

Roteamento IP

Parte 4

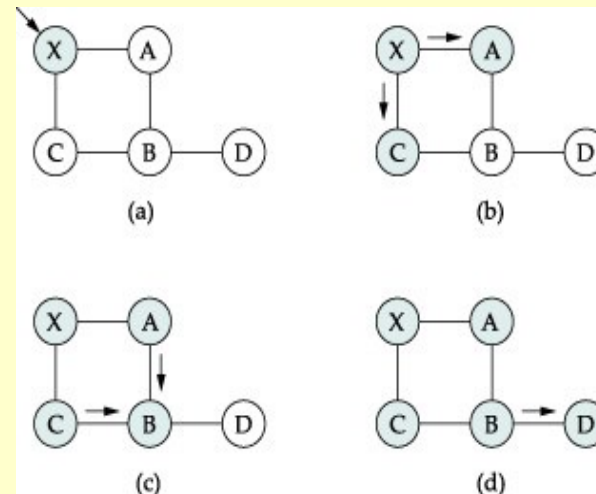
Prof. Dr. S. Motoyama

Estado de enlace

- Estratégia
 - envia para todos os nós (não apenas vizinhos) informações sobre enlaces conectados diretamente (não a tabela de roteamento inteira)
- Link State Packet (LSP)
 - id do nó que criou o LSP
 - custo do enlace para cada vizinho conectado diretamente
 - número de seqüência (SEQNO)
 - tempo de vida (TTL) para este pacote

Estado do enlace (cont.)

- Inundação confiável
 - armazena LSP mais recente de cada nó
 - encaminha LSP a todos os nós, menos aquele que enviou
 - gera novo LSP periodicamente
 - incrementa SEQNO
 - inicia SEQNO em 0 quando reinicializar
 - decrementa TTL de cada LSP armazenado
 - descarta quando TTL=0
- O cálculo do caminho mais curto é feito através do algoritmo de Dijkstra.



Algoritmo de caminho mais curto de Dijkstra

1º Passo: (Neste passo procura-se o nó cuja a métrica, por ex., a distância, seja a menor entre os nós vizinhos)

Encontre $i \notin N$ (conjunto de nós) tal que

$$D_i = \min_{j \notin N} D_j$$

Faça $N = N \cup \{i\}$. Se N contém todos os nós, então pare. O algoritmo está completo.

2º Passo: (Neste passo é feita a atualização das métricas para cada nó vizinho)

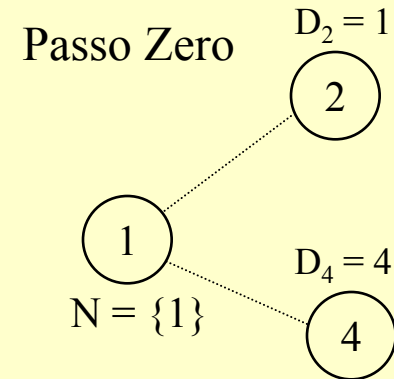
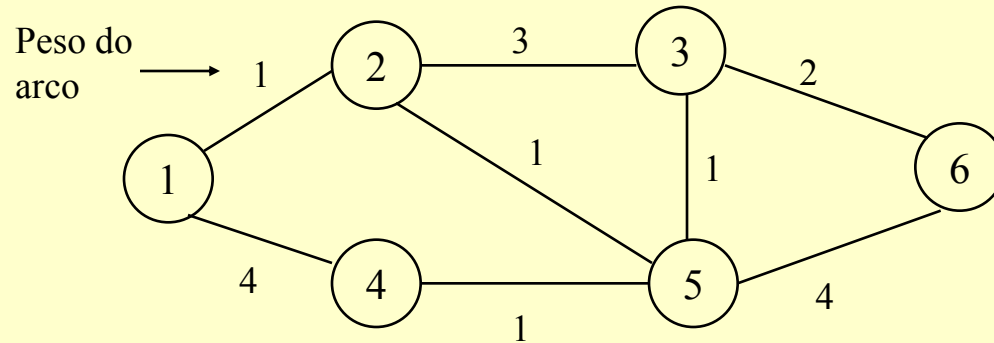
Para todo $j \notin N$, calcule

$$D_j = \min[D_j, d_{ji} + D_i]$$

Volte ao passo 1.

