

这里我们注意到, GBN 协议中综合了我们将在 3.5 节中学习 TCP 可靠数据传输构件时遇到的所有技术。这些技术包括使用序号、累积确认、检验和以及超时/重传操作。

3.4.4 选择重传

在图 3-17 中, GBN 协议潜在地允许发送方用多个分组“填充流水线”, 因此避免了停等协议中所提到的信道利用率问题。然而, GBN 本身也有一些情况存在着性能问题。尤其是当窗口长度和带宽时延积都很大时, 在流水线中会有很多分组更是如此。单个分组的差错就能够引起 GBN 重传大量分组, 许多分组根本没有必要重传。随着信道差错率的增加, 流水线可能会被这些不必要重传的分组所充斥。想象一下, 在我们口述消息的例子中, 如果每次有一个单词含糊不清, 其前后 1000 个单词(例如, 窗口长度为 1000 个单词)不得不被重传的情况。此次口述会由于这些反复述说的单词而变慢。

顾名思义, 选择重传(SR)协议通过让发送方仅重传那些它怀疑在接收方出错(即丢失或受损)的分组而避免了不必要的重传。这种个别的、按需的重传要求接收方逐个地确认正确接收的分组。再次用窗口长度 N 来限制流水线中未完成、未被确认的分组数。然而, 与 GBN 不同的是, 发送方已经收到了对窗口中某些分组的 ACK。图 3-23 显示了 SR 发送方看到的序号空间。图 3-24 详细描述了 SR 发送方所采取的动作。

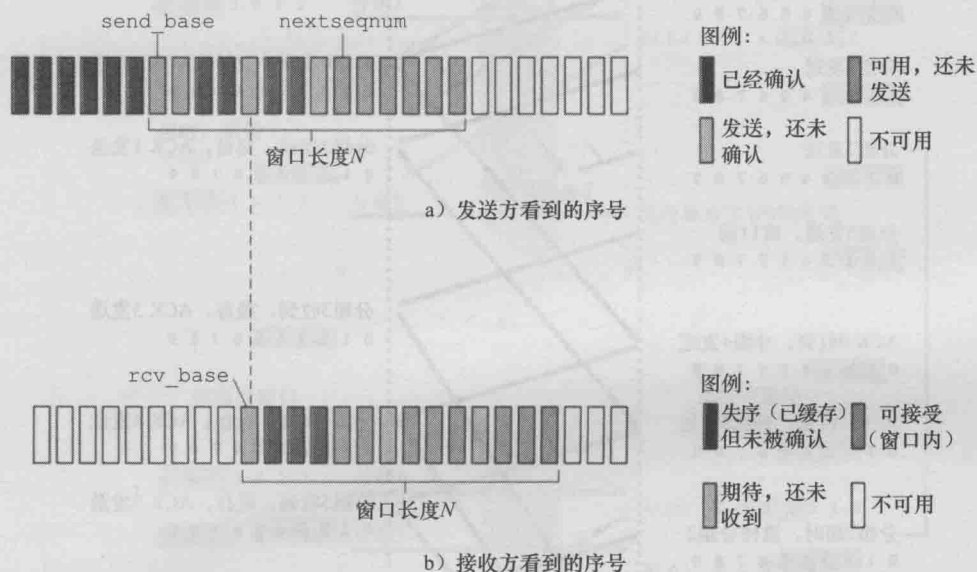


图 3-23 选择重传 (SR) 发送方与接收方的序号空间

1. 从上层收到数据。当从上层接收到数据后, SR 发送方检查下一个可用于该分组的序号。如果序号位于发送方的窗口内, 则将数据打包并发送; 否则就像在 GBN 中一样, 要么将数据缓存, 要么将其返回给上层以便以后传输。
2. 超时。定时器再次被用来防止丢失分组。然而, 现在每个分组必须拥有其自己的逻辑定时器, 因为超时发生后只能发送一个分组。可以使用单个硬件定时器模拟多个逻辑定时器的操作 [Varghese 1997]。
3. 收到 ACK。如果收到 ACK, 倘若该分组序号在窗口内, 则 SR 发送方将那个被确认的分组标记为已接收。如果该分组的序号等于 `send_base`, 则窗口基序号向前移动到具有最小序号的未确认分组处。如果窗口移动了并且有序号落在窗口内的未发送分组, 则发送这些分组。

图 3-24 SR 发送方的事件与动作