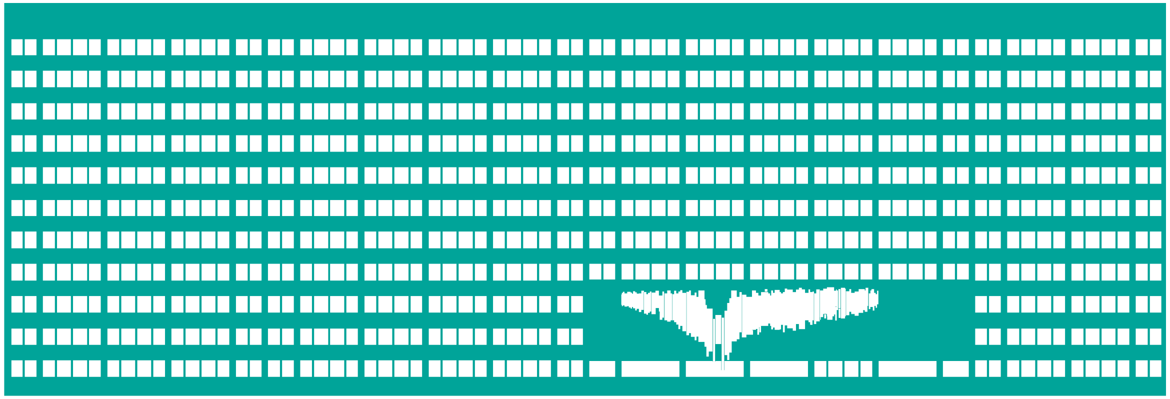
VSB TECHNICAL UNIVERSITY OF OSTRAVA | FACULTY OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE | DEPARTMENT OF COMPUTER SCIENCE

Error-Control Methods



Computer Networks
Lecture 3

Feedback Implementation Options

- Positive (ACK) and negative (NAK) acknowledgments
- Sending back the CRC of the received packet
- Sending back the whole data frame

Note that both data packets and acknowledgments may be **damaged** or **lost** in during the transfer

Packet Numbering

- Used to ensure the proper packet sequencing
- Allows to detect missing packets
- Protects against duplicated packets coming to the receiver
    - e.g. in case of the packet retransmissions caused by lost ACKs

Communication Protocol

- A set of syntactic and semantic rules for communication of two or more devices
  - Includes definition of timing, e.g. timeout handling

- May include procedures to detect/correct errors

Error Control Methods

Common Protocols to Ensure
Error-Free Communication between
Transmitter and Receiver

