

COMUTAÇÃO DE PACOTES

Sistemas de comutação são formados por um conjunto de nós e enlaces de interconexão cuja estrutura pode ser utilizada para comunicação de longa distância (WAN) para a transmissão de voz, dados etc., conforme demonstra a figura a seguir:

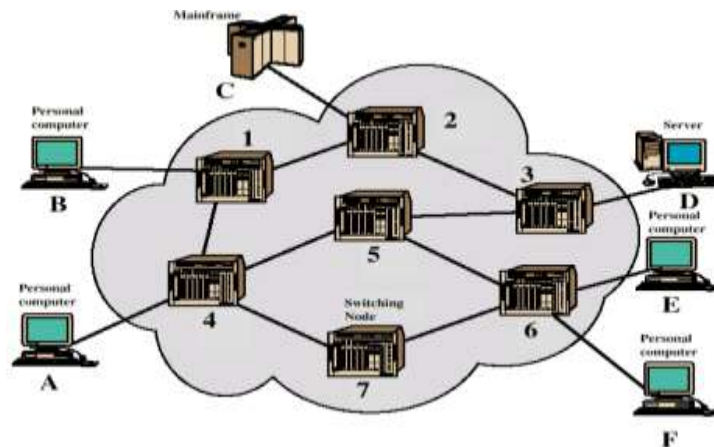


Fig.1

Conceitualmente os sistemas de comutação podem ser vistos como uma sub-rede de comunicação (representada pela nuvem da figura) com as seguintes características:

- ☐ Normalmente a rede não é completamente conectada (não existem enlaces cruzados entre todos os nós)
- ☐ Alguns nós possuem apenas a função de repassar adiante as informações que chegam, como é o caso dos nós 5 e 7. Os demais nós, além de repassar as informações adiante, devem atender as conexões com as estações usuárias finais
- ☐ Os troncos de interligação de nós normalmente usam algum esquema de multiplexação FDM ou TDM.

O sistema de transmissão de dados nas redes de comutação pode ser feito sob dois aspectos: o primeiro deles é a transmissão por comutação de circuitos, muito utilizados para telefonia; o segundo é a transmissão por comutação de pacotes, mais voltada para a transferência de dados. As seções que se seguem fazem uma abordagem sobre a comutação de pacotes:

1. Princípios da Comutação de Pacotes

Desde a invenção do telefone, a comutação de circuitos tem sido a tecnologia dominante nas operadoras de telecomunicações para transmissão de voz, apesar de, atualmente, existirem várias outras tendências tecnológicas em função das novas necessidades dos usuários. A comutação de pacotes surgiu para atender essas novas demandas. A transmissão de dados em redes de comutação de circuitos possui as seguintes características:

- ☐ A premissa da comutação de circuitos é que os recursos são dedicados para os usuários e não há mais de um usuário usufruindo um mesmo recurso (banda alocada, por exemplo).
- ☐ Numa transmissão entre computadores não existe a característica de transmissão contínua de dados. Normalmente o que ocorre são rajadas de dados seguidas por momentos de silêncio. Por isso os recursos da rede acabam ficando disponíveis na maior parte do tempo.
- ☐ A taxa de dados é fixa, ou seja, as entidades pares só podem se comunicar via rede de comutação de pacotes se as taxas de transmissão de ambas forem iguais.

Por sua vez, na comutação de pacotes os dados são transmitidos em pequenos blocos de dados de tamanho fixos, normalmente em torno de 1000 octetos. No caso de mensagens grandes, essas são quebradas em pedaços menores para se adequarem ao tamanho máximo do pacote. Além disso, cada pacote deve conter uma parte de dados do usuário e outra parte contendo informações de controle (dados sobre endereçamento de origem/destino etc.). Os pacotes, por sua vez, são recebidos, armazenados brevemente (bufferizados) e passados para o próximo nó. Essa técnica é conhecida como “store-and-forward”. A figura abaixo apresenta uma visão geral da utilização de pacotes para comunicação entre dois computadores:

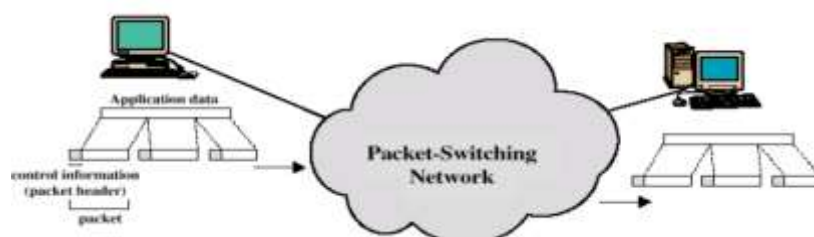


Fig.2

Algumas vantagens podem ser mencionadas sobre comutação de pacotes:

- ⇒ Melhoria no uso (mais eficiência) das linhas de transmissão. Isso ocorre por que um mesmo enlace ponto-a-ponto pode ser compartilhado por muitos pacotes no tempo e os pacotes são enfileirados e transmitidos tão rápido quanto possível, liberando o recurso para os demais.
- ⇒ Quanto à taxa de conversão de dados, cada estação pode se conectar ao nó local na velocidade que julgar mais apropriada. Por consequência, os comutadores devem possuir *buffers* de dados para compatibilizar essas velocidades.
- ⇒ Uma característica interessante é que os pacotes são aceitos mesmo quando a rede está ocupada. Nesse caso o que ocorre é que a liberação dos pacotes para o destino fica mais lenta.
- ⇒ Por último, é possível definir prioridades entre pacotes.

