

第七章 网络层

路由相关

网络层协议：

IP ICMP（internet 控制协议） OSPF（内部网关） BGP（外部网关）

为传输层服务（面向连接（复杂功能在网络层） 无连接->复杂功能在传输层）

网络层内部结构：

* 数据报

数据报分组交换

* 虚电路

先建立虚电路 所有分组在虚电路上传输 然后拆除虚电路

区别：数据包每个都要带着完整原地址 目的地址 但是虚电路不用

虚电路路由器里面会维护虚电路的状态信息

时间差别 地址查找信息和建立连接的时间差

路由选择的次数 每次和一次

路由算法：

静态路由： 最短路径路由 洪泛 基于流量

动态路由： DV LS

距离向量： 将自己对于全网的认知都告诉邻居

链路状态： 将自己对于邻居的认知洪范给全网

汇集 

狄杰斯特拉算法

洪泛算法-> 收到啥就往外发啥（不往来的地方发） 产生大量分组 有可能有回路

要怎么解决回路呢？ 站点计数器 每经过一次就-1（但是应该会设定一个初值（ppt没讲 但是只要设一个最后大的正值就可以了）吧）

基于流量的路由算法

估计延迟 找延迟最小的那个

有一个通信量矩阵 带宽矩阵



i	Line	λ_i (pkts/sec)	C_i (kbps)	μC_i (pkts/sec)	T_i (msec)	Weight
1	AB	14	20	25	91	0.171
2	BC	12	20	25	77	0.146
3	CD	6	10	12.5	154	0.073
4	AE	11	20	25	71	0.134
5	EF	13	50	62.5	20	0.159
6	FD	8	10	12.5	222	0.098
7	BF	10	20	25	67	0.122
8	EC	8	20	25	59	0.098

Fig. 5-9. Analysis of the subnet of Fig. 5-8 using a mean packet size of 800 bits. The reverse traffic (*BA*, *CB*, etc.) is the same as the forward traffic.

- **$1/\mu = 800$ bits**
- **根据排队论，平均延迟 $T = 1/(\mu C - \lambda)$**

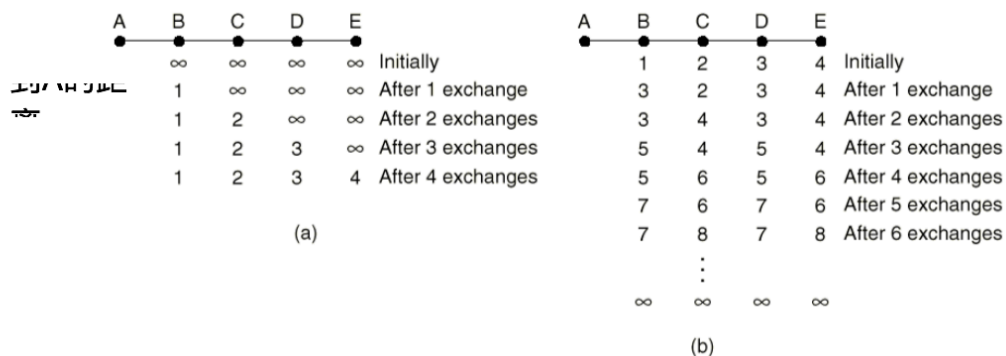
距离向量路由算法 rip arpanet BF算法

在计算中不使用老路由表

有无穷计算问题

解释一下这个图片就是 AB距离不断的在变大 但是要穿很多次才能把E的也变大

- **无穷计算问题 (count-to-infinity problem)**
 - **对好消息反应迅速，对坏消息反应迟钝**



距离响亮算法的 水平分裂->从邻居那里学来的x 不告诉邻居 坏消息就会传播快了（为什么呢 之前A 给b说 我和c很近 然后A和c断掉之后 b却还给A说很近 然后就出问题了）水平分割就是要记录下来 我从那里学到的这个东西 每次我就不给他赘述了

