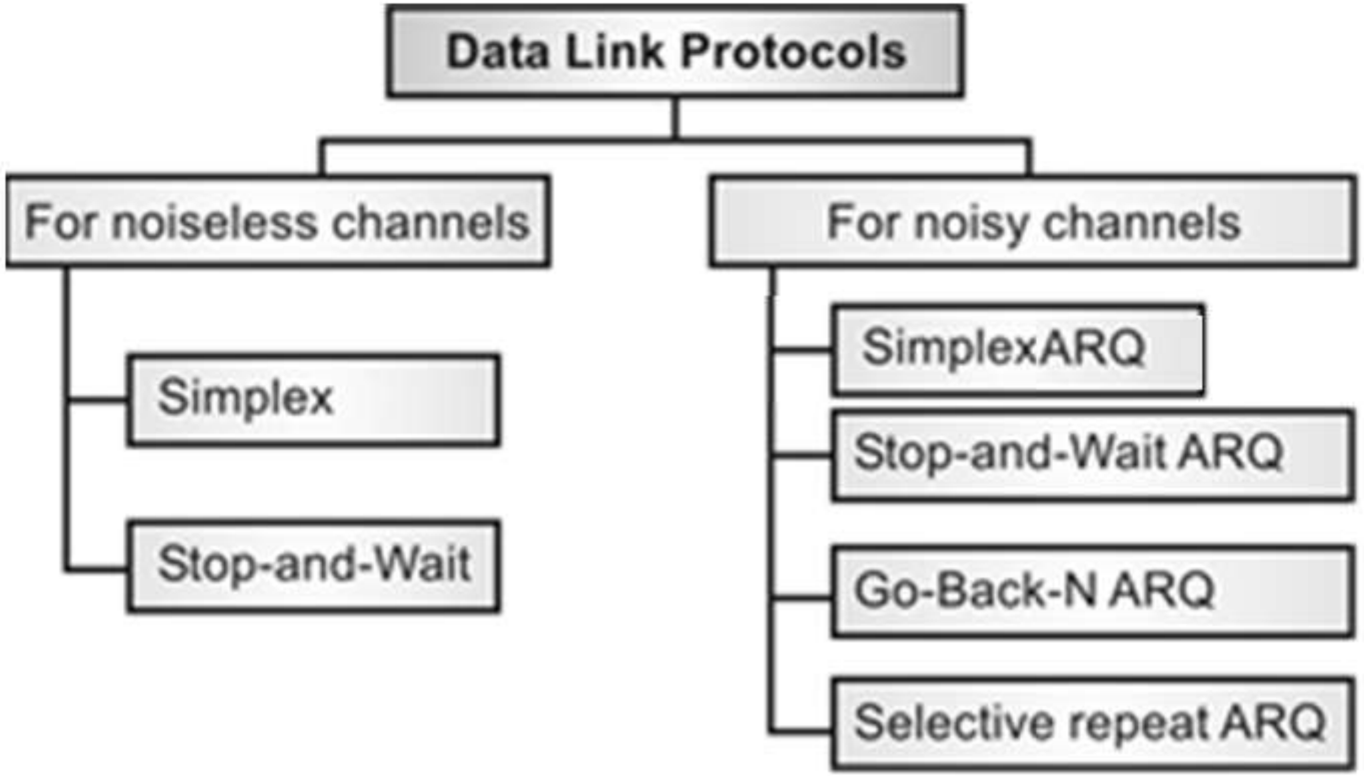
Computer Network(CSC 503)

Shilpa Ingoley

Lecture 18

Types of Protocols



Sliding Window protocol

Three types of sliding window protocol

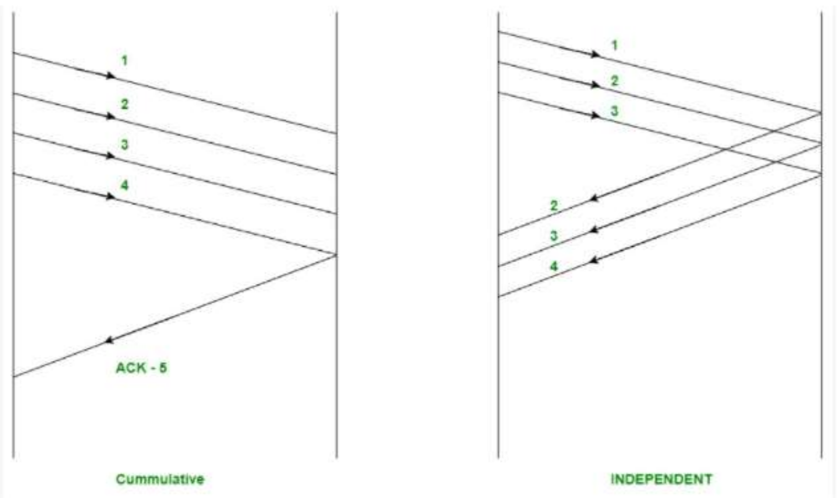
1. One-bit sliding window protocol:**[Protocol:4]**

2. Go-back n sliding window protocol (GBN) **:[Protocol:5]**

3. Selective repeat sliding window protocol (SR) **:[Protocol:6]**

Acknowledgements

- There are 2 kinds of acknowledgements :
  - **1. Cumulative Ack**:
  - **2. Independent Ack**:
- **Cumulative Ack**: is used for many packets. The main advantage is traffic is less. A disadvantage is less reliability as if one ack is the loss that would mean that all the packets sent are lost.
- **Independent Ack**: If every packet is going to get acknowledgement independently. Reliability is high here but a disadvantage is that traffic is also high.



