

PTC 3360 - 2025

3.2 Controle de acesso ao canal compartilhado

3.3 Endereçamento MAC e *switches*

(Kurose, Seções 6.3 e 6.4)

Setembro 2025

CAPÍTULO 3. Camada de enlace e física

3.1 Introdução

3.2 Controle de acesso ao canal compartilhado

3.2.1 Particionamento de canal

3.2.2 Acesso Aleatório

A *Slotted* ALOHA

B ALOHA puro

C CSMA

D Exemplos: Ethernet e Wi-Fi

3.2.3 Revezamento

3.3 Endereçamento MAC e *switches*

3.4 Camada física: meios de transmissão

3.5 Rádio Enlaces

CSMA (*Carrier Sense Multiple Access*)

CSMA: nó “escuta” canal antes de transmitir:

Ideia básica:

- ❖ se detecta que o canal está livre: transmitir quadro inteiro
- ❖ se detecta que canal está ocupado, adia transmissão

- ❖ Analogia humana:
 - ALOHA: Começar a falar independentemente de outros estarem falando
 - CSMA:
 - Escute antes de falar – se alguém está falando, espere ele terminar – **C**ollision **A**voidance (CSMA puro ou CSMA-**CA**) – Wi-Fi e outros enlaces sem fio
 - Se, além disso, alguém começar a falar ao mesmo tempo – pare de falar – **C**ollision **D**etection (CSMA-**CD**) – Ethernet e outros enlaces cabeados

Colisões CSMA-CA

- ❖ **Colisões ainda podem ocorrer:** atraso de propagação faz com que 2 nós possam não escutar a transmissão um do outro no momento de começar a transmitir

time
↓

distribuição espacial dos nós



t_1

- ❖ **Colisão:** tempo de transmissão de pacote inteiro é perdido
 - distância e atraso de propagação desempenham papel na determinação da probabilidade de colisão

I

CSMA-CD (*Detecção de Colisão*)

CSMA-CD: monitoração de portadora, adiamento como no CSMA

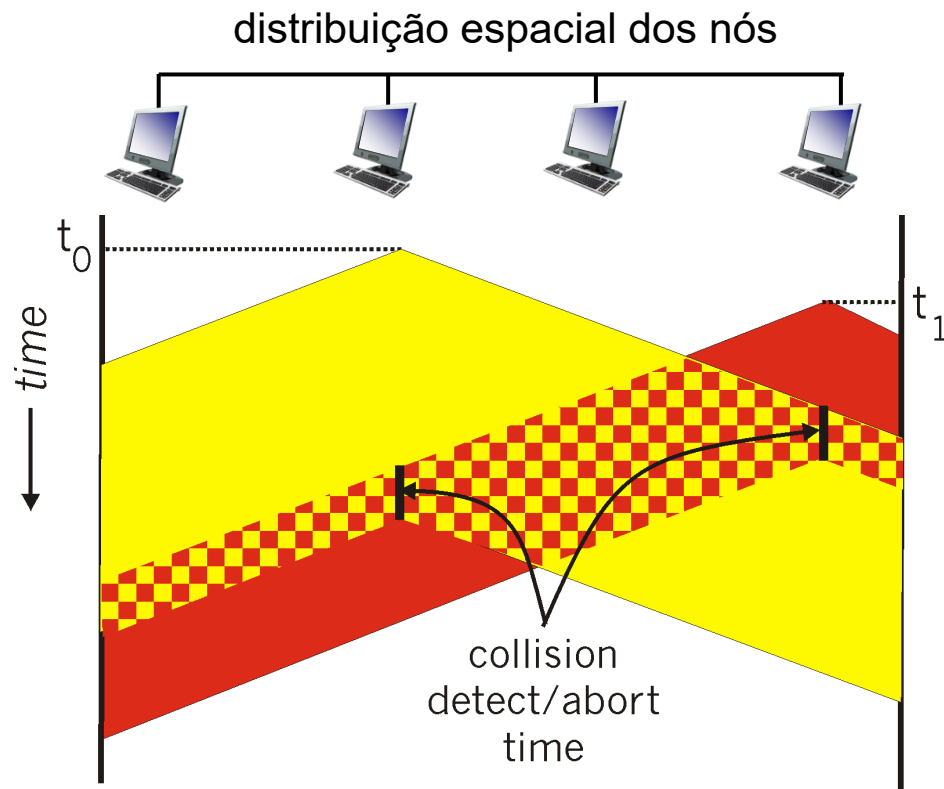
- Colisões *detectadas* em curto tempo
- Transmissões colidindo abortadas, reduzindo desperdício de canal

