

# Redes de Computadores

## Encaminhamento com Base em Difusão (*broadcasting*)

### Parte 1 — Canais Ethernet

Departamento de Informática da  
FCT/UNL

# Objetivos do Capítulo

- Do ponto de vista do endereçamento, isto é, do número de interfaces que ligam, os canais podem ser classificados em
  - Canais ponto a ponto - só ligam duas interfaces e só precisam de 2 endereços ou mesmo de nenhum
  - Canais multi-ponto - canais que necessitam de mais do que 2 endereços e que muitas vezes se baseiam na difusão do sinal (*broadcast*)
  - Necessitam de um mecanismo de endereçamento
  - Um canal baseado em difusão constitui por si uma rede de interligação de  $N$  nós, com  $N \geq 2$
- Como os canais baseados em difusão suportam mais do que um emissor, potencialmente simultâneos, é necessário estabelecer uma ordem pela qual estes podem emitir
- Neste capítulo veremos como funciona uma classe de canais deste tipo que resolve este problema com base em aleatoriedade

*Metcalfe's Law: "The value of a network grows as the square of the number of its users."*

*Autor: Robert Metcalfe, inventor of the Ethernet network*

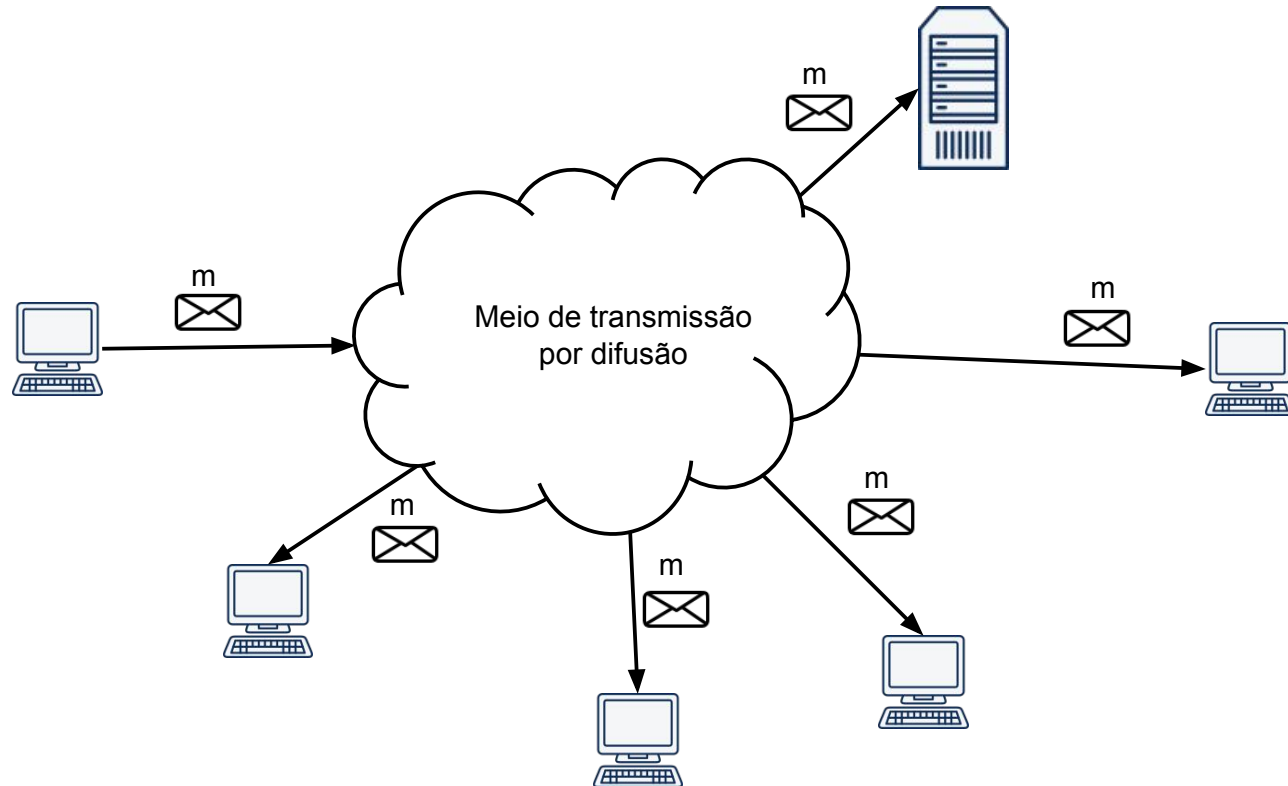
# Canais Ponto-a-Ponto

- Exemplos de canais ponto-a-ponto
  - Canais ponto-a-ponto em fibra ótica entre dois nós de comutação
  - Canais ponto-a-ponto entre um comutador Ethernet e um computador
- Os canais ponto-a-ponto são geralmente *full-duplex*, isto é, o canal pode transmitir simultaneamente nos dois sentidos
- Um canal ponto-a-ponto é equivalente a dois canais *simplex*, um em cada sentido
  - Os dois canais *simplex* são uni-direcionais, cada um num sentido, e independentes um do outro

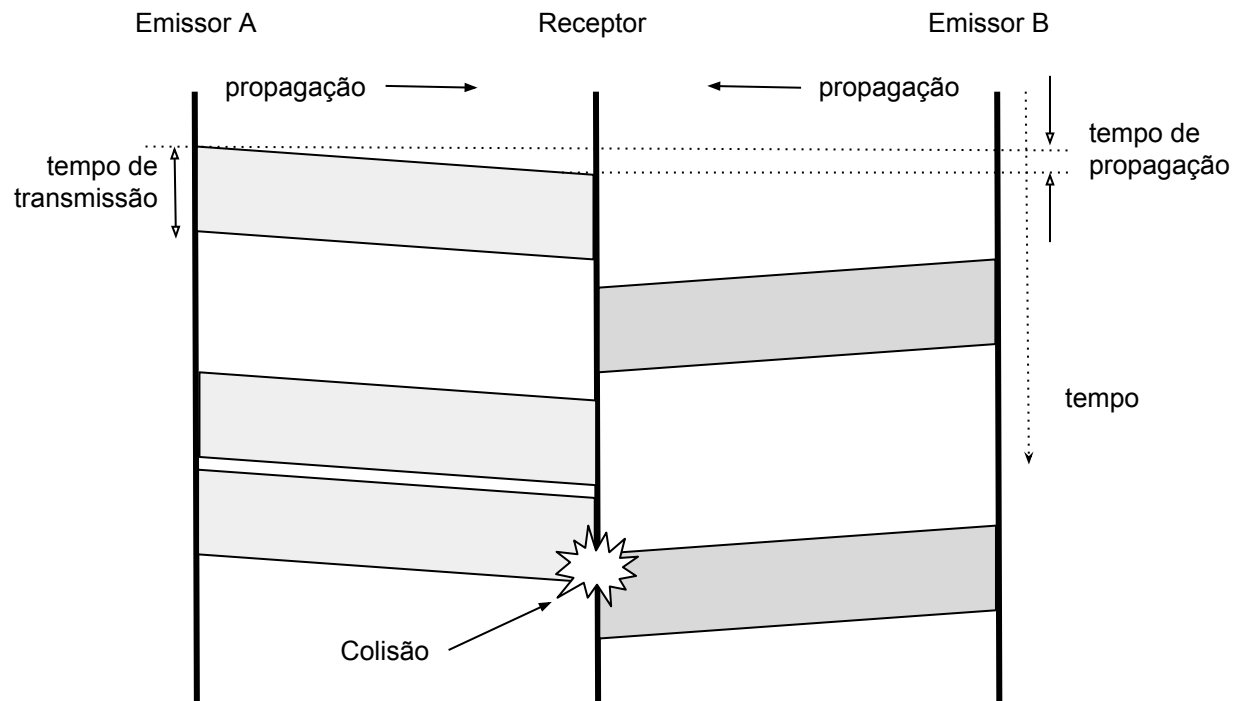
# Canais Multi-Ponto

- Os canais multi-ponto são geralmente baseados em difusão - *broadcasting (shared medium)*
- Exemplos: Ethernet tradicional (IEEE 802.3 — Ethernet com fios), WIFI (IEEE 802.11 — WiFi)
- Em cada momento só pode existir um único emissor
- Um canal baseado em difusão é *half-duplex* pois só há uma comunicação de cada vez:
  - um emissor para um recetor específico
  - um emissor para todos os outros
- Os canais baseados em difusão introduzem o problema do controlo de acesso ao meio (*Medium Access Control*)

