

计算机网络笔记：中大 2022 人工智能学院课程

1 杂七杂八零碎知识点

- 运输层

- 无连接的套接字用二元组（源端口号，目的端口号）标识，相同目的端口号的数据报会定向到同一套接字。面向连接的套接字用四元组（源 IP，源端口号，目的 IP，目的端口号），每个套接字与同个进程相联系。

源端口号或目的端口号相同，但是源 IP 或目的 IP 不同是什么样的情况？

网络层可以实现根据 IP 地址的传输或者广播。源端口号相同而源 IP 不同是从不同主机发送的相同端口进程。目的端口号相同但目的 IP 不同相当于对同一端口进行广播。

- UDP 计算检验和：相加，进位当作 1 加在后面，最后取反。
- 选择重传协议窗口大小和序号空间的关系：
假设接收方窗口大小为 M ，序号空间为 $\{0, 1, \dots, N\}$ 。则需要满足当所有数据报都选择重传的时候，

- 网络层

- 链路状态 LS 算法就是每一次选择开销最小路径。
- LS 算法路由震荡是因为更新路由影响链路开销。
- 距离向量 DV 算法就是每一次只得到邻居信息。
- DV 算法好消息快，坏消息慢。
- 内部网关 OSPF 协议：洪泛控制 + 全局 Dijkstra
- BGP 热土豆路由选择：只考虑域内开销。
- 因特网控制报文协议 ICMP 用于发送错误报告和网络状态信息的网络层协议。

2 零碎问题以及解答

2.1 如何逐步构建可靠数据传输协议？

经完全可靠信道的可靠数据传输：**rdt1.0**：

- 发送方：`rdt_send(data)` 接受上层数据，`packet=make_pkt(data)`，`udt_send(packet)`。
- 接收方：`rdt_rcv(packet)` 接受分组。
- 关键：直接发送和接受。

具有比特差错可靠信道传输：**rdt2.0**

- 有差错：ACK
- 无差错：NCK

具有比特差错的丢包信道的可靠数据传输：

3 数据链路层

3.1 数据链路层的功能

数据链路层实现帧在一段链路或一个网络中进行传输。

数据链路层使用的**信道**主要有两种：

- 点对点信道：使用一对一的通信方式。**PPP** 协议是目前应用最广泛的点对点协议。
- 广播信道：这种信道有多主机连接，使用一对多的广播通信方式。
 - 有线局域网普遍使用 **CSMA/CD** 协议。
 - 无线局域网则使用 **CSMA/CA** 协议。

数据链路层所处的地位：下图展示了数据链路层在数据流动中的地位和层级。

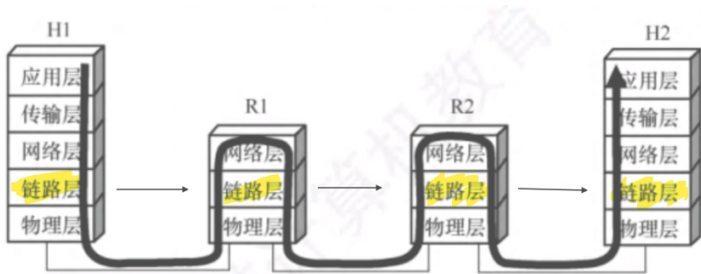


图 1: 数据流动图

点对点信道相关概念

→ 某些概念对广播信号也是适用的。

- 链路：一个节点到相邻节点的一段物理线路，通常是通信路径的一部分。

- 数据链路：链路 + 通信协议 = 数据链路。

所以也称链路为“物理链路”，数据链路为“逻辑链路”。

- 帧：数据链路层对等实体之间进行逻辑通信的协议数据单元。

数据链路层把网络层的数据封装成帧并发送，收到帧后提取数据并上交到网络层。

为网络层提供服务：数据链路层为网络层提供三种服务如下。

- 无确认的无连接服务：

- 不需要先建立链路连接和返回确认。
- 数据传输的可靠性由高层负责。
- 适用于如以太网之类误码率低的信道。

- 有确认的无连接服务：

- 不需要先建立链路连接。
- 目的主机收到帧之后需要返回确认。源主机在规定时间内未收到则超时重传。
- 适用于如无线通信之类误码率较高的信道。

- 有确认的面向连接服务：

- 帧传输过程分为三个阶段：建立链路 → 传输帧 → 释放链路。
- 目的主机收到帧后需要返回确认。
- 适用于可靠性要求较高的场合。

- **链路管理**

- * 链路层连接的建立、维持和释放过程称为链路管理。

- * 首先确认对方是否处于就绪状态，并交换一些必要的信息进行初始化。

数据链路层协议有多种，有**三个基本问题**是共同的：封装成帧、透明传输和差错检验。

封装成帧

- 在一段数据的前后分别添加首部和尾部。

- 帧长：数据长度 + 尾部长度 + 首部长度。

- 帧的数据长度要尽可能大于首尾部长度，从而使得信息传输效率更高。
- 帧越长，传输差错的概率越大。
- 每种链路层协议都规定了数据部分的长度上限：最大传送单元。

