

# Chapter 6

## Medium Access Control protocols and Local Area Networks

### 介质访问控制协议和局域网

---

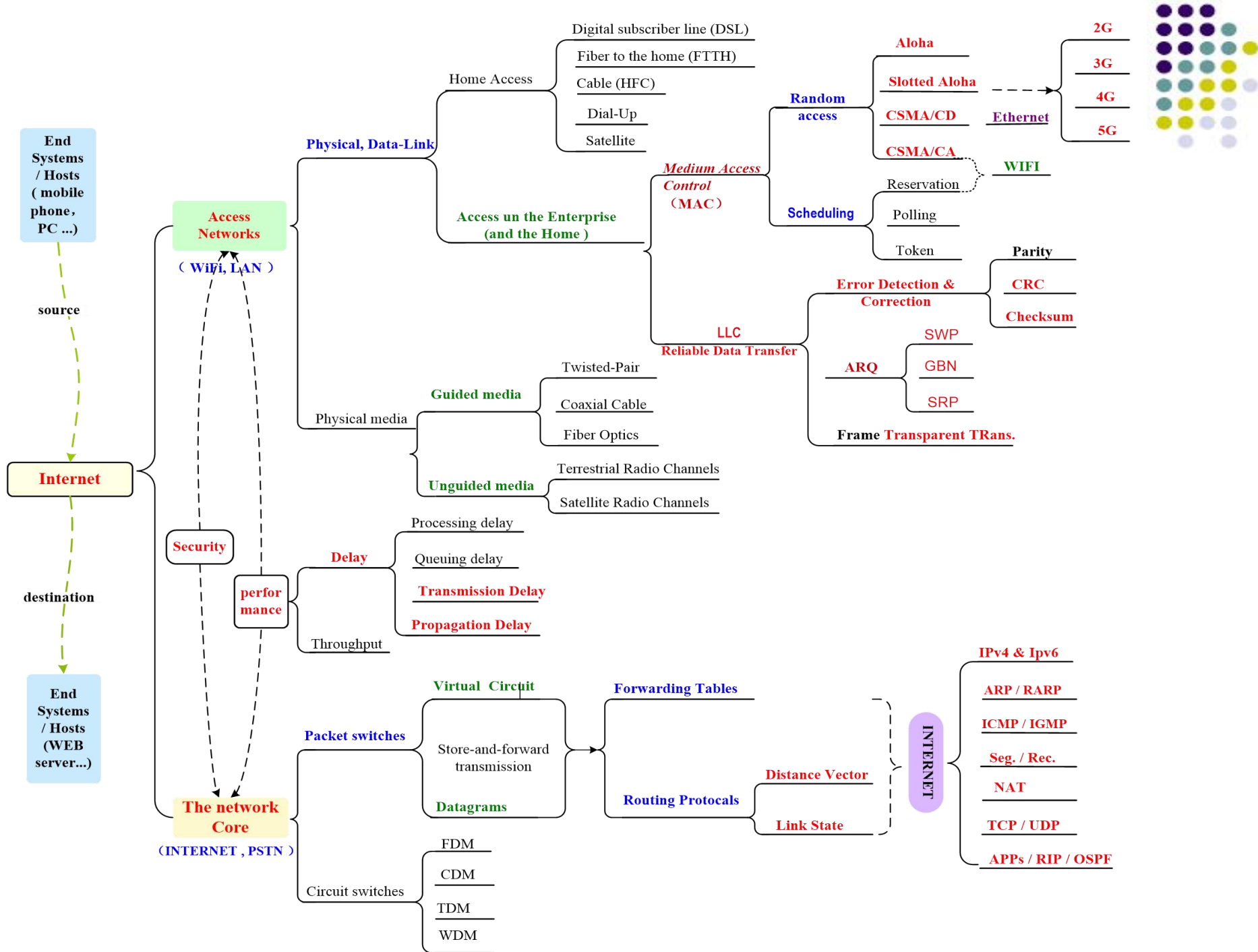
*Part I: Medium Access Control*

*介质访问控制协议*

*Part II: Local Area Networks*

*局域网*



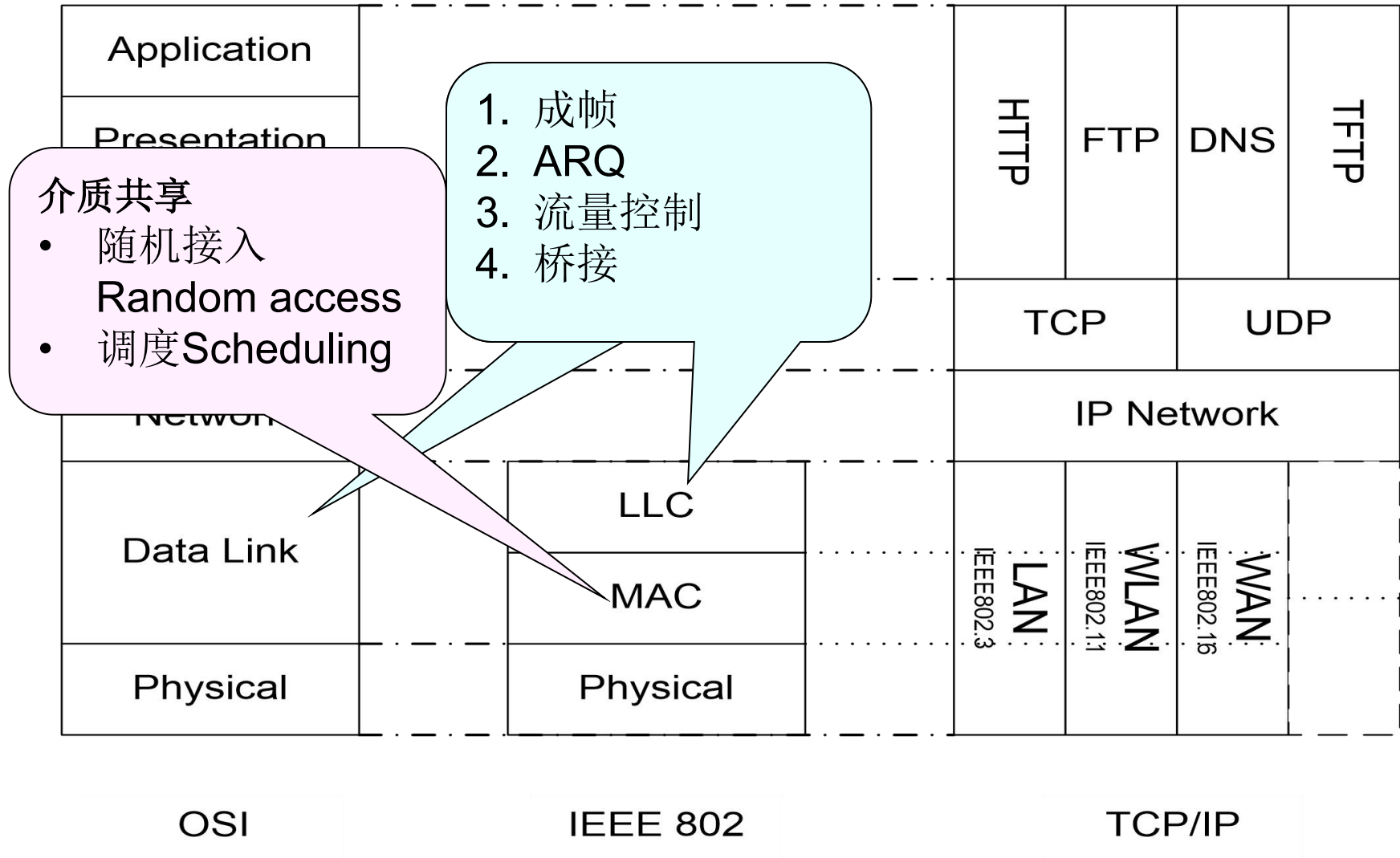


# Chapter Overview



- **广播网络（Broadcast Networks）**
  - 所有信息发给所有用户
  - 无路由
  - 共享介质
  - 无线
    - 蜂窝电话通讯（Cellular telephony）
    - 无线局域网Wireless LANs
  - 铜线 & 光
    - 以太网（有线局域网）
    - 电缆调制解调器接入（Cable Modem Access）
- **介质访问控制（Medium Access Control）**
  - 协调对共享介质的接入
  - 直接传输帧，属于数据链路层
- **局域网（Local Area Networks）**
  - 临近计算机之间的高速、低成本通信方案
  - 通常基于广播网络
  - 简单便宜
  - 支持用户数有限

