Chapter 7

DATA LINK CONTROL PROTOCOLS

Data Transmission

Lecture#9

Dr. Emad Tammam

Contents

- Flow Control.
- Error Control.
- High-Level Data Link Control (HDLC)

Data link control

- Because of the transmission errors, the receiver of data may need to regulate the rate at which data arrive.
- Synchronization and interfacing techniques are insufficient by themselves.
- It is necessary to impose a layer of control in each communicating device that provides functions such as flow control, error detection, and error control.
- This layer of control is known as a data link control protocol.

Data link control functions

- Flow control enables a receiver to regulate the flow of data from a sender so that the receiver's buffers do not overflow.
- Error control is achieved by retransmission of damaged frames that have not been acknowledged or for which the other side requests a retransmission.
- High-level data link control (HDLC) is a widely used data link control protocol. It contains virtually all of the features found in other data link control protocols.

- To achieve the necessary control, a layer of logic is added above the physical layer; this logic is referred to as **data link control** or a data link control protocol.

Requirements and objectives for effective data communication

- **Frame synchronization**: The beginning and end of each frame must be recognizable.
- **Flow control**: The sending station must not send frames at a rate faster than the receiving station can absorb them.
- **Error control**: Bit errors introduced by the retransmission system should be corrected.
- **Addressing**: On a shared link, such as a local area network (LAN), the identity of the two stations involved in a transmission must be specified.

- **Control and data on same link**: It is not desirable to have a physically separate communications path for control information. At the same time, the receiver must be able to distinguish control information from the data being transmitted.
- **Link management**: The initiation, maintenance, and termination of a sustained data exchange require a fair amount of coordination and cooperation among stations.

FLOW CONTROL

- Is a technique for assuring that a transmitting entity does not overwhelm a receiving entity with data.
- In the absence of flow control, the receiver's buffer may fill up and overflow while it is processing old data.

Vertical-time sequence diagram

- **Transmission time**: is the time it takes for a station to emit all of the bits of a frame onto the medium.
- **The propagation time**: is the time it takes for a bit to traverse the link between source and destination.
- **It is assumed that**:

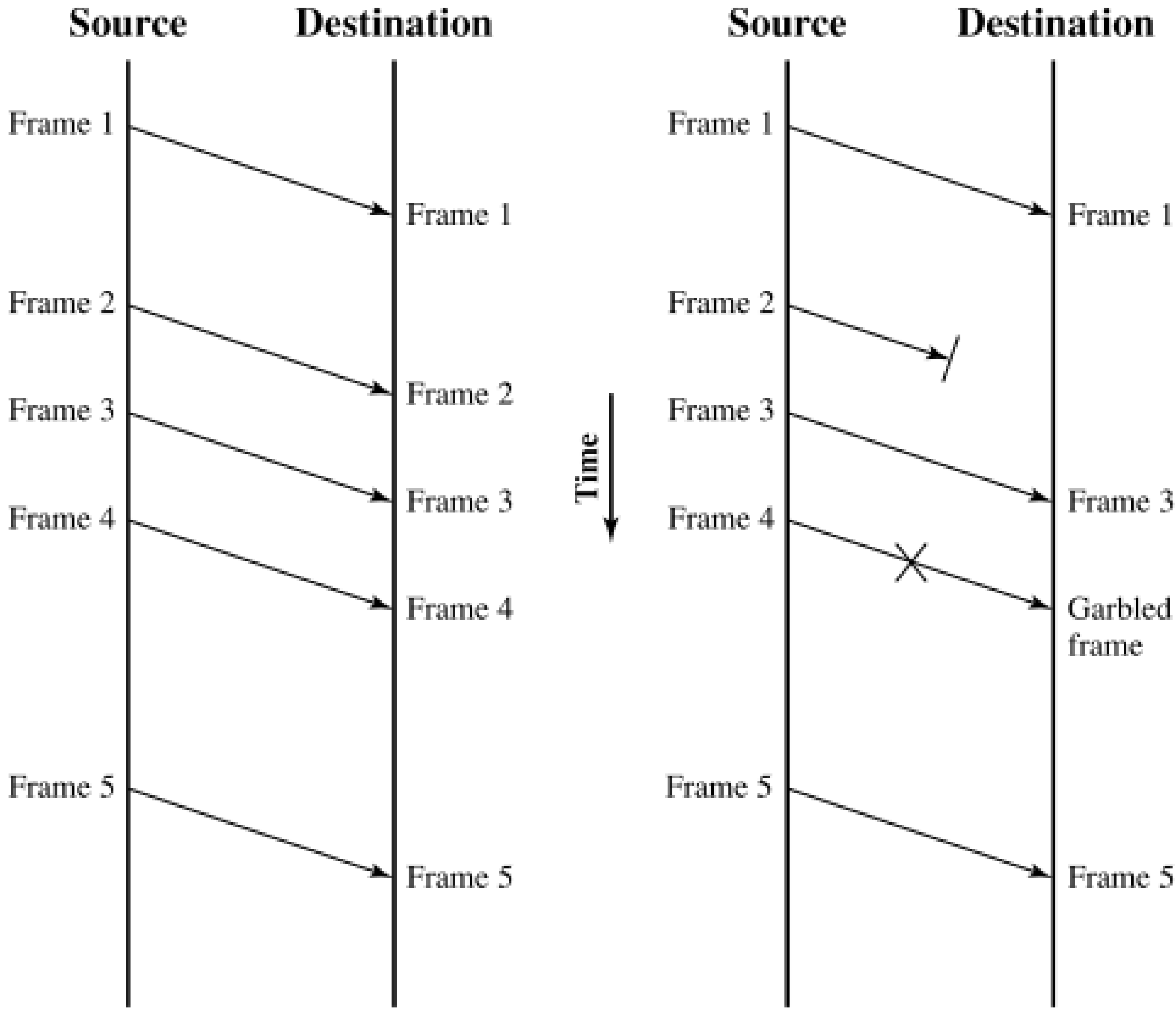
1- All the transmitted frames are successfully received.

