length of 429 packets left to be sent...
SUCCESS: Sent DATA packet (73)...
type: 2 73seqnum: 46584 checksum(received): 46584       checksum(calculated):
SUCCESS: Received ACK packet.
ERROR: Unable to receive ACK!
ERROR: Timeout when receiving ACK.
STATE: ESTABLISHED
INFO: DATA length of 429 packets left to be sent...
SUCCESS: Sent DATA packet (73)...
type: 2 73seqnum: 46584 checksum(received): 46584       checksum(calculated):
SUCCESS: Received ACK packet.
ERROR: Unable to receive ACK!
ERROR: Timeout when receiving ACK.
STATE: ESTABLISHED
INFO: DATA length of 429 packets left to be sent...
ERROR: Max attempts are reached.
STATE: CLOSED
gbn_send: Success

Process finished with exit code 255
```

Ouput from `receiver`:

```
$ ./receiver 9999 output.txt
FUNCTION: gbn_socket()... Create socket.... socket_descriptor: 4
FUNCTION: gbn_bind() 4...
FUNCTION: gbn_listen() 4...
FUNCTION: gbn_accept() 4...
STATE: CLOSED
SUCCESS: Received SYN
SUCCESS: Received a valid SYN packet
STATE: SYN_RCVD
SUCCESS: Sent SYNACK.
SUCCESS: Accepted a valid ACK packet.
STATE: ESTABLISHED.
FUNCTION: gbn_accept returns 4.
FUNCTION: gbn_recv()
INFO: keep reading data until no more new data to be received.
SUCCESS: Received a packet.
SUCCESS: Receiving a valid DATA packet
SUCCESS: DATA packet has the correct sequence number.
SUCCESS: Sent duplicate ACK packet.
FUNCTION: gbn_recv()
INFO: keep reading data until no more new data to be received.
SUCCESS: Received a packet.
INFO: keep reading data until no more new data to be received.
ERROR: Unable to receive a packet.
INFO: keep reading data until no more new data to be received.
SUCCESS: Received a packet.
SUCCESS: Receiving a valid DATA packet
INFO: DATA packet has the incorrect sequence number.
SUCCESS: Sent duplicate ACK packet.
INFO: keep reading data until no more new data to be received.
SUCCESS: Received a packet.
SUCCESS: Receiving a valid DATA packet
INFO: DATA packet has the incorrect sequence number.
SUCCESS: Sent duplicate ACK packet.
INFO: keep reading data until no more new data to be received.
SUCCESS: Received a packet.
SUCCESS: Receiving a valid DATA packet
INFO: DATA packet has the incorrect sequence number.
SUCCESS: Sent duplicate ACK packet.
INFO: keep reading data until no more new data to be received.
SUCCESS: Received a packet.
SUCCESS: Receiving a valid DATA packet
INFO: DATA packet has the incorrect sequence number.
SUCCESS: Sent duplicate ACK packet.
INFO: keep reading data until no more new data to be received.
```

Reference

- The TCP/IP Guide: http://www.tcpipguide.com/free/index.htm