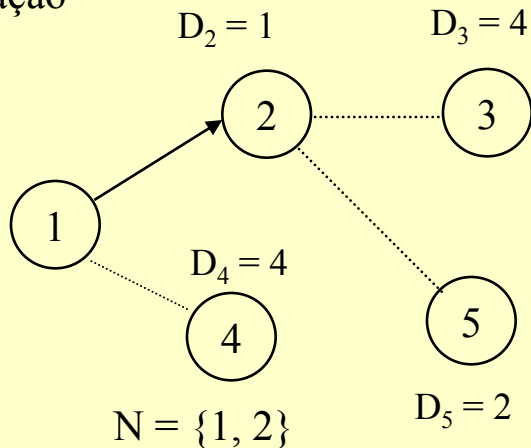
Algoritmo de Dijkstra - Exemplo

Seja a rede mostrada abaixo

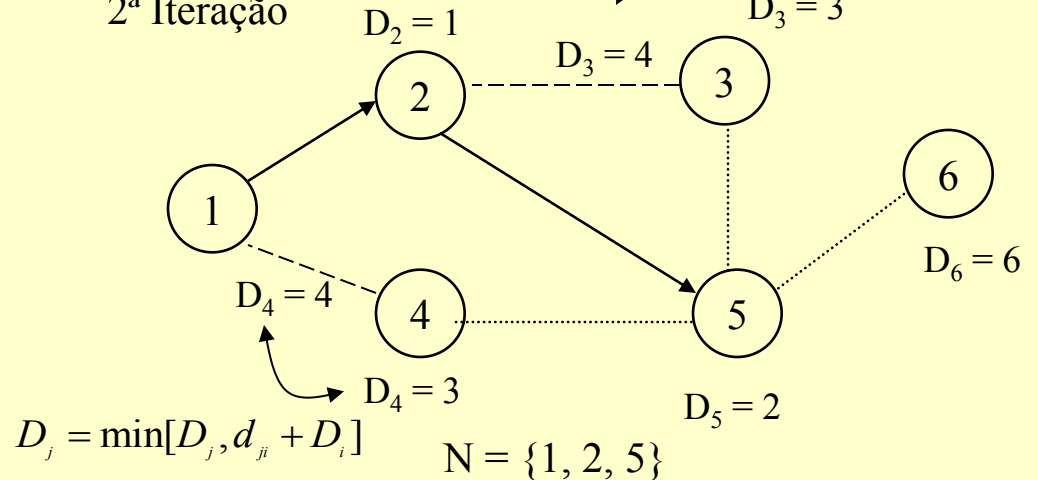


$$D_j = \min[D_j, d_{ji} + D_i]$$

1ª Iteração

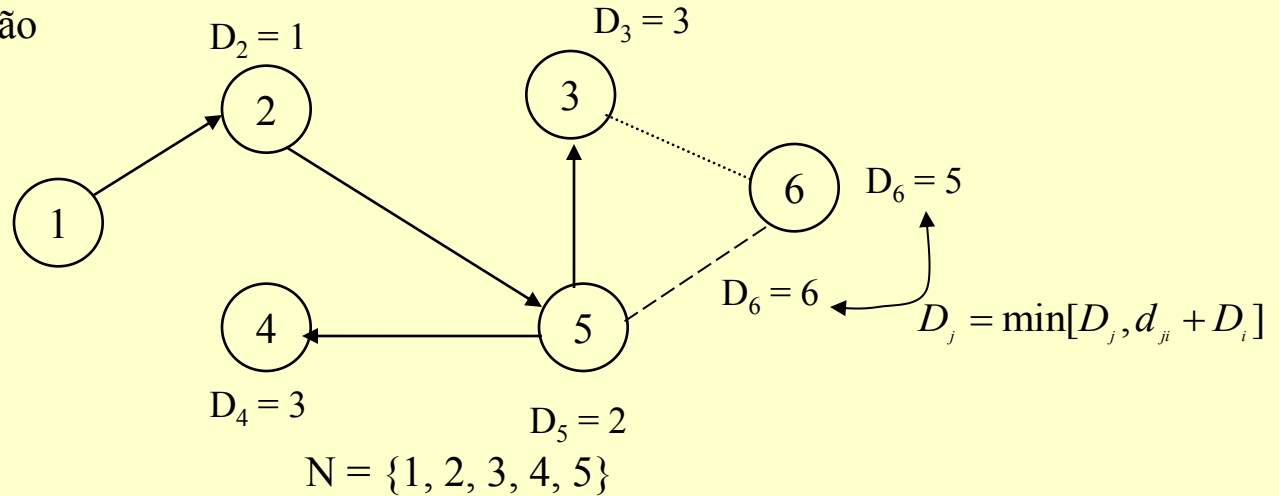


2ª Iteração

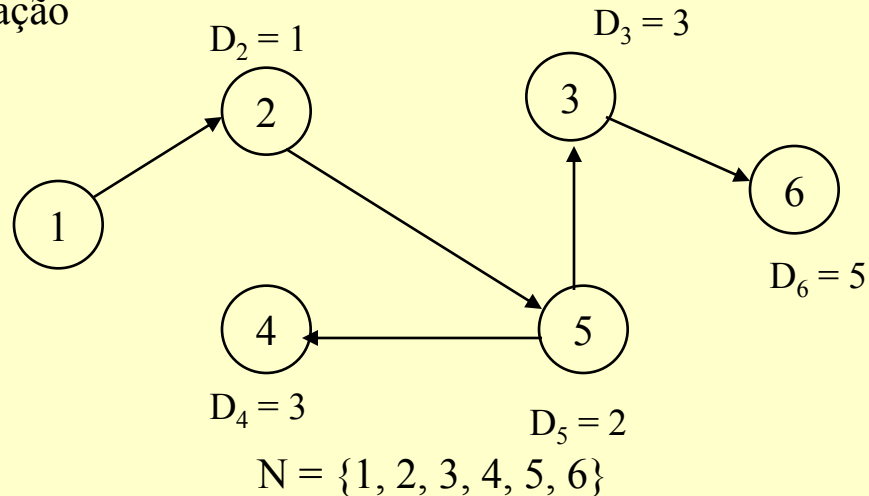


Algoritmo de Dijkstra - Exemplo

3ª Iteração

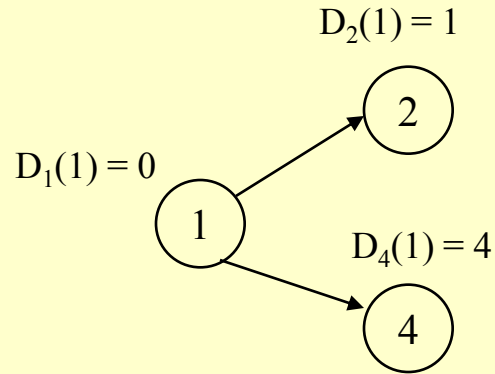


4ª Iteração

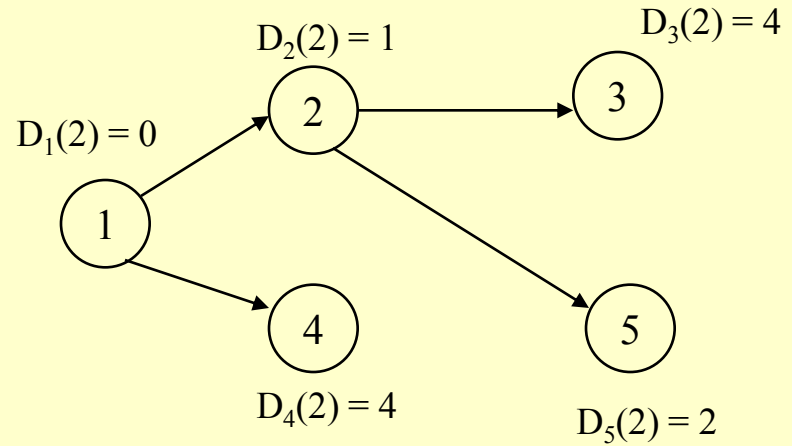


Por Bellman-Ford

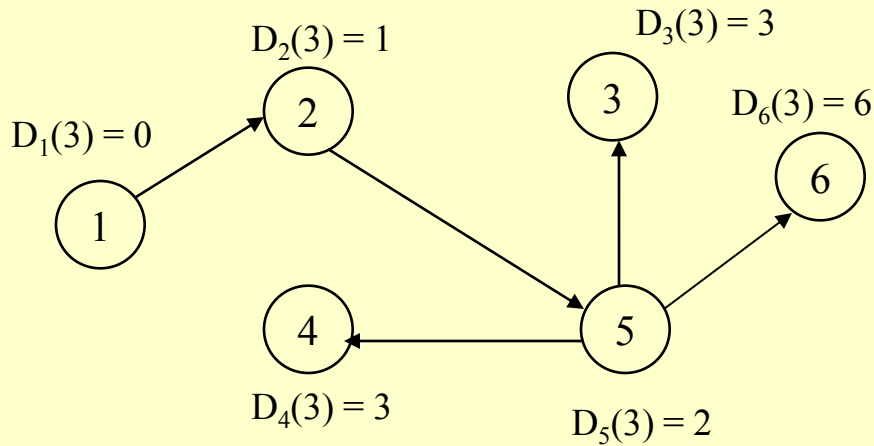
1 arco



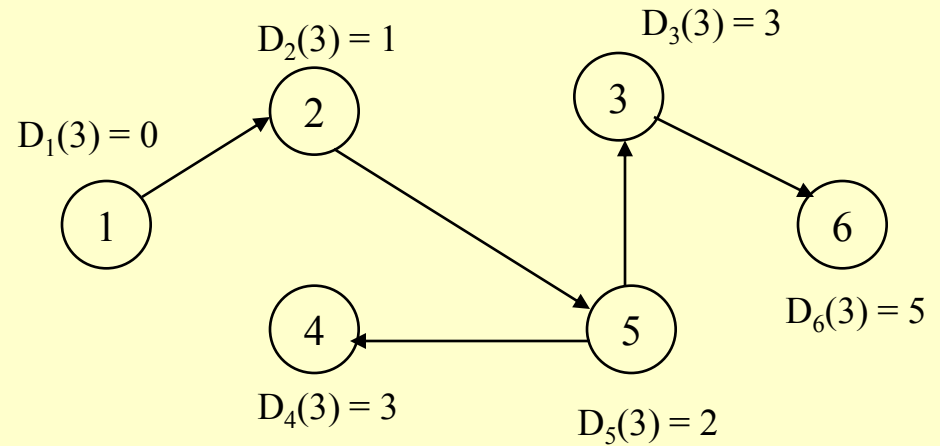
2 arcos



3 arcos

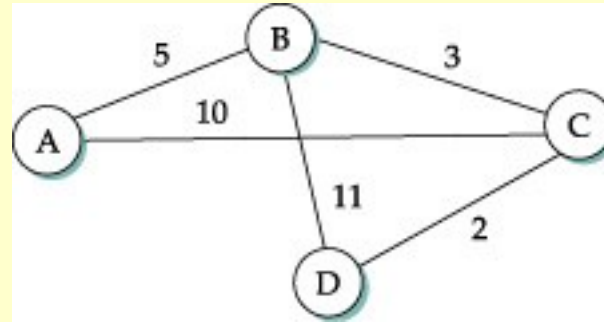


4 arcos



Implementação do algoritmo: Exemplo

Etapas para a montagem da tabela para o nó D



Etapa	Confirmada	Tentativa	Comentários
1	(D,0,-)		Como D é o único membro, examine seu LSP.
2	(D,0,-)	(B,11,B) (C,2,C)	LSP de D diz: alcança B com custo 11 e C com custo 2.