缺点：

收敛慢

没有考虑链路贷款

无穷计算

路由报文开销大（不是增量更新） 因为我对全局的理解都要传给你

Rip最大15跳

链路状态算法

直接测量和邻居之间的关系

每一个包有 序号和年龄 序号用来标示这个包我读过没有（因为有洪范）

age用来防治路由崩溃和序号出错之后 之前的包就不应该要了 所以可以看看他的age是不是过时的 过时的不要了

链路包需要应答

来了一个包 会延迟一会跟其他同样来的比较一下 看看是不是重

复的包 保留新的



例子：路由器 B 中的链路状态信息

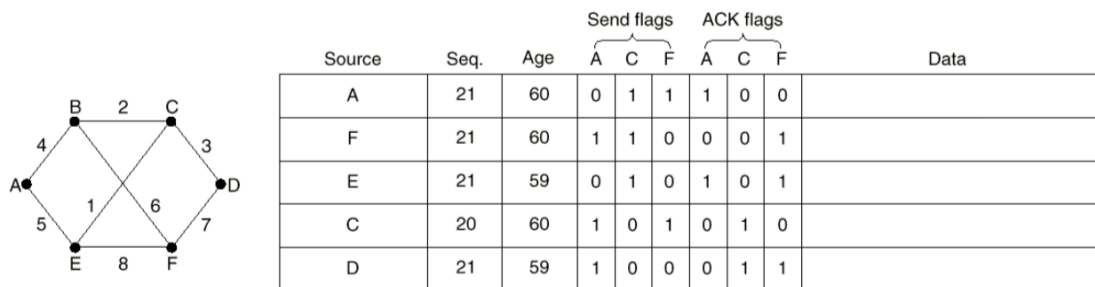


Fig. 5-16. The packet buffer for router B in Fig.5-15.

- 从 E 发来的链路状态分组有两个，一个经过 **EAB**，另一个经过 **EFB**
- 从 D 发链路状态分组有两个，一个经过 **DCB**，另一个经过 **DFB**

32

知道的是整个图 谁连着谁 可以用迪基斯特拉

OSPF ISIS



链路状态算法（LS）和距离向量算法（DV）的比较（续）

■ 收敛（Convergence）速度

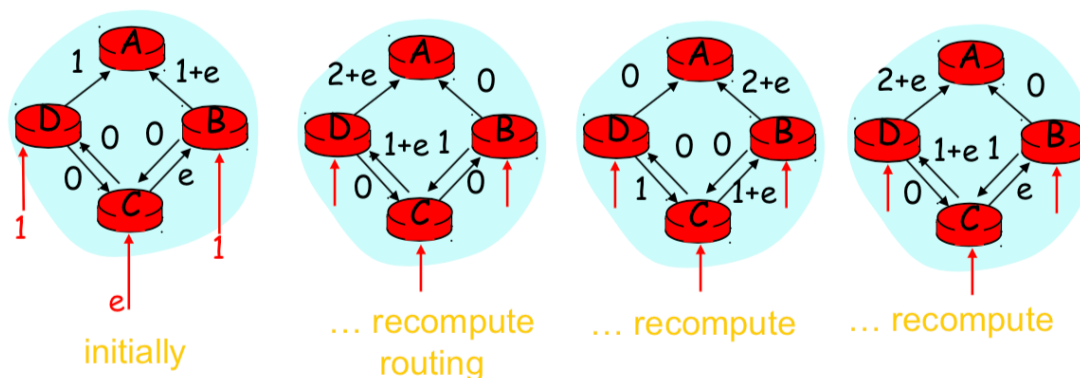
■ LS

- 使用最短路径优先算法，算法复杂度为 $O(n^2)$
 - n 个结点（不分组括源结点），需要 $n(n+1)/2$ 次比较
 - 使用更有效的实现方法，算法复杂度可以达到 $O(n \log n)$
- 可能存在路由振荡（oscillations）