2- No frames are lost and none arrive with errors.

3- Frames arrive in the same order in which they are sent.

4- Each transmitted frame suffers an arbitrary amount of delay before reception.

Model of Frame Transmission



(a) Error-free transmission

(b) Transmission with losses and errors

Stop-and-Wait Flow Control

- The simplest form of flow control.
- A source entity transmits a frame.
- After the destination entity receives the frame, it indicates its willingness to accept another frame by sending back an acknowledgment to the frame just received.
- The source must wait until it receives the acknowledgment before sending the next frame.
- The destination can thus stop the flow of data simply by withholding acknowledgment.

Why the large block of data is broken into frames?

- The source will break up a large block of data into smaller blocks and transmit the data in many frames for the following reasons:
  1. The buffer size of the receiver may be limited.
  2. With smaller frames, errors are detected sooner, and a smaller amount of data needs to be retransmitted.
  3. On a shared medium, such as a LAN, it is usually desirable not to permit one station to occupy the medium for an extended period.

- With the use of multiple frames for a single message, the stop-and-wait procedure may be inadequate. The essence of the problem is that only one frame at a time can be in transit.

Propagation delay

- The bit length of a link

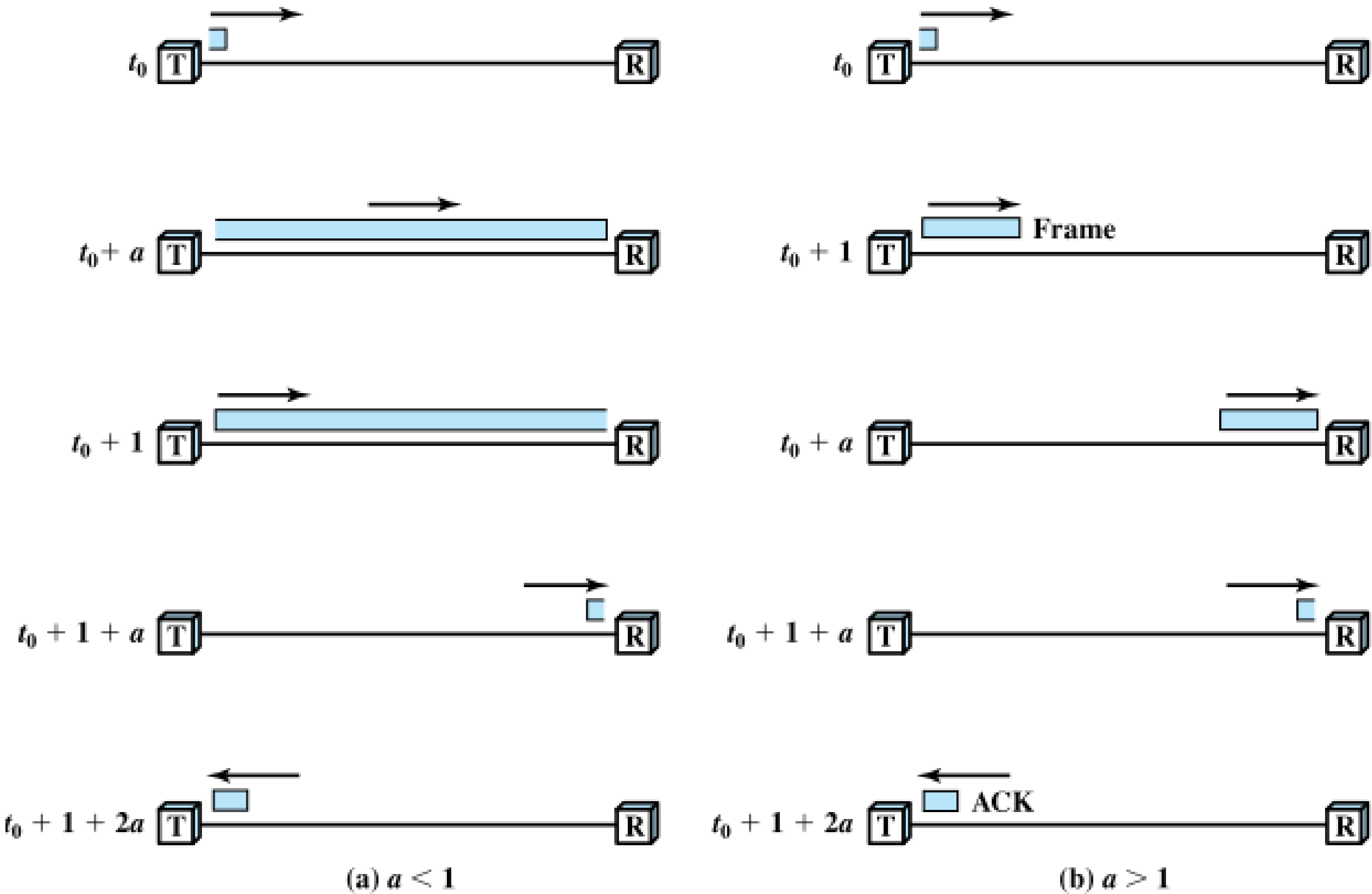
$$B = R \times \frac{d}{V}$$

B = length of the link in bits
R = data rate of the link, in bps
d = length, or distance, of the link in meters
V = velocity of propagation, in m/s

- When the bit length of the link is greater than the frame length, serious inefficiencies result.
- Propagation delay is given as: $a = \frac{B}{L}$

where L is the number of bits in the frame

Stop-and-Wait Link Utilization (transmission time = 1; propagation time = a)



- For $a > 1$, the line is always underutilized and even for $a < 1$, the line is inefficiently utilized.
- In essence, for very high data rates, for very long distances between sender and receiver, stop-and-wait flow control provides inefficient line utilization.

