

计算机网络笔记

陈鸿峥

2019.03*

目录

1 计算机网络概述	1
1.1 网络连接方式	1
1.2 因特网	2
1.3 网络服务	3
1.4 网络性能分析	4
2 物理层	4
2.1 编码方式	4
2.2 物理介质	5
3 数据链路层	6
3.1 简介	6
3.2 差错检测	6
3.3 可靠数据传输	7

本课程使用的教材为James F. Kurose和Keith W. Ross的《计算机网络—自顶向下方法（第七版）》。

1 计算机网络概述

计算机网络是自主计算机的互连集合。

1.1 网络连接方式

直接连接的网络

- 点对点(point-to-point)网络：包括专用介质(dedicated medium)、节点/主机

*Build 20190313

- 单向(simplex)
- 半双工(half duplex)
- 全双工(full duplex)
- 多路访问(multiple access)网络：共享介质(shared medium)、广播、碰撞(collision)
 - 单播(unicast)：一对一
 - 多播(multicast)：一对多
 - 广播(broadcast)：一对所有

间接连接的网络

- 中间节点、路由器(router)
- 包(packet)
- 存储转发(store and forward)
- 路由选择(routing)
- 路由表(routing table)
- 目的地(destination)、下一跳(next hop)

1.2 因特网

网络互连：用路由器（或网关gateway）连接起来构成的网络称为互连网络(internetwork)。用实际的物理通信介质及相应的设备把两个或两个以上的网络连接起来的一种网络，如LAN和WAN都可看作是互连网络。

因特网/互联网(Internet)是一种互连网络，可以看作是把世界各地的广域网互连的网络

- 系统域网(System Area Network, SAN)
- 局域网(Local Area Network, LAN)：某一区域内由多台计算机互联成的计算机组，一般是方圆几千米以内
- 城域网(Metropolitan Area Network, MAN)
- 广域网(Wide Area Network, WAN)

因特网：

- 终端系统/主机(end system)：运行网络应用程序
- 通信链路(communication link)：光纤、铜线、无线电、卫星等
- 路由器(router)

因特网的结构：ISP(Internet Service Provider)

- 顶层ISP：主干网（中国电信、中国移动、中国网通）
- 区域ISP：可以私自互联
- 本地ISP

- Access network: 连接订阅者和服务提供商
- Core network: 连接局部提供商
- Network edge: 路由、路由交换其、集成访问设备(IAD)

1.3 网络服务

网络提供的服务:

- 可靠/不可靠: 会不会丢包
- 面向连接/无连接: 需要建立通信线路
- 有确认/无确认
- 数据报服务: 无连接无确认 (因特网)
- 请求相应和消息流服务

因特网体系结构:

- 应用层: 提供对某些专门应用的支持, 如FTP、SMTP、HTTP
- 传输层: 将网络层获得的包在**进程之间**数据传送 (端到端), 如TCP、UDP
- 网络层: **路由选择**, 实现在互联网中的数据传送 (主机到主机), IP、路由协议
- 数据链路层: 在物理网络中传送包 (跳到跳, 节点到节点), PPP、Ethernet
- 物理层: 线上的**比特** (传送原始比特流)

网络层以下不可靠, 以上可靠; 防止丢包的机制: 重发

协议(protocol): 在网络实体(entities)之间传送消息的规则, 如消息的格式、收发消息的次序等

协议栈: 发送时封装(encapsulation), 接收时拆封。每层传输的数据单元都称为包(packets), 都属于某个协议, 又被称为协议数据单元(protocol data unit, PDU)=协议控制信息(protocol control data, PCI)+服务数据单元(SDU)



不同协议则添加不同头部。路由做得事情是拆一层封装, 然后重新加一层。同一个互连网络中网络层协议需要相同, 链路层协议可以不同。

ISO/OSI(open system interconnection)网络七层协议, 在应用层和传输层中间添加两层:

- 表示层(presentationn): 提供数据转换服务, 例如, 加密解密, 压缩解压缩, 数据格式变换
 - 会话层(session): 简化会话实现机制, 例如, 数据流的检查点设置和回滚以及多数据流同步
- 对等实体: 实现相同协议