3	(D,0,-) (C,2,C)	(B,11,B)	Coloca membro de menor custo da Tentativa (C) na lista Confirmada e examina o LSP de C.
4	(D,0,-) (C,2,C)	(B,5,C) (A,12,C)	Custo para alcançar B através de C é 5, substitui (B,11,B). LSP de C diz: alcança A com custo 12
5	(D,0,-); (C,2,C) (B,5,C)	(A,12,C)	Move membro de menor custo de Tentativa (B) para Confirmada, depois examina seu LSP.
6	(D,0,-); (C,2,C) (B,5,C)	(A,10,C)	Como pode-se alcançar A com custo 5 através de B, substitui a entrada de Tentativa.
7	(D,0,-); (C,2,C) (B,5,C); (A,10,C)		Move membro de menor custo de Tentativa (A) para Confirmada e fim.

Métrica

- Métrica ARPANET original
 - mede número de pacotes enfileirados em cada enlace
 - não leva em conta latência ou largura de banda
- Nova métrica ARPANET
 - marca cada pacote que chega com sua hora de chegada (**AT**)
 - registra hora de saída (**DT**)
 - quando chega ACK em nível de enlace, calcula
$$\mathbf{Retardo} = (\mathbf{DT} - \mathbf{AT}) + \mathbf{Transmissão} + \mathbf{Latência}$$
 - se timeout, reinicia **DT** para hora de saída, para retransmissão
 - custo do enlace = retardo médio por algum período de tempo
- Ajuste fino
 - intervalo dinâmico compactado (leva em conta tipo e capac. do enlace)
 - substitui **Retardo** por utilização do enlace

Protocolo OSPF (Open shortest path first)

- OSPF implementa a abordagem de estado de enlace.
- OSPF suporta diferentes tipos de métricas e consequentemente pode suportar diferentes tipos de roteamento.
- OSPF admite balanceamento de carga.
- OSPF menos quantidade de mensagens de atualização quando comparado com o RIP.
- OSPF proporciona convergência rápida e sem loop.