- 帧定界：首部和尾部中控制信息的一种，确定帧的界限。使得接收方可区分出帧的起始和终止（帧同步）。在 HDLC 协议中，用标志位 F 来标志帧的开始和结束。

透明传输

- 在数据帧中恰好出现了与帧定界符相同的比特组合（F），可能会误以为传输结束。

- 用透明传输解决该问题：无论传输什么样的数据都无差错。

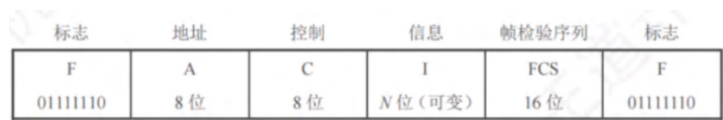


图 2: HDLC 标准帧格式

差错检验

- 将错误分为位错和帧错：
 - 位错：某 bit 出现差错，通常采用循环冗余检验（CRC）发现位错。
 - 帧错：丢失、重复、失序等错误。

流量控制（非基本问题）

- 限制发送方的发送速率，使其不超过接收方的接受能力。
- 在 OSI 体系架构中，数据链路层具有流量控制能力。
- 在 TCP/IP 体系结构中，流量控制被转移到了传输层。
 - 数据链路层：相邻结点流量控制。
 - 传输层：源到目的端之间的流量控制。

3.2 组帧（封装成帧）

数据链路层之所以要将数据封装成帧，是为了以帧为单位处理异常（重发等）。组帧主要解决帧定界、帧同步、透明传输等问题。常见的组帧方法通常有四种。

字符计数法

- 适用帧首部第一个计数字符来记录该帧字节数。
- 多米诺骨牌效应：只要有一帧出错，全盘崩溃。

字节填充法

- 使用特定字节来定界一帧的开始与结束。



图 3: 字节填充法

- 以上图为例：一共有三种特殊字符 SOH、EOT、ESC。
 - SOE：标识帧开始。

- EOT: 标识帧结束。
- ESC: 标识转义。

零比特填充法

- 使用比特流01111110标识开始与结束。
- 发送前：扫描整个数据字段，每遇到 5 个连续的1就自动插入0。
- 收到后：每遇到 5 个连续的1就自动删除后面紧跟的0。
- 优点：容易用硬件实现，性能优于字符填充法。HDLC 协议使用。



图 4: 零比特填充法

违规编码法

- 采用违规的物理层变法作为定界符。
 - 比如在曼切斯特编码中，合法编码有：“高-低”电平（0）和“低-高”电平（1）。违规编码有“高-高”和“低-低”。
- 优点：违规编码法不需要任何填充技术实现透明传输。局域网 IEEE 802 标准使用。

3.3 差错控制

4 网络层

4.1 网络层的功能

- OSI 曾主张在网络层提供面向连接的虚电路服务。
- TCP/IP 网络层提供的是无连接的数据服务。

异构网络互连

TCP/IP 采用标准化协议进行网络互联，参与互联的计算机网络都是用相同的 IP 协议，可以把互联后的网络视为一个虚拟 IP 网络，简称为 IP 网络。

IP 网络的好处是：在 IP 网上主机进行通信，就好像在单个网络上通信看不到各个网络的具体异构细节。

各个层次的中继结构:

物理层：转发器、集线器；数据链路层：网桥或交换机；网络层：路由器；网络层以上：网关。

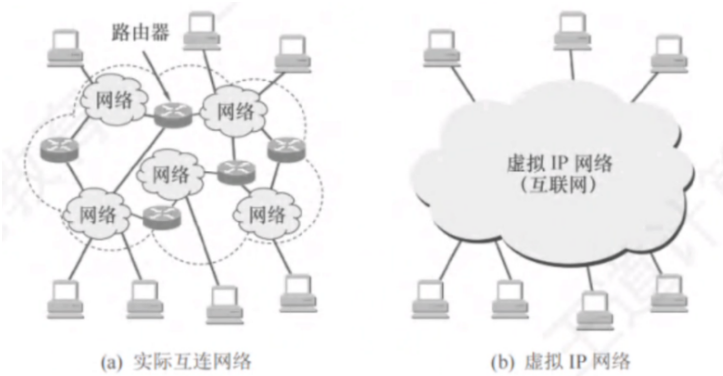


图 5: IP 网络

路由与转发

路由器的两个主要功能：路由选择和分组转发。

- 路由选择：根据协议构造路由表并定期与相邻路由器交换、更新维护信息，以决定分组到达目的地节点的最优路径。
- 分组转发：根据转发表将分组从合适地端口转发出去。

网络层服务：虚电路和数据报

- 虚电路：面向连接
- 数据报：无连接

SDN 及其基本概念

拥塞控制

4.2 IPv4

IPv4 定义了数据传送基本单元：IP 分组的格式及其处理和差错控制细节。

IPv4 分组格式

一个 IP 数据报由首部和数据部分组成。首部前一部分长度固定，共 20B。后一部分长度可变，用于提供错误检测和安全等机制。

IPv4 首部的部分重要字段含义：

- 版本号：IP 协议的版本，确定如何解释数据包的剩余部分。
- 首部长度：由于 IPv4 数据报首部长度可变，所以需要记录首部长度。
- 服务类型（区分服务）：将不同类型的 IP 数据报区分开（如实时与非实时）。
- 数据报长度：IP 总长度，包括首部长度和数据长度。
- 生存时间 TTL：确保数据报不会永远在网络中循环，每当一台路由器处理数据报时， $TTL-1$ 。当 $TTL = 0$ 则丢弃该数据报。



图 6: IPv4 数据报格式

IPv4 编址

4.3 IPv6

4.4 路由算法与路由协议

动态路由算法分为：距离-向量路由算法和链路状态路由算法。

距离-向量路由算法以 *Bellman-Ford* 算法为基础，不断更新：

$$d_x(y) = \min\{c(x, v) + d_v(y)\}$$

其中， v 是 x 的所有邻居。

最常见的是 *RIP* 算法，用跳数作为距离的度量。

链路状态路由算法向全局节点广播链路信息（只广播与之相邻的节点）。每个节点都是用链路状态数据独立计算路径。

4.5 暂存内容：未区分小结

路由器的交换结构

- 经内存交换：输入端口和输出端口之间的交换是在 CPU 的直接控制下完成的。当一个分组到达输入端口时，会通过中断方式向路由选择器发出信号。
- 经总线交换：用共享总线将分组直接传送到输出端，不需要处理器干预。输入端口分配交换机内部标签，输出端口摘除标签。
一次只有一个分组能够跨越总线。

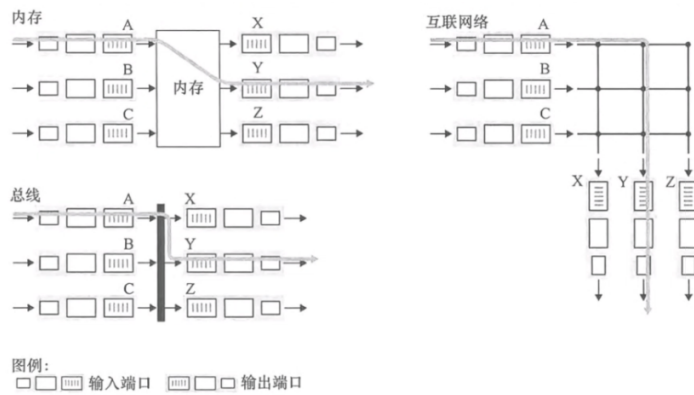


图 7: 三种交换技术

- 经互联网络交换：可以并行分发多个不同输出端口的分组。

分组调度

- 先进先出
- 优先权排队
- 循环排队
- 加权公平排队 *WFQ*：每个类在任何时间间隔内可能收到不同数量的服务。
在类 i 有分组要发送的任何时间间隔中，类 i 将确保收到的服务部分等于 $w_i/(\sum w_j)$
在最坏的情况下，即使所有类都有分组排队，类 i 也可以分配到带宽的 $w_i/(\sum w_j)$ 。

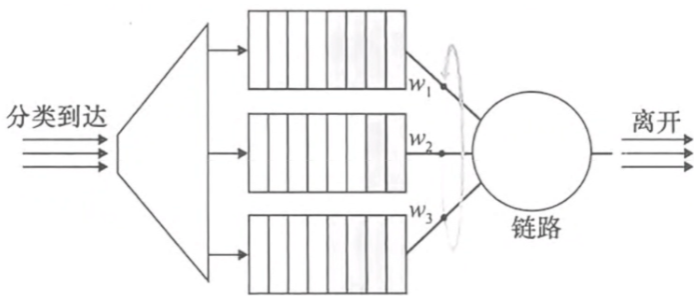


图 8: 加权公平排队

目录与索引

- 1 杂七杂八零碎知识点..... 1
- 2 零碎问题以及解答..... 2
 - 2.1 如何逐步构建可靠数据传输协议？ 2
- 3 数据链路层..... 2
 - 3.1 数据链路层的功能..... 2
 - 3.2 组帧（封装成帧） 4
 - 3.3 差错控制..... 5
- 4 网络层..... 5
 - 4.1 网络层的功能..... 5
 - 4.2 IPv4..... 6
 - 4.3 IPv6..... 7
 - 4.4 路由算法与路由协议..... 7
 - 4.5 暂存内容：未区分小结..... 7
- A 附录 1：协议表..... 9

A 附录 1：协议表