Universidade Federal de Ouro Preto

A Camada de Enlace

Links desta videoaula - P1: youtu.be/2cZff2UajUY
P2: youtu.be/jMN6XHdK8ns

Referências:

- Redes de Computadores. **A. S. Tanenbaum.** Campus/Elsevier, 2003 - Seções 3.1 e 3.2

- Redes de Computadores e a Internet. **J. Kurose, K. Ross.** Pearson, 2010 - Capítulo 5

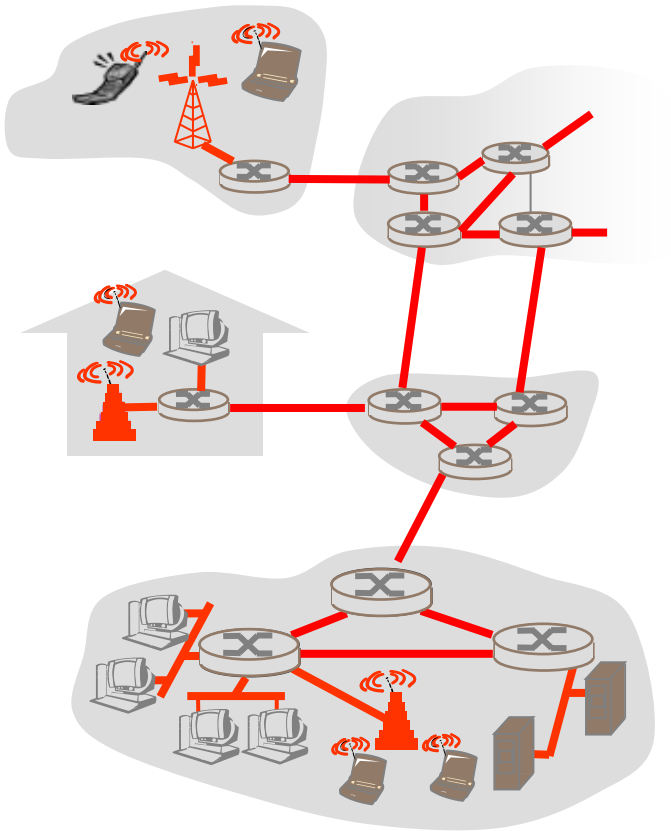
Camada de enlace

- **5.1 Introdução e serviços**
- **5.2 Detecção e correção de erros**
- **5.3 Protocolos de acesso múltiplo**
- **5.4 Endereçamento na camada de enlace**
- **5.5 Ethernet**

Camada de enlace: introdução

Alguma terminologia:

- **hospedeiros e roteadores são nós**
- **canais de comunicação que se conectam a nós adjacentes pelo caminho de comunicação são enlaces**
  - enlaces com fio
  - enlaces sem fio
  - LANs
- **pacote na camada-2 é um quadro, encapsula datagrama**

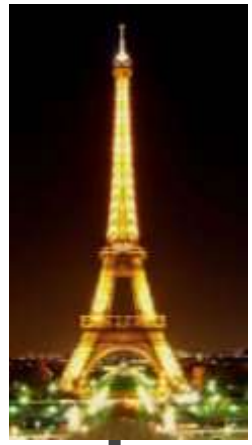


Serviços da camada de enlace

- Um protocolo da camada de enlace é usado para transportar um datagrama **por um enlace individual.**
- **Camada de transporte:** movimentar pacotes da camada de aplicação fim a fim.
  - Desde o host origem ao destino
- **Camada de enlace:** movimentar datagramas da camada de rede nó a nó por um único *enlace* no caminho.
- Analogia: sistema de transporte

Serviços da camada de enlace

- **Analogia:** Agente de viagens planejando um passeio para turista de BH a Paris, passando por Frankfurt.



Agente de transporte: protocolo de roteamento.

Turista: datagrama.

Meio de transporte: protocolo da camada enlace.

Taxi, avião e trem-bala **são os enlaces!**

Cada um fornece o serviço básico, que é levar passageiros de uma localidade a outra adjacente.

Serviços da camada de enlace

- ***enquadramento :***
  - encapsula datagrama no quadro, incluindo cabeçalho para transmiti-lo no enlace.

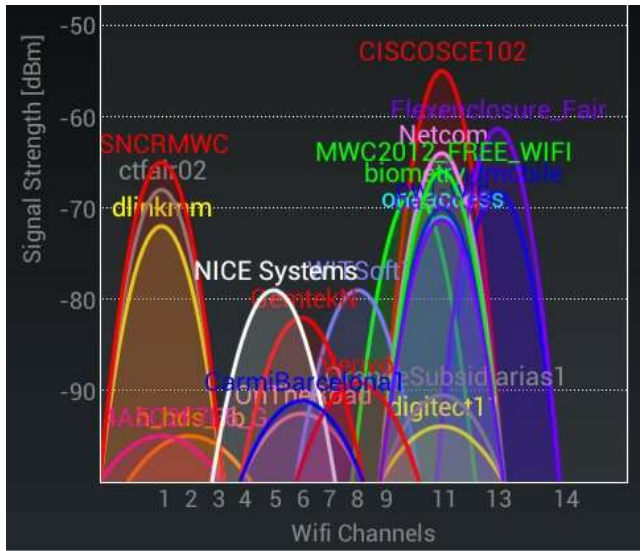
- ***acesso ao enlace***
  - Define regras para acesso ao canal de meio compartilhado;
  - endereços “MAC” usados nos cabeçalhos de quadro para identificar origem, destino.
    - diferente do endereço IP !

Serviços da camada de enlace

- ***entrega confiável entre nós adjacentes***
  - raramente usado em enlace com pouco erro de bit (fibra, alguns pares trançados);
    - Pode ser considerado uma sobrecarga desnecessária para este tipo de enlace.
    - Muitos protocolos dessa camada não oferece esse serviço.

  - Por que ter confiabilidade em **nível de enlace** e **fim a fim**?

  - Usado em enlaces sem fio: altas taxas de erro. **Ex: Wi-Fi**

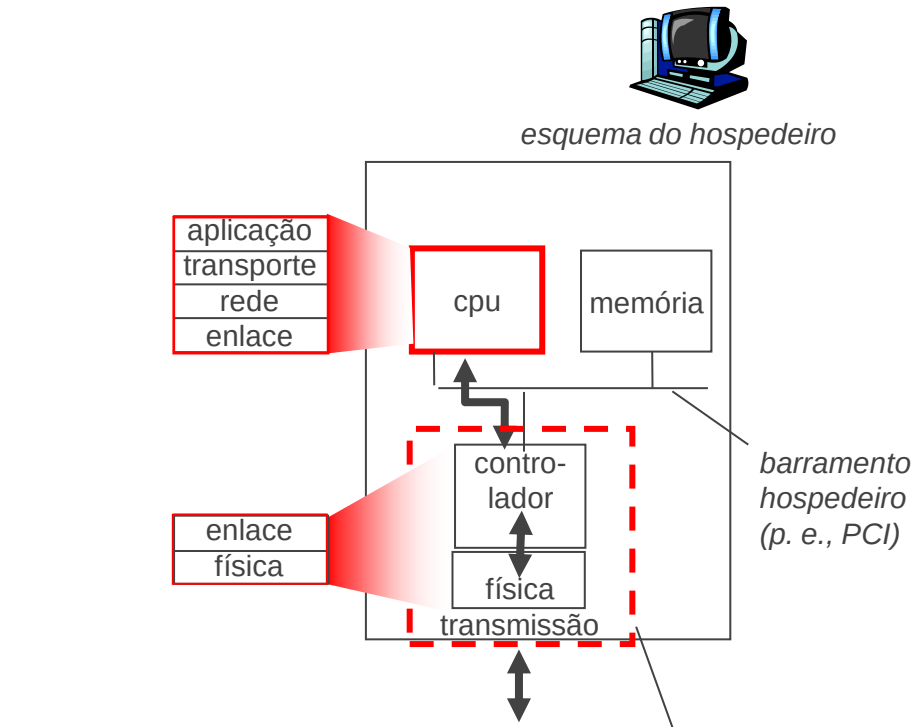


Serviços da camada de enlace

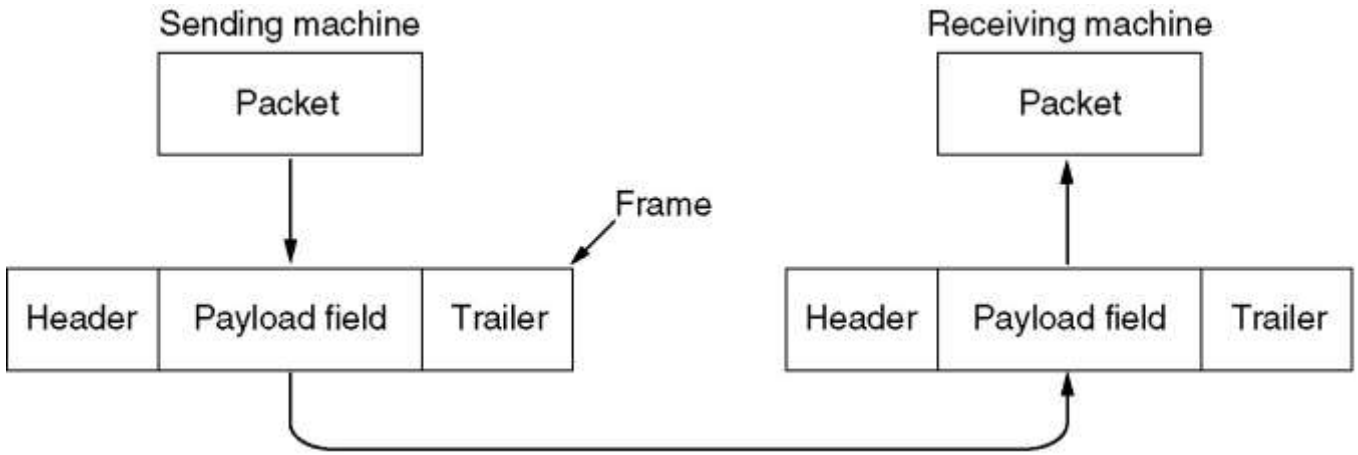
- ***controle de fluxo:***
  - controle entre nós de emissão e recepção adjacentes
  - semelhante à camada de transporte, porém faz o controle nó a nó
- ***detecção de erro*:**
  - erros causados por atenuação de sinal, ruído.
  - receptor detecta presença de erros:
    - Não há necessidade de repassar erros
    - descarta quadro ou corrige os erros.
- **correção de erro:**
  - receptor identifica *e corrige* erro(s) de bit sem lançar mão da retransmissão

Onde é implementada a camada de enlace?