Roteamento Interdomínio

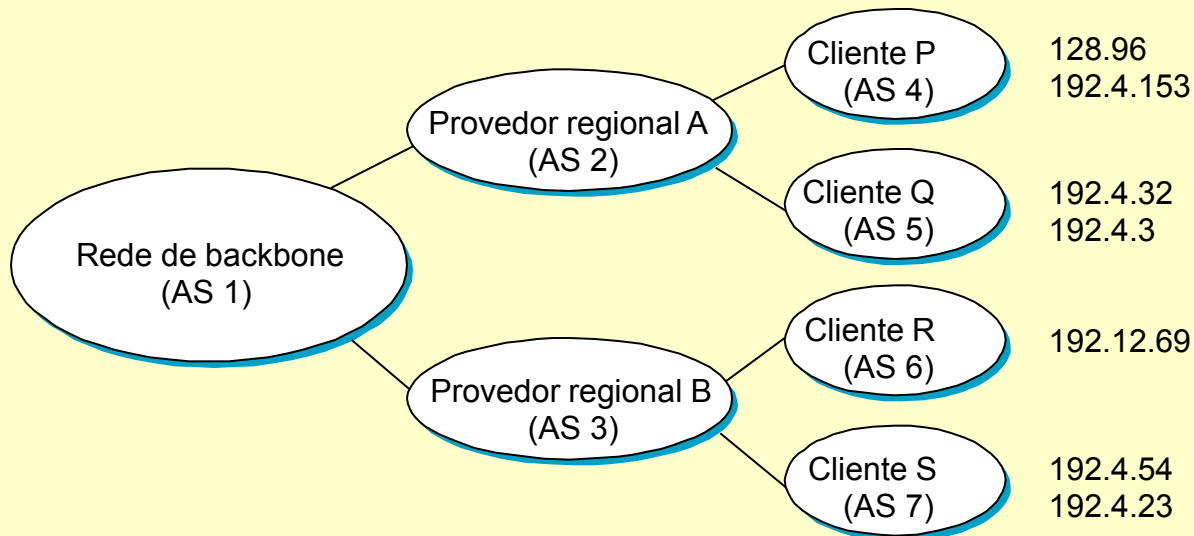
- Roteamento interdomínio é importante especialmente para os roteadores de backbone.
- Enquanto o roteamento intradomínio é baseado em quantidades mensuráveis, o roteamento interdomínio é baseado em confiança e em política.
 - A métrica usada em um domínio pode não ser significativa em outro domínio.
- Portanto, o objetivo de roteamento interdomínio é alcance e não otimalidade.
 - Um protocolo interdomínio deve ser capaz de encontrar um caminho sem loops para alcançar um outro AS.

BGP-4: Border Gateway Protocol

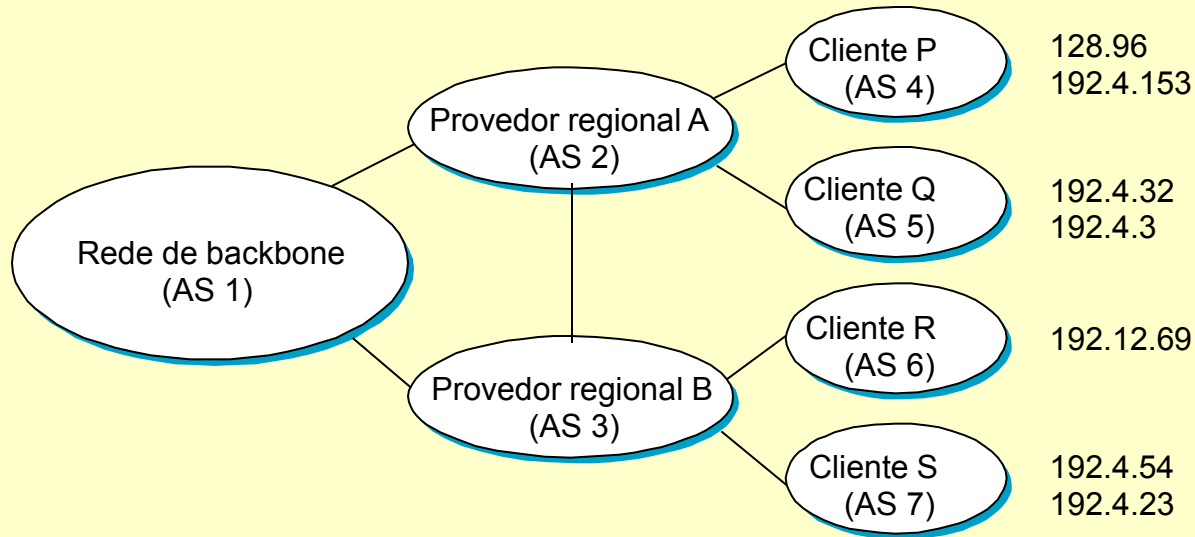
- Tipos de AS
 - AS stub: tem uma única conexão com um outro AS
 - transporta apenas tráfego local
 - AS multihomed: tem conexões com mais de um AS
 - recusa-se a transportar tráfego de trânsito
 - AS de trânsito: tem conexões com mais de um AS
 - transporta tráfego de trânsito e local
- Cada AS tem:
 - um ou mais roteadores de borda
 - um *speaker* BGP que anuncia:
 - redes locais
 - outras redes alcançáveis (somente AS de trânsito)
 - oferece informações de *caminho*

Exemplo de BGP

- Speakers de AS4-5 anunciam informações de alcance dos seus provedores regionais.
- Speaker de AS2 anuncia alcance para P e Q
 - redes 128.96, 192.4.153, 192.4.32, e 192.4.3, podem ser alcançadas diretamente de AS2
- Speaker de AS1 anuncia alcance para AS2
 - redes 128.96, 192.4.153, 192.4.32, e 192.4.3 podem ser alcançadas ao longo do caminho (AS1, AS2).
- Speaker pode cancelar caminhos previamente anunciados.



