

Camada de Rede

[Sumário]

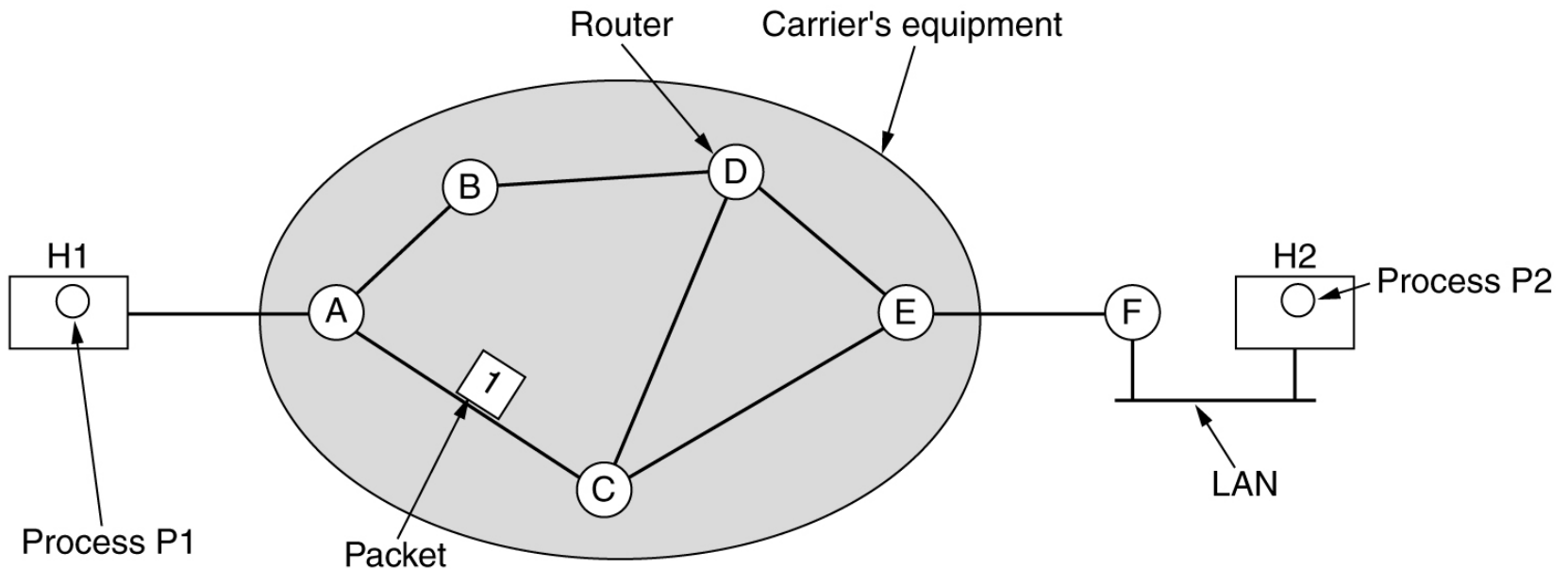
- Questões de projeto
- Algoritmos de roteamento
- Algoritmos de controle de congestionamento
- Qualidade de serviço
- Interligação de redes
- Camada de rede na Internet

[Questões de Projeto]

- No desenvolvimento de protocolos para a camada de rede, o projetista deve lidar com
 - Comutação de pacotes store-and-forward
 - Serviços oferecidos à camada de transporte
 - Serviços sem conexão ou orientados à conexão

Comutação de Pacotes store-and-forward

O ambiente de protocolos da camada de rede

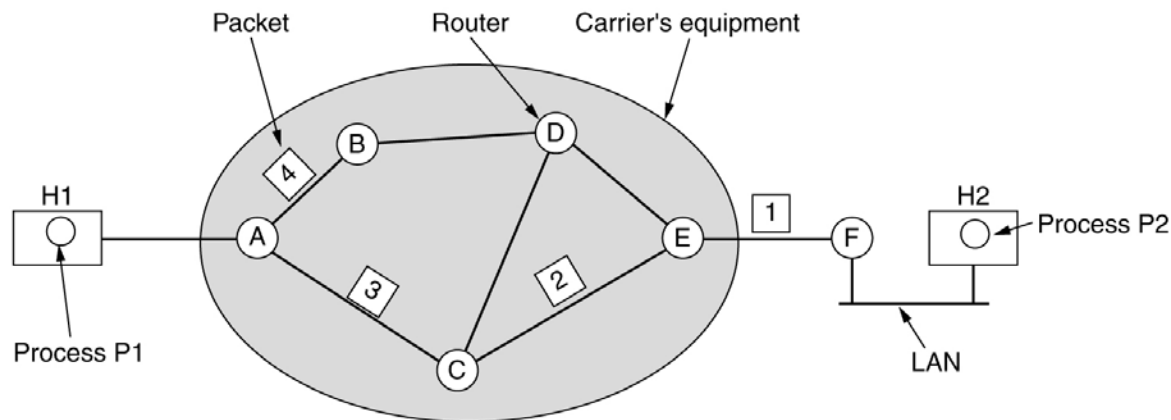


Serviços oferecidos à camada de transporte

- Os serviços são independentes da tecnologia de roteadores
- A camada de transporte deve ser isolada da topologia dos roteadores
- Endereçamento uniforme independente do tamanho das redes
- Serviços sem conexão e orientados à conexão

Serviços sem Conexão (Datagramas)

Roteamento em uma sub-rede de datagramas



A's table

	initially	later
A	-	-
B	B	B
C	C	C
D	B	B
E	C	B
F	C	B

Dest. Line

C's table

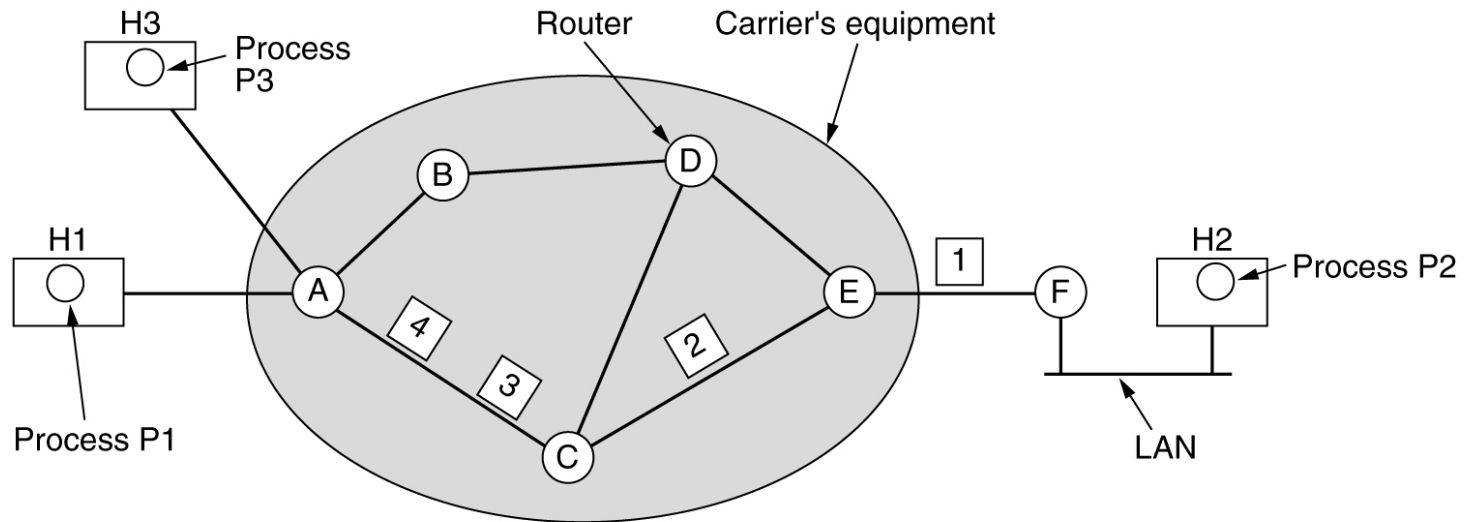
A	A
B	A
C	-
D	D
E	E
F	E

E's table

A	C
B	D
C	C
D	D
E	-
F	F

Serviços Orientados à Conexão

Roteamento em uma sub-rede de circuitos virtuais



A's table		C's table		E's table	
H1	1	A	1	C	1
H3	1	A	2	C	2
In		Out		Out	

Comparação entre redes de datagramas e circuitos virtuais

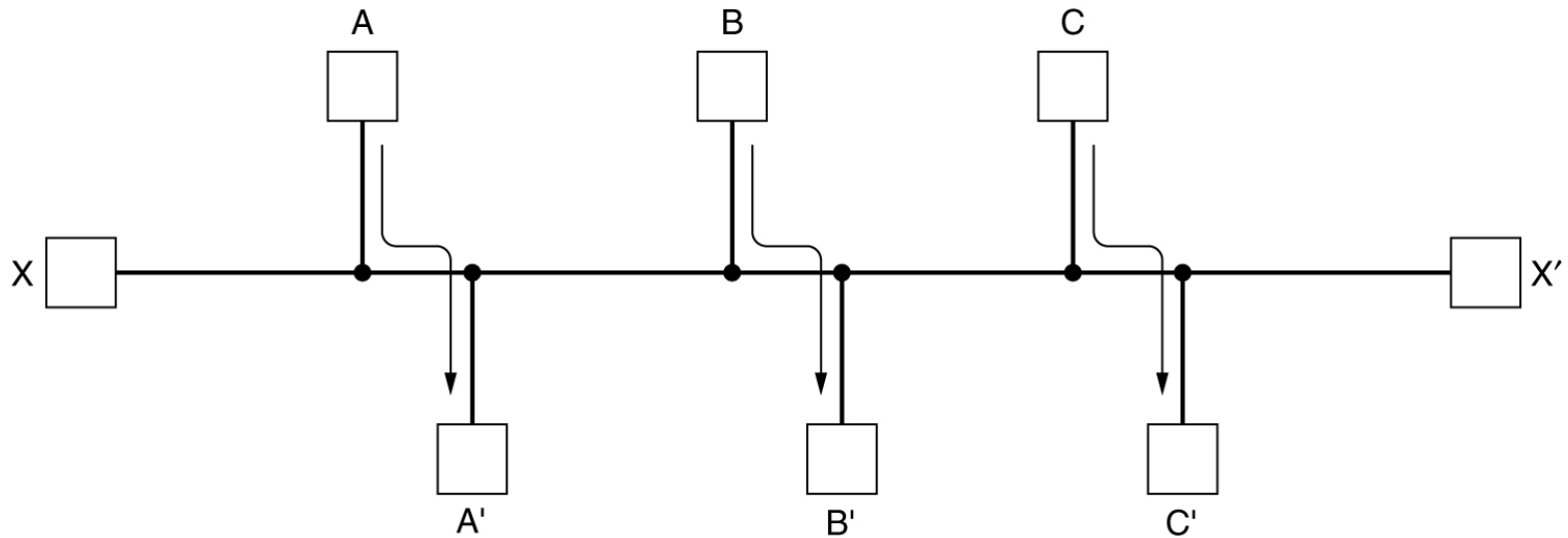
Issue	Datagram subnet	Virtual-circuit subnet
Circuit setup	Not needed	Required
Addressing	Each packet contains the full source and destination address	Each packet contains a short VC number
State information	Routers do not hold state information about connections	Each VC requires router table space per connection
Routing	Each packet is routed independently	Route chosen when VC is set up; all packets follow it
Effect of router failures	None, except for packets lost during the crash	All VCs that passed through the failed router are terminated
Quality of service	Difficult	Easy if enough resources can be allocated in advance for each VC
Congestion control	Difficult	Easy if enough resources can be allocated in advance for each VC

[Algoritmos de Roteamento]

- Software responsável pela escolha da linha de saída
 - Criação de tabelas de encaminhamento ou repasse
- Devem obedecer algumas propriedades
 - Simplicidade
 - Robustez
 - Estabilidade
 - Justiça
 - Otimização

[Algoritmos de Roteamento]

Conflito entre justiça e otimização



[Algoritmos de Roteamento]

- Os algoritmos de roteamento podem ser
 - Estáticos
 - Escolha das rotas é feita offline e enviada para os roteadores
 - Dinâmicos (adaptativos)
 - Escolha das rotas é feita com base na topologia e no tráfego da rede
 - Assimilam mudanças que possam eventualmente ocorrer

[Algoritmos de Roteamento]

■ Princípio da otimização

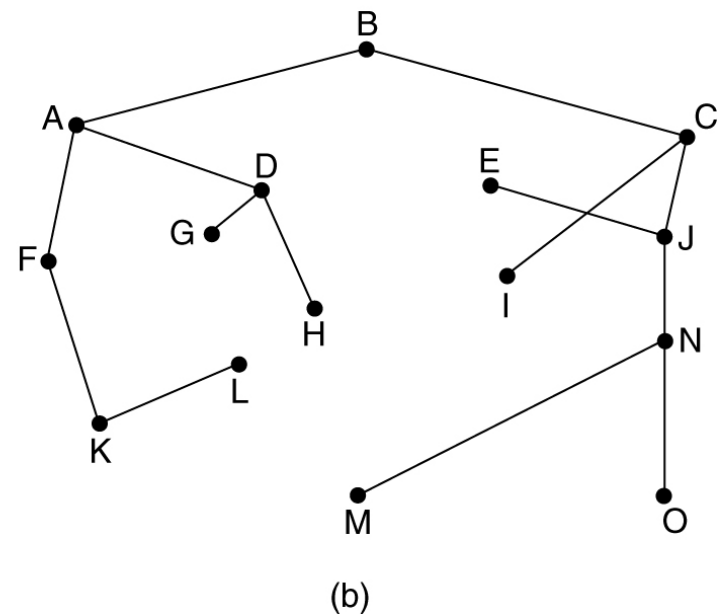
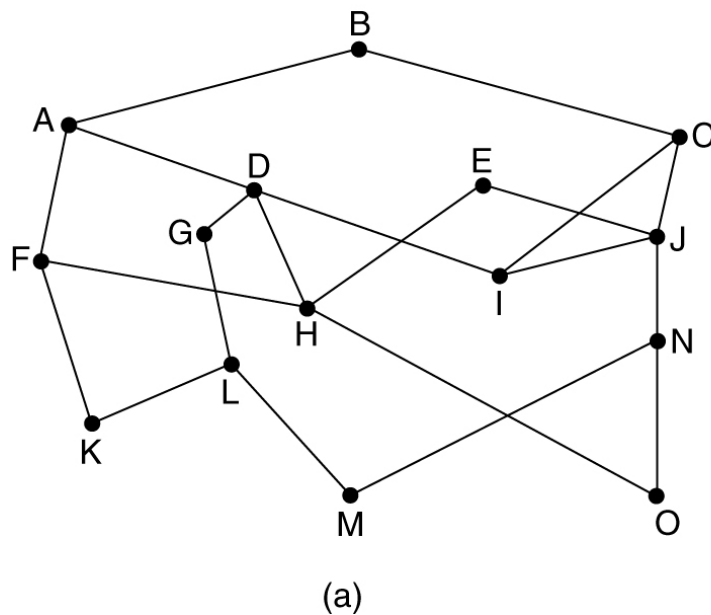
- Se o roteador J estiver no caminho ótimo entre I e K, então o caminho ótimo entre J e K estará na mesma rota

■ Consequência

- Conjunto de rotas ótimas de todas as origens para um determinado destino formam uma árvore com raiz no destino
- Árvore de escoamento
- Árvore não contém loops

[Princípio da otimização]

(a) Uma sub-rede (b) Uma árvore de escoamento para o roteador B



[Algoritmo de Dijkstra]

- Algoritmo pelo caminho mais curto
 - Curto aqui se refere ao caminho de menor custo
 - O custo pode englobar variáveis como o número de saltos, a distância geográfica, o de menor retardo, etc.

