Course Name: A Level (2nd Sem) Subject: DCN

Topic: Go-Back-N ARQ Date: 07-04-20

- In stop-and-wait ARQ, after each frame sent the host must wait for an ACK which results inefficient use of bandwidth. To improve efficiency ACK should be sent after multiple frames.
- To improve the efficiency, the two protocols are used that are also called **Sliding Window protocols:**
  - **Go-back-N ARQ**
  - **Selective Repeat ARQ**
- These protocols improve the **Pipelining**: A task is begun before the previous task has ended.
- In computer networking, pipelining is the method of sending multiple data units without waiting for an acknowledgment for the first frame sent.
- Pipelining ensures better utilization of network resources and also increases the speed of delivery, particularly in situations where a large number of data units make up a message to be sent.
- There is no pipelining in stop and wait ARQ because we need to wait for a frame to reach the destination and be acknowledged before the next frame can be sent.
- Pipelining improves the efficiency of the transmission.
- Sliding window protocols improve the efficiency multiple frames should be in transition while waiting for ACK. More than one frame to be outstanding.
  - Outstanding frames: frames sent but not acknowledged
  - We can send up to W frames and keep a copy of these frames (outstanding) until the ACKs arrive.
  - This procedures requires additional feature to be added: Sliding Window

Sequence Numbers

- *Sent frames are numbered sequentially*
- *Sequence number is stored in the header of the frame*
- *If the header of the frame allow m bits for the sequence number, the sequence numbers range from 0 to (2^m -1).*

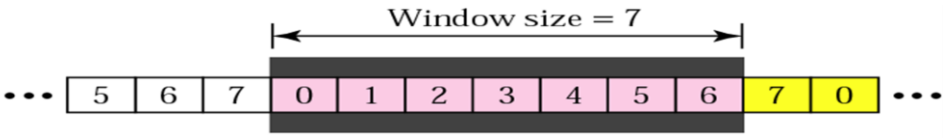
The sequence numbers are modulo 2^m, where m is the size of the sequence number field in bits.
If m = 3, sequence number range from 0 to 7(8 numbers): 0, 1, 2 , 3, 4, 5, 6, 7, 0 , 1 ,.....

Sliding window
used to hold the unacknowledged outstanding frames (frames sent but not acknowledged)

- To improve the efficiency of transmission (filling the pipe), multiple frames must be in transition while waiting for acknowledgment.

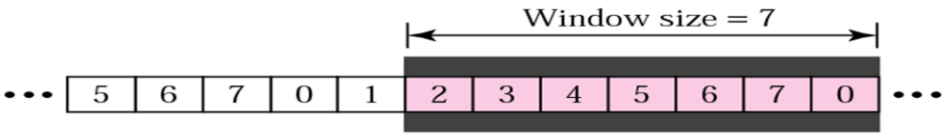
Sender sliding window:

The sender window is an abstract concept defining an imaginary box of size $2^m - 1$ *(sequence numbers –1) The sender window can slide one or more slots when a valid acknowledgment arrives.*



a. Before sliding

If m = 3; sequence numbers = 8 (0 to 7) and window size =7

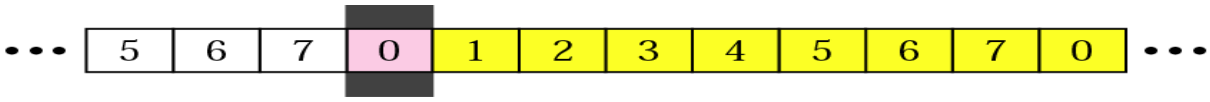


b. After sliding two frames

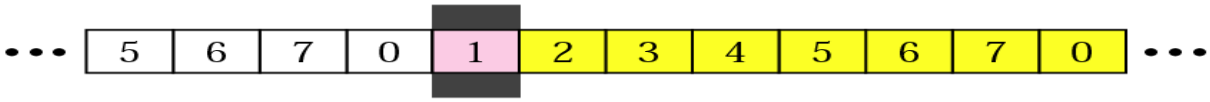
Acknowledged frames

Receiver sliding window:

- The receive window is an abstract concept defining an imaginary box of size 1 with one single variable Rn.
- The window slides when a correct frame has arrived; sliding occurs one slot at a time.