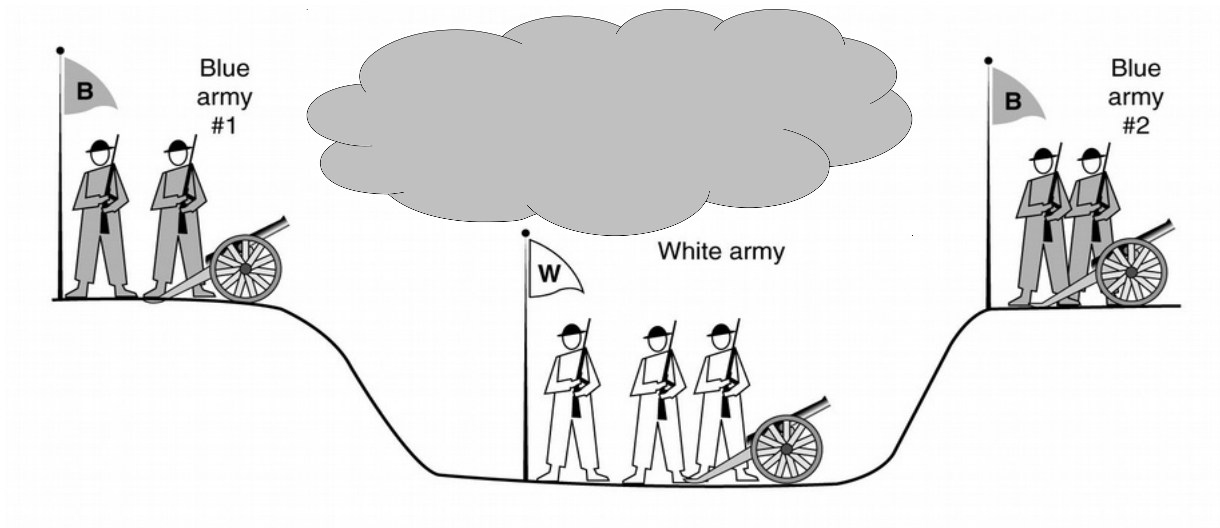
Acknowledgment Types

- **positive** (ACK) – acknowledges a correct reception
  - transmitter is blocked if ACK is lost
- **negative** (NAK) – informs about reception of erroneous packet
  - it is not sufficient by itself as a error-control method, but makes the error detection faster
- combined approach – uses both ACKs and NAKs

Retransmission Timeouts

- The problem of transmitter blocking in case of lost ACK may be solved by introduction of a waiting timeout
    - Automatic Repeat reQuest (ARQ) methods
- Complicates the formal protocol description
    - It is necessary to introduce time context
- Selection of an optimal timeout duration is a tradeoff between 2 conflicting requirements:
    - a need to detect error and retransmit lost/damaged packet as soon as possible
    - avoiding of unnecessary network load caused by early retransmissions

"Two Generals' Problem"



- State synchronization over an unreliable channel
  - may not be reached with a finite number of message exchanges
- A receiver is never sure whether its acknowledgment successfully arrived to the transmitter

Error-Control Protocol Classification (1)

- Stop-and-wait
  - Transmitter sends just one frame at a time and waits for its acknowledgment
  - If no acknowledgement comes during a specified time period, the frame is automatically repeated
    - Automatic Repeat reQuest (ARQ)
  - Very inefficient on channels with a long propagation delay

Error-Control Protocol Classification (2)

- Pipelining
  - Necessary for links with a long propagation delay
    - also in case of communication over intermediate systems, as in the Internet
  - A transmitter may send a group of frames without waiting for individual frames' ACKs
  - 100% efficiency may be reached
    - on full-duplex links
  - ACKs are commonly treated as inclusive
    - i.e. acknowledge all packets up to the specified sequence number (note the integer overflow issues)
    - Protects against the situation when some ACKs are lost
    - Saves the bandwidth by limiting the number of ACKs
    - There is a possibility to delay ACK for some time and combine ACKs of multiple data frames that arrived during the delay interval

Sliding Window Protocols

Sliding Window – Basic Principle (1)

- Transmitter may send multiple frames without waiting for an ACK
  - The maximum number is given by the sending window size

- A separate timeout timer is started after transmission of every frame
  - a frame is maintained in a sending window until it is acknowledged
  - if a timeout expires, the frame is retransmitted

Sliding Window – Basic Principle (2)

- Receiver sends ACK after each successfully received frame
  - In case of reception of an erroneous frame, receiver sends NAK or silently discards it, causing the transmitter's timeout to expire