# Canal Multi-Ponto - A Abstração



# Canais Multi-Ponto e Colisões



# Controlo de Acesso ao Meio

- **Controlo centralizado**

- uma interface pode começar a transmitir quando um árbitro centralizado lhe afeta esse direito (como a mesa de uma assembleia).
- Característico do antigo sistemas de telefones móveis

- **Controlo distribuído**

- uma interface pode começar a transmitir quando recebe o direito de o fazer através de um testemunho (“token”); a gestão do testemunho é feita por um algoritmo distribuído (exemplo: rede Token Ring)

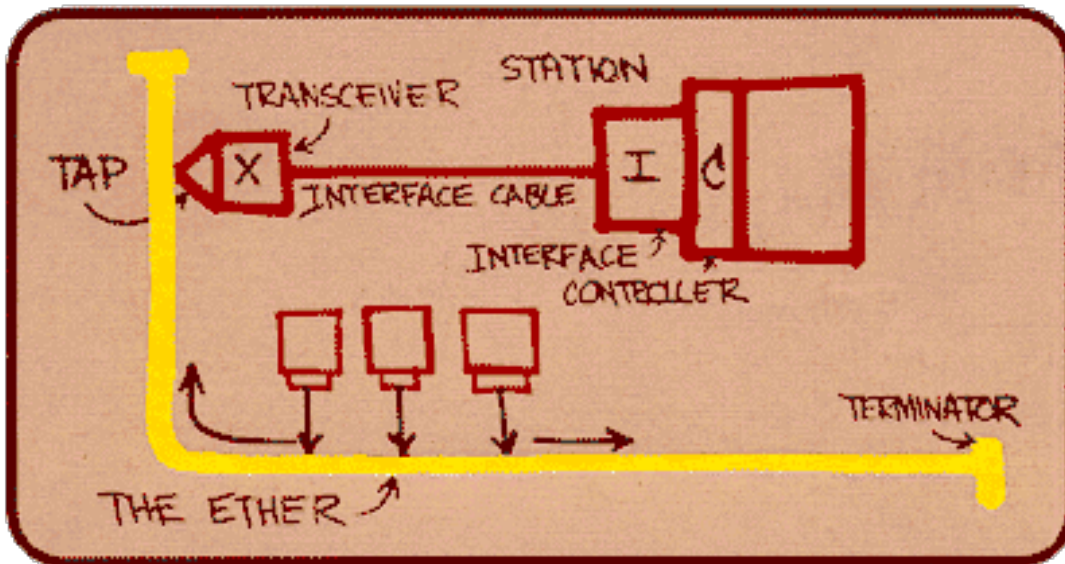
- **Controlo aleatório**

- uma interface pode começar a transmitir logo que o canal esteja livre, ou pelo menos desde que lhe pareça que o canal está livre (exemplo: rede Ethernet)

# Ethernet com Fios (e.g. IEEE 802.3)

É a tecnologia dominante nos canais "guiados" (com fios) atuais

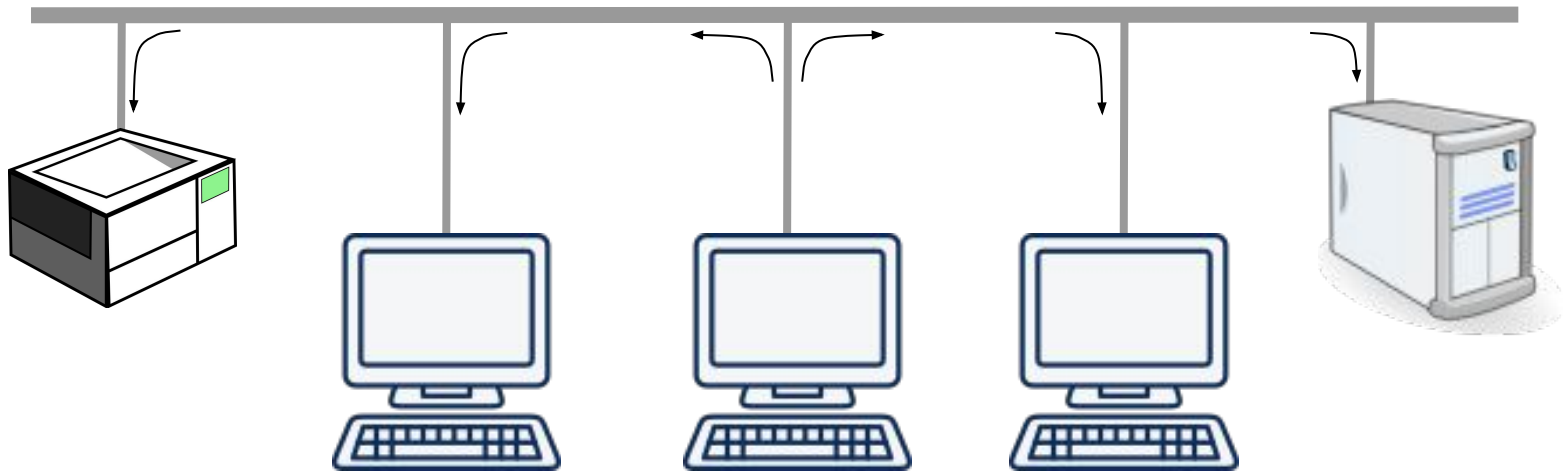
- Tem imensas variantes que partilham entre si o formato dos *frames* e o endereçamento
- Não pára de evoluir: 10, 100, 1000 Mbps (1 Gbps), 10 Gbps, ...
- Os *Switches* Ethernet transformam-na numa rede completa e mais abrangente que funciona hoje em dia predominantemente ponto-a-ponto e não baseada num cabo único



O diagrama inicial do inventor da Ethernet:

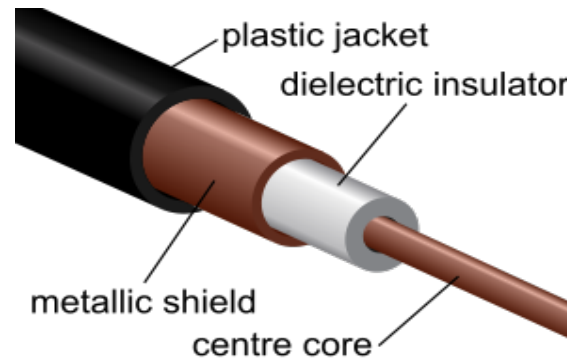
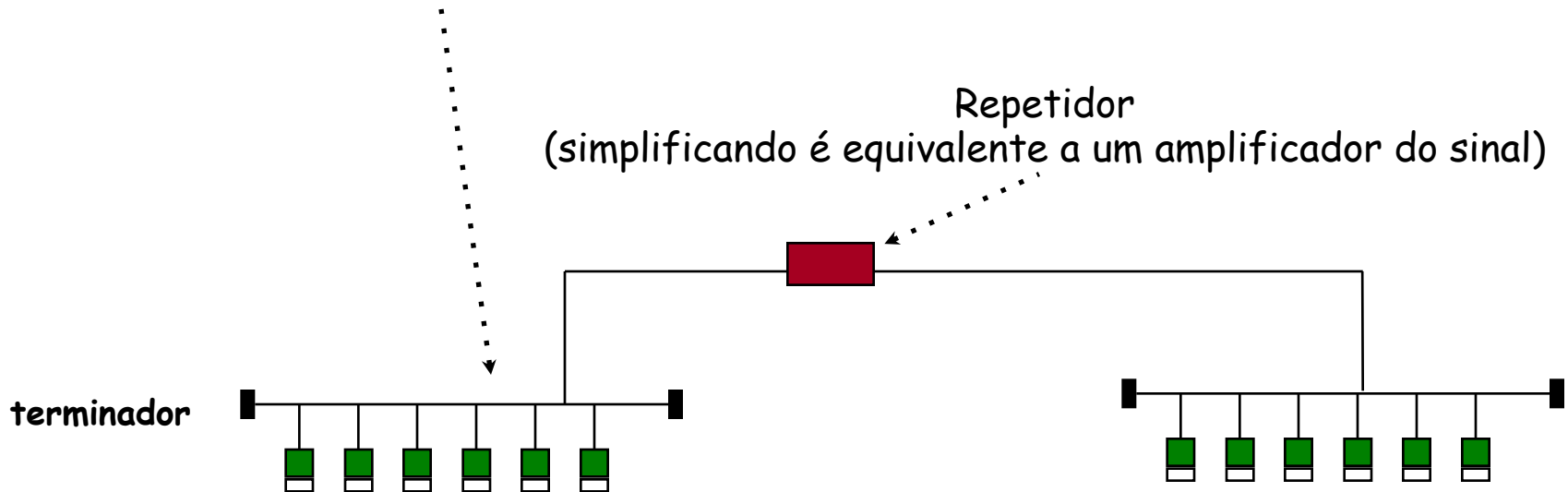
Metcalfe's Ethernet

# Canal de Difusão em *Bus* (*old Ethernet*)

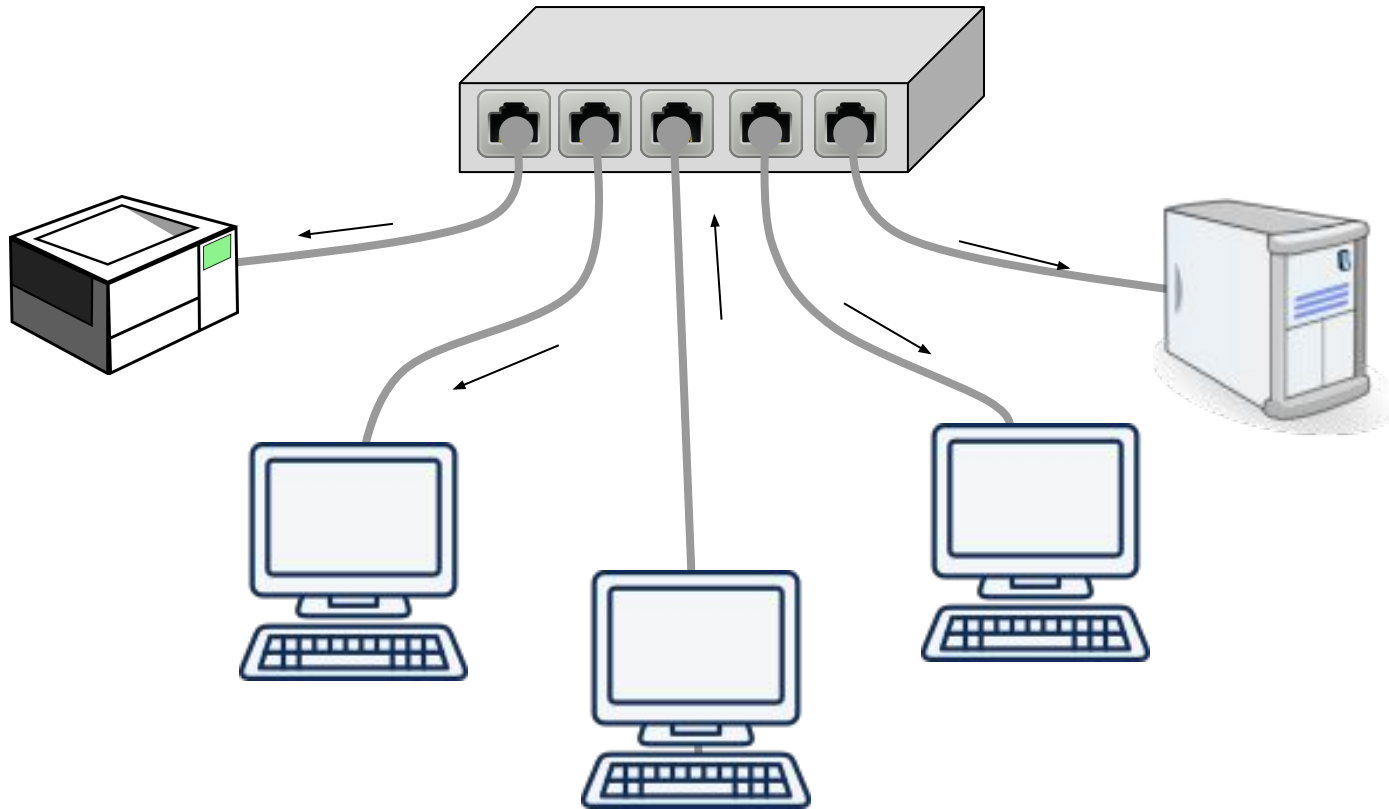


# Ethernet sobre Cabo Coaxial

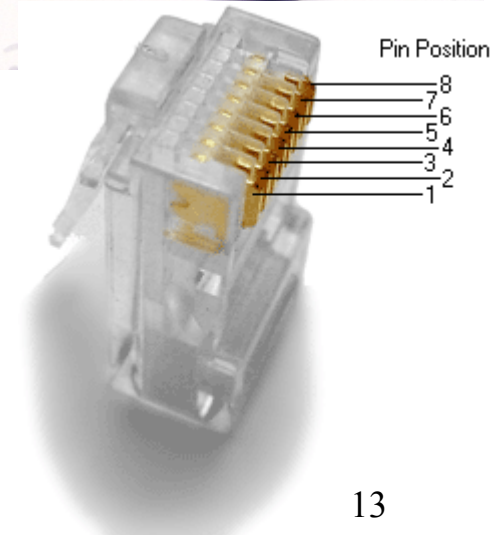
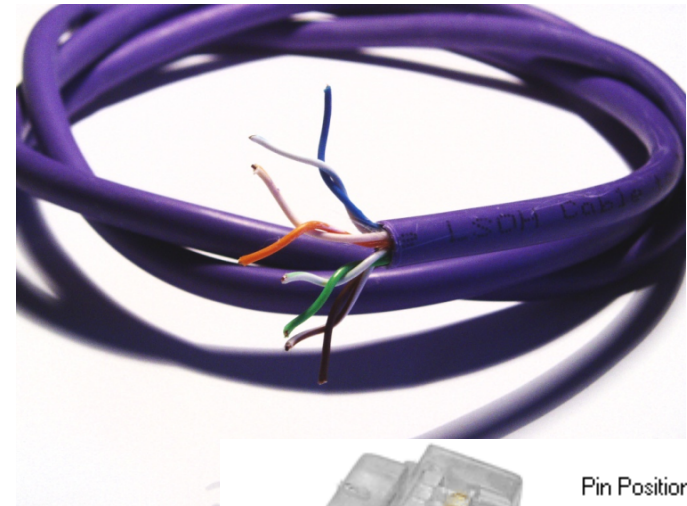
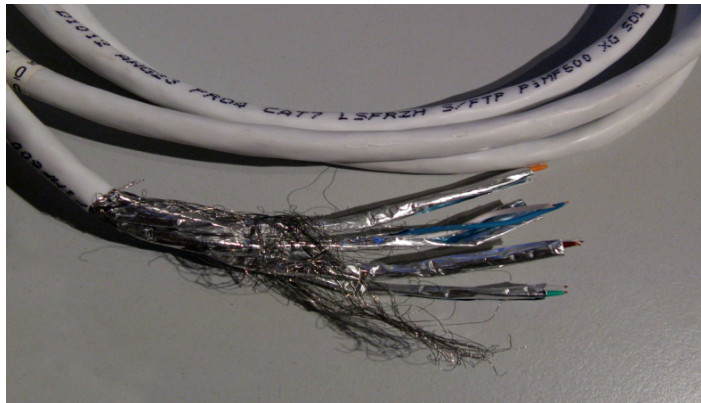
"bus" baseado em cabo coaxial