❖ Detecção de colisão:

- **Fácil em LANs cabeadas:** medir intensidade dos sinais, comparar sinais transmitido e recebido
- **Difícil em LANs sem fio:** intensidade do sinal recebido dominada pela intensidade da transmissão local

❖ Analogia humana : o conversador educado

CSMA/CD (detecção de colisão)



Eficiência do CSMA/CD

- ❖ d_{prop} = máximo atraso de propagação entre 2 nós na LAN
- ❖ d_{trans} = tempo para transmitir quadro de máximo comprimento (aprox. 1,2 ms para uma Ethernet a 10 Mbps)

$$E_{max} = \frac{1}{1 + 5 \frac{d_{prop}}{d_{trans}}}$$

Exercício: $E_{max} \rightarrow 1$ quando $d_{prop} \rightarrow 0$ ou quando $d_{trans} \rightarrow +\infty$. Explique porquê isso faz sentido.

- ❖ Melhor desempenho do que ALOHA em condição de tráfego elevado.

CAPÍTULO 3. Camada de enlace e física

3.1 Introdução

3.2 Controle de acesso ao canal compartilhado

3.2.1 Particionamento de canal

3.2.2 Acesso Aleatório

A *Slotted* ALOHA

B ALOHA puro

C CSMA

D Exemplos: Ethernet e Wi-Fi

3.2.3 Revezamento

3.3 Endereçamento MAC e *switches*

3.4 Camada física: meios de transmissão

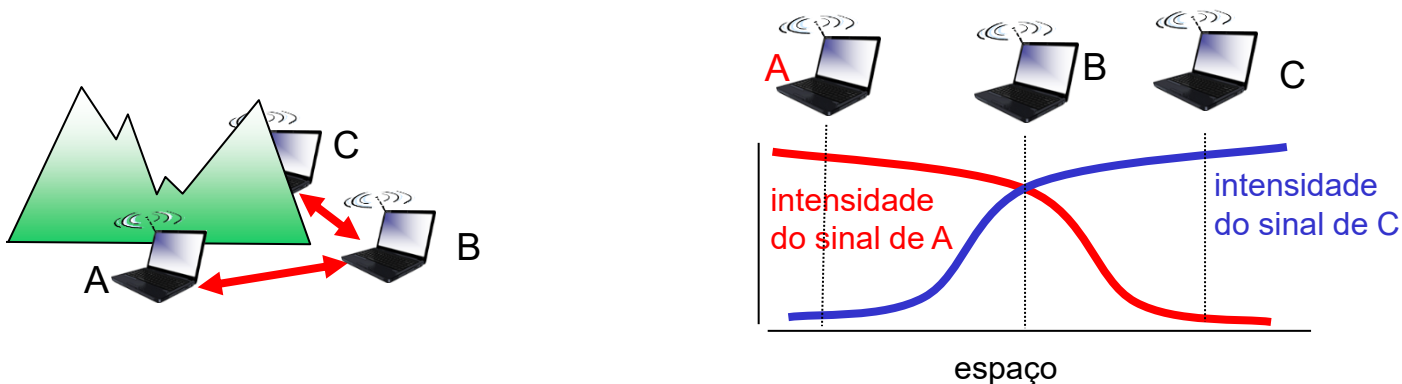
3.5 Rádio Enlaces

Exemplo 1: Algoritmo CSMA-CD do Ethernet

1. Camada de enlace recebe datagrama da camada de rede, cria quadro
2. Se detecta canal livre, inicia transmissão do quadro. Se detecta canal ocupado, espera até canal livre e então transmite.
3. Se transmite quadro inteiro sem detectar outra transmissão, missão cumprida!
4. Se detecta uma outra transmissão enquanto transmite, aborta e envia sinal de congestionamento
5. Depois de abortar, entra em *backoff exponencial binário*:
 - depois da m -ésima colisão, escolhe K aleatoriamente de $\{0, 1, 2, \dots, 2^m - 1\}$. Espera $K \times 512$ tempos de bit e retorna para Passo 2
 - intervalo de *backoff* mais longo com mais colisões

