Roteamento Baseado em Requisitos de Qualidade de Serviço

Luís Henrique Maciel Kosmalski Costa
luish@gta.ufrj.br

Otto Carlos Muniz Bandeira Duarte
otto@gta.ufrj.br

Grupo de Teleinformática e Automação – GTA
COPPE/EE - Programa de Engenharia Elétrica/Departamento de Eletrônica
Universidade Federal do Rio de Janeiro
Caixa Postal 68504 - CEP 21945-970 - Rio de Janeiro - RJ - Brasil

Resumo

Existem diversos motivos para considerar o roteamento baseado em reserva de recursos como uma alternativa para se prover aplicações com múltiplos requisitos de Qualidade de Serviço (QoS) na Internet. Vários aspectos merecem atenção, como os requisitos deste tipo de roteamento, que tipo de arquitetura pode ser utilizada na prática, quais as considerações técnicas e políticas que devem ser feitas numa arquitetura de roteamento Internet baseada em QoS. Neste trabalho, é realizada uma análise dos principais problemas, incluindo o cálculo de rotas, eficiência, robustez e escalabilidade. São apresentadas algumas propostas realizadas no âmbito do IETF (*Internet Engineering Task Force*), considerando a adaptação dos protocolos de roteamento ao protocolo RSVP (*Resource ReSerVation Protocol*).

Abstract

There are many reasons to consider resource reservation based routing as an alternative to provide applications with several Quality of Service (QoS) requirements in the Internet. Several questions deserve attention, as the requirements for this routing model, what kind of architecture is practical, what are the technical and political issues that arise in implementing a QoS-based Internet routing architecture. In this work, an analysis of the main problems is made, including path computation, efficiency, robustness and scalability. Some proposals made in the scope of the IETF (*Internet Engineering Task Force*) are presented, considering the adaptation of routing protocols to the RSVP (*Resource ReSerVation Protocol*).

1 Introdução

A Internet vem sendo usada cada vez mais no transporte de informação multimídia , como imagens, voz e vídeo. Com o aumento da capacidade de processamento dos computadores pessoais e o surgimento de novas aplicações multimídias, a Internet tende a receber uma maior quantidade deste tipo de tráfego. Este tipo de utilização , no entanto, não é o uso para o qual o protocolo IP (*Internet Protocol*) foi projetado, há mais de vinte anos. A arquitetura de rede da Internet foi projetada para o envio da informação usando a disciplina de serviço de melhor esforço, ou seja, sem quaisquer garantias de vazão ou de atraso.

Para o suporte de tráfego de tempo real na Internet, algumas novas características e serviços devem ser adicionados. Novas classes de serviço precisam ser oferecidas, além do envio de melhor esforço, para prover alguma garantia de atraso e vazão às aplicações. É necessário um mecanismo que permita às aplicações sinalizar seus requisitos de QoS à rede, e mecanismos de gerenciamento de tráfego nas estações e nos roteadores.

O IETF (*Internet Engineering Task Force*) vem trabalhando nas duas primeiras questões, a definição de novas classes de serviço [1] e um protocolo de reserva de recursos, o RSVP (*Resource ReSerVation Protocol*) [2, 3, 4], que as aplicações poderão utilizar para solicitar à rede a QoS desejada. Um tema bastante polêmico que se discute atualmente é se uma arquitetura de roteamento que permita a descoberta de caminhos que dão suporte a determinados requisitos de QoS é necessária. Numa tal arquitetura, as rotas seriam definidas através do estado da rede e dos requisitos de QoS dos fluxos.

A necessidade de mecanismos de QoS poderia ser eliminada através da provisão de enlaces ou comutadores com mais recursos, porém esta alternativa torna-se inviável economicamente. Em geral, provedores de serviço públicos e administradores de rede desejam transportar o maior número de fluxos com o menor custo possível. Nestas circunstâncias, existe uma justificativa para o roteamento baseado em QoS.

Em segundo lugar, a engenharia econômica de uma rede depende de um esquema de roteamento eficiente. Com o esquema de roteamento IP tradicional, uma única rota é escolhida para todo o tráfego entre dois pontos. Para que alguma QoS seja garantida, é necessário que esta rota suporte a demanda de taxa binária de pico. Esta aproximação leva a uma sub-utilização dos recursos da rede, pois podem haver outros caminhos não usados entre os dois pontos, e durante a maior parte do tempo, o tráfego pode não estar demandando a taxa binária máxima. Por outro lado, um algoritmo de roteamento baseado em QoS permitiria que a capacidade da rede fosse projetada eficientemente graças à sua habilidade em levar em conta o estado da rede atual nas decisões de roteamento.

Finalmente, um esquema de roteamento dependente do estado da rede pode compensar um projeto de engenharia da rede impreciso. Especificamente, quando a carga de tráfego da rede excede os limites devido a circunstâncias excepcionais, um roteamento dependente do estado da rede permite que uma maior quantidade de tráfego seja roteada, com uma degradação de desempenho menor que a experimentada com um esquema de roteamento insensível ao estado da rede. O roteamento baseado em QoS deve portanto ser considerado como um componente do esquema geral para a provisão de garantias de QoS para as aplicações na Internet.

Na seção 2 deste trabalho, são analisados os principais requisitos que o roteamento baseado em requisitos de QoS deve respeitar. Na seção 3 são apresentadas as principais características relacionadas com os algoritmos de roteamento, considerando requisitos de QoS comuns em grande parte das aplicações. A seção 4 apresenta uma análise do relacionamento entre o RSVP e os protocolos de roteamento, uma vez que este será a interface que possibilitará ao usuário informar seus requisitos de QoS à rede. Na seção 5 são feitas algumas considerações finais.

2 Requisitos do Protocolo de Roteamento

Considerando a organização da Internet, a natureza do tráfego gerado, as classes de serviço que devem ser providas e os requisitos dos usuários, podem ser definidos diversos pré-requisitos ao roteamento baseado em reservas de recursos. Para uma melhor organização, a seguir são apresentados diversos requisitos, divididos nos grupos principais: métricas e rotas; robustez; eficiência; escalabilidade; prioridades; inter-domínios e políticas. Outros requisitos, como o suporte a mobilidade, podem ser definidos [5].

Quanto a métricas e rotas, é desejável que o protocolo de roteamento suporte múltiplas métricas para os caminhos (atraso, faixa de transmissão, variação de atraso etc.). A definição das métricas necessárias deve ser realizada baseada nas classes de serviço suportadas [6]. O protocolo deve ser capaz de calcular dinamicamente rotas satisfazendo múltiplas restrições. Deve oferecer suporte de QoS para vários tipos de fluxos, incluindo ponto-a-ponto (*unicast*) e ponto-a-multiponto (*multicast*) e, além disso, deve suportar receptores heterogêneos em termos dos requisitos de QoS.

O protocolo deve ser robusto, ou seja, possuir as habilidades de detectar e responder a mudanças nas capacidades de QoS dos enlaces e de continuar o roteamento mesmo diante de falhas múltiplas na rede. Em resumo, o protocolo deve ser adaptável às variações da rede.

Em se tratando de eficiência, deseja-se que o protocolo seja capaz de rotear fluxos de curta duração com a menor sobrecarga de processamento possível, além de utilizar os recursos da rede de forma eficiente, através de balanceamento da carga, e controles de admissão de chamadas globais. A sobrecarga de processamento devida ao roteamento deve ser mínima. Deve ser possível o controle dos recursos, ou seja, a capacidade de limitar o consumo de recursos da rede pelas diferentes classes de tráfego.

O protocolo deve ser escalável, pré-requisito indispensável quando se considera a dimensão de uma rede como a Internet. O algoritmo utilizado deve ser capaz de lidar com o aumento do número de nós e de enlaces da rede sem se tornar inviável em termos da capacidade de processamento das máquinas. Deve existir um mecanismo para estabelecer prioridades de transmissão entre os fluxos e para permitir que tráfegos de alta prioridade sejam estabelecidos com precedência em relação aos menos prioritários.

Os requisitos para o roteamento entre domínios diferentes constituem um aspecto importante que tem gerado bastante discussão [7]. Mais especificamente, a natureza da informação trocada na fronteira entre dois domínios administrativos deve ser determinada. Se a troca de informação for pequena, o roteamento inter-domínio pode ser relativamente simples. Em todo caso, a capacidade de introdução de restrições de acordo com políticas no roteamento do fluxo de informação na vizinhança entre duas organizações é indispensável. Nas seções seguintes, alguns dos aspectos relacionados com o protocolo de roteamento são explorados.

