AGREGAÇÃO DE ESTADOS DO PROTOCOLO RSVP PARA FLUXOS *MULTICAST* *

PAULO CESAR S. VIDAL, OTTO CARLOS M. B. DUARTE
Grupo de Teleinformática e Automação - GTA
Programa de Engenharia Elétrica - COPPE
Universidade Federal do Rio de Janeiro
CP 68504 - CEP 21945-970 - Rio de Janeiro - RJ
{vidal,otto}@gta.ufrj.br

Resumo— **Reduzir o número de estados do protocolo RSVP armazenados nos roteadores é uma importante questão para tornar escalável a Arquitetura de Serviços Integrados na *Internet*. Este artigo apresenta um mecanismo de agregação de estados do protocolo RSVP para tráfego em trânsito utilizando túneis IP-sobre-IP, estendendo o modelo de [1] para fluxos *multicast*. Foram realizadas simulações para avaliar o mecanismo e os resultados mostram uma significativa diminuição dos estados.**

Abstract— **Reducing the number of RSVP protocol states stored in routers is a important issue in order to scale the Internet Integrated Services Architecture. This paper presents a state aggregation mechanism for the RSVP protocol for traffic in transit using IP in IP tunneling. The mechanism is an extension of the model [1] for multicast flows. The simulation results show a significant decrease of the number of states.**

I. INTRODUÇÃO

Muitas aplicações multimídias de tempo real estão sendo desenvolvidas para *Internet*, onde o modelo de encaminhamento de melhor esforço do protocolo IP (*Internet Protocol*) é inadequado. Portanto, existe a necessidade de fornecer às aplicações multimídias, classes de serviços com melhor qualidade de serviço em relação à largura de banda, atraso nas filas e perdas de pacotes. Buscando atingir estes objetivos, o IETF (*Internet Engineering Task Force*) definiu uma arquitetura de serviços integrados da *Internet* [2], onde são definidas classes de serviços que, se suportado pelos roteadores, podem fornecer um fluxo de dados com garantias de qualidade de serviço. O nível de qualidade de serviço fornecido por estas classes é programável por fluxo, de acordo com os requisitos das aplicações. Estes requisitos podem ser passados para os roteadores, usando o protocolo RSVP (*Resource Reservation Protocol*) [3]. Duas classes de serviço [4] tem sido especificadas para uso com o RSVP: a de serviço garantido [5] e de carga controlada [6].

Um fluxo é identificado pelo endereço *unicast* ou *multicast* e seu estado é armazenado nos roteadores entre a fonte e o destino do fluxo de dados. Existem quatro tipos de estados por fluxo usados na arquitetura de serviços integrados. Primeiro, o estado do escalonador que consiste de filas de serviço de diferentes tipos de tráfego para encaminhar o pacote. Segundo, o estado do classificador que é usado para atribuir, na chegada do pacote, uma fila no escalonador. Terceiro, o estado para os pedidos de reserva de recursos do protocolo RSVP. Por último, existem as tabelas de estados do protocolo de roteamento. A larga utilização dos serviços integrados, significa uma grande quantidade de estados armazenados nos roteadores e um grande processamento para gerenciá-los, podendo sobrecarregar os roteadores. Consequentemente, a técnica de agregação de fluxos tem sido utilizada para alcançar a escalabilidade necessária para fornecer os serviços integrados, reduzindo a quantidade de estados armazenados para fluxos *unicast* ou para fluxos *multicast*.

Em uma região de agregação (um sistema autônomo ou um conjunto de sistemas autônomos contíguos) podem existir dois tipos de tráfego: interno e trânsito. O tráfego em trânsito é composto de fluxos de dados que somente atravessam a região, sem iniciar ou terminar no seu interior. Por outro lado, o tráfego interno é aquele que não ultrapassa as fronteiras da região.

Muitos estudos tem sido realizados para alcançar a redução de estados e vários mecanismos de agregação tem sido propostos [1, 7, 8, 9, 10].

