

大的区别。但在运输层和应用层是否也有重大差别呢？我们很容易认为这些差别是很小的，因为在有线和无线网络中的网络层均为上层提供了同样的尽力而为服务模式。类似地，如果在有线和无线网络中都是使用诸如 TCP 和 UDP 的协议提供运输层服务，那么应用层也应该保持不变。在某个方面我们的直觉是对的，即 TCP 和 UDP 可以（也确实）运行在具有无线链路的网络中。在另一方面，运输层协议（特别是 TCP）通常在有线和无线网络中有时会有完全不同的性能。这里，在性能方面区别是明显的，我们来研究一下其中的原因。

前面讲过，在发送方和接收方之间的路径上，一个报文段不论是丢失还是出错，TCP 都将重传它。在移动用户情况下，丢失可能源于网络拥塞（路由器缓存溢出）或者切换（例如，由于重路由选择报文段到移动用户新的网络接入点引入的时延）。在所有情况下，TCP 的接收方到发送方的 ACK 都仅仅表明未能收到一个完整的报文段，发送方并不知道报文段是由于拥塞，或在切换过程中，还是由于检测到比特差错而被丢弃的。在所有情况下，发送方的反应都一样，即重传该报文段。TCP 的拥塞控制响应在所有场合也是相同的，即 TCP 减小其拥塞窗口，如 3.7 节讨论的那样。由于无条件地降低其拥塞窗口，TCP 隐含地假设报文段丢失是由于拥塞而非出错或者切换所致。我们在 6.2 节看到，在无线网络中比特错误比在有线网络中普遍得多。当这样的比特差错或者切换丢失发生时，没理由让 TCP 发送方降低其拥塞窗口（并因此降低发送速率）。此时路由器的缓存的确可能完全是空的，分组可以在端到端链路中丝毫不受拥塞阻碍地流动。

研究人员在 20 世纪 90 年代早期到中期就认识到，由于无线信道的高比特差错率和切换丢失的可能性，TCP 的拥塞控制反应在无线情况下可能会有问题。有三大类可能的方法用于处理这一问题：

- 本地恢复。本地恢复方法的目标是在比特差错出现的当时和当地（如在无线链路中）将其恢复。如在 6.3 节学习的 802.11 ARQ 协议，或者使用 ARQ 和 FEC 的更为复杂的方法 [Ayanoglu 1995]）。
- TCP 发送方知晓无线链路。在本地恢复方法中，TCP 发送方完全不清楚其报文段跨越一段无线链路。另一种方法是让 TCP 发送方和接收方知道无线链路的存在，从而将在有线网络中发生的拥塞性丢包和在无线网络中发生的差错/丢包区分开，并且仅对有线网络中的拥塞性丢包采用拥塞控制。在假设端系统能够做出这种区分的情况下，[Balakrishnan 1997] 详细研究了多种类型的 TCP。[Liu 2003] 研究了在一个端到端路径中区分有线部分丢包和无线部分丢包的技术。
- 分离连接方法。在分离连接方法中 [Bakre 1995]，移动用户和其他端点之间的端到端连接被打断为两个运输层连接：一个从移动主机到无线接入点，一个从无线接入点到其他通信端点（我们假定它是有线的主机）。该端到端连接因此是由一个无线部分和一个有线部分级连形成的。经无线段的运输层能够是一个标准的 TCP 连接 [Bakre 1995]，或是一个特别定制运行在 UDP 上的差错恢复协议。[Yavatkar 1994] 研究了经无线连接使用运输层选择性重传协议。在 [Wei 2006] 中的测量报告指出了分离 TCP 连接广泛用于蜂窝数据网络中，通过使用分离 TCP 连接，上述问题的确能够有很大改进。

我们这里有关无线链路上的 TCP 的讨论是十分简要的。在无线网络中有关 TCP 挑战和解决方案的深入展望能够在 [Hanabali 2005; Leung 2006] 中找到。我们鼓励读者去查