

Redes de computadores II

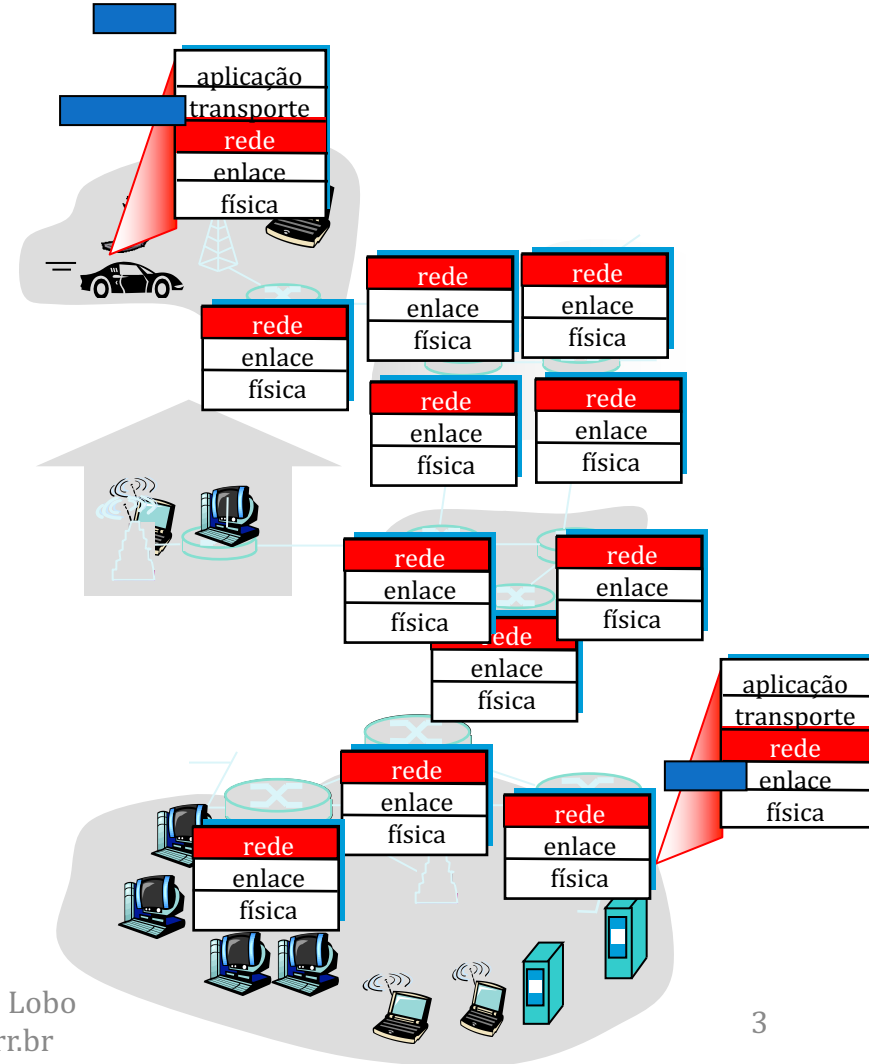
Aula 01 – Camada de Rede

Camada de Rede

- A camada de rede implementa o serviço de comunicação *host a host*;
- Existe um pedaço da camada de rede em cada um dos *hosts* e roteadores na rede;
- Protocolos estão entre os mais desafiadores;

Camada de rede

- roteador examina campos de cabeçalho em todos os datagramas IP que passam por ele;



Camada de Rede

- A camada de rede é uma das mais complexas:
 - Visão geral e serviços;
 - Estruturação da entrega do pacote;
 - Modelo de datagrama e circuitos virtuais;
 - O papel do endereçamento;
 - Distinção entre repasse e roteamento;