Em [1], o protocolo RSVP é utilizado em túneis IP-sobre-IP onde várias sessões individuais ponto-a-ponto podem ser mapeadas para uma sessão túnel entre os dois pontos da extremidade do túnel. A redução de estados é alcançada porque os roteadores interiores não processam mensagens RSVP fim-a-fim, somente as mensagens RSVP túnel que transportam informações agregadas dos fluxos fim-a-fim, deste modo os roteadores interiores somente armazenam estados das sessões túneis RSVP.

Em [7] é proposto uma arquitetura para reservas de recursos, onde as aplicações fazem pedidos de reserva através de um agente responsável pelo controle de admissão das reservas no seu domínio. O agente reúne vários pedidos de diferentes endereços fontes em um grupo que possui o mesmo prefixo do endereço de destino (mesmo domínio). Assim, o agente armazena uma entrada contendo o endereço destino da rede e a soma da largura de banda de todos os receptores.

*Este trabalho foi desenvolvido com suporte da UFRJ, FUJB, PROTEM-CC, CNPq e CAPES.

A arquitetura proposta em [8] mantém constante o número de estados do escalonador em todos os roteadores da região de agregação e do classificador nos roteadores interiores devido ao número fixo de classes de tráfego *unicast* configuradas na região de agregação. Não existe estados do protocolo RSVP para fluxos *unicast* armazenados nos roteadores interiores, entretanto para o tráfego *multicast*, os estados são armazenados em todos os nós.

Em [9] é realizado uma análise matemática da utilização da agregação em fluxos de serviço garantido que possuem características de tráfego idênticas e que percorrem a mesma rota até o destino. Verificou-se que quantidade de largura de banda necessária para reservar um grupo de fluxos pode ser menor que a soma das reservas individuais.

O artigo de Fukuda et al. [10] apresenta um algoritmo de agregação de fluxos para transporte de vídeo *multicast*. O algoritmo combina requisitos similares de qualidade de serviço dos usuários em um único requisito de qualidade de serviço, assim o número necessário de fluxos de vídeo que o servidor de vídeo necessita enviar pode ser diminuído. O total de largura de banda consumido na rede também é reduzido pelo compartilhamento do mesmo fluxo de vídeo por diversos usuários.

Este artigo apresenta um mecanismo de agregação de estados do protocolo RSVP que identifica os fluxos *multicast* que estão em trânsito em uma determinada região de agregação, agrupa seus estados entre os roteadores da borda, utilizando a técnica de tunelamento IP-sobre-IP. O mecanismo está baseado no artigo de Zhang e Terzis [1], adicionando funcionalidades para agregar fluxos *multicast*. Foram realizadas simulações para grupos esparsos, apresentando uma substancial redução dos estados nos roteadores interiores.

O artigo está organizado da seguinte forma. A Seção 2 apresenta uma visão geral do protocolo RSVP e a Seção 3 descreve o modelo de túneis RSVP para fluxos *unicast*. A Seção 4 descreve um mecanismo estendido para agregar fluxos *multicast*. A Seção 5 apresenta resultados da simulação do mecanismo de agregação. A Seção 6 mostra considerações finais e trabalhos futuros.

II. O PROTOCOLO RSVP

O protocolo RSVP [11] pode ser utilizado por uma estação para solicitar reserva de recursos, através da classe de serviço garantido ou de carga controlada, para fluxos de dados de uma aplicação. As reservas de recursos se aplicam tanto para fluxos *unicast* quanto *multicast*, adaptando-se dinamicamente às mudanças de grupo, bem como às mudanças de rota. É orientado a receptor, isto é, o receptor do fluxo de dados inicia e mantém as reservas de recursos usadas para o fluxo. O protocolo RSVP identifica uma sessão de um determinado fluxo, pela combinação do endereço destino, tipo de protocolo de transporte e número da porta de destino.

As principais mensagens usadas pelo RSVP são as mensagens *Path*, transmitidas pelos emissores, e as mensagens *Resv*, transmitidas pelos receptores. A mensagem *Path* instala o estado de roteamento reverso em cada roteador ao longo do caminho, isto é, o estado do caminho percorrido pelo fluxo é armazenado em todos os roteadores entre a fonte e o destino. A mensagem *Resv* conduz os pedidos de reservas para os roteadores ao longo do caminho entre os emissores e receptores.

