**CS 655 Programming Assignment 2**

**Reliable Transport Protocol**

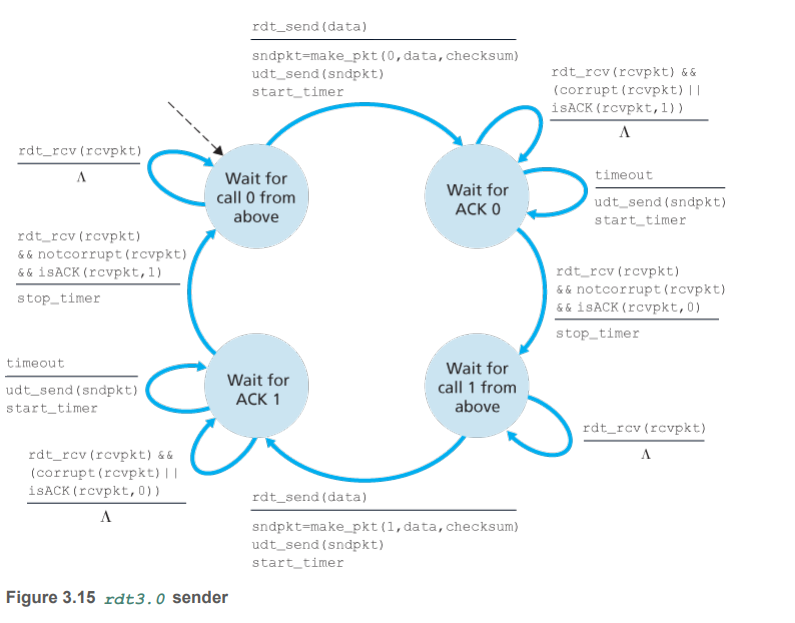
Collaboration

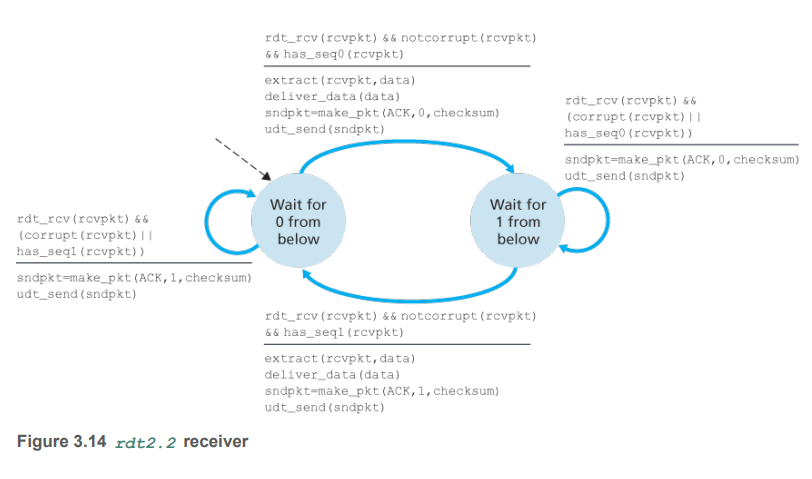
Xueyan Xia U82450191

Ziqi Tan U88387934

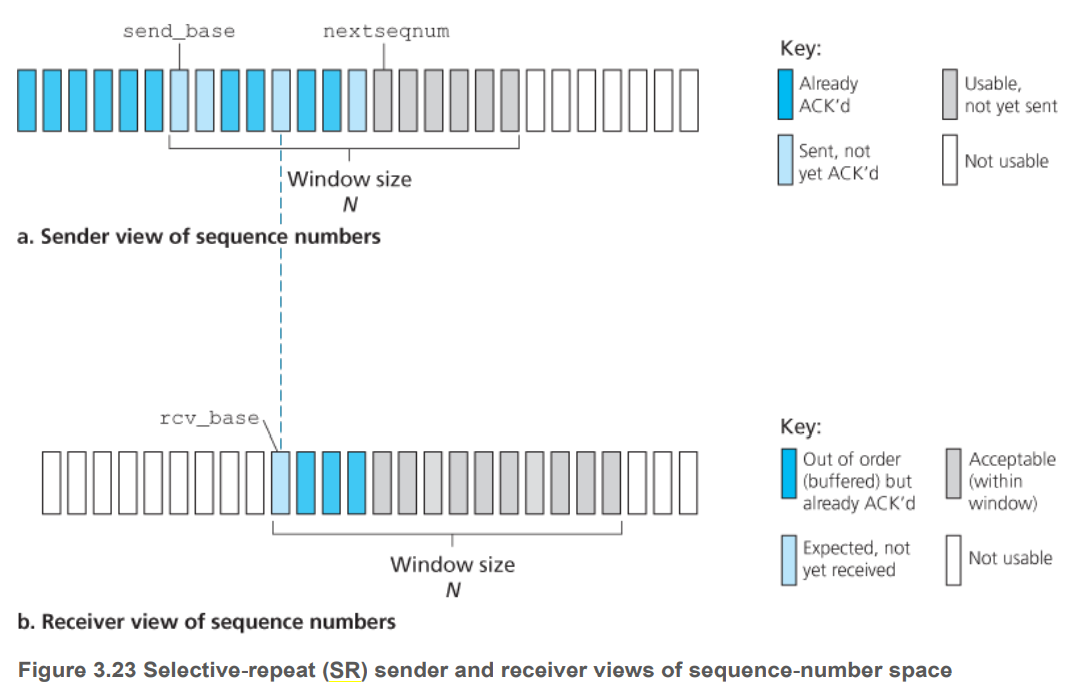
Implementation

Stop and wait





Selective Repeat with cumulative ACKs



**Instructions:**

1. Use only ACK (i.e. no NACK).
2. Buffer out-of-order packets.
3. Send cumulative ACKs.
4. The sender should retransmit only the next missing unACK’ed packet either on a timeout or duplicate ACK.

**Sender behaviors:**

1. Event: data received from above.

Action:

1. Buffer it.