2.1 Determinação da qualidade de serviço e reserva de recursos

O Grupo de Trabalho em Serviços Integrados [1, 8, 9] do IETF vem se concentrando no gerenciamento local de recursos nos roteadores. A noção de controle de admissão [10] foi introduzida, permitindo ao roteador a decisão de se um novo fluxo deve ser aceito ou rejeitado, baseado na disponibilidade local de recursos. O controle de admissão requer o conhecimento da QoS disponível em todos os enlaces conectados ao roteador. É uma questão em aberto, no entanto, como os valores de QoS são calculados para enlaces de diferentes tipos. Esta questão está no domínio do Grupo de Trabalho ISSLL (*Integrated Services over Specific Link Layers*) do IETF.

Este problema ocorre, por exemplo, no caso de um roteador conectado a uma rede não *broadcast* de múltiplo acesso, como a rede ATM (*Asynchronous Transfer Mode*). Nesta situação, o roteador deverá ter diversas escolhas para o próximo nó, com caminhos correspondentes através da rede ATM, cada um dos quais possivelmente com diferente disponibilidade de QoS. As questões que se impõem são como o roteador pode determinar quais são as opções de roteamento através da rede ATM, qual é a disponibilidade de recursos em cada um destes caminhos, e que valores de QoS devem ser informados pelo roteador à rede ATM, no caso do roteamento baseado em reserva de recursos ser implementado na Internet. Como uma ilustração destes problemas, os padrões atualmente definidos para o roteamento IP sobre ATM, como o NHRP (*Next Hop Routing Protocol*), permitem a seleção de um único ponto de saída da rede ATM para um datagrama IP. Com requisitos de QoS nos fluxos IP, pode não existir um caminho entre o roteador de entrada e o roteador de saída definido pelo NHRP que suporte a exigência de QoS.

Um problema relacionado é a reserva de recursos numa rede *broadcast*, como a rede Ethernet. Como o acesso à rede é distribuído, uma coordenação entre roteadores e estações torna-se indispensável para uma reserva de recursos.

2.2 Propagação da informação

A arquitetura de roteamento baseada no estado do enlace requer que os roteadores informem através de difusão o *status* dos recursos locais (como a faixa de transmissão disponível e os atrasos medidos) e as informações relacionadas com a topologia local para todos os outros roteadores da rede. A difusão da informação de *status* de um roteador para os outros tende a ser um processo dispendioso em termos de sobrecarga do sistema de comunicação e das capacidades de processamento. Até então, o roteamento intra-domínio tem sido baseado em métricas fixas para os enlaces. Neste contexto, a eficiência na propagação da informação não era um pré-requisito. Por esse motivo, os algoritmos de roteamento como o implementado pelo OSPF (*Open Shortest Path First*) utilizam-se de inundação com um período de *refresh* como forma de propagar as atualizações.

Para o roteamento baseado em QoS eficiente, no entanto, mecanismos de inundação não são adequados para a distribuição da informação. Técnicas alternativas para reduzir a sobrecarga da difusão, como o envio baseado em árvores, foram propostas. Além disso, a informação do *status* dos enlaces é em muitos casos volátil, ou seja, seus valores podem mudar significativamente em curtos intervalos de tempo com a admissão de fluxos de tempo real. A maneira como essa informação é quantificada de forma a diminuir a freqüência das atualizações é uma questão importante. Quanto menos precisa for a informação, menor a chance de um roteador fonte encontrar um caminho ideal. Por outro lado, os custos da manutenção de informação muito acurada em cada roteador podem ser muito altos.

2.3 Prioridades entre os fluxos

No caso de existirem fluxos críticos que precisem ter uma maior prioridade que outros fluxos, um mecanismo deve ser implementado na rede para reconhecer os níveis de prioridade. Assume-se que o RSVP possa ser utilizado para sinalizar a prioridade de um fluxo. Existem dois aspectos na priorização entre fluxos. O primeiro, diz respeito à maneira como diferentes usuários poderão controlar a prioridade de seus fluxos. A rede deve ser capaz de conferir se um determinado fluxo tem o direito de utilizar a prioridade requisitada. Em segundo lugar, o mecanismo de roteamento deve assegurar que o caminho com uma determinada QoS será encontrado com uma probabilidade proporcional ao nível de prioridade do fluxo em questão. Ou seja, para uma determinada carga da rede, um fluxo com maior prioridade deve ter mais chances de ser aceito que um fluxo com os mesmos requisitos de QoS, mas com uma menor prioridade.

O mecanismo de roteamento para implementação de prioridades entre os fluxos deve ser baseado em preempção combinada com o re-roteamento dinâmico dos fluxos de baixa prioridade. Num primeiro momento, o roteador deve procurar caminhos com a disponibilidade de recursos necessária para o novo fluxo ser aceito. Não havendo tal caminho, o mecanismo deve tentar um caminho onde recursos adequados possam ser alocados pelo remanejamento de fluxos de menor prioridade. Em último caso, seria usada a preempção dos recursos alocados para os fluxos de menor prioridade. Os roteadores devem tentar o re-roteamento dos fluxos atingidos pela preempção utilizando o mesmo procedimento. O esquema de roteamento considerando prioridades pode, portanto, se tornar complexo.

2.4 Controle de recursos

Dada a existência de múltiplas classes de serviço, é necessário planejar uma rede para suportar as demandas de tráfego previstas para cada classe. Neste sentido, um administrador de rede pode particionar logicamente os recursos dos roteadores e dos enlaces entre as diversas classes de serviço. Este particionamento rigoroso, porém, resultará em uso ineficiente dos recursos da rede, pois durante determinados períodos de tempo pode haver carga excessiva de tráfego de uma classe de serviço e ociosidade de recursos em outra classe. É desejável, portanto, permitir que recursos não utilizados sejam alocados a tráfegos que necessitem. Este compartilhamento deve ser feito de forma controlada, de maneira a impedir que tráfego de uma determinada classe de serviço possa ter acesso a mais recursos do que fora projetada a rede para esta classe por períodos de tempo prolongados.

Esta é uma tarefa do mecanismo de roteamento. Pode ser realizada através de preempção, de maneira não contraditória à atribuição de prioridades dos fluxos de tráfego ativos. O projeto de um esquema de compartilhamento de recursos apropriado e sua incorporação ao roteamento baseado em QoS são importantes. Em particular, as sobrecargas impostas pela implementação do particionamento lógico de recursos e compartilhamento dinâmico é uma questão a ser estudada.

2.5 Fluxos de curta duração

O roteamento baseado em requisitos de QoS incorpora determinadas sobrecargas de processamento durante o estabelecimento do fluxo, por exemplo, no cálculo de uma rota a partir de uma fonte. A questão que se coloca é se este tempo gasto a mais é desproporcional ao tempo de duração das sessões. Em geral, o problema se torna mais grave em redes com circuitos virtuais, como a rede ATM, onde vários circuitos virtuais chaveados (*Switched Virtual Circuit* - SVC) de curta duração resultam em maiores sobrecargas de estabelecimento de conexão. Mesmo sem o roteamento baseado em QoS, as sobrecargas devidas ao estabelecimento de fluxo através do RSVP são adicionadas em cada sessão. A diminuição da sobrecarga devida ao roteamento no estabelecimento de fluxos deve ser buscada, uma das formas possíveis pode ser a utilização de *caches* com as rotas recentemente computadas, de maneira que novos fluxos para os mesmos destinos não exijam novo cálculo dos caminhos.

2.6 Algoritmos para aceitação de fluxos ponto-a-ponto

O cálculo de um caminho possível para um fluxo pode ser feito de maneira simples: o roteador elimina primeiramente todos os nós e enlaces impraticáveis, ou seja, os que não podem dar suporte aos requisitos de QoS determinados, da sua representação local da topologia da rede e encontra um caminho fim-a-fim na rede remanescente. Porém este tipo de algoritmo resulta na aceitação de um número menor de fluxos do que seria possível de outra forma. Este tipo de admissão de fluxos corresponde ao trabalho realizado pelo Grupo de Trabalho em Serviços

Integrados do IETF; neste caso, um fluxo passa pelo controle de admissão à medida em que é aceito por cada roteador ao longo do caminho. Enquanto este tipo de controle é necessário para administrar os recursos em um roteador, não leva em conta qualquer visão global do consumo dos recursos pelos diversos fluxos individuais. A eficiência no roteamento deve ser obtida através de uma camada de controle de admissão a mais. Este controle de admissão de mais alto nível pode considerar o consumo de recursos de cada fluxo em relação aos recursos disponíveis ao longo de um caminho de forma a determinar se a aceitação deste fluxo é desejável globalmente. Sendo assim, um fluxo individual pode ser rejeitado mesmo se existir um caminho possível para roteá-lo, se a aceitação deste fluxo puder resultar num declínio geral do número de fluxos carregados pela rede. A determinação desta camada de controle de admissão de mais alto nível, com justiça para todos os fluxos que desejem utilizar-se da rede, é uma área de interesse atual.

