

6 选择性回传协议

回退 N 帧协议简化了接收方的进程。接收方只记录一个变量，没有必要缓冲失序分组；它们被简单地丢弃。然而，如果下层网络层丢失很多分组，那么这个协议是低效的。每当一个分组丢失或被破坏，发送方要重新发送所有未完成分组，即使有些失序分组已经被安全完整地接收了。如果网络层由于网络拥塞，丢失了很多分组，那么重发所有这些未完成分组将会使得拥塞更严重，最终更多的分组丢失。这具有雪崩效应，可能导致网络全部瘫痪。另一个协议，称为选择性回传协议（Selective-Repeat (SR) protocol），已经被设计出来，正如其名字所示，只是选择性重发分组，即那些确实丢失的分组。

选择性回传协议也使用两个窗口：一个发送窗口和一个接收窗口。然而，这些窗口与回退 N 帧中的不同。首先，发送窗口的最大值更小；它是 $2m - 1$ 。

选择性回传协议接收窗口与回退 N 帧中的接收窗口完全不同。接收窗口的大小和发送窗口等大（最大 $2m - 1$ ）。选择性回传协议允许和接收窗口一样多的分组失序到来并被存储，直到有一组连续分组被传递到应用层。因为发送窗口和接收窗口的大小是相同的，在发送窗口的所有分组可以失序到达并被存储，直到它们可以被传递。然而，我们需要强调的是，在可靠协议中，接收方从不向应用层传递失序分组。

计时器

理论上讲，选择性重复为每个未完成分组使用一个计时器。当一个计时器终止，只有一个相应分组被发送。换言之，GBN（回退 N 帧）将未完成分组看做一个组；SR（选择性重复）将它们单独处理。然而，绝大多数实现了 SR 的传输层协议只使用一个计时器。出于这个原因，我们只使用一个计时器。

确认

这两个协议之间还有一点不同。在 GBN 中 `ackNo` 是累积的；它定义了下一个预期分组的序号，确认了之前的分组都安全完整到达。在 SR 中确认的语义是不同的。在 SR 中，`ackNo` 定义了被安全完整接收的一个分组；对其他分组没有反馈信息。

窗口大小

为什么发送窗口和接收窗口最多为 $2m$ 的一半。例如，我们选择 $m = 2$ ，这意味着窗口大小为 $2m/2$ 或 $2(m - 1) = 2$ 。如果窗口大小为 2，并且所有确认丢失，那么分组 0 的计时器超时且分组 0 被重发，因此这个重复分组被正确地丢弃（序号 0 不在窗口内）。当窗口大小为 3，并且所有确认丢失，那么发送方发送分组 0 的副本。然而，这次，接收方窗口期待接收分组 0（0 是窗口的一部分），因此它接收了分组 0，并且不把它看做一个重复分组，但是作为下一个循环中的分组。这明显是一个错误。因此在选择性重复中，发送方和接收方窗口的大小最多为 $2m$ 的一半。

参考 <http://book.51cto.com/art/201212/375308.htm>