- **em todo e qualquer host.**
- **camada de enlace implementada no "adaptador" (ou *placa de interface de rede*, NIC)**
  - placa Ethernet, placa PCMCI, placa 802.11
  - implementa camada de enlace, física
- **conecta aos barramentos de sistema do hospedeiro**
- **combinação de hardware, software, firmware**

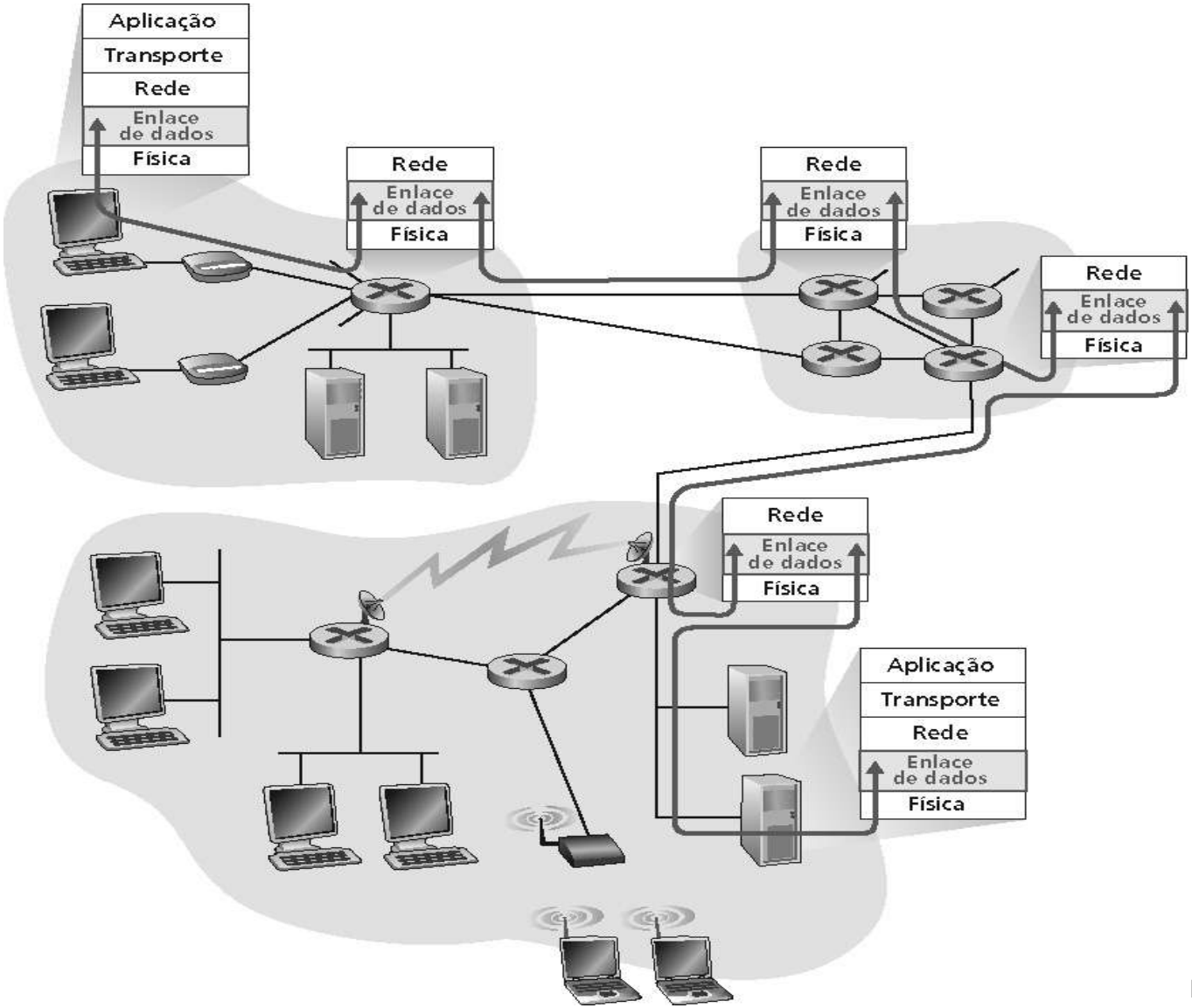


Comunicação entre adaptadores



- **lado emissor:**
  - encapsula datagrama no quadro
  - inclui bits de verificação de erro, rdt, controle de fluxo etc.
- **lado receptor**
  - procura erros, rdt, controle de fluxo etc.
  - extrai datagrama, passa para camada superior no lado receptor

Comunicação entre adaptadores



Camada de enlace

- 5.1 Introdução e serviços
- 5.2 Detecção e correção de erros
- 5.3 Protocolos de acesso múltiplo
- 5.4 Endereçamento na camada de enlace
- 5.5 Ethernet

Controle de erros

Durante a transmissão de dados erros podem ocorrer:

- interferências eletromagnéticas;
- Atenuação de sinal;
- falha de sincronização entre emissor e receptor;
- Problemas em componentes de rede;

- Necessária alguma técnica para correção e detecção de erros.
  - Reduz o número de retransmissões do remetente.
  - Vantagem potencialmente importante para aplicações de tempo real.

- **Diferentes técnicas atualmente disponíveis:**
  - Estratégias mais sofisticadas têm maior probabilidade de detectar erros, mas...
    - ficam sujeitas a sobrecarga maior.

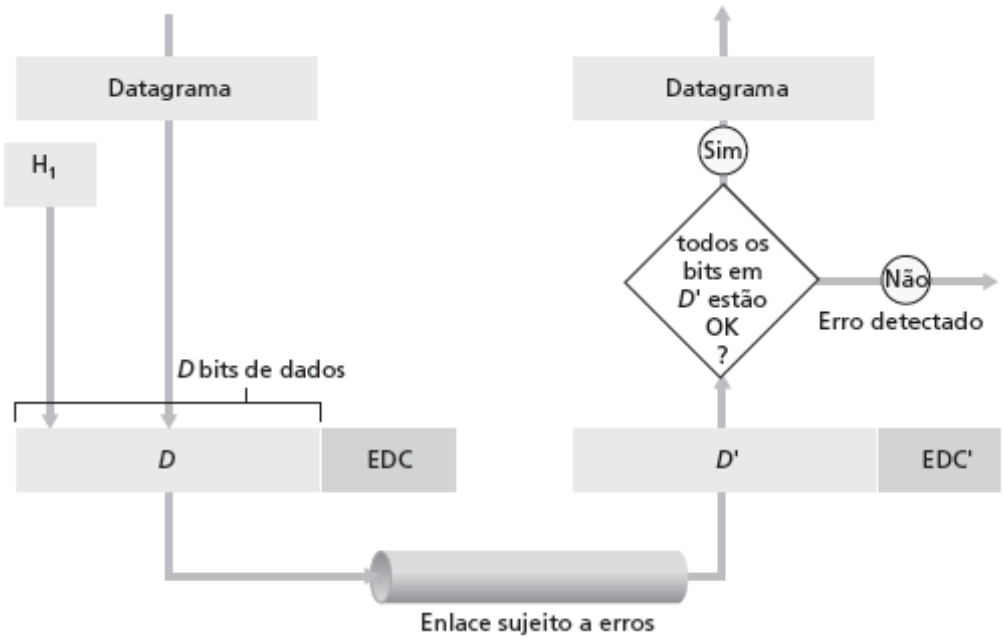
Detecção de erros

EDC = Bits de detecção e correção de erros (redundância)

D = Dados protegidos por verificação de erro (pode incluir cabeçalho)

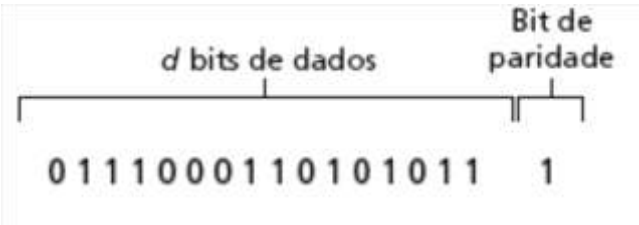
Detecção de erro não é 100% confiável!