[Algoritmo de Dijkstra]

- Topologia de rede e custo dos enlaces são conhecidos por todos os nós
- Computa caminhos de menor custo de um nó (fonte) para todos os outros nós
 - Fornece uma tabela de roteamento para aquele nó
- Convergência: após k iterações, conhece o caminho de menor custo para k destinos
- Notação:
 - $C(i,j)$: custo do enlace do nó i ao nó j . Custo é infinito se não houver ligação entre i e j
 - $D(v)$: valor atual do custo do caminho da fonte ao destino V
 - $P(v)$: nó predecessor ao longo do caminho da fonte ao nó v , isto é, antes do v
 - N' : conjunto de nós cujo caminho de menor custo é definitivamente conhecido

[Algoritmo de Dijkstra]

1 **Inicialização:**

2 $N' = \{u\}$

3 para todos os nós v

4 se v é adjacente a u

5 então $D(v) = c(u,v)$

6 senão $D(v) = \infty$

7

8 **Loop**

9 ache w não em N' tal que $D(w)$ é um mínimo

10 acrescente w a N'

11 atualize $D(v)$ para todo v adjacente a w e não em N' :

12 $D(v) = \min(D(v), D(w) + c(w,v))$

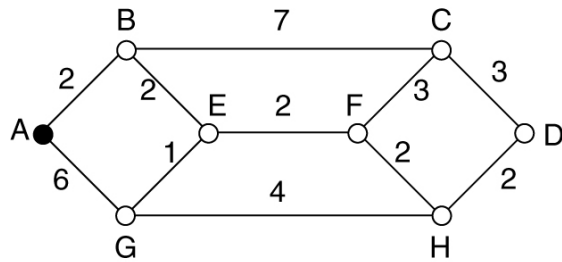
13 /* novo custo para v é ou o custo anterior para v ou o menor

14 custo de caminho conhecido para w mais o custo de w a v */

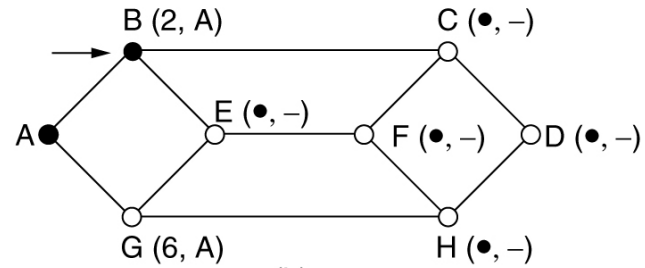
15 **até que todos os nós estejam em N'**



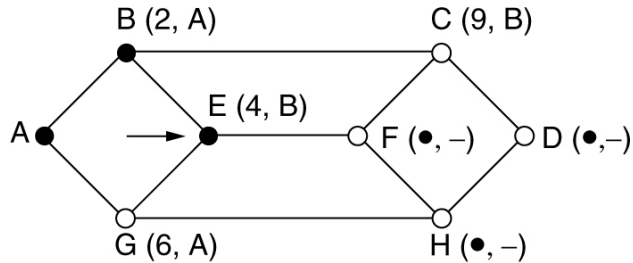
[Algoritmo de Dijkstra



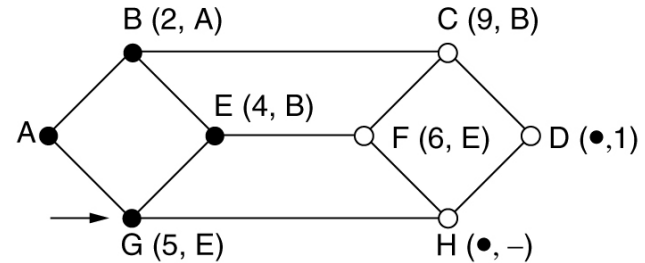
(a)



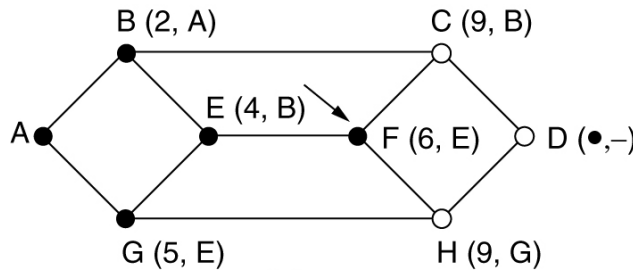
(b)



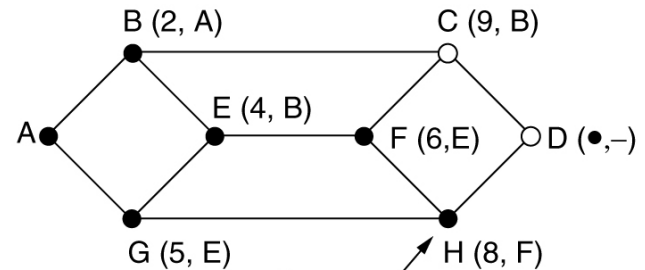
(c)



(d)



(e)



(f)

[Algoritmo de Inundação]

- Técnica usada em protocolos de roteamento por difusão (broadcast)
- O roteador envia uma cópia do pacote recebido para todas as suas linhas de saída com exceção daquela em que o pacote chegou
 - Problema com loops
 - Números de sequência podem ser necessários

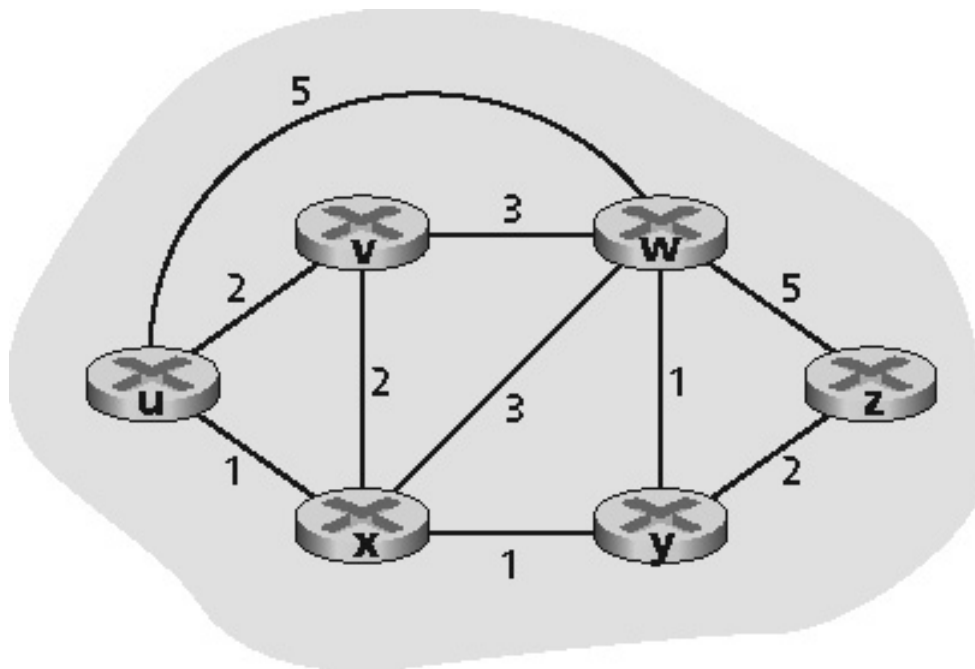
Roteamento com vetor de distância

- Um nó recebe informação de seus vizinhos, realiza cálculos e repassa o resultado para seus vizinhos
- Algoritmo distribuído e assíncrono
- Algoritmo usado originalmente na ARPANET e na Internet (RIP)
 - Foi substituído em razão da convergência lenta

Roteamento com vetor de distância

- Equação de Bellman-Ford (programação dinâmica)
- Define-se
 - $d_x(y)$ = custo do caminho de menor custo de x para y
- Então
 - $d_x(y) = \min \{c(x,v) + d_v(y)\}$
 - Em que min é calculado sobre todos os vizinhos de x

Roteamento com vetor de distância



Claramente, $d_v(z) = 5$, $d_x(z) = 3$,
 $d_w(z) = 3$

A equação B-F diz que:

$$\begin{aligned} d_u(z) &= \min \{ c(u,v) + d_v(z), \\ &\quad c(u,x) + d_x(z), \\ &\quad c(u,w) + d_w(z) \} \\ &= \min \{ 2 + 5, \\ &\quad 1 + 3, \\ &\quad 5 + 3 \} = 4 \end{aligned}$$

O nó que atinge o mínimo é o próximo salto no caminho mais curto

Roteamento com vetor de distância

- $D_x(y)$ = estimativa do menor custo de x para y
- Vetor de distância: $D_x = [D_x(y): y \in N]$
 - N sendo a vizinhança de x
- O nó x conhece o custo para cada vizinho v : $c(x,v)$
- O nó x mantém $D_x = [D_x(y): y \in N]$
- O nó x também mantém os vetores de distância de seus vizinhos
 - Para cada vizinho v , x mantém $D_v = [D_v(y): y \in N]$

Roteamento com vetor de distância

Para cada nó x:

1 Inicialização:

2 Para todos os destinos y em N:

3 $D_x(y) = c(x,y)$ /* se y não é um vizinho então $c(x,y) = \infty$ */

4 Para cada vizinho w

5 $D_w(y) = \infty$ para todos os destinos y em N

6 Para cada vizinho w

7 Envia um vetor de distâncias (DV) $D_x = [D_x(y): y \in N]$ para w

8

9 loop

10 Espera (até que ocorra uma mudança no custo do enlace ao vizinho w ou até a recepção de um vetor de distâncias do vizinho w)

12

13 Para cada y em N:

14 $D_x(y) = \min_v \{c(x,y) + D_v(y)\}$

15

16 Se $D_x(y)$ mudou para algum destino y

17 Envia um DV $D_x = [D_x(y): y \in N]$ para todos os vizinhos

18 para sempre

Roteamento com vetor de distância

- Idéia básica:

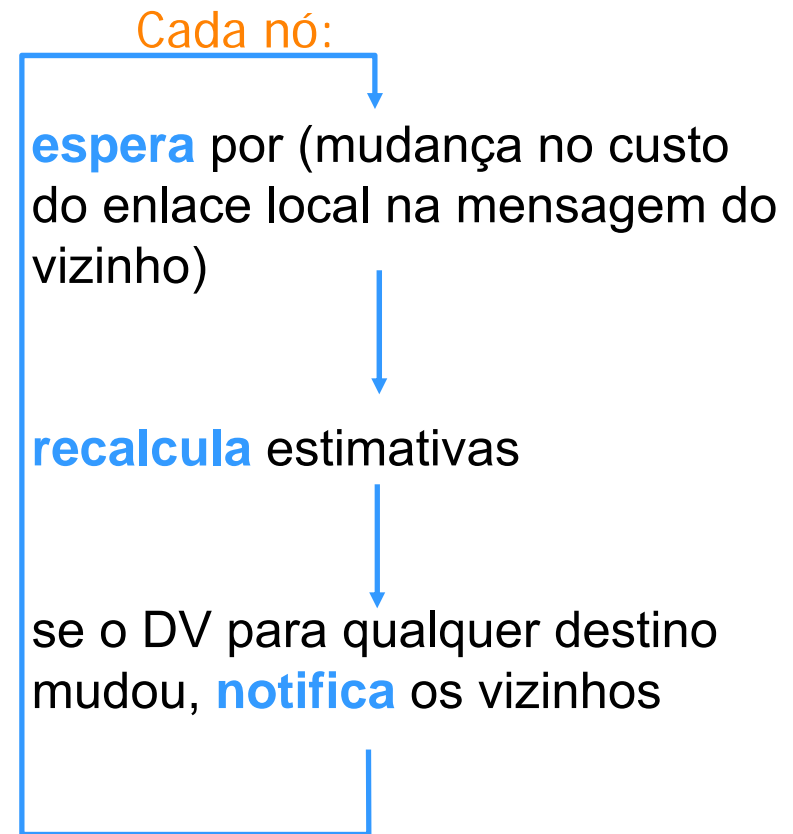
- Cada nó envia periodicamente sua própria estimativa de vetor de distância aos vizinhos
- Quando o nó x recebe nova estimativa de DV do vizinho, ele atualiza seu próprio DV usando a equação B-F:
 - $D_x(y) = \min_v \{c(x,v) + D_v(y)\}$ para cada nó $y \in N$
- Ao menos em condições naturais, a estimativa $D_x(y)$ converge para o menor custo atual $d_x(y)$

Roteamento com vetor de distância

- Iterativo, assíncrono:** cada iteração local é causada por:
- Mudança no custo do enlace local
 - Mensagem de atualização DV do vizinho

Distribuído:

- Cada nó notifica os vizinhos apenas quando seu DV mudar
- Os vizinhos então notificam seus vizinhos, se necessário



Roteamento com vetor de distância

Tabela do nó x

		Custo até		
		x	y	z
De	x	0	2	7
	y	∞	∞	∞
	z	∞	∞	∞

Tabela do nó y

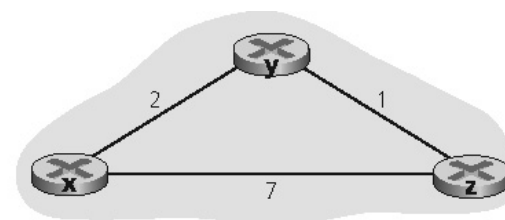
		Custo até		
		x	y	z
De	x	∞	∞	∞
	y	2	0	1
	z	∞	∞	∞

Tabela do nó z

		Custo até		
		x	y	z
De	x	∞	∞	∞
	y	∞	∞	∞
	z	7	1	0

$$D_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + D_z(y)\} \\ = \min\{2+0, 7+1\} = 2$$

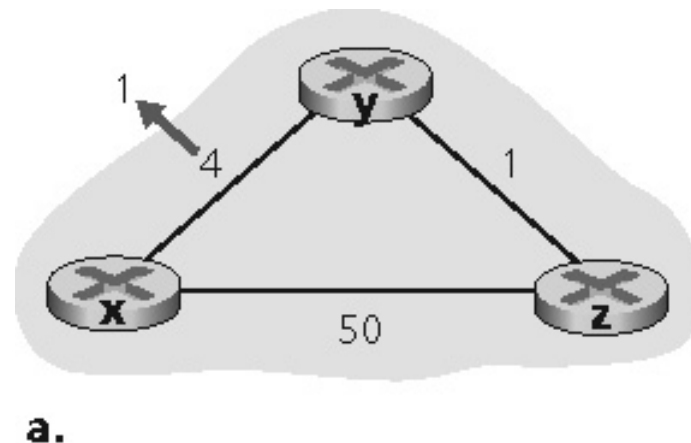
$$D_x(z) = \min\{c(x,y) + D_y(z), c(x,z) + D_z(z)\} \\ = \min\{2+1, 7+0\} = 3$$



Roteamento com vetor de distância

Mudanças no custo do enlace:

- Nó detecta mudança no custo do enlace local
- Atualiza informações de roteamento, recalcula o vetor de distância
- Se o DV muda, notifica vizinhos



No tempo t_0 , y detecta a mudança no custo do enlace, atualiza seu DV e informa seus vizinhos.

No tempo t_1 , z recebe a atualização de y e atualiza sua tabela.

Ele calcula o menor custo novo para x e envia seu DV para os vizinhos.

No tempo t_2 , y recebe a atualização de z e atualiza sua tabela de distância. O menor custo de y não muda e então y *não* envia nenhuma mensagem para z.

“boas notícias viajam depressa”



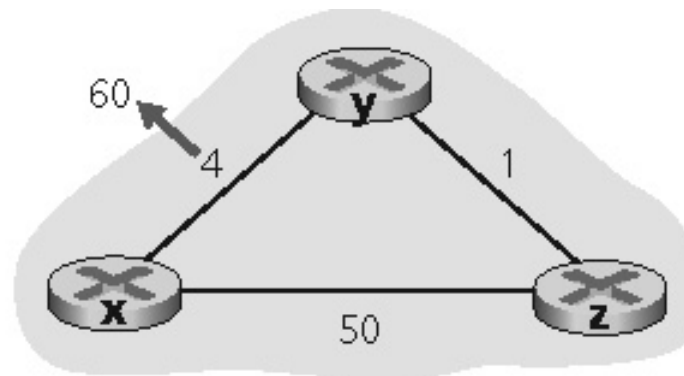
Roteamento com vetor de distância

Mudanças no custo do enlace:

- Boas notícias viajam rápido (redução no custo)
- Más notícias viajam devagar — problema da “contagem ao infinito”!
- 44 iterações antes de o algoritmo estabilizar

Reversão envenenada:

- Tentativa de solucionar esse problema
- Não funciona bem no caso geral



b.

Roteamento por Estado de Enlace

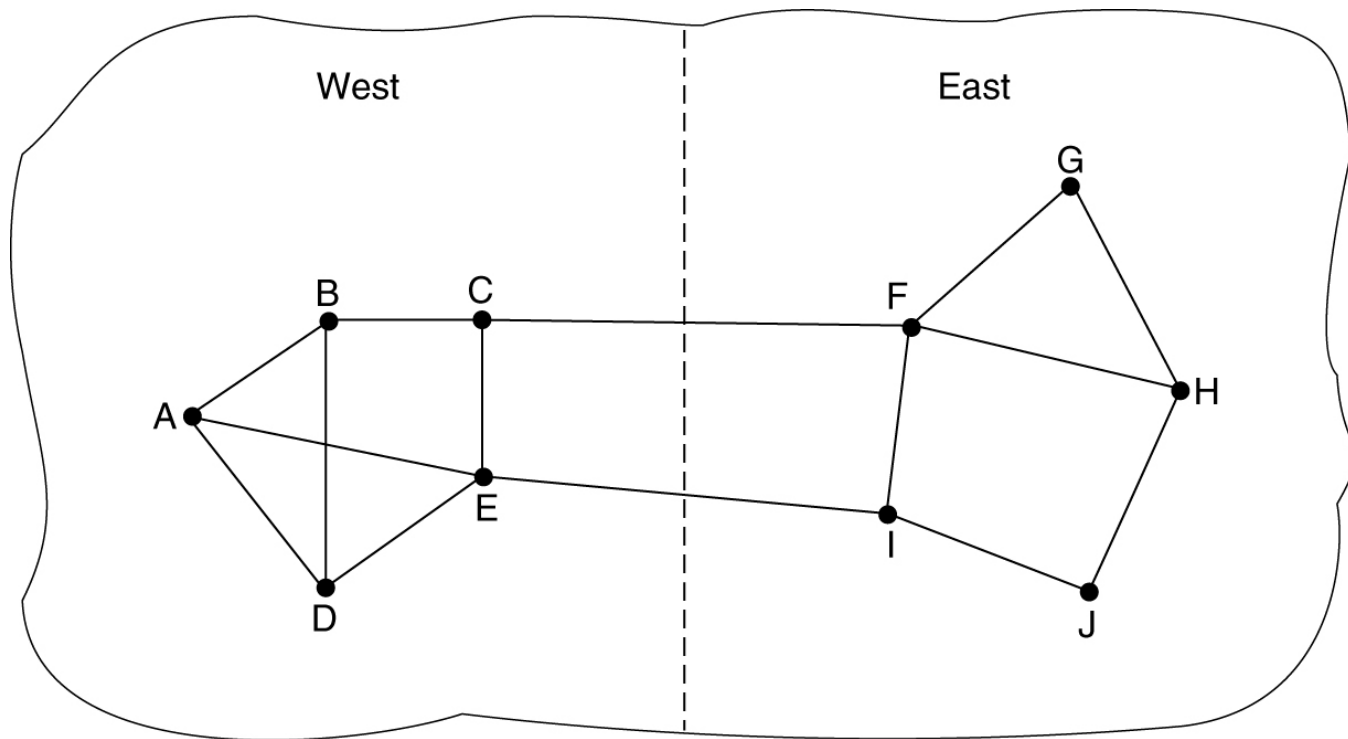
- Amplamente utilizado
- Cada roteador deve fazer o seguinte
 - Descobrir seus vizinhos e aprender seus endereços de rede
 - Medir o retardo ou o custo até cada um de seus vizinhos
 - Criar um pacote que informe tudo o que ele acabou de aprender
 - Enviar esse pacote a outros roteadores
 - Calcular o caminho mais curto até cada um dos outros roteadores (Dijkstra)

Roteamento por Estado de Enlace

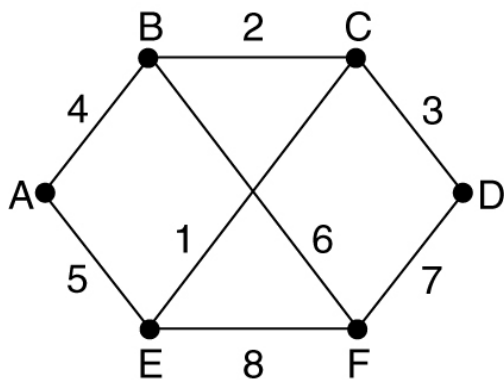
- Conhecendo os vizinhos
 - Envio de um pacote HELLO especial em cada linha ponto a ponto
 - Roteador do outro lado envia uma resposta
- Medição do custo da linha
 - Enviar um pacote ECHO
 - Mede-se o tempo e divide-se por dois
 - Pode-se levar em conta ou não o retardo das filas nos roteadores
 - No primeiro caso, o timer é iniciado quando o pacote ECHO é enfileirado
 - No segundo caso, o timer é iniciado quando o pacote ECHO atinge o início da fila

Roteamento por Estado de Enlace

Se o roteamento levar em conta o atraso da fila, as tabelas de roteamento podem oscilar entre as linhas CF e EI



Roteamento por Estado de Enlace



(a)

Link		State		Packets	
B		C		D	
Seq.		Seq.		Seq.	
Age		Age		Age	
A	4	B	2	C	3
C	2	D	3	F	7
F	6	E	1		

E		F	
Seq.		Seq.	
Age		Age	
A	5	B	6
C	1	D	7
F	8	E	8

(b)

(a) Sub-rede (b) Pacotes de estado de enlace para essa sub-rede

Números de seqüência permitem descartar pacotes velhos ou repetidos
 O campo **idade** permite que o roteador descarte informações antigas para os casos em que o roteador fique inativo, ou ocorra um problema com a seqüência.