# Canal de Difusão Ethernet em *Estrela*

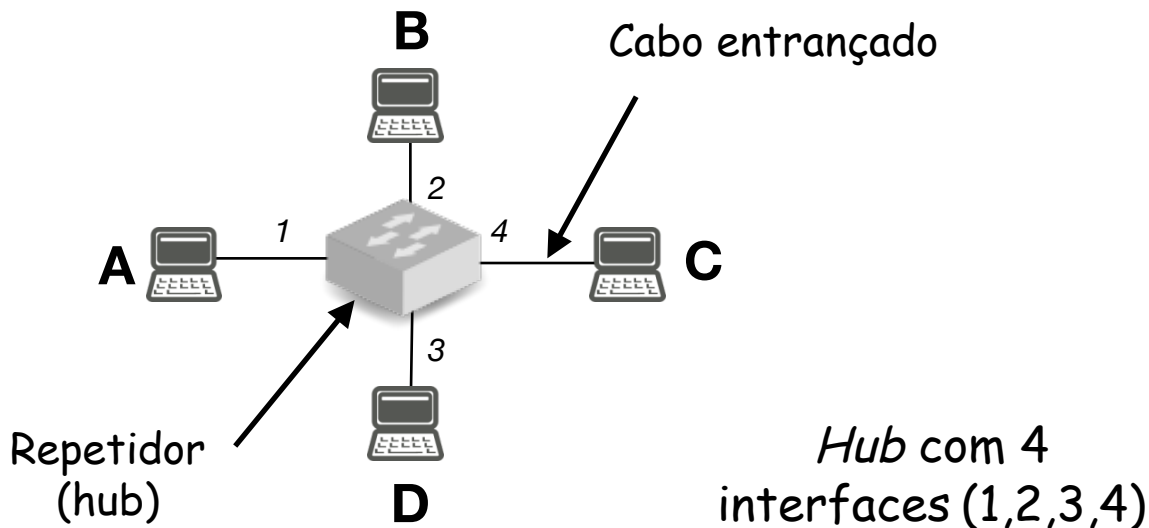


# Ethernet sobre cabo entrançado



# Hubs — Repetidores de Nível Físico

- Os sinais que chegam por uma porta são retransmitidos para todas as outras portas
- Os repetidores, ou *hubs*, não "interpretam" pacotes, só sinais eléctricos



# Ideias chave do controlo aleatório

- *Carrier sense*

- Verificar o estado do meio de transmissão
- Transmitir se o mesmo está livre

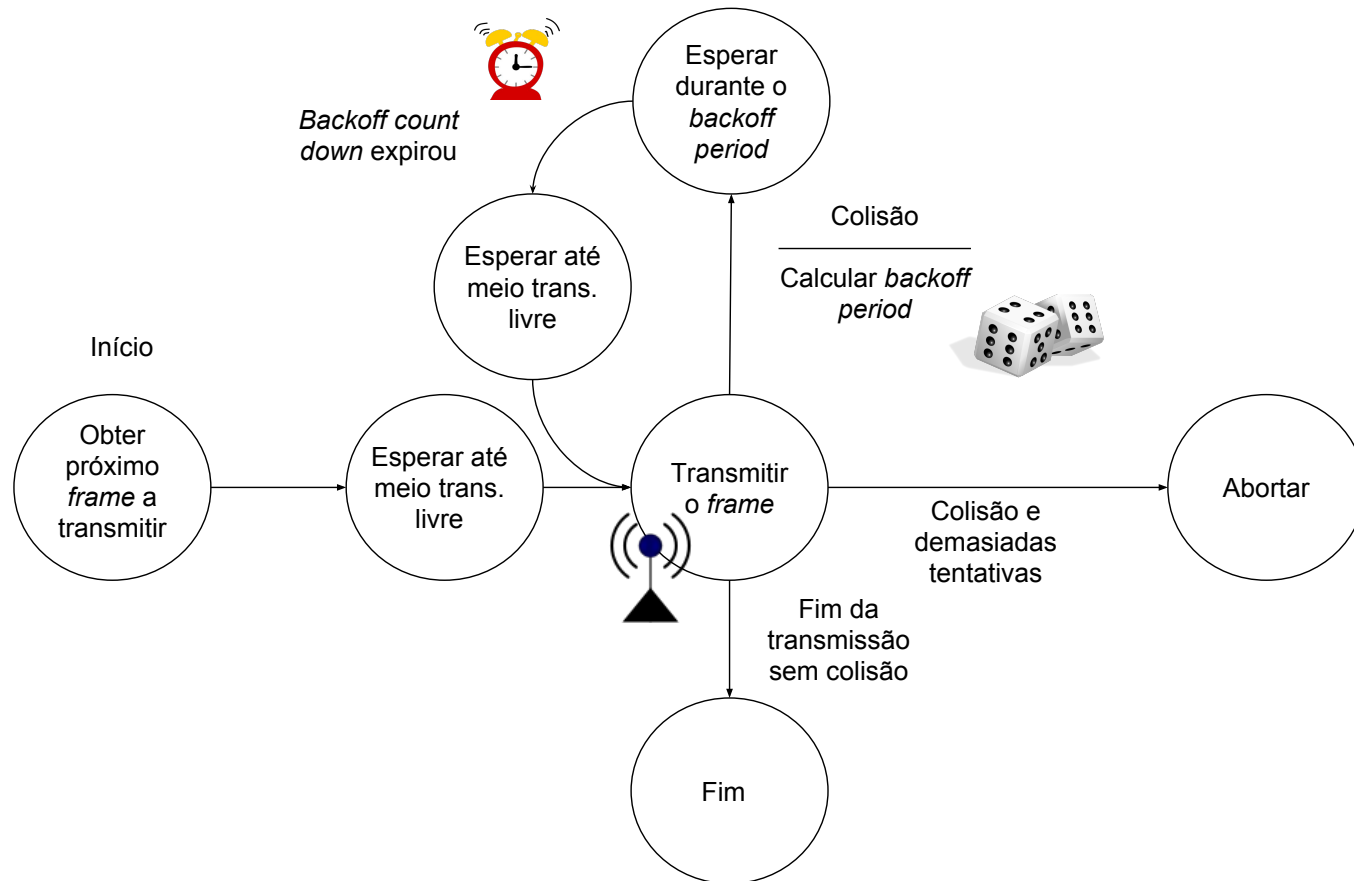
- *Deteção das colisões*

- Se alguém transmitir ao mesmo tempo, stop
- Detecta-se a colisão verificando a qualidade do sinal que circula; em caso de colisão o sinal está distorcido