# Chapter Overview (cont.)



# Chapter 6

## Medium Access Control protocols and Local Area Networks



### Part I: 介质访问控制 Medium Access Control

6.1 多址通信 Multiple Access Communications

6.2 随机接入 Random Access

调度 Scheduling

信道化 Channelization



# Chapter 6

## Medium Access

### Control protocols and

### Local Area Networks



#### Part II: 局域网Local Area Networks

6.6 局域网概览Overview of LANs

6.7 以太网Ethernet和IEEE 802.3

令牌环Token Ring和FDDI

802.11 无线局域网

6.8 局域网桥接



# Chapter 6

## Medium Access Control protocols and Local Area Networks

---



多址通信

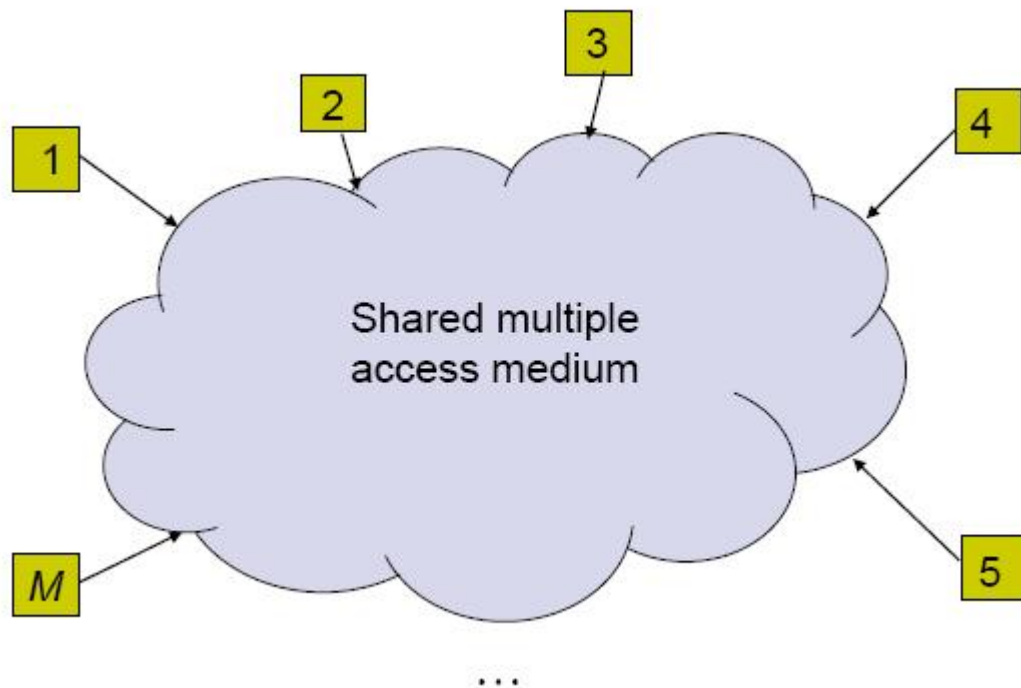
*Multiple Access Communications*



# 多址通信



- 以共享介质为基础的广播网络
  - 成本低：空中无线电；铜线或同轴电缆
  - $M$  个用户通过向介质广播实现通信
- 关键问题：怎样共享介质？





# 介质共享方式



每个站产生固定  
数据流

## 介质共享技术

突发业务

### 静态信道化

#### 频分多址 & 时分多址

- 资源划分
- 用户专用分配
- 卫星传输
- 蜂窝电话

### 动态介质访问控制

#### 调度

- **轮询**：按序发送
- **预约**传输时隙
- **令牌环Token ring**
- 无线局域网

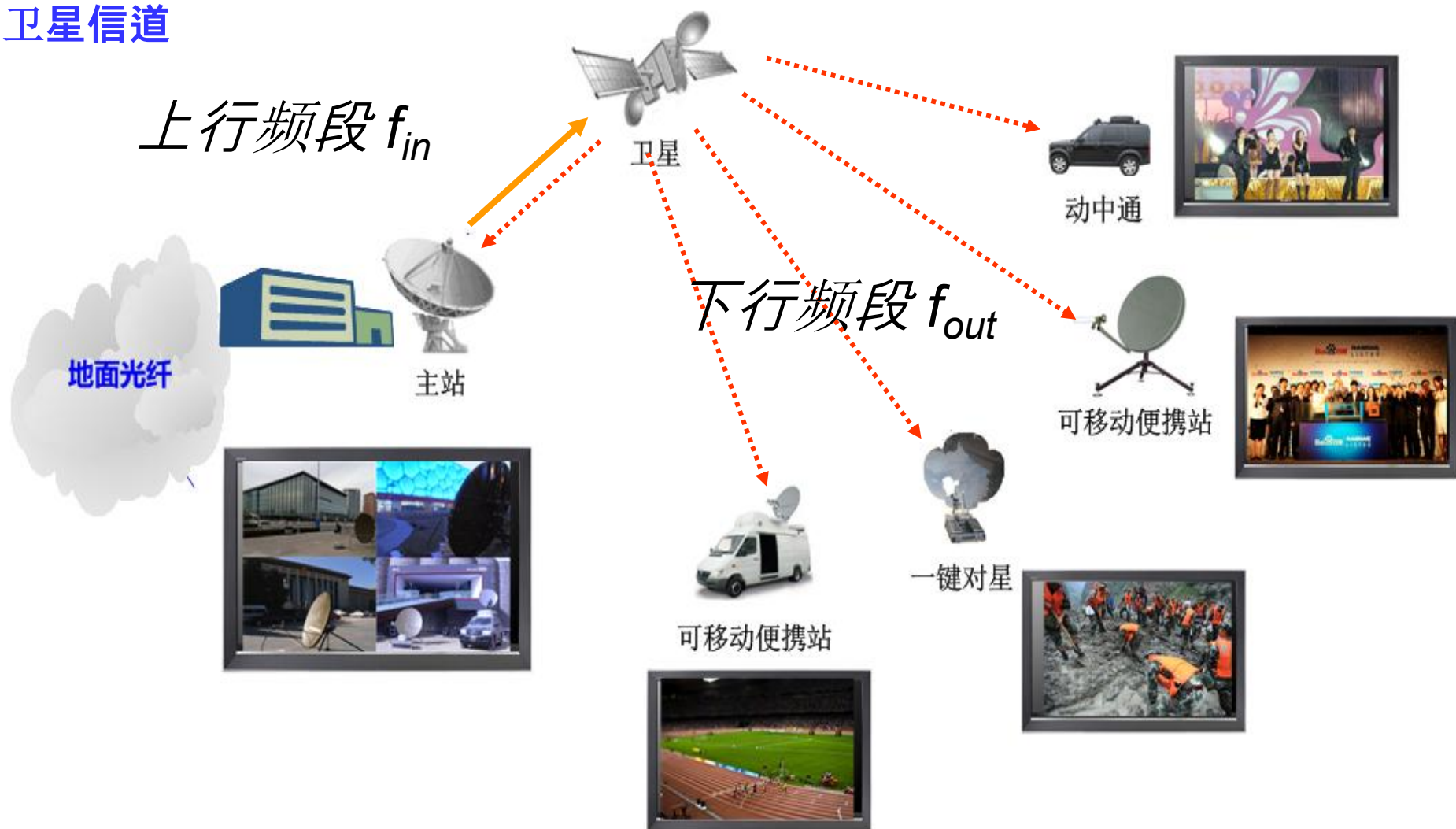
#### 随机接入

- 非严格同步Loose coordination
- 发送，等待，如果需要就重传
- **ALOHA**
- **CSMA**

# 信道化：卫星通信



## 卫星信道



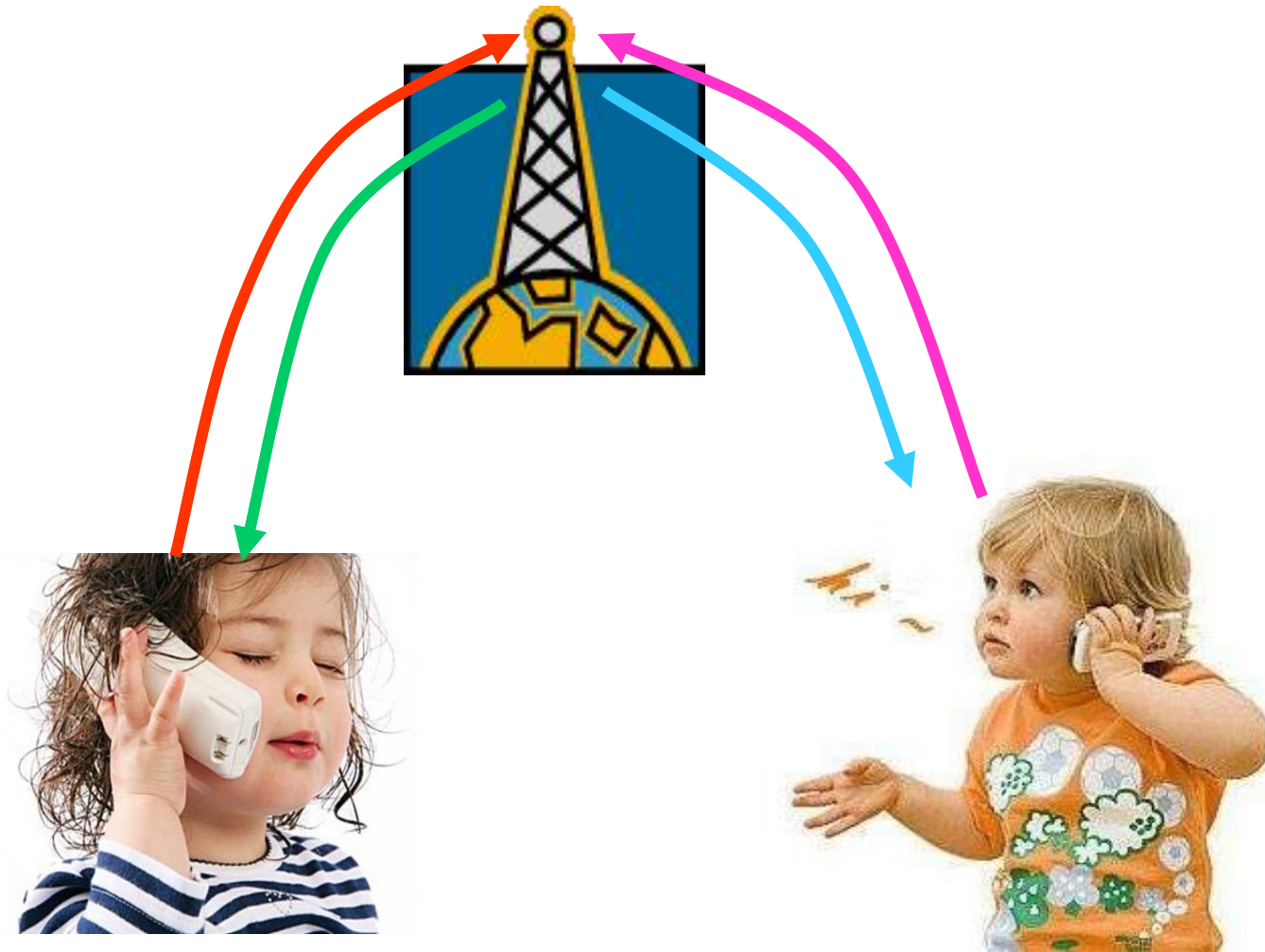
现场卫星通信网络示意图

# 信道化：蜂窝网

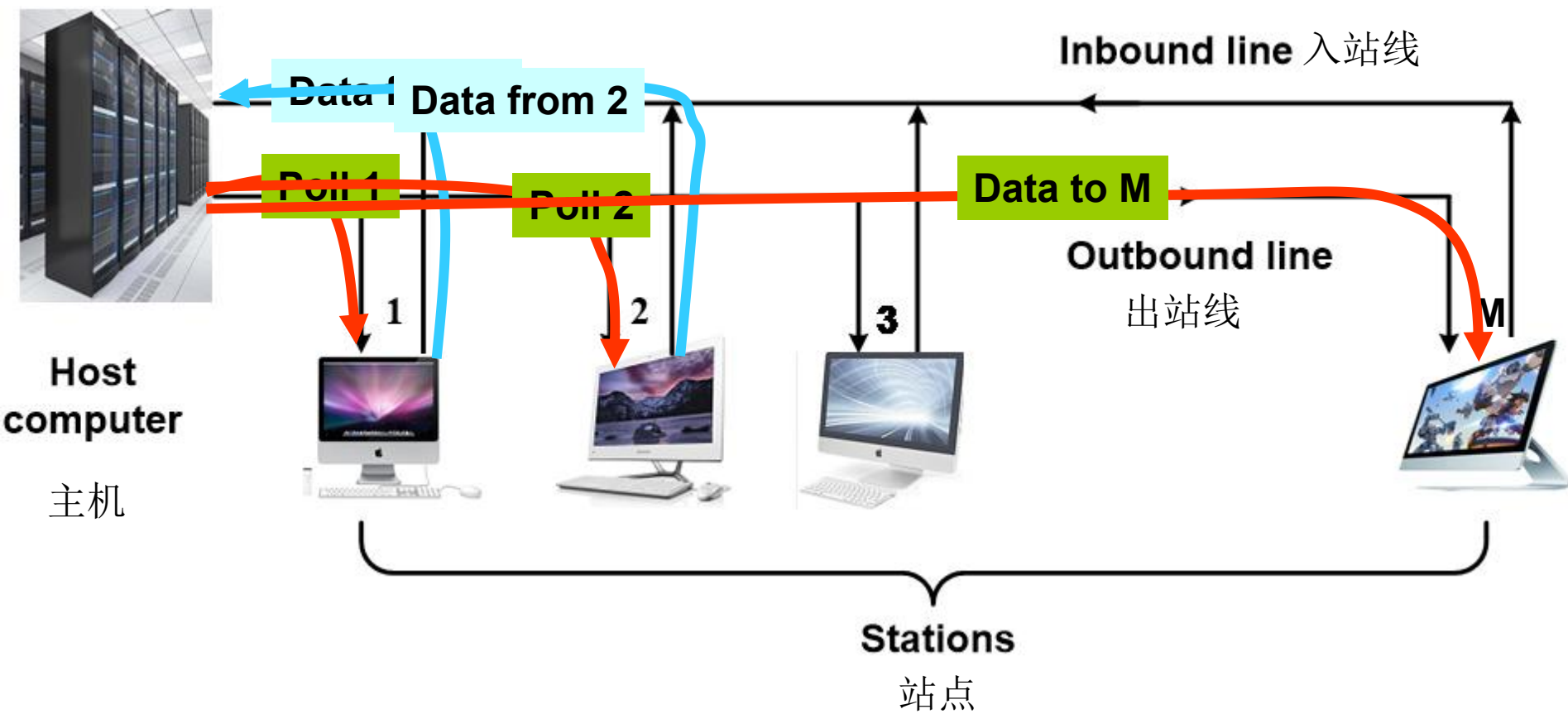


上行  $f_1$  ; 下行  $f_2$

上行  $f_3$  ; 下行  $f_4$



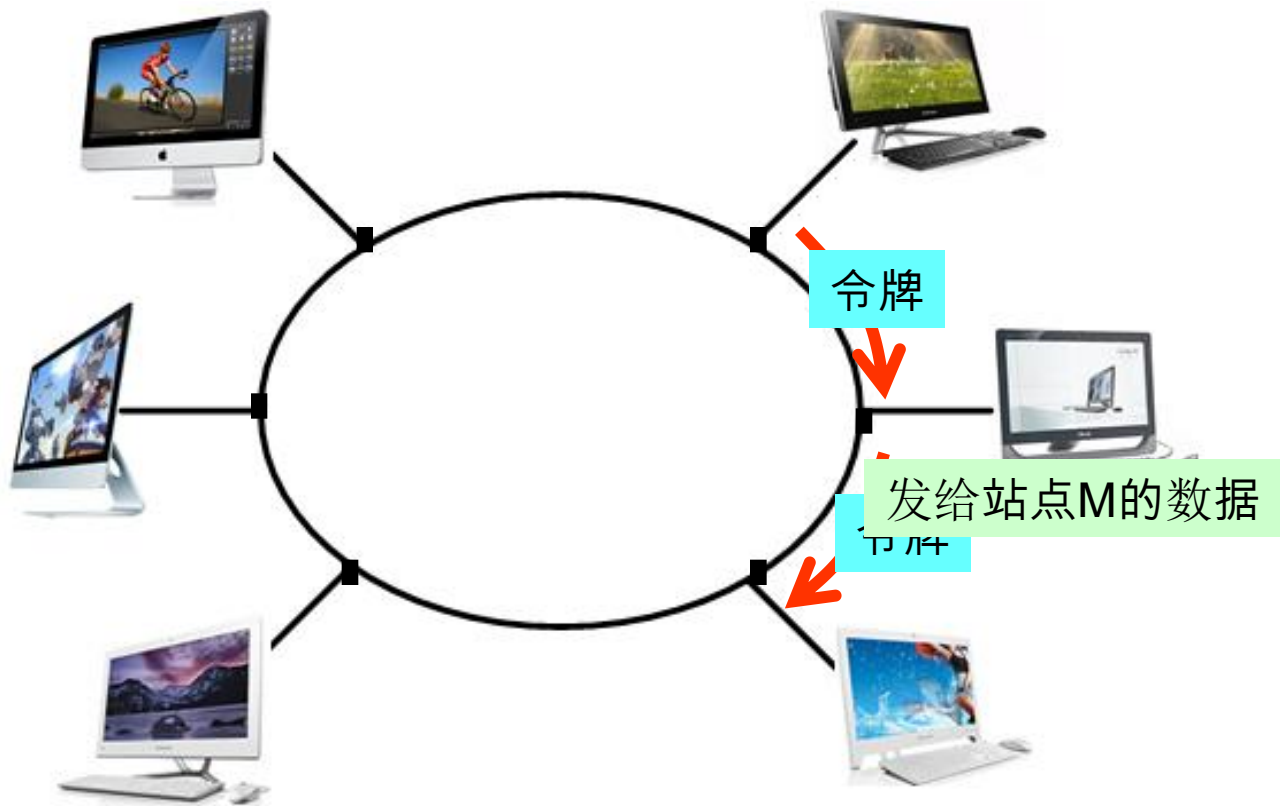
# 调度：轮询



# 调度：令牌传递Token-Passing



## 环形网络

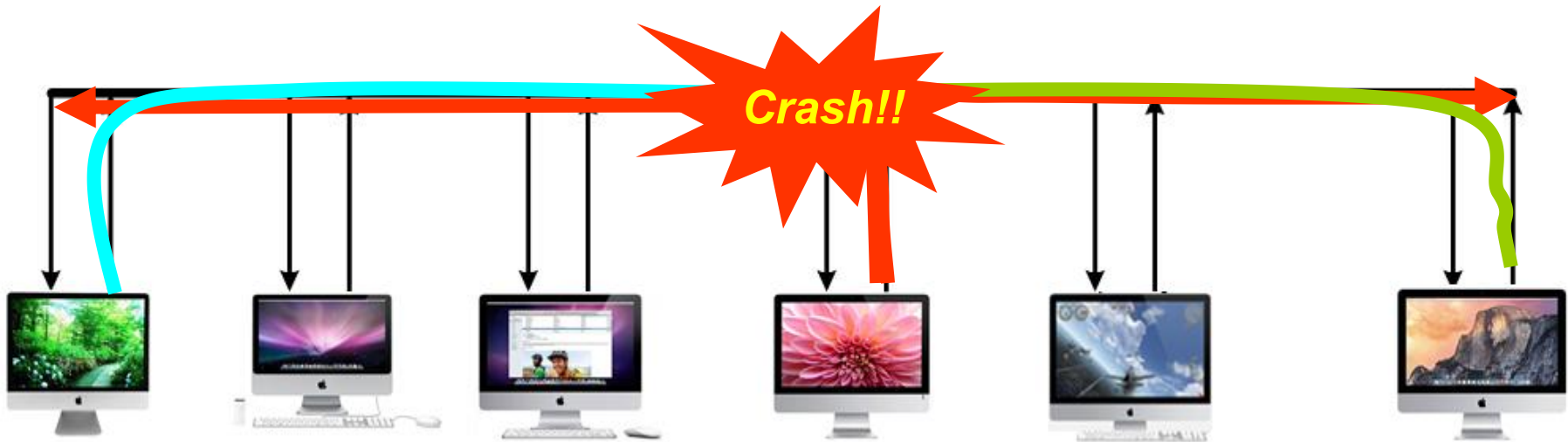


拿到令牌的站点向环形网络发送数据

# 随机接入



## 多接头总线

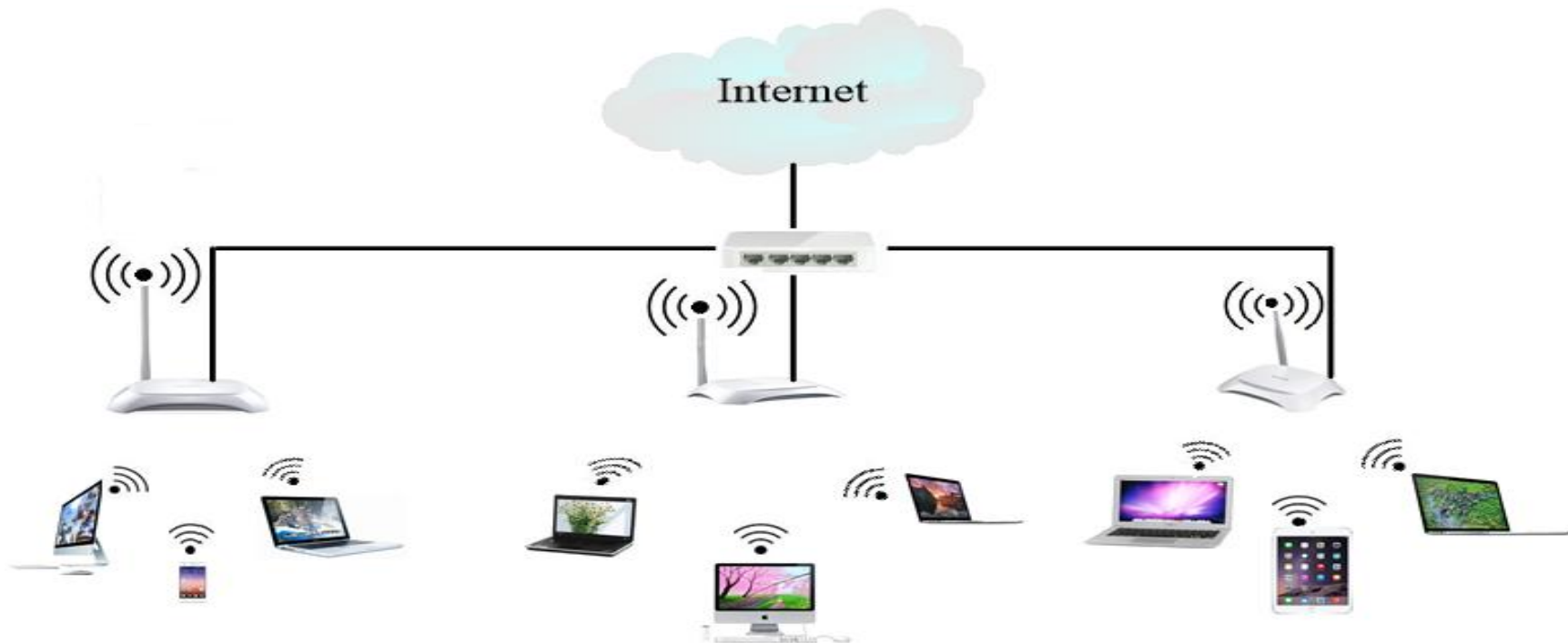


有数据就发

可能发生传输碰撞；需要重传机制

# 无线局域网

- 自组织网络Ad Hoc: 站点到站点
- 基础设施网络: 站点到热点（接入点）
- 随机接入 & 轮询





# 选择一种介质访问控制

---



- 应用Applications

- 什么类型的业务？
- 语音流？ 稳定业务，低时延/抖动
- 数据？ 短信？ 网页下载？
- 企业还是消费者市场？ 可靠性, 成本

- 扩展性Scale

- 能够承载多少业务？
- 能够支持多少用户？

- 举例Current Examples:

- 设计MAC，使其向郊区提供无线DSL等效的接入服务
- 设计MAC，使其向移动用户（130km/hr）提供无线局域网等效的接入服务



# 时延带宽积

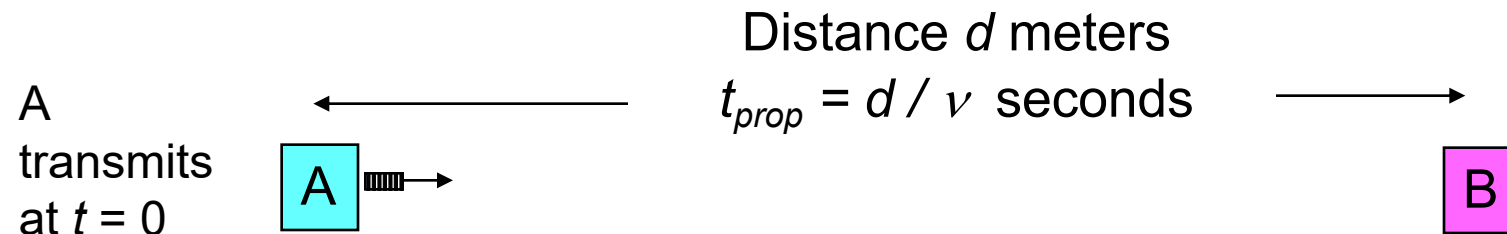


- **时延-带宽 (*Delay-bandwidth*) 积是一个关键参数**
  - 共享介质的协调，显性或隐性使用带宽
  - Difficulty of coordination commensurate with delay-bandwidth product
- 两站点的简单例子
  - 有帧要发的站点侦听信道，如果空闲就发送
  - 站点发送时仍然侦听信道
  - 如果检测到冲突发生，重新发送

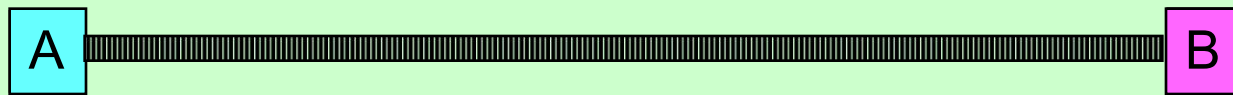
# 两站点MAC例子



两个站点试图共享同一个介质



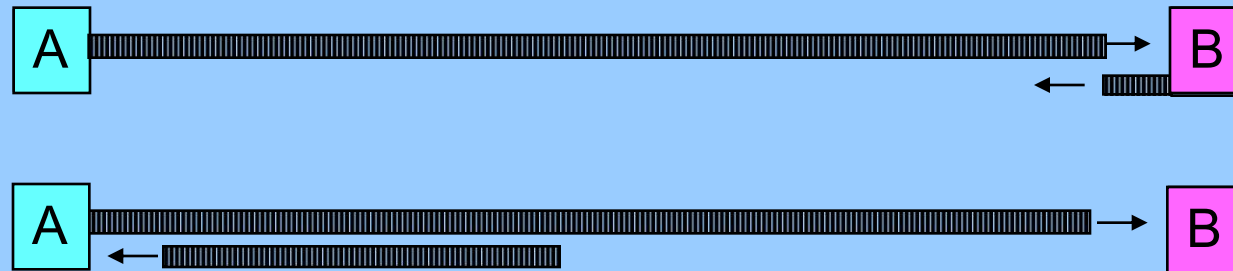
Case 1



B does not transmit before  $t = t_{prop}$  & A captures channel

Case 2

A detects collision at  $t = 2 t_{prop}$



B transmits before  $t = t_{prop}$  and detects collision soon thereafter



## 两站点例子的效率

- 每个帧得以发送需要  $2t_{prop}$  安静时间
  - 站点 B 在站点 A 发送前、后需要  $t_{prop}$  时间保持安静
- $R$ : 发送比特速率
- $L$ :  $L$  比特/帧

$$\text{Efficiency} = \rho_{\max} = \frac{L}{L + 2t_{prop}R} = \frac{1}{1 + 2t_{prop}R/L} = \frac{1}{1 + 2a}$$

$$\text{MaxThroughput} = R_{\text{eff}} = \frac{L}{L/R + 2t_{prop}} = \frac{1}{1 + 2a} R \text{ bits/second}$$

归一化时延带宽积

$$a = \frac{t_{prop}}{L/R}$$

← 传输时延

← 发送一个帧的时间

# 典型的MAC效率



- 两站点例子:

$$Efficiency = \frac{1}{1 + 2a}$$

- CSMA-CD (以太网) 协议:

$$Efficiency = \frac{1}{1 + 6.44a}$$

- 令牌环网:

$$Efficiency = \frac{1}{1 + a'}$$

$a'$  = 环延迟(比特)/平均帧长

- 如果  $a \ll 1$ , 效率接近100%
- 当  $a$  接近 1 时, 效率很低

# Chapter 6

## 介质访问控制协议 和局域网



*随机接入Random Access*



## 6.2.1 随机接入技术: ALOHA



- ALOHA: Additional Link On-line HAWAII system

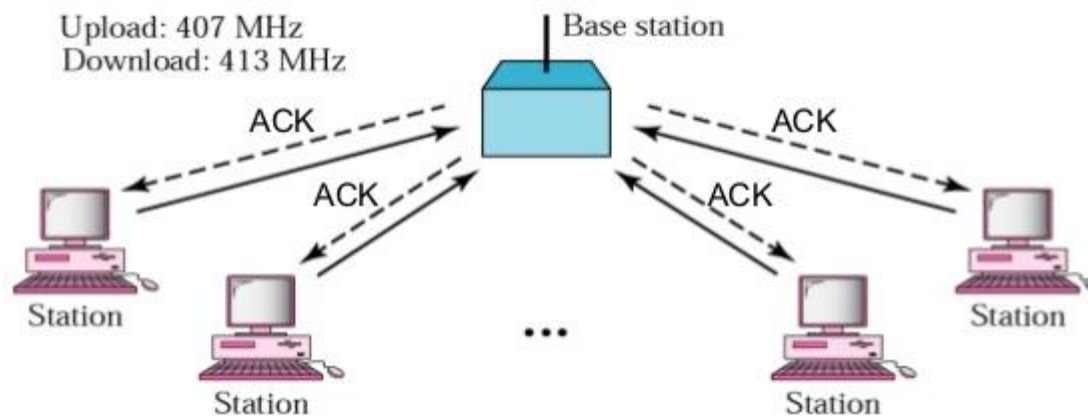


- 动机: 上世纪70年代早期, 为夏威夷大学主校区和分校区间提供无线链路

# 随机接入技术: ALOHA (cont.)



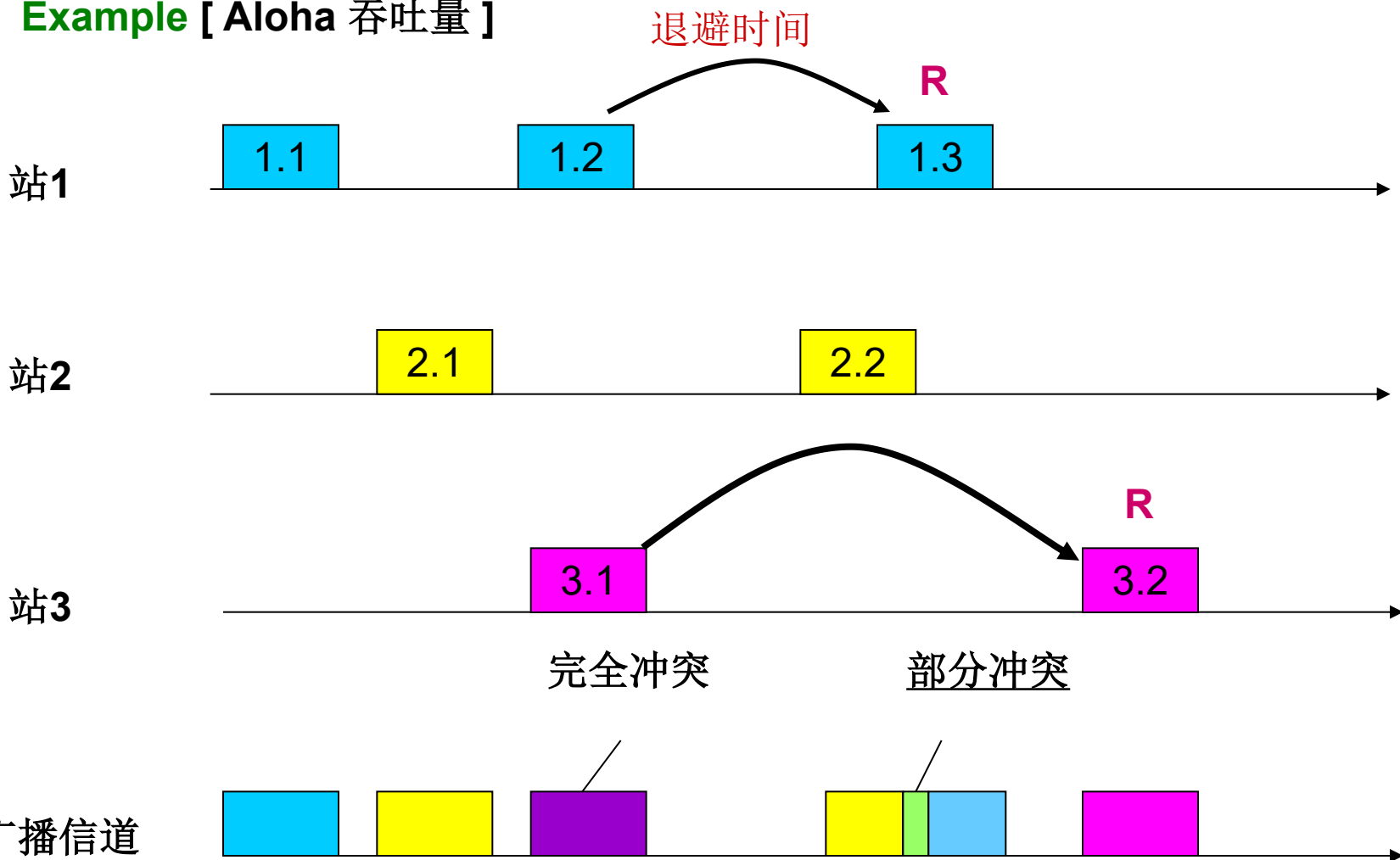
- 最简单的方案: 有就发
  - 站点有帧就发送
  - 如果多于一个帧被发送, 它们将冲突并丢失
  - 如果未能在超时时间内收到**ACK**, ( $2 * \text{propagation delay}$ ), 则发送了帧的站点随机选择一个退避时间 (**backoff time**) (来避免重复冲突) 并在该退避时间结束后重新发送帧



# 随机接入技术: ALOHA (cont.)



## Example [ Aloha 吞吐量 ]





# 随机接入技术: ALOHA (cont.)



- 性能指标
  - 吞吐量
  - 时延
- 与吞吐量分析有关的参数
  - $X$ : 帧发送时间(假设常数)
  - $S$ : 吞吐量 (每 $X$ 秒内平均发送的新帧的个数)
  - $G$ : 负载 (每 $X$ 秒内平均发送的所有帧的个数)
  - $P_{success}$ : 一个帧成功发送的概率

$$S = G P_{success}$$

# 随机接入技术: ALOHA (cont.)



- 帧到达的概率，近似于泊松分布

- Poisson 泊松分布：X时间内k个到达

- $\mathcal{P}[k \text{ frames submitted}] = \frac{G^k \cdot e^{-G}}{k!}$

- 则：

- Poisson 泊松分布：2X时间内k个到达

$$P[k \text{ arrivals in } 2X \text{ seconds}] = \frac{(2G)^k}{k!} e^{-2G}, k = 0, 1, 2, \dots$$

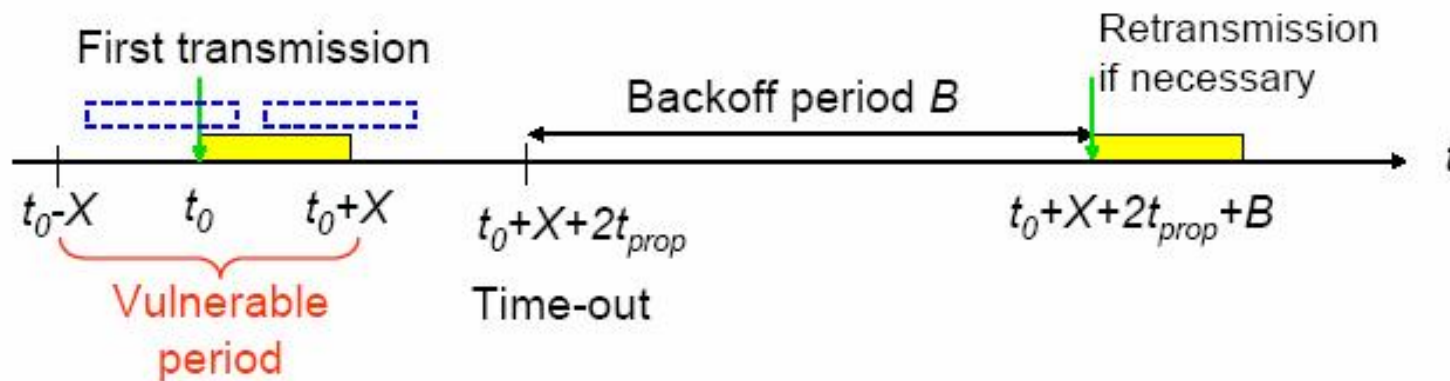
# 随机接入技术: ALOHA (cont.)



## 易损期

- 假设帧长 $L$ 和发送时间 $X=L/R$ 是常数
- 考虑一个从 $t_0$ 开始发送的帧，该帧成功传输的条件是没有与其它帧冲突
  - 任何在 $[t_0-X, t_0+X]$ 内发送的其它帧将导致冲突

**易损期vulnerable period =  $[t_0 - X, t_0 + X]$**



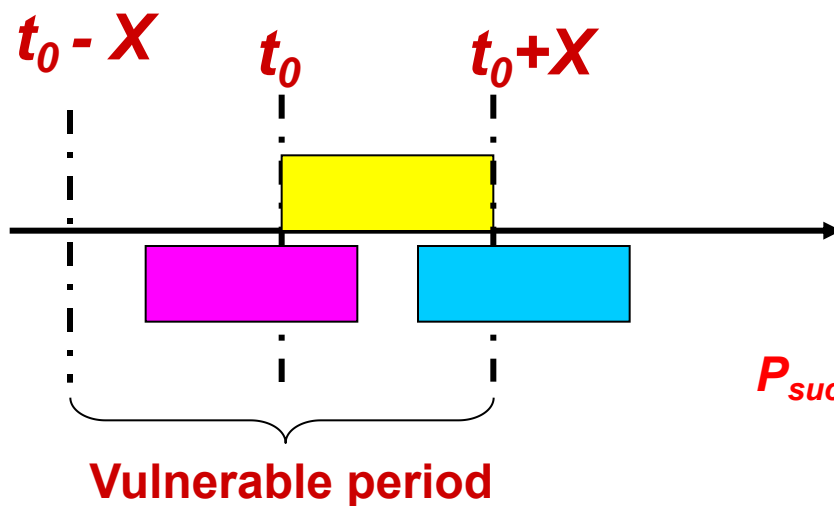
在易损期内，没有其它帧传输，即不冲突的概率是多少？

# 随机接入技术: ALOHA (cont.)



- 任何在易损期内开始发送的帧将导致冲突
- $2X$ 时间内没有其它帧发送, 该帧将得以成功传输

$$P[k \text{ arrivals in } 2X \text{ seconds}] = \frac{(2G)^k}{k!} e^{-2G}, k = 0, 1, 2, \dots$$