EXAMPLE 7.1 Consider a 200-m optical fiber link operating at 1 Gbps. The velocity of propagation of optical fiber is typically about 2×10^8 m/s. Using Equation (7.1), $B = (10^9 \times 200)/(2 \times 10^8) = 1000$ bits. Assume a frame of 1000 octets, or 8000 bits, is transmitted. Using Equation (7.2), $a = (1000/8000) = 0.125$. Using Figure 7.2a as a guide, assume transmission starts at time $t = 0$. After 1 μs (a normalized time of 0.125 frame times), the leading edge (first bit) of the frame has reached R, and the first 1000 bits of the frame are spread out across the link. At time $t = 8\ \mu$s, the trailing edge (final bit) of the frame has just been emitted by T, and the final 1000 bits of the frame are spread out across the link. At $t = 9\ \mu$s, the final bit of the frame arrives at R. R now sends back an ACK frame. If we assume the frame transmission time is negligible (very small ACK frame) and that the ACK is sent immediately, the ACK arrives at T at $t = 10\ \mu$s. At this point, T can begin transmitting a new frame. The actual transmission time for the frame was 8 μs, but the total time to transmit the first frame and receive and ACK is 10 μs.

Now consider a 1-Mbps link between two ground stations that communicate via a satellite relay. A geosynchronous satellite has an altitude of roughly 36,000 km. Then $B = (10^6 \times 2 \times 36{,}000{,}000)/(3 \times 10^8) = 240{,}000$ bits. For a frame length of 8000 bits, $a = (240000/8000) = 30$. Using Figure 7.2b as a guide, we can work through the same steps as before. In this case, it takes 240 ms for the leading edge of the frame to arrive and an additional 8 ms for the entire frame to arrive. The ACK arrives back at T at $t = 488$ ms. The actual transmission time for the first frame was 8 ms, but the total time to transmit the first frame and receive an ACK is 488 ms.

Sliding-Window Flow Control

- In situations where the bit length of the link is greater than the frame length serious inefficiencies result.

- Efficiency can be greatly improved by allowing multiple frames to be in transit at the same time.

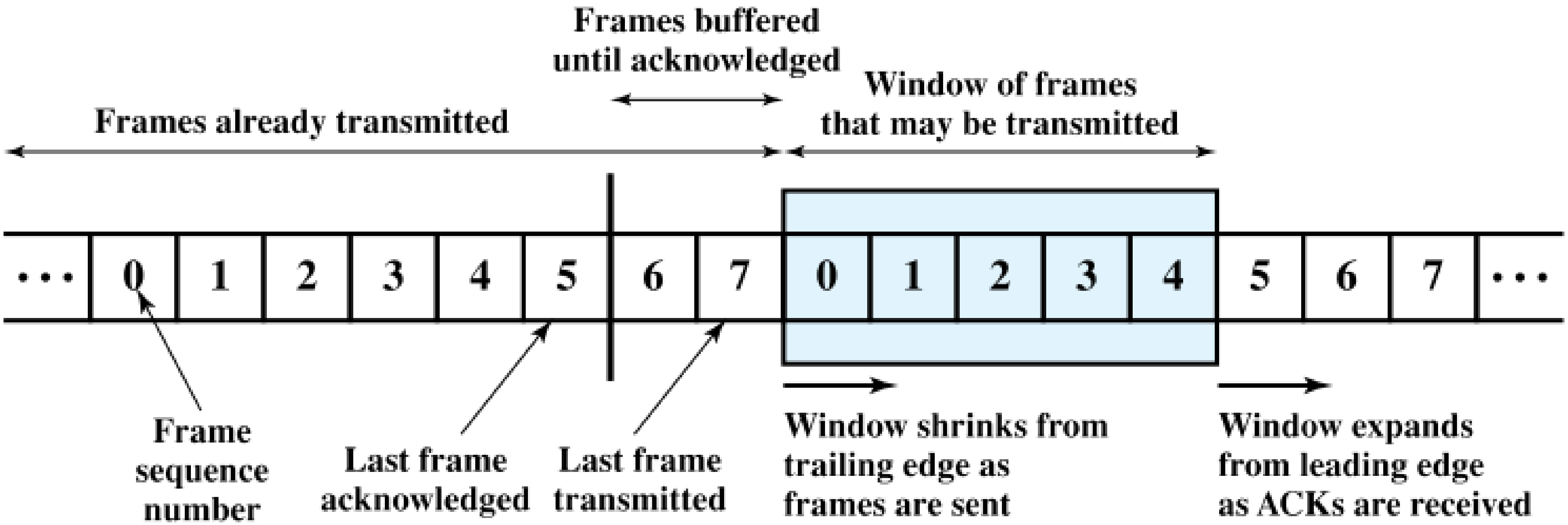
Sliding-Window Flow Control, cont.