2. Técnica de Comutação por Pacotes

De um modo geral a comutação vê as mensagens do usuário como blocos de dados a serem despachados para o destino. Dessa forma as mensagens que chegam para a rede são segmentadas em pacotes de dados associados a um cabeçalho de controle. Do ponto de vista de comutação, os pacotes podem ser manuseados de duas maneiras:

a) Uma das formas de transporte de dados por comutação de pacotes é o sistema de datagrama. Nesse mecanismo os dados do usuário trafegam em pacotes independentes e há o risco de pacotes chegarem fora de ordem ou mesmo se perderem. Cabe ao receptor reordenar os pacotes e recuperar os perdidos, caso ocorram.

b) Uma outra maneira de transportar pacotes é na forma de circuito virtual. Nesse esquema pode-se mencionar as seguintes características:

- Uma rota deve ser definida antes que os pacotes sejam enviados
- Uma requisição de chamada e sua consequente aceitação (“call request” e “call accept”) estabelecem uma conexão
- Cada pacote contém um identificador de circuito virtual ao invés de um endereço de destino
- Nenhuma decisão de rota é requerida para cada pacote individualmente
- O encerramento da conexão (“clear request”) desfaz o circuito
- Não existe um caminho dedicado, ou seja, os recursos podem ser compartilhados.

Independente da forma de transporte de dados adotada na comutação de pacotes, é possível perceber que há uma melhoria na utilização da rede, que passa a ser mais bem compartilhada entre os diversos usuários. Para ilustrar esse esquema, é interessante analisar a figura a seguir:

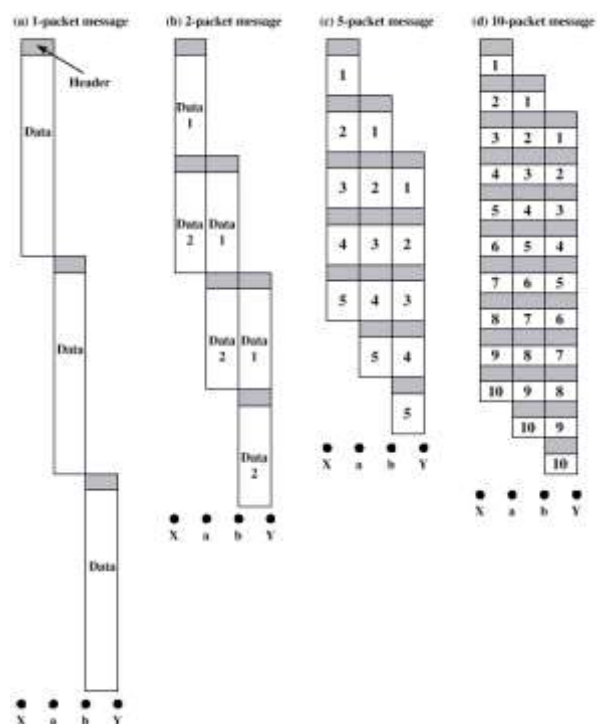


Fig.3

Neste exemplo, é assumido que existe um circuito virtual da estação X até a estação Y, passando pelos nós a e b. A mensagem a ser enviada possui 30 octetos e cada pacote contém 3 octetos de informação de controle (cabeçalho). Se considerarmos a transmissão serial dessa mensagem, ela será enviada como um único pacote de 33 octetos (30 de dados + 3 de cabeçalho). Dessa forma o pacote

é transmitido de X para a, de a para b e, finalmente, de b para Y, ocupando um tempo de transmissão equivalente a 99 octetos. Por outro lado, se essa mensagem for quebrada em dois pacotes de 15 octetos cada, ocorrerá o fenômeno de *pipelining*, onde os nós da rede trabalham em paralelo para atender uma determinada demanda. Nesse caso, a parte b da figura mostra que o tempo de transmissão será menor do que no primeiro caso (o equivalente a transmissão de 72 octetos). Seguindo esse raciocínio, se a mensagem for quebrada em 5 partes de 6 octetos cada, o tempo de transmissão diminui para 63 octetos. Por fim, a parte d da figura mostra que se houver uma fragmentação muito próxima ao tamanho do cabeçalho, o *pipelining* pode não ser tão compensatório (em outras palavras, existe um limite a ser considerado para o particionamento de mensagens). Ainda mais, o exemplo não apresenta o processamento e o enfileiramento das mensagens nos comutadores. Esses atrasos são tão maiores quanto mais pacotes são manuseados por uma única mensagem.