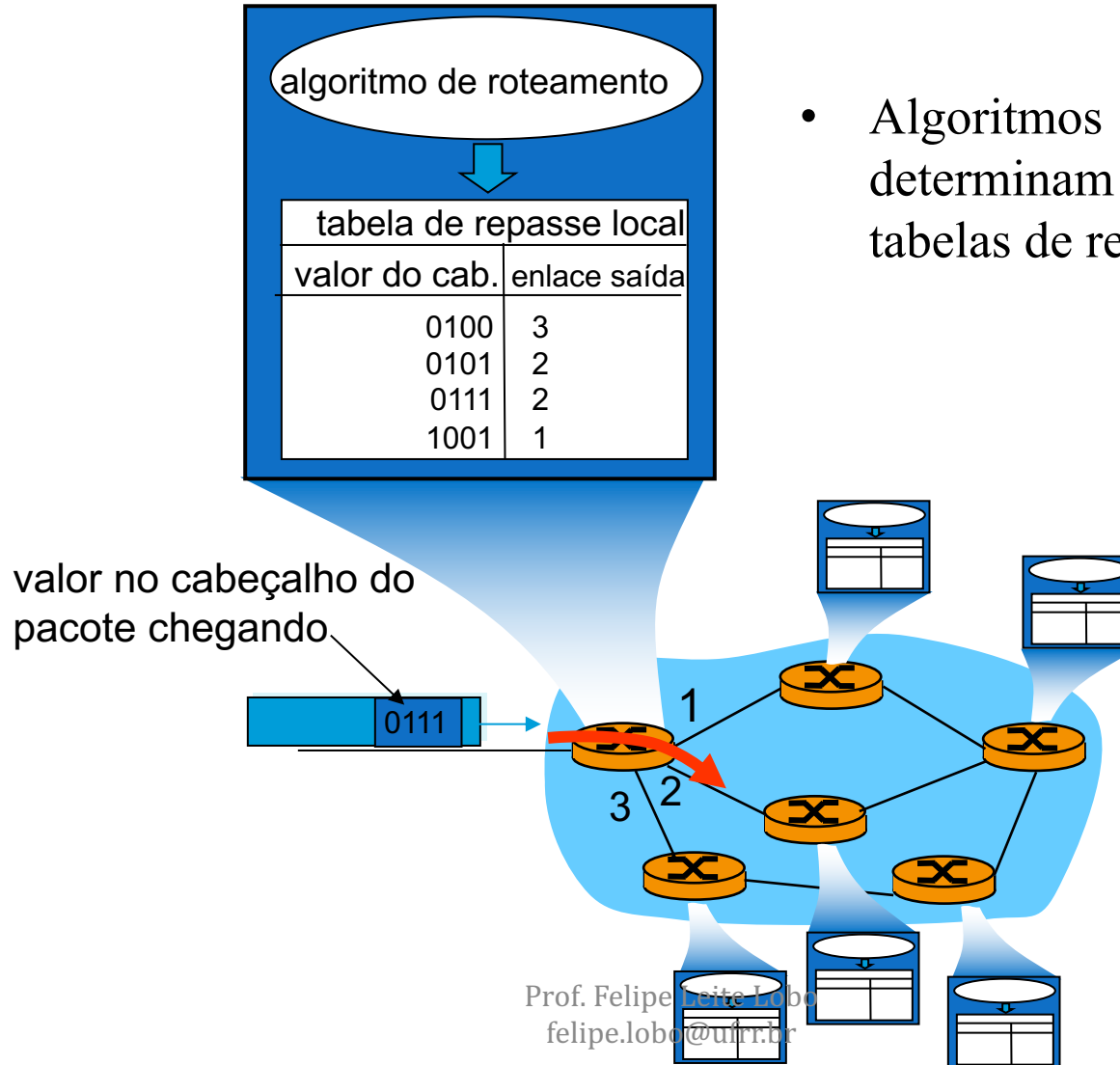
Camada de Rede

- O papel da camada de rede é aparentemente simples:
 - Transportar pacotes de um *host* remetente a um destino;
 - É feito através de duas funções importantes:
 - Repasse: encaminha um pacote do enlace de entrada até o enlace de saída;
 - Roteamento: Determina rota ou caminho tomados pelos pacotes desde a origem até o destino;