- Two stations, A and B, connected via a full-duplex link.
- Station B allocates buffer space for W frames. Thus, B can accept W frames.
- A is allowed to send W frames without waiting for any acknowledgments.
- Each frame is labeled with a sequence number.
- B acknowledges a frame by sending an acknowledgment that includes the sequence number of the next frame expected.
- This acknowledgment also implicitly announces that B is prepared to receive the next W frames, beginning with the number specified.

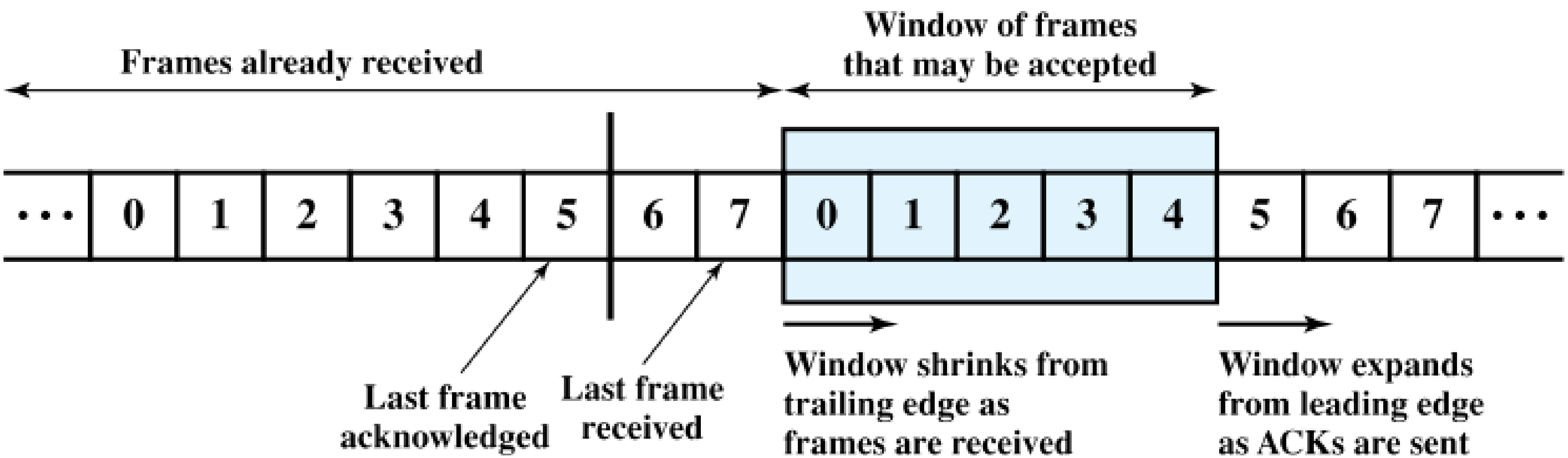
Sliding-Window Flow Control, cont.

- This scheme can also be used to acknowledge multiple frames.
- B could receive frames 2, 3, and 4 but withhold acknowledgment until frame 4 has arrived. By then returning an acknowledgment with sequence number 5, B acknowledges frames 2, 3, and 4 at one time.
- Because the sequence number to be used occupies a field in the frame, it is limited to a range of values. For example, for a 3-bit field, the sequence number can range from 0 to 7.