- If receiver has received N packets, then the Acknowledgement number will be N+1.
- Time out timer at the sender side should be greater than Acknowledgement timer.

Slide window (Contd…)

Goal- Keep transmission medium busy.

- **Proposed scheme**

Permit the sender to send more than one frame while waiting for the first ACK. This technique is called as pipelining.

- Sender maintains a **sending window**.
- Receiver maintains a **receiving window**.
- Sending window –Sequence nos of frames that have been sent but not yet acknowledged.
- Receiving window –Sequence nos of frames the receiver can accept
- Eg: 3-bit sequence no-frames are numbered from 0-7.

If the sender has more than 8 frames--→0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 1 2……

Contd…

- Max window size- 2^n (n- no of bits in the sequence no)
- Eg: $\boxed{0\ 1\ 2\ 3\ 4\ 5\ 6\ 7}$ 0 1 2 3….

  Sending window

- i.e sender can send 2^n frames without waiting for an ACK.
- When ACK for a frame comes in –Lower edge of the window is advanced by 1
- When a frame is sent upper edge of the window is advanced by 1.
- **Issue for a noisy channel**
- Frame in the middle of a long stream is damaged/lost.
- What should the receiver do with all the correct frames following the damaged one ?
- **Constraint** : Receiving Data link layer has to handover the packets to the network layer in sequence

Contd…

Two approaches to deal with errors in the presence of pipelining :

1. Go back N
2. Selective Repeat

1. Go back N:

- ▶Discard all subsequent frames following the damaged frames sending no ACKs.
- ▶Eventually the sender times out and retransmits all the unacknowledged frames in order starting with the damaged or lost one.

- **Operation :**
- Assume no of bits in sequence no 2 .
- Window size : 4 frames

Sliding window(Go back N)

- Sending window

$\boxed{0\ 1\ 2\ 3}$ 0 1 2 3......

- Sender has sent 4 frames 0 to 3 and waits for an ACK.
- ACK0 and ACK1 received by the sender

$\boxed{2\ 3\ 0\ 1}$ 2 3 0 1

Now sent 0 1

- Frame 2 lost not received but received 3 0 1
- Receiver discards 3 0 and 1

Note:

A receiving station does not acknowledge each received frame explicitly . If a sending station received an ack for frame j and later receives an ack for frame k it assumes all frames between j and k have been received correctly.

Contd…

- Adv: Reduces no. of acks and lessens the n/w traffic.
- What should be the max sender's and receivers window size ?
- Assume k ---→ no. of bits in sequence no.
- Frames are numbered from 0 to 2^k-1
- Window size cannot be larger than 2^k

Case1 : Window size larger than 2^k

- Let k=3
- Assume window size =9
- Sender's window---→ $\boxed{0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 0}$ 1 2 3……
- Problem --→

Sender receives an ACK0 . Does not know if it was for first/last frame.

Contd…

Window size must be less than or equal to 2^k

Case 2 : Window size = 2^k

1. At time t1 A sends frames 0-7 to B.
2. B receives each one correctly .
3. At time t2, B sends ACK for the most recent received frame ACK7.
4. **ACK gets lost**
5. Next frame expected by B is frame numbered 0.
6. A does not receive ACK and hence retransmits frames 0 through 7 at time t3.
7. At t4 B receives frame 0. Sequence no matches with the one it is expecting. Hence B accepts it.

Protocol fails(Since B has accepted a duplicate and not a new frame)

Contd…

Reason

- 2 consecutive windows contain the same sequence no.
- Solution reduce the sender's window size by 1

Case 3 : Window size less than 2^k

Sender's window -→ $\boxed{0\ 1\ 2\ 3\ 4\ 5\ 6}$ 7 0 1 2 3 4….

1. A sends frames 0 through 6 at t1.
2. B receives each one correctly .
3. At time t2, B sends ACK for the most recent received frame ACK6.
4. **ACK gets lost**
5. Next frame expected by B is frame numbered 7.
6. A does not receive ACK and hence retransmits frames 0 through 6 at time t3.

Contd...

7. B receives frame 0 through 6 at t4 .

8. B expects frame 7.

9.Hence ignores them.