Exemplo 2: CSMA-CA do WiFi

- ❖ Tenta-se evitar colisões: 2 ou mais nós transmitindo ao mesmo tempo
- ❖ Wi-Fi: *não há* detecção de colisão!
 - Difícil detectar colisões quando transmitindo devido à potência baixa dos sinais recebidos
 - De qualquer forma, não é possível detectar todas as colisões devido ao problema de terminal escondido e ao desvanecimento
 - Objetivo: *evitar colisões*: CSMA/C(ollision)A(voidance)



CAPÍTULO 3. Camada de enlace e física

3.1 Introdução

3.2 Controle de acesso ao canal compartilhado

3.2.1 Particionamento de canal

3.2.2 Acesso Aleatório

A *Slotted* ALOHA

B ALOHA puro

C CSMA

D Exemplos: Ethernet e Wi-Fi

3.2.3 Revezamento

3.3 Endereçamento MAC e *switches*

3.4 Camada física: meios de transmissão

3.5 Rádio Enlaces

Protocolos MAC por revezamento

Protocolos MAC por particionamento de canal

- Compartilha canal *eficientemente* de forma *justa* quando há altas cargas
- Ineficiente com baixas cargas: atraso no acesso ao canal, alocado 1/N da capacidade do canal mesmo que somente 1 nó ativo!

Protocolos MAC por acesso aleatório

- Eficiente com baixas cargas: nó único pode usar a capacidade inteira do canal
- Altas cargas: *overhead* de colisões

Protocolos de revezamento

- Procura pelo melhor dos 2 mundos!
- Muitas propostas. Vamos exemplificar duas: protocolos de seleção (*polling*) e protocolos de passagem de permissão (*token*)

Protocolos MAC por revezamento

Protocolos de seleção:

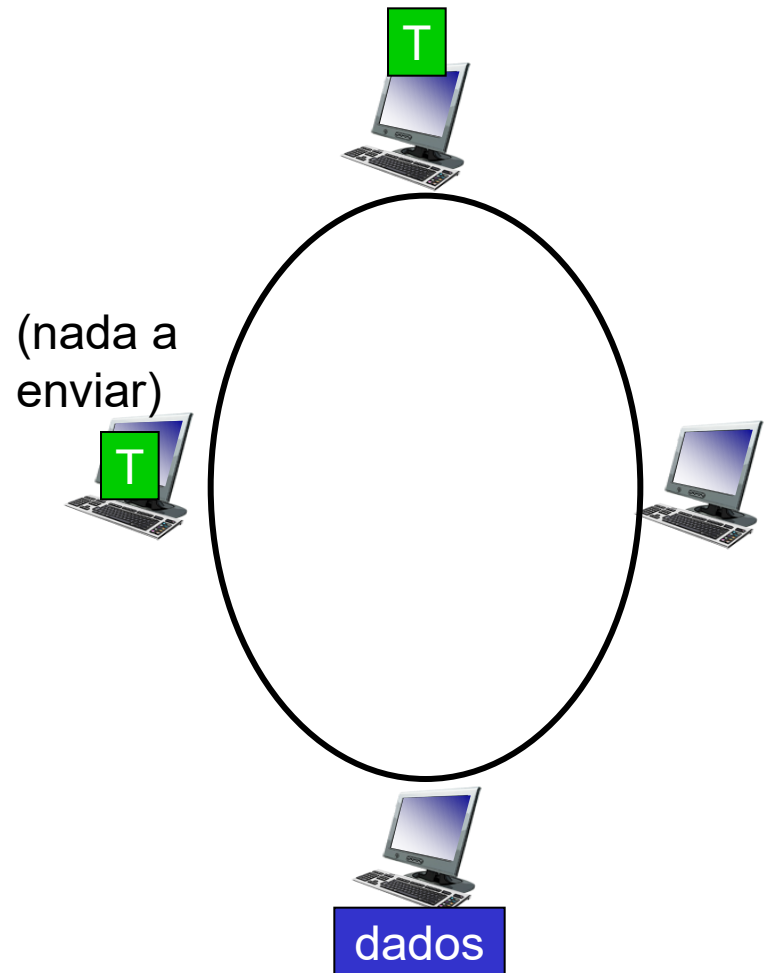
- ❖ Nó mestre “convida” nós escravos a transmitir em turnos
- ❖ Tipicamente usados com dispositivos escravos “burros”
- ❖ Problemas:
 - *Overhead* de seleção
 - Latência
 - Ponto de falha único (mestre)
- ❖ Usado no [Bluetooth](#)



Protocolos MAC por revezamento

Passagem de permissão:

- ❖ *Token* de controle passado de 1 nó para o seguinte sequencialmente.
- ❖ Mensagem *token*
- ❖ Problemas:
 - *Overhead* do token
 - Latência
 - Ponto único de falha (*token*)
- Usado no [IEEE 802.5](#) (*Token Ring*) e [FDDI](#) (*Fiber Distributed Data Interface*)



CAPÍTULO 3. Camada de enlace e física

3.1 Introdução

3.2 Controle de acesso ao canal compartilhado

3.2.1 Particionamento de canal

3.2.2 Acesso Aleatório

A *Slotted* ALOHA

B ALOHA puro

C CSMA

D Exemplos: Ethernet e Wi-Fi

3.2.3 Revezamento

3.3 Endereçamento MAC e switches

3.4 Camada física: meios de transmissão

3.5 Rádio Enlaces

Endereços MAC

❖ Endereço IP:

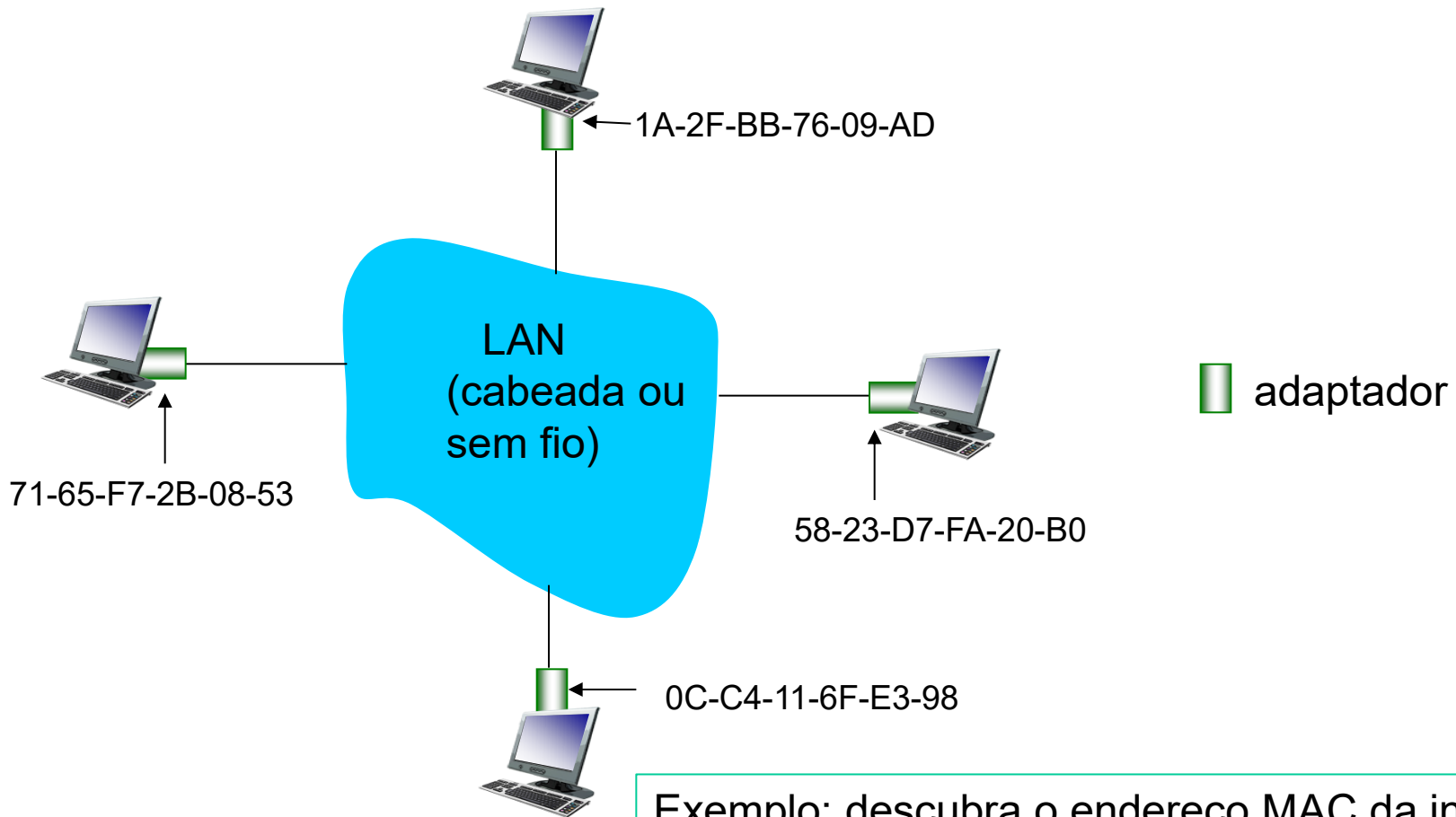
- Endereço de camada de rede para interface
- Usado para repasse e roteamento na camada 3 (*camada de rede*) – *padrão da Internet*