Sliding-Window Depiction



(a) Sender's perspective

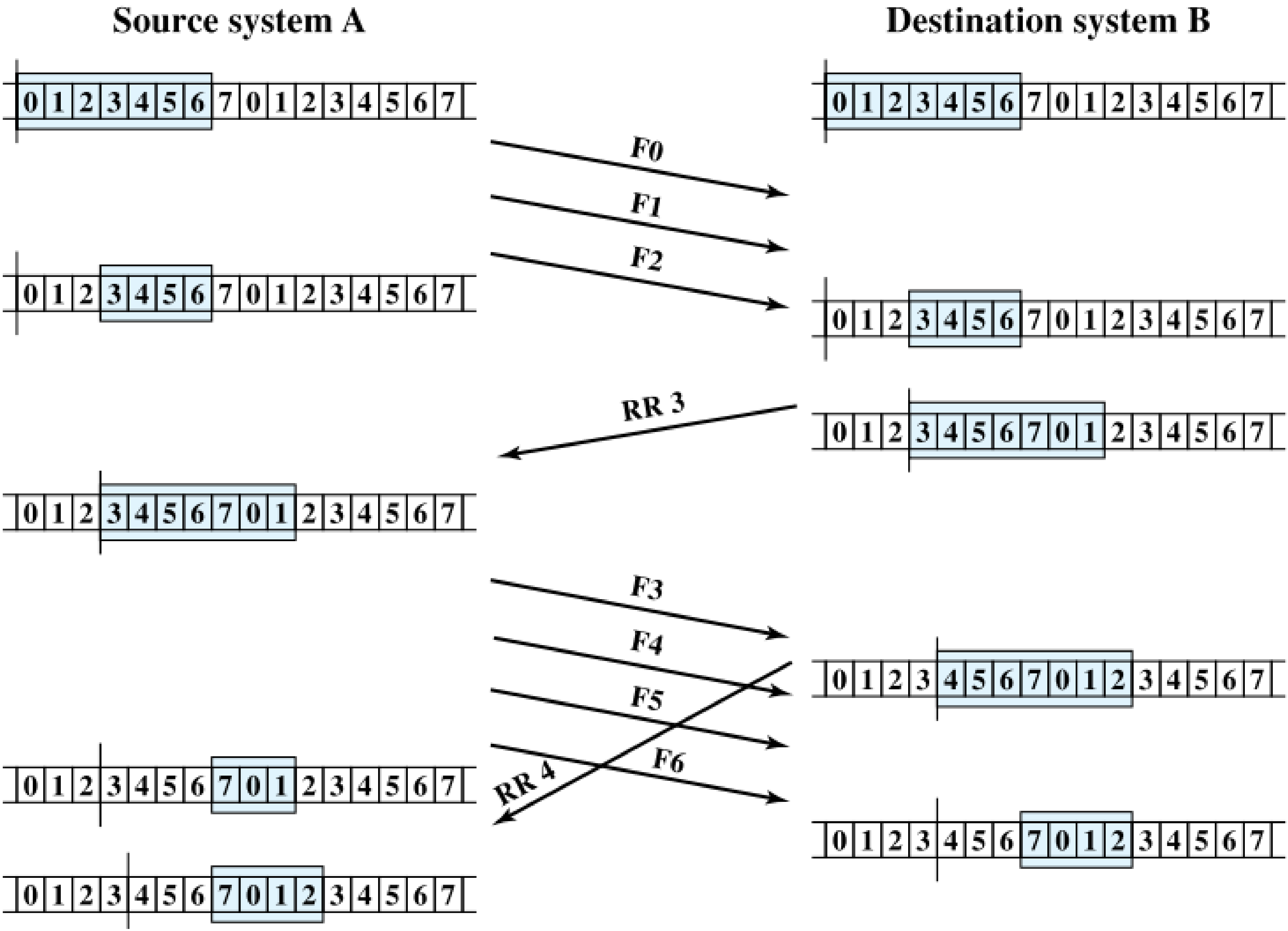


(b) Receiver's perspective

- Each time a frame is sent, the shaded window shrinks; each time an acknowledgment is received, the shaded window grows.
- The window size need not be the maximum possible size for a given sequence number length. For example, using a 3-bit sequence number, a window size of 5 could be configured for the stations using the sliding-window flow control protocol.

EXAMPLE 7.2 An example is shown in Figure 7.4. The example assumes a 3-bit sequence number field and a maximum window size of seven frames. Initially, A and B have windows indicating that A may transmit seven frames, beginning with frame 0 (F0). After transmitting three frames (F0, F1, F2) without acknowledgment, A has shrunk its window to four frames and maintains a copy of the three transmitted frames. The window indicates that A may transmit four frames, beginning with frame number 3. B then transmits an RR (receive ready) 3, which means "I have received all frames up through frame number 2 and am ready to receive frame number 3; in fact, I am prepared to receive seven frames, beginning with frame number 3." With this acknowledgment, A is back up to permission to transmit seven frames, still beginning with frame 3; also A may discard the buffered frames that have now been acknowledged. A proceeds to transmit frames 3, 4, 5, and 6. B returns RR 4, which acknowledges F3, and allows transmission of F4 through the next instance of F2. By the time this RR reaches A, it has already transmitted F4, F5, and F6, and therefore A may only open its window to permit sending four frames beginning with F7.

Example of a Sliding-Window Protocol



Forms of acknowledgments

- Receive Ready (RR): acknowledges the previous frame and ready to receive the following.
- Receive Not Ready (RNR): acknowledges the previous frame and Not ready to receive the following.
- If a station has an acknowledgment but no data to send, it sends a separate acknowledgment frame, such as RR or RNR.
- If a station has data to send but no new acknowledgment to send, it must repeat the last acknowledgment sequence number that it sent. When a station receives a duplicate acknowledgment, it simply ignores it.