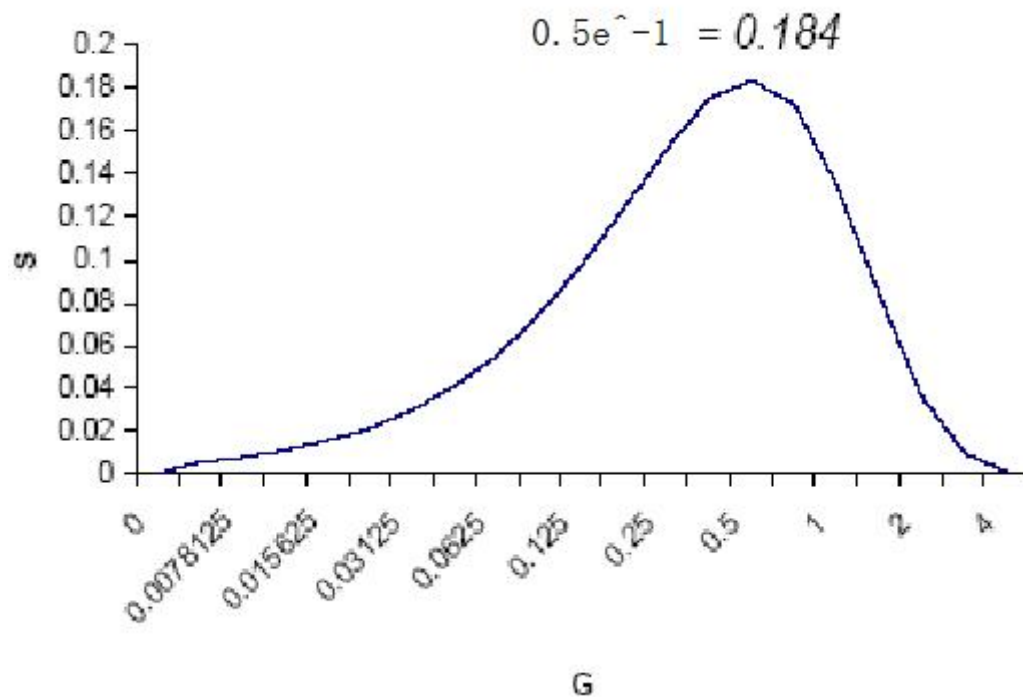
$$S = G P_{succ}$$

$$P_{succ} = P[0 \text{ arrivals in } 2X \text{ secs}] = e^{-2G}$$

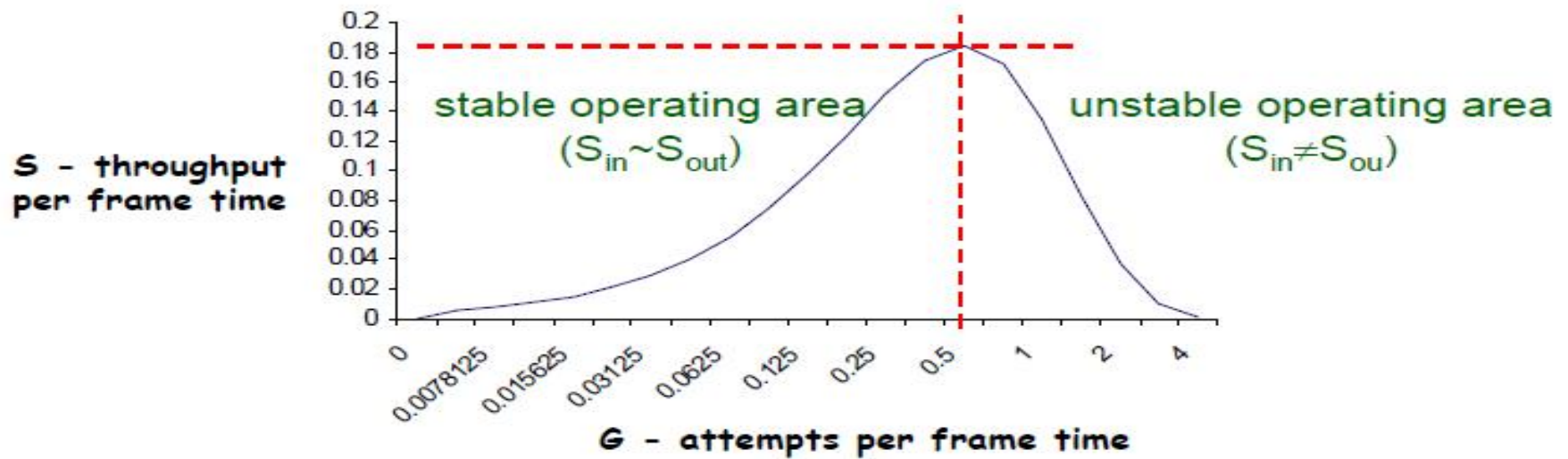
# 随机接入技术: ALOHA (cont.)



$$S = GP_{succ} = Ge^{-2G}$$



- 最大吞吐量  $\rho_{\max} = 1/2e$  (18.4%)
- 很小的  $G$ ,  $S \approx G$   
很大的  $G$ ,  $S \downarrow 0$
- $G$  很大时, 冲突会引起滚雪球导致吞吐量迅速下降至 0



- **注意:** 以上分析假设很多站点共享一个信道，站点的发送速率相近 (如果只有1个站点， $S=1$ )
- 最开始,  $S$  随  $G$  增加直到到达  $S_{max}$ 
  - 之后，网络进入不稳定工作区间，即更多冲突会发生，重传的站点增多 ( $S_{in} > S_{out}$ )
- **ALOHA 最大吞吐量 ( $S_{max} = 0.184$ ) 发生在  $G=0.5$ , 这对应于 “一个易损期内一个帧到达” 的到达速率**
- $S_{max} = 0.184 \Rightarrow$  **ALOHA 最大吞吐量 = 18.4% 信道容量**

## 2、随机接入技术: 时隙ALOHA



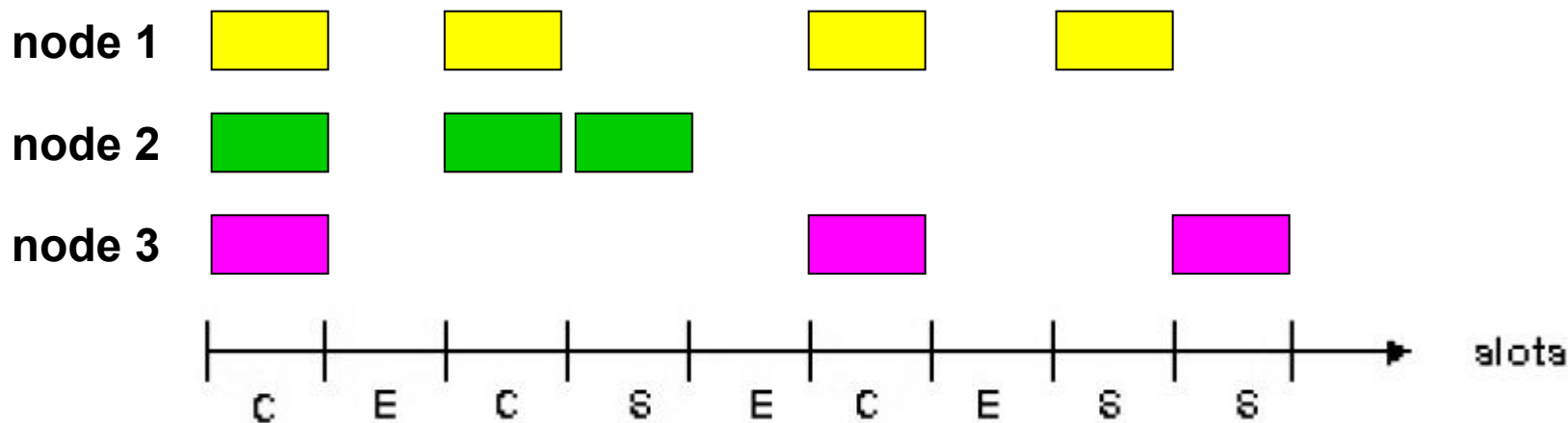
时隙ALOHA – “改进的 ALOHA”, 冲突概率降低

假设:

- 时间被划分为 $X=L/R$ 的时隙 (一帧时间)
- 站点只能在时隙的开始发送帧
- 站点间是同步的,

操作:

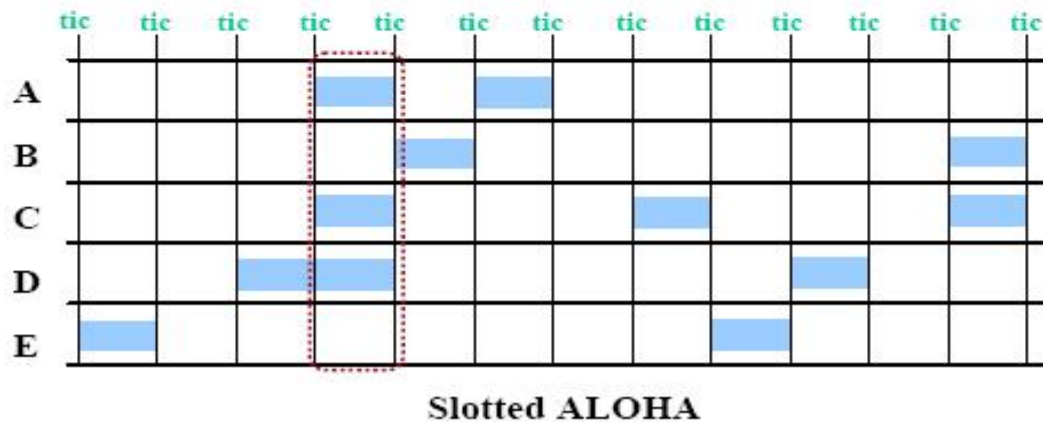
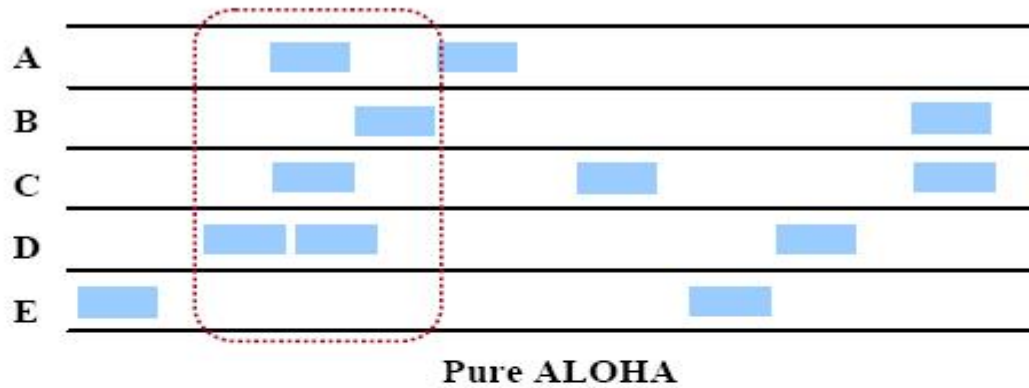
- 1) 当一个站点有新帧要发, 它等到下一个时隙开始的时刻发送
- 2) 如果发生冲突, 站点等待一个退避时间后重新发送该帧 (退避时间 = 帧长的整数倍)



# 随机接入技术: 时隙ALOHA(cont.)



## Example [ Aloha vs. Slotted Aloha ]

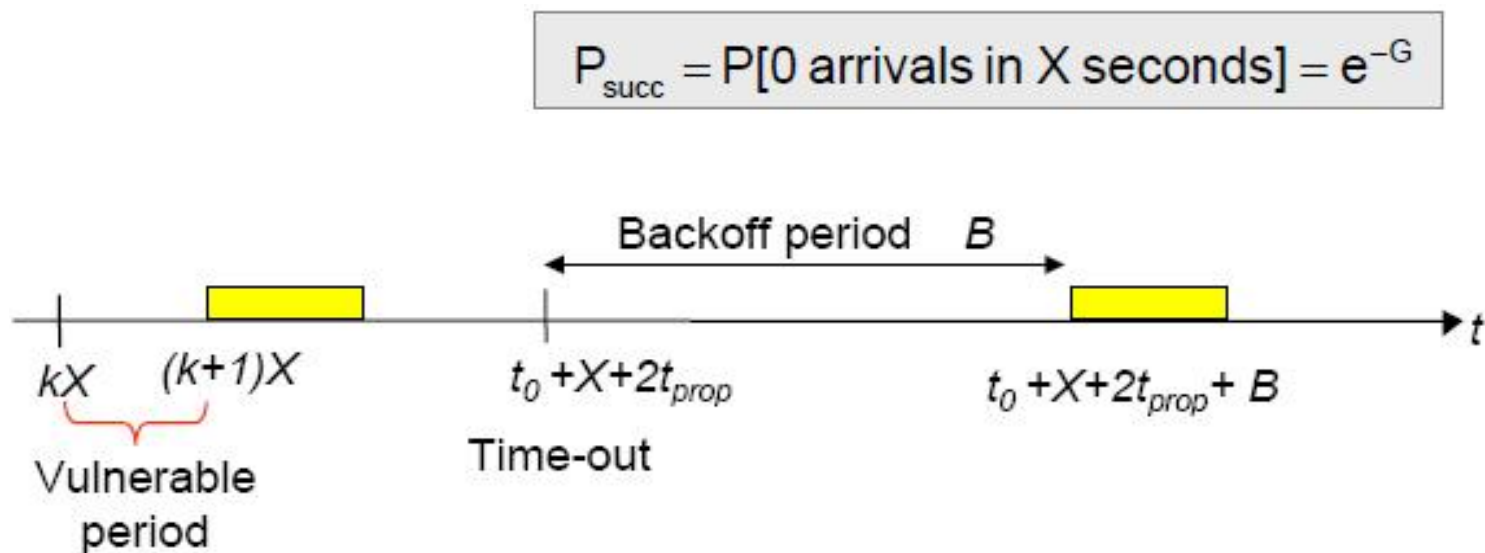




## 随机接入技术: 时隙ALOHA(cont.)



- 考虑在时隙 $[k, k+1]$ 内某个准备完毕的任意帧P
  - 帧P得以成功发送的条件是，同一个时隙内，没有其它帧准备完毕
- 易损期=  $[t_0 - X, t_0]$



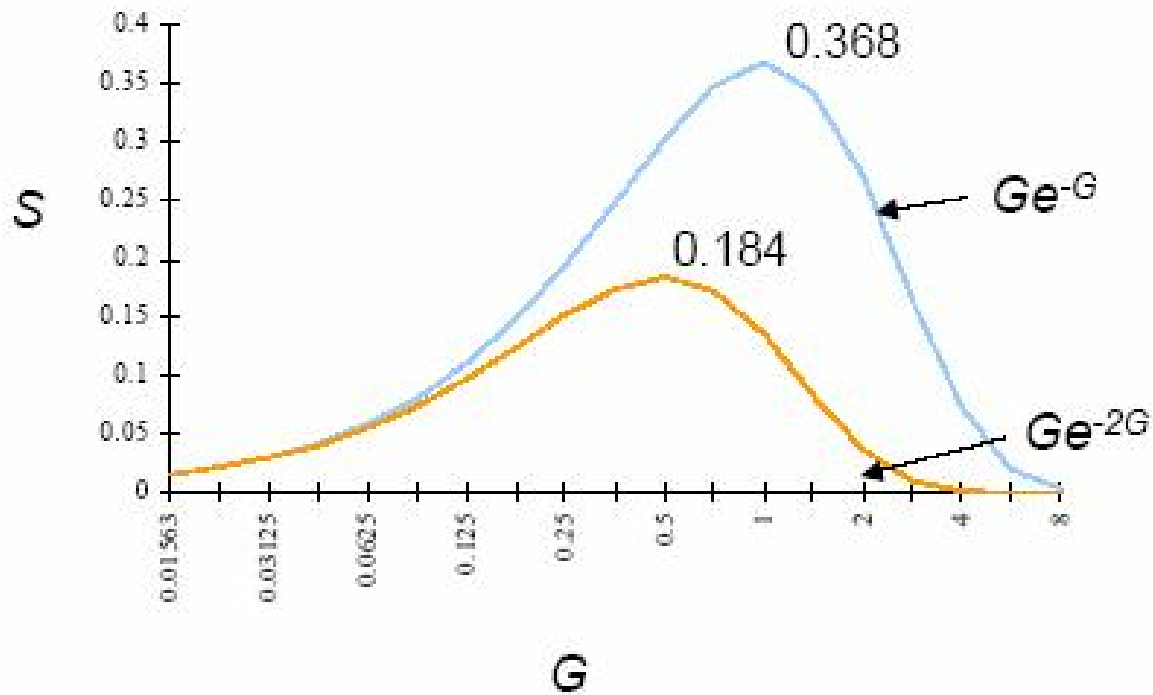
只有在发送前X秒内有到达的帧才会引起冲突

# 随机接入技术: 时隙ALOHA(cont.)



$$S = GP_{\text{succ}} = Ge^{-G}$$

- **时隙ALOHA最大吞吐量 ( $S_{\text{max}} = 0.368$ )** 发生在 $G=1$ , 对应于‘一个易损期内有一个帧到达’的到达速率
- $S_{\text{max}} = 0.368 \Rightarrow$  **时隙ALOHA 最大吞吐量 = 36.8% 信道容量**