❖ Endereço MAC (*Media Access Control*) ou LAN ou físico ou Ethernet

- Endereço de *camada de enlace* em protocolos como Ethernet e Wi-Fi – *padrão desses protocolos*
- Funções:
 - Identificar unicamente placa de rede de forma global (como um documento de identificação)
 - Usado localmente para levar quadro de uma interface para outra fisicamente conectada (mesma sub-rede, no sentido de endereçamento IP)
- Endereços de 48 bits (6 bytes) gravado na ROM da interface de rede, às vezes também ajustável via *software*
- Exemplo: 1A-2F-BB-76-09-AD ————— *notação hexadecimal (base 16)*
(cada “número” representa 4 bits)

Endereços LAN

Cada adaptador na LAN tem endereço **MAC** único



Exemplo: descubra o endereço MAC da interface Wi-Fi do seu notebook ou celular!

Endereços LAN (mais)

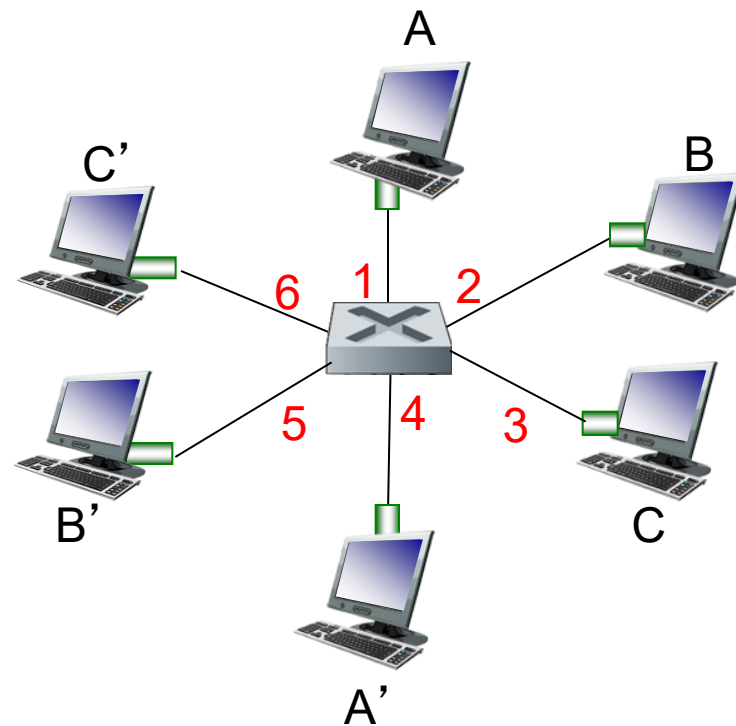
- ❖ Alocação de endereços MAC administrada pelo IEEE
- ❖ Fabricante compra faixa de espaço de endereços MAC (para assegurar unicidade) – conjunto de 2^{12} endereços (primeiros 36 bits fixos) - US\$ 880,00
- ❖ Analogia:
 - Endereço MAC : parecido com CPF
 - Endereço IP : parecido com endereço postal
- ❖ Sem hierarquia, o endereço é portátil ou seja, pode-se mover interface de uma LAN para outra sem mudar o endereço MAC
- ❖ Note o contraste com o endereço IP que é hierárquico e *não portátil*

Switch Ethernet

- ❖ Em português: “comutador da camada de enlace”
- ❖ **Dispositivo da camada de enlace:** com papel **ativo**
 - Só lê endereço MAC (não IP)
 - Filtra, armazena (*buffer*), repassa quadros Ethernet
 - Examina endereço MAC de quadro que chega e sua tabela de comutação; repassa quadro **seletivamente** para um ou mais enlaces de saída
 - Elimina colisões!! Aumento de desempenho considerável em relação a *enlaces com difusão*
 - Aumento na segurança (*sniffers*)
- ❖ **Transparente**
 - *Hosts* não têm ciência da presença de *switches*
- ❖ **Plug-and-play, autoaprendizagem**
 - *Switches* não precisam ser configurados. É só ligar 😊