- If two stations exchange data, each needs to maintain two windows, one for transmit and one for receive.
- Using the feature of **piggybacking** while Each data frame includes a field for the sequence number plus a field for the sequence number of the acknowledgment.
- If a station has an acknowledgment but no data to send, it sends a separate acknowledgment frame.
- If a station has data to send but no new acknowledgment to send, it must repeat the last acknowledgment sequence number.

Comment

- Sliding-window flow control is potentially much more efficient than stop-and-wait flow control.

- The reason is that, with sliding-window flow control, the transmission link is treated as a pipeline that may be filled with frames in transit.

ERROR CONTROL

- Refers to mechanisms to detect and correct errors that occur in the transmission of frames.

- **Two types of errors:**
- **Lost frame:** the receiver is not aware that a frame has been transmitted, e.g. noise burst effect.

- **Damaged frame:** A frame is recognized, but some of the bits are in error

Techniques for error control

- **The most common techniques for error control are based on:**

- **Error detection**.

- **Positive acknowledgment**: error-free frames.

- **Retransmission after timeout**.

- **Negative acknowledgment and retransmission**: an error is detected and the source retransmits the frames.

Automatic repeat request (ARQ)

- Is the mechanism of error control.

- The effect of ARQ is **to turn an unreliable data link into a reliable one.**

- **Three versions of ARQ** have been standardized:
- **Stop-and-wait ARQ**
- **Go-back-N ARQ**
- **Selective-reject ARQ**

Stop-and-Wait ARQ

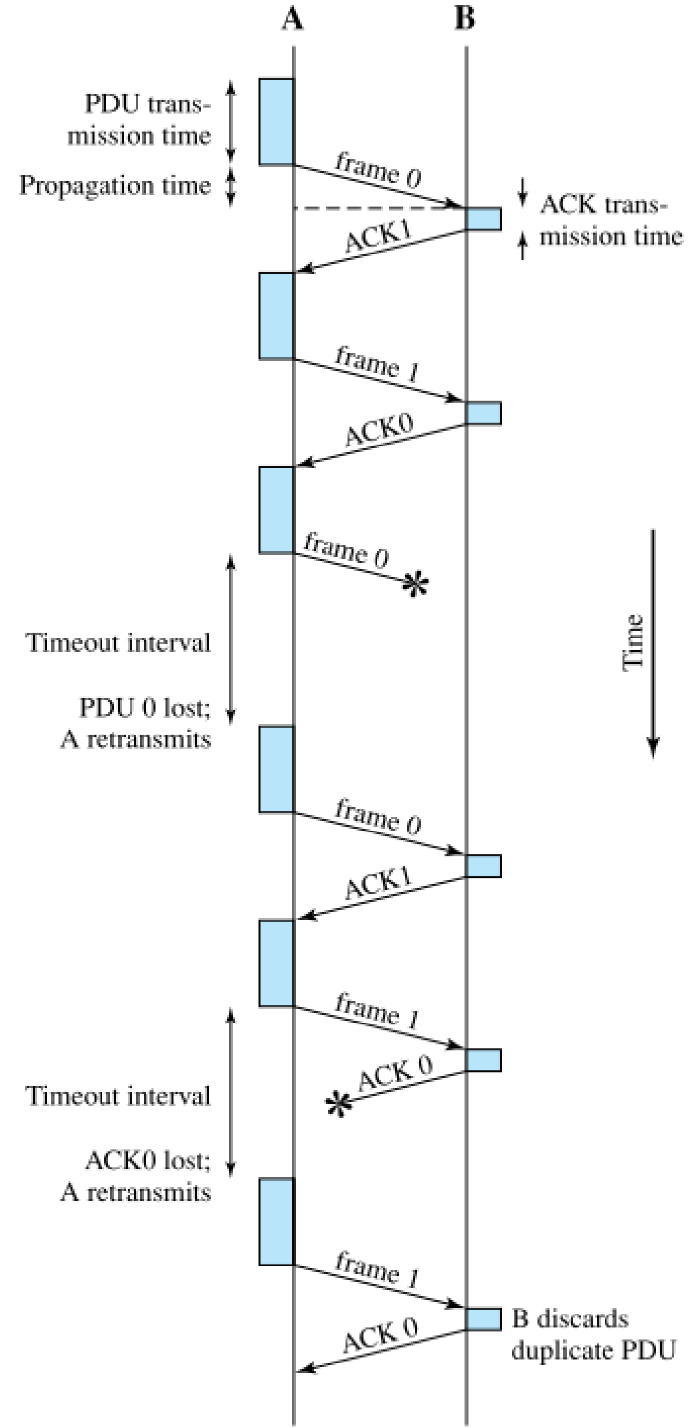
- The source transmits a single frame and must await an acknowledgment (ACK).

- **Two sorts of errors could occur**
- **First,** the frame that arrives could be **damaged**.
- If no acknowledgment is received by the time that the timer expires, the same frame is sent again.
- A copy of the transmitted frame has to be maintained at the transmitter until an acknowledgment is received.