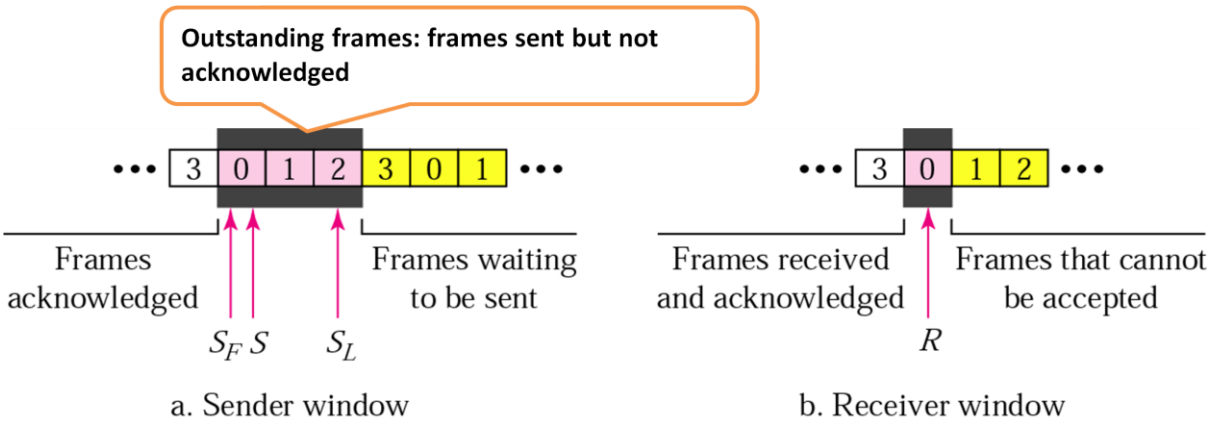


a. Before sliding



b. After sliding

Control Variables:

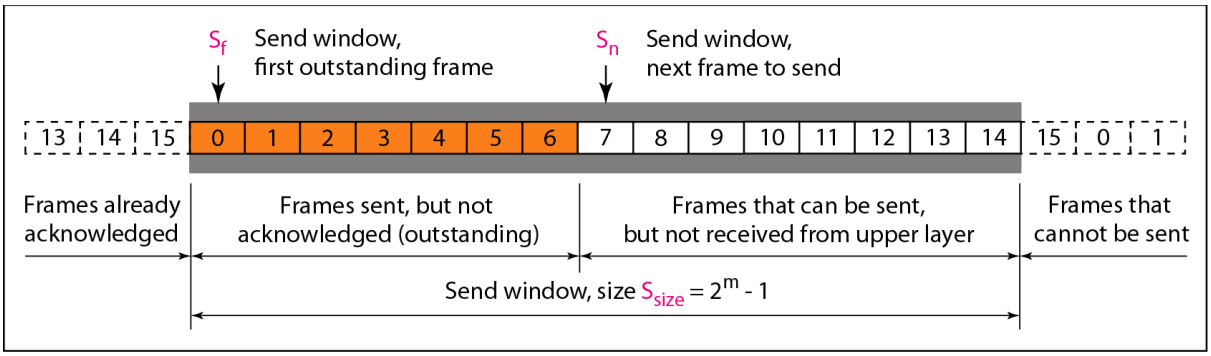


S: hold the sequence number of the recently sent frame

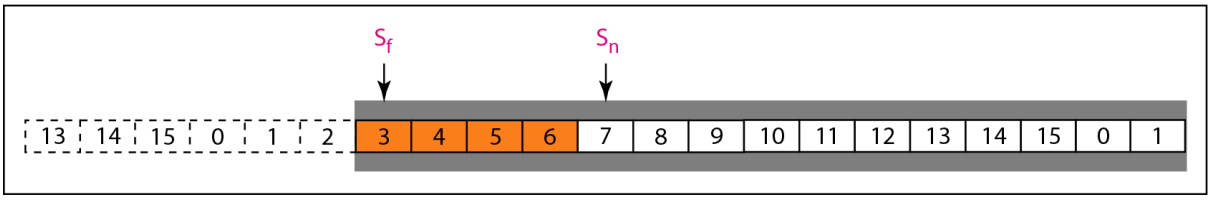
SF: holds sequence number of the first frame in the window