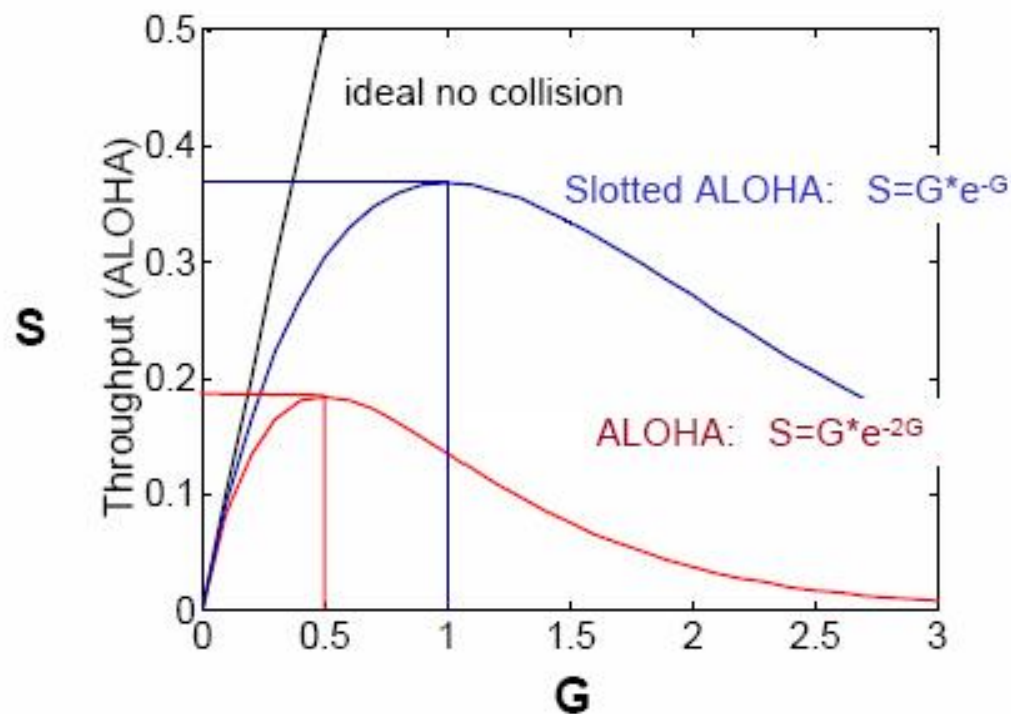


# 随机接入技术: 时隙ALOHA(cont.)



## 时隙 ALOHA vs. 纯 ALOHA

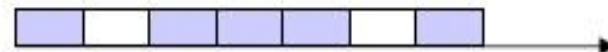
- 时隙 ALOHA 降低了冲突的概率, 但增加了一个等待时间
- 如果冲突率很低, 它会避免很少的冲突, 并对大部分帧增加等待时间



## Example [ slotted Aloha ]

Measurements of slotted ALOHA channel with an infinite number of users show that 10% of the slots are idle.

- a) What is the channel load,  $G$ ?
- b) What is the throughput,  $S$ ?
- c) Is the channel underloaded or overloaded?

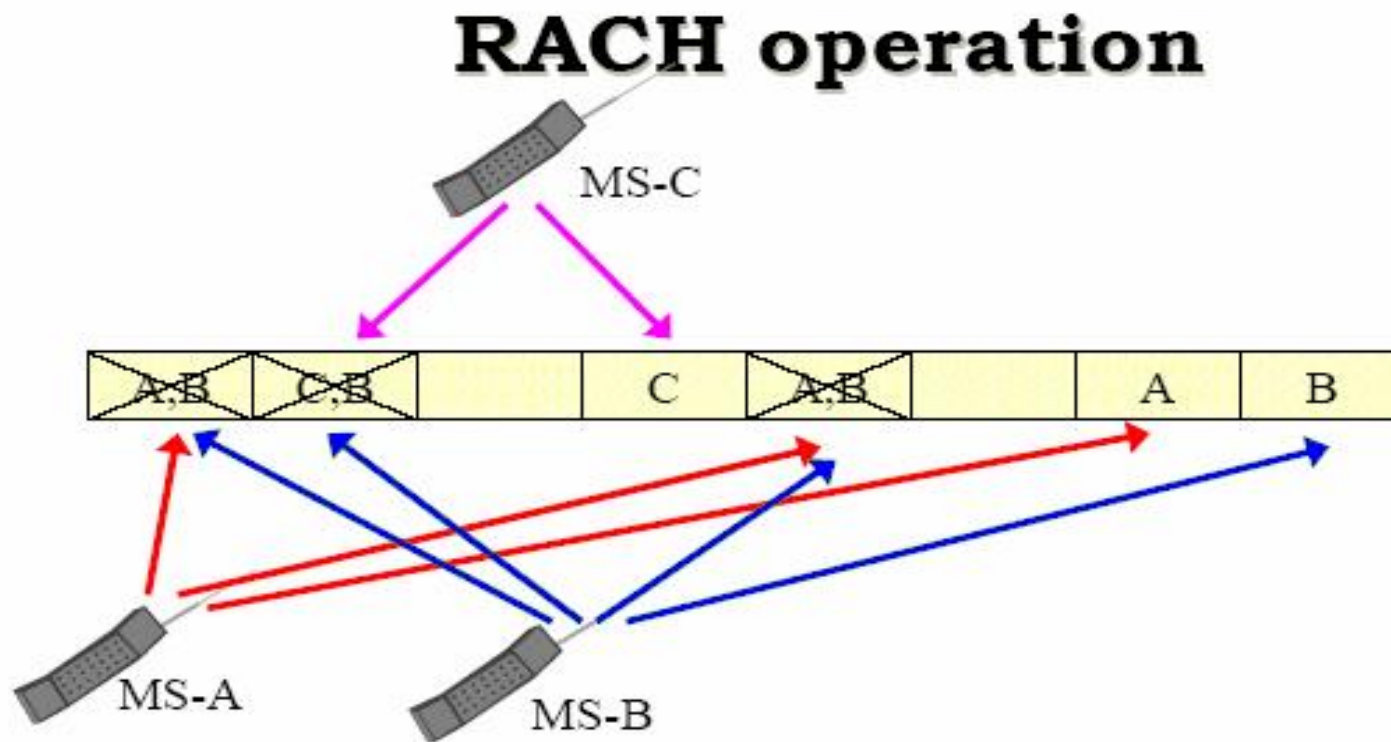


- 
- a) 10% of slots idle  $\Rightarrow$   
frame will be successfully transmitted if sent in those 10% of slots  $\Rightarrow$   
 $P_{\text{succ}} = 0.1$   
According to theory,  $P_{\text{succ}} = e^{-G} \Rightarrow G = -\ln(P_{\text{succ}}) = -\ln(0.1) = 2.3$
  - b) According to theory,  $S = P_{\text{succ}} * G = G * e^{-G}$   
as  $G=2.3$  and  $e^{-G}=0.1 \Rightarrow S = 0.23$
  - c) Whenever  $G > 1$ , the channel is overloaded, so it is overloaded in this case.

# 随机接入技术: 时隙ALOHA(cont.)



## Example [ 蜂窝系统 (GSM) 中的时隙Aloha]



Multiple Access Technique for simultaneous access

Collision resolution based on

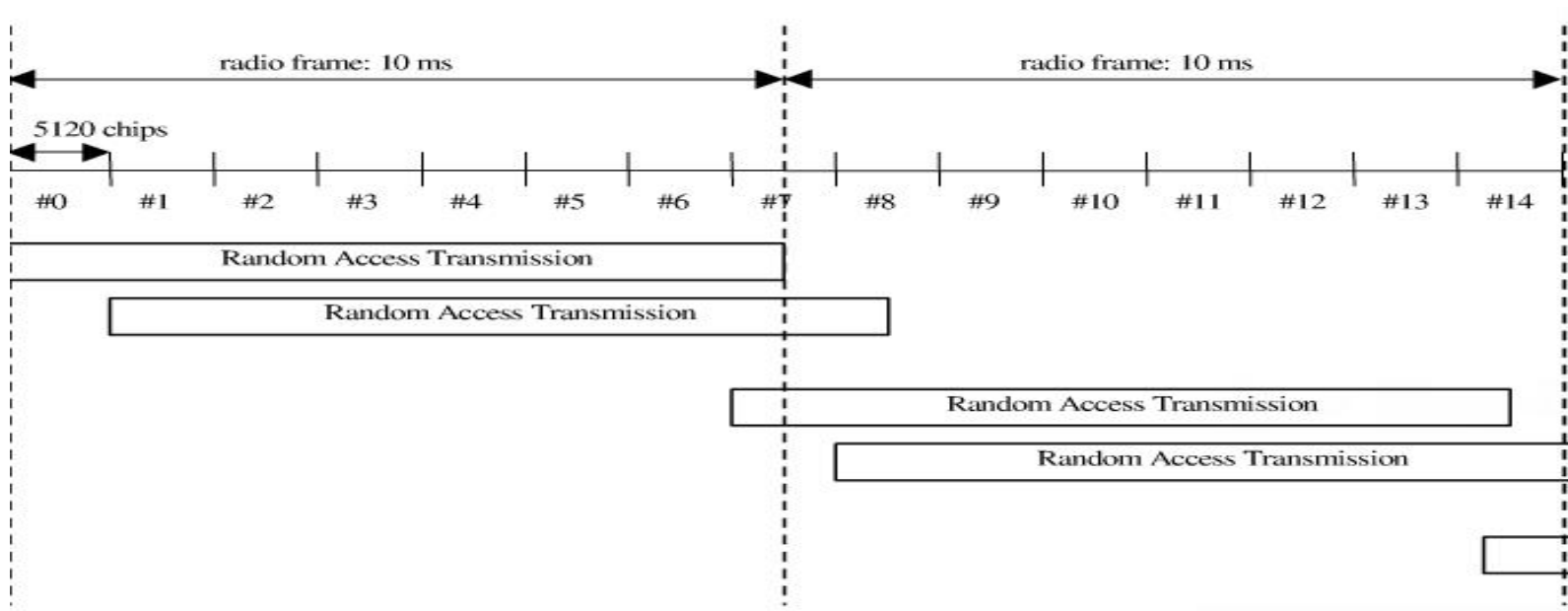
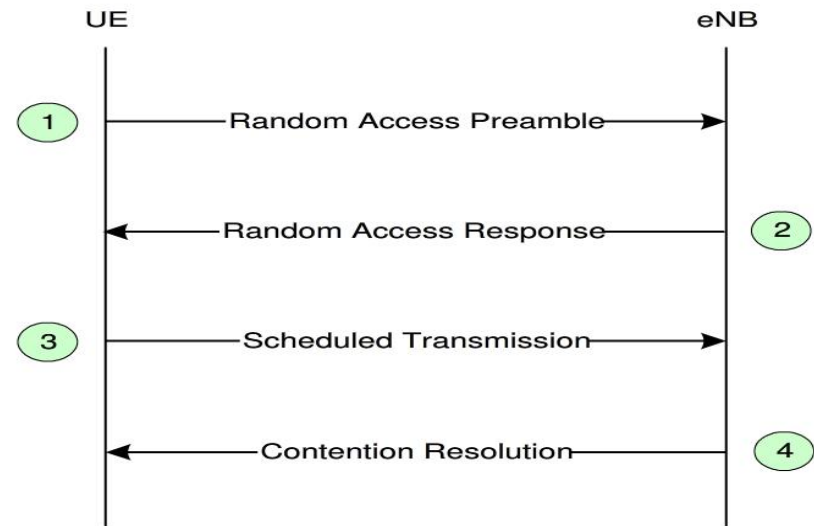
- random retrial period
  - “permission” probability
- } Same thing..!

(SLOTTED ALOHA protocol)

# 随机接入技术: 时隙ALOHA(cont.)



## Example [LTE – 随机接入]

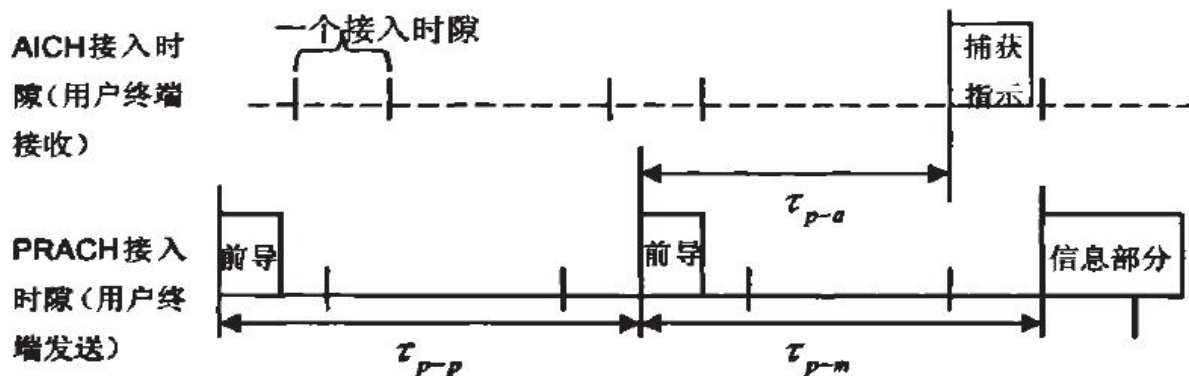
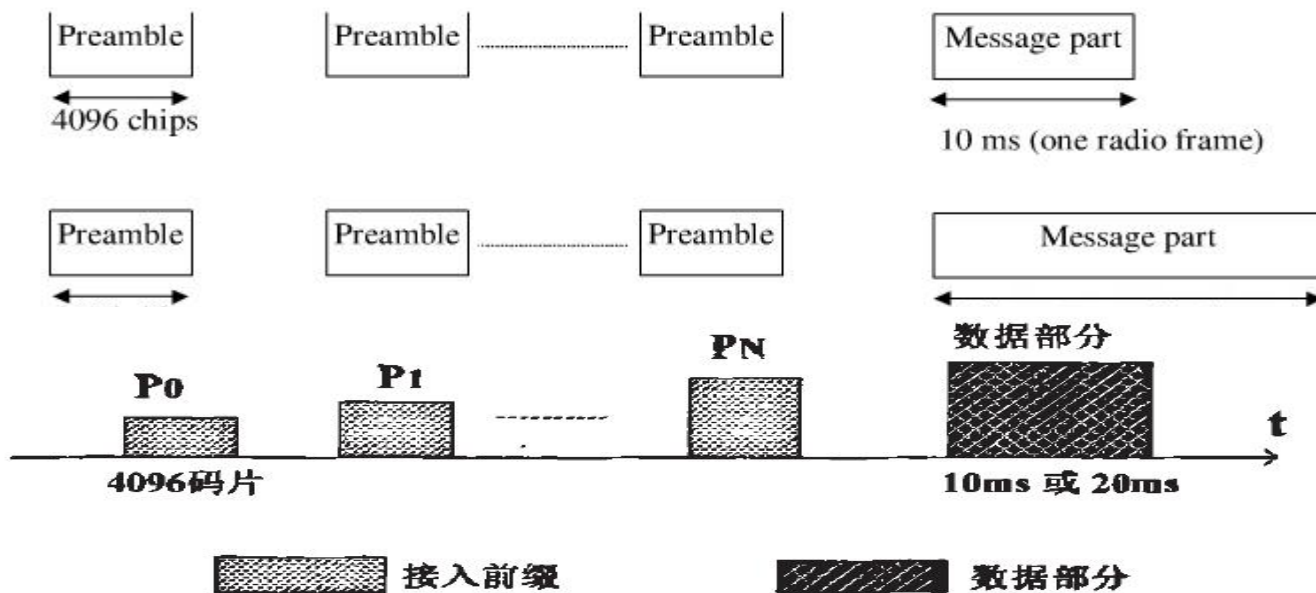




# 随机接入技术: 时隙ALOHA(cont.)



## Example [WCDMA中的扩展 Aloha]



### 3、载波侦听多址接入

## Carrier Sensing Multiple Access



#### ALOHA 劣势

- 每个站点独立决定如何发送
- 站点不关注其它站点是否在发送 (CSMA 关注)
- 站点不停止发送，即使其它站点已干扰到自己的发送 (CSMA/CD 就停止发送)

#### CSMA

- 通过以下规则，降低冲突概率：
- 载波侦听 **carrier sensing** – 站点在发送前侦听信道
  - 如果信道空闲 **idle**  $\Rightarrow$  发送整个帧
  - 如果信道忙碌 **busy**  $\Rightarrow$  随机退避一段时间并重新侦听信道



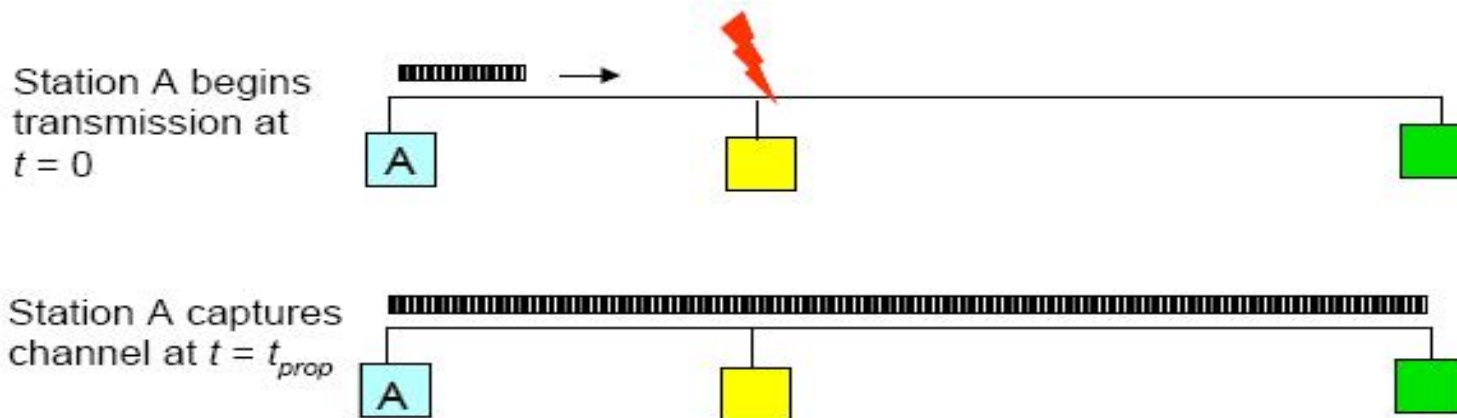
# 载波侦听多址接入(cont.)