O cálculo de rotas sujeito a múltiplas restrições de QoS em um fluxo é um problema complexo. Para algumas combinações de requisitos de QoS, o problema é NP-completo [11]. No entanto, alguns algoritmos foram propostos considerando requisitos de faixa de transmissão e de atraso, podendo ser incorporados aos mecanismos de roteamento. Na seção seguinte são apresentados três destes algoritmos.

2.7 Roteamento baseado em QoS para fluxos ponto-a-multiponto

O cálculo de rotas para a acomodação de fluxos ponto-a-multiponto com requisitos de QoS é um problema delicado, especialmente com a noção de um nível de controle de admissão mais alto adicionado. O dinamismo permitido pelo IP *Multicast* no grupo de receptores, aliado à heterogeneidade dos receptores complicam a questão. Considerando um esquema do tipo adotado pelo MOSPF (*Multicast* OSPF) [12, 13] a principal dificuldade diz respeito à escalabilidade. Para considerar QoS, o cálculo do caminho *multicast* no roteador deve levar em conta conhecimento não apenas da identidade das sub-redes onde membros do grupo estão presentes, mas também a identificação dos ramos na árvore existente. Ou seja, nesta versão do MOSPF, os roteadores precisam manter informação específica das fontes para cada grupo. Mudanças no grupo de destinos ou reservas realizadas por receptores resultam em informação sobre múltiplas árvores sendo distribuídas para todos os roteadores na rede.

Uma vez que a informação necessária está disponível, existem diversos algoritmos baseados na heurística para modificar incrementalmente árvores de distribuição existentes [11]. O cálculo de árvores compartilhadas baseadas no estilo de reserva compartilhado [2, 3, 4], no entanto, requer novos algoritmos.

3 Algoritmos de Roteamento

Os esquemas de roteamento pela origem (*source routing*) e roteamento salto-a-salto (*hop-by-hop routing*) estão presentes em grande parte das arquiteturas de roteamento para redes de transmissão de dados. O roteamento salto-a-salto é a forma mais comum de roteamento de propósito geral nas redes atuais, enquanto o roteamento pela origem é utilizado principalmente para o diagnóstico da rede e para realização de rotas com políticas especiais.

No esquema de roteamento pela origem, a rota de envio é calculada sob demanda na fonte da transmissão e listada explicitamente no cabeçalho do pacote. Os pacotes são encaminhados de acordo com a informação em seu cabeçalho. Uma vez que o cálculo é realizado para cada requisição, de maneira centralizada, o roteamento pela origem é flexível. No entanto, a fonte necessita de informação relativa a todos os enlaces envolvidos no caminho, e os pacotes possuem um cabeçalho longo. Existe também um tempo de cálculo durante a inicialização.

Já no roteamento salto-a-salto, os pacotes são encaminhados salto a salto a cada roteador. Cada nó possui uma tabela de roteamento com o próximo salto para todos os destinos, sendo em geral esta tabela atualizada periodicamente em resposta a mudanças na topologia da rede. Quando um pacote chega a um roteador, este precisa apenas checar na tabela qual a porta de saída para o destino desejado. Em relação ao roteamento pela origem, o cabeçalho dos pacotes pode ser menor, uma vez que não carrega a descrição completa da rota a ser seguida. O roteamento salto-a-salto pode ser implementado através de algoritmos completamente distribuídos.

Estes dois esquemas de roteamento possuem características que se encaixam nas necessidades do roteamento baseado em requisitos de QoS. Os requisitos de QoS das aplicações são bastante diversos, sendo difícil especificar um conjunto de parâmetros que sirva de base para a maioria das aplicações. Deste ponto de vista, o roteamento pela origem seria bastante apropriado, uma vez que o cálculo de rotas é realizado de acordo com cada fluxo, sob demanda. Por outro lado, o roteamento salto-a-salto permite a implementação de algoritmos distribuídos e possui as vantagens de menores sobrecarga de processamento e tempo gasto na inicialização.

Nesta seção, primeiramente é feita uma análise dos fatores de decisão na escolha da métrica ou conjunto de métricas adequado para a implementação de protocolos de roteamento baseados em QoS. Em seguida, são discutidas diferentes maneiras de combinarem-se as métricas. Esta combinação pode ser feita através da criação de uma medida que englobe as diferentes métricas, ou mediante a análise das métricas separadamente, por meio de algoritmos específicos.

3.1 Análise das métricas possíveis

A métrica escolhida pelo roteamento define basicamente a forma como a rede será representada. Esta decisão influirá não somente na complexidade do algoritmo de roteamento a ser implementado, mas também no conjunto de requisitos de QoS que tal esquema será capaz de suportar. Diversos aspectos devem ser levados em conta na escolha da métrica ou do conjunto de métricas que constituirão o fator de decisão de roteamento [11].

Qualquer métrica eleita deve permitir algoritmos eficientes para o cálculo das rotas, de forma a garantir a escalabilidade do protocolo de roteamento, indispensável quando leva-se em consideração o tamanho de uma rede como a Internet. De preferência, a complexidade dos algoritmos propostos para o roteamento baseado em QoS não deve ser maior que a dos algoritmos utilizados atualmente.

As métricas utilizadas devem refletir as características da rede, de forma a suportar requisitos de QoS básicos. Os parâmetros de QoS negociados deverão sempre ser mapeados na métrica (ou conjunto) usada pelo protocolo de roteamento na decisão de rotas. Desta forma, a escolha da métrica é fator determinante dos tipos de requisitos de QoS que poderão ser suportados. Por exemplo, se o atraso de transmissão e o custo forem as métricas, todos os requisitos terão que ser mapeados em termos de custo e atraso. Neste tipo de métrica, um requisito do tipo probabilidade de perda de pacotes não poderia ser suportado.

As métricas utilizadas devem ser disjuntas, sem qualquer informação redundante. Isto porque a informação redundante pode introduzir inter-dependência entre as métricas, o que torna impossível o cálculo de cada métrica independentemente. O cálculo de rotas torna-se mais complexo com a introdução de algoritmos recursivos entre as métricas.

3.2 Métrica única combinada

Algoritmos para o cálculo de rotas baseados em uma única métrica são bastante conhecidos, sendo utilizados pela maioria dos protocolos de roteamento atuais. A aproximação mais natural seria, então, o mapeamento de diferentes requisitos de QoS do usuário em uma única medida, através da qual a decisão possa ser tomada utilizando algoritmos tradicionais.

Para a definição de uma tal métrica, poderia-se criar uma função geradora, que receba como parâmetros de entrada um conjunto de requisitos de QoS do usuário e forneça na saída uma medida única. Por exemplo, poderia ser definida a função de saída para cada caminho como sendo proporcional à faixa de transmissão disponível e inversamente proporcional ao atraso e à probabilidade de perda de pacotes:

$$f(c) = \frac{B(c)}{D(c) \times L(c)},$$

onde c é um determinado caminho, $B(c)$ é sua banda passante, $D(c)$ o atraso de transmissão associado com este caminho e $L(c)$ a probabilidade de perda de pacotes correspondente. O caminho com o maior valor de $f(c)$ seria o melhor em termos desta medida.

A métrica única combinada, no entanto, não pode ser utilizada na prática, pois não contém informação suficiente para garantir que os requisitos de QoS do usuário estão sendo respeitados. Outro problema advém das diferentes regras de composição para parâmetros de QoS diversos. Para dois caminhos diferentes, a métrica do caminho total será a soma das duas individuais, se esta representar o atraso de transmissão por exemplo. Se, no entanto, a característica de QoS for a banda passante, a métrica do caminho total será a menor das duas. Desta forma, a função definida acima não preserva nenhuma das duas regras de composição.

3.3 Métricas múltiplas

A utilização de métricas múltiplas é capaz de modelar a rede de maneira mais precisa, porém o problema de encontrar um caminho dadas diversas restrições pode se tornar complexo [11]. A quantidade de diferentes requisitos de QoS é um fator de acréscimo da complexidade dos algoritmos. Além disso, os requisitos mais importantes dependem diretamente da aplicação em questão.