Obs.: Em contradição ao que foi dito, a tecnologia ATM trabalha com pacotes extremamente pequenos e, mesmo assim, produz uma performance bastante aceitável. Essa contradição ocorre por que existem outros aspectos do projeto ATM que devem ser considerados e que serão considerados num estudo mais adiante.

3. Comparação entre Comutação de Circuito e Comutação de Pacotes

Tendo visto o funcionamento interno de uma rede de comutação de pacotes, é possível compará-la com a comutação de circuitos com relação à performance. A figura abaixo ilustra um diálogo entre duas estações usando as tecnologias vistas. Nesse caso estão sendo considerados três tipos de atrasos:

- ❑ Atraso de propagação – É o tempo que o sinal leva para se propagar de um nó até o próximo. Este tempo normalmente é insignificante em função das capacidades de transmissão dos meios físicos atuais. Como exemplo, um sinal eletromagnético viaja por um cabo a uma velocidade em torno de 2×10^8 metros/s.
- ❑ Tempo de transmissão – É o tempo que o transmissor leva para enviar um bloco de dados. Como exemplo, um transmissor em geral leva um segundo para transmitir um bloco de dados de 10 mil bits em uma linha de 10kbps.
- ❑ Atraso do nó – Esse é o tempo que um nó demora em realizar a comutação propriamente dita.

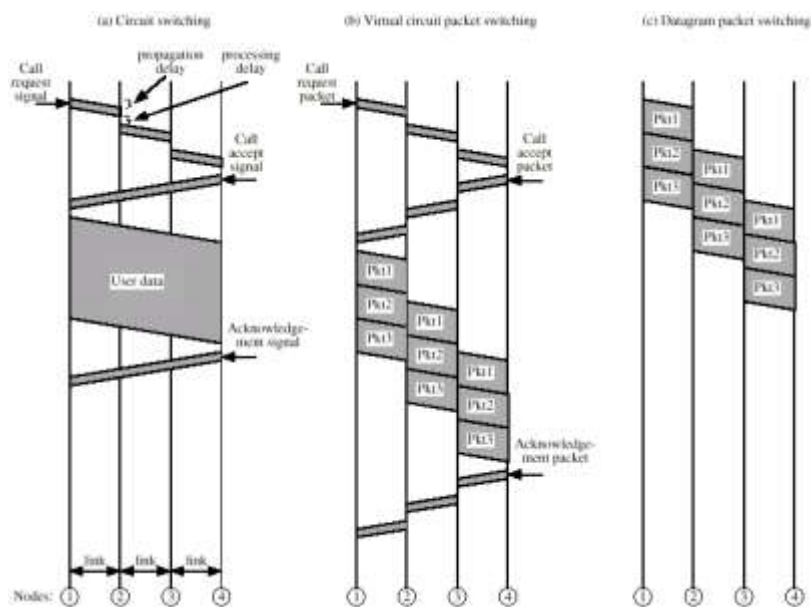


Fig.4

Algumas observações importantes:

- Na comutação de circuitos existe um atraso antes do envio da mensagem. Na solicitação de chamada ("call request") esse atraso é formado pelo atraso de propagação mais o atraso de processamento dos comutadores; na aceitação da chamada ("call accept") esse atraso é formado apenas pelo atraso de propagação uma vez que os recursos já foram alocados e registrados nos comutadores. Uma vez que o caminho está pronto, a mensagem pode ser enviada como um grande bloco e o atraso nos comutadores é insignificante.
- Na comutação de pacotes por comutação de circuitos também existe a necessidade de um estabelecimento de circuito, mas tanto a solicitação de circuito quanto a resposta da solicitação experimentam atrasos de propagação e de processamento dos nós. Além disso, como a transmissão de dados ocorre na forma de pacotes, os atrasos de processamento dos nós devem ser considerados (gerenciamento das filas de entrada e saída de pacotes). Deve estar claro para o leitor que a comutação de pacotes por circuito virtual não é necessariamente mais rápida do que a comutação de circuitos. Para concluir isso, basta perceber que na comutação de circuitos a taxa de transmissão é constante enquanto que na comutação de pacotes os atrasos são variáveis e podem aumentar com a carga da rede.

- Nas redes de pacotes por datagrama não existe fase de estabelecimento. Os dados são enviados diretamente e a performance, principalmente para mensagens curtas, é bem superior do que nas duas técnicas anteriores.
- É importante lembrar que apesar da figura acima apresentar uma caracterização de atraso, a performance de uma rede de comutação depende de uma série de fatores além dos que foram mencionados, tais como o tamanho e a topologia da rede, o padrão de carga etc.