- protocolo pode perder alguns erros, mas raramente
- maior campo EDC gera melhor detecção e correção



Verificação de paridade

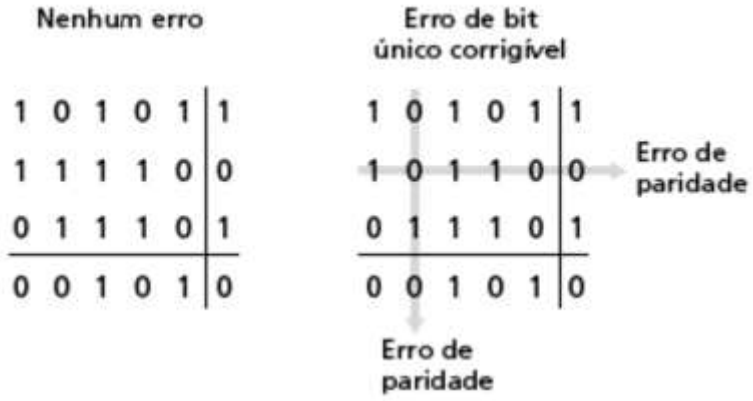
- Maneira mais simples de detectar erros: **utilizar um bit de paridade**
- Incluir um bit adicional aos dados
- **Paridade impar e paridade par**



- **PROBLEMA:**
- Erros frequentemente se aglomeram em rajadas
- Se muitos bits forem transmitidos e ocorrer muitos erros?
- Essa técnica só detecta um número ***ímpar*** de erros.

Paridade Bidimensional

- **Suponha que tenha ocorrido erro em 1 bit.**
- **Agora receptor pode detectar erros e corrigir alguns!!**
- Não é 100% eficiente para correção de erros.
- Pode detectar (mas não corrigir) qualquer combinação de 2 erros no pacote.



Soma de verificação da Internet (análise)

Objetivo: detectar “erros” (p. e., bits invertidos) no pacote transmitido (nota: usada *somente* na camada de transporte)

Emissor:

- trata conteúdo do segmento como sequência de inteiros de 16 bits
- soma de verificação: adição (soma no complemento de 1) do conteúdo do segmento
- emissor colocar valor da soma de verificação no campo de soma de verificação UDP

Receptor:

- calcula soma de verificação do segmento recebido
- verifica se soma de verificação calculada é igual ao valor do campo de soma de verificação:
  - NÃO – erro detectado
  - SIM – nenhum erro detectado. *Mas pode haver erros, apesar disso?*

CRC (Cyclic Redundancy Check)

• CRC é um código detector de erros que gera um valor expresso em poucos bits para detectar erros de transmissão.

•O CRC é calculado e anexado aos dados transmitidos e depois verificado para confirmar se não ocorreram alterações.

- Simples de implementar em hardware;
- Eficiente em detectar erros causados por ruídos;
- Simples de ser analisado matematicamente.

• O CRC é um tipo de função de hash.

- Mudança em apenas um bit provoca uma mudança no CRC.
- Mesmo mudanças pequenas nos dados levam a CRCs bem diferentes.
- É muito raro que a introdução de erros nos dados não seja detectado pelo CRC (principalmente o CRC32)

Código polinomial ou CRC

- Os códigos polinomiais tratam uma *cadeia de bits* como polinômios de coeficientes 0 e 1.
- **K** *bits* = polinômio $\mathbf{x^{k-1} + x^{k-2} + \ldots x^0}$

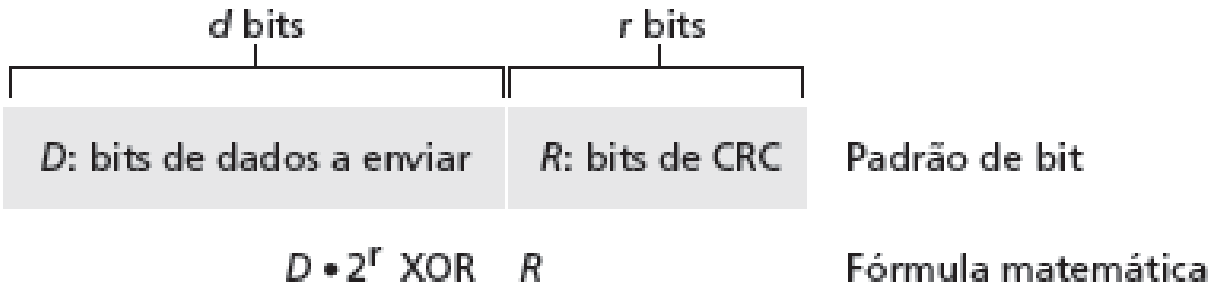
  1 1 0 0 0 1 é representado por $\mathbf{x^5 + x^4 + x^0}$

- Aritmética polinomial em módulo 2 (soma e subtração = XOR)
- Transmissor e receptor devem concordar em relação ao **polinômio gerador G(x).**
- *Mecanismo detecta até r-1 erros!*

CRC (Cyclic Redundancy Check)

Verificação de Redundância Cíclica

- **seja bits de dados, D, como um número binário**
- **escolha padrão de bits r + 1 (gerador), G**
- **objetivo: escolher r bits de CRC, R, tal que**
  - ***<D,R> exatamente divisível por G (módulo 2)***
  - receptor sabe G, divide <D,R> por G. Se resto diferente de zero: erro detectado!
  - pode detectar todos os erros em rajada menores que r + 1 bits
- **muito usada na prática (Ethernet, IEEE 802.11, ATM)**



Exemplo de CRC

Queremos:

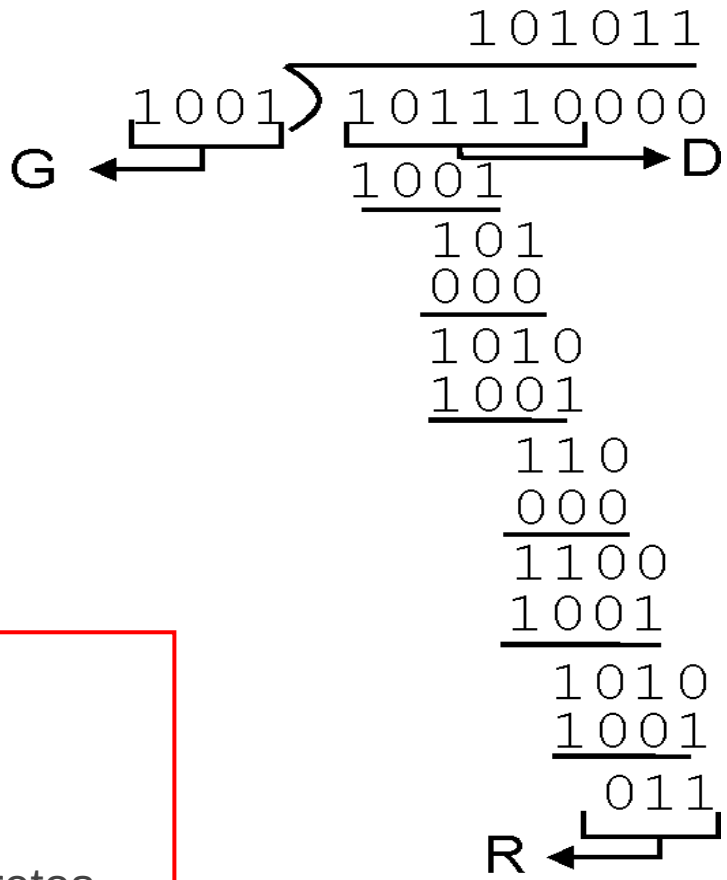
$D \cdot 2^r$ XOR R = nG

de modo equivalente:

$D \cdot 2^r$ = nG XOR R

de modo equivalente:

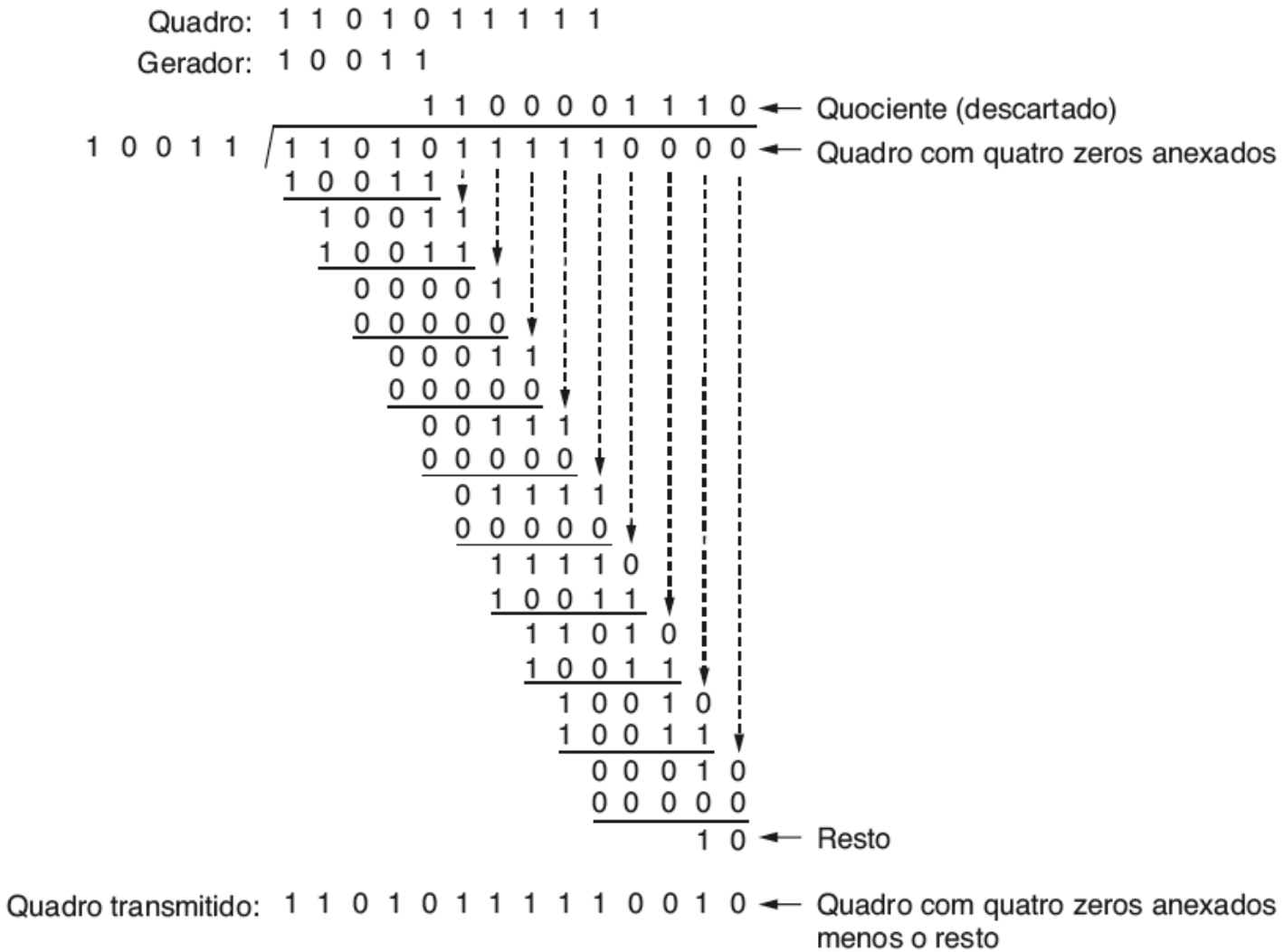
se dividirmos $D \cdot 2^r$ por G, queremos resto R