■ DV

- 收敛时间不确定
 - 可能会出现路由循环
 - count-to-infinity 问题

路由振荡



分层路由

移动主机的路由

拥塞控制：

在网络和传输层（tcp）进行控制

流控制和拥塞控制

流：开水放水 发的快 收的慢 局部问题

拥塞控制：全局问题

开环 闭环

开环应该就是绕一绕 不用跑 设计合理的路径

闭环的话就是监控 根据当前的情况

反馈拥塞

开环控制 **流量整形** 拥塞的造成源于突发的网络 设定一种方法 来避免和约束突然的

漏桶算法 不灵活

对于固定的分组长度 变长分组长度都可以（使用字节计数）

令牌桶（攒钱）

Loutong来存放令牌 要传输就要令牌 传完了之后删除这个令牌
-》允许积累发送权利 最大就是桶那么大

流说明：

发之前大家先同意一下，发送方，接收方 第三方。发送方提出请求，子网和接收方作出回复

虚电路的拥塞控制：

准入控制：首先是你发之前有个说明，说明来允不允许发送
发生拥塞之后 你也不能去挤了。或者你去的是不堵的地方 也可

以去

资源预留：主机和子网达成协议之后，子网根据协议来给你留资源

抑制分组：

路由器来监控 监控的是输出电路 如果超过了一个阈值 就把这个输出资源设为报警资源 以后有人想从这里输出的时候就给他说不行 憋往别处输出 少一点

减少的时候比例（指数减少） 增加的时候常熟增加

逐跳抑制分组

如果网络距离长的话 叫源头主机抑制就太慢了 所以对来的路上的每一个路由器都起作用

公平队列：

路由器的每个输出都有多个队列 遍历队列 发送队头 （如果是变长的协议（IP） 以byte为单位

有优先级的队列

负载丢弃：

上面都不能满足才丢

文件wine/多媒体milk 丢的不一样 wine比牛奶珍贵 所以一点都不能洒了

网络互联：

多种网络构成的互联网络

互联的设备：

中继器（物理层） 网桥（数据链路层）（局域网之间 可以改变帧格式） 多协议路由器（网络层） 作用是**储存转发**

传输网关 传输层 应用网关 应用层

无连接网络互联

每个包单独路由 网络利用率++

可以根据子网的类型来做协议转换 分组格式转换 地址转换（各种转换）

隧道技术

（来自课堂笔记（不知啥意思）： 经过翻译的包可能丢失信息 但是tunnel会完整送达）

就是现在局域网里面发 然后通向广域网的路由器 去掉局域网的头和尾 封装到广域网的网络层分组中 目的地地址是 对面的路由器的ip 然后那边再收 然后再局域网里面穿过去