SL: holds the sequence number of the last frame

R: sequence number of the frame expected to received



a. Send window before sliding



b. Send window after sliding

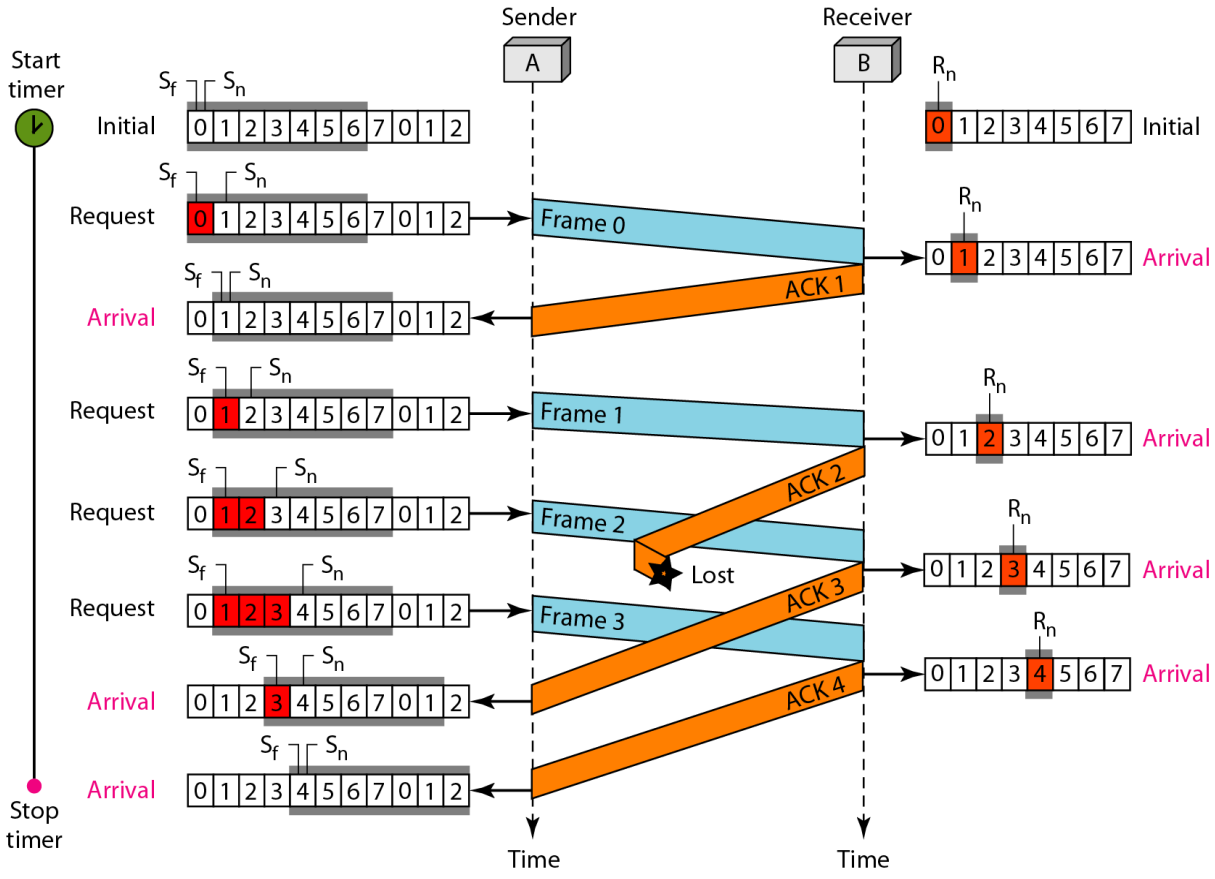
In **Go-Back-N ARQ** we use one timer for the first outstanding frame.

- The receiver sends a positive ACK if a frame has arrived safe and in order.
- If a frame is damaged or out of order, the receiver is silent and will discard all subsequent frames.