Roteamento por Estado de Enlace

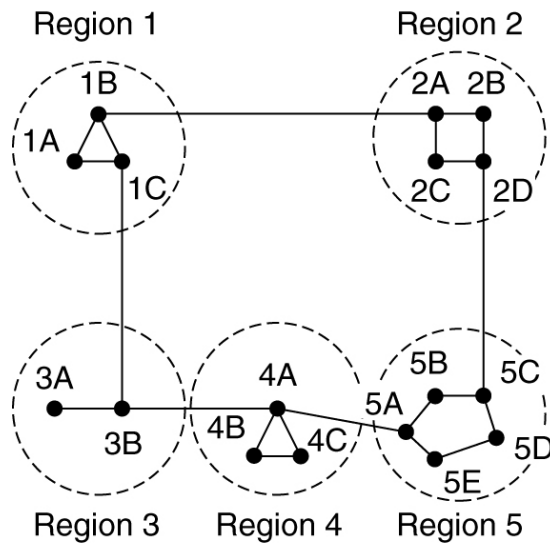
Buffer de pacotes para o roteador B

Source	Seq.	Age	Send flags			ACK flags			Data
			A	C	F	A	C	F	
A	21	60	0	1	1	1	0	0	
F	21	60	1	1	0	0	0	1	
E	21	59	0	1	0	1	0	1	
C	20	60	1	0	1	0	1	0	
D	21	59	1	0	0	0	1	1	

[Roteamento Hierárquico]

- As tabelas de roteamento se tornam maiores a medida que o número de roteadores aumenta
- A solução é segmentar os roteadores em regiões
 - Tabelas mais simples
 - Soluções de roteamento podem não ser mais ótimas

Roteamento Hierárquico



(a)

Full table for 1A

Dest.	Line	Hops
1A	—	—
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

(b)

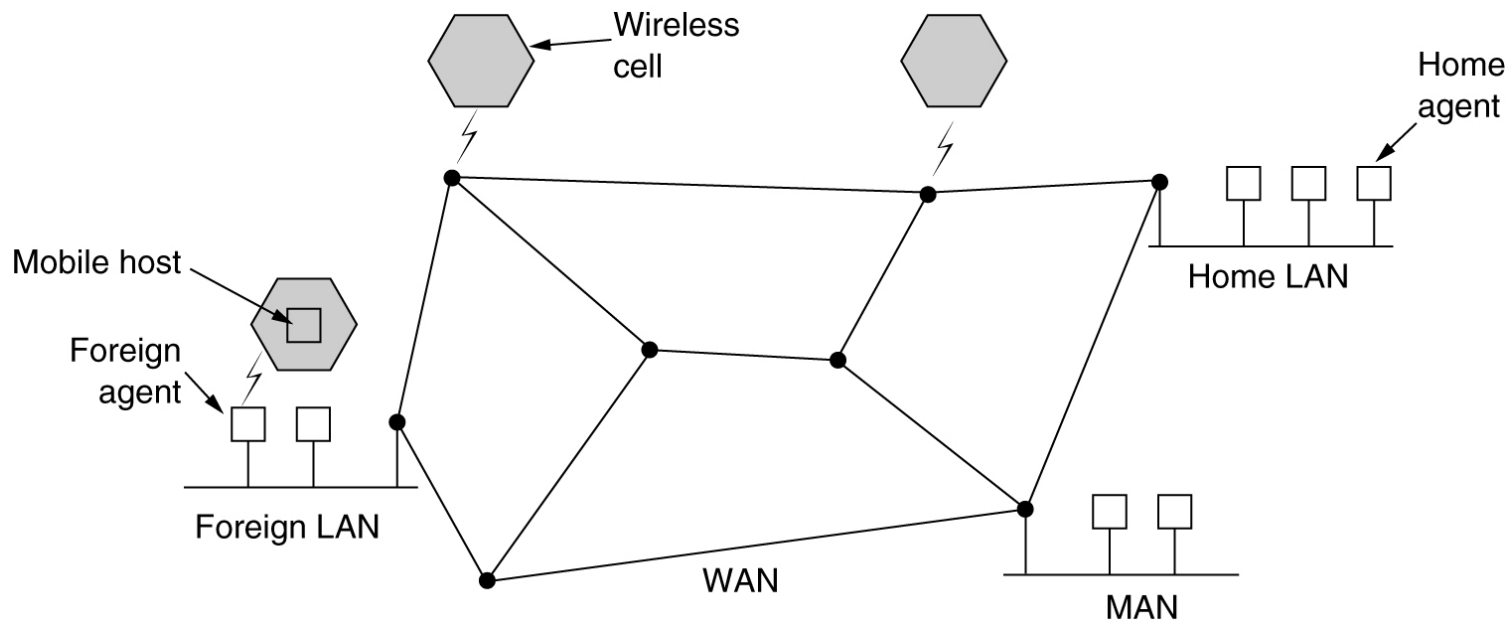
Hierarchical table for 1A

Dest.	Line	Hops
1A	—	—
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

(c)

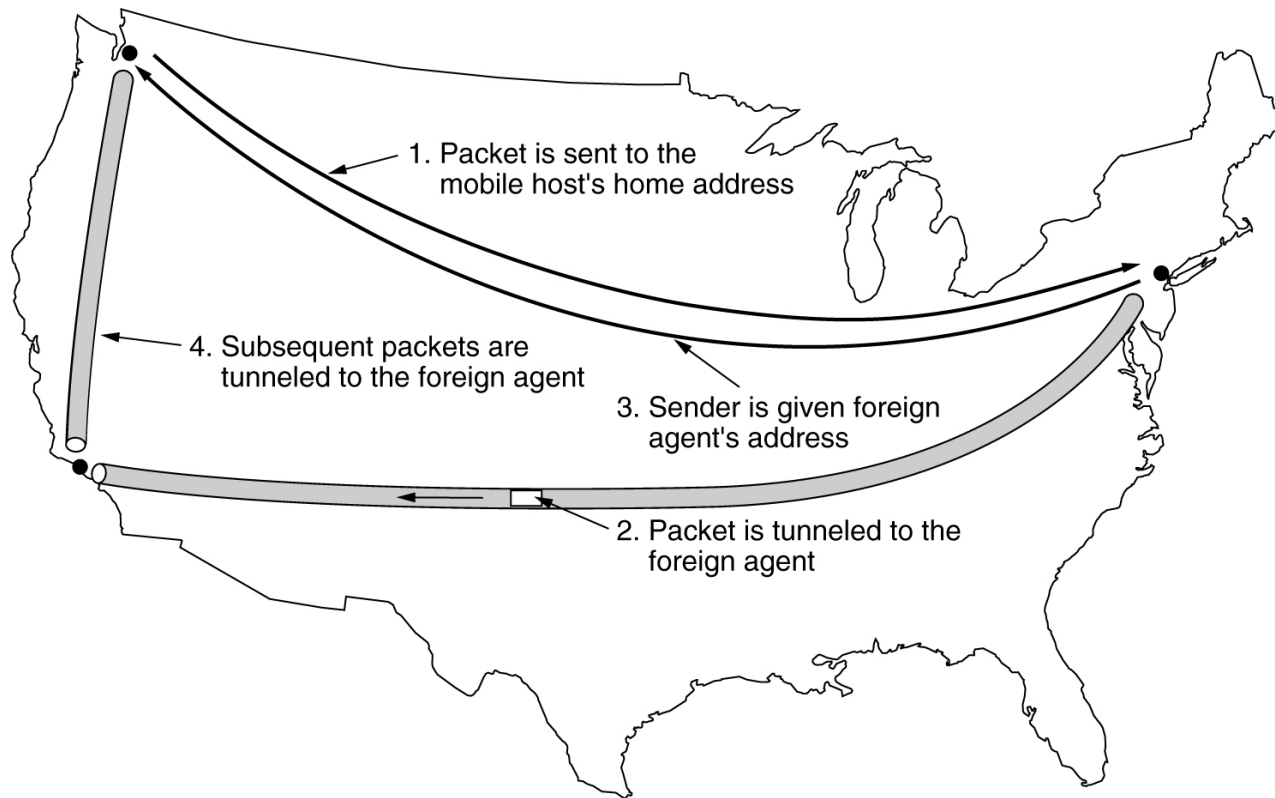
Roteamento para Hosts Móveis

Cenário típico



Roteamento para Hosts Móveis

Roteamento de pacotes para usuários móveis.



Roteamento em Redes Ad Hoc

■ Roteadores móveis

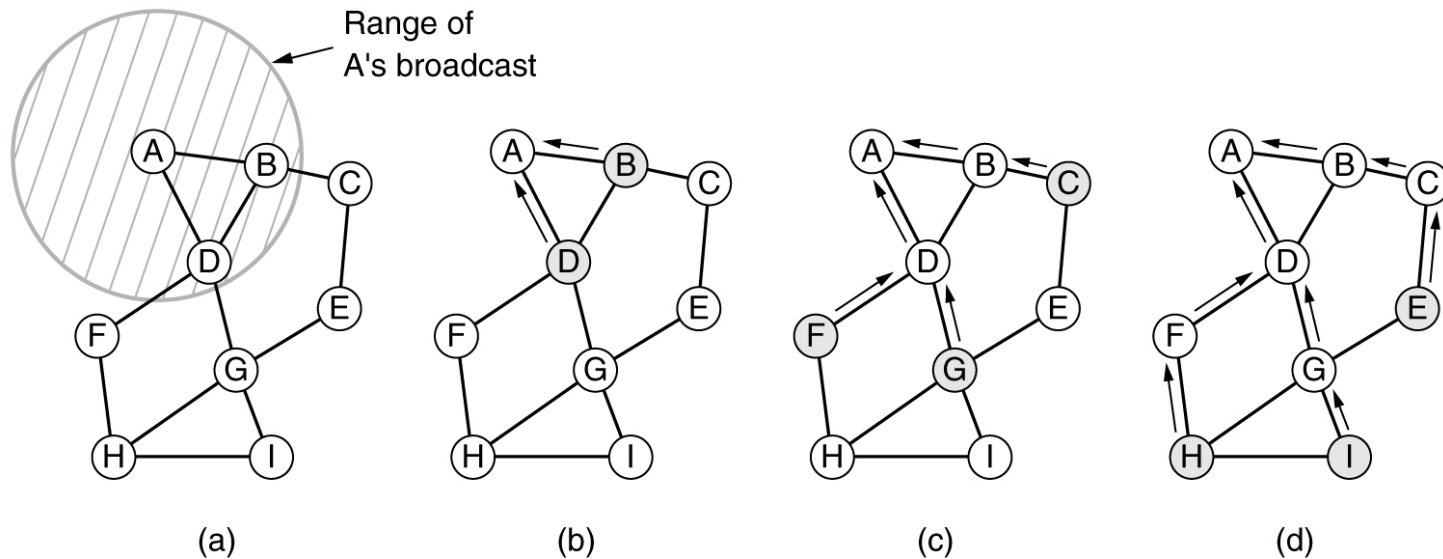
- Veículos militares em um campo de batalha sem nenhuma infra-estrutura
- Uma frota de navios no mar se movendo ao mesmo tempo
- Trabalhos em calamidades com infra-estrutura destruída
- Grupo de pessoas com notebooks usando o padrão 802.11

■ Topologia muda constantemente

■ Solução

- Algoritmo AODV – Ad hoc On-demand Distance Vector.
- Só é utilizado quando se necessita transmitir algo

Roteamento em Redes Ad Hoc (Descoberta da rota)



(a) Alcance de difusão para A. (b) Depois de B e D receberem a difusão de A. (c) Depois de C, F e G receberem a difusão de A. (d) Depois de E, H e I receberem a difusão de A. Os nós sombreados são novos destinatários. As setas mostram as rotas inversas possíveis.

Roteamento em Redes Ad Hoc (Descoberta da rota)

Source address	Request ID	Destination address	Source sequence #	Dest. sequence #	Hop count
----------------	------------	---------------------	-------------------	------------------	-----------

Formato de um pacote ROUTE REQUEST.

Roteamento em Redes Ad Hoc (Descoberta da rota)

Source address	Destination address	Destination sequence #	Hop count	Lifetime
----------------	---------------------	------------------------	-----------	----------

Formato de um pacote ROUTE REPLY.

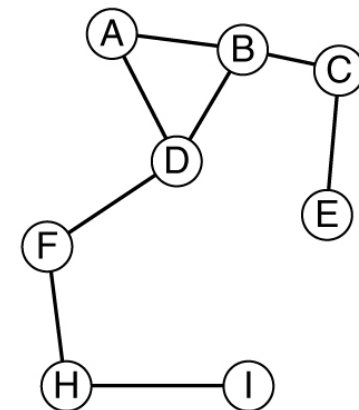
Roteamento em Redes Ad Hoc (Manutenção da rota)

(a) Tabela de roteamento de D antes de G ficar inativo

(b) O grafo depois que G fica inativo.

Dest.	Next hop	Distance	Active neighbors	Other fields
A	A	1	F, G	
B	B	1	F, G	
C	B	2	F	
E	G	2		
F	F	1	A, B	
G	G	1	A, B	
H	F	2	A, B	
I	G	2	A, B	

(a)

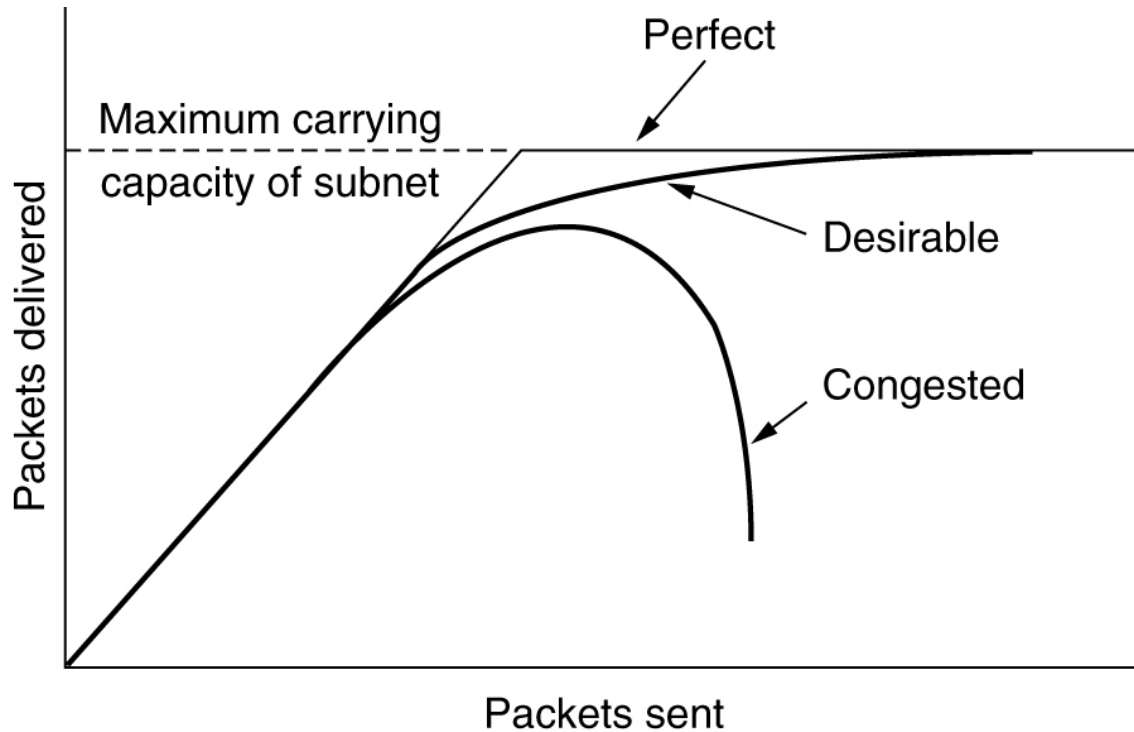


(b)

[Algoritmos de Controle de Congestionamento]

- Quando há mais pacotes do que os roteadores podem processar em uma determinada região de uma sub-rede, o desempenho cai
 - Congestionamento
- Pacotes podem ser descartados ou apresentar um retardo acima do permitido

Algoritmos de Controle de Congestionamento



Algoritmos de Controle de Congestionamento

- Causas do congestionamento
 - Processadores lentos
 - Enlaces de saída com banda insuficiente
- Controle de congestionamento
 - Técnicas usadas para que a sub-rede possa garantir que o tráfego seja encaminhado
- Controle de fluxo
 - Se refere a enlaces ponto a ponto
 - Compatibilização de velocidades

Princípios de Controle de Congestionamento

- Soluções podem ser classificadas em
 - Malha aberta
 - Decisões são tomadas sem levar em conta o estado atual da rede
 - Malha fechada
 - Decisões são tomadas com base na realimentação de algum parâmetro como: a percentagem de pacotes descartados, a média do comprimento das filas, o número de pacotes que atingem o **timeout**, etc.