- 随着信号传播，其它站点都意识到A的发送
- 在 $t=t_{prop}$  A的信号到达介质的另一端 – 如果在这个时间内没有其它站点发送，则A成功捕获信道

易损期 =  $t_{prop}$

- 注意: 如果  $t_{prop} > t_{frame}$ ，CSMA 相比于时隙ALOHA并无增益

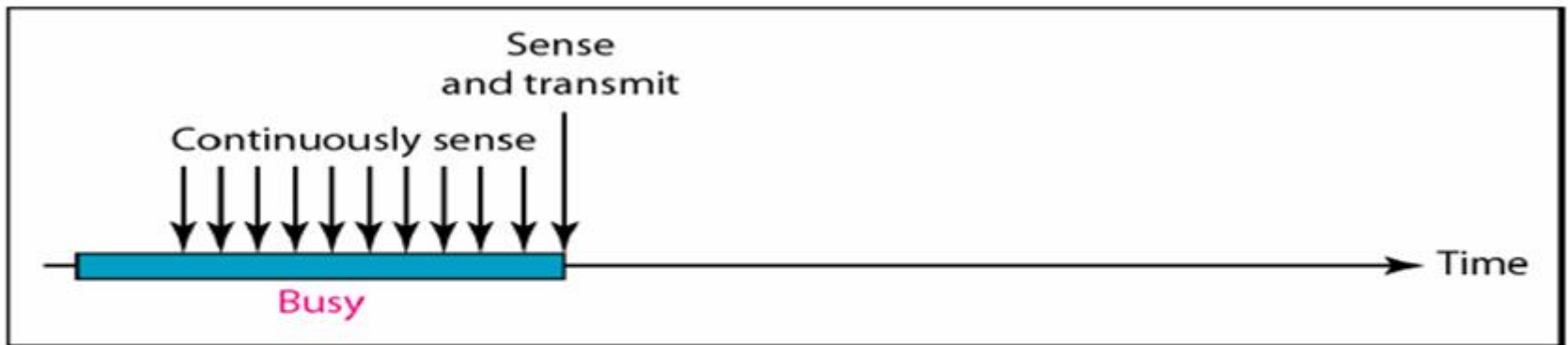


# 载波侦听多址接入(cont.)

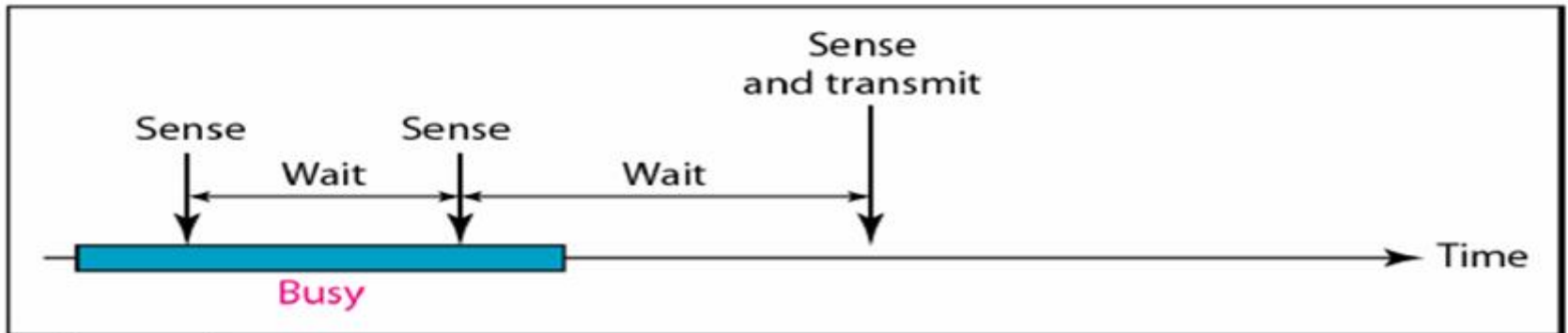


## CSMA 选项

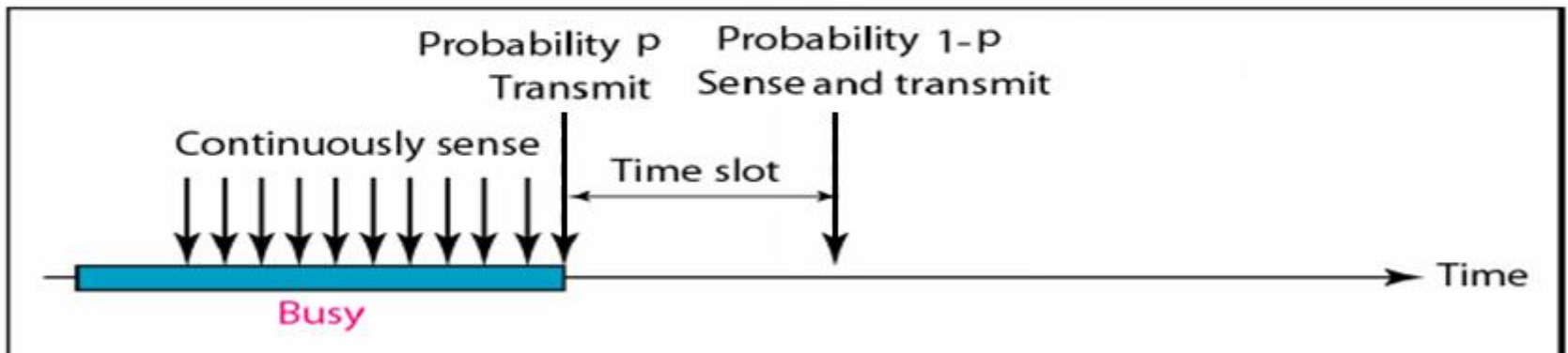
- 当侦听到信道忙碌后，发送端的行为
  - 1-坚持CSMA 1-persistent CSMA (最贪婪)
    - 一旦信道空闲就立刻发送
    - 低时延低效率
  - 非坚持CSMA Non-persistent CSMA (最不贪婪)
    - 等待一个退避时间，然后再次侦听
    - 高时延高效率
  - p-坚持 CSMA  $p$ -persistent CSMA (可调贪婪)
    - 等待直到信道空闲，以概率 $p$ 发送，或以概率 $(1-p)$ 退避一个最短时隙mini-slot并重新侦听
    - 时延和效率达到平衡



a. 1-persistent



b. Nonpersistent

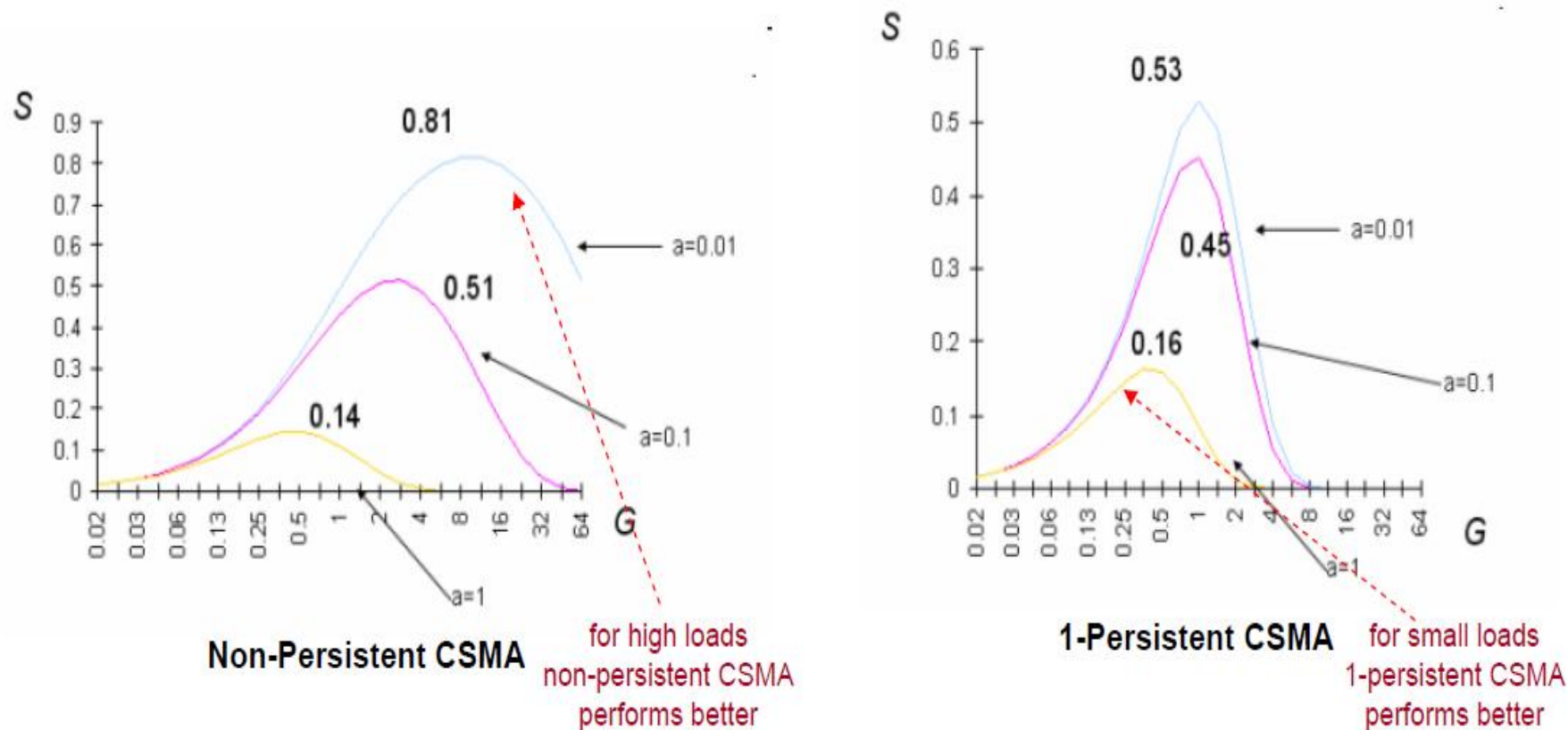


c. p-persistent

# 载波侦听多址接入(cont.)



## CSMA 吞吐量



参数对最大可达吞吐量具有显著影响.