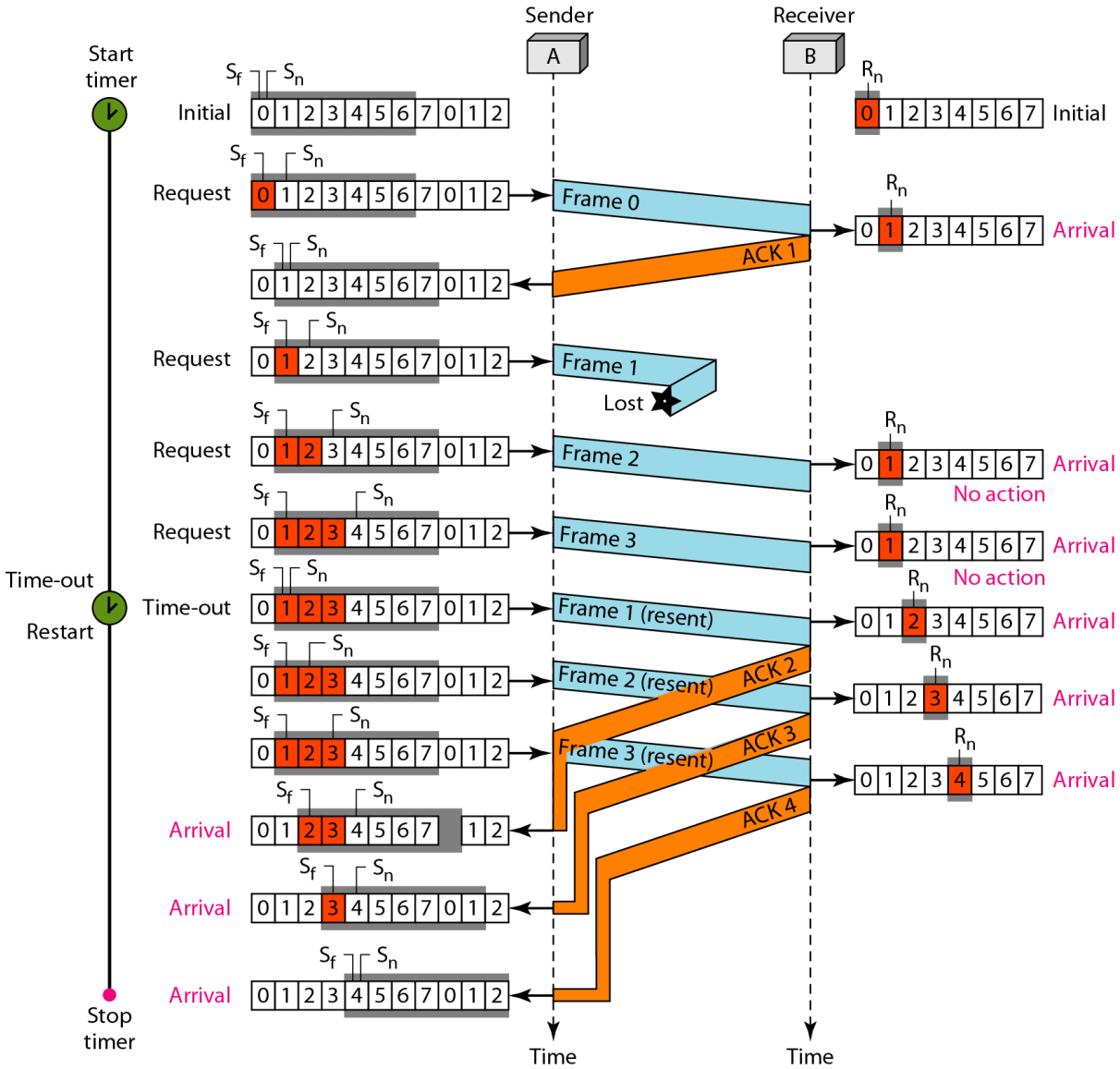
- When the timer of an unacknowledged frame at the sender site is expired, the sender goes back and resend all frames , beginning with the one with expired timer (that is why the protocol is called Go-Back-N ARQ).
- The receiver doesn't have to acknowledge each frame received. It can send cumulative Ack for several frames.
- Thus, the receive window is an abstract concept defining an imaginary box of size 1 with one single variable R_n.
  The window slides when a correct frame has arrived; sliding occurs one slot at a time.

Example of Go-Back-N ARQ:

- This is an example of a case where the forward channel is reliable, but the reverse is not. No data frames are lost, but some ACKs are delayed and one is lost. The example also shows how cumulative acknowledgments can help if acknowledgments are delayed or lost. After initialization, there are seven sender events. Request events are triggered by data from the network layer; arrival events are triggered by acknowledgments from the physical layer. There is no time-out event here because all outstanding frames are acknowledged before the timer expires. Note that although ACK 2 is lost, ACK 3 serves as both ACK 2 and ACK 3.



- What happens when a frame is lost. Frames 0, 1, 2, and 3 are sent. However, frame 1 is lost. The receiver receives frames 2 and 3, but they are discarded because they are received out of order. The sender receives no acknowledgment about frames 1, 2, or 3. Its timer finally expires. The sender sends all outstanding frames (1, 2, and 3) because it does not know what is wrong. Note that the resending of frames 1, 2, and 3 is the response to one single event. When the sender is responding to this event, it cannot accept the triggering of other events. This means that when ACK 2 arrives, the sender is still busy with sending frame 3.
- The physical layer must wait until this event is completed and the data link layer goes back to its sleeping state. We have shown a vertical line to indicate the delay. It is the same story with ACK 3; but when ACK 3 arrives, the sender is busy responding to ACK 2. It happens again when ACK 4 arrives. Note that before the second timer expires, all outstanding frames have been sent and the timer is stopped.



In Go-Back-N ARQ, the size of the sender window must be less than $2^m = (2^m - 1)$ *; the size of the receiver window is always is 1..*

Exercises:

A. **What do you understand by Sliding Window Protocol? What is Go-Back-N ARQ? How is it better than Stop and Wait ARQ?**

B. **Explain the following cases in Go-Back-N ARQ:**

   a) **Lost Data Frame**
   b) **Lost ACK Frame**
   c) **Lost ACK Frame and Lost Data Frame**