Solução: (D + R) / G.

Se resto ≠ 0 → ocorreu erro
Senão, dados aceitos como corretos

Exemplo de CRC



Exemplo de cálculo de CRC.

Polinômios Geradores

- Existem alguns polinômios geradores padronizados usados no cálculo de CRC. Os mais comuns são:

- **CRC-32:** $CRC_{32}(x) = x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x^1+x^0$
  - polinômio gerador acima ou *100000100110000010001110110110111* em binário.

- **CRC-16:** utiliza o polinômio gerador $CRC_{16}(x) = x^{16} + x^{15} + x^2 + x^0$ ou *1100000000000101* em notação binária.

- **CRC-12** que usa o polinômio gerador $CRC_{12}(x) = x^{12} + x^3 + x^1 + x^0$ ou *1000000001011* em notação binária.

- **CRC-8** que usa o polinômio gerador $CRC_8(x) = x^8 + x^2 + x^1 + x^0$ ou *100000111* em notação binária.

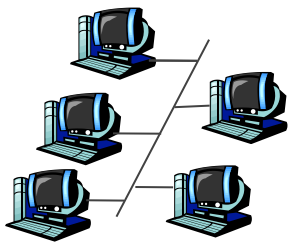
Camada de enlace

- 5.1 Introdução e serviços
- 5.2 Detecção e correção de erros
- 5.3 Protocolos de acesso múltiplo
- 5.4 Endereçamento na camada de enlace
- 5.5 Ethernet

Enlaces e protocolos de acesso múltiplo

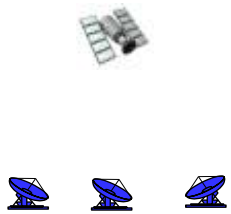
Dois tipos de "enlaces":

- **ponto a ponto**
  - PPP para acesso discado
  - enlace ponto a ponto entre comutador Ethernet e hospedeiro
- **broadcast (fio ou meio compartilhado)**
  - Ethernet à moda antiga
  - HFC
  - LAN sem fio 802.11



fio compartilhado (p. e., Ethernet cabeado)

RF compartilhada (p. e., WiFi 802.11)



RF compartilhada (satélite)

humanos em uma festa (ar e acústica compartilhados)

Protocolos de acesso múltiplo

- **único canal de broadcast compartilhado**
- **duas ou mais transmissões simultâneas por nós:**
  - colisão se o nó recebe dois ou mais sinais ao mesmo tempo
  - Quando ocorre colisão de pacotes os quadros ficam embaralhados...

protocolo de acesso múltiplo

- Deve existir um algoritmo distribuído que determina como os nós compartilham canal,
  - ou seja, determinam quando o nó pode transmitir.

Protocolo de acesso múltiplo ideal

Características desejáveis em um protocolo de acesso múltiplo

Canal de broadcast de velocidade X bps

1. quando um nó quer transmitir, ele pode enviar na velocidade X.
2. quando M nós querem transmitir, cada um pode enviar na velocidade média de transmissão R/M
3. totalmente descentralizado:
   - *nenhum nó especial para coordenar transmissões*
   - *nenhuma sincronização de clocks, intervalos*
4. *Simples → implementação barata!*
5. *Evitar ao máximo o desperdício de intervalos de tempo.*

Protocolos MAC: uma taxonomia

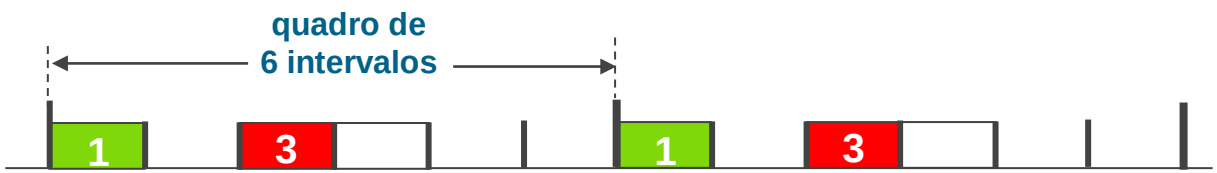
Três classes gerais:

- **Particionamento de canal**
  - divide o canal em “pedaços menores” (intervalos de tempo, frequência, código)
  - aloca pedaço ao nó para uso exclusivo
- **Acesso aleatório**
  - canal não dividido, permite colisões
  - “recupera” de colisões
- **“Revezando”**
  - os nós se revezam, mas os nós com mais a enviar podem receber mais tempo

Time Division Multiple Access

TDMA: Time Division Multiple Access

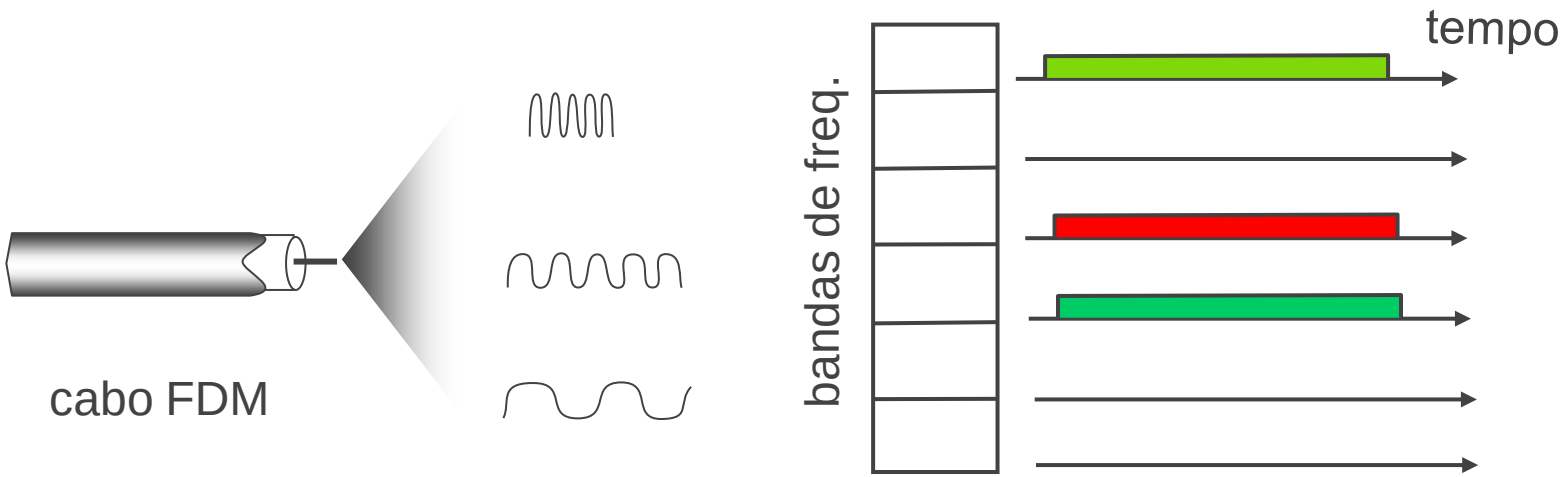
- acesso ao canal em “rodadas”
- cada estação recebe intervalo de tamanho fixo (tamanho = tempo transm. pacote) a cada rodada
- intervalos não usados ficam ociosos
- exemplo: LAN de 6 estações, 1, 3, 4 têm pacote, intervalos 2, 5, 6 ociosos



Frequency Division Multiple Access

FDMA: Frequency Division Multiple Access

- espectro do canal dividido em bandas de frequência
- cada estação recebe banda de frequência fixa
- tempo de transmissão não usado nas bandas de frequência fica ocioso
- exemplo: LAN de 6 estações, 1, 3, 4 têm pacote, bandas de frequência 2, 5, 6 ociosas