Switch: múltiplas transmissões simultâneas

- ❖ *Hosts* têm conexão direta, dedicada com *switch*
- ❖ Protocolo Ethernet usado em *cada* enlace que chega, não há colisões entre eles; *full duplex*
 - cada enlace é o seu próprio domínio de colisão
- ❖ **Comutação:** A-para-A' e B-para-B' podem transmitir simultaneamente, sem colisões



Switch com 6 interfaces
(1,2,3,4,5,6)

Tabela de repasse do switch

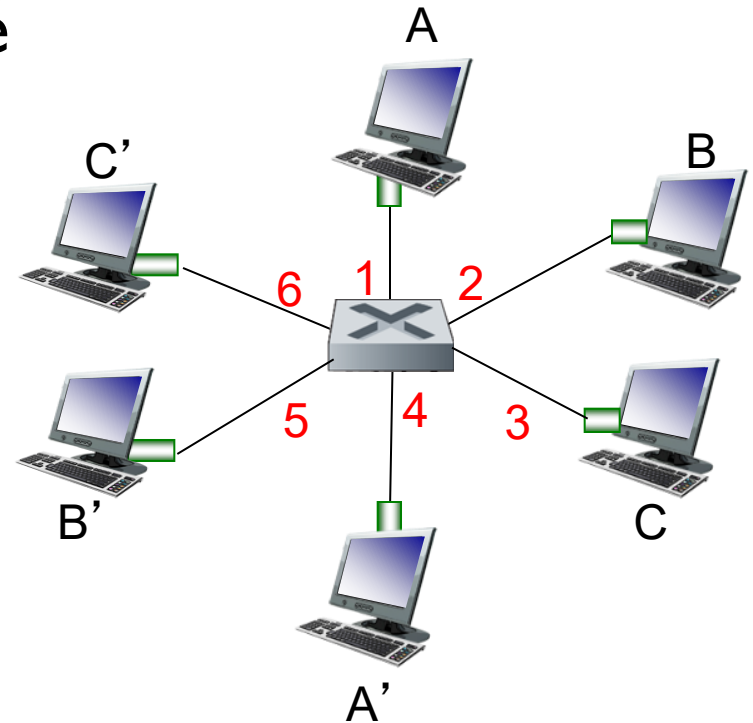
Q: Como o switch sabe que A' é alcançável via interface 4, B' alcançável via interface 5, etc.?

❖ R: Cada switch tem uma **tabela de comutação**, cada entrada:

- (Endereço MAC do host, Interface para alcançar o host, Time stamp)
- Parece uma tabela de roteamento!

Q: Como as entradas são criadas, mantidas na tabela de comutação?

- Algum tipo de protocolo de roteamento? **Não!**



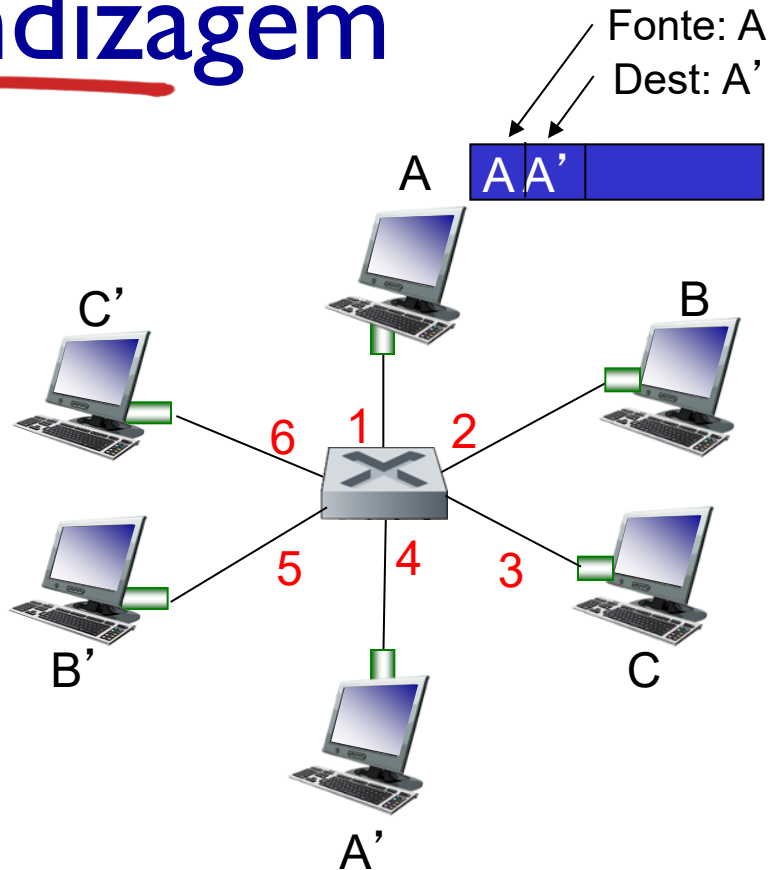
switch com 6 interfaces
(1,2,3,4,5,6)