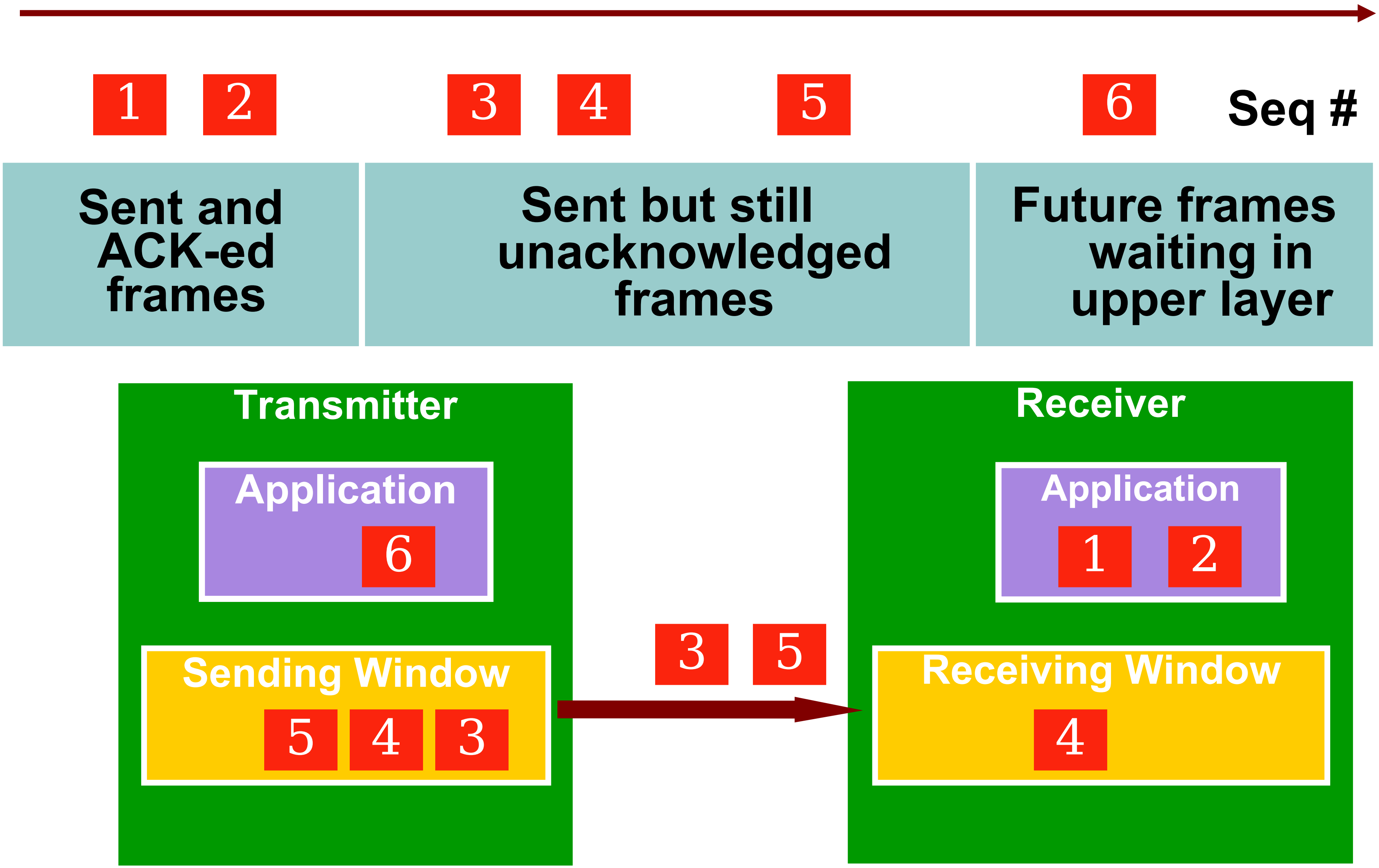
- A transmitter can discard the „oldest“ frame from the sending window after it is acknowledged
  - and shift the window, i.e accept the another frame for transmission from the upper layer

Buffers in Sliding Window

- Sending Window – maintains the frames that were transmitted but were not acknowledged yet
  - they might have been potentially lost and thus have to be retransmitted again in case of timeout expiration
- Receiving Window – maintains the received frames that could not be passed to the upper layer because previous packet(s) of the sequence are still missing