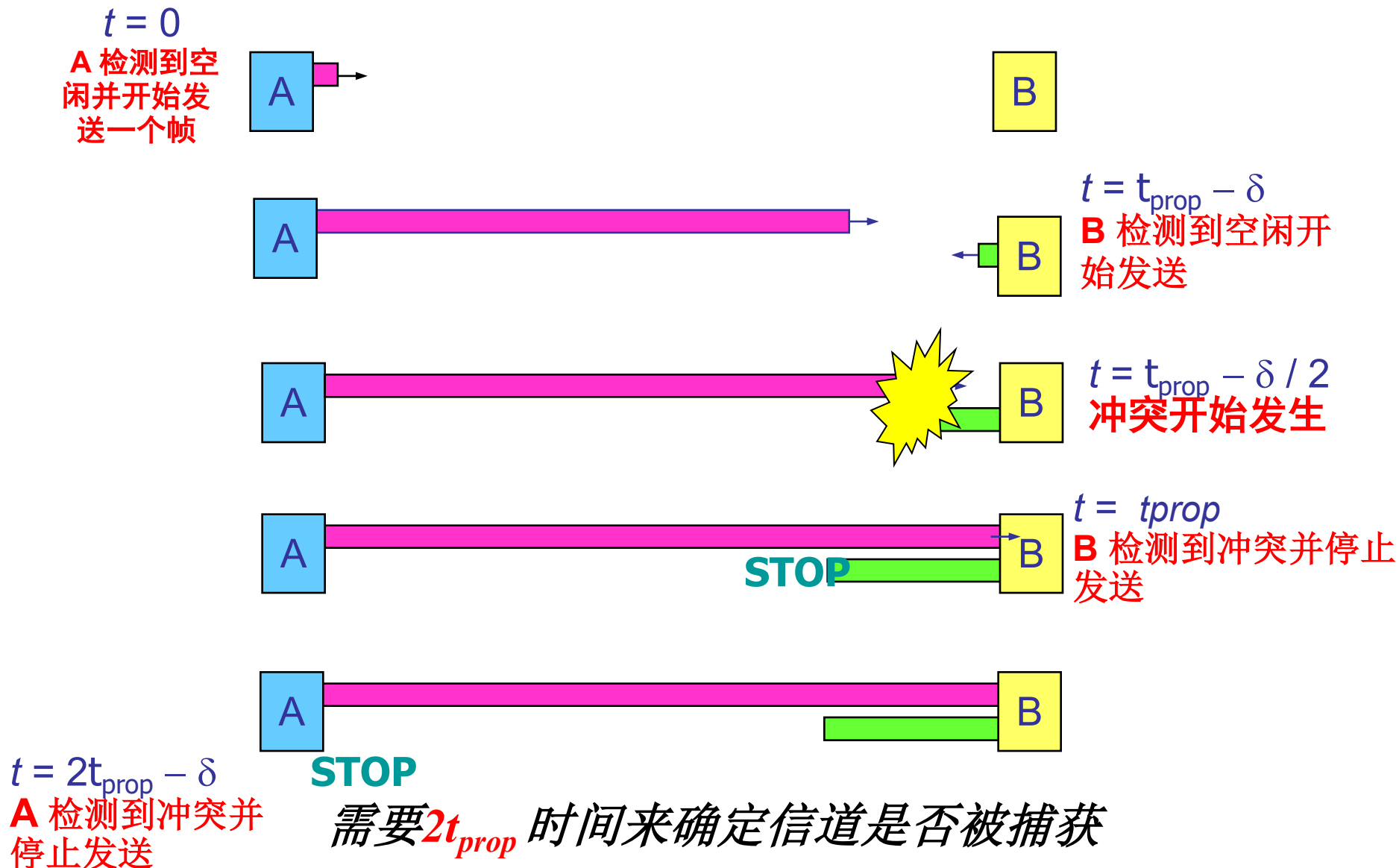
1-坚持CSMA的吞吐量随 $G$ 的增加而下降的更加明显

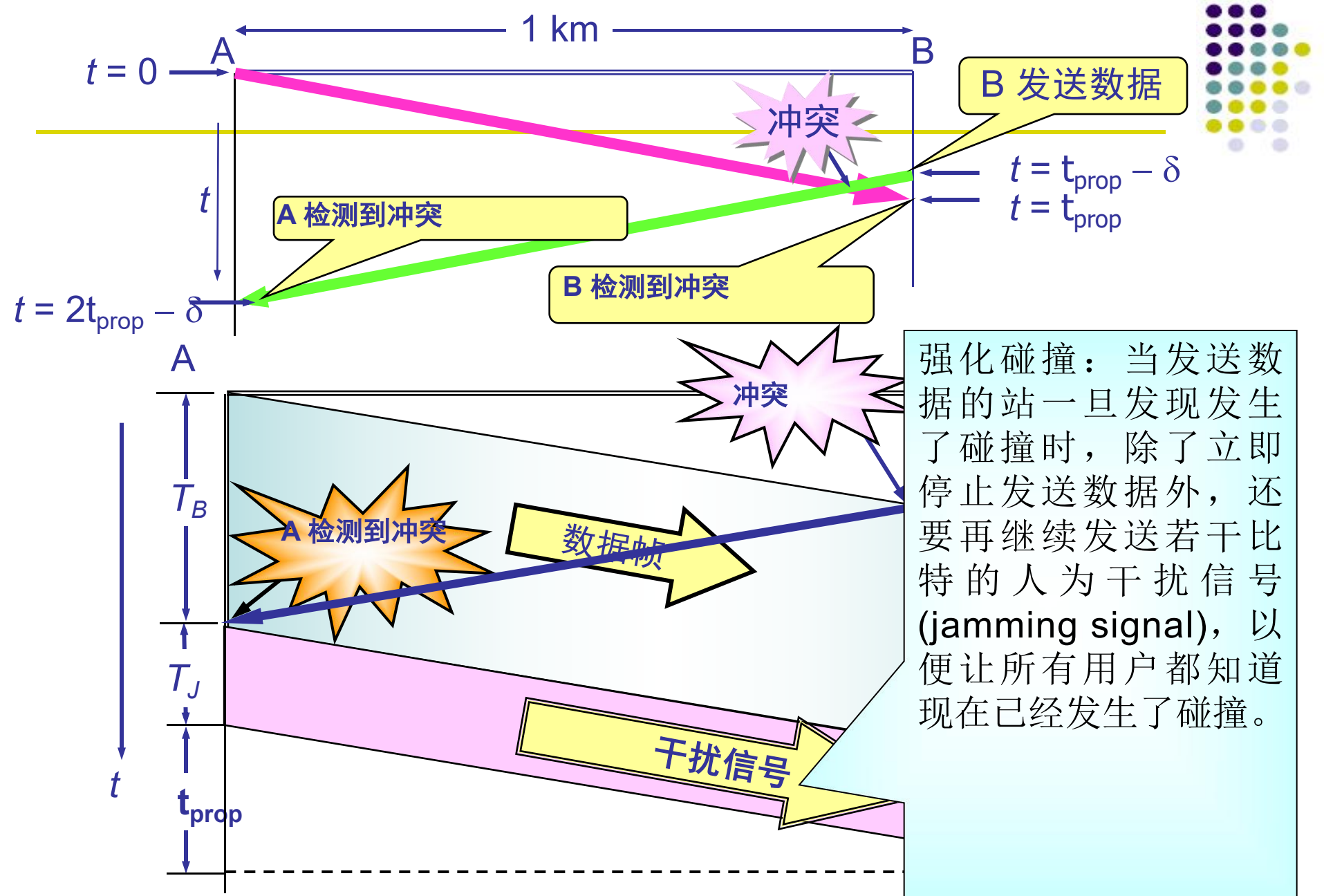


# 具有冲突检测collision detection的CSMA

- CSMA中，冲突将导致X秒浪费，即一整个帧长的时间
- CSMA-CD 通过冲突检测并退出传输来减小时间浪费
- 监测冲突 & 退出发送:
  - 有帧要发送的站点，首先进行载波侦听
  - 发送开始后，站点持续监听信道并检测是否有冲突
  - 如果检测到冲突，所有正在发送帧的站点发送一个短的干扰信号并停止发送，重新设定退避时间，等待后重新试着发送（侦听...）

# CSMA/CD 反应时间reaction time



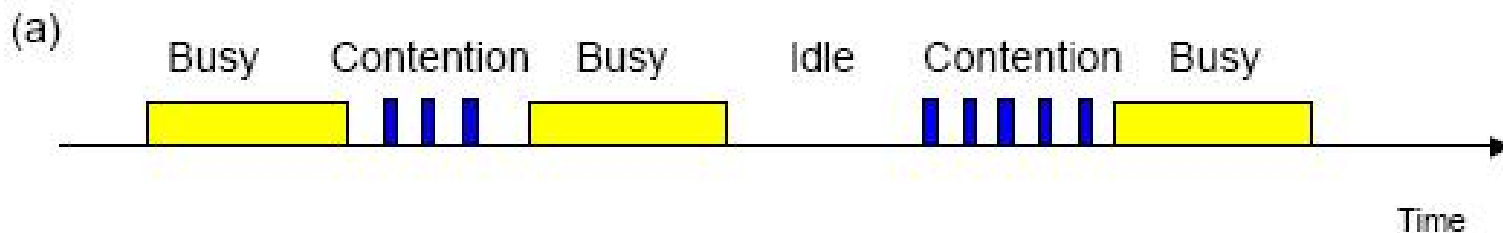


$$t_{\text{prop}} + T_j \leq \text{冲突时间} \leq 2t_{\text{prop}} + T_j$$

## 4、CSMA-CD 模型



- 假设
  - 争用一次需要  $2t_{prop}$  时间，即每个竞争时隙长度为  $2t_{prop}$
  - 假设有  $n$  个站，每个站在竞争时隙（contention time slot）内以概率  $p$  发送数据
  - 当竞争期结束时（某个站成功捕获信道），它还需要  $X$  秒将帧发送出去
  - 在下一次竞争期开始前需要  $t_{prop}$  时间





# 争用解决



- 需要多少时间解决争用？
- 争用被解决（“成功”）的条件是：在某个时隙内仅有1个站点发送：

$$P_{success} = np(1-p)^{n-1}$$

- 对 $P_{success}$ 求导（以 $p$ 为偏导数），得到当 $p=1/n$ 时达到最大值

$$P_{success}^{max} = n \frac{1}{n} \left(1 - \frac{1}{n}\right)^{n-1} = \left(1 - \frac{1}{n}\right)^{n-1} \rightarrow \frac{1}{e}$$

- 平均来看，需要 $1/P^{max} = e = 2.718$  个时隙来解决争用

$$\text{平均争用时间} = 2t_{prop}e \text{ 秒}$$

# CSMA/CD 吞吐量



- 在最大吞吐量状态，系统交替进行争用和帧的传输

$$\rho_{\max} = \frac{X}{X + t_{prop} + 2et_{prop}} = \frac{1}{1 + (2e + 1)a} = \frac{1}{1 + (2e + 1)Rd / v L}$$

- 其中:

$R$  bits/sec,  $L$  bits/frame,  $X=L/R$  seconds/frame

$$a = t_{prop}/X$$

$v$  meters/sec. 介质中光速

$d$  meters 是系统的最大端到端长度

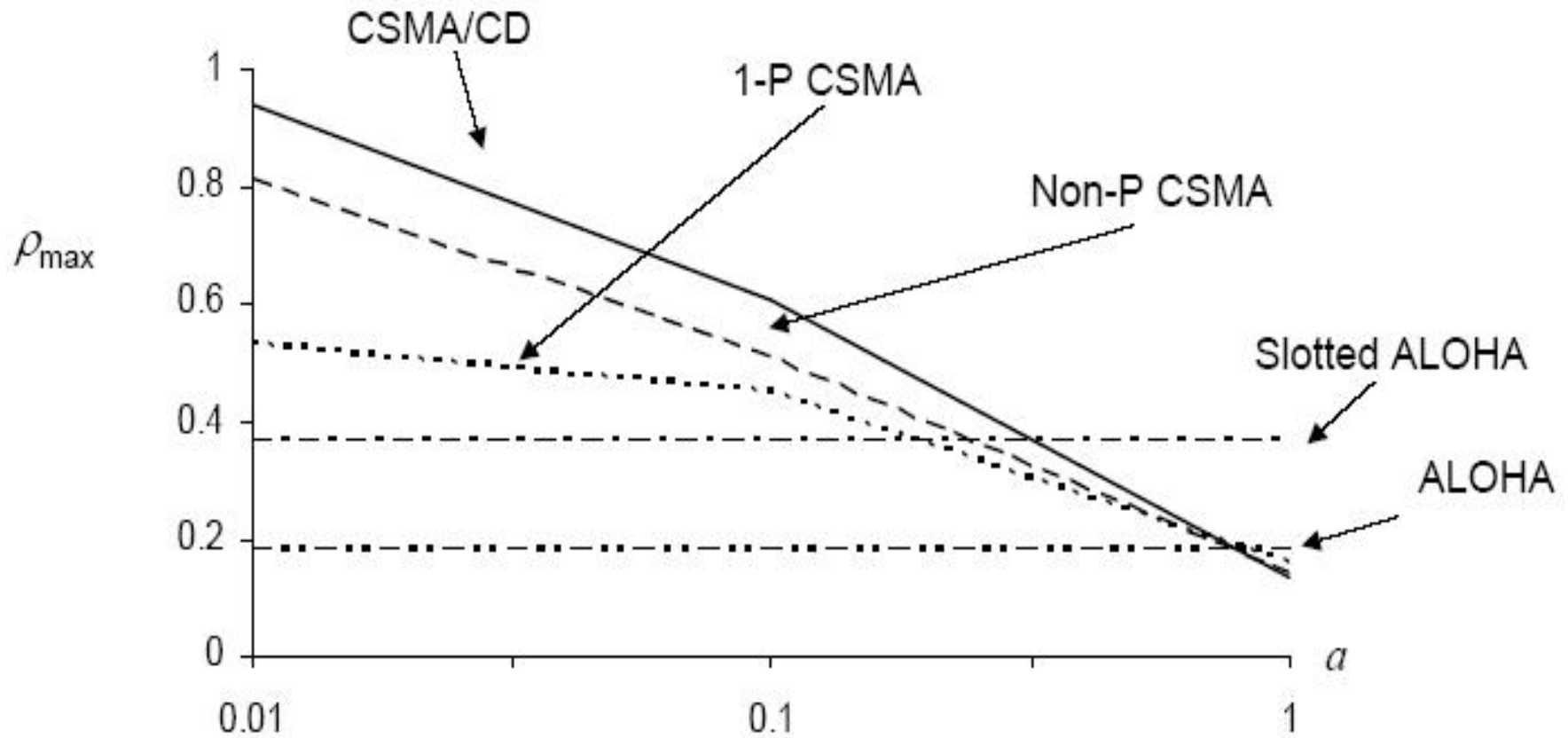
$$2e+1 = 6.44$$

# CSMA-CD 应用: 以太网



- 第一个以太网局域网标准采用了CSMA-CD协议
  - 1-坚持载波侦听
  - $R = 10 \text{ Mbps}$
  - $t = 51.2 \text{ microseconds} > 2t_{\text{prop}}$ 
    - $R * t = 512 \text{ bits} = 64 \text{ byte slot}$
    - 支持  $2.5 \text{ km} + 4 \text{ 中继器}$  远小于  $10 \text{ km}$  ( $2 * 10^5 \text{ km/s} * 51.2 \text{ ms} \approx 10.24 \text{ km}$ )
  - 截断二进制指数退避 (Truncated Binary Exponential Backoff)
    - 第 $n$ 次重传冲突发生后, 随机从 $\{0, 1, \dots, 2^k - 1\}$ 中选择退避时隙, 其中 $k = \min(n, 10)$

# 随机接入MAC协议的吞吐量



- 对于比较小的 $a$ : CSMA-CD 吞吐量最大
- 对于比较大的 $a$ : Aloha 和时隙 Aloha 吞吐量更高

# 载波侦听和具有优先级的传输



- 某些帧的优先级更高，如**ACK**信息，需要尽快发送
  - 高优先级业务侦听信道的时间是 $T_1$
  - 低优先级业务侦听信道的时间是 $T_2 > T_1$
  - 这导致高优先级业务率先捕获信道
- 这种优先级机制应用于**IEEE 802.11** 无线局域网

# Chapter 6

## Medium Access

### Control protocols and Local Area Networks

---

*调度Scheduling*



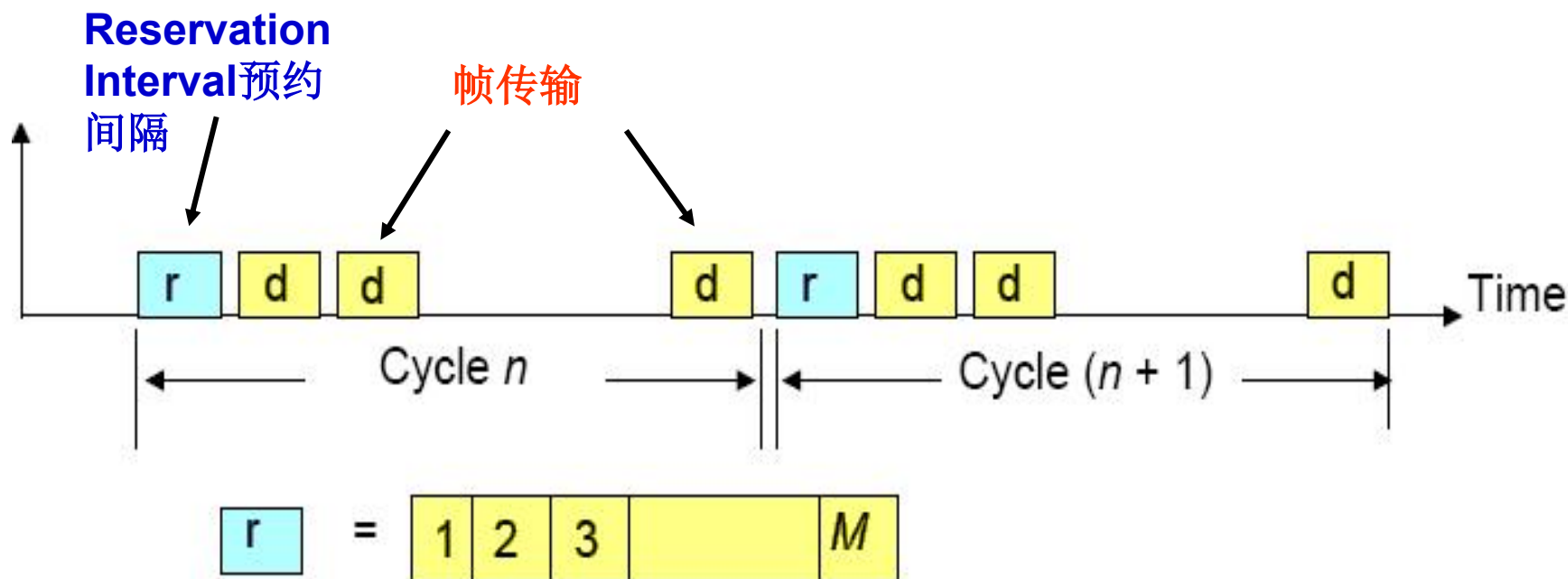


# 介质访问控制中的调度方法

---

- 在共享介质中采用帧调度来避免冲突
  - ✓ 更高效的信道利用
  - ✓ 时延不确定性更小
  - ✓ 可提供站点间的公平性
  - ✗ 计算和处理复杂度增加
- 两种主要方式
  - Reservation 预约
  - Polling 轮询

# 1、预约系统



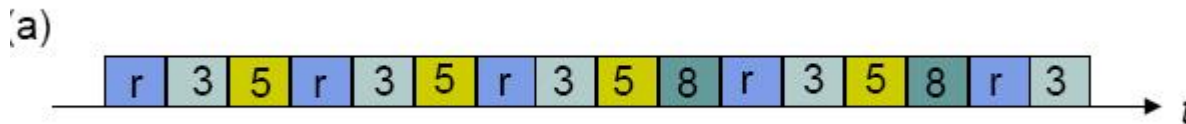
- 传输被组织为一定的循环周期cycle
- 循环周期: 预约间隔+帧传输
- 预约间隔为每个站点分配了一个微时隙 (minislot) 来请求传输
- 某个周期的长度对应有帧要发送的站点的个数!



