

主讲人: 聂兰顺

本讲主题

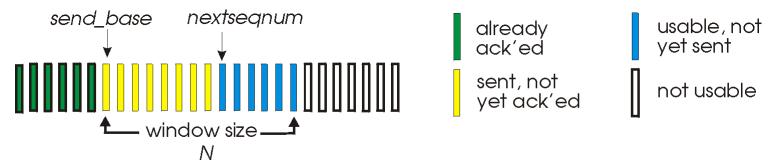
Go-Back-N协议



Go-Back-N(GBN)协议: 发送方



- *分组头部包含k-bit序列号
- ❖窗口尺寸为N,最多允许N个分组未确认



- ❖ACK(n): 确认到序列号n(包含n)的分组均已被正确接收
 - 可能收到重复ACK
- ❖为空中的分组设置计时器(timer)
- ❖超时Timeout(n)事件: 重传序列号大于等于n, 还未收到ACK的所有分组



GBN: 发送方扩展FSM

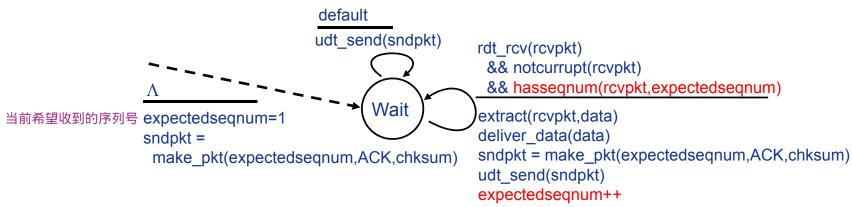


```
rdt send(data)
                               if (nextseqnum < base+N) { 如果已发送的数量小于可发送数量
                                 sndpkt[nextseqnum] = make pkt(nextseqnum,data,chksum)
                                 udt send(sndpkt[nextsegnum])
                                 if (base == nextsegnum)
                                   start timer
                                 nextsegnum++
                               else
                                refuse data(data)
发送的第一个 base=1
发送的最后一个nextsegnum=1
                                                 timeout 超时
                                                 start timer
                                    Wait
                                                 udt send(sndpkt[base])
                                                 udt send(sndpkt[base+1])
         rdt rcv(rcvpkt)
          && corrupt(rcvpkt)
                                                 udt send(sndpkt[nextseqnum-1])
                                 rdt rcv(rcvpkt) &&
                                   notcorrupt(rcvpkt)
                                 base = getacknum(rcvpkt)+1
                                 If (base == nextsegnum)
                                   stop_timer
                                  else
                                   start timer
```



GBN:接收方扩展FSM ▽

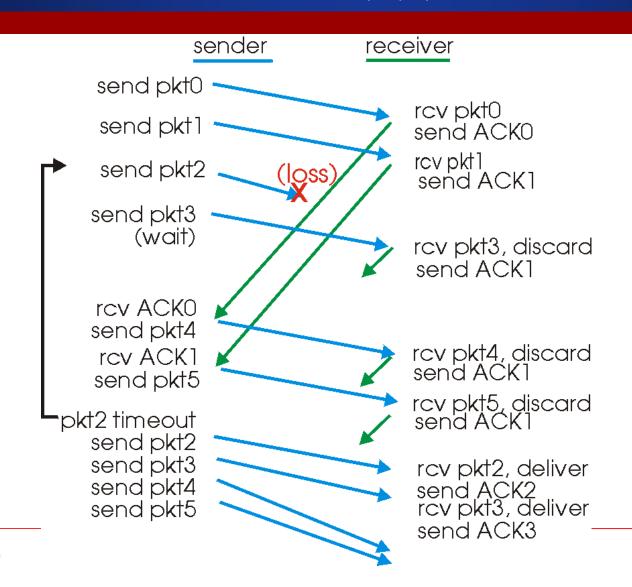




- ❖ACK机制:发送拥有最高序列号的、已被正确接收的分组的ACK
 - 可能产生重复ACK
 - 只需要记住唯一的expectedseqnum
- *乱序到达的分组:
 - 直接丢弃→接收方没有缓存
 - 重新确认序列号最大的、按序到达的分组



GBN示例





练习题

□数据链路层采用后退N帧(*GBN*)协议,发送方已经发送了编号为 0~7的帧。当计时器超时时,若发送方只收到0、2、3号帧的确认 ,则发送方需要重发的帧数是多少?分别是那几个帧?

□解:根据GBN协议工作原理,GBN协议的确认是累积确认,所以此时发送端需要重发的帧数是4个,依次分别是4、5、6、7号帧。