Princípios de Controle de Congestionamento

- Com a realimentação, a ação de controle de congestionamento é dividida em
 - Monitorar o sistema para detectar quando e onde ocorre congestionamento
 - Enviar essas informações para lugares onde alguma providência possa ser tomada
 - Ajustar a operação do sistema para corrigir o problema

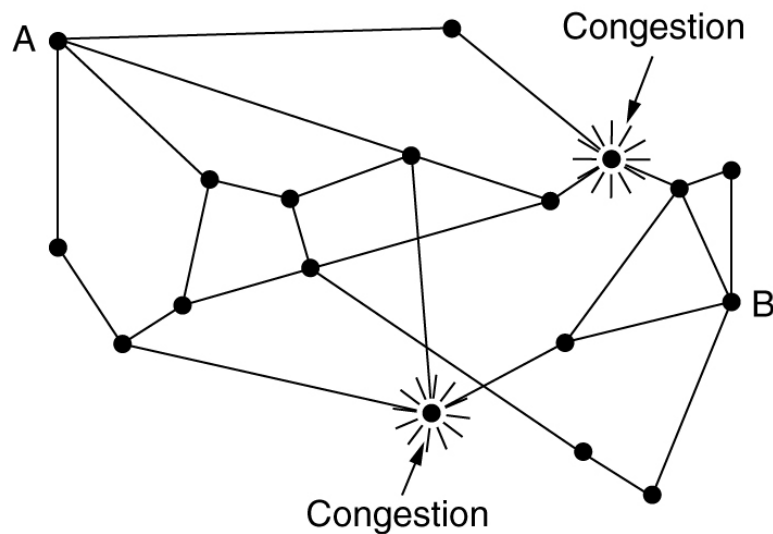
Políticas de Prevenção de Congestionamento

Layer	Policies
Transport	<ul style="list-style-type: none">• Retransmission policy• Out-of-order caching policy• Acknowledgement policy• Flow control policy• Timeout determination
Network	<ul style="list-style-type: none">• Virtual circuits versus datagram inside the subnet• Packet queueing and service policy• Packet discard policy• Routing algorithm• Packet lifetime management
Data link	<ul style="list-style-type: none">• Retransmission policy• Out-of-order caching policy• Acknowledgement policy• Flow control policy

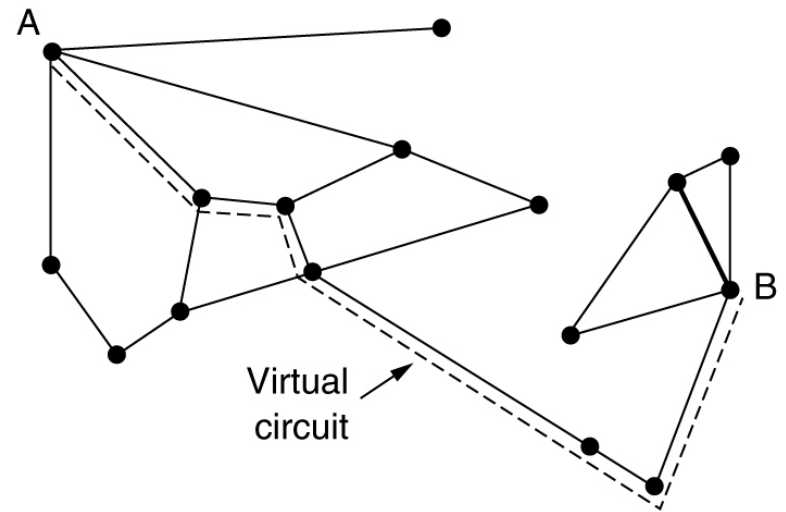
[Controle de Congestionamento em Redes de Circuitos Virtuais]

- Controle de admissão
 - Quando houver sinal de congestionamento, nenhum outro circuito virtual será estabelecido
 - Idéia empregada no sistema telefônico
- Uma modificação dessa idéia é criar novos circuitos virtuais evitando áreas problemáticas
- Outra possibilidade é negociar uma certa qualidade de serviço no estabelecimento do circuito virtual

Controle de Congestionamento em Redes de Circuitos Virtuais



(a)



(b)

(a) Rede Congestionada

(b) Rede redesenhada que elimina o congestionamento

[Controle de Congestionamento em Redes de Datagramas]

- Os roteadores monitoram as suas linhas de saída e associam alguns parâmetros de desempenho a uma variável de controle
 - Valor acima de um limiar indica que há congestionamento
- Detectado o congestionamento, várias ações podem ser tomadas

[Controle de Congestionamento em Redes de Datagramas]

■ Bit de advertência

- Bit de advertência é colocado nos pacotes de confirmação
- A origem das transmissões faz uma média e reduz a taxa de transmissão se necessário

■ Pacotes reguladores

- Pacote regulador é enviado pelo roteador ao host de origem
- A cada pacote regulador recebido o host reduz a sua taxa em $x\%$

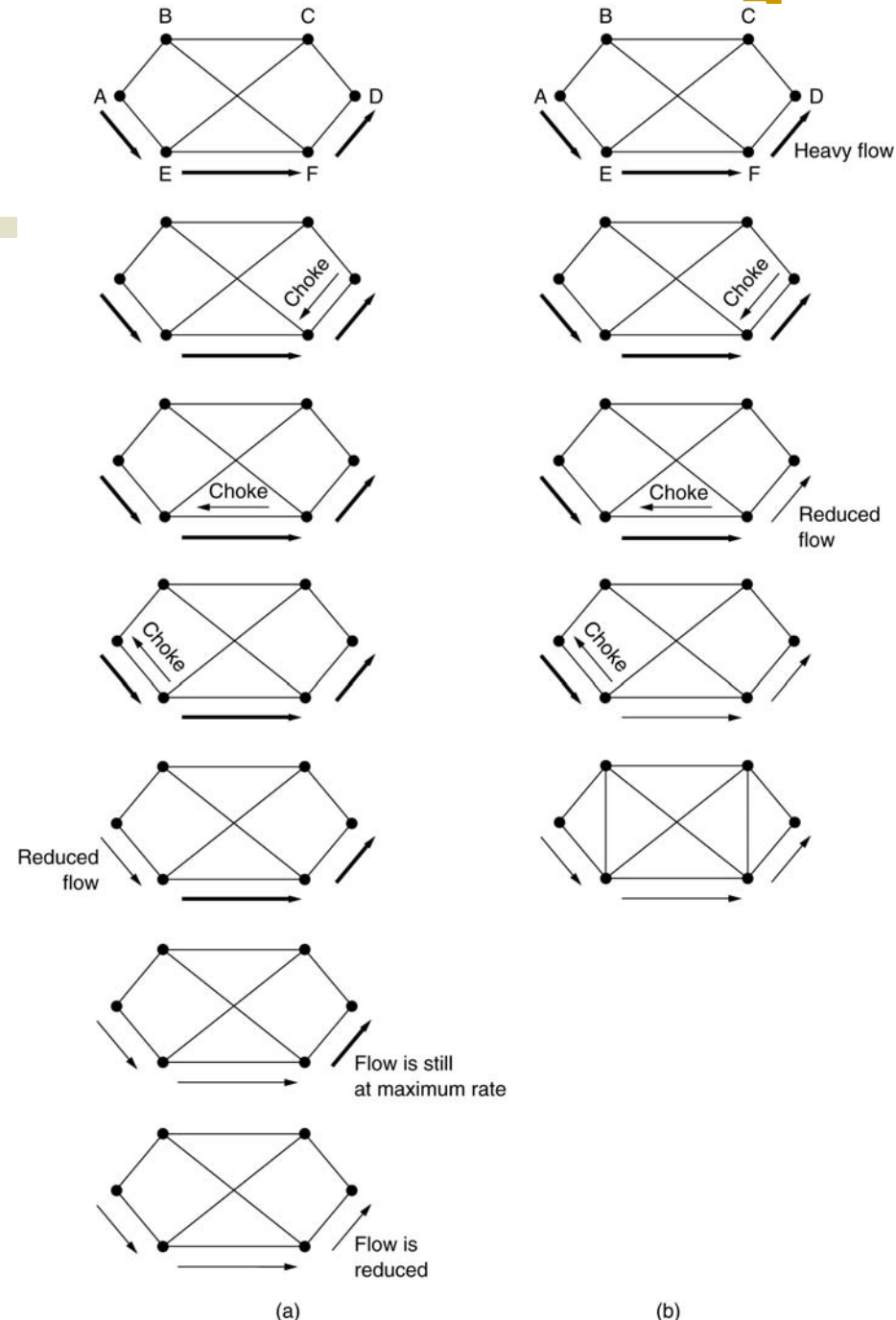
[Controle de Congestionamento em Redes de Datagramas]

- Pacotes reguladores salto a salto
 - Pacotes reguladores do roteador congestionado ao host de origem podem demorar para chegar
 - Nesse tempo o host continua mandando dados
 - A redução da taxa em pontos intermediários alivia o destino final mais rapidamente

Pacotes reguladores

(a) Um pacote regulador que afeta apenas a origem

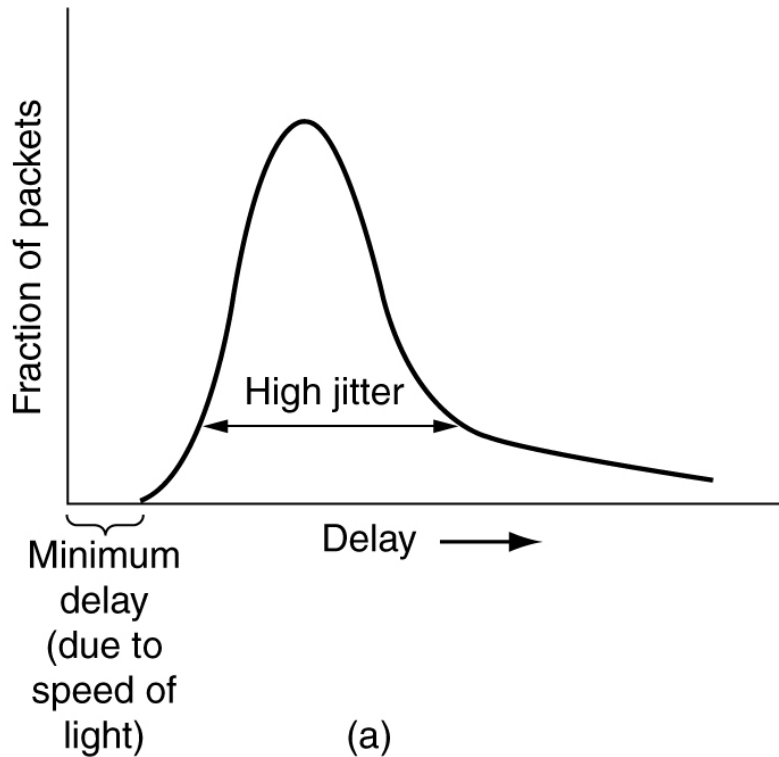
(b) Um pacote regulador que afeta cada salto por onde passa.



[Controle de Congestionamento em Redes de Datagramas]

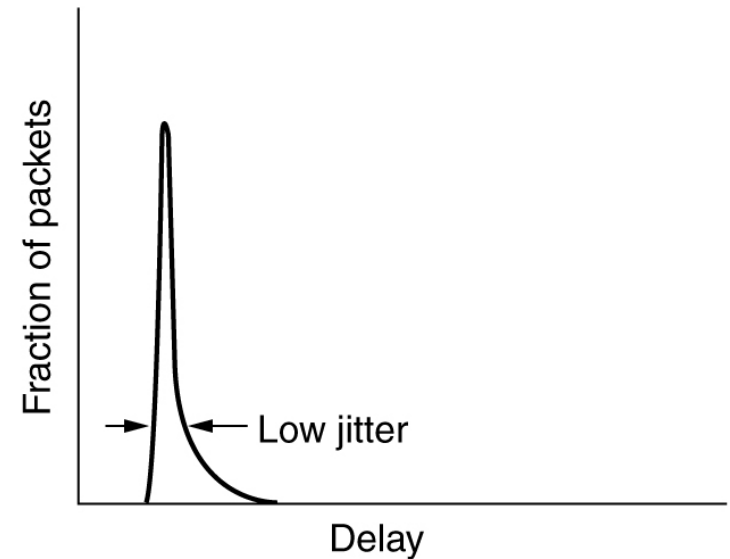
- Escoamento de carga
 - Pacotes devem ser descartados quando os roteadores não puderem lidar com eles
 - Esquemas de prioridade podem ser usados para um descarte mais inteligente
- Detecção aleatória prematura
 - Roteadores descartam pacotes aleatoriamente antes da linha ficar congestionada
- Controle de Jitter (Flutuação)

Controle de Congestionamento em Redes de Datagramas



(a)

(a) Jitter alto



(b)

(b) Jitter baixo

Qualidade de Serviço

Rigidez dos requisitos de qualidade de serviço

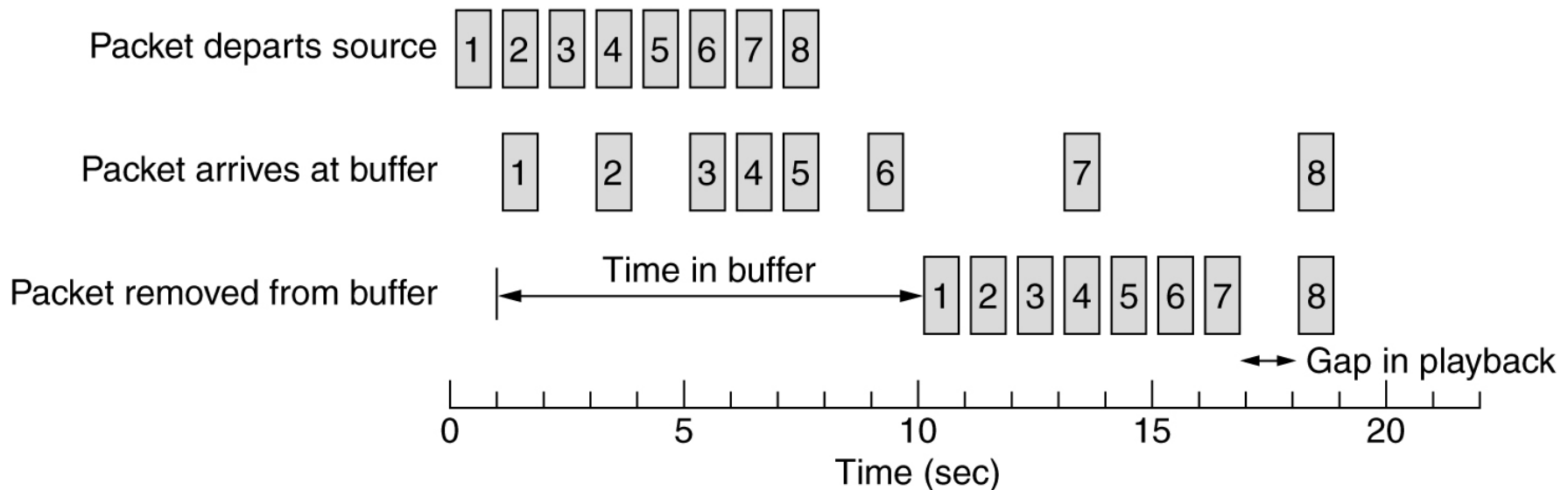
Application	Reliability	Delay	Jitter	Bandwidth
E-mail	High	Low	Low	Low
File transfer	High	Low	Low	Medium
Web access	High	Medium	Low	Medium
Remote login	High	Medium	Medium	Low
Audio on demand	Low	Low	High	Medium
Video on demand	Low	Low	High	High
Telephony	Low	High	High	Low
Videoconferencing	Low	High	High	High

[Qualidade de Serviço]

- Técnicas para alcançar qualidade de serviço
 - Superdimensionamento
 - Solução dispendiosa
 - Armazenamento em buffers
 - Permite lidar com o jitter
 - Moldagem de tráfego
 - Tráfego uniforme

[Qualidade de Serviço]

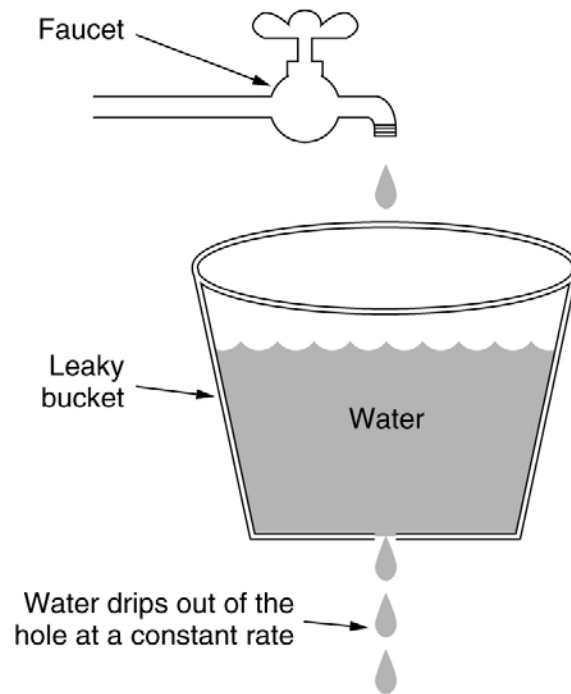
O fluxo de saída é suavizado por meio do armazenamento em buffers



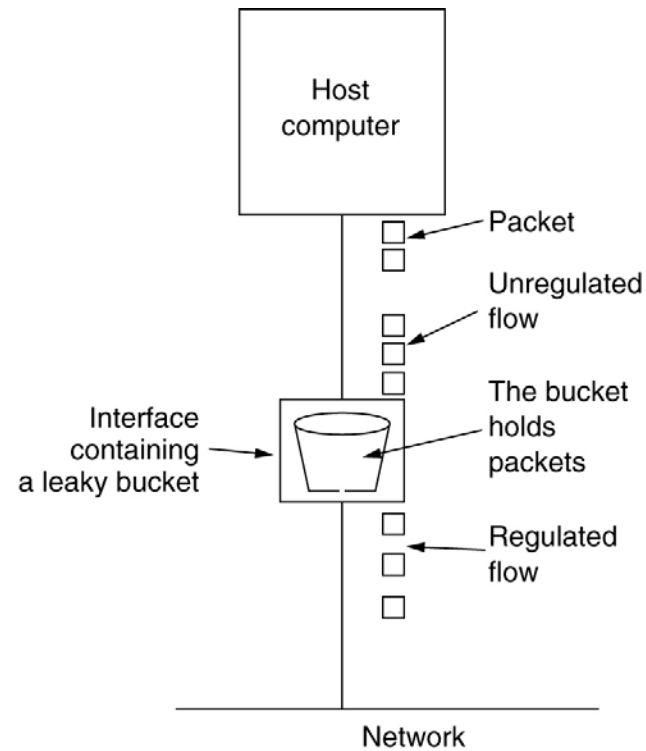
[Qualidade de Serviço]