Além das questões de performance existem outros pontos de comparação que podem ser mencionados. A tabela abaixo relaciona essas três tecnologias apresentando outras características de cada uma delas:

Comutação de Circuitos	Com. Pacotes – Datagrama	Com. Pacotes - Circuito Virtual
Caminho de transmissão dedicado	Não há caminho dedicado	Não há caminho dedicado
Transmissão contínua de dados	Transmissão de pacotes	Transmissão de pacotes
A unidade de informação é a <u>mensagem</u> e essas não são armazenadas nos comutadores	A unidade de informação é o pacote e esses podem ser armazenados e depois liberados.	A unidade de informação é o pacote e esses são recebidos e liberados imediatamente.
O caminho é estabelecido para o diálogo inteiro	Uma rota é estabelecida para cada pacote	Rota estabelecida para a conversação inteira
Atraso de "call setup"; atraso de transmissão insignificante	Atraso de transmissão do pacote	Atraso de "call setup" e atraso de transmissão dos pacotes.
Sinal de ocupado se destinatário está ocupado	Chamador pode ser avisado se o pacote não for entregue.	Chamador será notificado sobre conexão não permitida.
Sobrecarga pode bloquear novos "call setup's"; Carga grande não gera atraso nas chamadas já ativas.	Sobrecarga aumenta atraso na entrega dos pacotes	Sobrecarga pode bloquear "call setup's"; aumenta o atraso na entrega de pacotes
Nós de comutação são eletromecânicos ou computadorizados	Pequenos nós de comutação (computadores)	Pequenos nós de comutação (computadores)
Usuário é o responsável por proteção quanto à perda de mensagens	A rede pode ser responsável por pacotes individuais	A rede pode ser responsável por sequências de pacotes
Normalmente não há conversão de códigos ou de velocidades de transmissão.	Há conversão de códigos e taxas de transmissão	Há conversão de códigos e taxas de transmissão
Transmissão em largura de banda fixa	Usa largura de banda variável	Usa largura de banda variável
Não há bits de "overhead" após o "call setup"	Há bits de "overhead" em cada pacote	Há bits de "overhead" em cada mensagem

4. Operação Interna e Externa das Redes de Comutação

Neste ponto é possível diferenciar entre o que é oferecido para o usuário (visão externa da rede) e o modo como a rede operacionaliza o serviço proposto. A figura abaixo mostra uma visão externa de uma rede de comutação de pacotes:

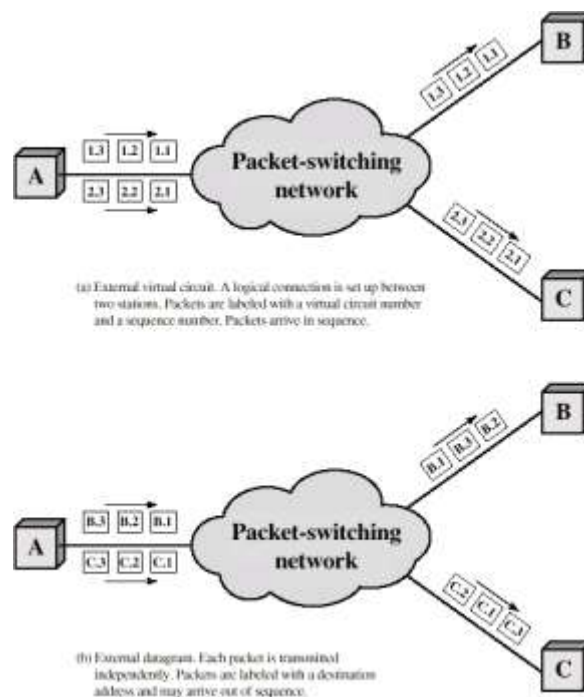


Fig.5

Na parte (a) da figura, uma conexão lógica é estabelecida entre as duas estações finais, garantindo a entrega ordenada de pacotes. Cada pacote é rotulado com um identificador de circuito virtual e um número de sequência para prover garantias ao usuário. Como exemplo, é possível citar o protocolo TCP da camada de transporte do modelo TCP/IP. Na parte (b) da figura, não existe uma conexão lógica entre as partes comunicantes e os pacotes podem chegar fora de ordem no destino. Um exemplo de implementação desse mecanismo é o protocolo UDP do modelo TCP/IP.