10. Eventually B sends another ACK6 which A receives and A advances it's window to include frame 7 0 1 2 3 4 5 and the protocol continues

Summary

- **Sender's window size – strictly less than 2^k**

(1 less than MAXSEQ)

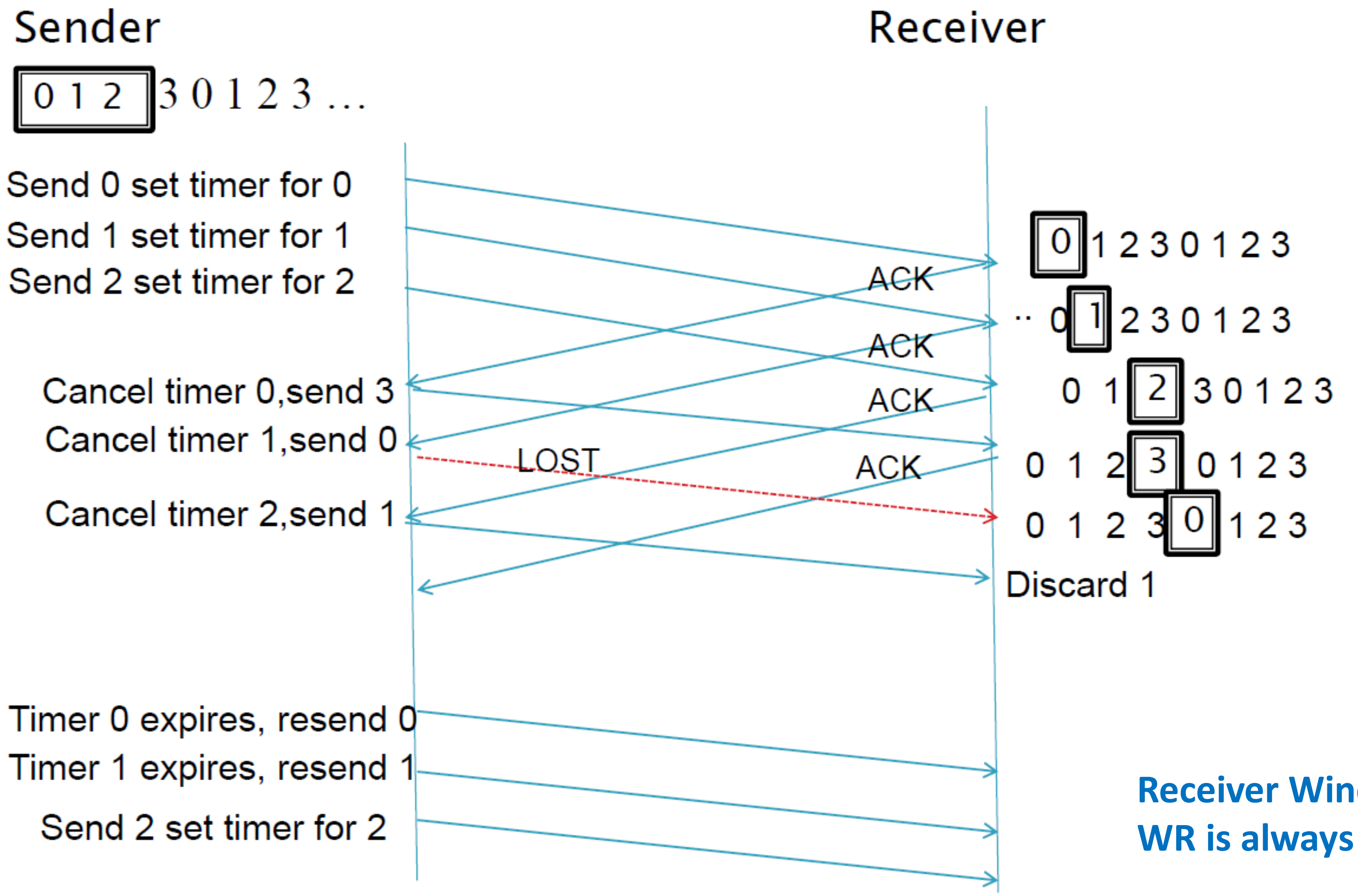
- What should be the receiver's window size?

Frames are always received in order.

