

时延角度看,却远不是一个理想状态。甚至在这种(极端)理想化的情况中,我们已经发现了拥塞网络的一种代价,即当分组的到达速率接近链路容量时,分组经历巨大的排队时延。

2. 情况2: 两个发送方和一台具有有限缓存的路由器

现在我们从下列两个方面对情况1稍微做一些修改(参见图3-45)。首先,假定路由器缓存的容量是有限的。这种现实世界的假设的结果是,当分组到达一个已满的缓存时会被丢弃。其次,我们假定每条连接都是可靠的。如果一个包含有运输层报文段的分组在路由器中被丢弃,那么它终将被发送方重传。由于分组可以被重传,所以我们现在必须更小心地使用发送速率这个术语。特别是我们再次以 λ_{in} 字节/秒表示应用程序将初始数据发送到套接字中的速率。运输层向网络中发送报文段(含有初始数据或重传数据)的速率用 λ'_{in} 字节/秒表示。 λ'_{in} 有时被称为网络的供给载荷(offered load)。

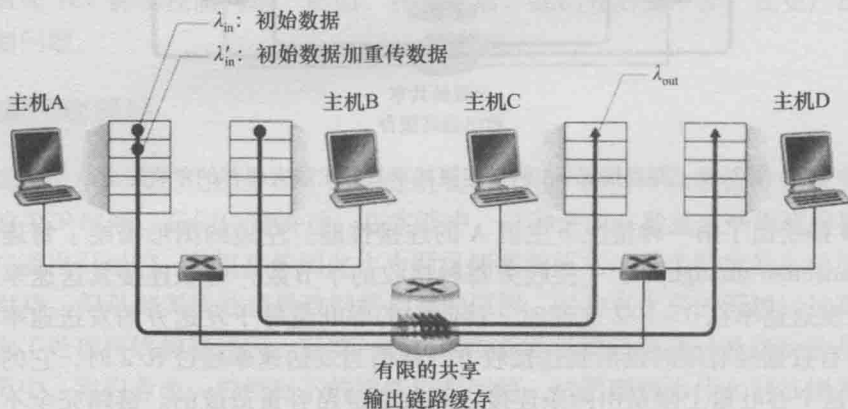


图 3-45 情况 2: (有重传的) 两台主机与一台拥有有限缓存的路由器

在情况2下实现的性能将完全取决于重传的方式。首先,考虑一种不真实的情况,即主机A能够以某种方式(不可思议地!)确定路由器中的缓存是否空闲,因而仅当缓存空闲时才发送一个分组。在这种情况下,将不会产生丢包, λ_{in} 与 λ'_{in} 相等,并且连接的吞吐量就等于 λ_{in} 。图3-46a中描述了这种情况。从吞吐量的角度看,性能是理想的,即发送的每个分组都被接收到。注意到在这种情况下,平均主机发送速率不能超过 $R/2$,因为假定不会发生分组丢失。

接下来考虑一种更为真实的情况,发送方仅当在确定了一个分组已经丢失时才重传。(同样,所做的假设有一些弹性。然而,发送主机有可能将超时时间设置得足够长,以无形中使其确信一个还没有被确认的分组已经丢失。)在这种情况下,性能就可能与图3-46b所示的情况相似。为了理解这时发生的情况,考虑一下供给载荷 λ'_{in} (初始数据传输加上重传的总速率)等于 $R/2$ 的情况。根据图3-46b,在这一供给载荷值时,数据被交付给接收方应用程序的速率是 $R/3$ 。因此,在所发送的0.5R单位数据当中,从平均的角度说,0.333R字节/秒是初始数据,而0.166R字节/秒是重传数据。我们在此看到了另一种网络拥塞的代价,即发送方必须执行重传以补偿因为缓存溢出而丢弃(丢失)的分组。

最后,我们考虑下面一种情况:发送方也许会提前发生超时并重传在队列中已被推迟但还未丢失的分组。在这种情况下,初始数据分组和重传分组都可能到达接收方。当然,