Do ponto de vista interno a rede pode também funcionar como um circuito virtual ou como um datagrama, independente do que foi proposto ao usuário como serviço, conforme demonstra a figura a seguir:

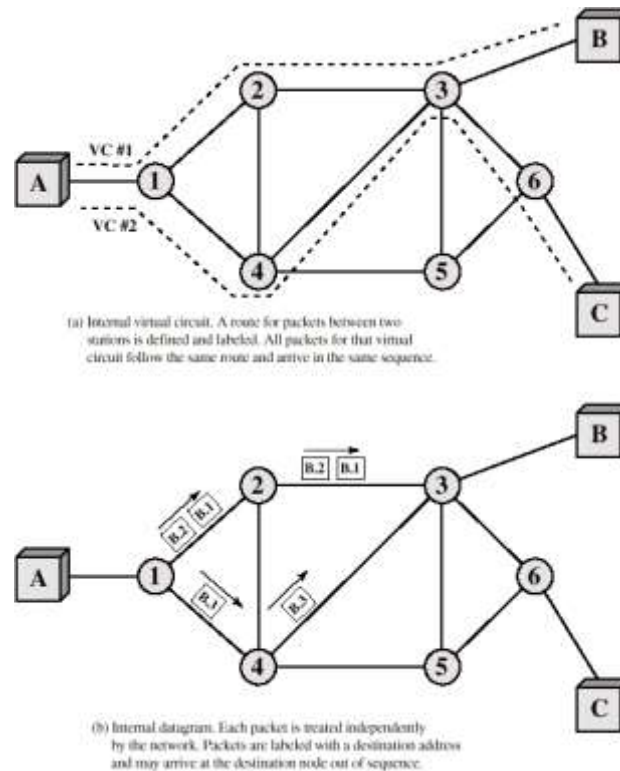


Fig.6

Na parte (a) da figura, uma rota entre duas estações é formada e rotulada. Todos os pacotes para um determinado circuito virtual seguem a mesma rota e chegam em sequência. Na parte (b) não há formação de circuito virtual e os pacotes são roteados independentemente, podendo até mesmo chegar fora de ordem. Como exemplo do primeiro caso, pode-se mencionar a tecnologia Frame Relay (vista mais adiante). O protocolo IP do modelo TCP/IP é um caso prático de rede por datagrama.

A questão que surge é qual a melhor escolha a ser feita tanto para a visão externa como para a visão interna da rede. Essa resposta depende dos objetivos de comunicação da rede e os fatores de custo que devem prevalecer. Algumas observações podem ser lembradas quanto aos dois mecanismos:

- O serviço de datagrama interno e externo permite uso eficiente da rede; não há fase de “call setup” e não há necessidade de pendurar pacotes enquanto uma retransmissão por erro é feita. Essa última característica é interessante para aplicações de tempo real.
- O serviço de circuito virtual pode prover um sequenciamento fim-a-fim e controle de erro. Este serviço é atrativo para suporte de aplicações orientadas à conexão tais como transferência de arquivo e terminal de acesso remoto.

Na prática o serviço de circuito virtual é muito mais comum do que o serviço de datagrama, em função da confiabilidade e conveniência associadas (90% das aplicações Internet trabalham com TCP, 10% usam UDP).

5. Roteamento em Redes de Comutação de Pacotes

Um dos aspectos mais complexos e cruciais no projeto de uma rede é o roteamento. Uma estratégia de roteamento em geral envolve os seguintes requisitos: correteza, simplicidade, robustez, estabilidade, justiça, otimalidade e eficiência.

Correteza e simplicidade são auto explicativos. Robustez é a habilidade da rede em liberar pacotes via alguma rota alternativa para fugir de eventuais falhas e/ou sobrecargas em partes da rede. Idealmente a rede deve reagir a tais contingências sem perder pacotes ou quebrar circuitos virtuais. O projetista que procura robustez talvez não consiga atender o requisito de estabilidade, uma vez que esses são conflitantes. As técnicas que reagem às condições de mudança têm uma tendência infeliz de, ou reagir muito lentamente

aos eventos ou experimentar movimentos instáveis de um extremo ao outro. Por exemplo, uma rede pode reagir a um congestionamento em uma área por transportar a maior parte da carga para uma segunda área. Agora a segunda área está sobrecarregada e a primeira está subutilizada, causando um segundo movimento de carga. Durante essa movimentação de carga, os pacotes podem viajar através da rede por um bom tempo sem chegar a lugar algum.

Justiça e otimalidade também são critérios contraditórios. Por exemplo, uma situação ótima e que resulta numa melhoria de performance da rede como um todo, é dar maior prioridade para troca de informações entre estações mais próximas entre si, como é o caso das estações A/A', B/B' e C/C' na figura abaixo. O problema é que essa otimização acaba produzindo uma injustiça com as estações mais distantes, como é o caso das estações X/X' apresentadas na figura.

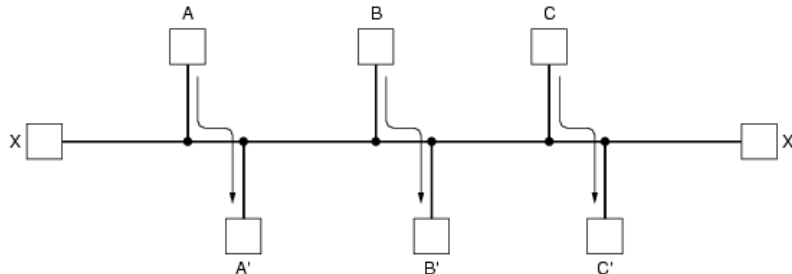


Fig.7

Finalmente, qualquer técnica de roteamento envolve algum "overhead" de processamento em cada nó e também nas linhas de transmissão, impactando na eficiência da rede. Dependendo das características das métricas e parâmetros utilizados para a decisão de roteamento, esse impacto pode ser maior ou menor.