- Algoritmo do balde furado
 - Pacotes são armazenados até o balde encher
 - A saída é regulada pelo furo no balde
- Algoritmo do balde de símbolos
 - O balde armazena símbolos
 - Para um pacote ser transmitido, é necessário um símbolo do balde

[Algoritmo do balde furado]

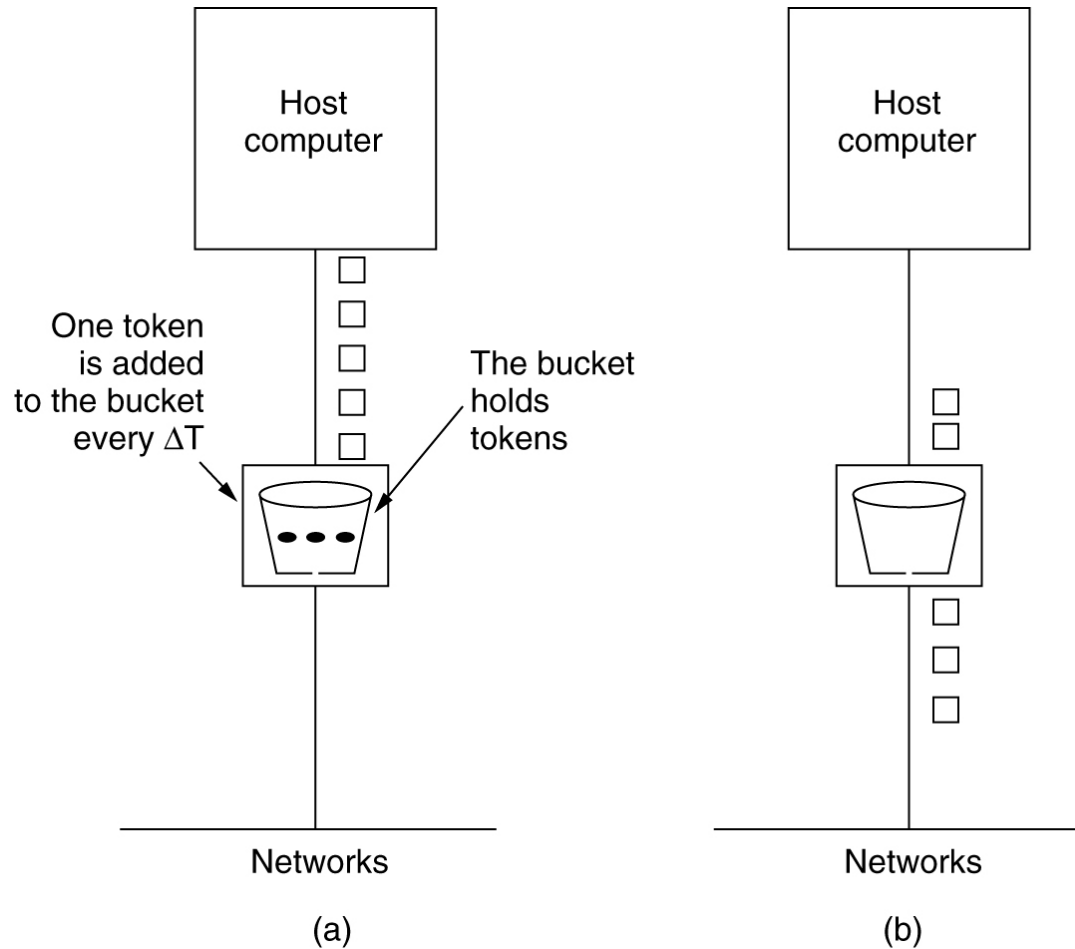


(a)



(b)

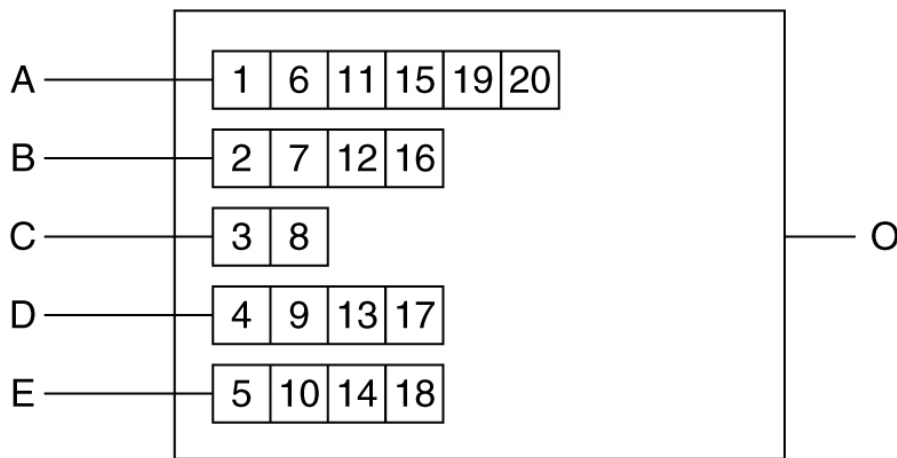
Algoritmo do balde de símbolos



[Qualidade de Serviço]

■ Programação de pacotes

- O objetivo é garantir justiça quando o roteador lida com vários fluxos
- Algoritmo do enfileiramento justo



(a)

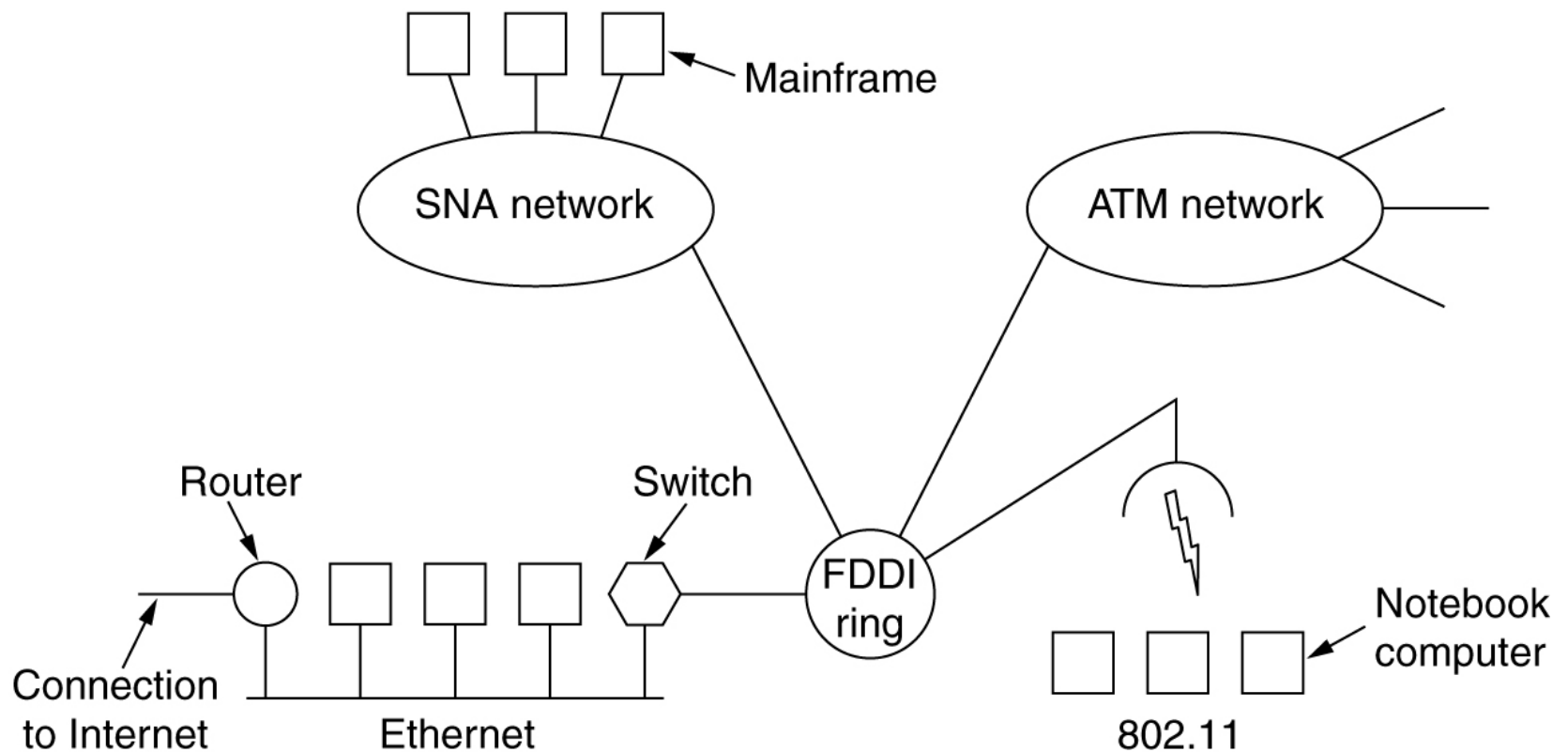
Packet	Finishing time
C	8
B	16
D	17
E	18
A	20

(b)

[Interligação de Redes]

- Existe uma quantidade abundante de tipos de redes
 - TCP/IP, redes ATM, LANs com os protocolos Novell NCP/IPX, etc.
- Por questões de custos, é preferível muitas vezes trabalhar com diversos tipos de redes do que uniformizar
- As tecnologias são muitas vezes distintas e os serviços oferecidos são incompatíveis

[Interligação de Redes]



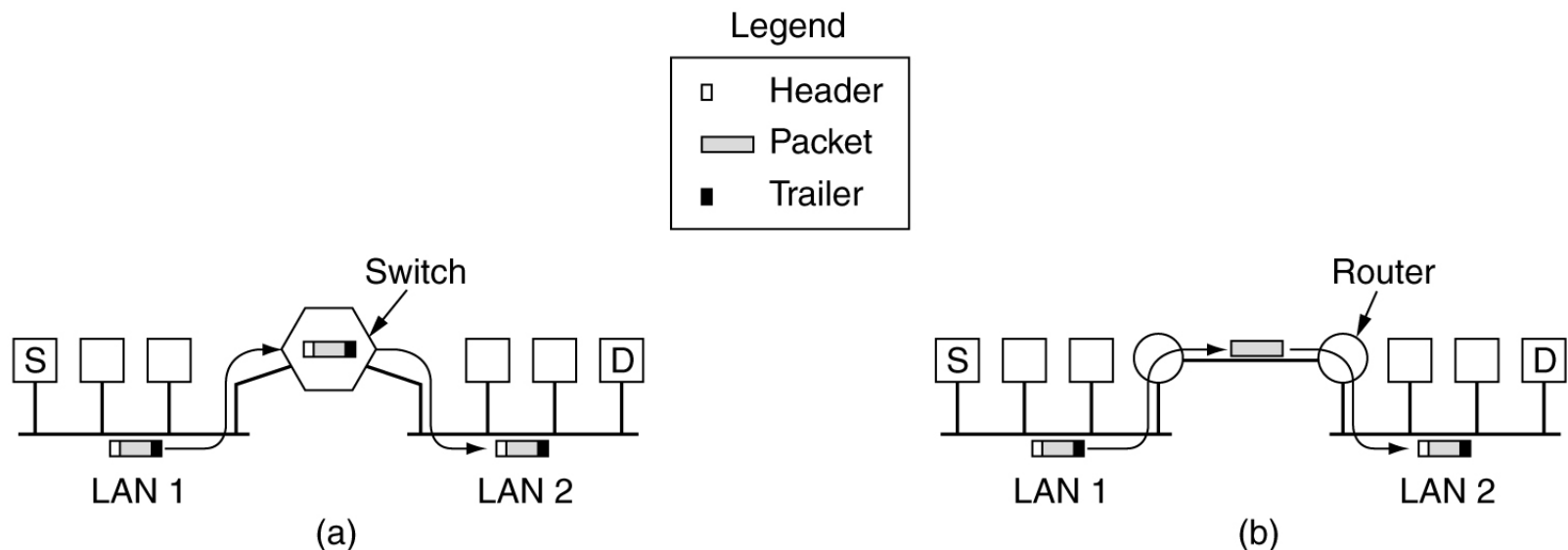
Interligação de Redes

Diferenças entre redes

Item	Some Possibilities
Service offered	Connection oriented versus connectionless
Protocols	IP, IPX, SNA, ATM, MPLS, AppleTalk, etc.
Addressing	Flat (802) versus hierarchical (IP)
Multicasting	Present or absent (also broadcasting)
Packet size	Every network has its own maximum
Quality of service	Present or absent; many different kinds
Error handling	Reliable, ordered, and unordered delivery
Flow control	Sliding window, rate control, other, or none
Congestion control	Leaky bucket, token bucket, RED, choke packets, etc.
Security	Privacy rules, encryption, etc.
Parameters	Different timeouts, flow specifications, etc.
Accounting	By connect time, by packet, by byte, or not at all

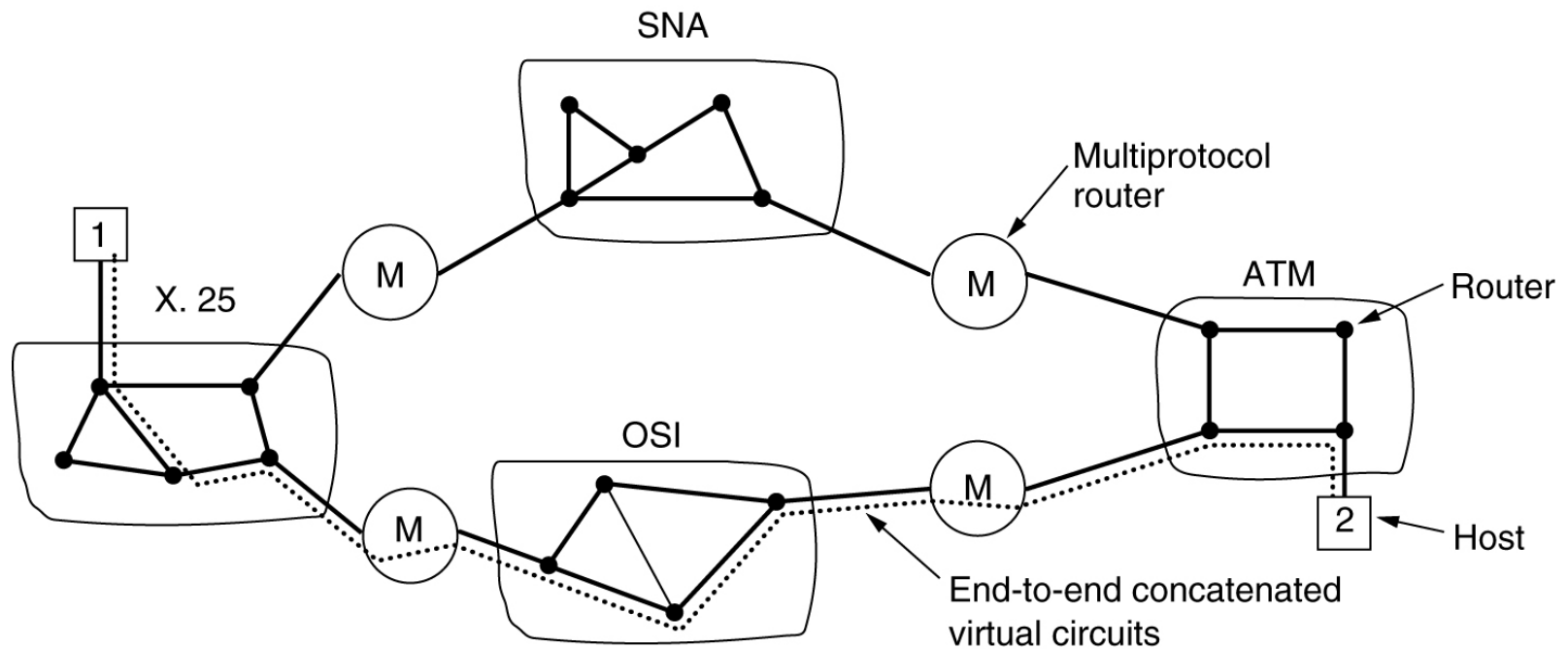
[Interligação de Redes]

- (a) Duas redes Ethernet conectadas por um switch.
- (b) Duas redes Ethernet conectadas por roteadores.