Camada de Rede

- Cada roteador possui uma tabela de repasse;
- Examina o valor de um campo no cabeçalho do pacote que está chegando e o utiliza para indexar sua tabela;
- O resultado da tabela de repasse indica qual das interfaces de enlace do roteador o pacote deve ser repassado;
- O responsável pelos valores inseridos nas tabelas de repasse dos roteadores são os algoritmos de roteamento;

Camada de Rede: **Interação entre roteamento e repasse**



- Algoritmos de roteamento determinam valores em tabelas de repasse:

Camada de Rede

- O algoritmo de roteamento pode ser:
 - Centralizado: roda em um local central e descarrega informações de roteamento a cada um dos roteadores;
 - Descentralizado: algoritmo de roteamento distribuído funcionando em cada roteador;
 - Ambos os casos o roteador recebe mensagens do protocolo de roteamento, utilizadas para configurar sua tabela de repasse;

Camada de Rede

- Em algumas redes existe uma terceira função importante: Estabelecimento de conexão:
- Algumas arquiteturas de camada de rede (ATM, *frame-relay*), mas não a internet:
 - Exigem o estabelecimento de conexão entre roteadores ;
 - Antes dos pacotes de dados de camada de rede possam começar a fluir;

Camada de Rede

- **O modelo de serviço de rede** define as características do transporte de dados fim a fim entre uma borda da rede e a outra:
 - Serviços possíveis que a camada de rede poderia prover para datagramas individuais:
 - Entrega garantida: Assegura que o pacote mais cedo ou mais tarde irá chegar ao seu destino;
 - Entrega garantida com atraso limitado: Assegura a entrega do pacote, porém com o atraso especificado;

Camada de Rede

- Serviços que a camada de rede pode prover para um fluxo de datagramas:
 - Entrega de pacotes na ordem: Assegura a entrega dos pacotes na ordem em que foram enviados;
 - Largura de banda mínima garantida: Nenhum pacote será perdido e cada um chegará dentro de um atraso previamente especificado;
 - Jitter máximo garantido: tempo entre a transmissão de dois pacotes sucessivos no remetente seja igual no recebimento;

Camada de Rede

Arquitetura da rede	Modelo de serviço	Garantia de largura de banda	Garantia contra perda	Ordenamento	Temporização	Indicação de congestionamento
Internet	Melhor esforço	Nenhuma	Nenhuma	Qualquer ordem possível	Não mantida	Nenhuma
ATM	CBR	Taxa constante garantida	Sim	Na ordem	Mantida	Não ocorrerá congestionamento
ATM	ABR	Mínima garantida	Nenhuma	Na ordem	Não mantida	Indicação de congestionamento

Camada de Rede

- Redes de circuitos virtuais e de datagramas:
- Rede de datagrama fornece serviço sem conexão da camada de rede;
- Rede de Circuitos Virtuais fornece serviço com conexão da camada de rede;
- Análogo aos serviços da camada de transporte, mas:
 - **serviço:** hospedeiro a hospedeiro;
 - **sem escolha:** a rede oferece um ou outro;
 - **implementação:** no núcleo da rede (roteadores);

Circuitos virtuais

“Caminho da origem ao destino comporta-se como um circuito telefônico”

- com respeito ao desempenho
 - ações da rede ao longo do caminho da origem ao destino
-
- estabelecimento e término para cada chamada *antes* que os dados possam fluir;
 - cada pacote carrega identificador CV (não endereço do hospedeiro de destino);
 - cada roteador no caminho origem-destino mantém “estado” para cada conexão que estiver passando;
 - recursos do enlace e roteador (largura de banda, *buffers*) podem ser alocados ao CV;