Both windows "slide" over a set of sequence numbers

Sliding Window Example



Alternative Approaches of Processing Out-of-order in Sliding Window

- Go-Back-N
- Selective Repeat

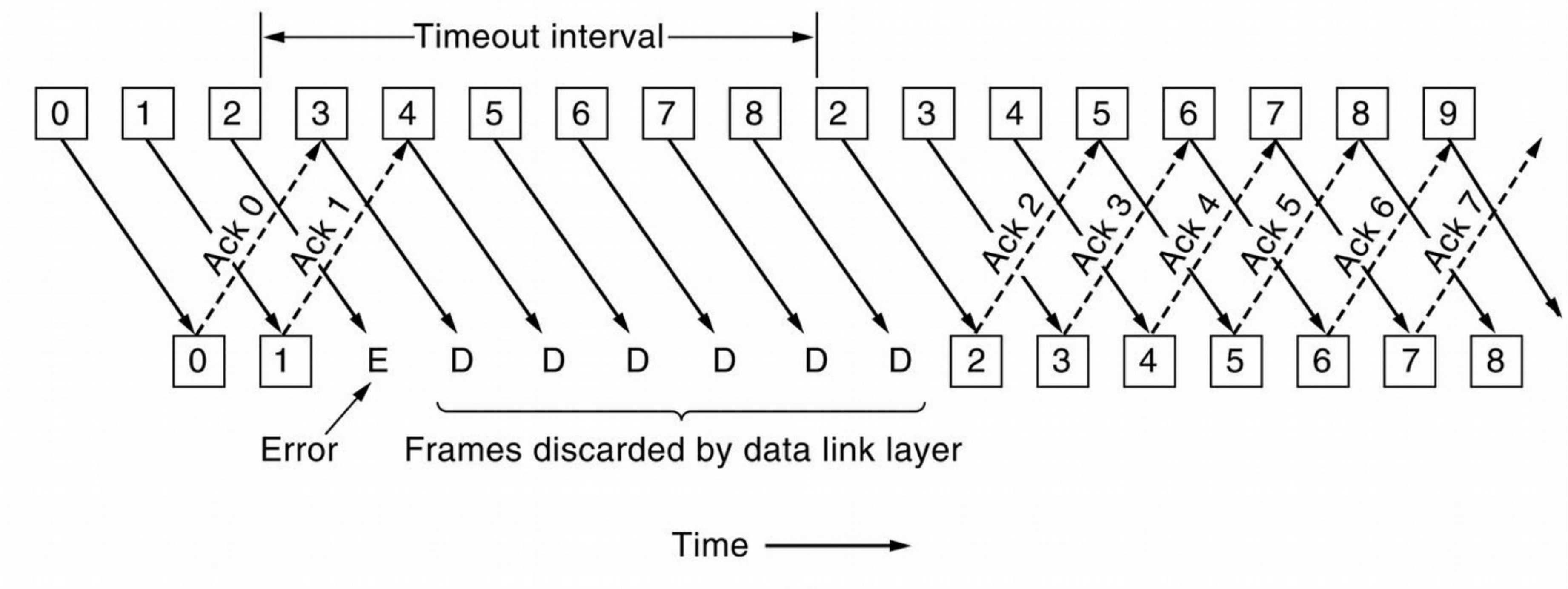
The methods differ in the way they react to the erroneous or lost frame

- The receiver may detect a lost frame when the following frame with a greater sequence number arrives