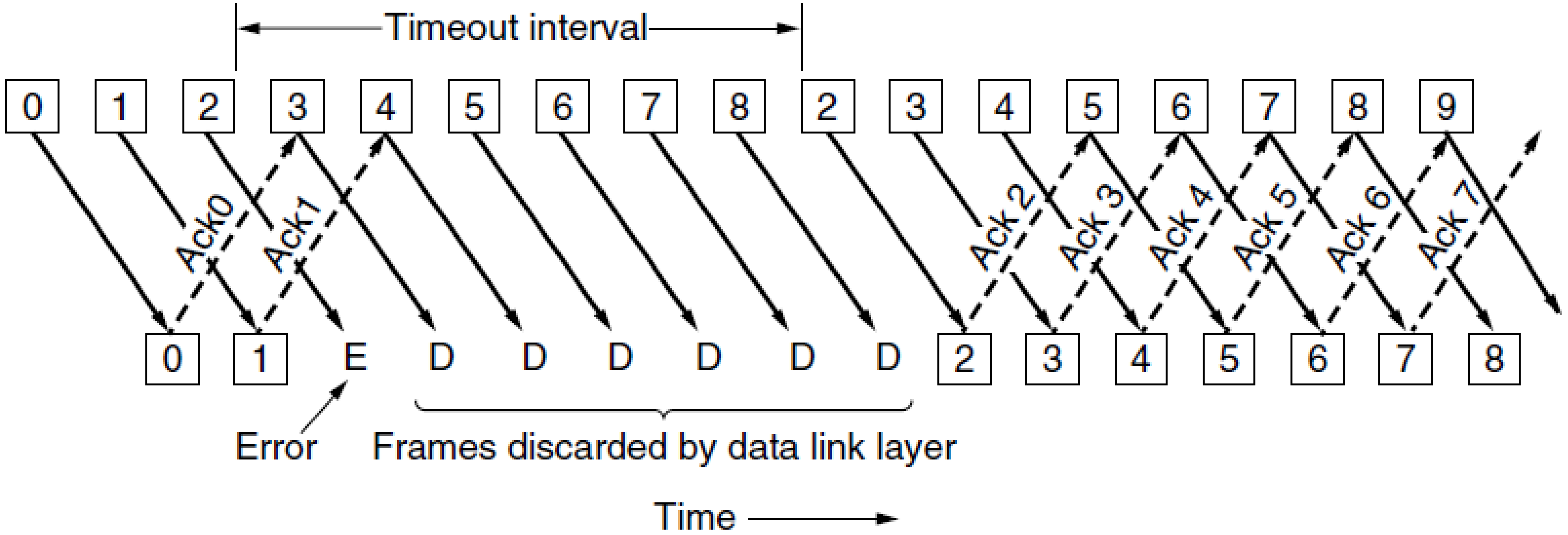
Hence Receiver's window size should not be grater than 1

Contd...

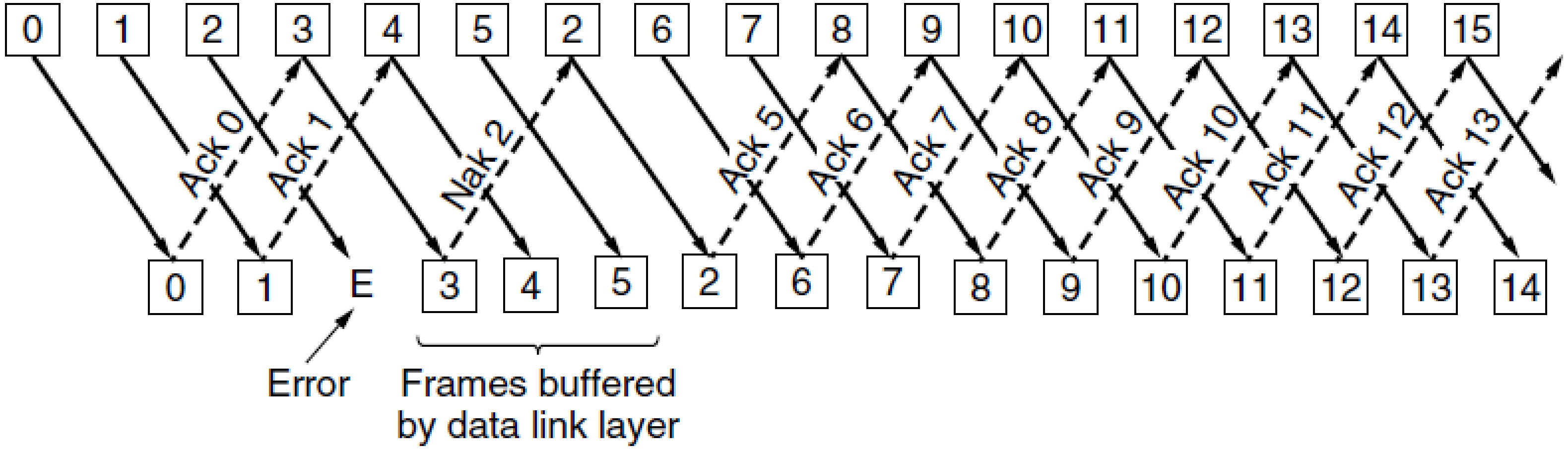
Go back means sender has to go back N places from the last transmitted packet in the unacknowledged window and not from the point where the packet is lost.



Receiver Window Size (WR)
WR is always 1 in GBN.



(a)