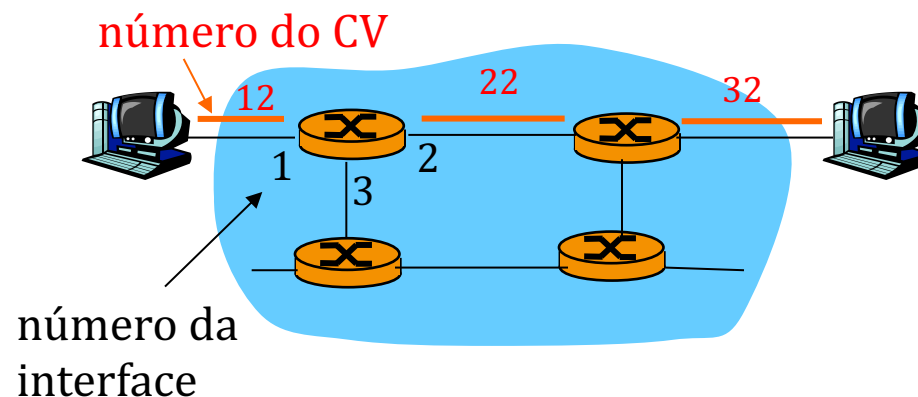
Implementação do Circuito Virtual

Um CV consiste em:

1. caminho da origem ao destino;
 2. números de CV, um número para cada enlace ao longo do caminho;
 3. entradas em tabelas de repasse nos roteadores ao longo do caminho;
- pacote pertencente ao CV carrega número do CV (em vez do endereço de destino);
 - número do CV pode ser alterado em cada enlace:
 - novo número de CV vem da tabela de repasse;

Tabela de repasse

tabela de repasse no roteador noroeste:

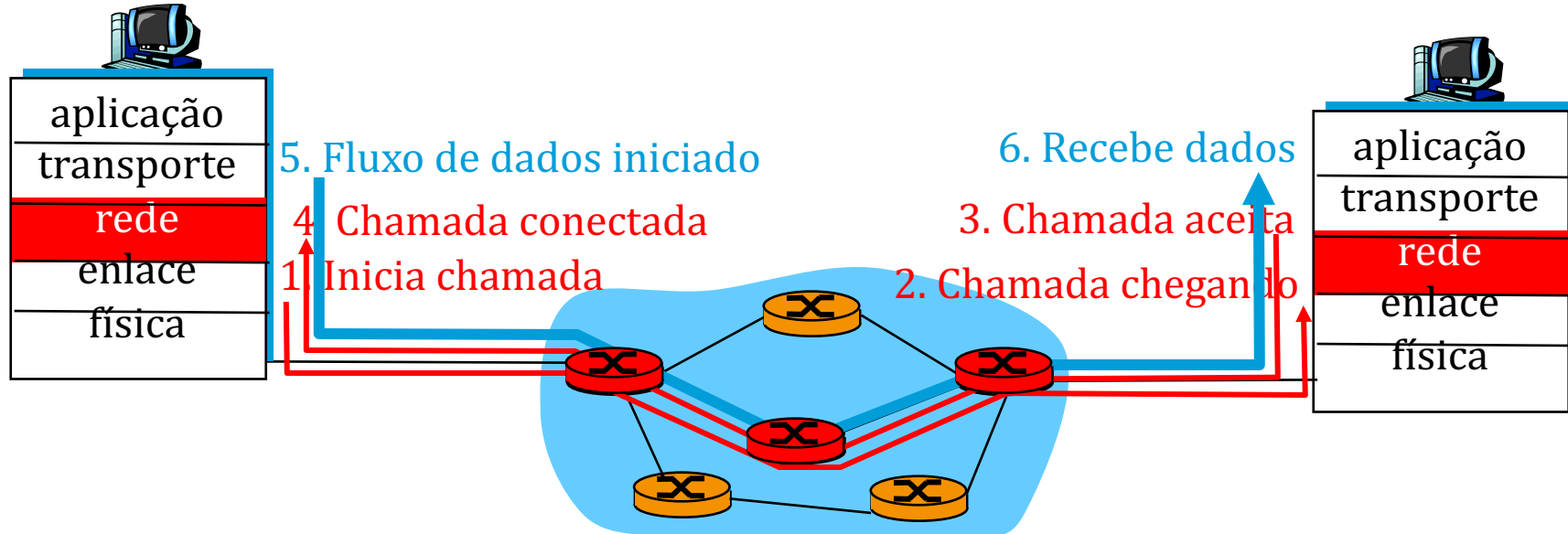


Interface de entrada	Nº do CV de entrada	Interface de saída	Nº do CV de saída
1	12	2	22
2	63	1	18
3	7	2	17
1	97	3	87
...