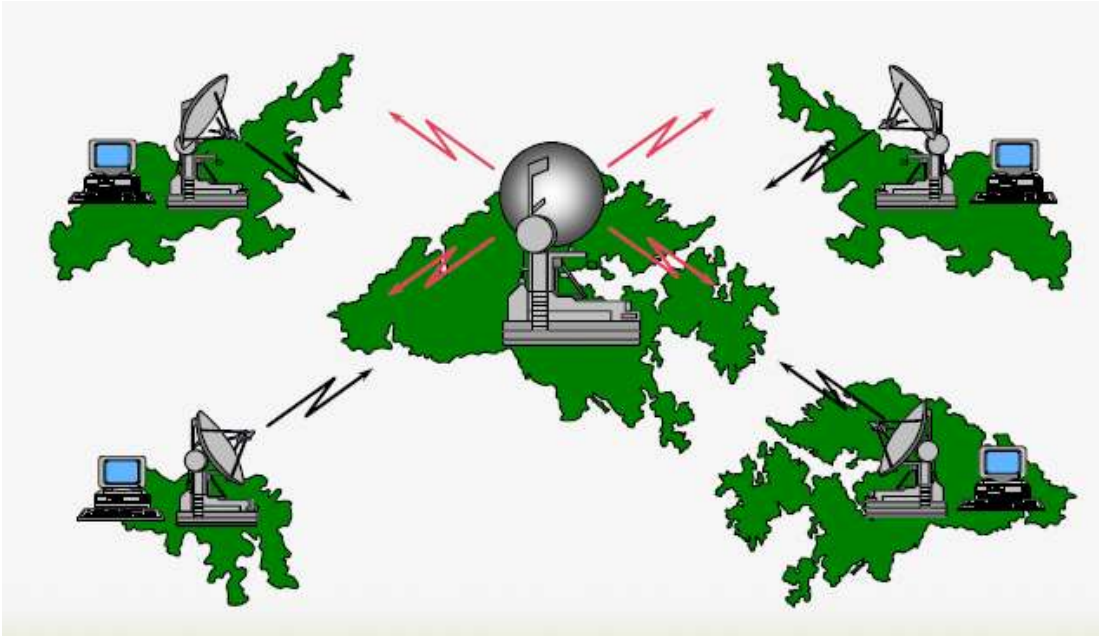


Protocolos de acesso aleatório

- **Quando o nó tem um pacote a enviar**
  - A estação que está transmitindo **utiliza toda a banda**.
  - sem coordenação *a priori* entre os nós
- dois ou mais nós transmitindo simultaneamente ➜ **"colisão",**
- **protocolo MAC de acesso aleatório especifica:**
  - como detectar colisões
  - como recuperar-se de colisões (p. e., via retransmissões adiadas)
- **Exemplos de protocolos MAC de acesso aleatório:**
  - ALOHA
  - Slotted ALOHA
  - CSMA, CSMA/CD, CSMA/CA

ALOHA Puro

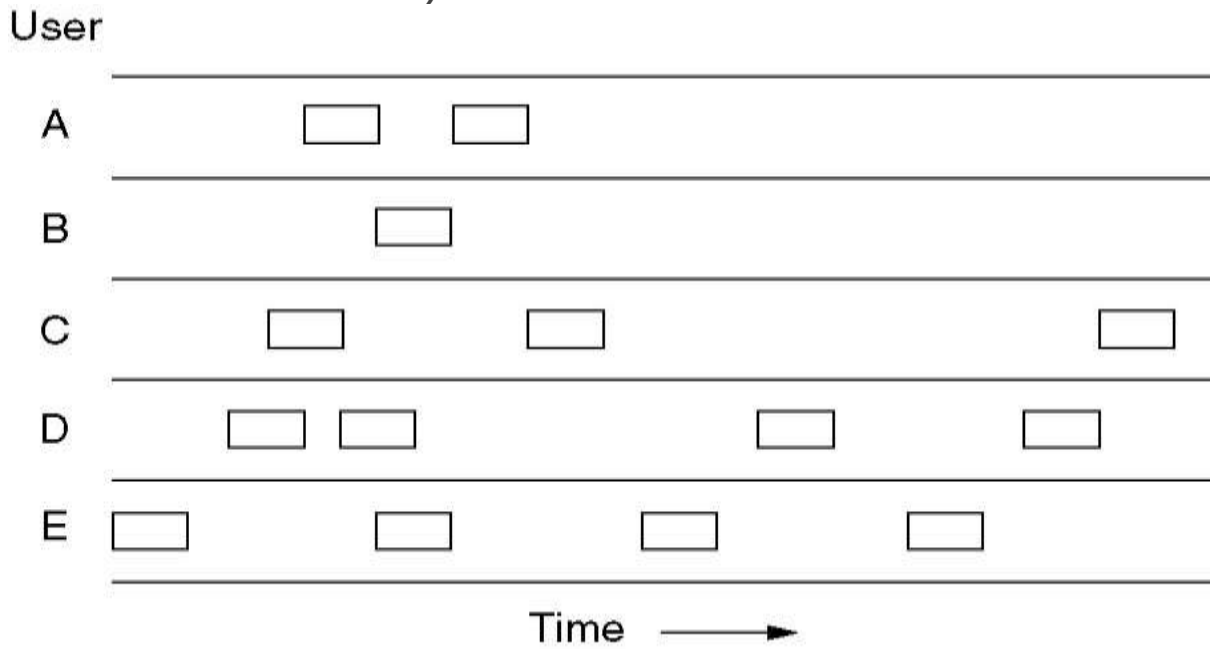
- Universidade do Havaí, comunicação sem fio por ondas de rádio
  - Rede Aloha começou em 1970;
  - Nó mestre em Honolulu (centro de computação)
  - escravos na outras ilhas do arquipélago
  - 2 faixas de freqüência:
    - *do mestre para os escravos (apenas 1 transmissor - não tem colisão)*
    - *dos escravos para o mestre (canal compartilhado - pode ter colisão)*



AlohaNet

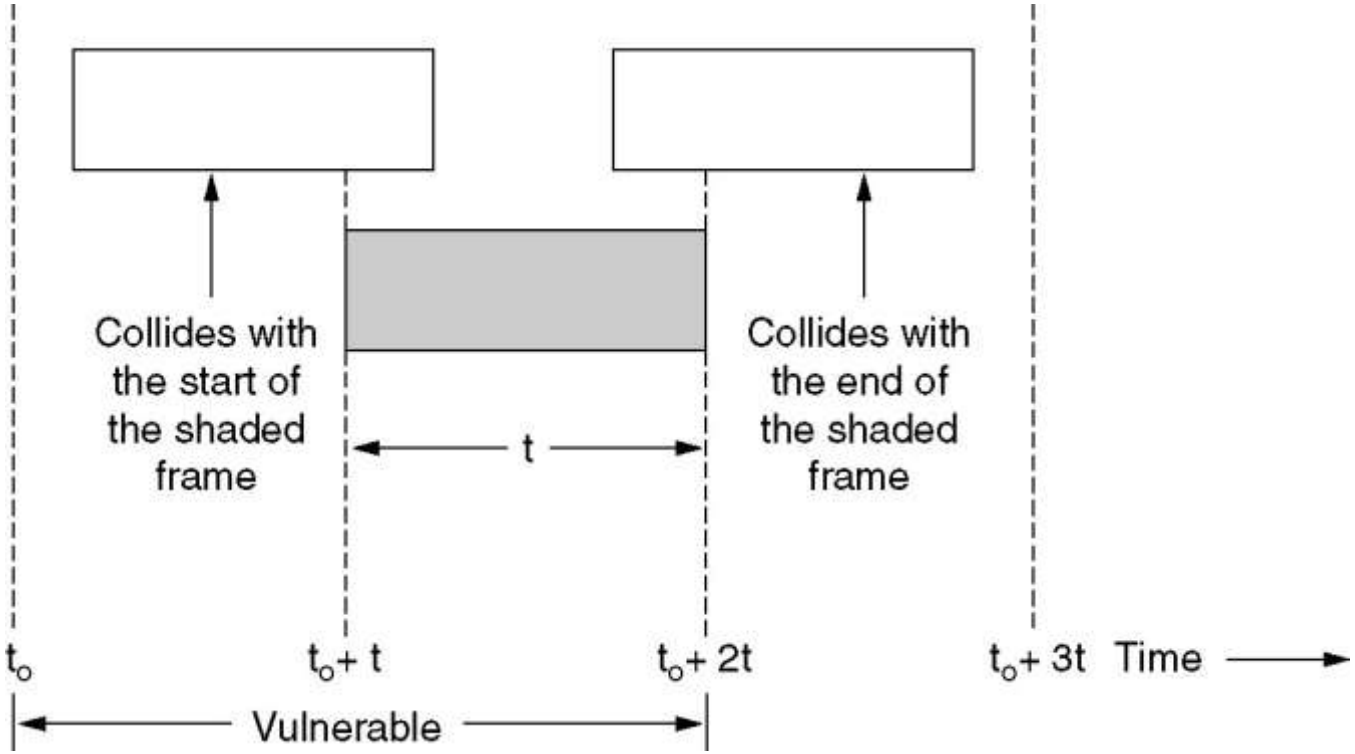
ALOHA Puro

- Usuários transmitem sempre que possuem dados
- Um transmissor pode detectar colisões (usando CRC)
- Retransmissão ocorre após um período de tempo aleatório
- **Problema:** Baixa eficiência (18% dos quadros são transmitidos corretamente no melhor caso)