外部网关协议EGP

BGP

分片

分片重组

- 分片重组对其他网络透明。 出口网管要知道什么时候到齐吧 同意网管离开 反复分片 开销大
- 不透明 要目的的主机来把这些组合起来

分片的例子

- 树型标记法
 - 分成n段 每段长度一致
- 偏移量法

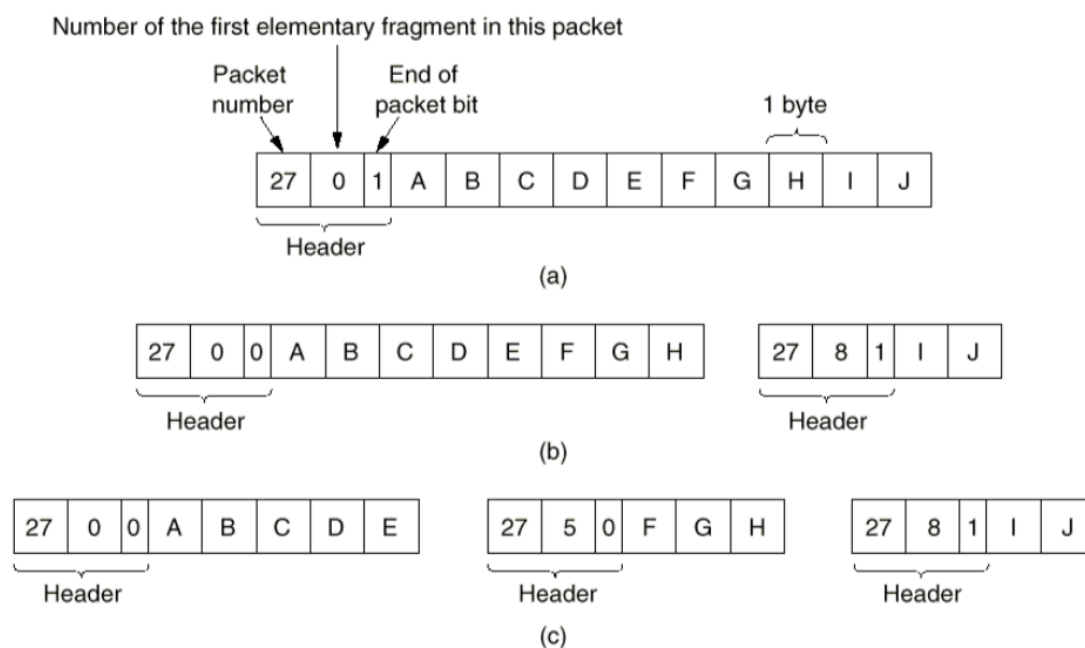


Fig. 5-42. Fragmentation when the elementary data size is 1 byte. (a) Original packet, containing 10 data bytes. (b) Fragments after passing through a network with maximum packet size of 8 bytes. (c) Fragments after passing through a size 5 gateway.

防火墙

ip协议

ip头

生存期: ttl 是0 就丢弃 还给源主机一个告警

协议域: 上层是啥 TCP UDP

头校验和

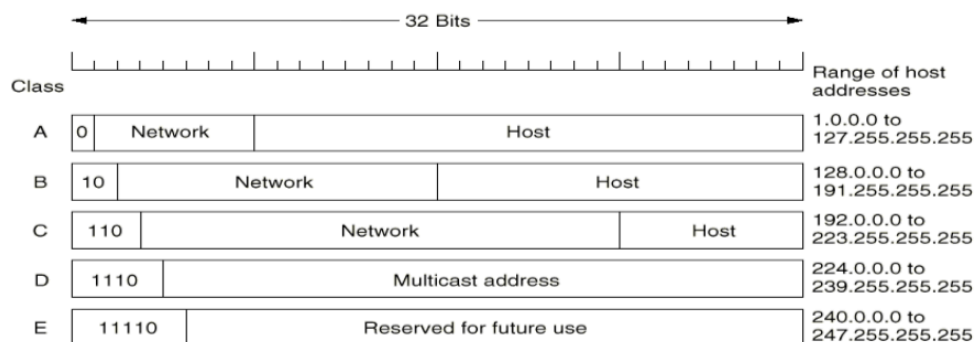
只对ip分组头做校验 每16位取反 循环相加 和求反

源地zhi 目的地址

IP地址

有类地址划分

■ 有类地址划分 (classful addressing)



全0 全1

协议们：

- icmp
- arp 地址解析协议：ip 和 Mac 映射（如果没有在arp表里面找到 会发送一个广播 目的收到之后会应答）
 - arp攻击 伪造arp响应包 存在于局域网
 - rarp 反过来 用来启动
-
- rip 采用跳来衡量距离 poison reverse**
 - 超时处理 经过他的就凉了 然后新的路由表给其他人说 邻居也会更新
-
- ospf 还是内部网关协议 之前说的链路状态算法

- 分区域 有一个主干区域 所有人都和主干项链 其余不相连 经过主干来进行跨区域的对话
- 内部路由器
- 跨区域路由器
- 主干路由器
- 跨自治系统路由器

•

BGP 边界网关协议

分层路由 减少路由表的大小

使用tcp

pv算法 () 类似距离向量

CIDR 无类域间路由

ipv4分完了

单播地址都采用cidr

就是先把c类地址分了 然后用掩码去最长前缀匹配

最长前缀匹配：有掩码的位置必须全部对 最后选择掩码最长的（匹配的最多的）

ipv6

128位

没有分片域 主机自己做分片