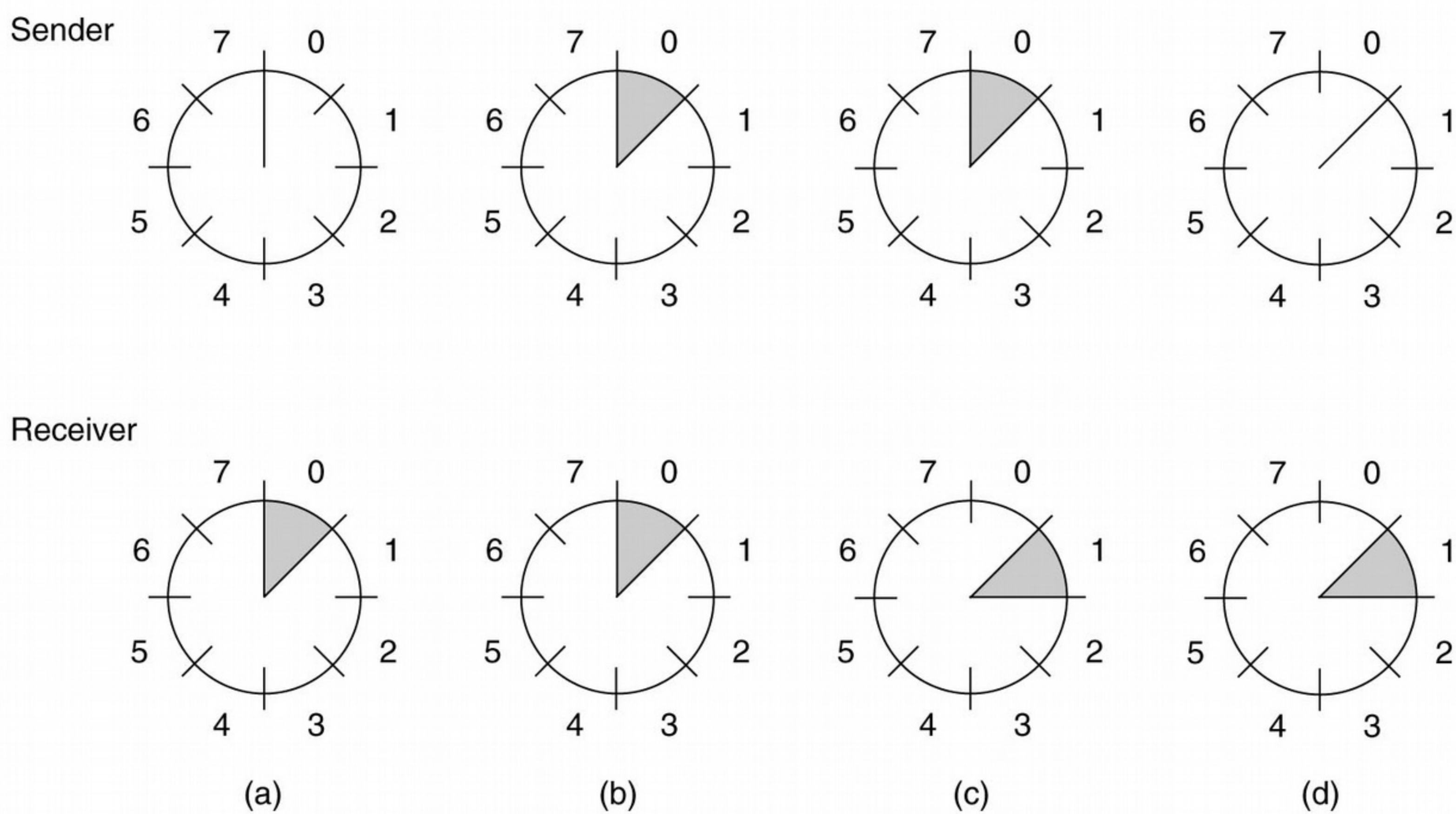
Go-Back-N

- Receiver discards all frames following the missing one
  - Receiver does not ACK such frames (it still repeats ACK for the last correctly received frame)
- The receiving window size is 1 frame
- If a frame in a sending window times out, the transmitter also retransmits **all the following frames in sending window**
  - even in case they arrived to the receiver, the receiver discarded them
- The receiver implementation is simple, but wastes the network capacity

Go-back-N: An Example



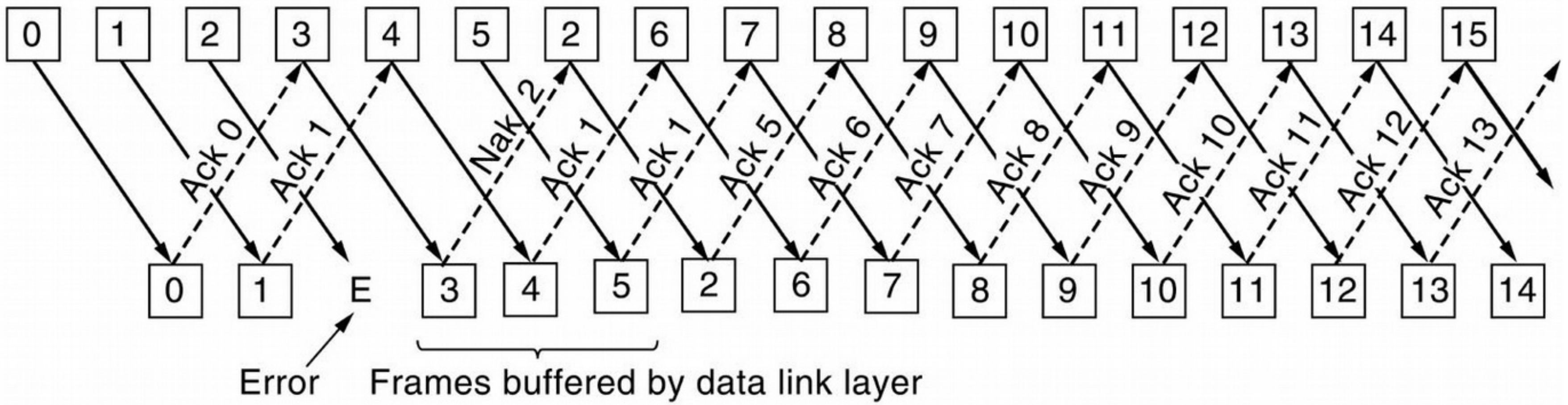
Go-back-N: Example of Usage of Sending and Receiving Window