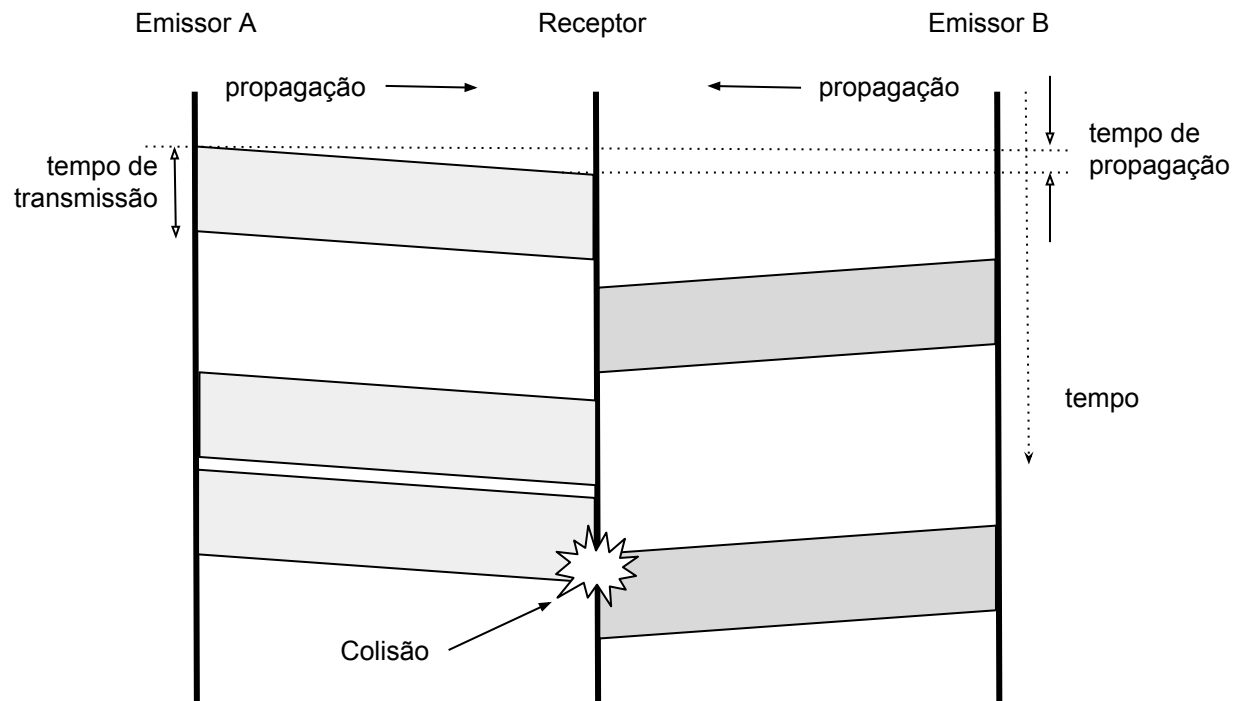
- *Aleatoriedade*

- Não recommençar a transmitir imediatamente
- Esperar um compasso de espera aleatório antes de tentar de novo para permitir desempatar

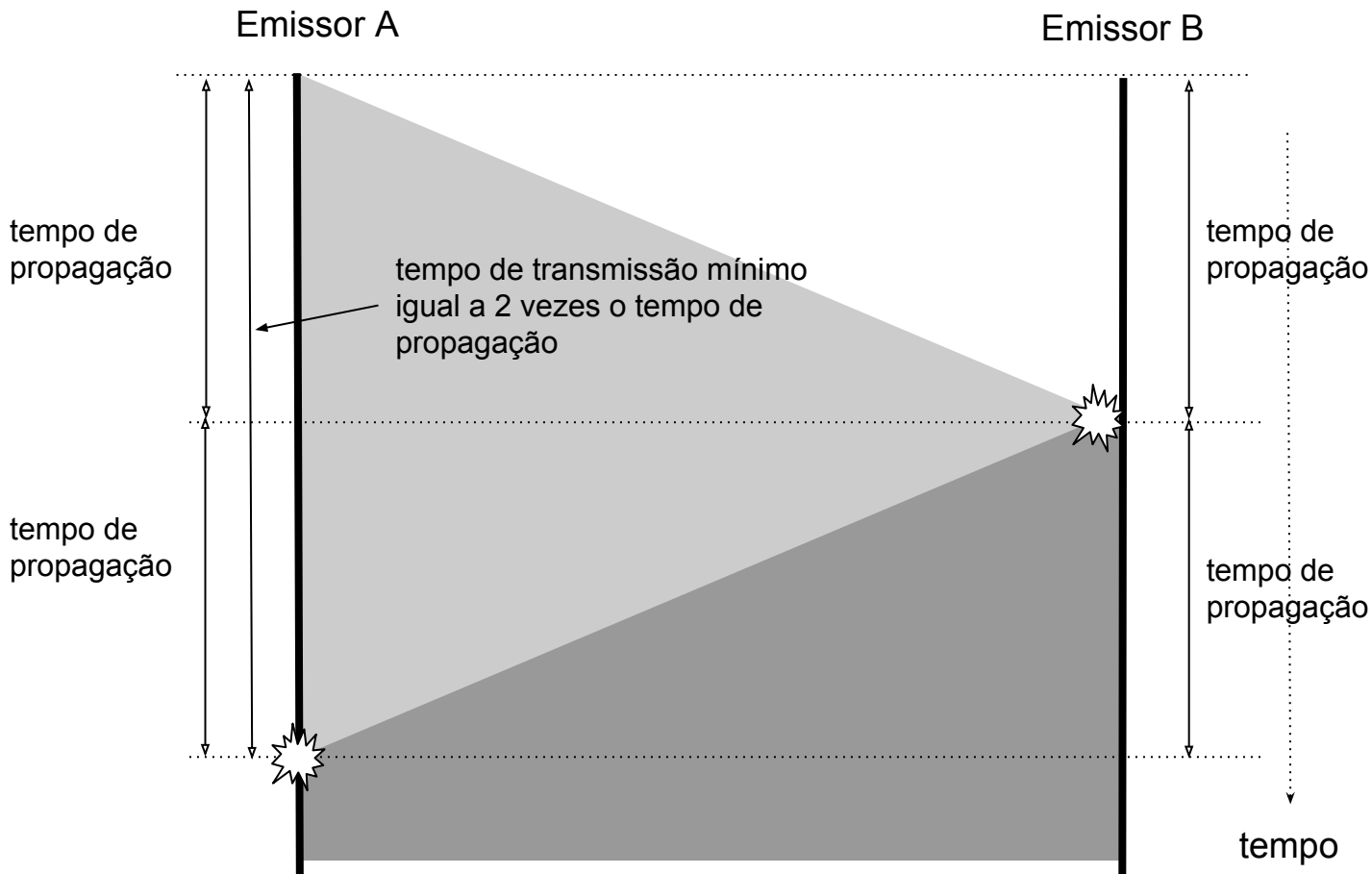
# CSMA/CD (Carrier Sense Multiple Access / Collision Detection)



# Com tempo de propagação nulo, as colisões seriam mais raras



# Colisões e Tempo de Propagação

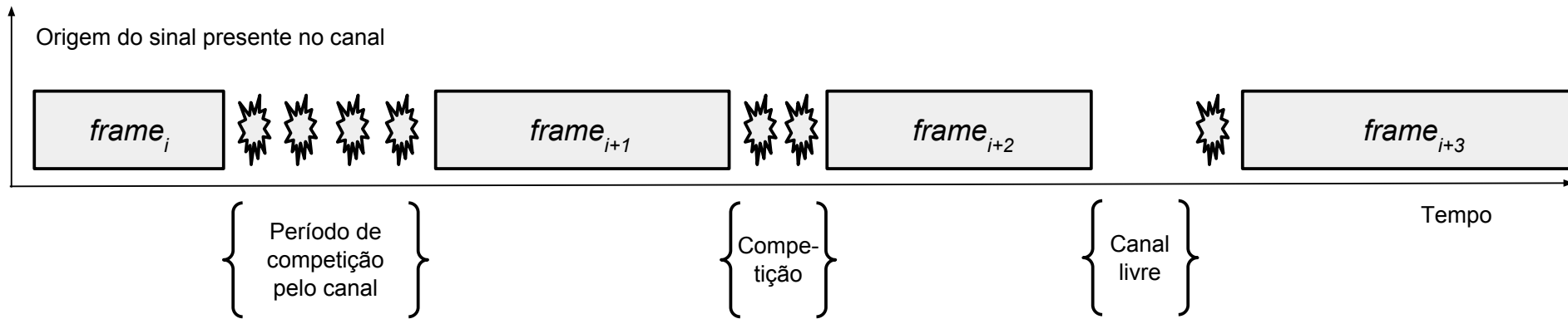


A dimensão do menor *frame* é condicionada pelo tempo de propagação máximo ou *worst case path delay*

