

# 一、后退N帧的滑动窗口



# 二、GBN 发送方必须响应的三件事

## 1. 上层的调用

上层要发送数据时，发送方先检查发送窗口是否已满，如果未**满**，则产生一个帧并将其发送；如果窗口已**满**，发送方只需将数据返回给上层，暗示上层窗口已满。上层等一会再发送。（实际实现中，发送方可以缓存这些数据，窗口不满时再发送帧）。

## 2. 收到了一个 ACK

GBN协议中，对n号帧的确认采用**累积确认**的方式，标明接收方已经收到n号帧和它之前的全部帧。

## 3. 超时事件：

协议的名字为后退N帧/回退N帧，来源于出现丢失和时延过长帧时发送方的行为。就像在停等协议中一样，定时器将再次用于恢复数据帧或确认帧的丢失。如果出现超时，发送方重传所有已发送但未被确认的帧。

即此处橙色标注的帧都会被重新发送。



已经发送但  
等待确认的

### 三、GBN接收方

**正确接收:** 如果正确收到n号帧, 并且按序, 那么接收方为n帧发送一个ACK, 并将该帧中的数据部分交付给上层。

**其余情况** 其余情况都丢弃帧, 并为最近按序接收的帧重新发送ACK。接收方无需缓存任何失序帧, 只需要维护一个信息: `expectedseqnum` (下一个按序接收的帧序号)。

### 四. 窗口长度

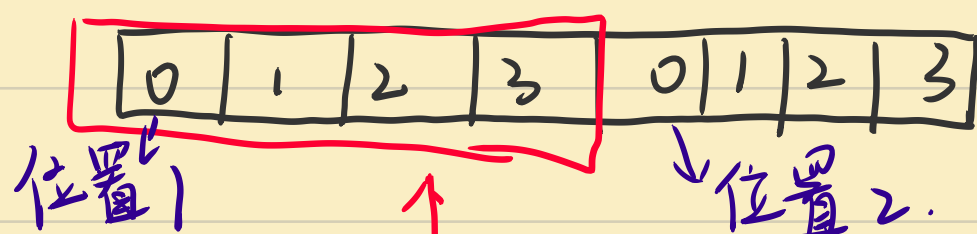
若采用n个比特对帧编号, 那么发送窗口的尺寸 $W_T$ 应满足:  $1 \leq W_T \leq 2^n - 1$ 。因为发送窗口尺寸过大, 就会使得接收方无法区别新帧和旧帧。

例如 2个比特编号.

那么应当有4种: 00, 01, 10, 11.

分别为帧0, 帧1, 帧2, 帧3.

发送时编号是重复利用的:



窗口尺寸应当为  $2^n - 1 = 3$

那么如果为(4)

那么0, 1, 2, 3全发送但全在途中丢失,  
当超时时定时器启动, 这4个帧重新发送,  
导致接收方无法确认重发的4个帧是位置1还是位置2

发送窗口大小为  $1 \leq W_T \leq 2^{n-1}$

接收窗口大小为 1.

## 例题一:

数据链路层采用了后退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 ( )。

A. 2   B. 3   C. 4   D. 5

收到0, 2, 3, 说明3及3以前的都确认了。

那么应重发4, 5, 6, 7. 共4帧。

故选C。

## 例题二:

主机甲与主机乙之间使用后退N帧协议 (GBN) 传输数据, 甲的发送窗口尺寸为1000, 数据帧长为1000字节, 信道带宽为100Mb/s, 乙每收到一个数据帧立即利用一个短帧 (忽略其传输延迟) 进行确认, 若甲、乙之间的单向传播时延是50ms, 则甲可以达到的最大平均数据传输率约为 ( )。

A. 10Mb/s   B. 20Mb/s   C. 80Mb/s   D. 100Mb/s

窗口全发送要的时间:

$$\frac{1000 \times 1000 \times 8 \text{ b}}{100 \times 10^6 \text{ b/s}} = 80 \text{ ms.}$$

$$\begin{aligned}\text{第一个帧确认的时间} &= \text{第一帧发送} + RTT \\ &= \frac{1000 \times 8}{100 \times 10^6} + 2 \times 50 \\ &= 100.08 \text{ ms}.\end{aligned}$$

$$100.08 > 80$$

相当于一个窗口内所有帧发送要  $100.08 \text{ ms}$ .

$$\begin{aligned}\text{那么速度} &= \frac{\text{总数据量}}{\text{时间}} = \frac{1000 \times 1000 \times 8 \text{ b}}{100.08 \text{ ms}} \\ &= 80 \text{ Mb/s}.\end{aligned}$$

故选 C.

## 五. 性能分析.

连续发送帧提高了信道的利用率.

但重传时一些正确的帧也需要被重传,  
使效率降低.

→ 选择重传协议.