西 北 师 范 大 学

教 师 授 课 教 案

课 程 名 称 ： 计算机网络

课 程 学 分 ： 3

课 程 类 别 ： 必修

授 课 班 级 ：

授 课 教 师 ：

周　　次	第 13 周	教案号	13
授课时间	（2 节/周）		
授课内容概要	第 5 章　运输层 5.7　TCP 的流量控制 5.7.1　利用滑动窗口实现流量控制 5.7.1　必须考虑传输效率 5.8　TCP 的拥塞控制 5.8.1　拥塞控制的一般原理 5.8.2　几种拥塞控制方法 5.8.3　随机早期检测 RED 5.9　TCP 的运输连接管理 5.9.1　TCP 的连接建立 5.9.2　TCP 的连接释放 5.9.3　TCP 的有限状态机		
目的要求	1、理解滑动窗口。 2、了解可靠传输的实现。 3、掌握 TCP 报文段的首部格式、流量控制、拥赛控制、连接管理。		
重点	TCP 报文段的首部格式、流量控制、拥赛控制、连接管理。		
难点	TCP 报文段的首部格式、流量控制、拥赛控制、连接管理。		
作业布置	课后习题。		
参考书	1、《计算机网络》（第五版），谢希仁编著，电子工业出版社。 2、《计算机网络》（第四版），Andrew S. Tanenbaum 著，潘爱民译，清华大学出版社。 3、《数据通信与计算机网络》，高传善、钱松荣、毛迪林编著，高等教育出版社。 4、《Computer Networks (Forth Edition)》，Andrew S. Tanenbaum 著，清华大学出版社。 5、《TCP/IP 协议簇》，Behrouz A.Forouzan & Sophia Chung Fegan 著，谢希仁译，清华大学出版社。		

课　　型	理论课	学时分配	复 习	10　　分钟
主要教具	计算机、投影仪、黑板、粉笔		讲 授	160　　分钟
教学方法	启发式		指 导	分钟
教学手段	讲授		总 结	10　　分钟
备注				

授 课 过 程 及 内 容	备 注
5.7 TCP 的流量控制 5.7.1 利用滑动窗口实现流量控制 • 流量控制(flow control)就是让发送方的发送速率不要太快，既要让接收方来得及接收，也不要使网络发生拥塞。 • 利用滑动窗口机制可以很方便地在 TCP 连接上实现流量控制。 • 图示举例解释。 5.7.2 必须考虑传输效率 • 为提高效率应尽可能的*避免在网络上传输很短的数据*。 • 可以用不同的机制来控制 TCP 报文段的发送时机。 • TCP 实现中广泛采用 Nagle 算法 • 对接收端的考虑：“糊涂窗口综合症” 5.8 TCP 的拥塞控制 5.8.1 拥塞控制的一般原理 • 在某段时间，若对网络中某资源的需求超过了该资源所能提供的可用部分，网络的性能就要变坏——产生拥塞(congestion)。 • 拥塞产生的条件 • 拥塞控制与流量控制的关系 • 拥塞控制所起的作用 • 拥塞控制的一般原理：开环控制和闭环控制 5.8.2 几种拥塞控制方法： • 前提假定：数据是*单向传送*的，另一方只传确认。*接收方缓存足够大*，不会约束发送窗口，发送窗口只由网络拥塞程度决定。 • 1. 慢开始和拥塞避免：发送方维持一个叫做拥塞窗口 cwnd (congestion window)的状态变量。拥塞窗口的大小取决于网络的拥塞程度，并且动态地在变化。 • 发送方让自己的发送窗口等于拥塞窗口。如再考虑到接收方的接收能力，则发送窗口还可能小于拥塞窗口。 • 发送方控制拥塞窗口的原则是：只要网络没有出现拥塞，拥塞窗口就再增大一些，以便把更多的分组发送出去。但只要网络出现拥塞，拥塞窗口就减小一些，以减少注入到网络中的分组数。	

- 慢开始算法的原理：在主机刚刚开始发送报文段时可先设置拥塞窗口 cwnd = 1，即设置为一个最大报文段 MSS 的数值(这里为了方便，以报文个数作为单位了，实际上 TCP 是以字节为单位)。
- 在每收到一个对新的报文段的确认后，将拥塞窗口加 1，即增加一个 MSS 的数值。(由小到大逐渐增大拥塞窗口的大小)
- 用这样的方法逐步增大发送端的拥塞窗口 cwnd，可以使分组注入到网络的速率更加合理。
- 传输轮次 (transmission round)：每经过一个传输轮次，拥塞窗口 cwnd 就加倍。“传输轮次”更加强调：把拥塞窗口 cwnd 所允许发送的报文段都连续发送出去，并收到了对已发送的最后一个字节的确认。
- 设置慢开始门限状态变量 ssthresh
- 当网络出现拥塞时：当发出的报文段没有按时收到确认。
- 无论在慢开始阶段还是在拥塞避免阶段，只要发送方判断网络出现拥塞，就要采取以下措施：
- 1、把慢开始门限 ssthresh 设置为出现拥塞时的发送方窗口值的一半： ssthresh = ssthresh /2;
- 2、然后把设置拥塞窗口： cwnd = 1;
- 3、执行慢开始算法。
- 举例
- 2. 快重传和快恢复：
- 快重传算法：
- 接收方：每收到一个失序的报文段后就立即发出重复确认。这样做可以让发送方及早知道有报文段没有到达接收方。
- 发送方：只要一连*收到三个重复确认*就应当立即重传对方尚未收到的报文段。 (比等到检测到超时才重传要迅速)
- “快” 恢复：
- 1、当发送端收到连续三个重复的确认时，就执行“乘法减小”算法： ssthresh = ssthresh /2
- 2、cwnd= ssthresh /2
- 3、执行拥塞避免算法（“加法增大”），使拥塞窗口缓慢地线性增大。
- 从连续收到三个重复的确认转入拥塞避免

5.8.3 随机早期检测 RED 5.9 TCP 的运输连接管理 5.9.1 TCP 的连接建立：用三次握手建立 TCP 连接；图示解释。 5.9.2 TCP 的连接释放：两个两次握手，图示解释。TCP 连接必须经过时间 2MSL 后才真正释放掉。 • 5.9.3 TCP 的有限状态机：TCP 有限状态机的图中每一个方框都是 TCP 可能具有的状态。 • 每个方框中的大写英文字符串是 TCP 标准所使用的 TCP 连接状态名。状态之间的箭头表示可能发生的状态变迁。 • 箭头旁边的字，表明引起这种变迁的原因，或表明发生状态变迁后又出现什么动作。 • 图中有三种不同的箭头。	