Quadros são transmitidos em instantes de tempo arbitrários

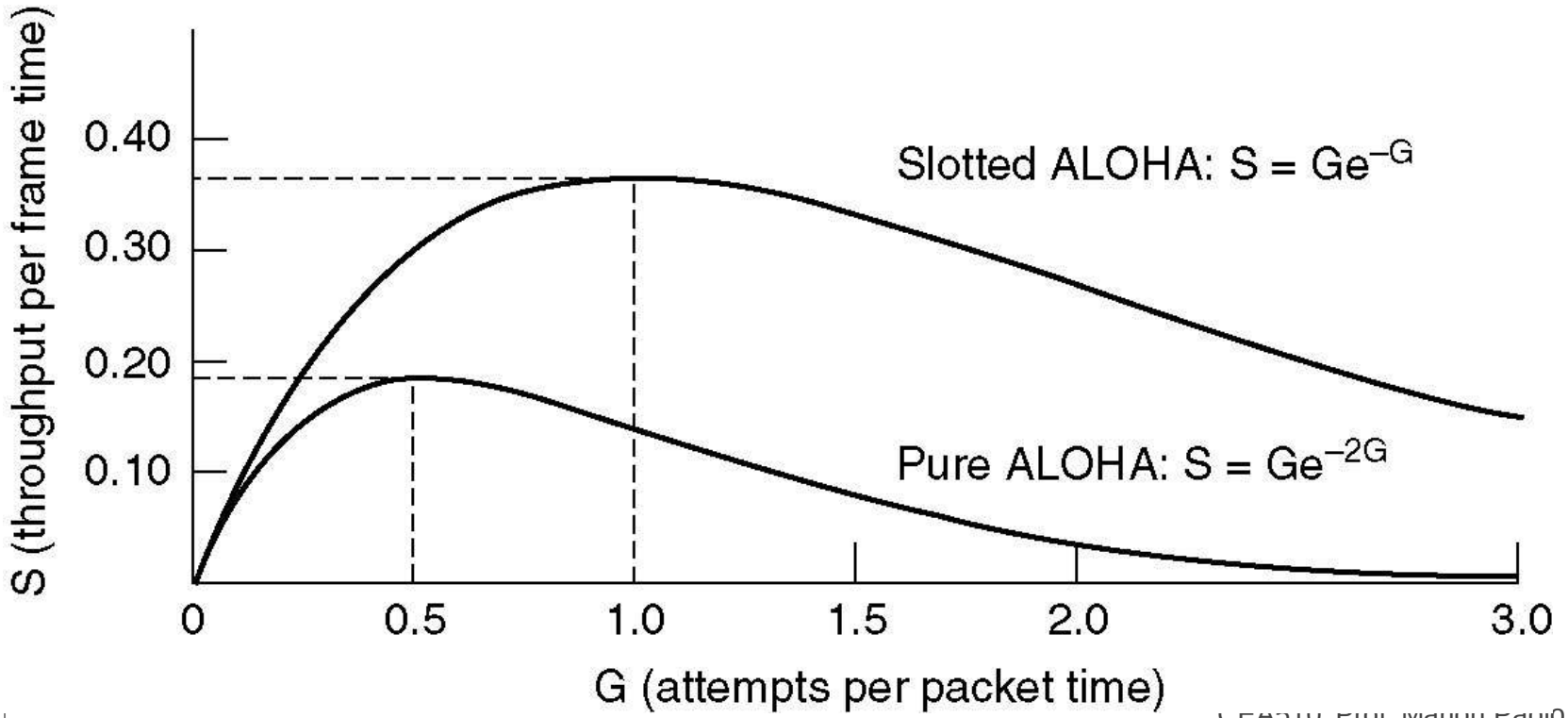
ALOHA Puro



Período de vulnerabilidade para o quadro sombreado

Slotted ALOHA

- **Necessidade de melhorar o desempenho de um sistema ALOHA**
- **Estações só podem transmitir em instantes de tempo específicos**
  - Necessário um relógio mestre
- **Desempenho é o dobro do ALOHA puro (37%)**



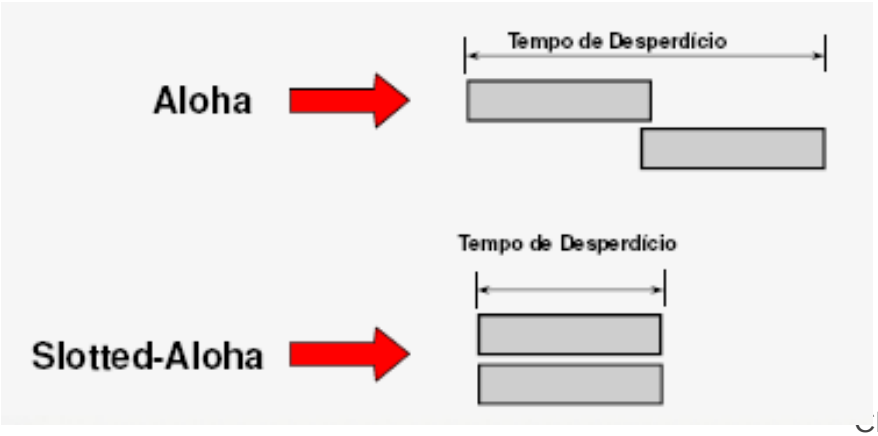
Slotted ALOHA

Suposições:

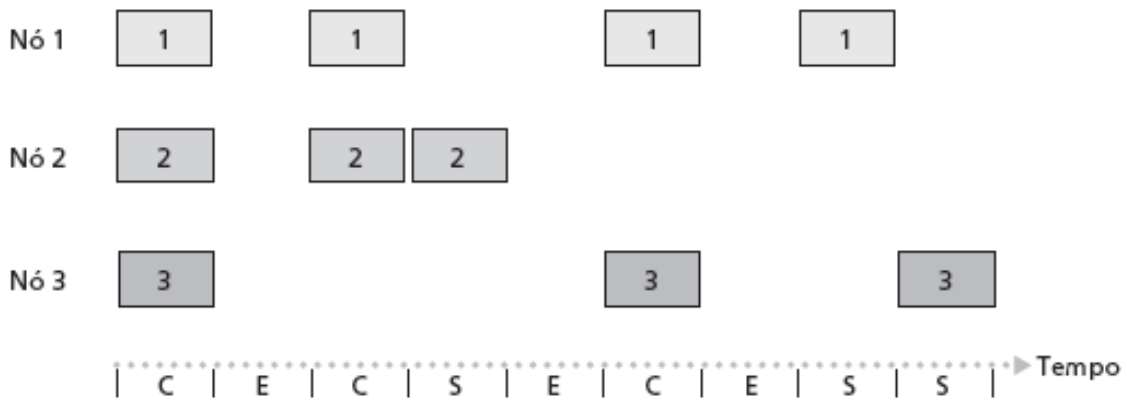
- quadros do mesmo tamanho
- tempo dividido em intervalos de mesmo tamanho (tempo para transmitir 1 quadro)
- nós começam a transmitir somente no início dos intervalos
- nós são sincronizados
- se 2 ou mais nós transmitem no intervalo, todos os nós detectam colisão.

Operação:

- **quando nó obtém quadro novo, transmite no próximo intervalo**
  - ***se não há colisão:*** nó pode enviar novo quadro no próximo intervalo;
  - ***se há colisão:*** nó retransmite quadro em cada intervalo subsequente com prob. até que haja sucesso.



Slotted ALOHA



Prós

- único nó ativo pode transmitir continuamente na velocidade plena do canal
- altamente descentralizado: somente intervalos nos nós precisam estar em sincronismo
- simples

Contras

- colisões, intervalos desperdiçados
- intervalos ociosos
- nós podem ser capazes de detectar colisão em menos tempo do que para transmitir pacote
- sincronismo de clock

CSMA (Carrier Sense Multiple Access)

Analogia do Aloha: uma reunião com pessoas mal-educadas e tagarelas!

Duas regras importantes em uma comunicação usadas pelo CSMA:

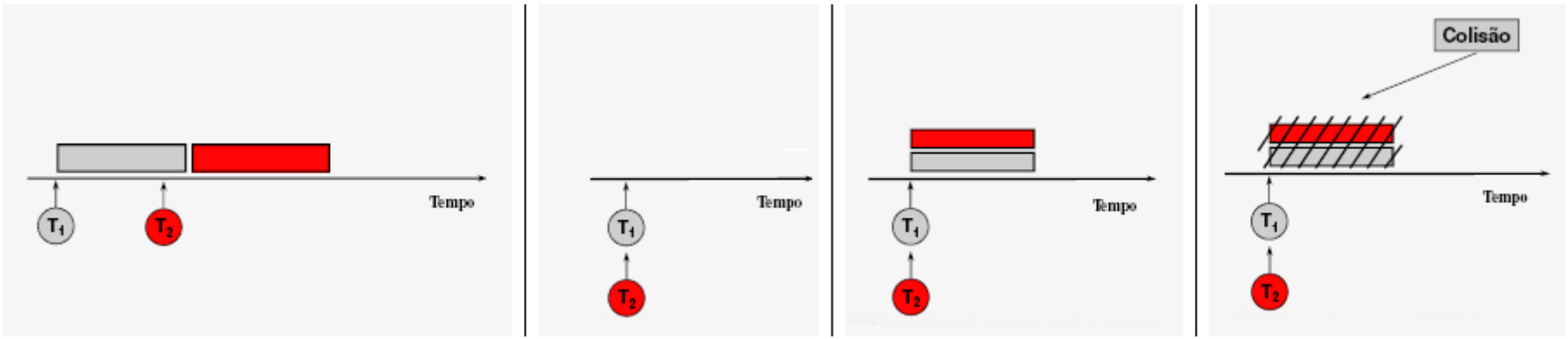
- **ouça antes de falar:**
  - No mundo das redes significa **sensoriamento de portadora**;
  - se perceber canal ocioso: transmite quadro inteiro
- **Se alguém falar ao mesmo tempo que você, pare de falar:**
  - No mundo das redes significa **detecção de colisão**;
  - se perceber canal ocupado, adia transmissão;
  - Não interromper a transmissão

Colisões CSMA

Se todos os nós realizarem detecção de portadora e seguirem as regras ainda poderá ocorrer colisões?