Stop-and-Wait ARQ, cont.

- **The second**, is a **damaged acknowledgment**.
- The transmitter will **time out and resend** the same frame.
- Duplicate transmission of a frame exists.
- To avoid the duplication problem, frames are alternately labeled with 0 or 1.
- When the receiver receives two frames with the same label, it discards the second frame but sends back an ACK0 to each.
- The principal **advantage** of stop-and-wait ARQ is its **simplicity**.
- Its principal **disadvantage**, is that stop-and-wait is an **inefficient mechanism**.

Stop-and-Wait ARQ, cont.



Go-Back-N ARQ

- **based on sliding-window** flow control.
- A station may send a series of frames sequentially numbered.
- The number of unacknowledged frames outstanding is determined by window size.
- While no errors occur, (**RR=receive ready**) is transmitted.
- If the destination detects an error in a frame, (**REJ=reject**) is transmitted.
- The destination will discard that frame and all future incoming frames until the frame is correctly received.
- The source must retransmit the frame in error plus all succeeding frames.

- Suppose B successfully received frame (i-1) and A has just transmitted frame i.

- **Three cases are taken into account:**
  - **Damaged frame.**
  - **Damaged RR.**
  - **Damaged REJ.**

Damaged frame

- B discards the frame and takes no further action.
- There are two subcases:

(a) Within a reasonable period of time, A sends frame i+1, B receives frame out of order and sends a REJ i. A must retransmit frame i and all subsequent frames.

(b) A does not send additional frames. B returns neither an RR nor a REJ. When A's timer expires, it transmits an RR frame that includes a bit known as the P bit, which is set to 1. B interprets the RR frame with a P bit of 1 as a command that must be acknowledged by sending an RR indicating the next frame that it expects, which is frame i. A retransmits frame i.

Damaged RR

- There are two subcases:

(a) **B receives frame i and sends RR** which suffers an error in transit. Because acknowledgments are cumulative, **A may be receive a subsequent RR to a subsequent frame** which will arrive before the timer associated with frame i expires.

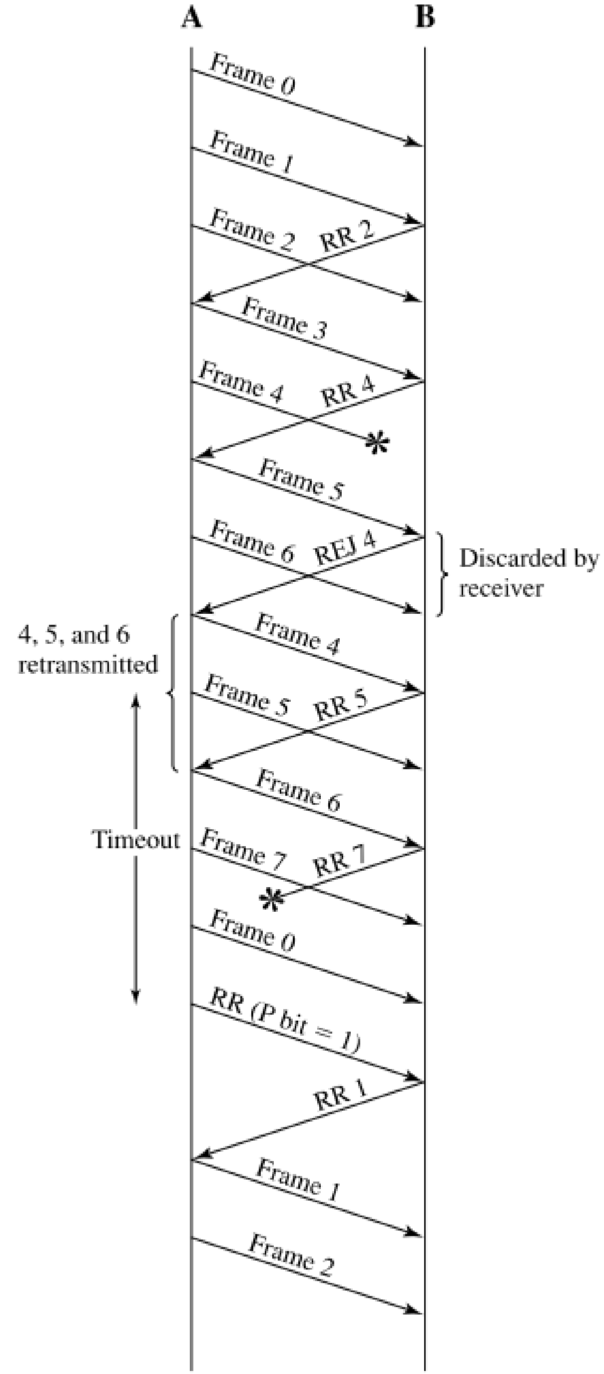
(b) **If A's timer expires**, it **transmits an RR command as in Case 1b.** It sets another timer, called the P-bit timer. If B fails to respond to the RR command, or if its response suffers an error in transit, then A's P-bit timer will expire. At this point, **A will try again by issuing a new RR command and restarting the P-bit timer.**

- This procedure is tried for a number of iterations**. If A fails to obtain an acknowledgment after some maximum number of attempts, it initiates a reset procedure**.