Selective Repeat

- Receiver buffers (correct) out-of-order frames
  - If there is a frame missing in a sequence, passing of data to the upper layer is delayed until the successful reception of (the retransmitted) frame
- If a timeout expires, the sender selectively repeats just the single frame
- Receiver always acknowledges the last frame of the correctly received frame sequence
- The operation may be optimized by sending NAKs for out-of-order or erroneous frames
  - transmitter does not have to wait for timeout expiration before retransmission

Selective Repeat: An Example with NAK



Relationship between Sending Window Size and a Number of Utilized Sequence Numbers

If w is the sending window size and n is a number of sequence numbers:

- w<n for Go-back-N
  - otherwise we would not be able to detect loss of all frames in the sending window
- w <= n/2 for Selective repeat:
  - Because there is an overlap of the receiving and sending window

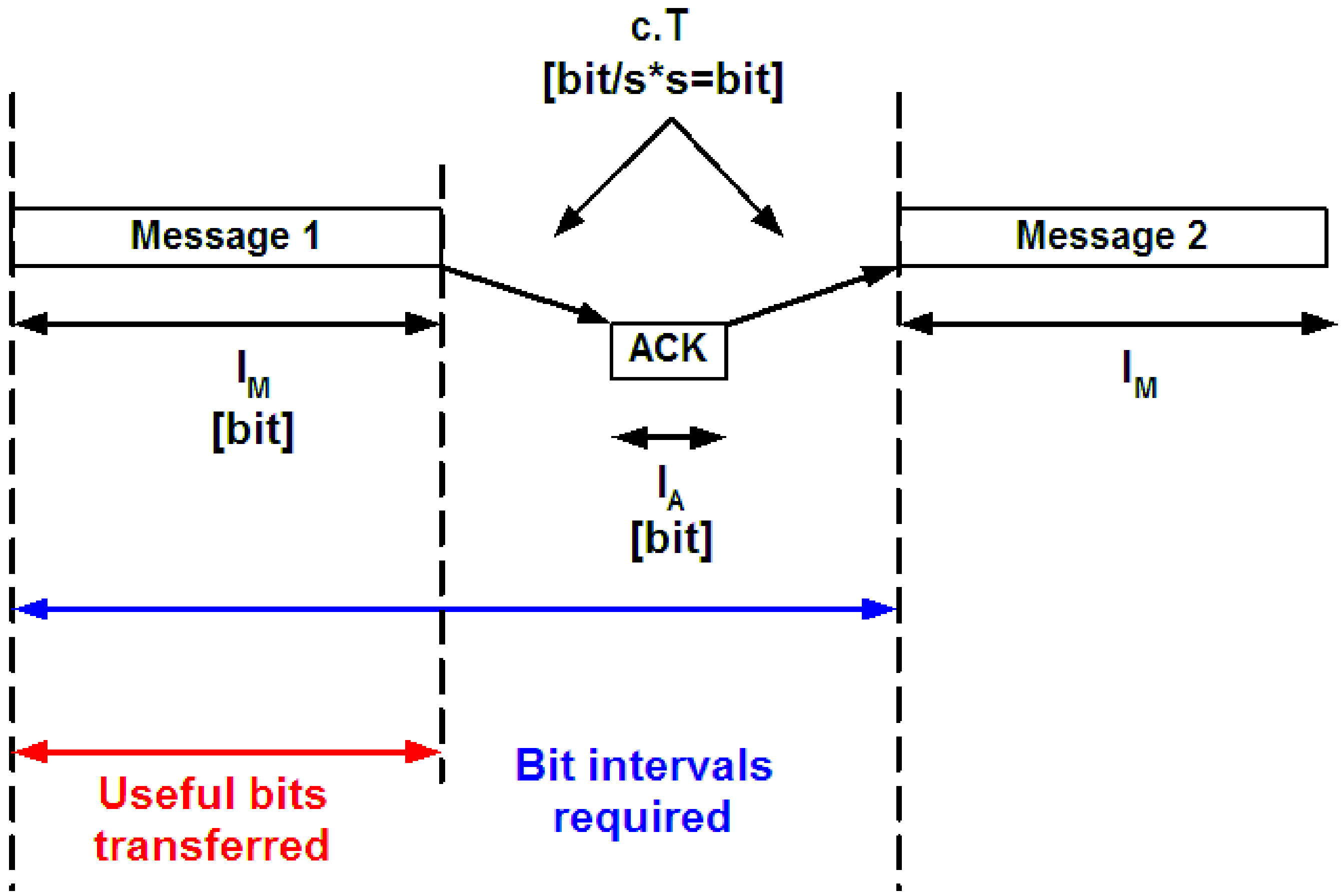
Flow Control

- It may be necessary to temporarily stop the transmitter if the receiver application does not consume data fast enough and receiving window becomes full.
  - Receiver advertises the remaining receiving window size to the transmitter
  - Transmitter dynamically adapts the sending window size accordingly

- Utilized e.g. in TCP protocol

Efficiency of Error Control Protocols

Stop and Wait



$$e_f = \frac{l_m}{l_m + cT + l_a + cT} = \frac{l_m}{l_m + l_a + 2cT}$$

Stop and Wait Efficiency Examples

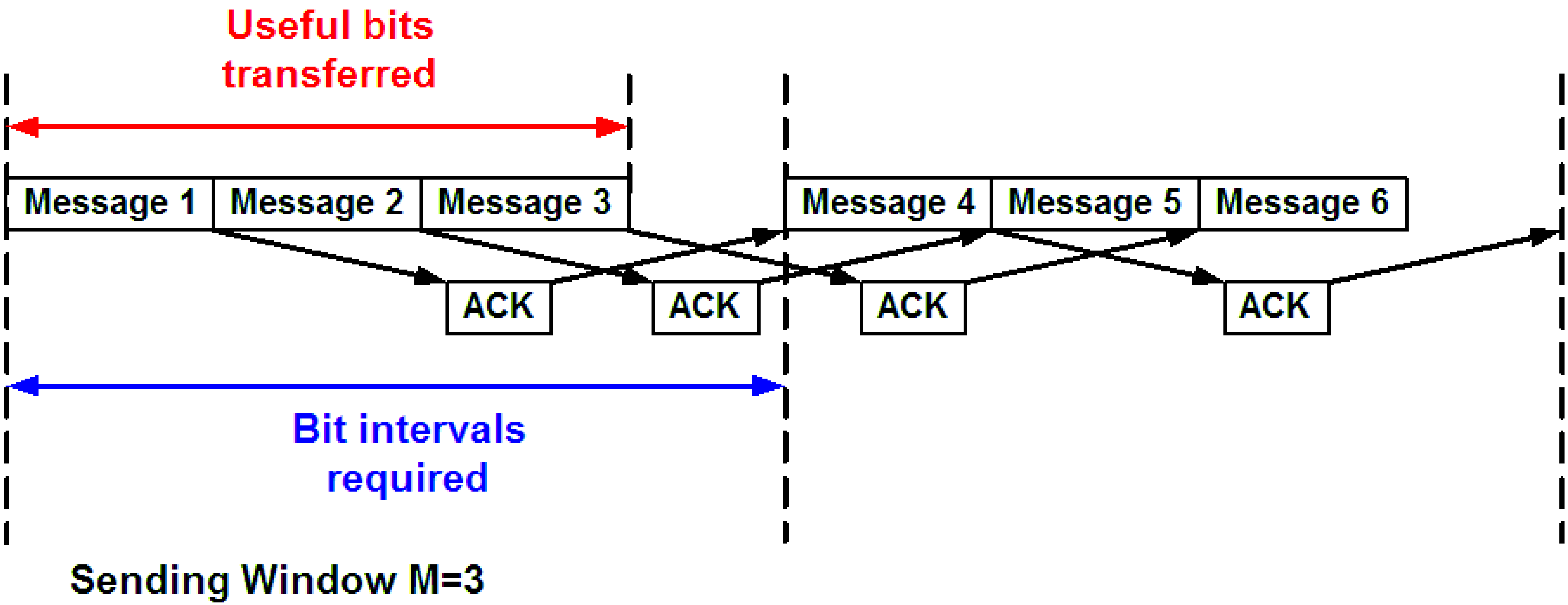
- Modem link optimized by
  - lm=80B, la=1B, c=14400 bps, T=1ms, ef=94.56%
- Satellite link
  - lm=80B, la=1B, c=14400 bps, T=270 ms, ef=7.6%