2. Check the next available sequence number for the packet. If the sequence number is within the sender’s window, the data is packetized and sent.
3. Start timer.
4. Event: timeout or duplicate ACK.

Action:

1. Retransmit only the next missing (unACK’ed) packet.
2. Restart timer.
3. Event: ACK received.

Action: slide window and transmit the packets that now fall within the window. If the packet corrupts, discard it.

**Receiver behaviors:**

1. Packet with sequence number in [rcv\_base, rcv\_base+N-1] is correctly received.

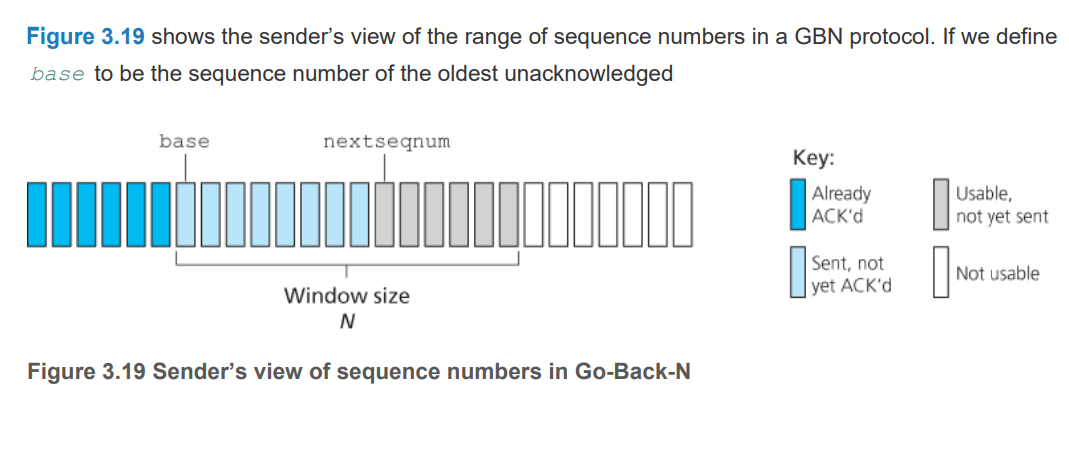
If the received packet falls within the receiver’s window, it is buffered.