A complexidade do algoritmo será dependente da regra de composição das métricas, sendo três as regras de composição mais comuns: a aditiva, a multiplicativa e a côncava. Na aditiva, as métricas dos arcos componentes do caminho devem ser somadas para gerar a métrica do todo. Na multiplicativa, a métrica resultante é o produto das métricas componentes. Na do mínimo, também chamada de côncava, a métrica final é a mínima entre as todas as métricas dos trechos componentes da rota.

Os parâmetros mais comuns utilizados como requisitos de QoS do usuário são o atraso, a variação de atraso (*jitter*), o custo do serviço, a probabilidade de perda de pacotes e a banda passante. O atraso, a variação de atraso e o custo seguem a regra de composição aditiva. A banda passante segue a regra de composição do mínimo. A regra de composição para a probabilidade de perda é um pouco mais complexa. Seja c o caminho cuja métrica será calculada, $d(c)$ a métrica deste caminho, $d(i,j)$ a métrica associada ao arco entre o nó i e o nó j, a regra de composição da probabilidade de perda será como a seguir:

$$d(c) = 1 - ((1 - d(i,j) \times (1 - d(j,k)) \times ... \times (1 - d(l,m))).$$

No entanto, se a probabilidade de perda for transformada na probabilidade de transmissão com sucesso, a métrica equivalente possui uma regra de composição multiplicativa. Em [11] é realizado um estudo acerca da complexidade dos algoritmos para n métricas de forma aditiva e de forma multiplicativa, mostrando que estes são NP-completos. Além disso, é demonstrado que a combinação de n métricas aditivas com m métricas multiplicativas também só é solúvel por algoritmos NP-completos.

Como conseqüência, pode-se concluir que dos requisitos de QoS citados anteriormente, atraso, *jitter*, custo, probabilidade de perda de pacotes e banda passante, as únicas combinações tratáveis seriam da banda passante com outra das métricas. Isto porque a banda passante possui uma regra de composição côncava, enquanto que as outras são aditivas ou multiplicativas.

3.4 Roteamento pela origem e salto-a-salto com duas restrições

A seguir, são apresentados algoritmos para o cálculo de rotas dadas duas restrições, uma delas a faixa de transmissão, a outra podendo ser o atraso, o custo ou o *jitter*, uma vez que todas possuem regras de composição similares.

Na análise a seguir, a banda passante considerada deve ser entendida como a taxa de transmissão residual disponível para os novos fluxos em cada enlace. A métrica banda passante é então definida como a menor banda passante disponível em todos os arcos componentes de um determinado caminho.

O atraso possui dois componentes básicos, o atraso nas filas e o atraso de propagação propriamente dito. É importante notar que o atraso nas filas é função da menor banda passante e das características do tráfego. Desta forma, o atraso utilizado como métrica deve considerar apenas a componente devida ao tempo de propagação, uma vez que a correspondente ao tempo gasto nas filas já está considerada na métrica faixa de transmissão garantindo-se a independência entre as métricas combinadas.

A banda passante mínima de um determinado caminho pode ser entendida como a sua largura, enquanto que o atraso pode ser tomado como o comprimento do caminho. Desta forma, o problema do algoritmo de roteamento passa a ser encontrar um caminho na rede mais curto que determinada restrição, e com uma taxa de transmissão com um certo valor mínimo.

Para muitas aplicações, em especial as de tempo real, o atraso fim-a-fim é um dos requisitos de QoS mais importantes. A utilização da banda passante e do atraso de propagação como métricas constitui um compromisso entre a complexidade e a melhor caracterização dos requisitos de QoS do usuário.

3.4.1 Algoritmo para roteamento pela origem

Nesta seção, é apresentado um algoritmo para o esquema de roteamento pela origem com restrições de faixa de transmissão mínima e de atraso máximo admitido. Seja $G = (N, A)$ um grafo representante da rede, com N nós e E enlaces. A cada arco (i, j) são associados dois números reais, b_{ij} como a banda passante disponível e a_{ij} como o atraso de propagação. Para um arco (i, j) inexistente no grafo, será considerado que $b_{ij} = 0$ e $a_{ij} = \infty$. Dado um caminho $c = (i, j, k, ..., l, m)$, a largura do caminho, $w(c)$, será dada por $min[b_{ij}, b_{jk}, ..., b_{lm}]$ e seu comprimento será a soma dos atrasos de propagação, $l(p) = a_{ij} + a_{jk} + ... + a_{lm}$. Dadas duas constantes B para a faixa de transmissão e A para o atraso máximo aceitável, o problema se reduz a encontrar um caminho c entre i e m tal que $w(c) \geq B$ e $l(c) \leq A$.

O algoritmo consiste primeiro em descartar todos os arcos cuja $b_{ij} < B$. Todos os caminhos possíveis dentro do grafo resultante estarão respeitando o requisito de banda passante. O algoritmo deve então apontar um caminho que respeite o atraso máximo permitido, se ele existir. O algoritmo a seguir encontra um caminho entre o nó 1 e o nó m com banda passante não menor que B e atraso não maior que A. A_i é considerado o comprimento estimado do caminho entre o nó 1 e o nó i.

Passo 1: $\forall i, j : a_{ij} = \infty$ se $b_{ij} = 0$.
Passo 2: $L = \{1\}; \forall i, i \neq 1 : A_i = a_{1i}$.
Passo 3: Encontre $k \notin L \mid A_k = \min_{i \notin L} A_i$.
Se $A_k > A$, o caminho não pode ser encontrado e o algoritmo termina.
Se L contém o nó m, um caminho foi encontrado e o algoritmo termina.

$L := L \cup \{k\}$.

`Passo 4:` $\forall i \notin L : A_i := \min[A_i, A_k + a_{ki}]$.

`Vá para o passo 3.`

O passo 1 elimina os arcos cuja faixa de transmissão é menor que B. Os passos 2-4 encontram o caminho de atraso mínimo para o nó m através do algoritmo de Dijkstra. O algoritmo termina se o nó m for incluído no caminho ou se o atraso ultrapassar o limite antes de alcançar o nó m. Cada passo no algoritmo requer da ordem de N operações, e no pior caso os passos são repetidos $N-1$ vezes. Sendo assim o número de operações total no pior caso é $O(N^2)$.

3.4.2 Algoritmo para roteamento salto-a-salto

Uma vez que o esquema de roteamento salto-a-salto pré-calcula entradas na tabela para todos os destinos possíveis, este deve suportar todos os requisitos de recursos possíveis. Usualmente, os algoritmos de roteamento salto-a-salto calculam o melhor caminho para todos os destinos. Com uma única métrica, a melhor rota pode ser encontrada facilmente. Com mais de uma métrica, no entanto, o melhor caminho, com todos os seus parâmetros possuindo valores ótimos pode sequer existir. Sendo assim, é necessária a definição de uma prioridade entre as diferentes métricas.

A precedência entre a banda passante e o atraso é de certa forma dependente da aplicação. De uma maneira geral, entretanto, o atraso nas filas é mais dinâmico e sensível ao tráfego, sendo a banda passante freqüentemente mais crítica para a maioria das aplicações multimídias. No caso de não haver banda passante suficiente, o atraso nas filas, e provavelmente a taxa de perdas, serão altos. Em contraste, se o atraso de propagação não puder ser satisfeito, o atraso total será maior, mas o acréscimo será previsível e estável. Sendo assim, falhar em qualquer dos dois requisitos aumentará o atraso total do sistema, no entanto as conseqüências da falta de banda passante podem ser mais graves. Sendo assim, nos dois algoritmos a seguir é dada precedência à faixa de transmissão em detrimento do atraso de propagação.

Seja o nó 1 a fonte da transmissão e h o número de arcos saindo do nó fonte. Sejam $B_i^{(h)}$ e $A_i^{(h)}$ a largura e o comprimento do caminho mais curto e mais largo escolhido entre o nó 1 e o nó i com h saltos. Por convenção, $B_1^{(h)} = \infty$ e $A_1^{(h)} = 0$ para todo h. Os algoritmos para descoberta do caminho mais largo e mais curto podem ser produzidos pela adição de um teste de menor comprimento quando forem encontrados caminhos com igual largura. O algoritmo baseado em vetores distância é como a seguir:

`Passo 1:` `Início:` $\forall i \neq 1 : h = 0$ `e` $B_i^{(0)} = 0$

`Passo 2:` `Encontre` $K = \{k \mid w(1,...,k,i) = \max_{1\leq j\leq N}[\min[B_j^{(h)}, bji]]\}$, $i \neq 1$.

`Passo 3:` `Se` $\#K > 1$, `encontre` $k \in K \mid l(1,...,k,i) = \min_{1\leq j\leq N}[A_j^{(h)} + a_{ji}]$, $i \neq 1$.