# Algoritmo CSMA/CD na Ethernet

```
CST = 0.000051 = 51  $\mu$ s ; // 10 Mbps
k = 1; // transmission attempts
frame = getNextFrameToTransmit();
while ( k < 16 ) {
    wait_for_free_media (); // carrier sense phase
    transmit ( frame );
    if ( success ) break;
    // else collision
    transmit ( jam_signal );
    cw = min( 2k-1, 1023); // collision window <= 1023
    n = random ( 0 .. cw );
    k++;
    wait ( n*CST );
}
if ( k < 16 ) report_success else abort_transmission ;
```

# Funcionamento



# *Binary Exponential Backoff*

Exponential Backoff = Recuo Exponencial

O tempo é dividido em slots de dimensão superior a  $2T_p$  ditos *collision windows* ou *collision slots*.

Segundo a norma IEEE 802.3 o *collision slot* vale 51,2 micro segundos ou 512 bits a 10 Mbps. Este valor é característico de uma rede a 10 Mbps com 2.500 metros e 4 repetidores no máximo. Tal rede poderia levar até 1024 interfaces segundo a mesma norma.

Cada *frame* tem de ter um mínimo de 512 bits ou 64 bytes para ser emitido durante pelo menos *collision slot*, daí o campo de *pading* para o caso em que os dados são inferiores a essa dimensão

# *Binary Exponential Backoff (2)*

Em caso de colisão, cada interface espera um número aleatório  $N$  de *collision slots*. O número  $N$  é gerado no intervalo 0 a  $2^N - 1$ , em que  $N$  toma valores 0, 1, 2, 3, ..., 10 e designa o número de tentativas de resolução da colisão.

Na primeira tentativa espera-se 0 ou 1 *slots*. Na segunda, 0, 1, 2 ou 3 *slots*. Na décima, esperam-se 0, 1, 2, 3, ... 1023 *slots*. Depois de 16 tentativas considera-se que há erro e o *frame* é rejeitado.

# Observações

- *Jam signal* — assegura que todos os emissores detectam a colisão; tem 48 bits
- O algoritmo dito com *exponential backoff* tem por objetivo adaptar o compasso de espera à carga da rede; se a carga é elevada, o intervalo tem de ser superior;
  - primeira colisão: escolher  $K$  em  $\{0,1\}$ ; o delay é  $K \times 512$  bit *transmission times*
  - segunda colisão: escolher  $K$  em  $\{0,1,2,3\}$ .....
  - com 10 colisões: escolher  $K$  em  $\{0,1,2,3,4,...,1023\}$ .....
  - e assim sucessivamente

# Discussão

Caso o intervalo usado para gerar números aleatórios fosse 0 a 1023 em todas as circunstâncias, a probabilidade de 2 estações colidirem uma segunda vez seria geralmente baixa mas introduzir-se-ia uma espera inutilmente prolongada no caso geral.

Por outro lado, se 100 estações a tentarem transmitir gerassem números aleatórios no intervalo 0 a 1, para que a colisão fosse resolvida era necessário que saísse 1 em 99 delas e 0 na outra.

O algoritmo tal como está definido tenta adaptar-se à situação real de carga da rede, isto é, à quantidade de interfaces a tentarem de facto emitirem simultaneamente.

# Débito e Dimensão do *Collision Slot*

Mantendo a mesma dimensão máxima da rede, mantem-se a duração do *collision slot*. Se pretendermos introduzir um canal de maior velocidade, o menor *frame* possível seria de 640 bytes para uma rede a 100 Mbps e de 6400 bytes para uma rede de 1 Gbps.

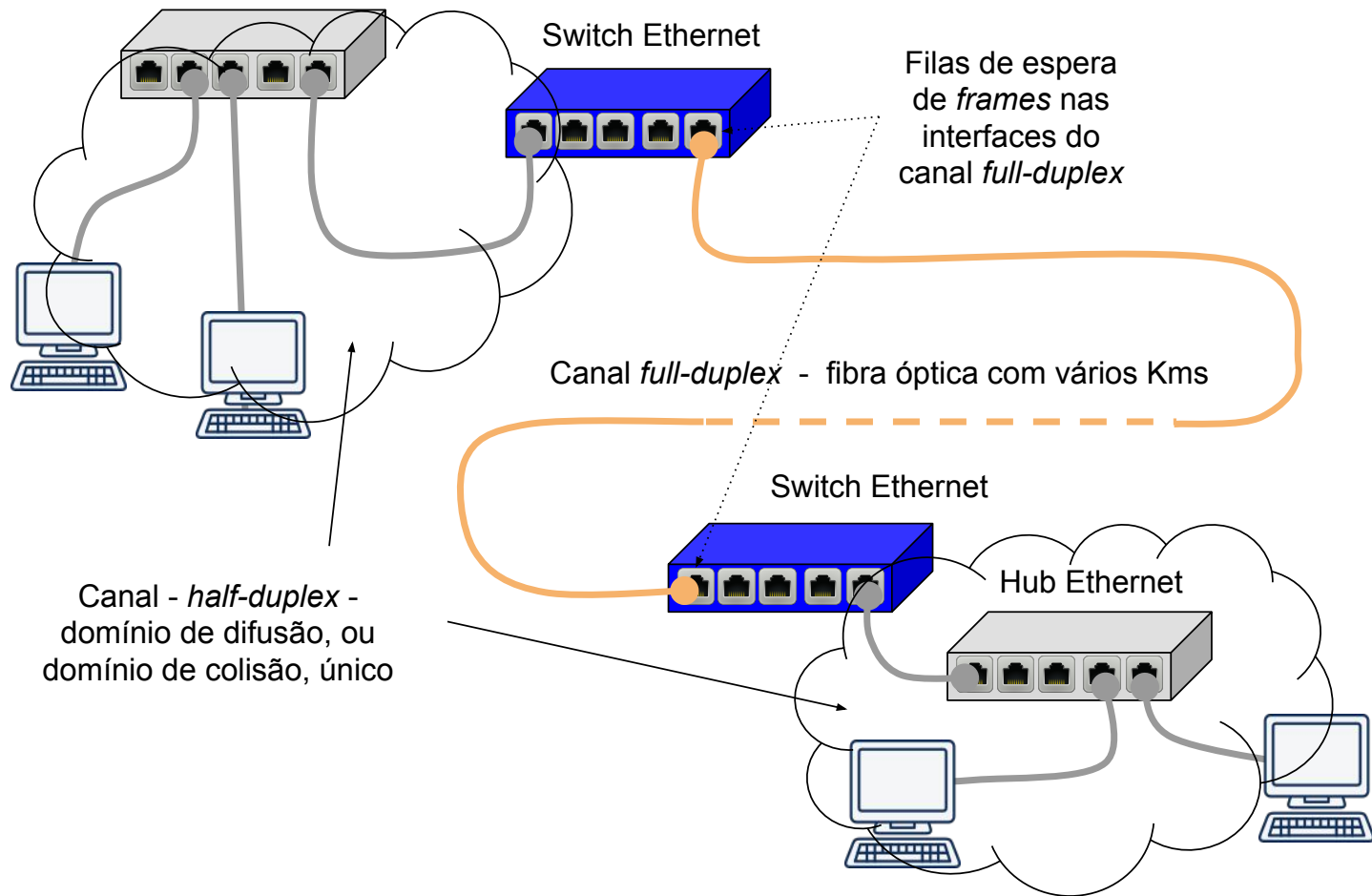
A outra opção é baixar a distância máxima de forma a manter o *collision slot* em termos de bits, mas diminuí-lo em termos de duração temporal

Foi a opção usada na norma *Fast Ethernet*, que corresponde a um canal ethernet que funciona a 100 Mbps, com a distância máxima de 200 metros, para que o mais pequeno *frame* continue a ser de 64 bytes.