Switch: autoaprendizagem

- ❖ Switch *aprende* quais *hosts* podem ser alcançados por quais interfaces
 - Quando quadro recebido, switch “aprende” localização do remetente: segmento LAN em que chega
 - Armazena par remetente/localização na tabela de comutação

Q: O que fazer com pacote cujo destino não está na tabela?

R: Mandar para todas as interfaces, menos a que recebeu o pacote.



| End. MAC | interface | TTL |
|----------|-----------|-----|
| A | 1 | 60 |

*Tabela de comutação
(inicialmente vazia)*

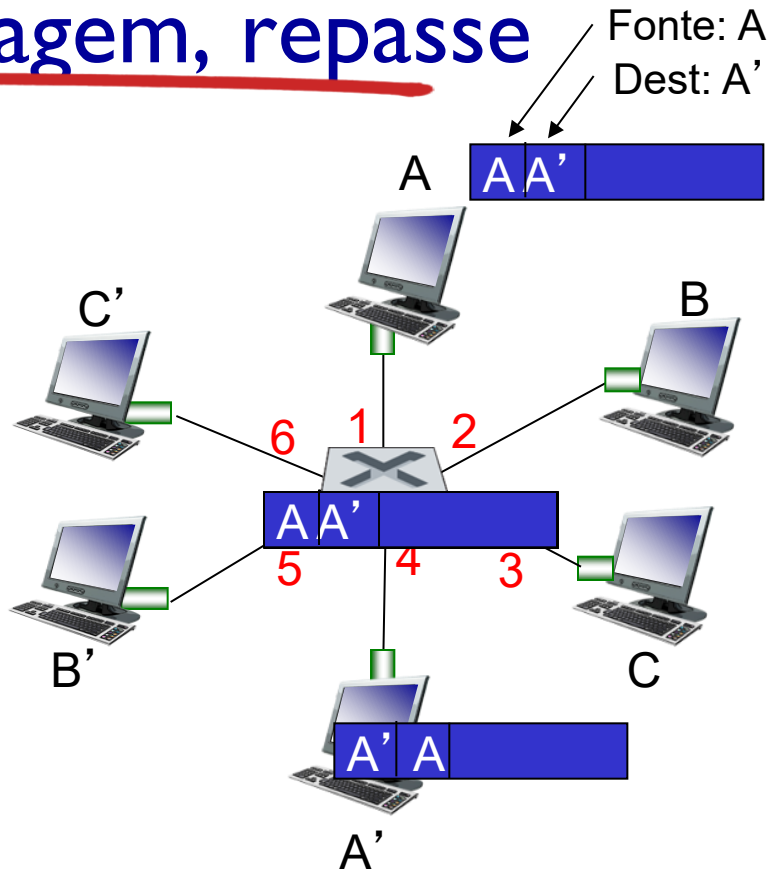
Switch: filtragem/repassse de quadro

Quando quadro é recebido no *switch*:

1. Grava enlace de chegada, endereço MAC do *host* enviando
2. Procura endereço MAC de destino na tabela de comutação
3. Se entrada para destino encontrada
Então {
 Se MAC destino alcançável na interface pelo qual chegou
 Então descarta quadro (*filtra*)
 Senão repassa quadro para interface indicada na tabela
}
Senão inunda /* repassa para todas as interfaces exceto a interface em que o pacote chega */

Exemplo: autoaprendizagem, repasse

- ❖ Destino do quadro, A', localização desconhecida: *inunda*
- ❖ Destino A localização conhecida: *seletivamente envia por apenas um enlace*