SIM:

atraso de propagação significa que dois nós podem não ouvir a transmissão um do outro.

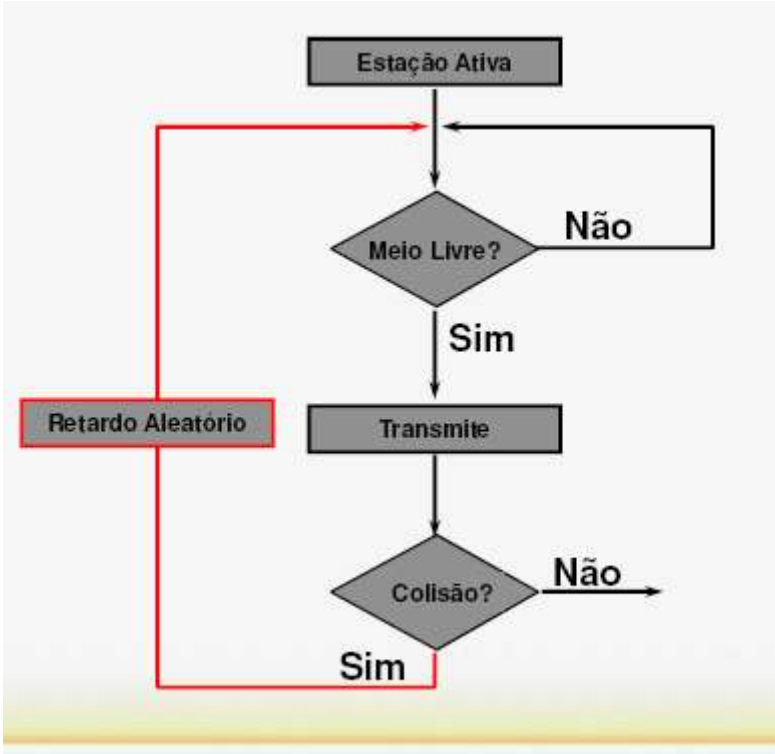


nota:

papel da distância & atraso de propagação determinando probabilidade de colisão

Colisões CSMA

Algoritmo CSMA

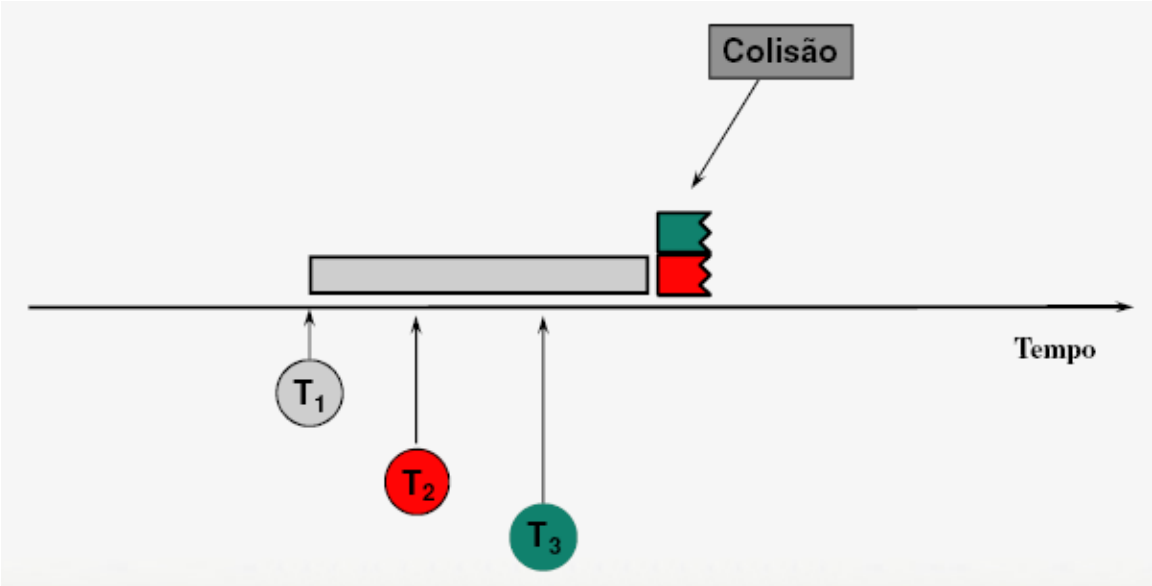


CSMA/CD (Collision Detection)

analogia humana: o interlocutor educado

CSMA/CD:

- colisões *detectadas* dentro de pouco tempo
- transmissões colidindo abortadas, reduzindo desperdício do canal.



CSMA/CD (Collision Detection)

CSMA/CD - detecção de colisão:

- **fácil em LANs com fio:** mede intensidades de sinal, compara sinais transmitidos, recebidos
- **difícil nas LANs sem fio:** intensidade do sinal recebido abafada pela intensidade da transmissão local

- CSMA/CD: utilizado em redes Ethernet

CSMA/CD (Collision Detection)

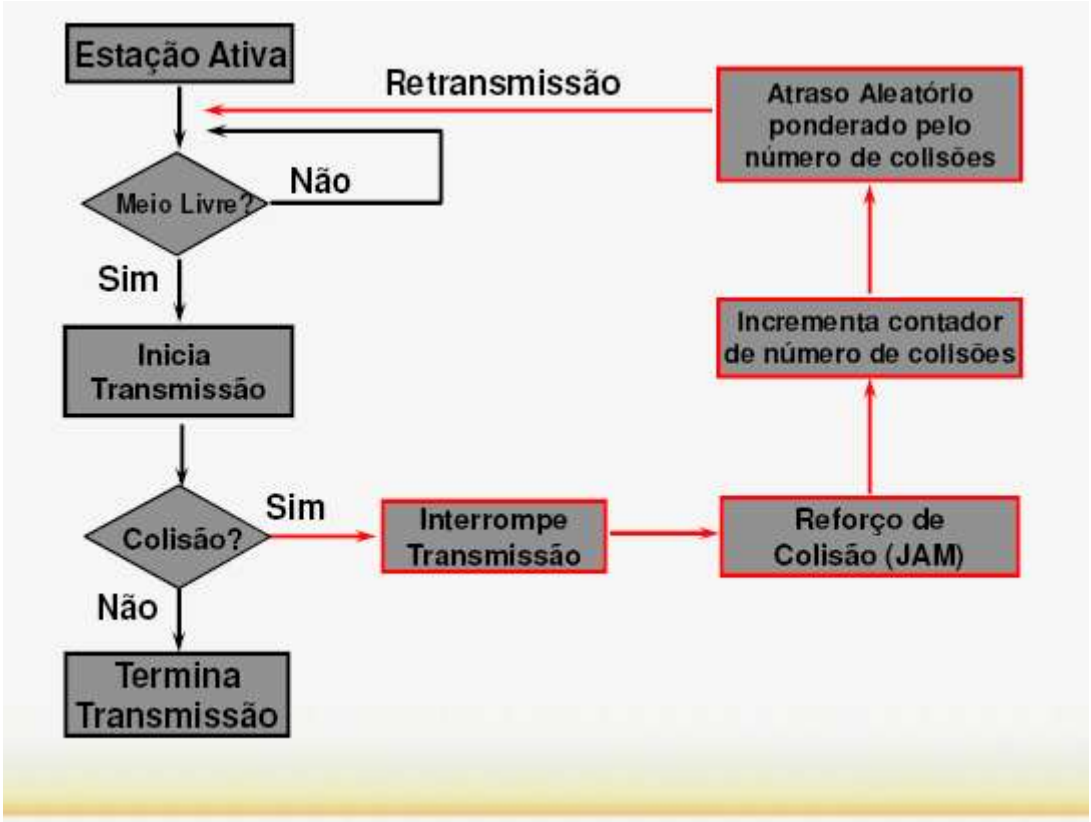
- **CSMA/CD**
  - Se houve colisão, espera tempo aleatório entre 0 e um limite
  - O limite é dobrado a cada colisão sucessiva até um número máximo de colisões estipulado.
    - A partir daí, o limite permanece inalterado por mais um certo número de tentativas máximo estipulado.
      - Se não conseguir transmitir até o número máximo de tentativas, então aborta a transmissão

- **No padrão IEEE 802.3:**

  –O limite dobra até 10 tentativas, depois permanece inalterado até no máximo 16 tentativas.

Colisões CSMA

Algoritmo CSMA/CD



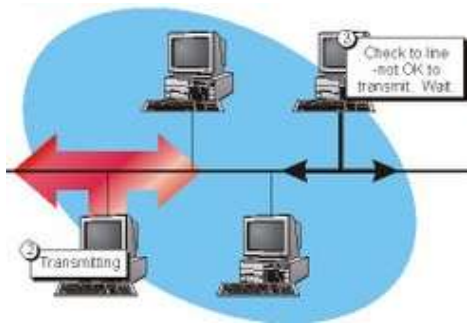
Colisão em redes sem fio

• A detecção de colisões em redes wireless é bem mais complicada que nas redes cabeadas, devido à complexidade inerente do ambiente sem fio.

  • Protocolo CSMA/CD utilizado pelo 802.3, não foi adotado nas redes Wireless.

• Em um ambiente cabeado, todas estações estão fisicamente conectadas, e é possível saber se houve ou não alguma colisão na rede.

• O mesmo não ocorre nas redes sem fio.



Controle de Acesso ao Meio

•Problema da estação oculta:

•Neste caso, C e D não estão no alcance do rádio de A.

•Ele “escutará” o meio e concluirá erradamente que este está livre, e iniciará uma transmissão com B.