# Parâmetros

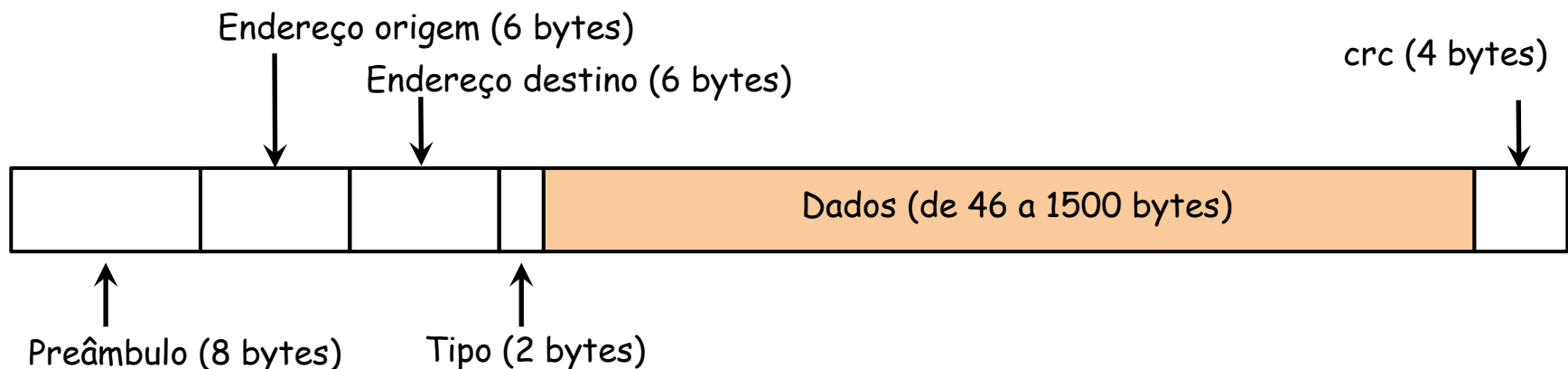
| Parâmetro            | 10/100 Mbps   | 1 Gbps         | 10/100 Gbps   |
|----------------------|---------------|----------------|---------------|
| Backoff slot time    | 512 bit times | 4096 bit times | Não se aplica |
| Inter Frame Space    | 96 bits       | 96 bits        | 96 bits       |
| # tentativas máximas | 16            | 16             | Não se aplica |
| Jam size             | 32 bits       | 32 bits        | Não se aplica |
| Minimum Frame Size   | 64 bytes      | 640 bytes      | 64 bytes      |
| Masimum Frame Size   | 1518 bytes    | 1518 bytes     | 1518 bytes    |

# Canais Ethernet Half e Full-Duplex



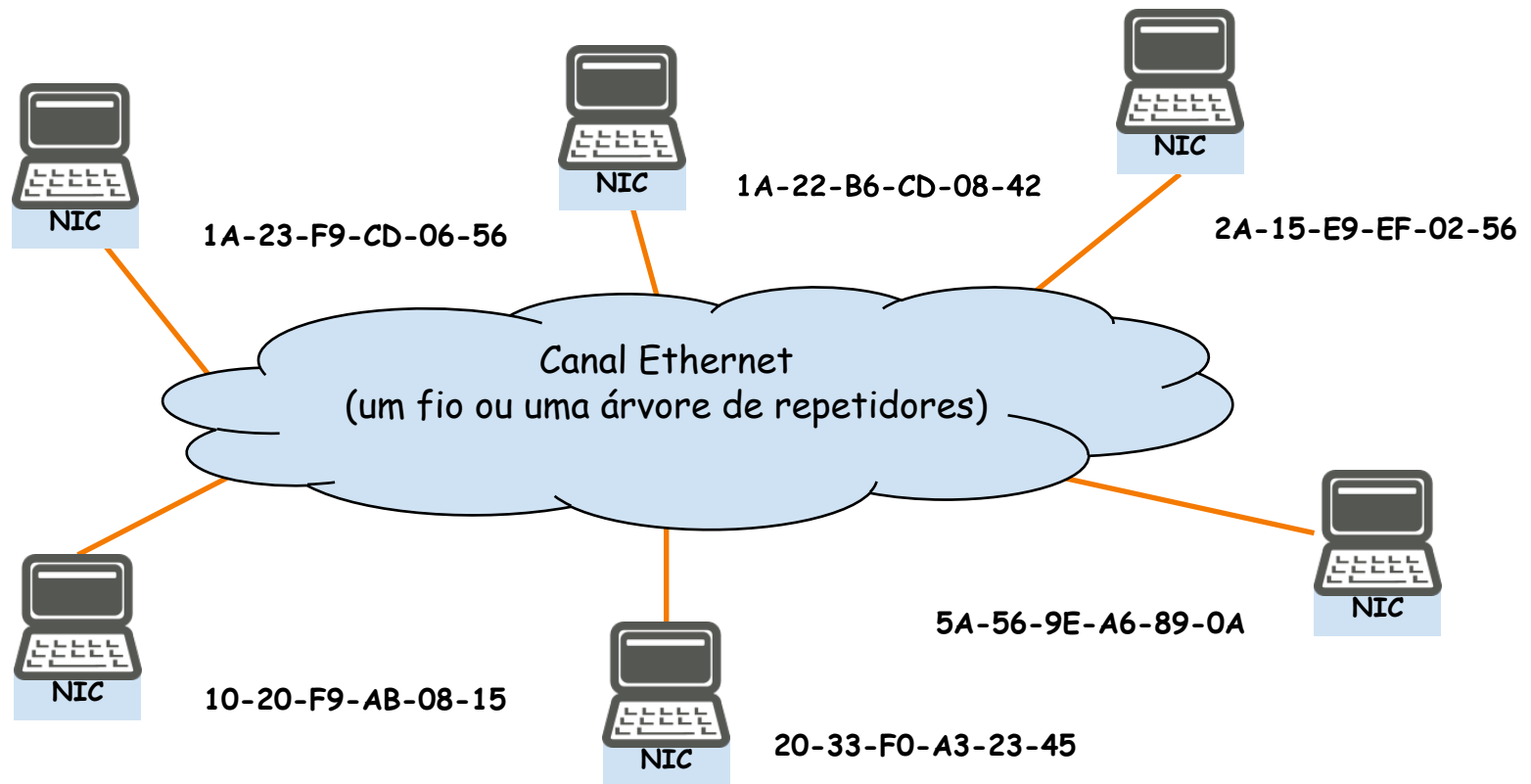
# O *frame* Ethernet

- Preâmbulo: 7 bytes com o padrão 10101010 seguidos de um byte com o padrão 10101011. Este preâmbulo serve para sincronizar os relógios do emissor e do receptor
- Endereços: 6 bytes cada (48 bits); os *frames* são recebidos por todas as interfaces numa mesma rede mas são ignorados se o endereço não é o endereço da interface receptora (ou um endereço de *multicasting* / *broadcasting* )
- CRC: testado pelo receptor; se está errado o *frame* é ignorado

