

# 基于滑动窗口的流量 控制机制



# 滑动窗口应用于流量控制

## 流量控制帧

- 肯定确认帧RRn: 准备接收从n开始的帧
- 否定确认帧RNRn: 已接收到n-1的所有帧但不能再接收了

## 双向传输（全双工）

- 每个站都保持两个窗口
- 发送窗口控制正向发送
- 接收窗口控制逆向接收



# 捎带确认技术

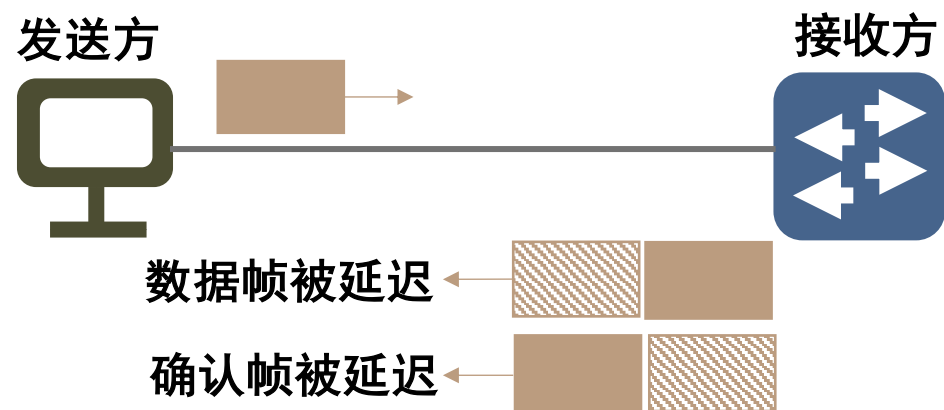
?

对于全双工操作来说，  
接收方有数据要发，还  
要回复确认帧，怎么办？

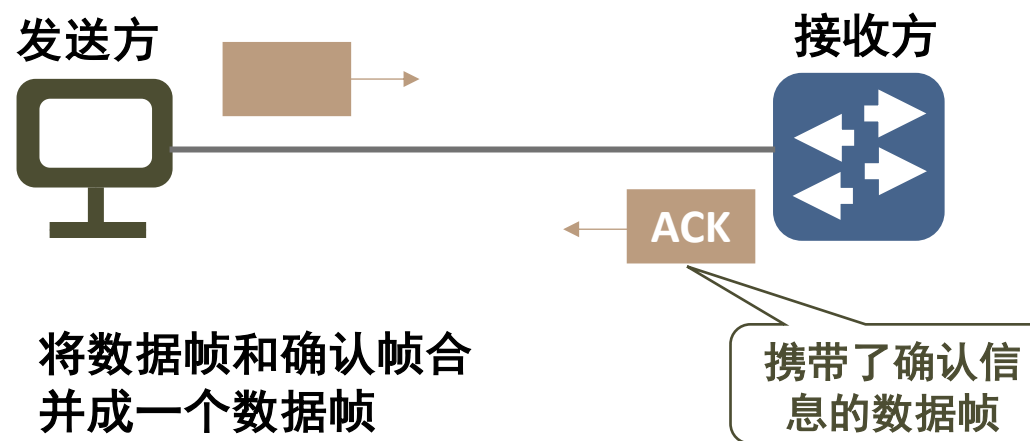
解决方案

**捎带技术：既有数据又有确认  
时，将两者合在一个帧中发送，  
即以数据帧捎带上确认信息。**

**普通确认：数据帧含有序号  
确认帧含有确认号**



**捎带确认：数据帧含有序号+确认号  
确认帧含有确认号**



确认帧    ACK    带确认的数据帧    数据帧



北京大学

# 累计确认技术

普通确认：假设发送方发送了5个数据帧，接收方没有数据要发。则接收方要发送5个确认帧。

节省带宽

累计确认：接收方可对收到的K个帧 ( $K < \text{发送窗口}$ ) 发一个确认，告知发送方已正确接收前(K-1)帧并期待第K个帧。



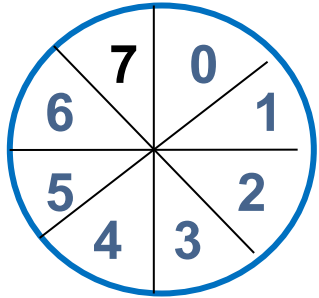
接收方用一个确认帧通知发送方自己的接收进展，由此节省了带宽。

 确认帧 (ACK)

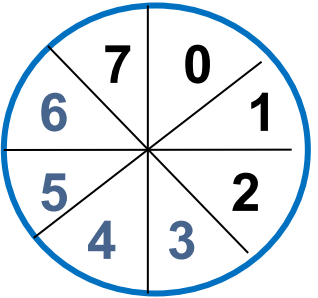
 数据帧



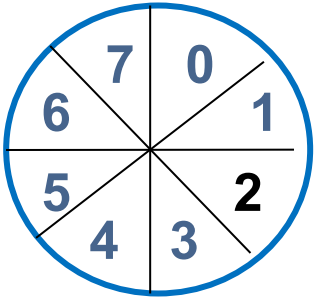
# 滑动窗口流量控制示例



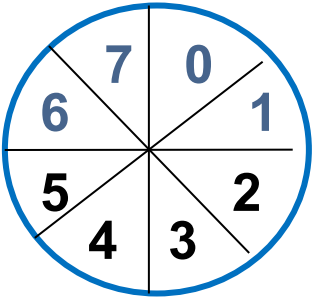
可发送帧0~帧6



连发帧0~帧2



收到确认帧



发送帧3~帧5后

待发帧



t1

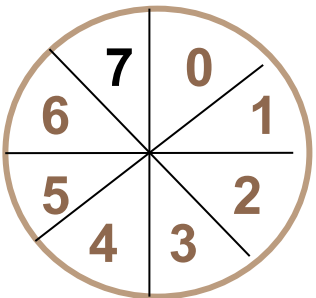


t4

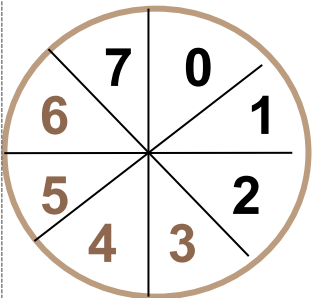


t5

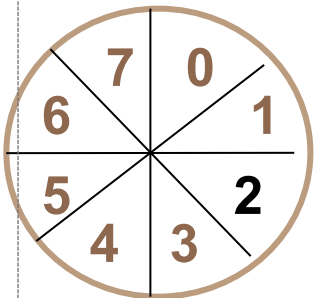
时间



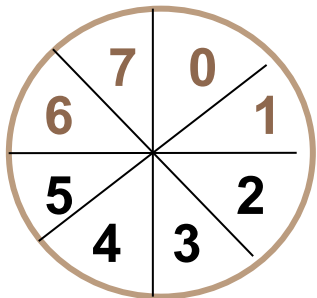
能接收帧0~帧6



接收帧0~帧2



反馈确认帧后



接收帧3~帧5



确认帧 ( ACK )

数据帧