CSMA/CA

• Para permitir a construção de redes com muitos computadores, a solução adotada foi utilizar um mecanismos de prevenção de colisão:

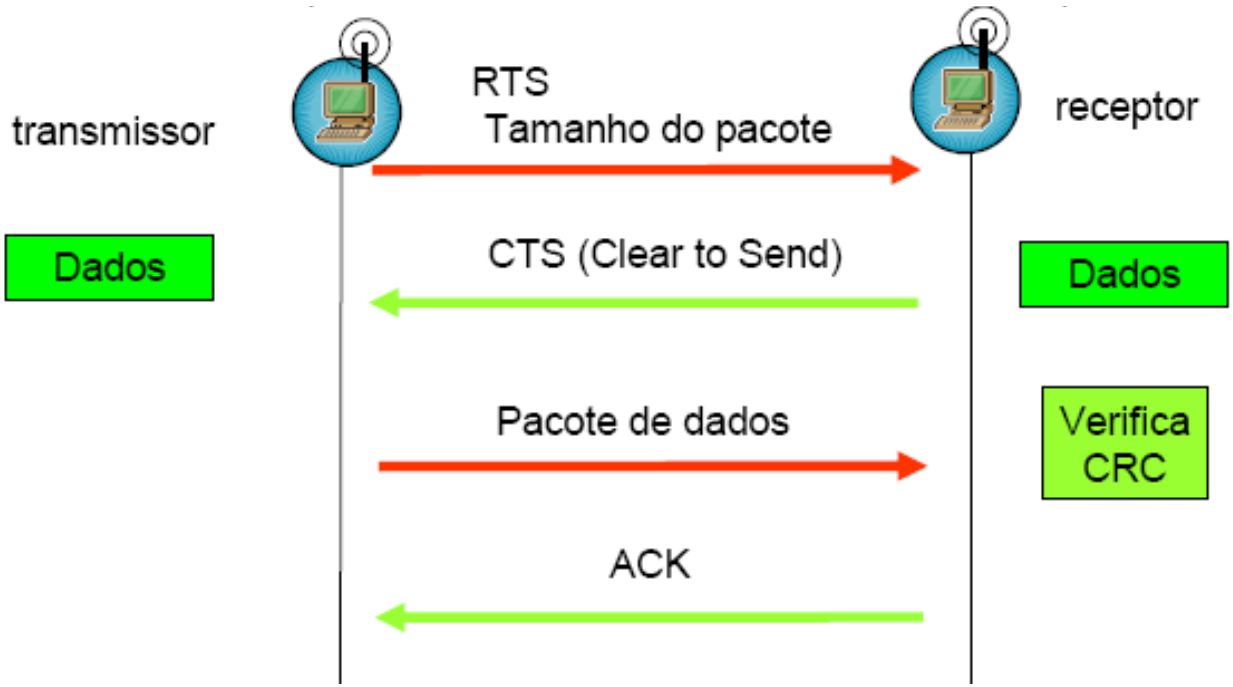
• O CSMA/CA é um método de transmissão que possui um grau de ordenação maior que o seu antecessor (CSMA/CD).

• O CSMA/CA possui também mais parâmetros restritivos, o que contribui para a redução da ocorrência de colisões em uma rede.

CSMA/CA

•Antes de transmitir efetivamente um pacote, a estação avisa sobre a transmissão e em quanto tempo a mesma irá realizar a tarefa.

• Dessa forma, as estações não tentarão transmitir, porque entendem que o canal está sendo usado por outra máquina.



CSMA/CA

FUNCIONAMENTO:

•Ao invés de enviar os dados aleatoriamente, o CSMA/CA permite ao transmissor “reservar” o canal antes.

- •Antes de enviar dados, transmissor escuta o meio (sensoriam. da portadora), buscando sinal de RF de outra estação.

•Se o meio estiver ocupado, aguarda até nova verificação.

•Se o meio estiver livre, envia quadro chamado *request to send* (RTS) à estação-base indicando a sua intenção de transmitir.

CSMA/CA

FUNCIONAMENTO:

• Ao receber o RTS, o AP envia em broadcast (*Clear To Send* – CTS) autorizando o hots enviar dados e reservando um espaço de tempo para sua transmissão.

• Enquanto isso as outras estações aguardam para iniciar suas transmissões.

Desta forma o protocolo evita colisões de quadros de dados usando pequenos quadros de reserva de banda!

• **Se o ACK é recebido, o pacote é considerado perdido e é realizada uma retransmissão deste pacote.**

“Revezando” protocolos MAC

protocolos MAC de particionamento de canal:

- compartilham canal de modo *eficaz* e *justo* com alta carga
- ineficaz com baixa carga: atraso no acesso ao canal, 1/N largura de banda alocada mesmo que apenas 1 nó ativo!

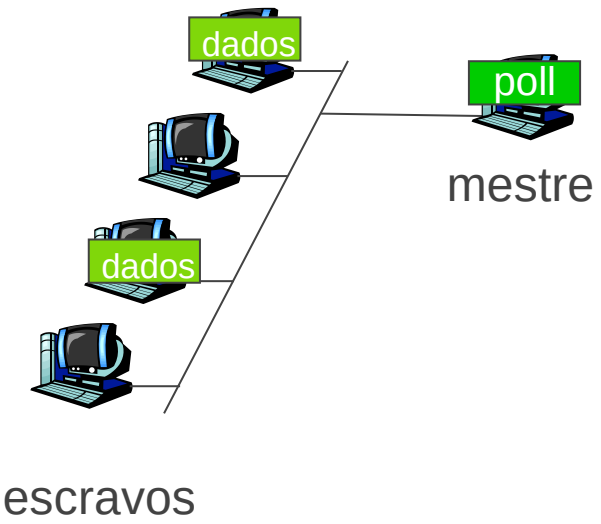
Protocolos MAC de acesso aleatório

- eficaz com baixa carga: único nó pode utilizar o canal totalmente
- alta carga: sobrecarga de colisão

“revezando” protocolos

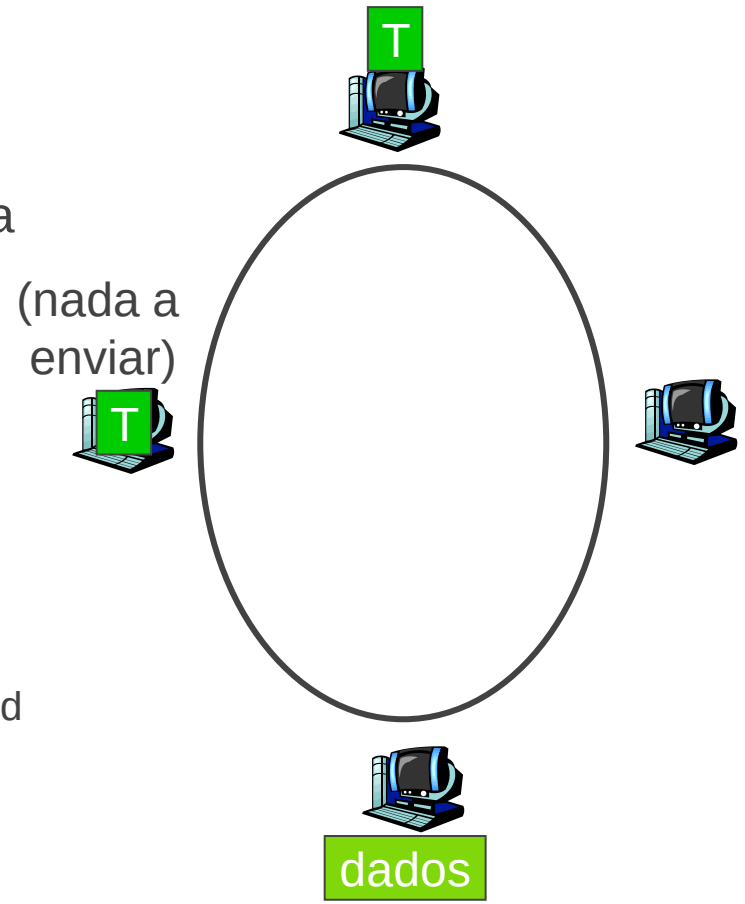
Polling (seleção)

- nó mestre “convida” nós escravos a alterarem a transmissão;
  - normalmente usado com dispositivos escravos “burros”;
- Seleciona cada um dos nós para transmitir por alternância circular
  - Envia msg para nó transmissor permitindo-o transmitir a o máximo de x quadros.
- preocupações:
  - sobrecarga da seleção
  - latência
  - único ponto de falha (mestre)
- **Exemplo:** Bluetooth



Passagem de permissão

- Não há nós mestres.
- **permissão** de controle passada de um nó para o próximo sequencialmente.
  - Nó retém mensagem de permissão apenas se tiver dados para enviar.
- Passagem de permissão descentralizada e com alta eficiência.
- preocupações
  - sobrecarga da permissão
  - latência
  - único ponto de falha (permissão) derruba toda a rede
- Usado em protocolos FDDI (Fiber Distributed Data Interface) e IEEE 802.5



Resumo de protocolos MAC

- ***particionamento de canal,*** **por tempo, frequência ou código**
  - Time Division, Frequency Division
- ***acesso aleatório*** **(dinâmico),**
  - ALOHA, S-ALOHA, CSMA, CSMA/CD
  - percepção de portadora: fácil em algumas tecnologias (com fio), difícil em outras (sem fio)
  - CSMA/CD usado na Ethernet
  - CSMA/CA usado na 802.11
- ***Revezamento***
  - passagem de permissão e passagem de permissão: Bluetooth, FDDI, Token Ring