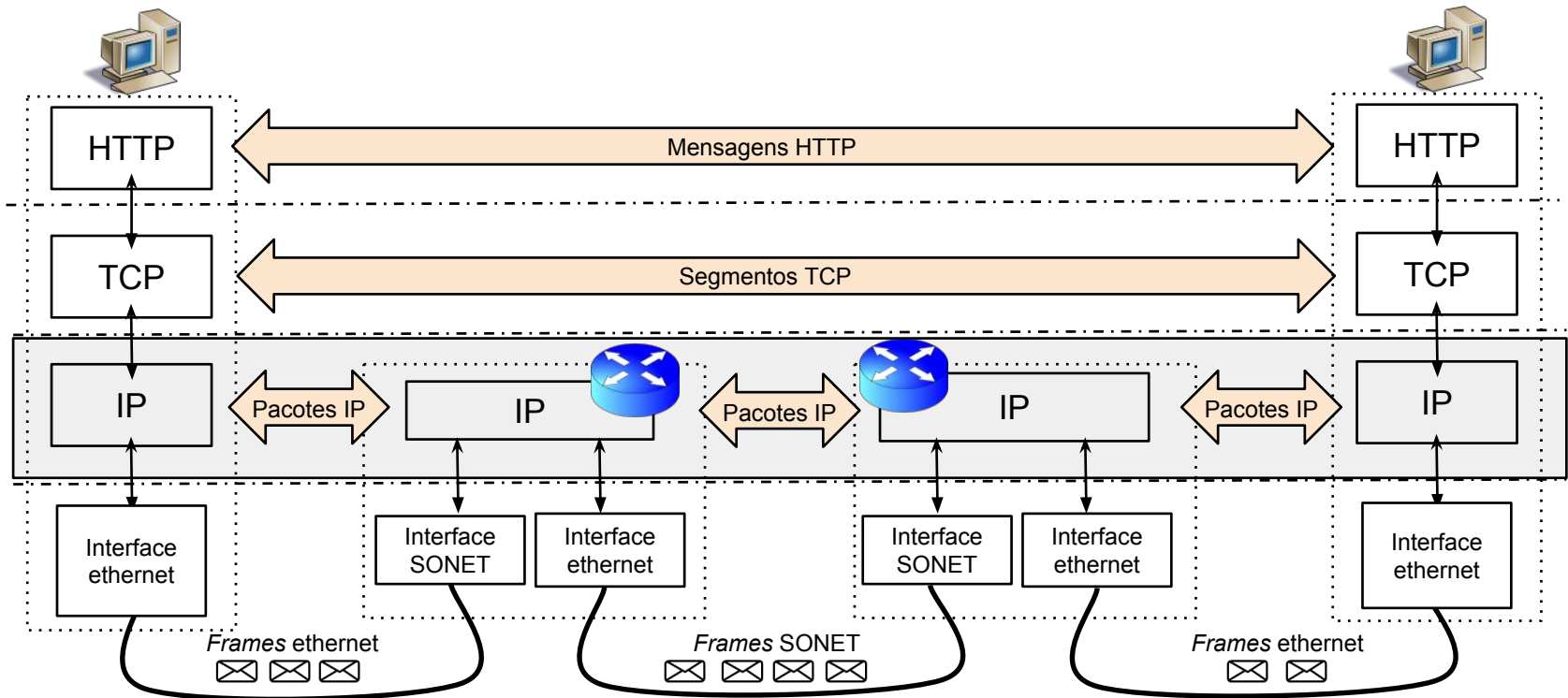


# MAC Addresses

- Cada placa tem um endereço "dito de canal", ou do nível MAC (Medium Access Control Layer), único com 48 bits, colocado na fábrica

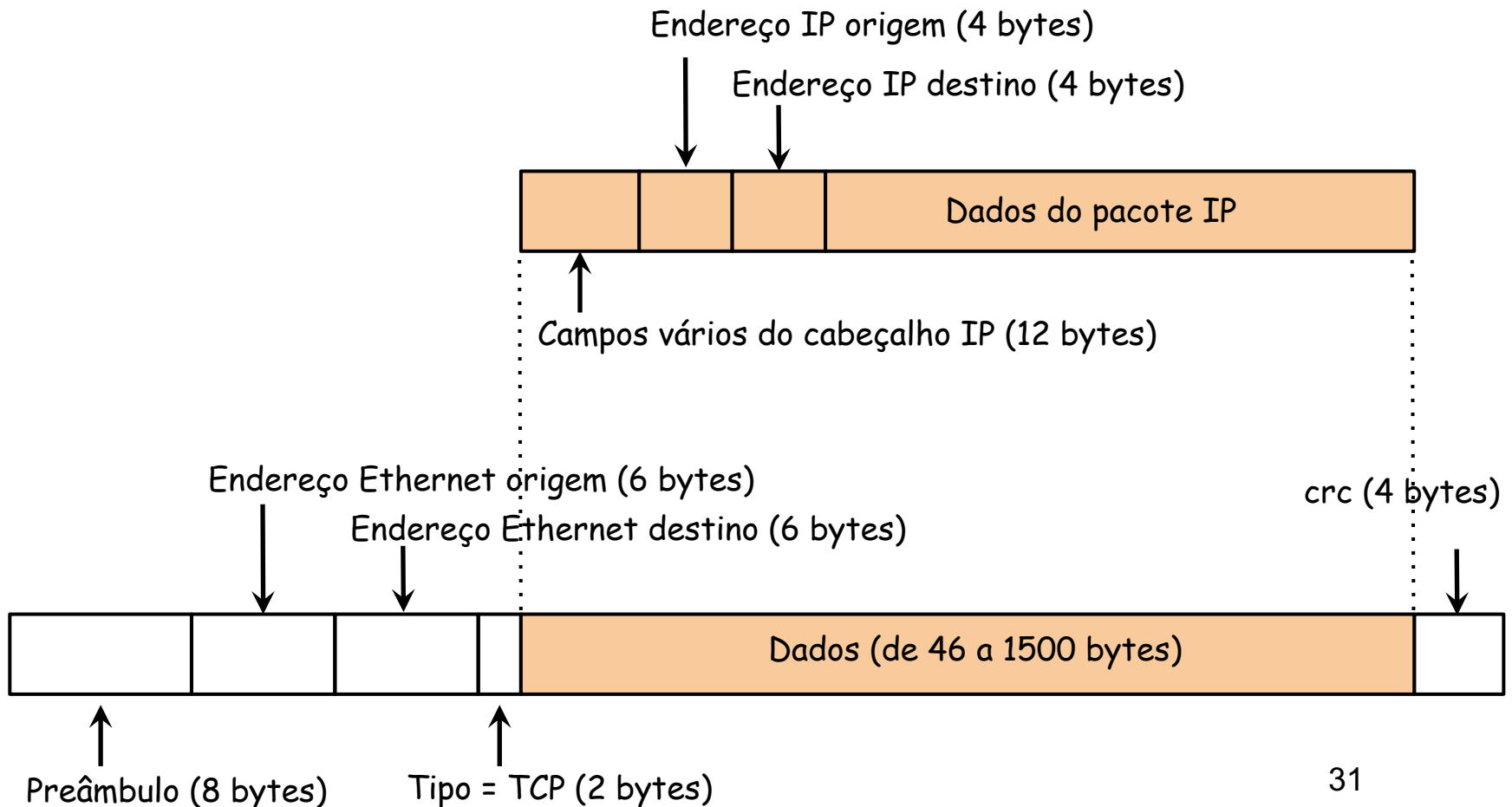


# Os Diferentes Níveis



# Encapsulamento

- Os *frames* contêm na parte de dados mensagens dos níveis superiores (encapsuladas). Frequentemente os frames Ethernet transportam pacotes IP



# Conclusões

- Um canal baseado em difusão num meio comum (*broadcast*) permite a mais do que duas interfaces comunicarem diretamente sem intermediários
  - Para esse efeito necessitam de um sistema de endereçamento ao nível canal (*MAC Layer*)
- Como suportam vários emissores, potencialmente simultâneos, é necessário estabelecer uma ordem pela qual eles podem emitir
  - Os métodos baseados em ordenação aleatória têm-se revelado como muito interessantes e são bastante usados
- Se o canal é baseado em fios, a qualidade do sinal e o baixo nível de erros permite a deteção de colisões. Nesse caso usa-se o método CSMA/CD