Roteadores mantêm informação de estado da conexão!

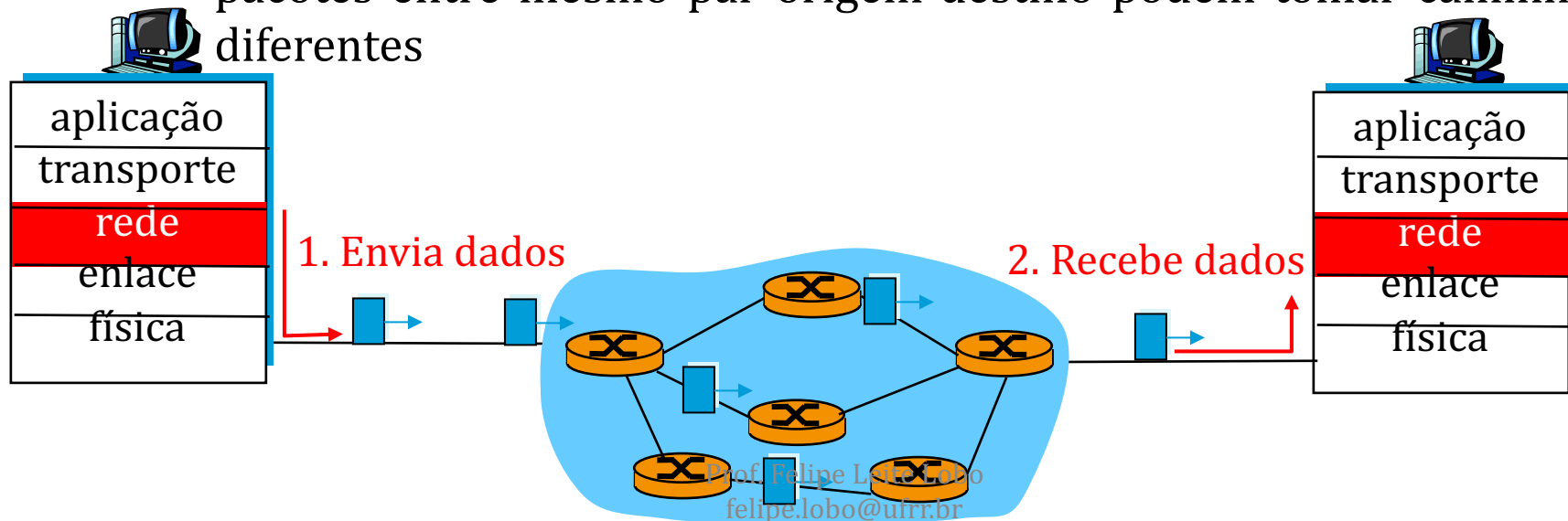
Circuitos virtuais: protocolos de sinalização

- usados para estabelecer, manter e terminar CV;
- usados em ATM, frame-relay, X.25;
- não usados na Internet de hoje;



Redes de datagrama

- sem estabelecimento de chamada na camada de rede;
- roteadores: sem estado sobre conexões fim a fim;
 - sem conceito em nível de rede da “conexão”
- pacotes repassados usando endereço do hospedeiro de destino
 - pacotes entre mesmo par origem-destino podem tomar caminhos diferentes



Redes de datagrama: Tabela de repasse

Mais de 4 bilhões de
entradas possíveis

Faixa de endereços de destino	Interface de enlace
11001000 00010111 00010000 00000000 até 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 até 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 até 11001000 00010111 00011111 11111111	2
senão	3

Regra da concordância do prefixo mais longo

Concordância do prefixo	Interface do enlace
11001000 00010111 00010	0
11001000 00010111 00011000	1
11001000 00010111 00011	2
senão	3

Exemplos

DA: 11001000 00010111 **00010**110 10100001

Qual interface?

0

DA: 11001000 00010111 **00011**000 10101010

Qual interface?

1