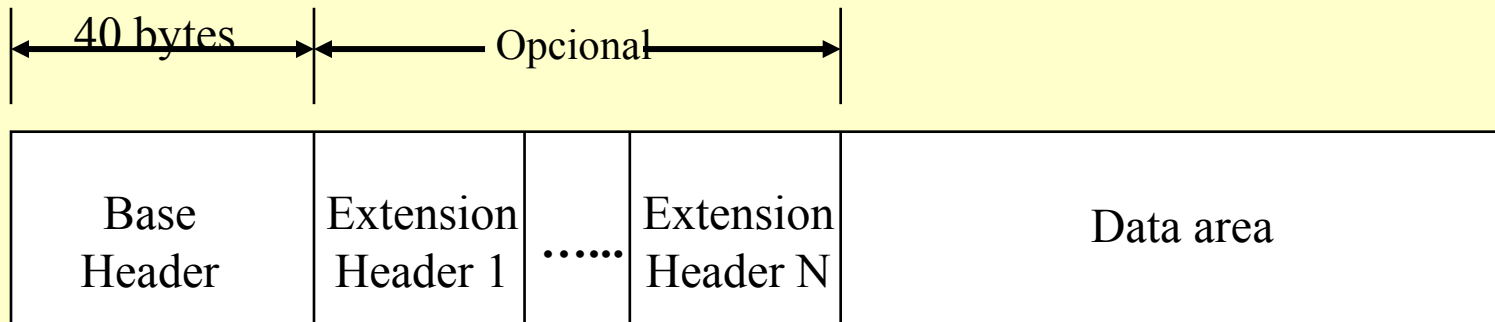
Outro Exemplo



- AS3 recebe agora dois conjuntos de informações de alcance para 128.96.0.0, 192.4.153.0:
 - Um de AS1 com (AS1, AS2, AS4), e
 - Outro de AS2 com (AS2, AS4).
 - AS3 pode decidir baseado no número mínimo de AS que atravessará.

