

P40. 考虑图 3-58。假设 TCP Reno 是一个经历如上所示行为的协议，回答下列问题。在各种情况中，简要地论证你的回答。

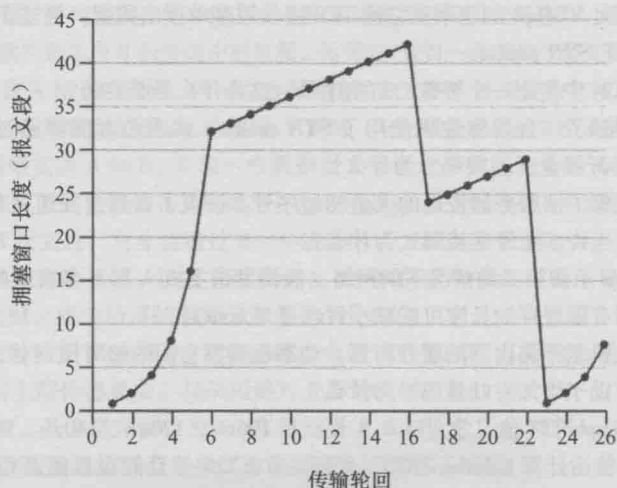


图 3-58 TCP 窗口长度作为时间的函数

- 指出 TCP 慢启动运行时的时间间隔。
 - 指出 TCP 拥塞避免运行时的时间间隔。
 - 在第 16 个传输轮回之后，报文段的丢失是根据 3 个冗余 ACK 还是根据超时检测出来的？
 - 在第 22 个传输轮回之后，报文段的丢失是根据 3 个冗余 ACK 还是根据超时检测出来的？
 - 在第 1 个传输轮回里，ssthresh 的初始值设置为多少？
 - 在第 18 个传输轮回里，ssthresh 的值设置为多少？
 - 在第 24 个传输轮回里，ssthresh 的值设置为多少？
 - 在哪个传输轮回内发送第 70 个报文段？
 - 假定在第 26 个传输轮回后，通过收到 3 个冗余 ACK 检测出有分组丢失，拥塞的窗口长度和 ssthresh 的值应当是多少？
 - 假定使用 TCP Tahoe（而不是 TCP Reno），并假定在第 16 个传输轮回收到 3 个冗余 ACK。在第 19 个传输轮回，ssthresh 和拥塞窗口长度是什么？
 - 再次假设使用 TCP Tahoe，在第 22 个传输轮回有一个超时事件。从第 17 个传输轮回到第 22 个传输轮回（包括这两个传输轮回），一共发送了多少分组？
- P41. 参考图 3-56，该图描述了 TCP 的 AIMD 算法的收敛特性。假设 TCP 不采用乘性减，而是采用按某一常量减小窗口。所得的 AIAD 算法将收敛于一种平等共享算法吗？使用类似于图 3-56 中的图来证实你的结论。
- P42. 在 3.5.4 节中，我们讨论了在发生超时事件后将超时时间间隔加倍。为什么除了这种加倍超时时间间隔机制外，TCP 还需要基于窗口的拥塞控制机制（如在 3.7 节中学习的那种机制）呢？
- P43. 主机 A 通过一条 TCP 连接向主机 B 发送一个很大的文件。在这条连接上，不会出现任何分组丢失和定时器超时。主机 A 与因特网连接链路的传输速率表示为 R bps。假设主机 A 上的进程能够以 S bps 的速率向 TCP 套接字发送数据，其中 $S = 10 \times R$ 。进一步假设 TCP 的接收缓存足够大，能够容纳整个文件，并且发送缓存只能容纳这个文件的百分之一。如何防止主机 A 上的进程连续地向 TCP 套接字以速率 S bps 传送数据呢？还是用 TCP 流量控制呢？还是用 TCP 拥塞控制？或者用其他措施？阐述其理由。
- P44. 考虑从一台主机经一条没有丢包的 TCP 连接向另一台主机发送一个大文件。
- 假定 TCP 使用不具有慢启动的 AIMD 进行拥塞控制。假设每当收到一批 ACK 时，cwnd 增加 1 个 MSS，并且假设往返时间大约恒定，cwnd 从 6MSS 增加到 12MSS 要花费多长时间（假设没有丢包事件）？