Damaged REJ

- If a REJ is lost, this is equivalent to Case 1b.

Go-Back-N ARQ, cont.



(a) Go-back-N ARQ

Window size limitation

- For a k-bit sequence number field, the sequence number range is 2^k .The maximum window size has to be limited to 2^k-1.
- If data are being exchanged in both directions, B must send ack. to A’s frames in the data frames transmitted by B, even if the ack. has already been sent.
- This is because B must put some number in the ack. field of its data frame.

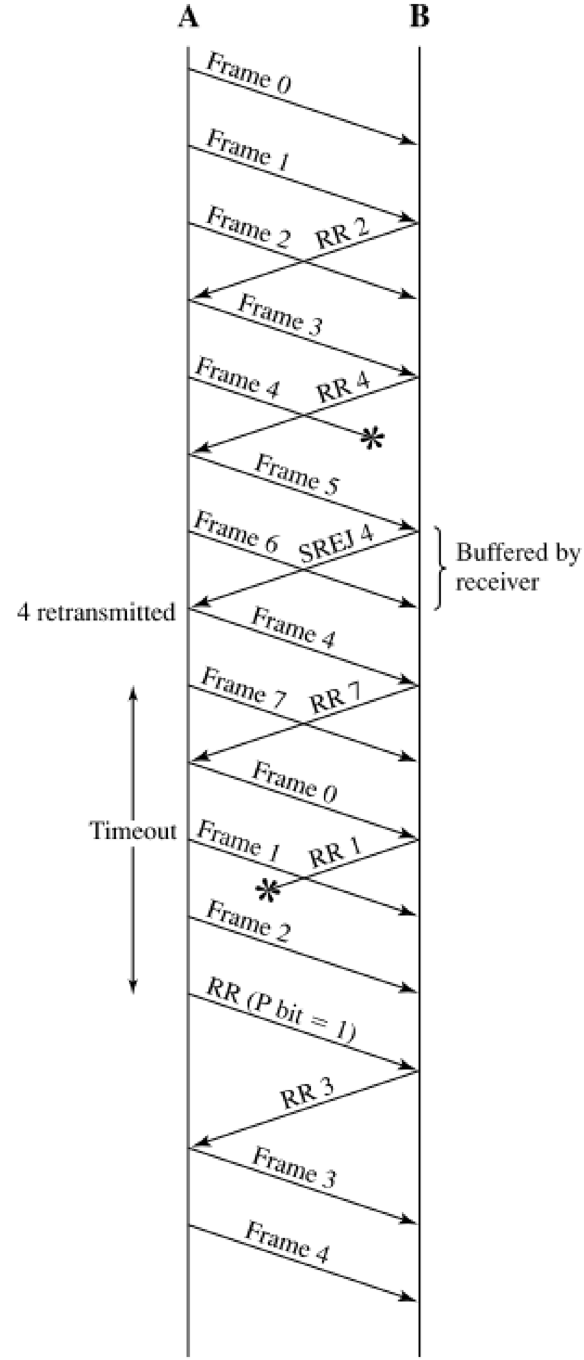
Window size limitation, cont.

- Assume a 3-bit sequence number.
- Suppose a station sends frame 0 and gets back an RR1 and then sends frames 1, 2, 3, 4, 5, 6, 7, 0 and gets another RR 1.
- This could mean that all eight frames were received correctly and the RR1 is a cumulative acknowledgment.
- It could also mean that all eight frames were damaged or lost, and the receiving station is repeating its previous RR1.
- The problem is avoided if the maximum window size is limited to 7 (2^3-1)

Selective-Reject ARQ

- The only frames retransmitted are those that receive a negative acknowledgment or those that time out.
- It more efficient than go-back-N, because it minimizes the amount of retransmission.
- The receiver must maintain a buffer large enough to save post-SREJ frames until the frame in error is retransmitted.
- The receiver must contain logic for reinserting that frame in the proper sequence.
- It requires more complex logic to be able to send a frame out of sequence.
- Because of such complications, it is much less widely used than go-back-N ARQ.
- It is a useful choice for a satellite link because of the long propagation delay.

Selective-Reject ARQ, cont.



(b) Selective-reject ARQ

Window size limitation

- The window size limitation is more restrictive for selective-reject than for go-back-N.
- Consider the case of a 3-bit sequence number size:

1. Station A sends frames 0 through 6 to station B.
2. Station B receives all seven frames and cumulatively acknowledges with RR 7.
3. Because of a noise burst, the RR 7 is lost.
4. A times out and retransmits frame 0.
5. B has already advanced its receive window to accept frames 7, 0, 1, 2, 3, 4, and 5. Thus it assumes that frame 7 has been lost and that this is a new frame 0, which it accepts.

Window size limitation, cont.

- The problem with the foregoing scenario is that there is an overlap between the sending and receiving windows.
- To overcome the problem, the maximum window size should be no more than half the range of sequence numbers.
- In the preceding scenario, if only four unacknowledged frames may be outstanding, no confusion can result.