1.4 网络性能分析

当一个包到达时如果有空闲缓存则排队等待转发, 产生延迟(delay); 如果没有空闲缓存, 则丢弃该包, 造成丢失(loss)。

- 电路交换技术(circuit switching): 电话通信, 实时性好
- 包交换技术(packet-switching): 存储转发

包交换网络中的延迟

- 处理(processing)延迟: 查路由, 存储转发(store-and-forward)技术则延迟很大
- 排队(queueing)延迟
- 发送/传输(transmission)延迟: 包长(bits)/链路带宽(bps, bit per second); 指从发送第一个包到发送最后一个包的间隔
- 传播(propagation)延迟: 指对于一个包来说从发送到接收所需的时间

接收延迟与传播延迟重合。故总延迟 (从第一个包被发送到最后一个包被接收的时间) = 传播延迟 + 发送延迟。

往返时间(round trip time, RTT): 从源主机到目的主机再返回源主机所花的时间

- 带宽(bandwidth): 一条链路或通道可达到的最大数据传输速率(bps)
- 吞吐量(throughput): 一条链路或通路实际数据传输速率

2 物理层

直连网, 不管包。

信息能够被解释为数据(msg/data), 用符号(sign)记录, 用信号(signal)传递(transmit), 用熵(entropy)测量

- 信号: 光、电
- 模拟信号: 连续取值
- 数字信号/跳变信号: 离散取值
- 模拟传输: 模拟信号、放大器(amplifier)
- 数字传输: 数字信号、中继器(repeater)

2.1 编码方式

2.1.1 模拟信号

载波信号(carrier)一般采用正弦波信号: 角频率 ω 、频率 f 、周期 T 、振幅 A 、相位 (φ)

- 频移键控(frequency-shift keying, FSK): 通过不同频率表示不同信息0/1
- 幅移键控(amplitude-shift keying, ASK)
- 相移键控(phase-shift keying, PSK)
- 正交调幅(quadrature amplitude modulation, QAM): 用不同的振幅/相位表示不同的多位信息000 ~ 111

2.1.2 数字信号

1. 单极编码(unipolar): 0V即0, $+EV$ 为1, 但是会产生
 - 时钟漂移: 不同的时钟会有差别, 一定要有跳变
 - 基线漂移: 线很长会有 (积累很多电荷, 以为是1), 一定要有变化/正负
2. 不归零编码/双极编码(non-return-to-zero/bipolar, NRZ) : $-E$ 为0, $+E$ 为1, 解决基线漂移问题 (平衡01); 全是0或全是1, 还是没法区分
3. 不归零反转编码(Inverted, NRZI): 差分码波形, 相邻码元的电位改变表示1, 而电位不改变表示0; 也可以反过来。该表示方法与码元本身电位或极性无关, 而仅与相邻码元的电位变化有关
4. 曼彻斯特(Manchester)编码: 从相邻时刻的中间起 $M - E \sim +E$, $0 \rightarrow 10, 1 \rightarrow 01$, 可克服时钟漂移和基线漂移; 频率高, 传输有问题, 对传输介质要求高
5. 差分曼彻斯特编码: 在每一位开始时间如果跳变则为0, 否则为1
6. 4B/5B编码: 用5比特代表4比特, 多一位冗余; 每个编码没有多于1个前导零和多于2个末端零

2.2 物理介质

2.2.1 分类

有线介质

- 双绞线:
 - 非屏蔽双绞线(unshielded twisted pair, UTP): 四对线 (绿绿白、橙橙白、蓝蓝白、棕棕白), cat6千兆以太网
 - 屏蔽双绞线(STP)
- 同轴电缆(coaxial cable)
- 光导纤维(optical fiber)
 - 单模光纤(single mode): 最大传输速率
 - 多模光纤: 阶跃(step-index)光纤、渐变(graded-index)光纤

无线介质: 地面微波、WiFi、3G网络、卫星

2.2.2 多路复用

- 时分多路复用(time division multiplexing, TDM)
- 频分多路复用(frequency, FDM): 无线电台常用
- 波分多路复用(wavelength): 利用多个激光器在单条光纤上同时发送多束不同波长激光的技术
- 码分多路复用(code)
- 统计多路复用: 动态分配方法共享通信链路, 比如FIFO; 对于多个可变速率的数据流, SDM可以提高链路利用率

交换技术

- 电路交换技术(circuit-switching): 采用FDM、TDM、WDM、CDM技术
- 包交换技术(packet-switching): 采用SDM

3 数据链路层

数据链路层把数据包(packet)从一个节点通过链路(直连网络或物理网络)传给相邻另一个节点

3.1 简介

基本术语如下:

- 节点(node): 主机和路由器
- 链路(link): 连接相邻节点的通道, 有线链路、无线链路、局域网
- 帧(frame): 第二层数据包

功能

- 成帧(framing)
- 差错检测(error detect): 比特错, 纠错
- 差错控制(error control): 丢包、重复、错序、流控制(flow control)
- 介质访问控制(media access control): 多路访问, 碰撞(collision)

3.2 差错检测

在数据报后加校验码(头部加序号), 通过链路传输看是否有数据报/校验码错误
奇偶校验

- 一维偶校验: 只能检错; 最后补一位使得全部为偶数个1, 如010补为010—1, 而101补为101—0
- 二维偶校验: 检错+纠错一位; 横纵同时偶校验

若接收方收到奇数个1, 则有出错

校验和(checksum): 将所有数据加起来

由于需要使用加法器, 校验和一般不用于数据链路层, 而是更高层, 例如IP层和传输层

循环冗余校验码(Cyclic Redundancy Check, CRC): 补充n位后除以一个n+1位的除数, 模2除法 (按位异或, 做减法时没有借位)

接收方连带校验码一起除, 余数为0则没错

链路层常用CRC, 因为检错率很高, 且容易实现 (触发器+异或门)

3.3 可靠数据传输

发送方传送数据帧, 接收方回传确认帧(ACK)

超时则自动重发请求(Automatic Repeat reQuest, ARQ): 每发送一帧都启动一个超时定时器, 哪个帧超时将重传该帧, 并重启定时器

- 停等协议(stop-and-wait): ARQ协议, 只有收到前一个数据帧的确认帧才可以发送下一个数据帧
三种出错情况:

- 数据帧丢失
- 确认帧丢失
- 超时: 收到ACK表明接收方一定收到, 可以发送新的数据帧, 重传的也一定要发ACK

效率十分低, 信道空闲时间长

- 滑动窗口协议(sliding window): ARQ协议, 不需等待前面发送的帧的确认帧返回, 就可以连续发送下一个, 其个数不能超过发送窗口大小(sending window size, SWS) (连续发送数据帧可用序号范围, 用于流控制: 控制发送速度, 否则会发生溢出(overflow), 后面覆盖前面的)

这里的确认帧是指在此之前的帧都已收到

直连网中间没有节点, 后面收到前面一定收到; 只要出错纠正不了直接丢弃

- 回退N协议(go back N): 同滑动窗口连续发送, 但某个ACK没收到则重传在此ACK之后的所有帧 (超时重传), 丢3则4发2

发送窗口需要缓存SWS个帧, 以便重传

不是靠超时重传

发送窗口中序号最小的为sendbase

- 选择性重传(selective repeat): 否定性确认帧(negative acknowledgement, NAK), 要求重传某一帧; 如3丢失, 4发送NAK=3, 5发送ACK=2

接收窗口(receiving window size, RWS)表示接收缓冲区大小($RWS \leq SWS$, 最好是等于, 尽量减少重传帧), 用于确定应该保存哪些帧, 用序号范围表示

超时设长, 2次来回

没有后续帧也会超时重传

无论窗口内窗口外收到都要发确认

min序号个数= $RWS + SWS$

序号少的话导致重复

例 1. 序号8个, $SWS = RWS = 4$, 345670123456, 5丢失

分析. 回退N: 346705670123456

选择性重传: 346705123456

提高滑动窗口协议的效率:

- 选择性确认(selective acknowledgement): 接受方把已收到的帧的序号告诉发送方(收到不用重传告诉发送方哪些已收到, 则只用重传某帧)
- 捎带确认(piggybacking): 通信双方**全双工**方式工作, 接收方在发数据给对方时顺便把确认号也告诉对方(两个滑动窗口, 两边都要发数据), 需要结合下面一起使用
- 延迟确认(delayed acknowledgement): 接收方收到一帧后并不立即发送确认帧, 而是等待一段时间再发送

PPP协议(point-to-point): 点到点网络的数据链路层协议, 主要用于串行电缆、电话线(MODEM)等串行链路; 根据HDLC(high-level data link control)协议进行设计

标志位成帧, 区分包的起始和终结; 同时需要转义

链路层的实现: 在网络接口卡(network interface card, NIC)及其驱动程序上实现, 路由器在接口模块上实现