After extension of frame length to 8 × original size:

- Modem link
  - lm=640B, la=1B, c=14400 bps, T=1ms, ef=99.28%
- Satellite link
  - lm=640B, la=1B, c=14400 bps, T=270 ms, ef=40.38%

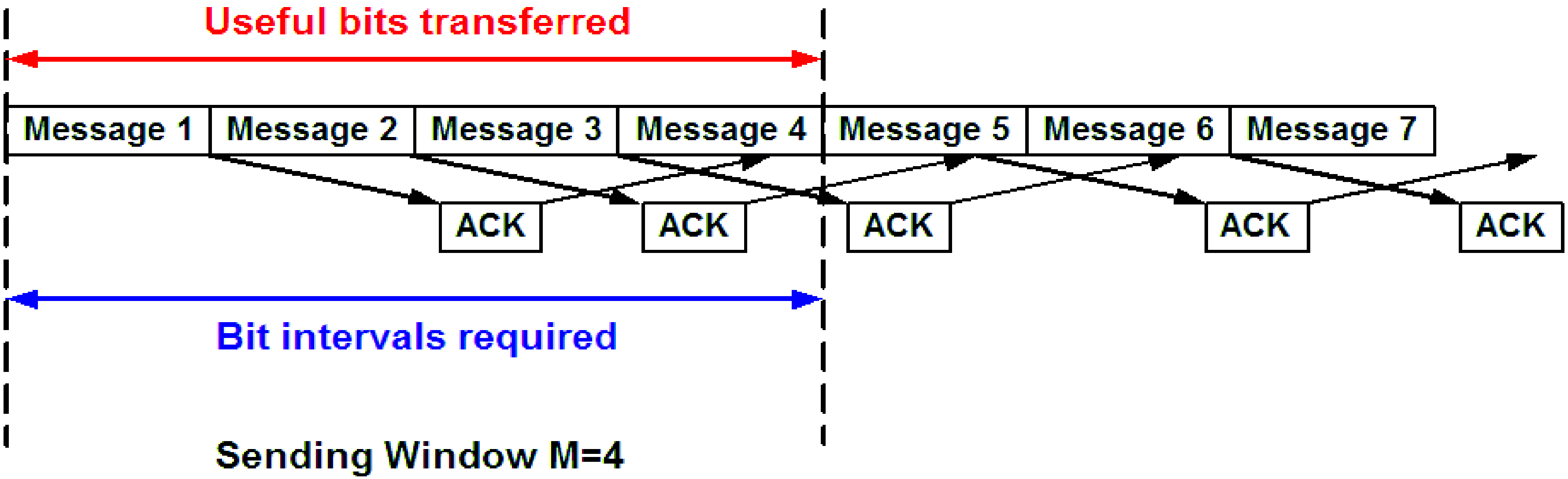
The extension of frame length improves the efficiency, but the whole (long) frame has to be discarded in case of error.

Sliding Window



$$e_f = \frac{M.l_m}{l_m + cT + l_a + cT} = \frac{M.l_m}{l_m + l_a + 2cT}$$

Reaching of 100% Efficiency using Sliding Window



Let's think about following task …

Calculate a minimal size of a sending window for a given data and ACK frame lengths and transmission channel bit rate and delay.

Problems of Communication in the Real Network

- Packets may be damaged or lost
  - ⟹ it is necessary to introduce a feedback to correct errors
- Packets may arrive out of order if there are alternate paths over the network
  - ⟹ It is needed to insert sequence numbers into packets
- Packets may be duplicated in some cases