Existem critérios inerentes aos algoritmos de roteamento que vale a pena mencionar. Alguns deles estão colocados nos itens a seguir:

- a) Critério de Performance – A seleção de uma rota é geralmente baseada no critério de melhor performance. O critério mais simples é a escolha de uma rota que passe pela menor quantidade de nós ("hops"). No caso da figura abaixo, o melhor rota entre o nó 1 e o nó 6 é 1-3-6. No entanto, se consideramos como métrica o caminho cujo custo das linhas seja o menor, então o melhor caminho entre 1 e 6 seria 1-4-5-6 (custo = 4, menor do que o custo = 10 de 1-3-6).

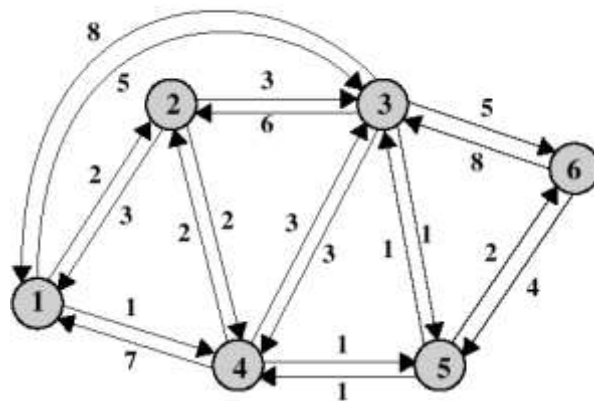


Fig.8

Obs.: O custo apresentado na figura pode ser oriundo de uma série de fatores, tais como: tamanho das filas nos comutadores, tráfego nas linhas, taxa de transmissão das linhas etc.

- b) Em que momento a decisão de roteamento irá ocorrer – Para redes tipo circuito virtual, a decisão de roteamento na fase de estabelecimento da conexão lógica, enquanto que nas redes datagrama a decisão ocorre no momento em que os pacotes chegam no comutador
- c) Quem vai tomar a decisão de roteamento – Essa decisão pode ser: centralizada num único roteador (que mantém informações sobre a rede como um todo), pode ser distribuída entre os diversos comutadores ou pode ser concentrada na estação de origem da informação ("source routing").
- d) Qual é a fonte de informação de rede para o roteamento – A maioria das estratégias de roteamento determinam que decisões sejam tomadas com base no conhecimento sobre a topologia da rede, sobre a carga na rede e sobre os custos dos enlaces. Nesse caso as informações normalmente vêm dos comutadores vizinhos ou mesmo de todos os comutadores da rede (como é o caso do roteamento centralizado). Existem estratégias que se baseiam apenas em informações locais como por exemplo, o tamanho das filas internas das portas de saída dos comutadores. Por fim, existem estratégias que não se baseiam em nada, como é o caso do "flooding" (visto mais adiante) que não analisa métrica nenhuma para realizar roteamento.

- e) Qual é o momento de atualizar as informações da rede – Se nenhuma informação de rede é necessária, então essa questão de tempo não se aplica. No entanto, para estratégias que coletam dados de outros nós da rede (nós da rede inteira ou dos nós adjacentes), é preciso definir de quanto em quanto tempo essas informações são trocadas. As alternativas de roteamento que são fixas não precisam sofrer atualizações, mas as técnicas que procuram adaptar o roteamento em função das mudanças na rede precisam ter dados coletados a intervalos regulares.

Quanto à estratégia de roteamento os algoritmos podem ser agrupados em algoritmos adaptativos e não adaptativos. As sub-seções que se seguem abordam cada um deles.

5.1 Algoritmos Não Adaptativos

Esses algoritmos não se baseiam nas medidas ou estimativas de tráfego corrente nem em mudanças na topologia da rede. A escolha de rota entre um par de nós qualquer é feita antecipadamente quando a rede é inicializada. Este procedimento é algumas vezes chamado de roteamento estático. Algumas técnicas podem ser descritas:

- ⇒ Menor Caminho (“Shortest Path”) – Existem várias estratégias para implementar esse tipo de algoritmo, dentre elas a mais famosa é uma solução dada por Dijkstra, na qual a rede é vista como um grafo de nós passíveis de fazerem parte de uma rota. Inicialmente todos os nós são marcados como nós-tentativa. Na medida em que a rota vai sendo formada, os nós-tentativa vão sendo transformados em nós-permanentes. A figura a seguir mostra o processo de descoberta do menor caminho entre A e D.