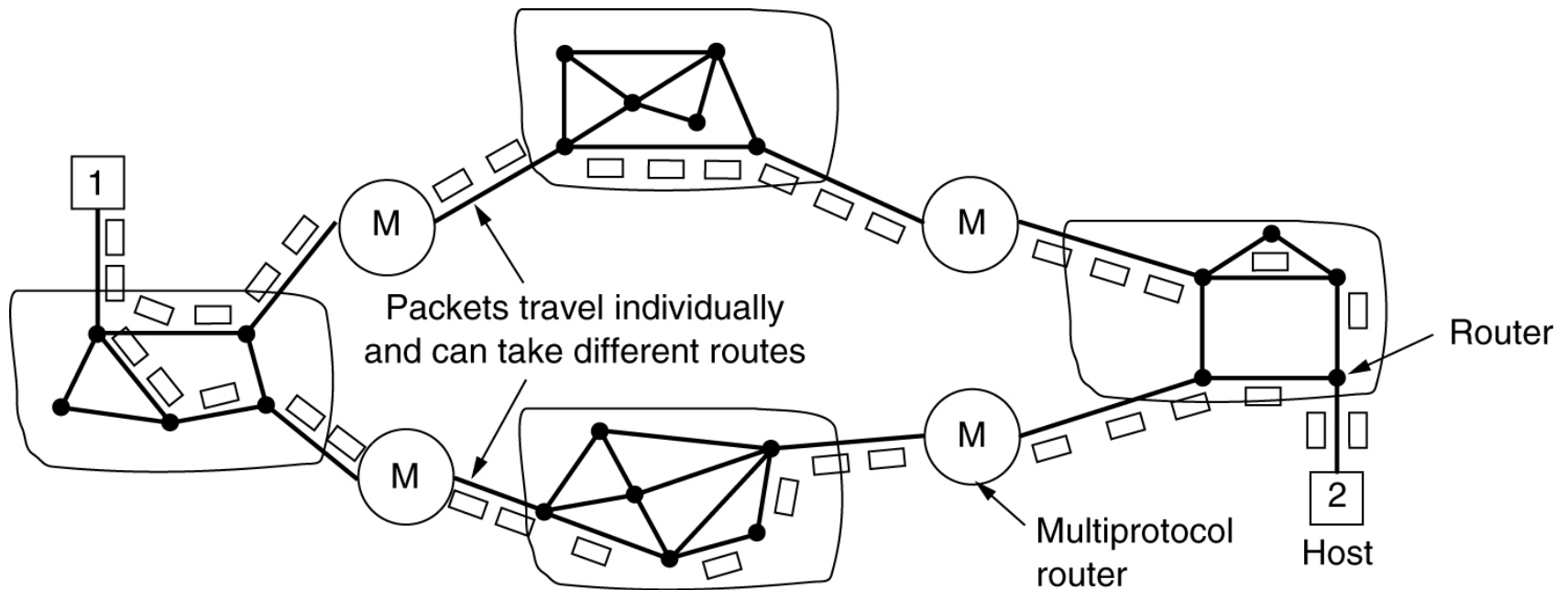
Interligação de Redes

Interligação através de circuitos virtuais concatenados



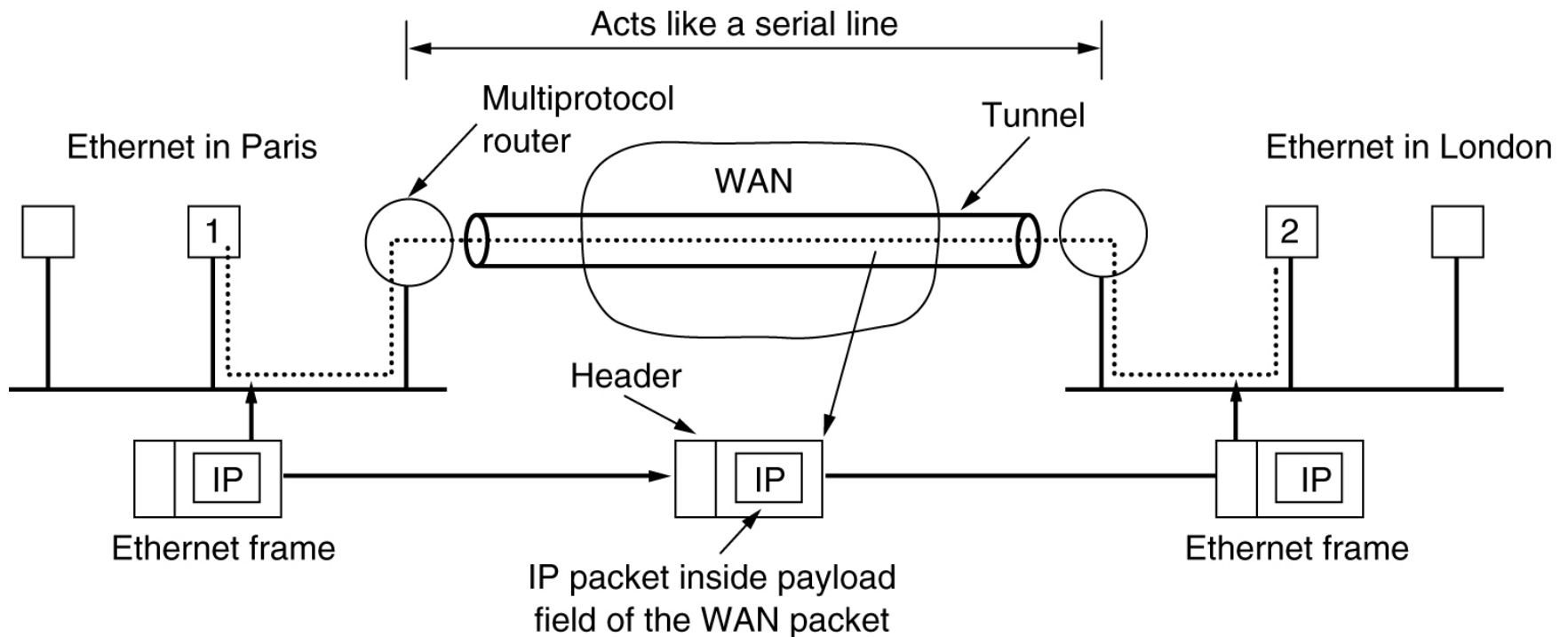
[Interligação de Redes]

Interligação de redes sem conexões



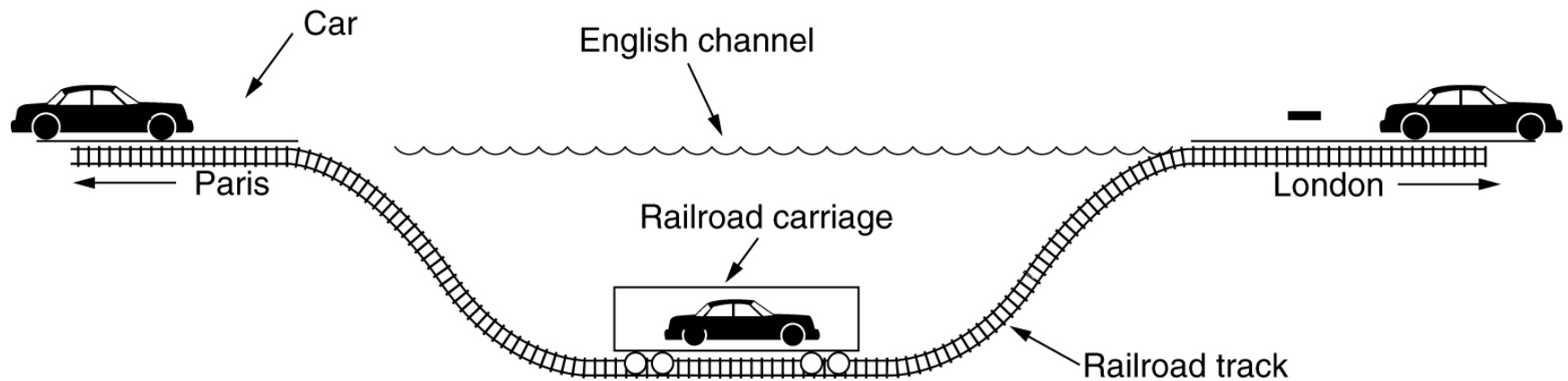
[Interligação de Redes]

Tunelamento



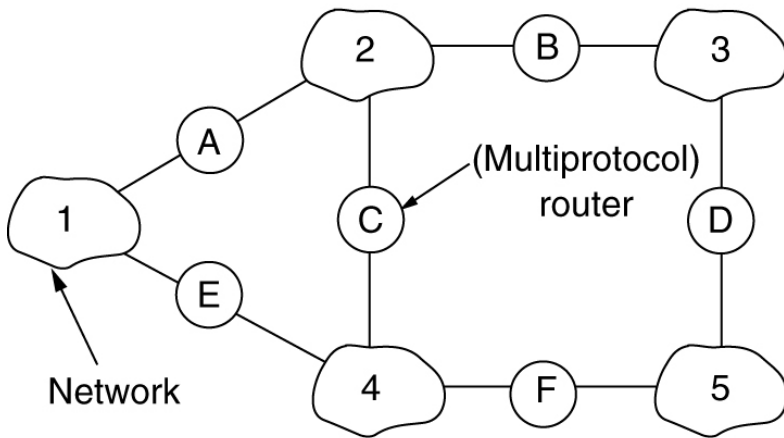
[Interligação de Redes]

Tunelamento

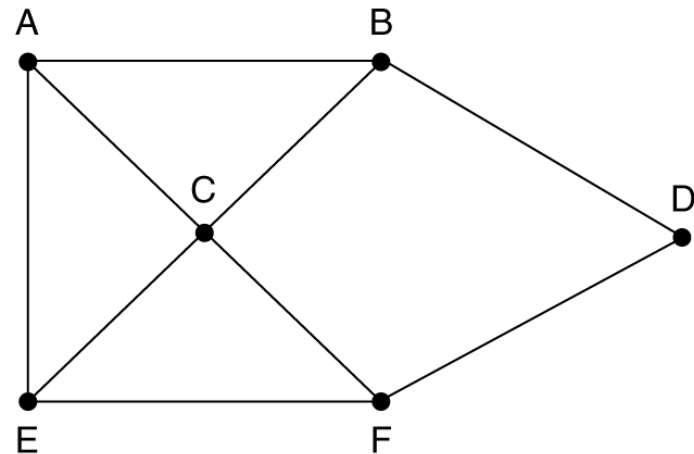


[Roteamento Inter-Redes]

(a) An internetwork. (b) A graph of the internetwork.



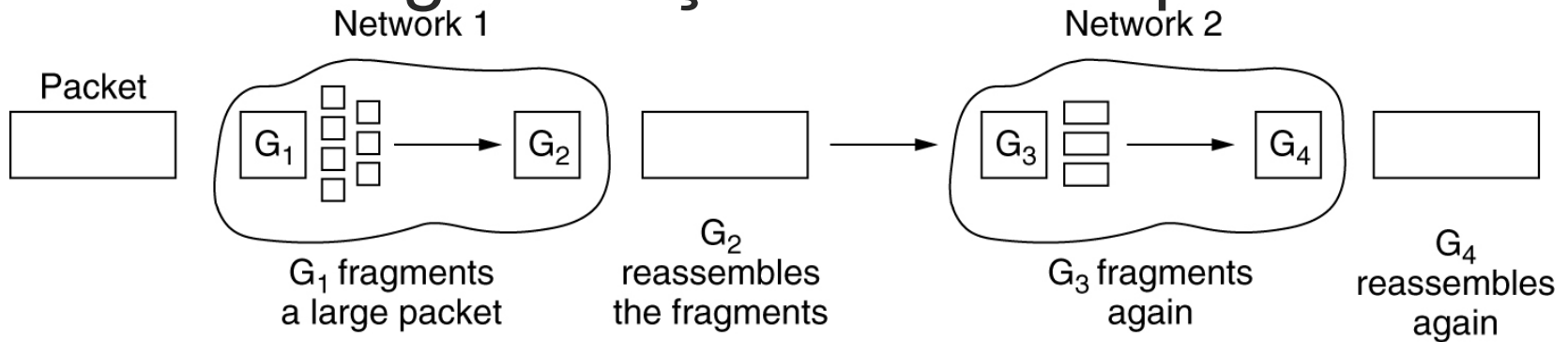
(a)



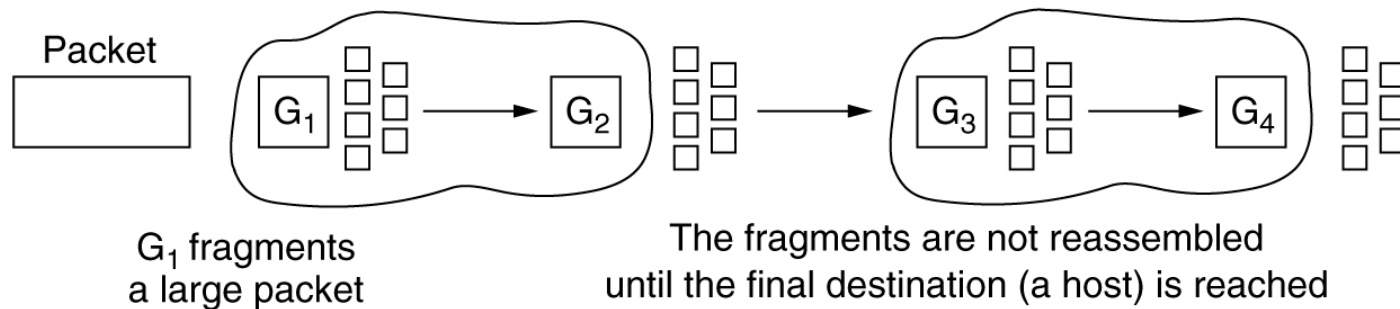
(b)

[Fragmentação]

(a) Fragmentação transparente. (b) Fragmentação não transparente.

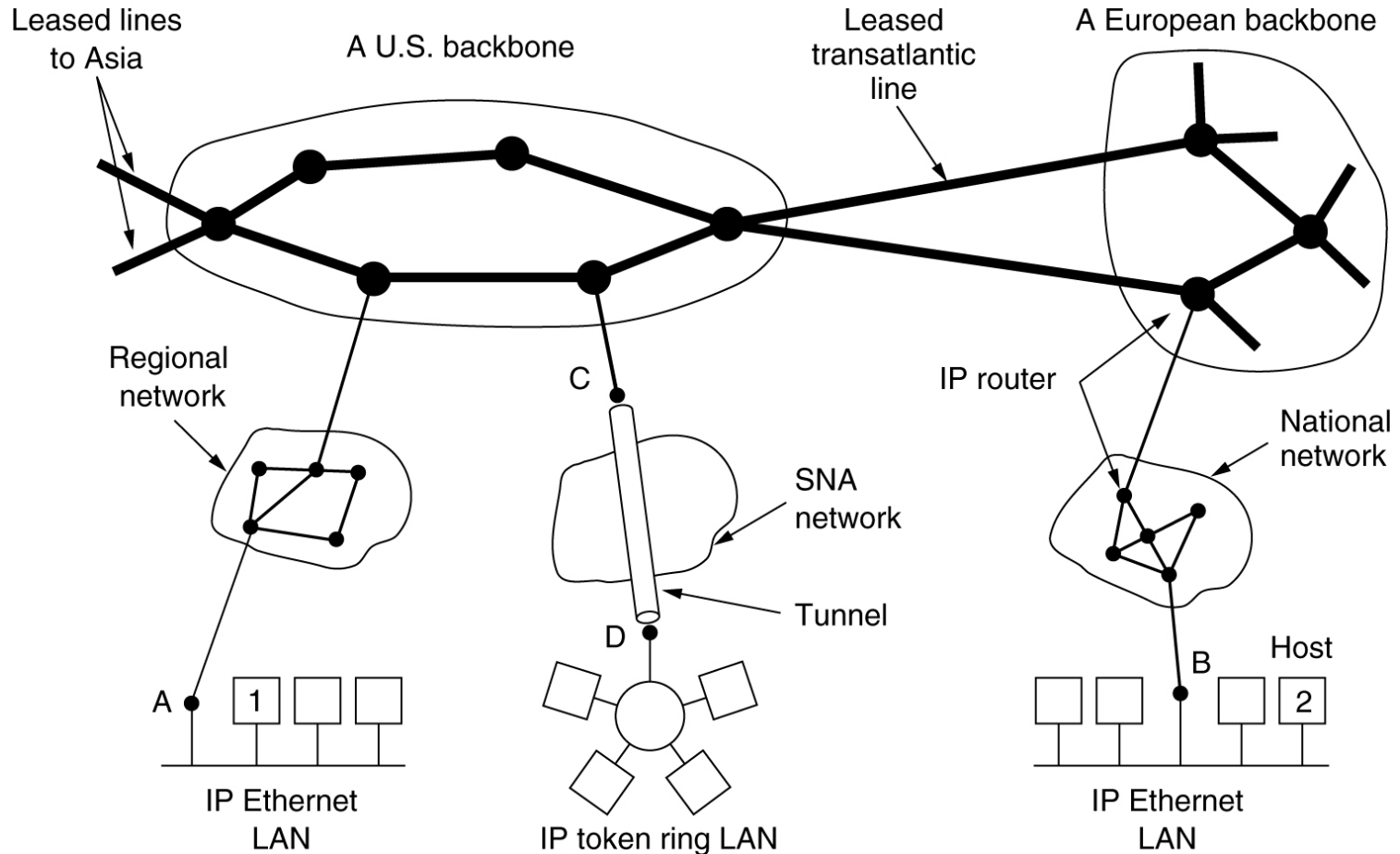


(a)



(b)

Camada de Rede da Internet



A Internet é uma coleção de redes interconectadas

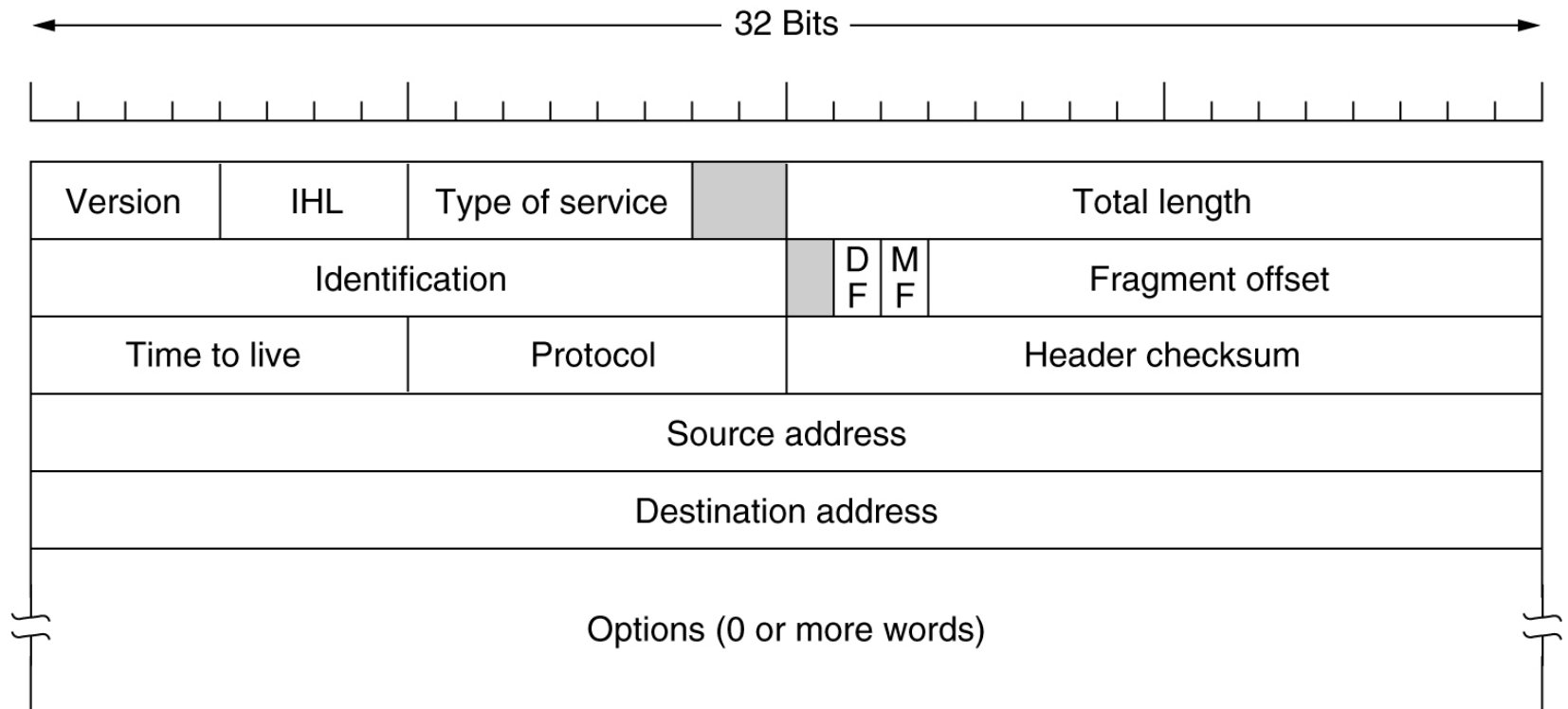
[Camada de Rede da Internet]

- Protocolo IP

- Ponto de união da Internet
- Todos os computadores interligados implementam este protocolo
- Atualmente ainda predomina a versão 4 (ipv4)
- A versão 6 (ipv6) está substituindo o ipv4
 - Maior capacidade de endereçamento
 - Versões atuais dos Sistemas Operacionais já implementam o ipv6

[Protocolo IP]

O Cabeçalho IPv4.