`Passo 4:` $B_i^{(h+1)} = w(1,...,k,i)$ `e` $A_i^{(h+1)} = l(1,...,k,i)$.

`Passo 5:` `Se` $h \geq A$, `o algoritmo termina.`

`Se não,` $h = h + 1$. `Vá para o passo 2.`

O passo 2 encontra o caminho de maior faixa de transmissão do nó 1 para cada nó i. Se mais de um caminho com a maior faixa for encontrado, o passo 3 elege o caminho com o menor comprimento (atraso). O passo 4 atualiza a largura e o comprimento para o caminho mais largo e mais curto do nó 1 para cada nó i.

Seja o nó 1 a fonte de dados, B_i e A_i a largura e o comprimento do caminho mais curto e mais largo do nó 1 para o nó i. Por convenção, $B_1 = \infty$ e $A_1 = 0$. O algoritmo baseado no estado dos enlaces é o seguinte:

`Passo 1:` `Início:` $\forall i \neq 1 : L = \{1\}, B_i = b_{1i}$ `e` $A_i = a_{1i}$.
`Passo 2:` `Encontre` $K = \{k \mid k \in L, B_k = \max_{i \notin L}[B_i]\}$.
`Passo 3:` `Se` $\#K > 1$, `encontre` $k \in K \mid l(1, ..., k, i) = \min_{j \in k}[A_{(1,...,j,i)}]$.
$L := L \cup \{k\}$. `Se` L `contém todos os nós, o algoritmo termina.`
`Passo 4:` $\forall i \notin L$, $B_i := \max[B_i, \min[B_k, b_{ki}]]$.
`Vá para o passo 2.`

O passo 2 encontra os nós com a faixa de transmissão máxima entre os nós provisoriamente marcados. Se mais de um nó for encontrado, o passo 3 elege o nó mais curto e o marca permanentemente. O passo 4 atualiza os nós marcados provisoriamente em torno do novo nó permanentemente marcado.

Os dois algoritmos para o cálculo do caminho mais curto e mais largo são escaláveis. Nas duas versões o número de operações necessárias em cada iteração é proporcional às versões correspondentes do algoritmos para o cálculo do caminho mais curto. Sendo assim, o tempo de processamento dos dois algoritmos é igual ao dos algoritmos de caminho mais curto tradicionais.

4 A Relação entre o RSVP e o Roteamento

Dentre as diretivas de projeto adotadas para o RSVP, as mais importantes são o suporte a receptores heterogêneos, que devem poder realizar reservas de acordo com suas necessidades e a possibilidade de estas reservas, baseadas nos requisitos do usuário final, serem agregadas ao longo da rede, de forma a uma melhor utilização de recursos. São definidos diversos estilos de reserva, que regulam a maneira como as reservas são combinadas. O RSVP possui então características particulares importantes no que diz respeito ao roteamento. É um protocolo unidirecional, orientado ao receptor. Inicialmente, uma fonte transmite mensagens *path* através de uma árvore de distribuição *multicast*, para um grupo de receptores inicial. As reservas são realizadas posteriormente, pelo receptor interessado, através de mensagens *reserve* enviadas na direção da fonte através do caminho definido pelas mensagens *path*. As reservas são mantidas através da definição de um *soft state*, sendo descartadas se não atualizadas periodicamente.

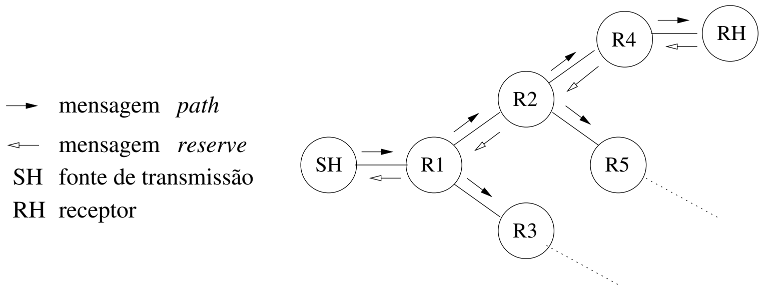


Figura 1: Funcionamento básico do RSVP.

4.1 Fluxos ponto-a-ponto x ponto-a-multiponto

O modelo de roteamento baseado em requisitos de QoS possui algumas implicações na sinalização. O QoS desejado para um fluxo deve ser especificado antes que um caminho possa ser calculado. Em contrapartida, no RSVP, um caminho precisa existir antes que uma reserva possa ser realizada. O RSVP utiliza-se de mensagens *path* para notificar os receptores de uma sessão das características do tráfego gerado pela fonte, e para estabelecer um estado para o caminhodo fluxo através dos roteadores. Com o roteamento baseado em QoS, a última função não é útil uma vez que o caminho do fluxo é calculado baseado apenas nas reservas dos receptores. Utilizando o modelo atual do RSVP com o roteamento baseado em QoS, o estabelecimento de um fluxo ponto-a-ponto seria realizado pela transmissão de uma mensagem *path* enviada da fonte ao receptor através de um canal de melhor esforço, por exemplo, e uma mensagem *reserve* enviada através de um caminho calculado de forma a acomodar os requisitos de QoS do fluxo por um roteador próximo ao receptor.

Com relação a fluxos ponto-a-multiponto, o caminho para um tal fluxo dentro das especificações atuais do RSVP é a árvore de distribuição do IP Multicast a partir da fonte. Para receber um fluxo, uma estação deve primeiramente aderir ao grupo *multicast*, e por conseqüência entrar na árvore *multicast* de uma determinada fonte. Para requisitar uma certa QoS para o fluxo, cada receptor deve enviar uma mensagem *reserve* na direção da raiz da árvore, reservando recursos nos enlaces e roteadores através do caminho. O conjunto de receptores de um fluxo *multicast* é dinâmico, e transparente à fonte de transmissão. O protocolo de roteamento ponto-a-multiponto dinamicamente mantém uma árvore por fonte, na qual as folhas são o conjunto atual de receptores. Este protocolo estabelece a árvore ponto-a-multiponto independentemente dos requisitos de QoS dos fluxos subseqüentemente roteados através desta. Sendo assim, não existe garantia de que uma folha terá sucesso na reserva de recursos para uma QoS desejada.

Uma solução para o roteamento ponto-a-multiponto baseado em requisitos de QoS é separar o fluxo de mensagens *path* do RSVP do fluxo de dados real de cada fonte. Neste caso, um protocolo de roteamento ponto-a-multiponto básico proveria a árvore de distribuição para as mensagens *path* para todo receptor participante do grupo. A recepção de uma mensagem *path* por um roteador iria disparar um algoritmo para, incrementalmente, determinar uma nova árvore para o fluxo de dados, ou seja, através da adição e remoção de ramos da árvore existente.

Outra solução poderia ser determinar a árvore *multicast* baseado na informação da própria mensagem *path*. Isto significa que a árvore é criada baseada no conjunto de receptores conhecido e assumindo a máxima alocação de recursos por todos os receptores. A heterogeneidade, por sua vez, pode ser conseguida através do ajuste das reservas baseado na informação das mensagens *reserve*. O dinamismo no grupo de receptores pode ser administrado, incrementalmente, recalculando as árvores toda vez que mensagens *path* de *refresh* forem geradas pela fonte. A desvantagem desta solução está no fato das árvores de distribuição *multicast* serem calculadas baseadas nos níveis de reserva de recursos máximos, tornando possível que árvores realizáveis não sejam computadas, quando seus níveis de reserva reais forem mais baixos.

Resumidamente, o modelo de sinalização definido pelo RSVP e o mecanismo de roteamento baseado em requisitos de QoS não se encaixam perfeitamente. A resolução destas incompatibilidades e o custo destas soluções são questões em aberto. Na seção seguinte é apresentada uma proposta recente para o gerenciamento dos caminhos considerando QoS com o protocolo RSVP.

4.2 Gerenciamento de caminhos com RSVP

Rotas baseadas em requisitos de QoS podem ser realizadas utilizando versões estendidas de alguns algoritmos de roteamento baseados no estado do enlace, como o OSPF [14]. No entanto, o uso de algoritmos de roteamento baseado em QoS possui algumas implicações a mais do que quando utilizam-se algoritmos convencionais.