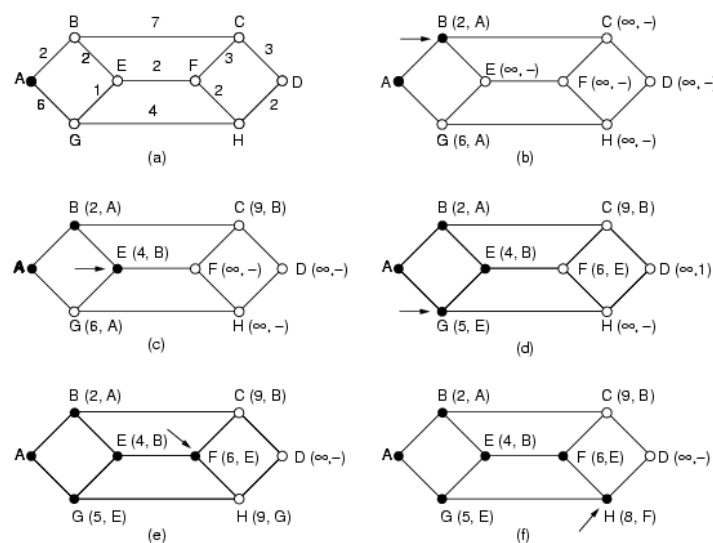


Fig.9

- ⇒ Inundação (“flooding”) – Essa técnica não requer informação da rede para funcionar. Um pacote é enviado da origem para cada um de seus vizinhos que, por sua vez, retransmitem os pacotes de entrada pelas portas de saída, conforme demonstra a figura abaixo:

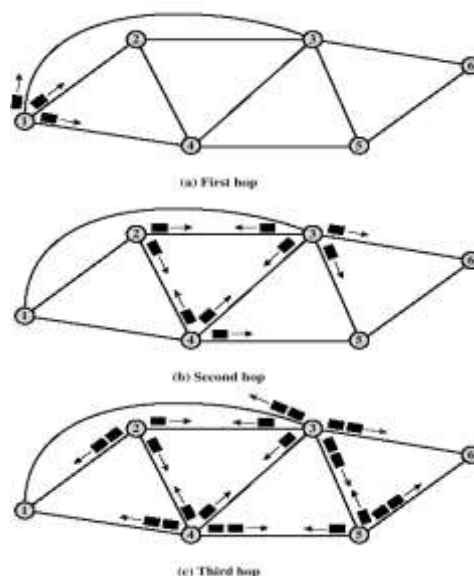


Fig.10

- ⇒ Roteamento Randômico – O roteamento randômico é similar ao “flooding”, só que os comutadores liberam os pacotes de entrada pelas portas de saída cuja probabilidade de alcançar o destino seja maior. O cálculo das probabilidades de cada porta não é assunto dessa apostila.

5.2 Algoritmos Adaptativos

Os algoritmos adaptativos são aqueles que possuem métricas associadas ao comportamento atual da rede, tais como tráfego na rede, tamanho das filas, falhas num comutador, queda de um enlace, congestionamento etc. Para o roteamento adaptativo ser possível, informações sobre o estado da rede devem ser trocadas entre os nós de comutação. Nesse caso é preciso existir um balanceamento entre a qualidade da informação trocada e o “overhead” gerado. Ou seja, quanto mais informações são trocadas e quanto menor o intervalo entre essas trocas, melhor a qualidade do roteamento. Por outro lado, quanto mais informações de controle na rede, maior o “overhead”, causando degradação de performance na rede. Existem vários pontos negativos associados com roteamento adaptativo, dentre eles:

- ⇒ A decisão de roteamento é mais complexa e, portanto, a responsabilidade de processamento nos comutadores cresce bastante;
- ⇒ Na maioria dos casos, as estratégias adaptativas dependem de informação de estado (“status”) que é coletada em um local e processada em outro, o que também gera aumento de tráfego na rede;
- ⇒ Uma estratégia adaptativa pode reagir muito rapidamente, causando oscilação na produção de congestionamento. Por outro lado, se a reação é lenta demais, o resultado não será significativo.

Independente desses perigos reais, as estratégias de roteamento adaptativo são largamente utilizadas por duas razões:

- ⇒ Uma estratégia de roteamento adaptativo pode melhorar a performance da rede, como pode ser percebido pelo usuário da rede;
- ⇒ Uma estratégia de roteamento adaptativo pode ajudar no controle de congestionamento da rede, conforme pode ser visto mais adiante.

Esses benefícios podem ou não ser alcançados, dependendo da clareza do projeto da rede e a natureza da carga. Em redes largas e complexas esses problemas são bastante difíceis de gerenciar apropriadamente. Uma demonstração disso é o fato da rede ARPANET (e outras redes produzidas pela IBM e DEC) que é o pilar da Internet já ter lançado várias versões do mesmo protocolo tentando resolver os problemas percebidos na versão anterior.