[Protocolo IP]

Possibilidades de uso do campo
“options” (limitado a 40 bytes)

Option	Description
Security	Specifies how secret the datagram is
Strict source routing	Gives the complete path to be followed
Loose source routing	Gives a list of routers not to be missed
Record route	Makes each router append its IP address
Timestamp	Makes each router append its address and timestamp

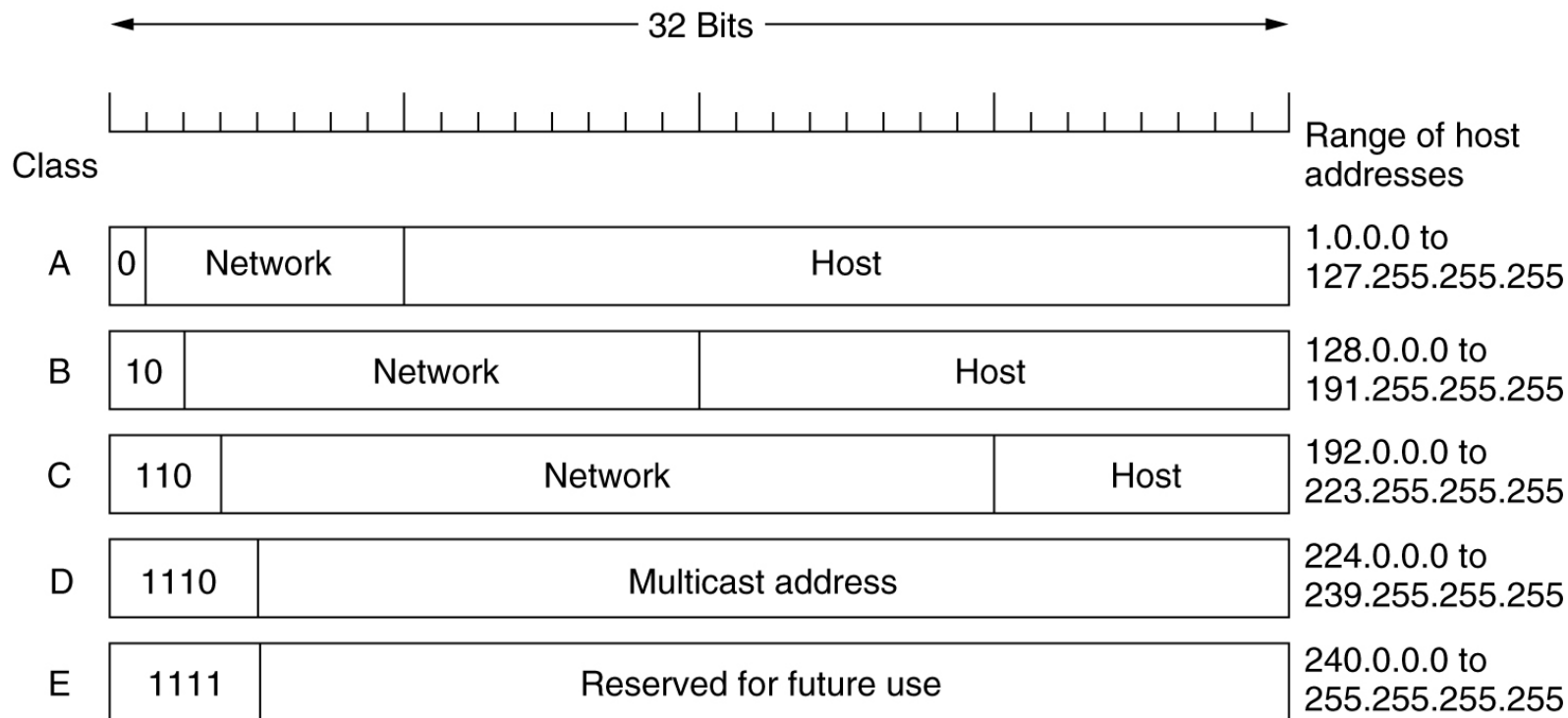
[Protocolo IP]

■ Endereços IP

- Número de 32 bits que identifica unicamente uma interface de rede na Internet
- Utiliza-se a notação decimal com pontos
 - xxx.yyy.zzz.kkk
 - xxx representa o valor decimal de um número de 8 bits
- Parte comum identifica a rede
- Originalmente divididos em classes
- Atribuídos pela ICANN (Internet Corporation for Assigned Names and Numbers)

[Endereços IP]

Classes de endereços IP

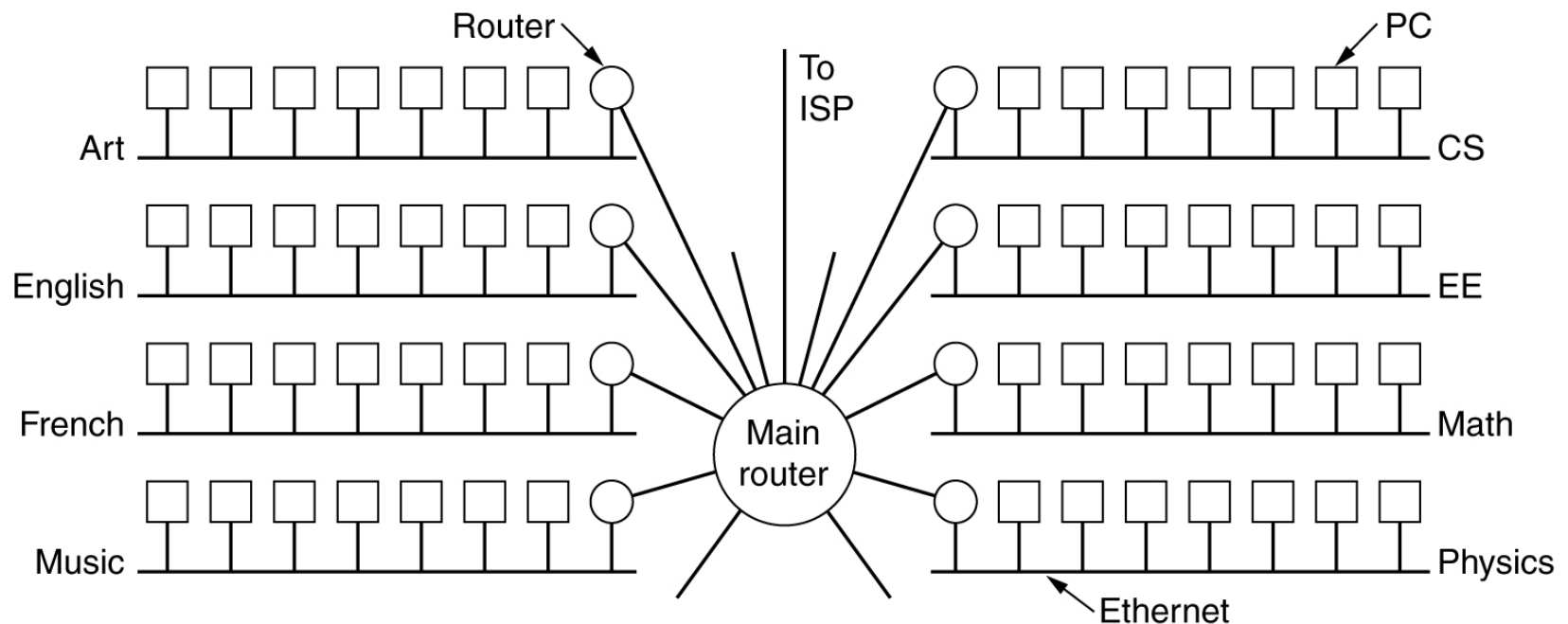


[Sub-Redes]

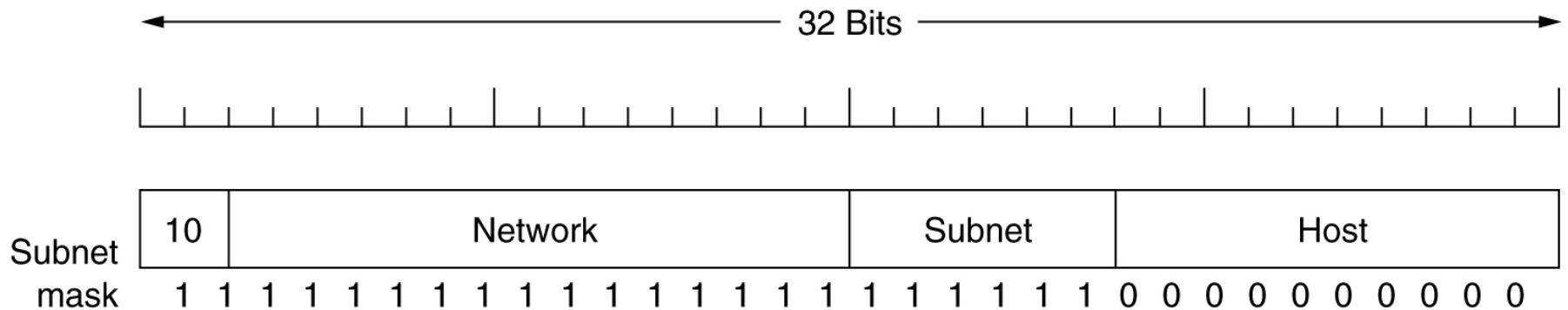
- Pode-se dividir uma rede em sub-redes
 - Bits usados para os hosts são usados para as sub-redes
 - Externamente, as sub-redes são vistas como uma única rede
 - A utilização de uma máscara da sub-rede permite distinguir as partes do endereço referente à sub-rede e ao host

[Sub-Redes]

Uma rede de um campus dividida em sub-redes de departamentos



[Sub-Redes]



Uma rede de classe B dividida em 64 sub-redes

[Endereços IP]

- A divisão em classes não proporciona uma utilização adequada do espaço de endereçamento
 - Endereços de classe C são insuficientes para muitas organizações
 - Endereços de classe B são superdimensionados
- Solução para melhoria do aproveitamento
 - CIDR (Classless InterDomain Routing)
 - Alocação de endereços independente das classes

[Endereços IP]

■ CIDR

- A porção de endereço de rede tem tamanho arbitrário
- Formato do endereço: a.B.C.D/x, em que x é o número de bits na parte de rede do endereço



200.23.16.0/23

[Endereços IP]

Atribuição de endereços IP usando o CIDR

University	First address	Last address	How many	Written as
Cambridge	194.24.0.0	194.24.7.255	2048	194.24.0.0/21
Edinburgh	194.24.8.0	194.24.11.255	1024	194.24.8.0/22
(Available)	194.24.12.0	194.24.15.255	1024	194.24.12/22
Oxford	194.24.16.0	194.24.31.255	4096	194.24.16.0/20

[Endereços IP]

- Conseqüências do CIDR
 - Melhor aproveitamento dos endereços
 - Roteamento mais complicado
- Roteamento com classes
 - Deslocar o endereço IP 28 bits para a direita
 - 4 bits restantes identificam a classe
 - AND com a máscara resulta no endereço de rede
 - Endereço de rede é procurado na tabela do roteador para uma classe específica

[Endereços IP]

- Roteamento com CIDR
 - Tabela contém prefixos das redes
 - Endereço IP é comparado com todos prefixos da tabela
 - O maior prefixo que combina com o IP pesquisado corresponde à rede em que o pacote será encaminhado
 - Agregação pode ser usada para simplificar as tabelas de roteamento

[Endereços IP]

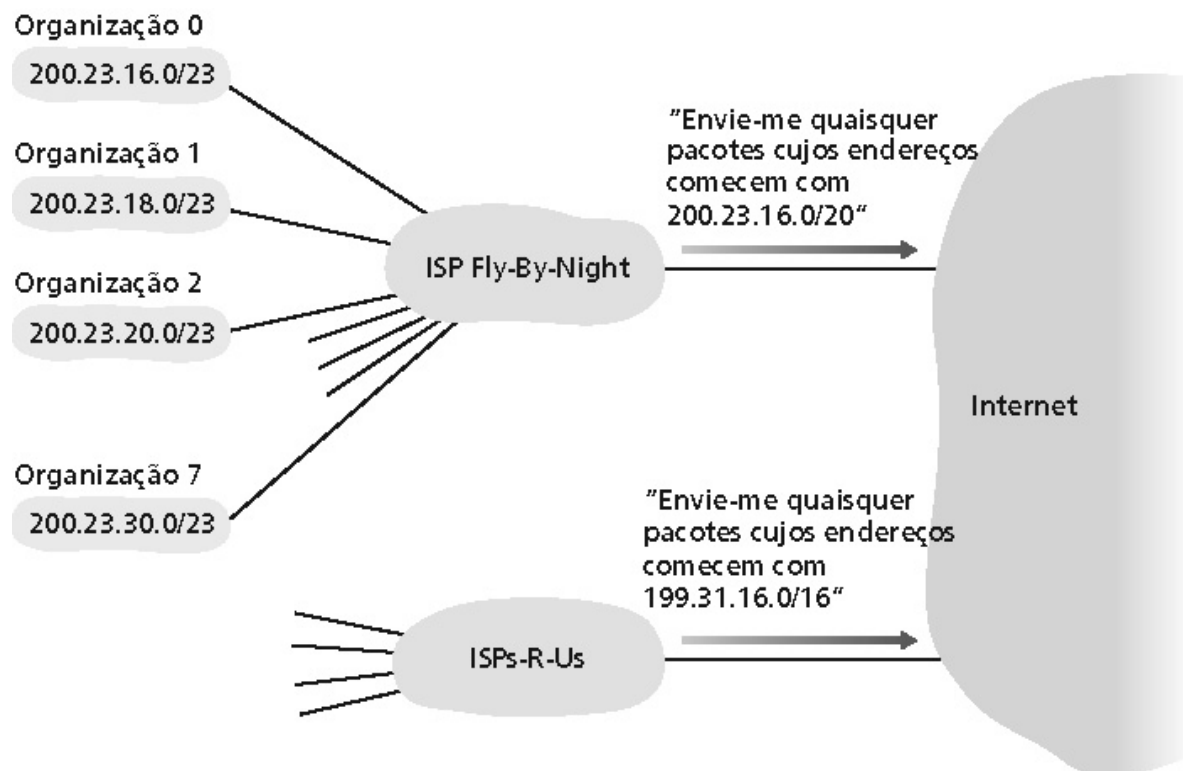
Prefix Match	Link Interface
11001000 00010111 00010	0
11001000 00010111 00011000	1
11001000 00010111 00011	2
otherwise	3

Qual Interface serão enviados os pacotes com IP?

- a) 200.23.22.161
- b) 200.23.24.170

[Endereços IP]