Primeiramente, um mecanismo específico precisa ser usado para identificar fluxos com requisitos de QoS, de maneira que possam ser escalonados para o algoritmo de roteamento baseado em QoS correspondente. Nesta proposta, assume-se que o protocolo RSVP é o responsável por esta tarefa. Mais especificamente, as mensagens *path* do RSVP são utilizadas para disparar o roteamento baseado em QoS. Em segundo lugar, devido às variações na disponibilidade de recursos na rede, rotas entre os mesmos endereços de origem e destino, com os mesmos requisitos de QoS, podem variar, dependendo de quando o pedido de reserva é feito. No entanto, é preciso assegurar que tais mudanças não sejam sempre refletidas nos caminhos existentes, de forma a evitar oscilações nestes caminhos, limitando as mudanças a casos onde a configuração anterior torne-se inadequada.

Como resultado, alguma informação deve ser associada com uma rota baseada em QoS para determinar sua validade, ou seja, decidir se o algoritmo de roteamento deve ser acionado para determinar uma nova rota potencialmente melhor ou se a rota atual é adequada e deve ser mantida. Diz-se que um caminho está “marcado” quando o seu estado significa que o roteamento baseado em QoS não precisa ser acionado novamente, enquanto um caminho é considerado “desmarcado” em caso contrário. A questão passa a ser quando uma rota deve ser marcada ou desmarcada. Deseja-se ainda que este processo seja feito de acordo com o protocolo de reserva utilizado, o RSVP. Desta maneira, no gerenciamento a ser descrito, a marcação e a desmarcação de rotas deve ser realizada consistentemente com os estados *soft* do RSVP, exigindo mudanças mínimas nas regras deste protocolo.

É necessário observar que certas modificações são inevitáveis, principalmente na interface entre o RSVP e o protocolo de roteamento (*Routing Support for Resource Reservations* - RSRR) [15]. Especificamente, o roteamento baseado em QoS requer, além dos endereços de origem e destino, no mínimo, conhecimento sobre as características de tráfego do fluxo (TSpec), e possivelmente também o tipo de serviço e outros parâmetros [16]. Assume-se que o TSpec do transmissor, pelo menos, seja adicionado aos endereços de origem e destino na RSRR. No mecanismo de gerenciamento de caminhos apresentado tenta-se realizar o mínimo de mudanças na RSRR, e não se modificam os formatos das mensagens do RSVP, embora esta opção pudesse trazer benefícios [16].

Durante o processamento de mensagens *path*, o RSVP aciona o roteamento (baseado em QoS) para obter o próximo salto de envio destas mensagens. Como mencionado, o TSpec do transmissor é disponível para o roteamento, podendo o algoritmo selecionar o próximo salto ao longo do caminho para o destino que tenha chances de suportar a QoS especificada. Desta forma, o encaminhamento da mensagem *path* através deste próximo salto deve aumentar as chances de sucesso das mensagens de reserva a serem produzidas subseqüentemente, através do caminho reverso ao percorrido pelas mensagens *path*.

Dentro do contexto de roteamento nó-a-nó descrito, duas questões são levantadas em termos da marcação/desmarcação de rotas com requisitos de QoS. A detecção de ciclos fechados (*loops*) pode ser feita através de inconsistências nas rotas retornadas pelo roteamento em diferentes nós. Tais inconsistências são tipicamente transientes, mas é importante que a marcação de rotas não resulte na formação de ciclos fechados permanentes. Outra questão diz respeito ao pedido de uma nova rota no caso de falhas. Um exemplo é a falha de uma reserva devido à mensagem *reserve* ter chegado substancialmente depois de a rota baseada em QoS ter sido inicialmente selecionada. Outras falhas incluem defeitos usuais nos enlaces, e em geral é importante permitir ao roteamento baseado em QoS ser um tanto imune à falha e escolher uma melhor rota, se esta estiver disponível.

O gerenciamento de caminhos proposto trata das duas questões através de duas regras de projeto. Primeiramente, o que é marcado é um caminho tomado por um fluxo RSVP específico e não a rota, calculada pelo algoritmo de roteamento. Sendo assim, a marcação e desmarcação são operações de responsabilidade do RSVP, e como resultado, completamente independentes do algoritmo de roteamento específico sendo utilizado. Em segundo lugar, a marcação/desmarcação de caminhos é temporária, sendo submetida ao mecanismo de estado *soft* existente no RSVP. Ou seja, a marcação e desmarcação de caminhos é feita através dos mecanismos de temporização e *refresh* do estado da reserva existentes no RSVP. Estas mudanças são realizadas apenas em função da informação atualmente utilizada pelo RSVP.

4.2.1 Regras para o gerenciamento de caminhos

O estado de um caminho, como definido pelo RSVP, consiste em um *flag* utilizado para indicar se um caminho está atualmente marcado ou não. Um caminho marcado significa que o roteamento baseado em QoS não precisa ser acionado para obtenção de um próximo salto para o envio de uma mensagem *path* de *refresh*. As regras para a marcação/desmarcação de caminhos são:

- os caminhos são marcados durante o processamento de mensagens *path*;
- os caminhos são desmarcados quando:
  1. estados de caminho correspondentes são removidos (por estouro de temporizador ou explicitamente);
  2. alguns dos parâmetros recebidos nas mensagens *path* se modificam;
  3. um erro devido a falha no controle de admissão local é detectado após o recebimento de uma mensagem *reserve*;
  4. uma mensagem de erro é recebida;
  5. a notificação da falha de um enlace local pertencendo ao caminho é recebida.

As regras acima se traduzem em algumas pequenas modificações nas regras atuais do RSVP. Estas podem ser agrupadas em dois conjuntos, o primeiro deles, reunindo mudanças na interface entre o roteamento e o RSVP, como definida na RSRR [15]. A principal mudança diz respeito à inclusão do parâmetro TSpec na requisição de uma rota ao mecanismo de roteamento. O segundo conjunto de mudanças se refere às relacionadas com as regras de processamento das mensagens, em termos da marcação/desmarcação de caminhos:

Mensagens *Path* - Quando recebe a primeira mensagem *path*, o RSVP determina que não existe um estado do caminho para o fluxo. Este aciona então o roteamento baseado em QoS para obter o próximo salto através do "melhor"caminho disponível. Este próximo passo é armazenado como parte do estado do caminho com sua *flag* de marcação verdadeira.

Quando recebe uma mensagem *path* de *refresh*, o RSVP checa se as mudanças no estado do caminho são relevantes para o roteamento baseado em QoS. Se não há mudanças e o próximo passo atual está marcado, este será utilizado para enviar adiante a próxima mensagem *path*. Se o estado do caminho mudou ou o próximo salto atual está desmarcado, o RSVP consulta o roteamento novamente para obter um novo próximo salto a ser usado no envio da mensagem *path*. De forma semelhante, no momento em que uma mensagem *path* deve ser enviada, o RSVP checa se o próximo salto atual está marcado ou não. Se estiver, é utilizado no envio desta mensagem, se não, o roteamento é outra vez consultado.

Mensagens ***Reserve*** **-** A única mudança necessária é para o caso de a reserva de recursos falhar. De acordo com a especificação atual, uma mensagem RESV_ERR com o código de erro "falha de controle de admissão" é enviada no sentido dos receptores. No entanto, algum processamento adicional é necessário para a seleção de um melhor caminho, se este existir. Este começa com a desmarcação do próximo salto atual, e prossegue em uma de duas formas: a tentativa, ou não, de um reparo local no caminho com requisitos de QoS.

No caso de um reparo local ser tentado, o RSVP consulta novamente sua tabela de roteamento baseado em QoS local. Se um próximo passo diferente for retornado, ou seja, a reserva pode ter sucesso desta vez, então o reparo local é tentado através da marcação do novo próximo salto e o envio de uma mensagem *path* através da nova rota. Se no entanto o mesmo próximo salto for retornado, então o reparo local falhou. Neste caso, ou quando o reparo local não é tentado, o próximo passo atual é desmarcado no estado do caminho. Além disso, uma mensagem PATH_ERR é enviada no sentido da fonte com um código de erro de falha de caminho com QoS, identificando a indisponibilidade de recursos. A recepção desta mensagem provoca a desmarcação do caminho no roteador acima deste nó, na direção da fonte. Isto assegura que o roteamento será acionado quando a próxima mensagem *path* for gerada.

Notificação de Mudança de Rota - Uma notificação de mudança de rota é acionada quando o roteamento baseado em QoS detecta que um enlace utilizado atualmente por um caminho com requisitos de QoS falhou. Quando recebe tal notificação, o RSVP imediatemente desmarca o próximo salto atual. Como no caso da falha de uma reserva, o RSVP pode primeiramente tentar o reparo local, isto é, acionar o roteamento para um novo próximo salto. Se um novo próximo salto for retornado, o RSVP utiliza-o na substituição do próximo salto anterior, marca-o, e gera uma mensagem *path* na direção deste novo próximo nó. Se o roteamento responder que não há rota disponível para o destino, ou se o reparo local não for tentado, o RSVP envia na direção da fonte uma mensagem PATH_ERR com código de erro indicando a falha de enlace.