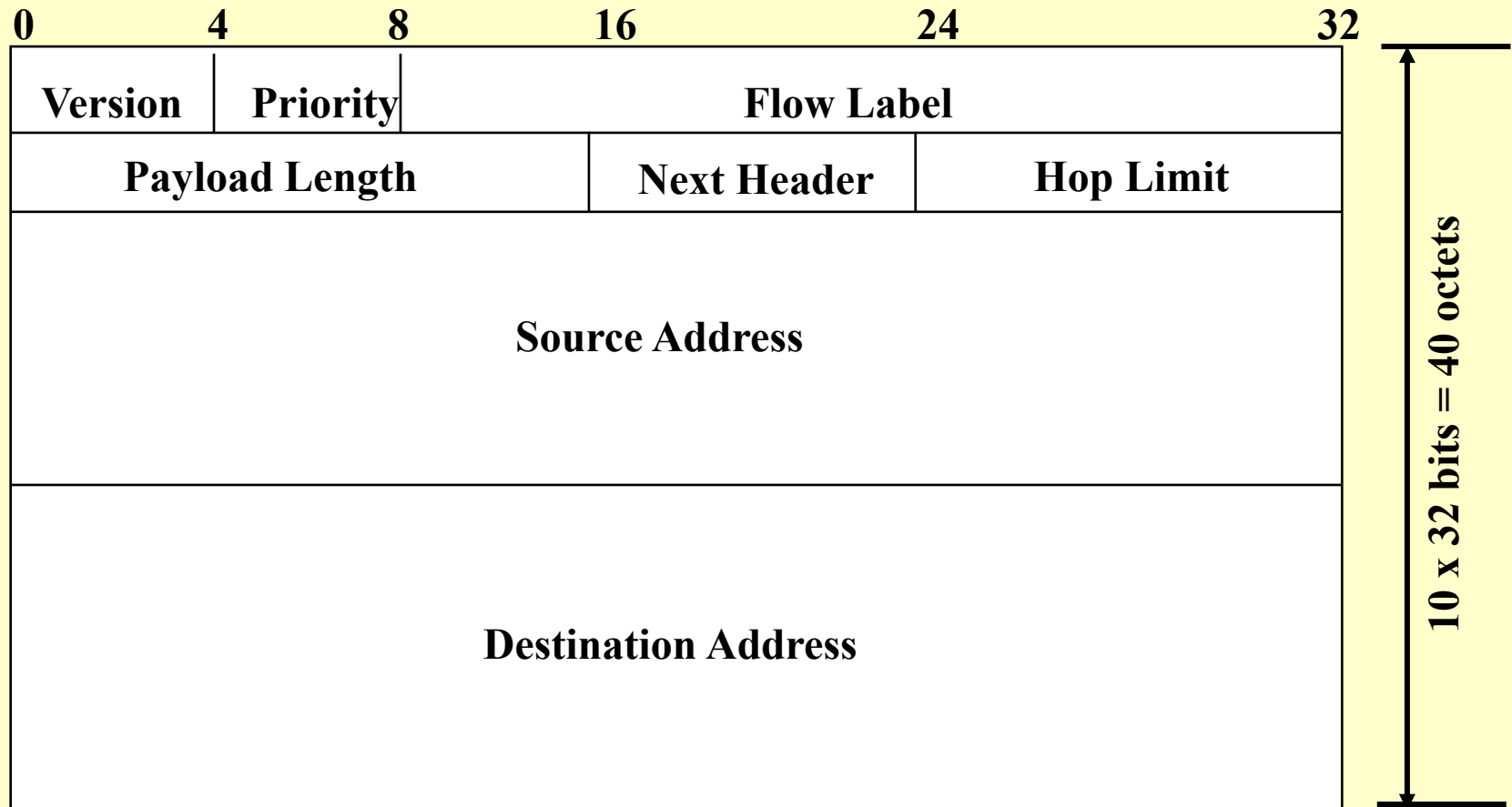
IPv6

- Motivação inicial: o espaço de endereços de 32 bits está próximo de ser completamente alocado.
- Motivação adicional:
 - Melhorar o formato do header para permitir maior velocidade de processamento e de transmissão
 - Mudanças no header para incorporar mecanismos de controle de QOS
 - Formato do datagrama IPV6: Cabeçalho fixo de 40 bytes



Cabeçalho de base Cabeçalhos de Extensão

IPv6 - Cabeçalho de base



IPv6 - Cabeçalho de base

Versão (Version) - 4 bits - Número de versão IP; o valor é 6

Prioridade (Priority) - 4 bits - Valor de prioridade de cada pacote. Especifica a classe de tráfego. Valores entre 0 e 7 são definidos para tráfegos controlados no caso de congestionamento (dados) e os valores entre 8 e 15 para tráfegos não controlados no caso de congestionamento (vídeo e áudio).

Rótulo de fluxo (Flow Label) - 24 bits - Utilizado para aplicações que necessitam garantia de desempenho. Um fluxo é definido como um seqüência de pacotes enviados de um fonte particular até um destino particular. Um fluxo é identificado pela combinação do endereço da fonte e um rótulo de fluxo de 24 bits. Desse modo, todos os pacotes que pertencem a um mesmo fluxo, possuem um único rótulo.

Comprimento de carga útil (payload length) - 16 bits - Especifica o tamanho dos dados transportados.

IPv6 - Cabeçalho base

Próximo Cabeçalho (Next header) - 8 bits - Identifica o tipo de cabeçalho que se segue imediatamente após o cabeçalho base. Por exemplo, um cabeçalho TCP/UDP ou um cabeçalho opcional do IPv6.

Limite de Saltos (Hop Limit) - 8 bits - Número de saltos (hops) restante para um particular pacote. Esse número é colocado pela fonte e decrementado por 1 em cada nó. Se esse número chega a zero, o pacote em questão é descartado.

Endereço da Fonte (Source Address) - 128 bits - Endereço de quem está enviando o pacote.

Endereço do Destino (Destination Address) - 128 bits - Endereço de quem está recebendo o pacote.

IPv6 - Cabeçalho de extensão

Há 7 tipos diferentes de cabeçalhos de extensão:

Cabeçalho de opções de salto-a-salto (Hop-by-hop Options Header) - Define opções especiais que são necessárias no processamento hop-by-hop.

Cabeçalho de opção de destino - 1 (Destination Option Header - 1) - Contém informação opcional a ser verificado pelo 1º destinatário listado no campo de endereço.

Cabeçalho de roteamento (Routing Header) - Permite a um nó de origem especificar uma lista de endereços IPs que norteará o caminho que o pacote deve seguir.

Cabeçalho de fragmento (Fragment Header) - Contém informações de fragmentação e remontagem.

IPv6 - Cabeçalho de extensão

Cabeçalho de Autenticação (Authentication Header) - Proporciona a integridade e autenticação (detecção de erro).

Cabeçalho de carga útil encapsulada de segurança (Encapsulated Security Payload Header) - Proporciona privacidade.

Cabeçalho de Opções de destinatário - 2 - Destination Options Header - 2
Contém informações opcionais a ser examinadas apenas pelo nó destino final.

Alocação de Espaço de Endereços no IPv6

O primeiro byte do campo de endereços de 128 bits é prefixo que indica como o campo será utilizado.