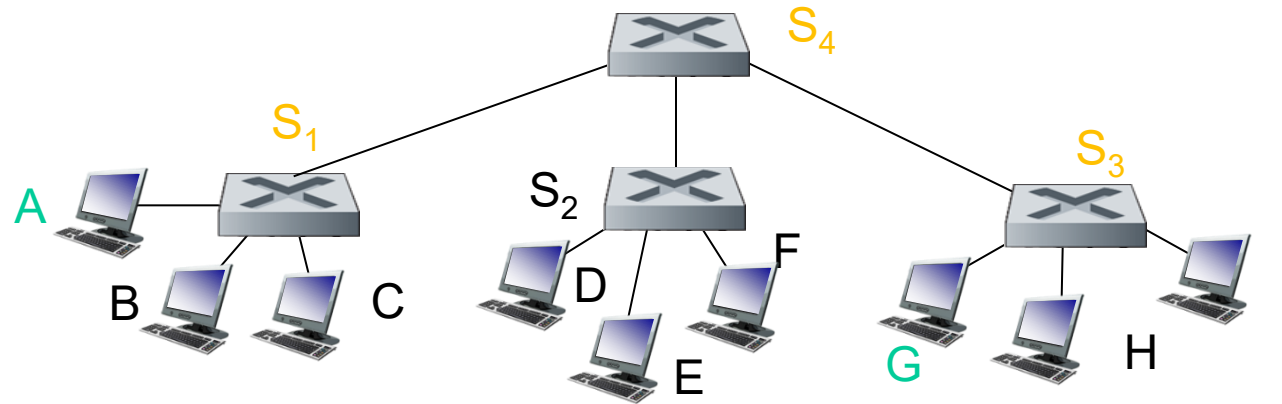


| end. MAC | interface | TTL |
|----------|-----------|-----|
| A | 1 | 60 |
| A' | 4 | 60 |

*Tabela de comutação
(inicialmente vazia)*

Interconectando switches

- ❖ Switches podem ser ligados entre si

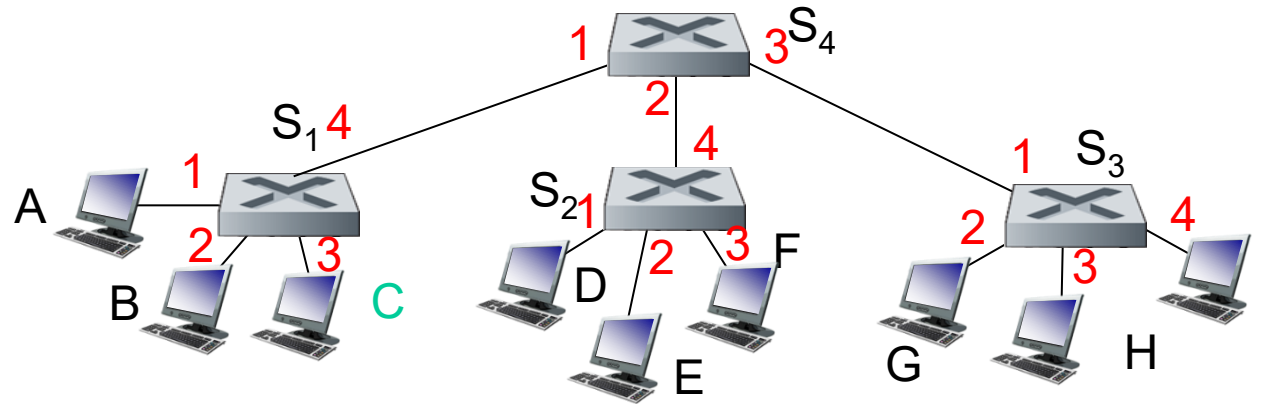


Q: Enviando de **A** para **G** – como **S₁** sabe que tem que repassar quadro destinado a **G** via **S₄** e **S₃**?

- ❖ **R:** Autoaprendizagem! (funciona *exatamente* do mesmo modo do caso com único *switch*!)

Exemplo de auto-aprendizagem no caso multi-switch

Suponha que **C** envia quadro para **I**, **I** responde para **C**



❖ Q: Mostre as tabelas de comutação em S₁, S₂, S₃, S₄

Switches vs. roteadores

Ambos armazenam-e-repassam:

- **Roteadores:** dispositivos de camada de rede (examinam cabeçalhos da camada de rede)
- **Switches:** dispositivos da camada de enlace (examinam cabeçalhos da camada de enlace)

Ambos tem tabelas de repasse:

- **Roteadores:** calculam tabelas usando algoritmos de roteamento, endereços IP
- **Switches:** aprendem tabela de repasse usando inundação, aprendizagem, endereços MAC