Mensagem de Erro de Caminho - Como mencionado anteriormente, a única modificação no processamento da mensagem PATH_ERR, é o reconhecimento do código de erro de falha de caminho com requisitos de QoS, e a desmarcação do próximo salto associado. Esta força o acionamento do roteamento baseado em QoS no processamento da próxima mensagem *path*.

4.3 Impacto do roteamento baseado em QoS no caminho de dados

A utilização do roteamento baseado em requisitos de QoS afeta apenas a escolha do caminho para os dados e não a maneira como estes dados são realmente enviados. Apesar disso, existe um aspecto importante a ser observado. Enquanto mensagens *path* são encaminhadas imediatamente através do próximo passo retornado pelo roteamento baseado em QoS, o mesmo não ocorre necessariamente com pacotes de dados. Isto se deve ao fato de poderem ocorrer ciclos fechados temporários nos caminhos com requisitos de QoS. O envio de mensagens *path* através de caminhos que contenham ciclos fechados provoca um impacto mínimo nos roteadores e é na verdade útil na detecção e eliminação de tais ciclos. Porém, dependendo do quão rápido os ciclos fechados possam ser solucionados, o envio de pacotes de dados deve ser melhor se retardado até que a ausência de ciclos fechados possa ser verificada.

Como conseqüência, é proposto que a modificação que resultará no envio de pacotes através do ciclo seja atrasada até o momento da recepção de uma mensagem *reserve*. Como discutido mais adiante, a recepção de uma mensagem *reserve* implica também que não existem ciclos fechados no caminho com requisitos de QoS.

4.3.1 Tratamento e prevenção de ciclos fechados

É importante detectar a ocorrência de ciclos fechados e desmarcar o caminho com requisitos de QoS selecionado (próximo passo), de forma que os ciclos possam ser eventualmente eliminados quando os nós tenham convergido para uma informação consistente.

A seguir, é feita uma descrição de como as regras de processamento descritas anteriormente são eficientes na detecção e eliminação de ciclos fechados. Esta é realizada através de dois cenários de ciclos fechados, que respectivamente ilustram a utilização da identificação de mudanças nos campos PHOP (*previous hop*) e IP TTL (IP *time-to-leave*) [17] para detecção de diferentes ocorrências de ciclos fechados.

Cenário 1:

Seja a situação descrita na figura 2. Assume-se que o roteador R2 (e possivelmente outros ao longo de RX...RY) possui informação de roteamento (métrica) inconsistente com a informação de R1. O roteador R1 identificou R2 como o próximo salto com a melhor QoS para o destino RH, baseado no fato de que o caminho com requisitos de QoS seria R1-R2-R3-...-RH. No entanto, devido à inconsistência na informação sobre as métricas, R2 identifica RX como o próximo salto no melhor caminho com requisitos de QoS para RH, sendo assim, a mensagem *path* enviada para RX eventualmente retorna a R2 através de RY. Como resultado, um ciclo fechado temporário é criado e ficará em efeito enquanto a informação de próximo salto incorreta for mantida marcada. No entanto, uma vez que a mensagem *path* de RY para R2 chega com um valor de PHOP diferente, este dispara a desmarcação da informação de próximo salto em R2. O roteamento baseado em QoS é, portanto, acionado novamente a próxima vez que R2 tiver que enviar uma mensagem *path* para o fluxo, de maneira que o ciclo fechado é eventualmente (uma vez que a informação em R2 se torne consistente) removido.

É importante notar que a rapidez na remoção do ciclo fechado é diretamente dependente dos valores dos temporizadores do RSVP e também da maneira como estes são sincronizados entre os diversos roteadores. Por exemplo, R2 pode receber uma mensagem *path* de RY logo

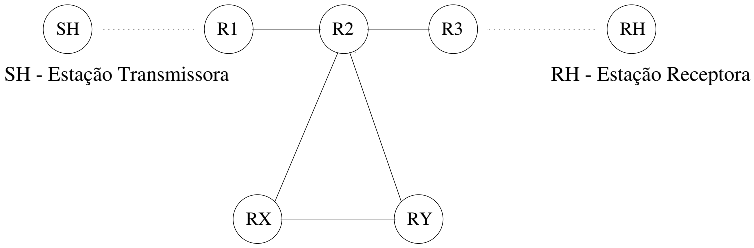


Figura 2: Ciclo fechando na máquina causadora do erro.

após ter recebido uma de R1. Isto não afeta a desmarcação do próximo salto (RX), mas significa que RY será mantido como o salto anterior para o fluxo, pelo menos até o próximo *refresh* de R1 (se a informação de roteamento de R2 estiver agora consistente, as mensagens *path* serão encaminhadas para R3 e não para RX, de maneira que RY não receberá mensagens *path* de *refresch*, e, assim, também não vai enviá-las). No entanto, isto não vai criar um ciclo fechado de reserva, uma vez que as mensagens *reserve* retornam para R2 de RX, R2 determina que RX não é o próximo salto correto para este fluxo.

Cenário 2:

O cenário 1 mostrou o caso onde o roteador que causou o ciclo fechado era também o nó onde o ciclo se fechava. Seja a situação descrita na figura 3. O cenário 2 considera o caso onde o ciclo não se fecha no roteador que o causou em primeiro lugar.

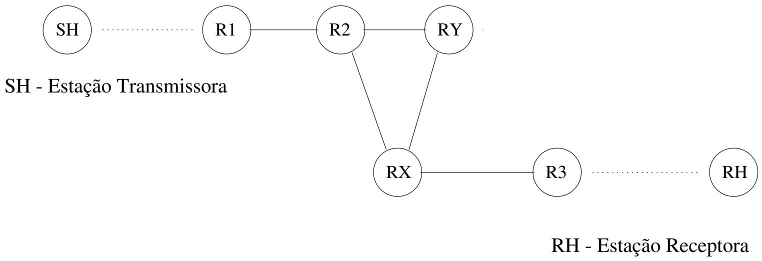


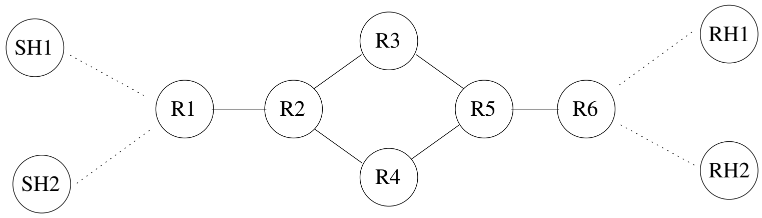
Figura 3: Ciclo fechando em uma máquina diferente.

Neste caso, o caminho com requisitos de QoS correto de SH para RH é SH...R1-R2-RX-R3...RH. No entanto, o roteador RX possui informação inconsistente, identificando RY como o melhor próximo salto. Isto eventualmente cria um ciclo fechando em R2, como RY identifica R2 como o melhor próximo salto para RH. Nesta situação, RX não identificará qualquer diferença no campo PHOP, e portanto este critério não pode ser utilizado para desmarcar a informação de próximo salto em RX. Porém, RX receberá mensagens *path* com diferentes valores de IP TTL, e este fato disparará a desmarcação do caminho de forma que quando a informação de roteamento em RX for atualizada, este poderá descobrir o próximo salto correto, R3.

4.3.2 Tratamento de condições de competição na alocação de recursos

O problema de condições de competição, ou seja, a existência de diversos caminhos competindo pelos mesmos recursos, é intrínseco à natureza distribuída das decisões do roteamento baseado em requisitos de QoS. O problema é agravado pelo atraso potencial entre a seleção de um caminho e o momento real da reserva dos recursos, ou seja, do recebimento de mensagens *reserve*. Conseqüentemente, é importante assegurar que o roteamento baseado em QoS e os mecanismos associados para o estabelecimento de caminhos com requisitos de QoS sejam suficientemente flexíveis para tratar tais condições de competição e mudar os caminhos se e quando for necessário.