Prefixo	Utilização
0000 0000	Reservado
0000 0001	Não atribuído
0000 001	Reservado p/ NNSAP
0000 010	Reservado p/ IPX
0000 011	Não atribuído
0000 1	Não atribuído
0001	Não atribuído
001	Endereço Global Agregavel
010 a 1111 1110 0	Não atribuído

Prefixo	Utilização
1111 1110 10	Endereços de uso local do enlace
1111 1110 11	Endereços de uso local do site
1111 1111	Endereços de multicast

← Endereços unicasts utilizados na Internet.

Alocação de Espaço de Endereços no IPv6

Notação de endereços

É diferente da representação decimal pontuada. A representação é da forma:

X:X:X:X:X:X:X

Onde X é uma representação hexadecimal de 16 bits.

Ex.: 47CD:1234:4422:AC02:0022:1234:A456:0124

Notação compacta.

A notação 47CD:0000:0000:0000:0000:0000:0000:0124
pode ser escrita como 47CD::0124

Notação para IPv4

::196.46.20.10

A alocação de espaço de endereços é similar ao utilizado em roteamento interdomínio sem classes (CIDR).

Alocação de Espaço de Endereços no IPv6

Como operar no período de transição. Duas soluções:

a) Operação a duas pilhas

Um nó roda tanto IPv6 como IPv4. Utiliza o campo Version para decidir qual pilha deve utilizar.

b) Tunelamento

A técnica de tunelamento é utilizado para enviar os pacotes IPv6 em redes que só entendem IPv4.

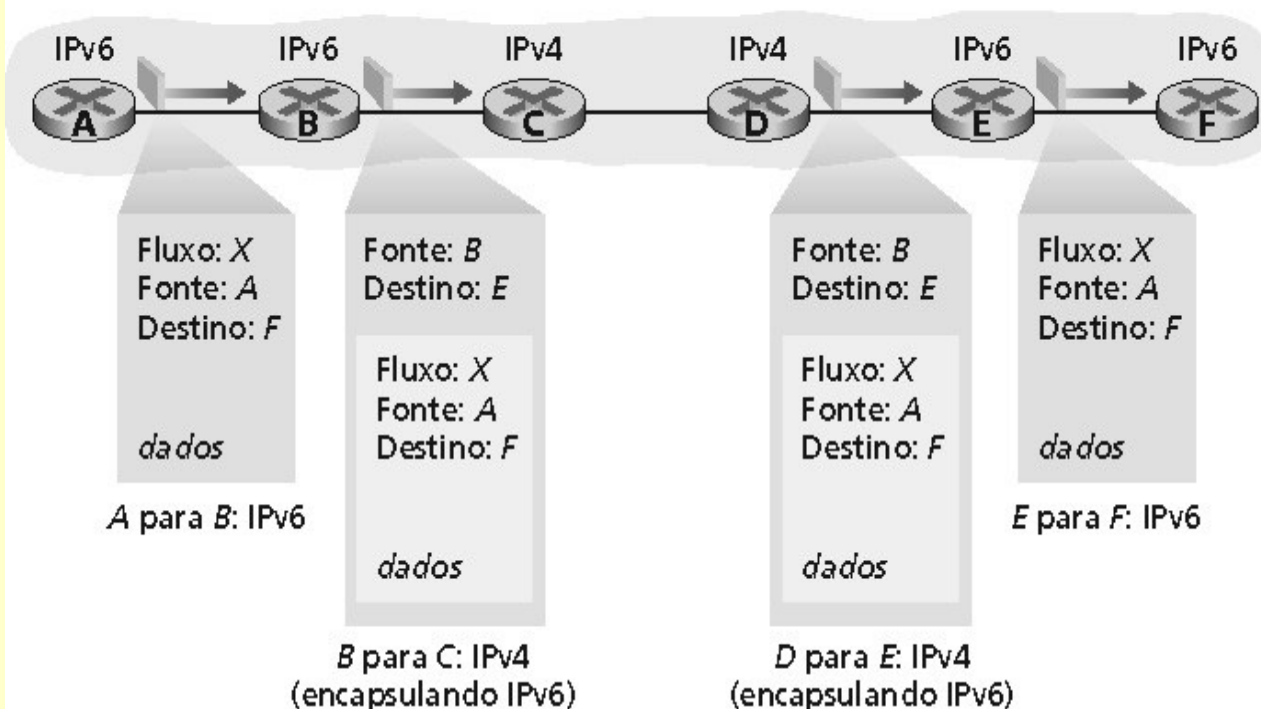
A técnica consiste em encapsular o pacote IPv6 dentro de um pacote com cabeçalho do IPv4.

Tunelamento

Visão lógica



Visão física



Exercício

14.1 Para a rede mostrada na figura abaixo, encontre o caminho de custo mínimo de cada roteador para atingir o roteador 1, utilizando o algoritmo de Dijkstra.