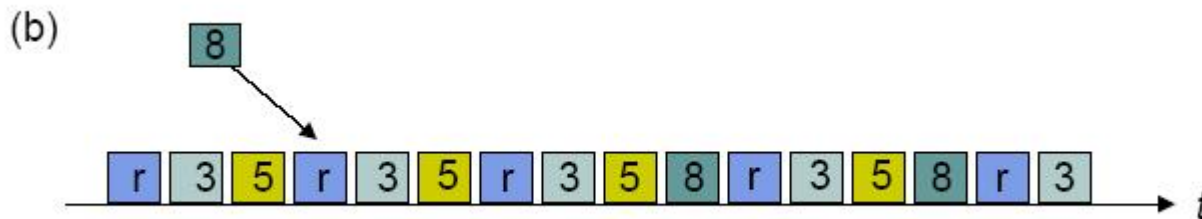
# 例子



- 最开始站 3 & 5 预约了要发送帧



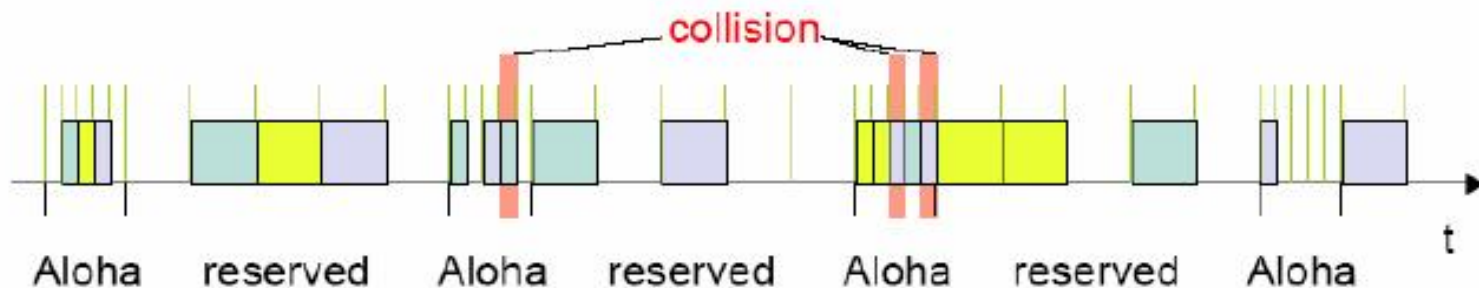
- 站点8开始活跃并进行了预约
- 现在的周期中包括了站点8的帧传输（可以规定预约在大于一定数量周期后才能生效）



# 例子：GPRS中的随机接入预约系统



- 通用分组无线服务 General Packet Radio Service
  - GSM蜂窝无线中的分组数据传输服务
  - GPRS 设备如手机和笔记本电脑，以无线方式发送分组数据连接到因特网
  - 预约使用时隙 ALOHA MAC 方式
    - 时隙ALOHA预约方法：站点在微时隙上采用时隙ALOHA的方式进行预约
  - 支持单/多时隙预约：
    - 单帧预约：一个预约周期内仅能预约1个帧
    - 多帧预约：一个预约周期内能预约多个帧



# 预约系统和服务质量QoS

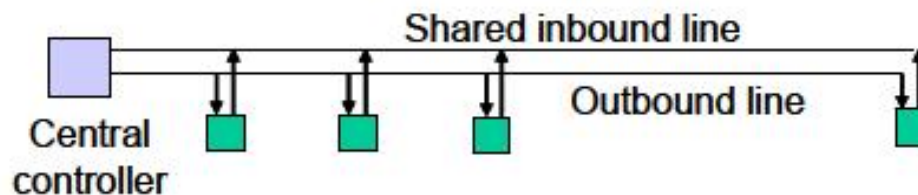


- 不同应用有不同要求
  - ACK帧需立即传输
  - 话音信号需低延时和稳定带宽
  - 网络传输需高带宽
- 预约提供了直接保证QoS的方法
  - 站点为每个帧进行预约请求
  - 站点可以请求持续传输（从而保证时延等QoS）
- 集中式协议中，控制单元发布许可
- 分布式协议中，允许站点来决定许可

## 2、轮询系统



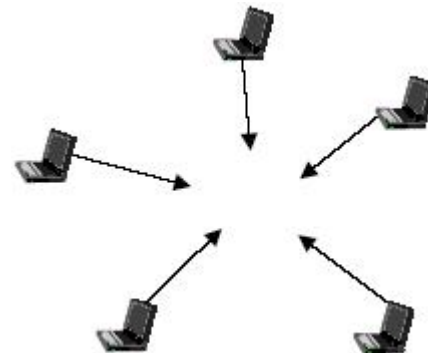
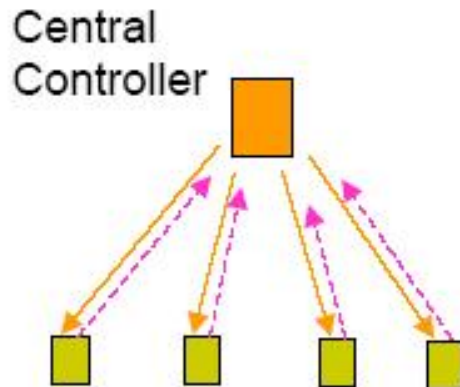
- 一个设备被指定为主站 **primary station** (中心控制器central controller)，其它设备被指定为从站**secondary stations**
- 系统具有两条线路
  - 带外线路**outbound line** – 用来从主站到从站发送数据
  - 带内线路**inbound line** – 从来从从站到主站发送数据，所有从站需共享该线路
- 中心控制器按照循环的方式发送轮询消息 ‘**polling messages**’ 到从站，询问它们是否有帧要发
- 从站只有当被中心控制器问询到才能发送帧– 在给定时间，仅有一个从站有权发送帧
- 从站发送‘**go-ahead message**’来指示帧传输已经完成，中心控制器收到该消息后，将轮询下一个从站



# 轮询系统



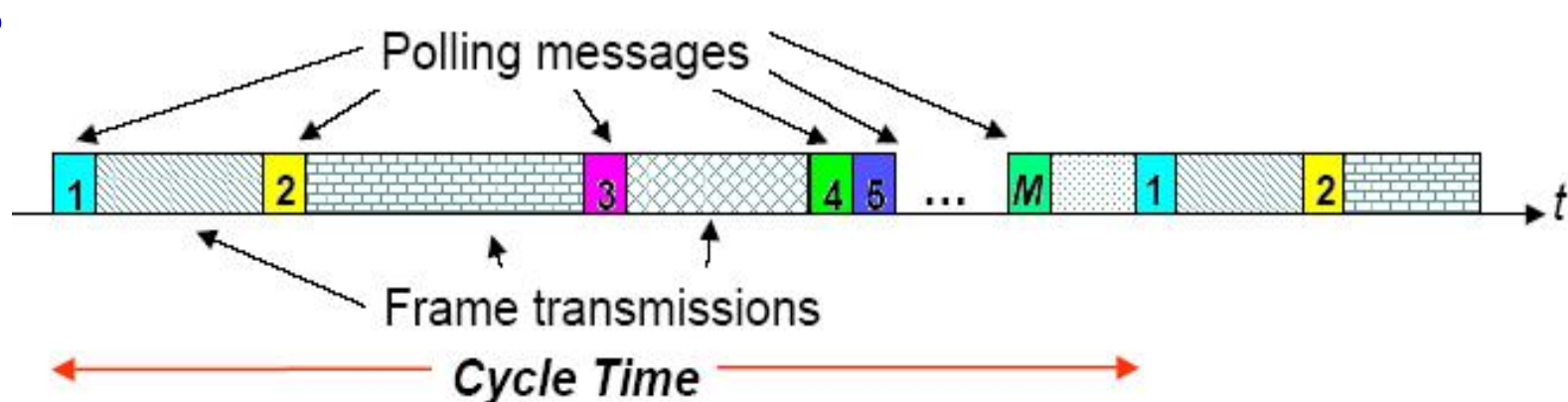
- **中心化系统Centralized systems:** 一个中心站接受从站的请求并进行发送的许可
  - 频分复用Frequency Division Duplex (FDD): 上下行采用分离的频段
  - 时分复用Time-Division Duplex (TDD): 上下行以时分方式使用同一个信道
- **分布式系统Distributed systems:** 站点采用某种分布式算法决定传输的次序



# 漫游时间和周期时间Walk Time & Cycle Time



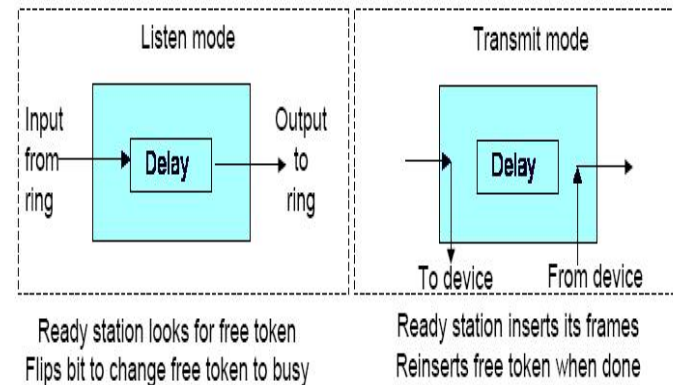
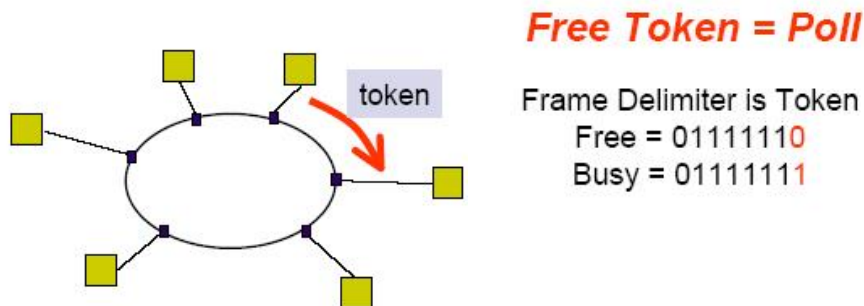
- 假设轮询是循环（round robin）进行的
- 对站点的轮询时间是一种额外开销
  - 准备和发送轮询消息的时间
  - 站点收到轮询消息的反应时间
- 漫游时间*Walk time*: 从一个站点结束发送数据的时刻到下一个站点开始发送数据的时刻
- 周期时间*Cycle time* 是指一个站前后两次被轮询到的时间间隔



### 3、令牌传送系统Token-Passing Systems



- ‘去中心化 / 分布式轮询’ 系统 – 站点组成一个环形网络
  - 一个帧传输的许可（即令牌token）从一个站传递到另一个站
  - 当不发送帧时, 每个站的接口就像转发器一样工作 – 它将所有从输入端口进来的比特复制到输出端口
  - 当站点收到一个空闲的令牌且有帧要发时, 其接口将令牌修改为“忙碌”状态并进行发送模式
  - 当传输结束时, 令牌被再次修改为“空闲”
  - 令牌持有时间Token Holding Time – 一个站能够持有令牌时间的上限 (i.e. 它决定了一个站一次最多发送多少数据)



# 令牌传送系统 (cont.)



## 帧删除

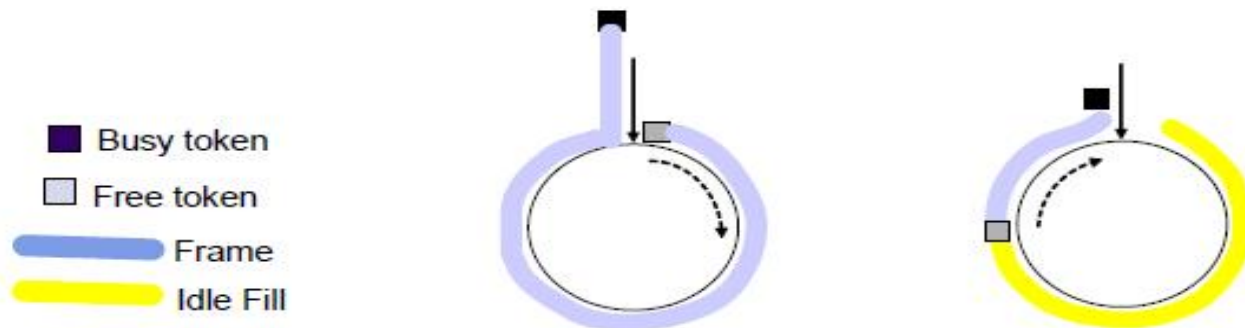
– 环上的每个帧最后都要删除掉; 删除帧的方法:

(a) 目的站删除帧

(b) 帧再次回到发送站, 并由发送站删除帧 (更倾向 – 一种非直接形式的确认)

环延迟Ring Latency– 环上能够同时运输的比特数

- 帧长 > 环延迟  $\Rightarrow$  重新返回到发送站的比特仍属于站点正在发送的那个帧
- 帧长 < 环延迟  $\Rightarrow$  环上可能同时在传送多个帧





# 令牌释放的方法

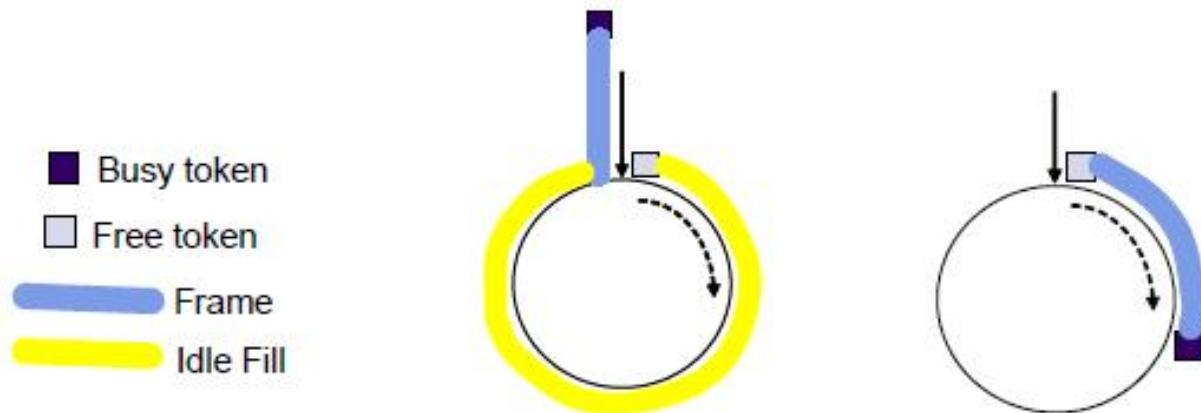


(1) **Delayed Token Release** 延迟的令牌释放 – 即单帧操作 **Single-Frame Operation** – 当整个帧返回发送站后再插入空闲令牌

- 简化了ACK过程 – 目的站在帧后边附加ACK表明已正确接收
- 用于较慢的网络! (帧长  $\approx$  环延迟)

(2) 尽早的令牌释放 **Early Token Release** – 即多令牌操作 **Multitoken Operation** – 帧发送完毕后立即插入空闲令牌

- 最小化了传送空闲令牌的时间
- 当 帧长  $\ll$  环延迟  $\Rightarrow$  多个帧可以在网络的不同位置同时传送  $\Rightarrow$  更高的吞吐量
- 用于较快的网络! (帧长  $\ll$  环延迟)



# 应用例子

---



- Single-frame reinsertion单帧操作
  - IEEE 802.5 Token Ring LAN @ 4 Mbps
- Single token reinsertion单令牌操作
  - IBM Token Ring @ 4 Mbps
- Multitoken reinsertion多令牌操作
  - IEEE 802.5 and IBM Ring LANs @ 16 Mbps
  - FDDI Ring @ 50 Mbps
- 所有这些局域网都加入了令牌优先级的机制

## 4、 MAC 方式比较

---



- **Aloha & 时隙 Aloha**

- 在低负载情况下简单和快速传输
- 支持大量突发、短数据的用户
- 中等负载情况延迟变化很大
- 效率与 $a$ 无关

- **CSMA-CD**

- 对于低时延带宽积情况，能够快速传输，具备高效率
- 支持大量突发数据用户
- 可变和不可预测的延迟

# MAC 方式比较



- **预约**

- 按序传输突发或稳定的数据流
- 使用时隙ALOHA预约，可支持大量低业务负载用户
- 可支持QoS
- 使用延迟的许可，可应对大的时延-带宽积

- **轮询**

- 时分复用的广义形式
- 提供公平性
- 可设定最大接入时延
- 当时延-带宽积很大时，性能下降

# 随机接入 vs. 调度接入



	Random Access (ALOHA, CSMA)	Scheduling Access
<b>delay</b>	small under light loads 😊	longer but generally less variable between stations
<b>throughput</b>	sufficient under light load, drops significantly under heavy loads	increases under heavy load 😊
<b>fairness</b>	not guaranteed 😞	guaranteed 😊
<b>sensitivity to node failure</b>	small 😊	high, particularly in polling and token ring systems 😞

# Chapter 6

## Medium Access Control protocols and Local Area Networks

*Homework* : 6.4