Para ilustrar roteamento adaptativo vamos considerar na rede da figura 8, que existe uma necessidade de roteamento do nó 1 até o nó 6. Nesse caso os comutadores tomam as decisões de roteamento baseado apenas nas informações internas do comutador e esses, por sua vez, liberam os pacotes nas portas cujo tamanho de fila seja o menor possível. Aqui existe um outro parâmetro, denominado BIAS que é o custo ou probabilidade dessa porta de saída ser apropriada para alcançar o destino.

Imagine agora que o nó 4 acabou de receber um pacote e precisa repassá-lo adiante em direção ao nó 6. A figura abaixo apresenta a fila Q e o BIAS B de cada uma das portas de saída. Nesse caso a porta escolhida será a que se conecta ao nó 3 por que é a porta que tem a menor soma entre esses dois parâmetros.

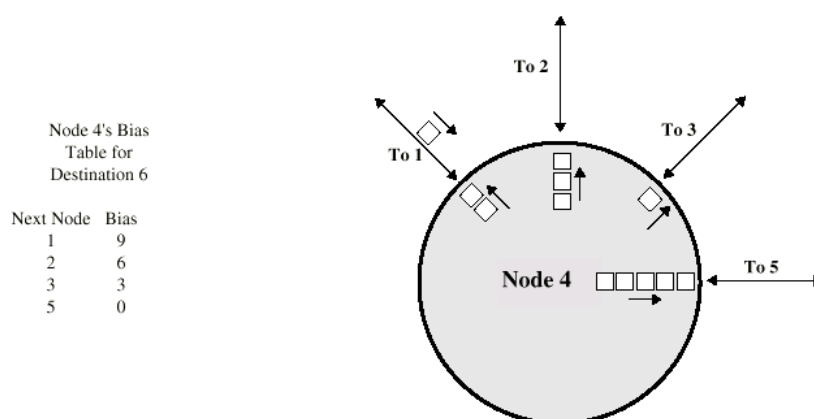


Fig.11

Esquemas baseados apenas em informação local são raramente usados por que eles não exploram facilmente a informação disponível. Ou seja, é difícil chegar a uma conclusão próxima do ideal. As estratégias baseadas em informações de nós adjacentes e de todos os nós são mais comumente encontradas. Dois algoritmos mais famosos que seguem essa linha de raciocínio são o “distance vector” usado no protocolo RIP (‘Routing Information Protocol’) e o “link state” usado no protocolo OSPF, bastante popular atualmente.

No “distance vector” por exemplo, cada roteador mantém uma tabela de rotas indexada por uma entrada para roteador da sub-rede. Esta entrada contém duas partes: uma linha de saída preferencial para usar como destino e uma estimativa de tempo e distância para aquele destino. A métrica usada pode ser o número de saltos (“hops”), o atraso em milissegundos, o número total de pacotes enfileirado

ao longo do caminho ou algo similar. Ao roteador é assumido conhecer a “distância” de cada um de seus vizinhos, de acordo com a métrica decidida. Como um exemplo assume-se que o atraso é usado como métrica e que cada roteador troca informações com seus vizinhos contendo os atrasos estimados para cada destino.

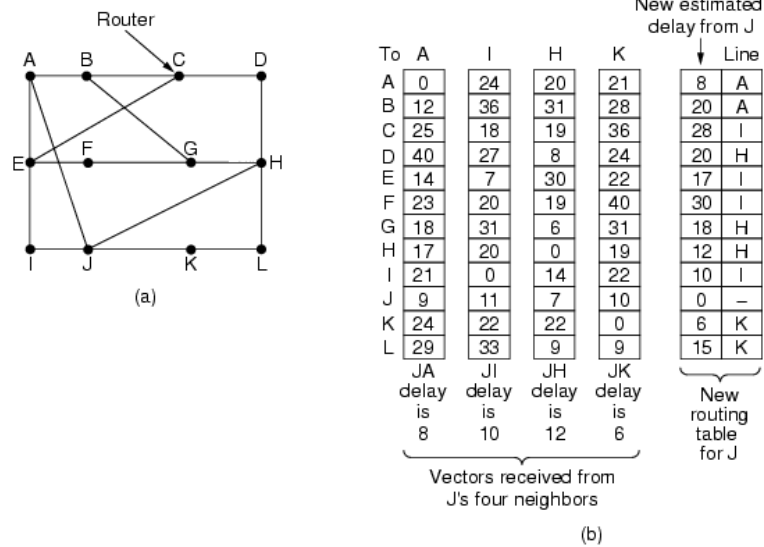


Fig.12

Na figura acima a parte (a) representa a sub-rede como um todo e a parte (b) apresenta os vetores de distância recebidos por J de seus vizinhos. Como J sabe o seu atraso em relação aos vizinhos, será possível recalcular um novo vetor de melhor caminho, considerando as linhas cujo atraso acumulado seja menor em relação a um determinado destino. A parte direita da figura representa a nova tabela de roteamento calculada em função das informações recebidas.