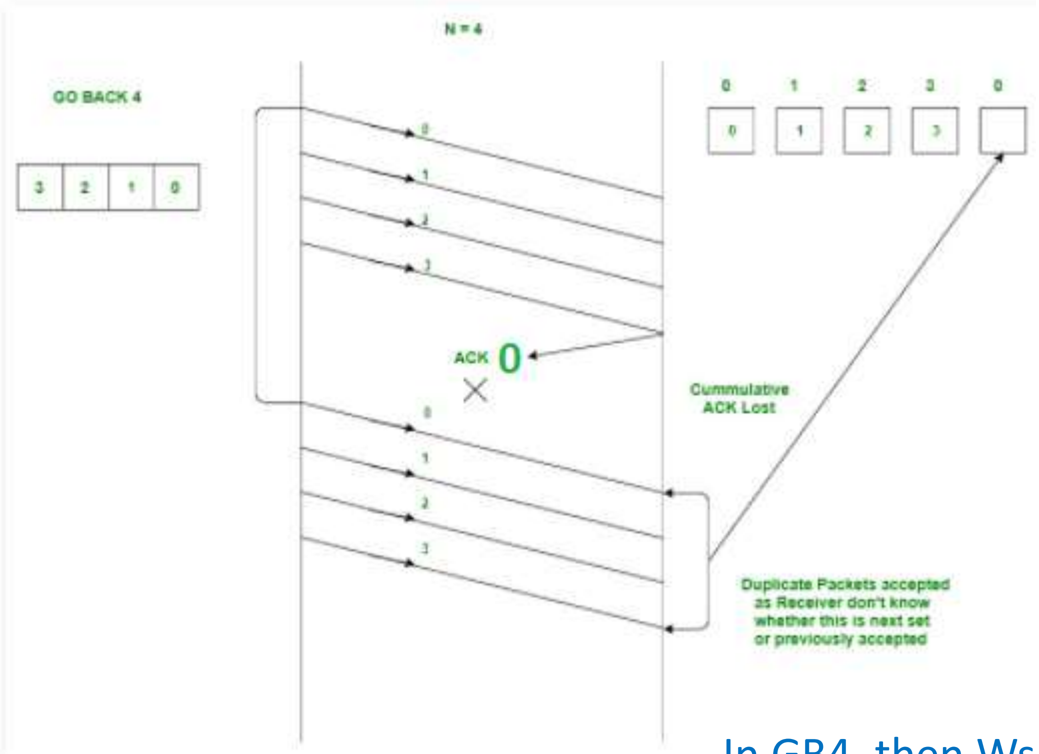


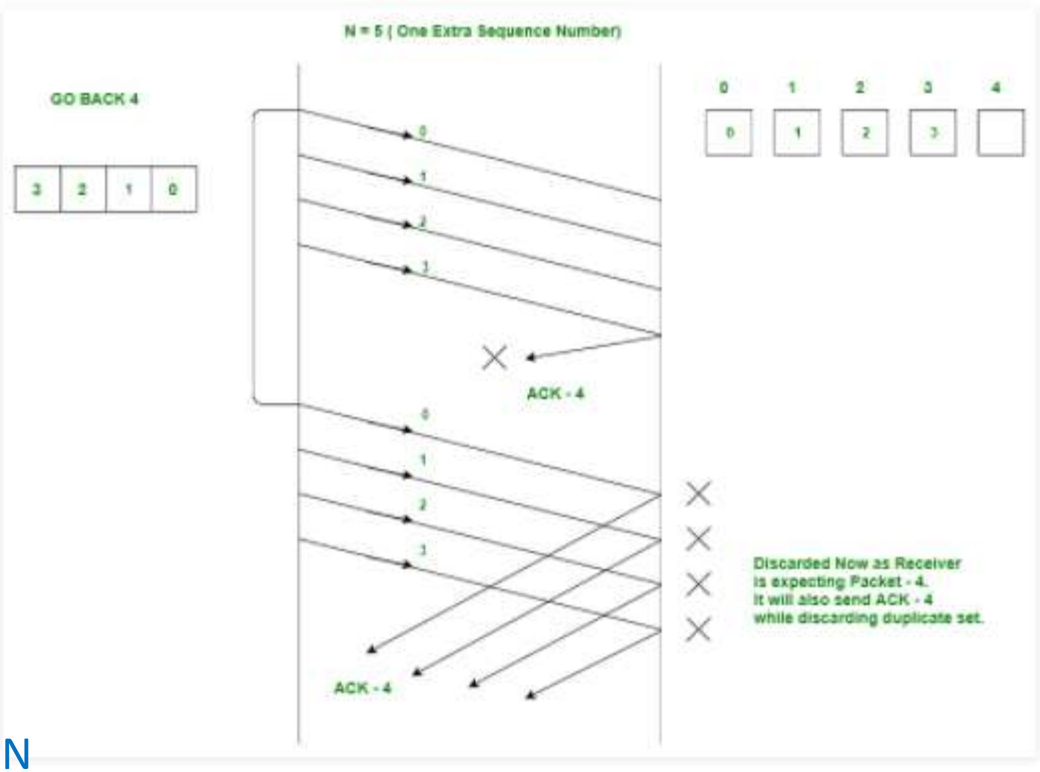
(b)

Fig: Pipelining and error recovery. Effect of an error when
(a) receiver's window size is 1 and (b) receiver's window size is large.

GBN –Sequence number

- The extra 1 is required in order to avoid the problem of duplicate packets





with Sequence numbers 4

(0,1,2,3): window size 4

In GB4, then Ws = 4. N should be always greater than 1 in order to implement pipelining.

with one extra Sequence Number.