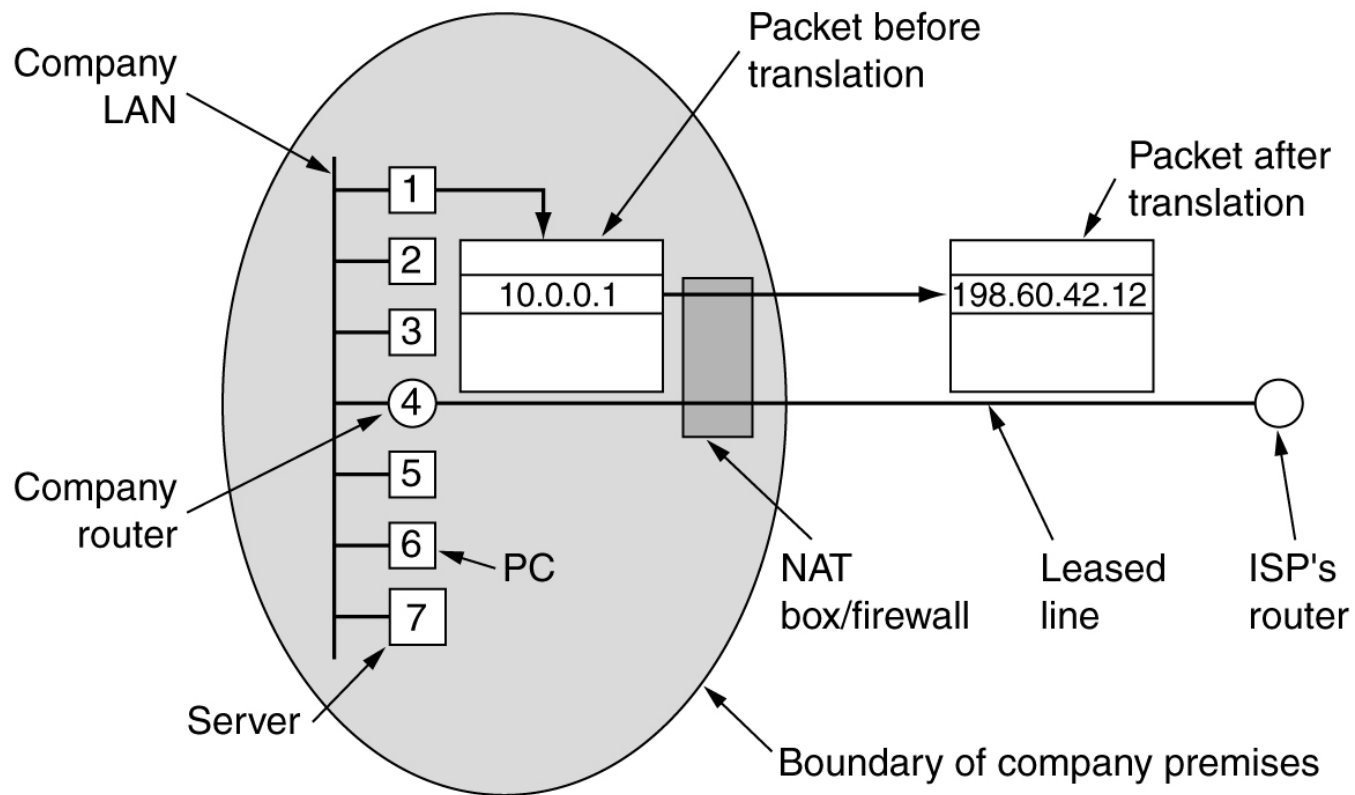
Agregação de Rotas



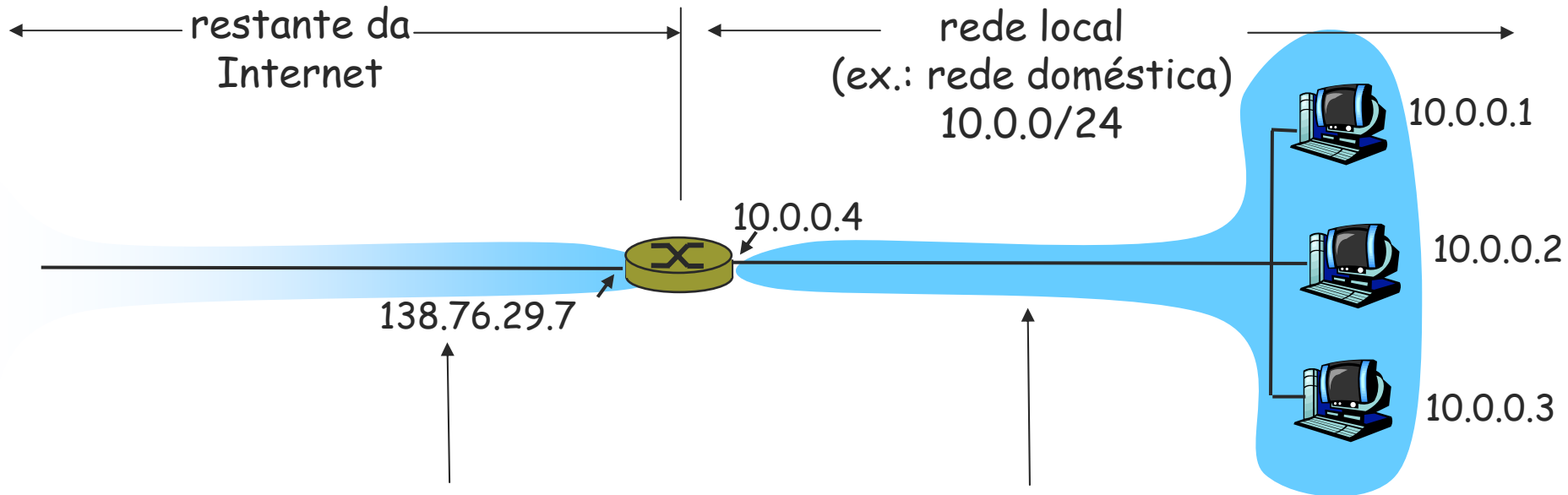
NAT – Network Address Translation

- Usuários domésticos possuem LANs em casa e interconectam suas redes com os seus ISPs
 - ISP fornece um único IP
- NAT permite que se possa ter um conjunto de IPs para uma rede doméstica
 - Endereços são válidos apenas internamente
 - São utilizadas as faixas reservadas
 - 10.0.0.0/8
 - 172.16.0.0/12
 - 192.168.0.0/16

[NAT]



[NAT]

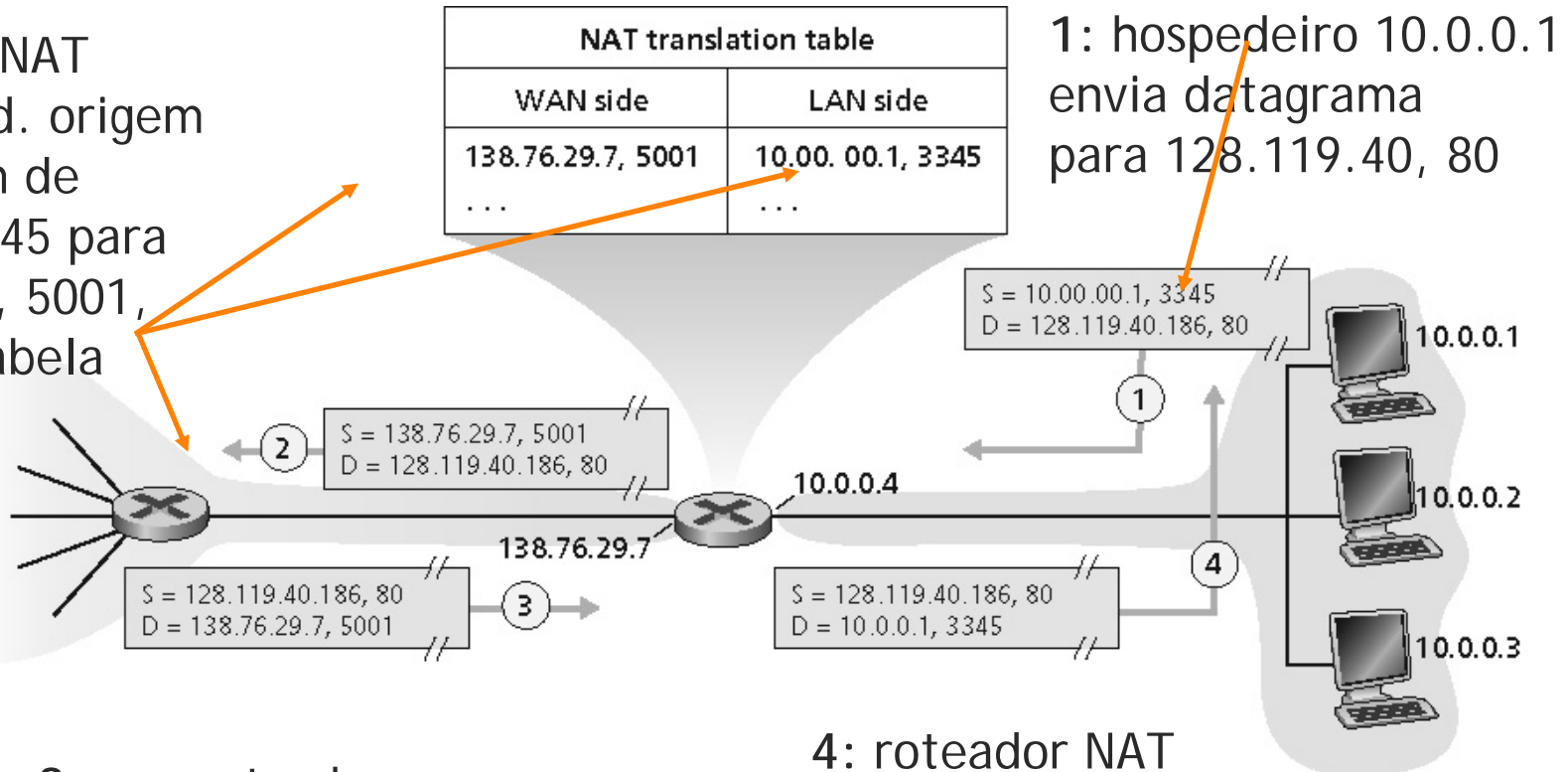


todos os datagramas que **saem** da rede local possuem o **mesmo** e único endereço IP do NAT de origem: 138.76.29.7, números diferentes de portas de origem

datagramas com origem ou destino nesta rede possuem endereço 10.0.0/24 para origem, destino (usualmente)

[NAT]

2: roteador NAT substitui end. origem do datagram de 10.0.0.1, 3345 para 138.76.29.7, 5001, atualiza a tabela



[Protocolos de Controle]

- ICMP (Internet Control Message Protocol)
- ARP (Address Resolution Protocol)
- RARP, BOOTP e DHCP

[ICMP]

- Usado por computadores e roteadores para troca de informação de controle da camada de rede
 - Error reporting: hospedeiro, rede, porta ou protocolo
 - Echo request/reply (usado pela aplicação ping)
- Transporte de mensagens:
 - Mensagens ICMP transportadas em datagramas IP
- **ICMP message:** tipo, código, mais primeiros 8 bytes do datagrama IP que causou o erro

Tipo	Código	Descrição
0	0	echo reply (ping)
3	0	dest. network unreachable
3	1	dest host unreachable
3	2	dest protocol unreachable
3	3	dest port unreachable
3	6	dest network unknown
3	7	dest host unknown
4	0	source quench (congestion control - not used)
8	0	echo request (ping)
9	0	route advertisement
10	0	router discovery
11	0	TTL expired
12	0	bad IP header

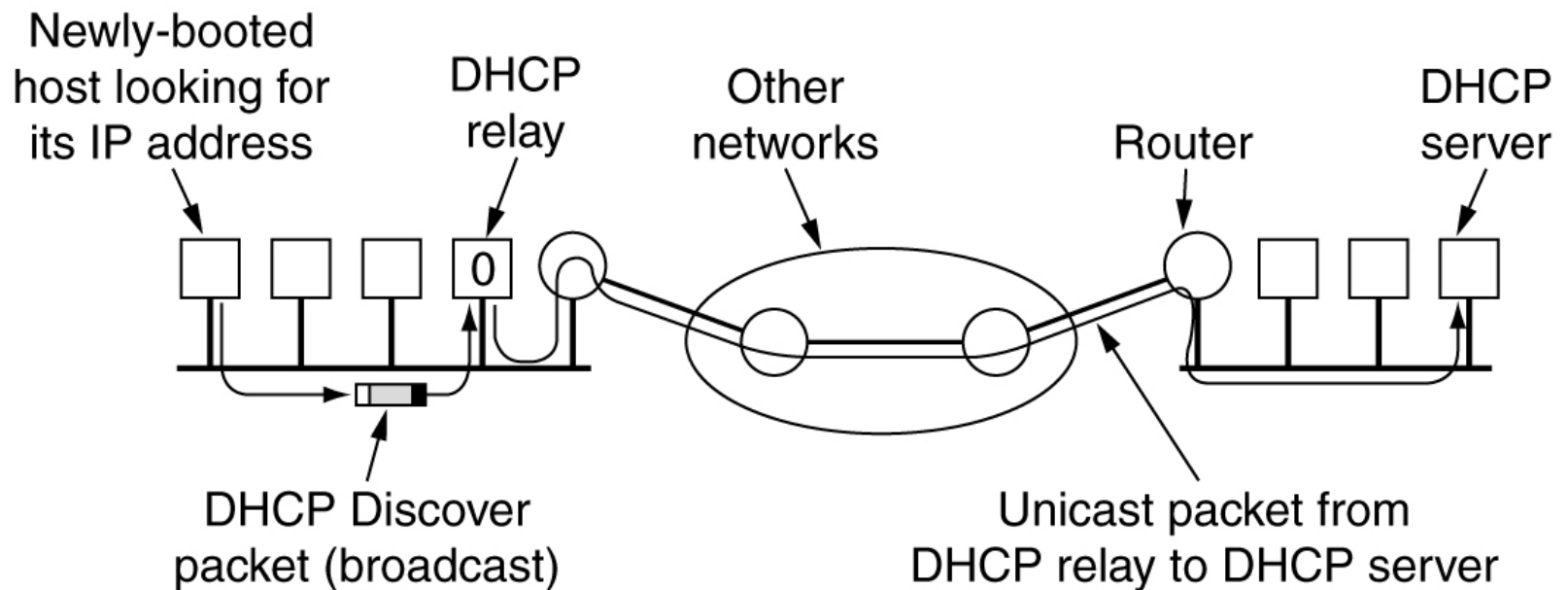
[ICMP]

- O transmissor envia uma série de segmentos UDP para o destino
 - O 1º possui TTL = 1
 - O 2º possui TTL = 2 etc.
 - nº de porta improvável
- Quando o enésimo datagrama chega ao enésimo roteador:
 - O roteador descarta o datagrama
 - E envia à origem uma mensagem ICMP (type 11, code 0)
 - A mensagem inclui o nome do roteador e o endereço IP
- Quando a mensagem ICMP chega, a origem calcula o RTT
- O traceroute faz isso três vezes
- **Critério de interrupção**
- O segmento UDP finalmente chega ao hospedeiro de destino
- O destino retorna o pacote ICMP “hospedeiro unreachable” (type 3, code 3)
- Quando a origem obtém esse ICMP, ela pára.

ARP– The Address Resolution Protocol

- Permite converter endereços IP para endereços físicos
- Máquina pergunta para todos “A quem pertence o endereço xx?”
- Apenas xx responde com o seu endereço físico.
- Informação é guardada em cache com um determinado tempo de vida
 - arp -a

DHCP - Dynamic Host Configuration Protocol



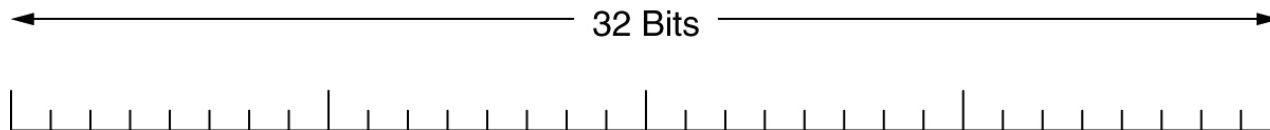
Protocolos de Roteamento na Internet

- OSPF (Open Shortest Path First)
 - Usado para roteamento intra-SA (Sistema Autônomo)
 - Algoritmo de estado de enlace
- BGP (Border Gateway Protocol)
 - Usado para roteamento inter-AS
 - Políticas externas devem ser consideradas e programadas nos roteadores

[IPv6]

- Motivação inicial
 - O espaço de endereços de 32 bits está próximo de ser completamente alocado
- Motivação adicional:
 - Melhorar o formato do cabeçalho para permitir maior velocidade de processamento e de transmissão
 - Mudanças no cabeçalho para incorporar mecanismos de controle de QOS
- Formato do datagrama IPV6
 - Não é permitida fragmentação

[IPv6]



Version	Traffic class	Flow label	
Payload length		Next header	Hop limit
Source address (16 bytes)			
Destination address (16 bytes)			

[IPv6]

■ Cabeçalho IPv6

- Priority: permitir definir prioridades diferenciadas para vários fluxos de informação
- Flow label: identifica datagramas do mesmo “fluxo.” (conceito de “fluxo” não é bem definido).
- Next header: identifica o protocolo da camada superior ou um header auxiliar
- Checksum: removido inteiramente para reduzir o tempo de processamento em cada salto
- Options: são permitidas, mas são alocadas em cabeçalhos suplementares, indicados pelo campo “Next header”

[IPv6]

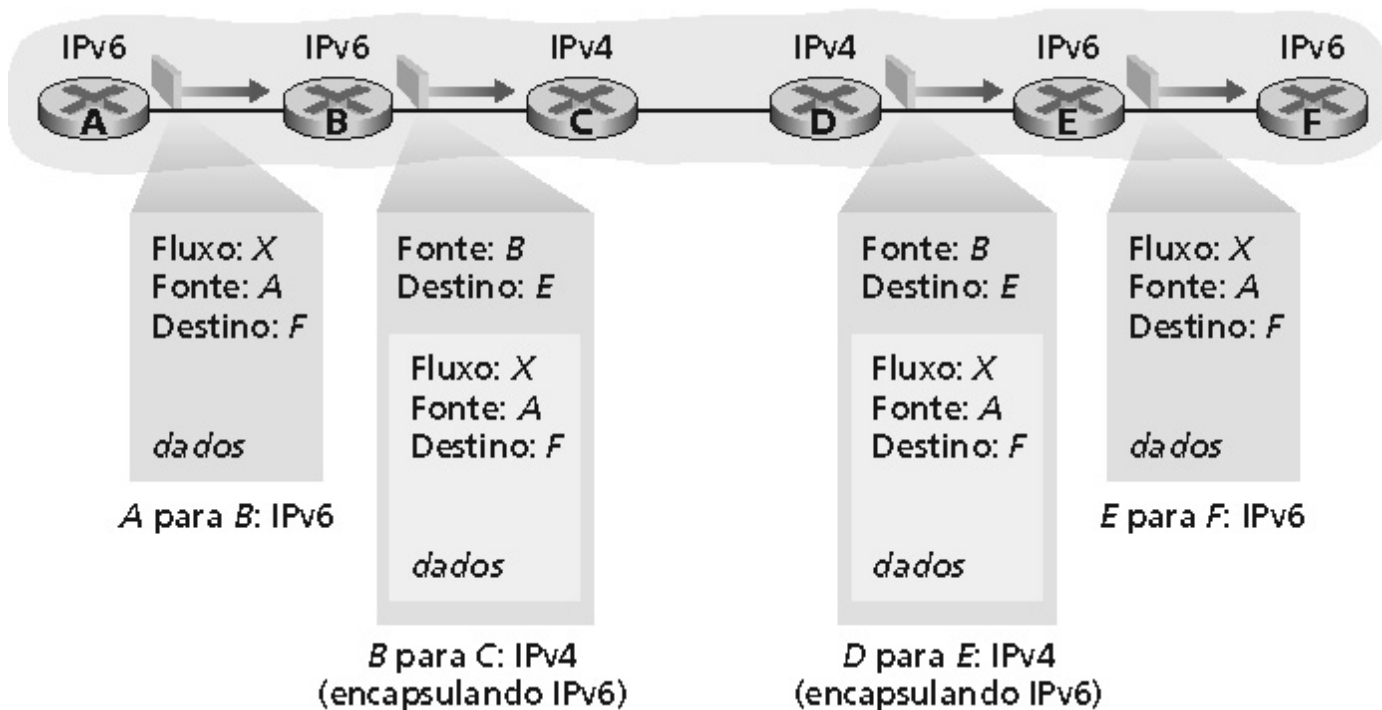
- Nem todos os roteadores poderão ser atualizados simultaneamente
 - Não haverá um dia da vacinação
 - Como a rede irá operar com roteadores mistos de IPV4 e IPV6?
- Tunelamento
 - IPv6 transportado dentro de pacotes IPV4 entre roteadores IPV4

IPv6

Visão lógica



Visão física



[IPv6]

- Endereçamento IPv6
 - 128 bits (16 bytes) fornece uma fonte de endereços inesgotável
 - Os endereços são exibidos em hexadecimal
 - 8000:0000:0000:0000:0123:4567:89AB:CDEF
 - 8000::123:4567:89AB:CDEF
 - ::192.31.20.46 (IPv4)