Seja a topologia mostrada na figura 4, onde o enlace entre R2 e R3 pode acomodar apenas um dos dois fluxos em questão por vez, e que o segmento R1-R2-R3-R5-R6 está marcado para ambos os fluxos antes que RH1 ou RH2 tenham gerado qualquer mensagem *reserve*. Em seguida, RH1 é o primeiro a gerar uma mensagem *reserve*, e esta reserva obtém sucesso no caminho inteiro. Quando RH2 gera sua própria mensagem *reserve*, sua reserva terá sucesso em R6, R5 e R3, mas falhará em R2. De acordo com as regras de processamento descritas anteriormente, R2 irá então desmarcar seu estado de próximo salto além de enviar uma mensagem RESV_ERR na direção do receptor. É importante notar que os estados do caminho e da reserva permanecem intactos no segmento nesta direção.



SH1 envia uma mensagem path para estabelecer o fluxo para RH1.
SH2 envia uma mensagem path para estabelecer o fluxo para RH2.

Figura 4: Uma situação de competição por recursos.

Se em seguida R2 tentar o reparo local, vai acionar o roteamento baseado em QoS. Se o mesmo próximo salto, R3, for retornado, R2 determina que o reparo local falhou e envia uma mensagem PATH_ERR na direção do transmissor com o código de erro de falha do caminho com requisitos de QoS, com o valor indicativo de QoS requisitada não disponível. A recepção desta mensagem PATH_ERR dispara a desmarcação do estado de próximo salto em todos os nós na direção do transmissor, ou seja, em R1 e SH2. Isto por sua vez vai definir o acionamento do roteamento baseado em QoS no momento do próximo *refresh* do caminho. Isto assegura que, se em algum tempo, recursos se tornarem disponíveis em R4 e esta informação for passada ao roteamento, então o fluxo entre SH2 e RH2 será roteado através deste caminho, isto é, R2 identificará R4 como seu próximo passo para este fluxo. Como resultado, uma mensagem *path* será enviada na interface para R4 de maneira que mensagens *reserve* vindas de R5 serão eventualmente enviadas para R4, e a reserva de fluxo poderá obter sucesso. Um processo semelhante iria ocorrer se o roteamento baseado em QoS tivesse retornado R4 como o melhor próximo salto, quando R2 tentara realizar o reparo local.

Deve ser notado que R2 não enviou uma mensagem PATH_TEAR (de apagamento de um caminho) para R3 na ocasião da falha de reserva. Existem diversas razões para este procedimento. Primeiramente, o caminho através de R3 pode realmente ser o melhor atualmente disponível, e qualquer reserva feita com sucesso sobre este caminho até o momento provavelmente não deve ser desfeita (uma mensagem PATH_TEAR acarretaria isso) a menos que o receptor o deseje. Em segundo lugar, manter as reservas também evita o problema de ter as reservas nos enlaces de R5 para R6 e de R6 para RH2 utilizadas por outros fluxos, enquanto o fluxo de SH2 para RH2 está em processo de troca de caminhos. Em vez disso, as reservas existentes no segmento na direção do receptor (R5-R6...RH2) serão reutilizadas, e os estados do caminho e da reserva para o fluxo em R3 serão descartados com o estouro de um temporizador. Por outro lado, manter as reservas pode afetar o efeito do roteamento baseado em QoS. Isto porque este roteamento tipicamente não saberá que em alguns enlaces as reservas realmente disponíveis para o fluxo são maiores do que é atualmente informado. Porém, isto deve ser aliviado pelo fato de que os recursos informados são sempre imprecisos e também porque o roteamento retornará sempre o melhor caminho possível, mesmo que este não reconheça que os recursos necessários estão disponíveis neste caminho.

5 Conclusões

O roteamento baseado em requisitos de QoS deve ser considerado como um componente fundamental na provisão de novas classes de serviço na Internet, que possam dar suporte à variedade de novas aplicações multimídias que vêm surgindo. Este esquema possui diversos pré-requisitos a serem respeitados, sendo que cada um de seus componentes, como o cálculo de rotas, a escalabilidade e o controle administrativo, apresenta questões específicas a serem consideradas.

Os algoritmos para computação dos caminhos são um dos principais focos deste esquema de roteamento. A utilização de algoritmos considerando mais que duas restrições é considerada inviável atualmente, dada sua complexidade. Propostas considerando o cálculo aproximado de rotas com mais que duas restrições devem surgir, na tentativa de evitar este problema. No entanto, para a maior parte das aplicações, as necessidades em termos de QoS podem ser resumidas em duas restrições, como por exemplo o atraso de propagação e a taxa binária necessária. Neste caso, os algoritmos para o cálculo exato da melhor rota tornam-se simples, como apresentado neste trabalho, adicionando portanto pouca sobrecarga de processamento aos protocolos, um dos pré-requisitos para um roteamento dinâmico.

A escalabilidade do esquema de roteamento é um dos requisitos mais importantes, uma vez que freqüentemente a dimensão das redes em questão chega a milhares de nós. O algoritmo definido não deve portanto se tornar computacionalmente inviável com o número de nós. Os requisitos para o roteamento inter-domínios constituem um aspecto importante, a capacidade de introdução de restrições de acordo com determinadas políticas na fronteira entre dois domínios administrativos é indispensável.

A definição de protocolos de roteamento baseados em requisitos de QoS é, portanto, uma questão importante na definição de uma arquitetura de Serviços Integrados para a Internet. Algumas extensões aos protocolos de roteamento existentes estão sendo propostas, no entanto não foi definido qualquer novo protocolo. A definição de protocolos baseados nos algoritmos apresentados neste trabalho, ou em algoritmos com pequenas modificações, como por exemplo com a probabilidade de perda de pacotes como um dos requisitos de QoS, mostra-se uma área promissora para pesquisas futuras.

Referências

[1] R. Braden, D. Clark e S. Shenker, *RFC1633: Integrated Services in the Internet Architecture: An Overview.* Network Working Group, junho 1994.

[2] L. Zhang, S. Deering, D. Estrin, S. Shenker e D. Zappala, "RSVP: A new resource reservation protocol", *IEEE Network Magazine*, setembro 1993.

[3] L. Delgrossi, R. G. Herrtwichi, C. Vogt e L. C. Wolf, "Reservation protocols for internetworks: A comparison of ST-II and RSVP", in *Proceedings of Fourth International Workshop on Network and Operating System Support for Digital Audio and Video*, Lancaster, UK, novembro 1993.

[4] D. J. Mitzel, D. Estrin, S. Shenker e L. Zhang, "An architectural comparison of ST-II and RSVP", in *Proceedings of IEEE INFOCOM*, 1994.

[5] F. S. G. Oliveira, "Mobilidade na internet: Mecanismos básicos, otimizações e desafios", tech. rep., PEE-COPPE/UFRJ, 1997.

[6] B. Rajagopalan e R. Nair, *Quality of Service (QoS)-Based Routing in the Internet - Some Issues.* Bellcore and Ascon Nexen, outubro 1996. Internet-draft.

[7] M. Steenstrup, *RFC1477: IDPR as a proposed standard.* Network Working Group, julho 1993.

[8] S. Berson e L. Berger, *IP Integrated Services with RSVP over ATM.* ISI and Fore Systems, março 1997. Internet-draft.

[9] D. J. Mitzel, D. Estrin, S. Shenker e L. Zhang, "A study of reservation dynamics in integrated services packet networks", *IEEE Network Magazine*, pp. 18–26, jan-fev 1995.

[10] H. G. Perros e K. M. Elsayed, "Call admission control schemes: a review", *IEEE Communications Magazine*, novembro 1996.

[11] Z. Wang e J. Crowcroft, "Quality of service routing for supporting multimedia applications", *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 7, pp. 1228–1234, setembro 1996.

[12] J. Moy, *RFC1584: Multicast Extensions to OSPF.* Network Working Group, março 1994.

[13] J. Moy, *RFC1585: MOSPF: Analysis and Experience.* Network Working Group, março 1994.

[14] R. Guerin, S. Kamat, A. Orda, T. Przygienda e D. Williams, *QoS Routing Mechanisms and OSPF Extensions.* IBM, Technion and Fore Systems, março 1997. Internet-draft.

[15] D. Zappala, *RSRR: A Routing Interface for RSVP.* USC/ISI, novembro 1996. Internet-draft.

[16] R. Guerin, S. Kamat e S. Herzog, *QoS Path Management with RSVP.* IBM, março 1997. Internet-draft.

[17] E. Allison Mankin, F. Baker, B. Braden, S. Bradner, M. O'Dell, A. Romanow, A. Weinrib e L. Zhang, *Resource ReSerVation Protocol (RSVP) Version 1 Applicability Statement Some Guidelines on Deployment.* USC/ISI, Cisco Systems, Harvard, UUNET Technologies, Sun Microsystems, Intel Corporation e UCLA, março 1997. Internet-draft.