(0,1,2,3,4): window size 4

Contd…

- **Setup –Receiver sliding window**
- The size of the receiver window is always **1** and points to the **next expected frame** number to arrive
- This means that frames should arrive **in order**
- If the expected frame is received without errors, the receiver window slides over the **next sequence number**.
- **Operation**
- The receiver sends a positive ACK if a frame has arrived **without error** and **in order** (with the expected sequence number)
- Receiver does not have to acknowledge each individual frame received correctly and in order.
- Receiver can send **cumulative ACK** for several frames (**ACK 4 acknowledges frames (0,1,2,3,4))**
- If the frame is **damaged or out-of-order**, the receiver **discards it** (and **stay** ***silent***) and also **discards all subsequent frames** until it receives the one expected.
  - In this case, **no** ACK will be transmitted
- If the sender timer expires before receiving an ACK, it will **resend** ***ALL*** **frames beginning with the one expired until the last one sent** (Go-Back-N).

Selective Repeat

Drawback of Go back n

- Receiver discards all the correct frames transmitted after the bad one.
- Channel bandwidth wasted on retransmitted frames.

Alternative strategy :

Allow the receiver to accept and buffer the frames following a damaged/lost one.

Called Selective repeat

- Since the receiving station is not required to receive frames in order, the receiving window size is greater than 1.
- Note: A frame arriving out of order can be received as long as it is in the receiving window.

Contd...

- If an arriving frame is in the receiving window it is buffered and not given to the n/w layer until all the predecessor have also arrived

Ex: Assume a 3-bit sequence no.

- Sender sends up to seven frames before waiting for an ACK.
- Sender's window $\boxed{0\ 1\ 2\ 3\ 4\ 5\ 6}$ 7 0 1 2 3 4….
- Receiver's window $\boxed{0\ 1\ 2\ 3\ 4\ 5\ 6}$ 7 0 1 2 3 4….

1. Sender transmits frames 0 through 6
2. All 7 frames are correctly received hence receiver sends ACK6
3. Receiver advances it's window to allow the receipt of next set of sequence nos. 0 1 2 3 4 5 6 $\boxed{7\ 0\ 1\ 2\ 3\ 4\ 5}$ 6 7 0….
4. ACK is lost
5. Sender times out and retransmits frame from 0 through 6

Contd…

6. Receiver receives frames 0-6 but accepts frames 0 1 2 3 4 5 since it lies within it's window.

   But frames 0-5 being duplicates should have been rejected by the receiving station.

   Hence the protocol fails.

PROBLEM

After the receiver advanced it's window, the new range of valid sequence nos. overlapped with the old one.

Consequently the following batch of frames might be either duplicates or new ones(if all ACKS were received).

The receiver has no way of distinguishing these 2 cases.

For a 3 –bit sequence no sending and receiving window should be 4.

- Sender

Receiver

[0 1 2 3] 4 5 6 7 0 1 2 3 4....

[0 1 2 3] 4 5 6 7 0 1 2 3 4....

[0 1 2 3] 4 5 6 7 0 1 2 3 4....

0 1 2 3 [4 5 6 7] 0 1 2 3 4....

ACK 3

ACK 3 Lost

Sender times out and retransmits frames 0 through 3

[0 1 2 3] 4 5 6 7 0 1 2 3 4....

0 1 2 3 [4 5 6 7] 0 1 2 3 4....

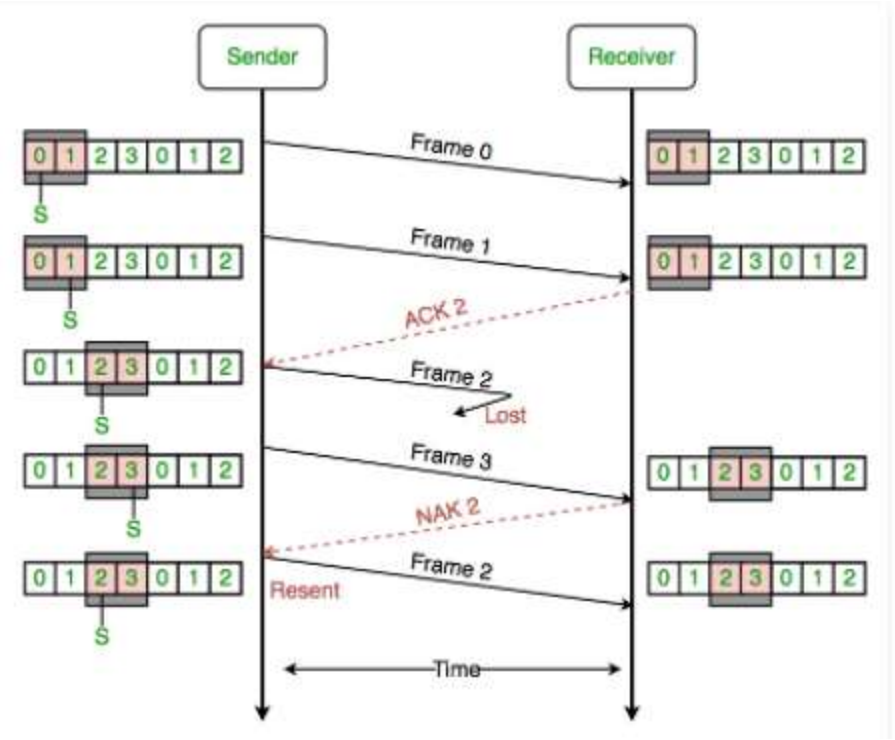
0 1 2 3

Receiver discards frames 0 1 2 3 and sends ACK3

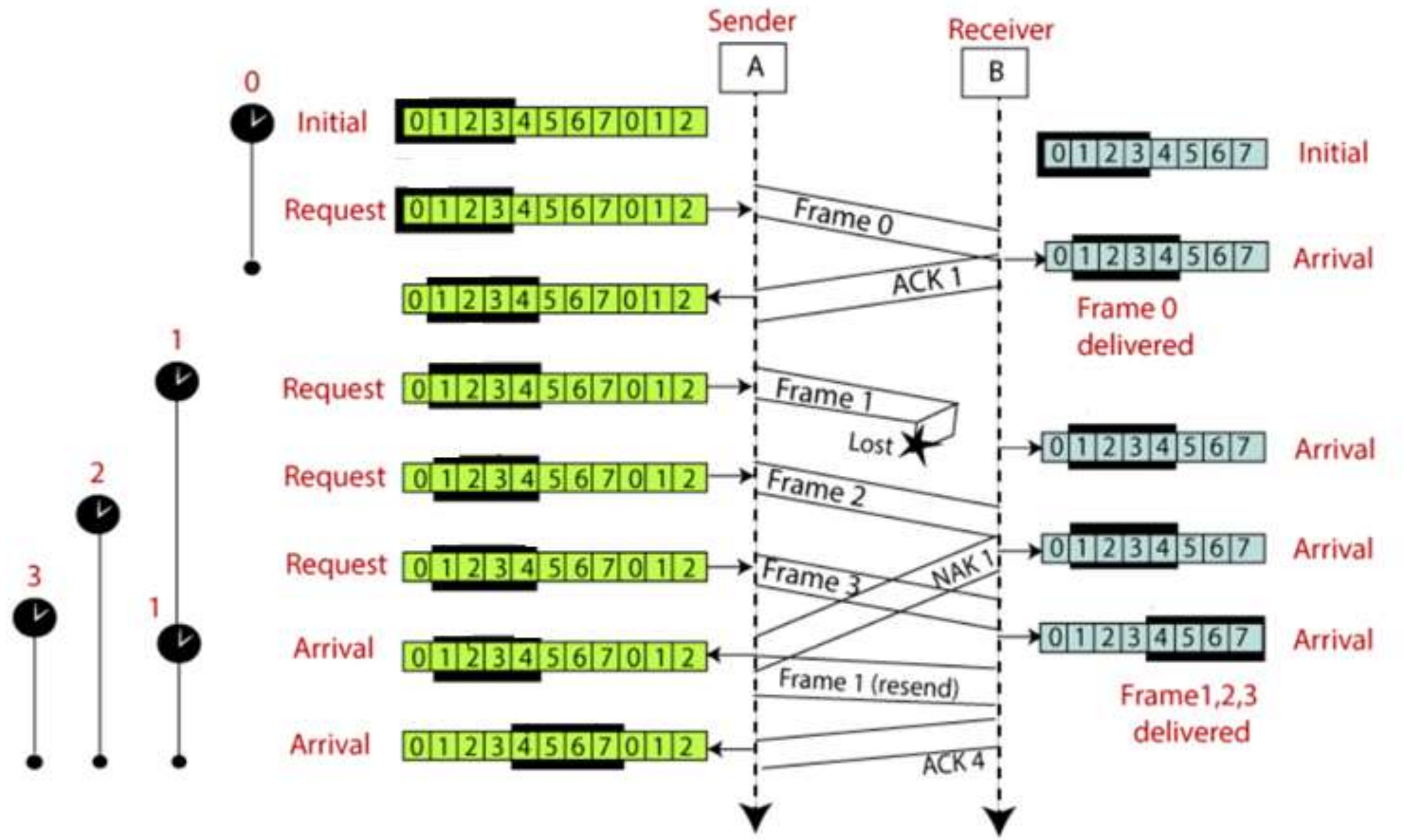
0 1 2 3 [4 5 6 7] 0 1 2 3 4....

0 1 2 3 [4 5 6 7] 0 1 2 3 4....

- **Sender’s Windows (Ws) = Receiver’s Windows (Wr).**
- Window size should be **less than or equal to half** the sequence number in SR protocol. This is to avoid packets being recognized incorrectly. If the windows size is greater than half the sequence number space, then if an ACK is lost, the sender may send new packets that the receiver believes are retransmissions.
- Sender can transmit new packets as long as their number is with W of all unACKed packets.
- Sender retransmit un-ACKed packets after a timeout – Or upon a NAK if NAK is employed.
- Receiver ACKs all correct packets.
- Receiver stores correct packets until they can be delivered in order to the higher layer.
- In Selective Repeat ARQ, the size of the sender and receiver window must be at most one-half of 2^m.



Ex: Window Size=2, Sequence No=0,1,2,3



Ex: Window Size =4, so Sequence No=0,1,2,3,4,5,6,7

Contd…

	One bit sliding window	Go back n	Selective repeat
Max Sender's Window size	1	$2^k - 1$	2^{k-1}
Max receiver's window size	1	1	2^{k-1}

Contd..

Sr. No.	Key	Stop and Wait protocol	GoBackN protocol	Selective Repeat protocol
1	**Sender window size**	In Stop and Wait protocol, Sender **window size is 1.**	In Go Back N protocol, Sender **window size is N**.	In Selective Repeat protocol, Sender **window size is N.**
2	**Receiver Window size**	In Stop and Wait protocol, Receiver **window size is 1.**	In Go Back N protocol, Receiver **window size is 1.**	In Selective Repeat protocol, Receiver **window size is N**.
3	**Minimum Sequence Number**	In Stop and Wait protocol, **Minimum Sequence is 2** Number is 2.	In Go Back N protocol, **Minimum Sequence Number is N+1** where N is number of packets sent.	In Selective Repeat protocol, **Minimum Sequence Number is 2N** where N is number of packets sent.
4	**Efficiency**	In Stop and Wait protocol, **Efficiency formula is 1/(1+2*a)** where a is ratio of propagation delay vs transmission delay.	In Go Back N protocol, **Efficiency formula is N/(1+2*a)** where a is ratio of propagation delay vs transmission delay and N is number of packets sent.	In Selective Repeat protocol, **Efficiency formula is N/(1+2*a)** where a is ratio of propagation delay vs transmission delay and N is number of packets sent.
5	**Acknowledgement Type**	In Stop and Wait protocol, **Acknowledgement type is individual.**	In Go Back N protocol, **Acknowledgement type is cumulative.**	In Selective Repeat protocol, **Acknowledgement type is individual and cumulative.**
6	**Supported Order**	In Stop and Wait protocol, **no specific order is needed** at receiver end.	In Go Back N protocol, **in-order delivery** only is accepted at receiver end.	In Selective Repeat protocol, **out-of-order deliveries** also can be accepted at receiver end.
7	**Retransmissions**	In Stop and Wait protocol, in case of **packet drop, number of retransmission is 1.**	In Go Back N protocol, in case of **packet drop, numbers of retransmissions are N.**	In Selective Repeat protocol, in case of **packet drop, number of retransmission is 1.**

Some important Terms

- **Transmission Delay (Tt)** – Time to transmit the packet from host to the outgoing link. If B is the Bandwidth of the link and D is the Data Size to transmit

  Tt = D/B

- **Propagation Delay (Tp)** – It is the time taken by the first bit transferred by the host onto the outgoing link to reach the destination. It depends on the distance d and the wave propagation speed s (depends on the characteristics of the medium).

  Tp = d/s

- **Efficiency** – It is defined as the ratio of total useful time to the total cycle time of a packet. For stop and wait protocol

- **Total cycle time** = Tt(data) + Tp(data) +
  Tt(acknowledgement) + Tp(acknowledgement)
  = Tt(data) + Tp(data) + Tp(acknowledgement)
  = Tt + 2*Tp

- **Efficiency** = Useful Time / Total Cycle Time
  = Tt/(Tt + 2*Tp) (For Stop and Wait)
  = 1/(1+2a) [Using a = Tp/Tt]

- **Efficiency Of GBN** = N/(1+2a)
  where a = Tp/Tt
  Effective Bandwidth or Throughput
  = Efficiency * Bandwidth
  = (N/(1+2a)) * B

- **Effective Bandwidth(EB) or Throughput** – Number of bits sent per second.
  EB = Data Size(L) / Total Cycle time(Tt + 2*Tp)
  Multiplying and dividing by Bandwidth (B),
  = (1/(1+2a)) * B [Using a = Tp/Tt]
  = Efficiency * Bandwidth

- **Efficiency of Selective Repeat** Protocol (SRP) is same as GO-Back-N's efficiency : N/(1+2a)
  Buffers = N + N
  Sequence number = N(sender side) + N (Receiver Side)

Example 1

Q1) Station A needs to send a message consisting of 9 packets to Station B using a sliding window (window size 3) and go-back-n error control strategy. All packets are ready and immediately available for transmission. If every 5th packet that A transmits gets lost (but no acks from B ever get lost), then what is the number of packets that A will transmit for sending the message to B?

a) 12

b) 14

c) 16

d) 18

Ans: c)

Example 2

Q) Station A uses 32 byte packets to transmit messages to Station B using a sliding window protocol. The round trip delay between A and B is 80 milliseconds and the bottleneck bandwidth on the path between A and B is 128 kbps. What is the optimal window size that A should use? [GATE CS 2006 exam]
(A) 20
(B) 40
(C) 160
(D) 320

Answer (B)

- Round Trip propagation delay = 80ms
- Frame size = 32*8 bits
- Bandwidth = 128kbps
- Transmission Time = 32*8/(128) ms = 2 ms

- Let n be the window size.

- UtiliZation = n/(1+2a) where a = Propagation time / transmission time
- = n/(1+80/2)
- For maximum utilization: n = 41 which is close to option (B)