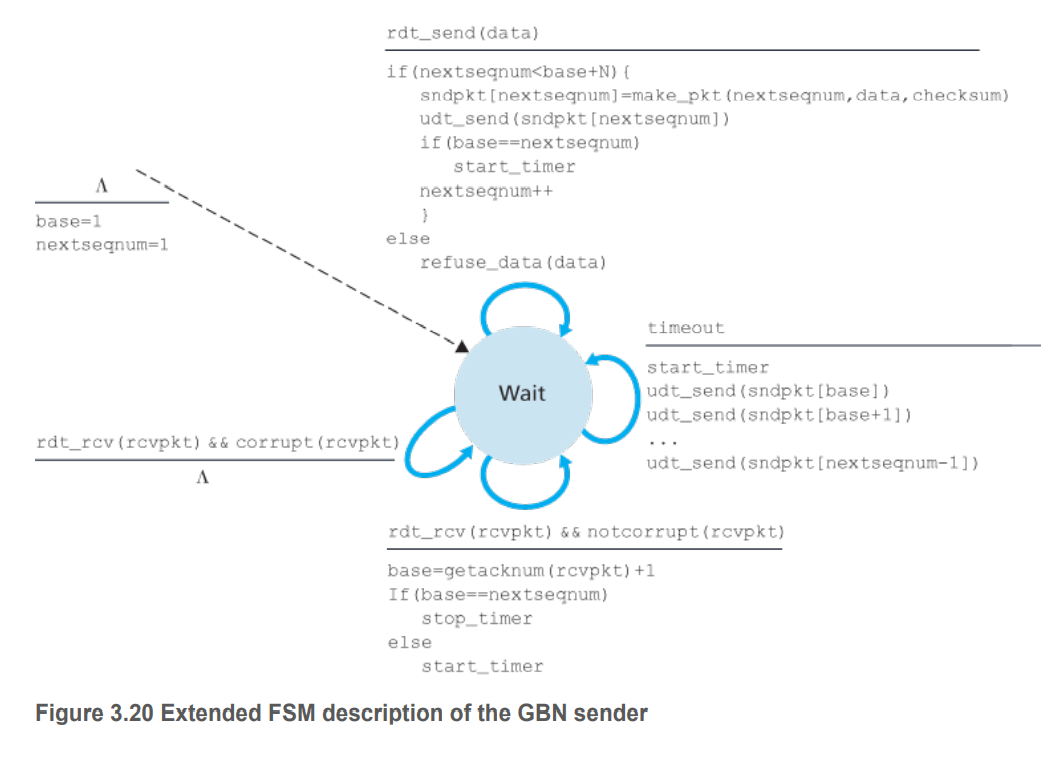
If this packet has a sequence number equal to the base of the receive window, then this packet, and any previously buffered and consecutively numbered packets are delivered to the upper layer.

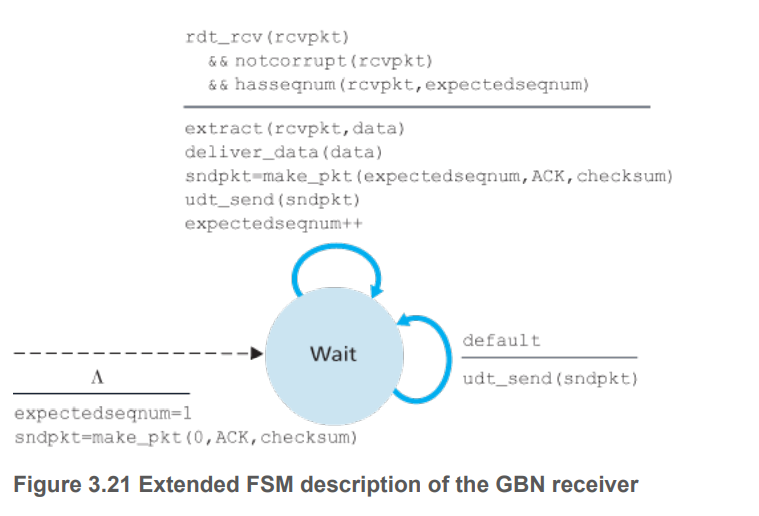
The receive window is then moved forward by the number of packets delivered to the upper layer.

1. Packet with sequence number in [rcv\_base – N, rcv\_base – 1] is correctly received. An ACK must be generated, even though this is a packet that the receiver has previously acknowledged.
2. Otherwise. Ignore the packet.

Go Back N with SACK option







Sender behavior: