Unit - 3

# Medium Access Sub Layer

# Topic-2
# Go-Back-N ARQ Protocol

Prepared by K.Lakshmipathi raju, Assistant professor
Dept of IT

# Go-Back-N ARQ

**Go-Back-N ARQ** is a flow control protocol where the sender continues to send several frames specified by a window size even without receiving feedback from the receiver node. It can be said that it's a special case of the general sliding window protocol where the transmitter or sender window size is **N** and the receiver's window size is **1**, which means that it can transmit N frames to the receiving node before waiting for a feedback.Now the receiver's duty is to keep the track of the sequence number of the next frame it expects to receive and sends the feedback after every data packet it receives. Once the sender has sent all the frames in its window, it makes sure that it has then received all the feedbacks of the transmitted data packets, if in any case, it hasn't received a feedback of any data packet and the time out timer expires, it then resends all the data packets again, starting from the lost data packet to the final data packet.

Let's take an example

Consider a sender has to send data packets indexing from **p1 to p5**, it sends all the data packets in order (from p1 to p5), but the receiver has only received p1 and the data packet p2 is lost somewhere in the network, then the receiver declines all the data packets after p2 ( i.e. p3, p4, p5 ) because the receiver is waiting for packet p2 and will not accept any other data packet than that. So, now as the time out time index of p2 expires, the sender goes back 3 packets and starts sending all the data packets from p2 to p5 again.

# Go-Back-N ARQ

**Sender window size (WS):**

In Go-Back-N ARQ, **N** is the sender window size, which we can see in the above example was **5**. Now, here **N should be greater than 1** in order to implement pipelining. **If N=1**, then our system reduces to Stop & Wait protocol.

Now the efficiency of Go-Back-N ARQ = **N/(1+2a)**, where **a = tp/tt**.

Where **tp** is **propagation delay** and **tt** is the **transmission delay**

Also, **tt = D/B**; and here **D = data size** and **B = bandwidth**

And **tp = d/s**, here **d = distance** and **s = wave propagation speed**.

Now to find the effective bandwidth (or throughput),

Effective bandwidth = efficiency * bandwidth, which means,

Effective bandwidth = **(N/(1+2a)) *B**

**Receiver window size (WR):**

WR is always equal 1 in the case of Go-Back-N ARQ.

# Feedbacks/Acknowledgements

There are basically 2 types of feedbacks/acknowledgments:

1. **Cumulative Ack:** Here we use only one feedback for many data packets, because of this the main advantage we get is that the traffic is less. But it can also result in a huge drawback which is, if one acknowledgment is lost then it means that all the data packets transmitted are lost.

2. **Independent Ack:** Here every data packet gets acknowledged independently. Here the reliability is high, but the main drawback is high traffic.

# Feedbacks/Acknowledgements

Go-Back-N ARQ uses a cumulative acknowledgment technique, which means receiver starts an acknowledgment timer whenever it receives a data packet which is fixed & when it expires, it will transmit a cumulative acknowledgment for the number of data packets received in that interval of time out timer. Now, if the receiver has received N data packets, then the feedback/acknowledgment number is going to be N+1. The important point here is that the acknowledgment timer will not start after the expiry of the first-timer, but it will do so when the receiver has received a data packet. The thing which should be kept in mind is that the time out timer at the sender node must be greater than the acknowledgment timer.

# Thank You