O serviço garantido fornece um nível garantido de largura de banda e um limite de atraso fim-a-fim. Ele é planejado para aplicações com requisitos severos de encaminhamento em tempo-real, tais como, aplicações de áudio e vídeo que utilizam *buffers* de exibição (*playback*), e não aceitam a chegada de datagramas após seu tempo de exibição.

O serviço de carga controlada reserva largura de banda, mas não garante limites para o atraso. O pedido de reserva sendo aceito, o roteador oferece um serviço equivalente ao melhor esforço em uma rede de carga leve. A diferença em relação ao serviço de melhor esforço, é que o fluxo de carga controlada não se deteriora de forma visível quando a carga da rede aumenta.

III. AGREGAÇÃO EM TÚNEIS

A técnica de túneis IP-sobre-IP é um mecanismo bastante utilizado na *Internet* para transportar pacotes através de regiões da rede que não fornecem diretamente o serviço desejado. Entre os exemplos mais comuns do uso da técnica de túneis estão o roteamento IPv6 em roteadores que executam o IPv4, o roteamento *multicast*, o IP móvel e a Rede Virtual Privada.

Atualmente existem diversos protocolos para tunelamento IP-sobre-IP. Em [12] é descrito um método de tunelamento de pacotes do protocolo IPv4. O artigo de Perkins [13] apresenta um modo de reduzir o tamanho do cabeçalho interno usado em [12] quando o datagrama original não é fragmentado. O método de tunelamento genérico em [14] pode ser usado para encapsular pacotes IPv4 ou IPv6 dentro do pacote IPv6.

A técnica de tunelamento IP-sobre-IP pode ser usada para criar uma região de agregação de fluxos. Desta forma, os fluxos de múltiplas fontes que chegam ao roteador de entrada da região de agregação são encaminhados, de uma forma agregada, para o roteador de saída da região de agregação em um túnel IP-sobre-IP. Assim, todos os roteadores internos à região de agregação processam um fluxo agregado ao invés de processar os fluxos individualmente. Baseado neste princípio, o artigo de Zhang e Terzis [1] implementa um mecanismo de agregação de estados do protocolo RSVP sobre túneis para fluxos *unicast*.

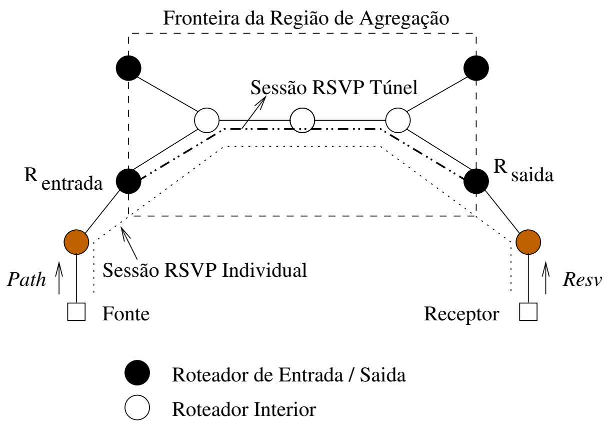

Figura 1. Modelo de túneis RSVP *unicast*.

O mecanismo de túneis RSVP está representado na figura 1, onde a fonte e o receptor de uma sessão RSVP individual (fim-a-fim) estão conectados pelo túnel. O roteador $R_{entrada}$ é um roteador de entrada do túnel que encapsula e encaminha os datagramas para o roteador R_{saida} através dos roteadores intermediários. O roteador R_{saida} é o ponto final do túnel que recebe, desencapsula e encaminha os datagramas baseado no cabeçalho original. Neste mecanismo existem duas sessões diferentes, uma sessão RSVP túnel e uma sessão RSVP fim-a-fim. A sessão RSVP fim-a-fim estabelece uma ligação com reserva de recursos entre a fonte e o receptor. A sessão RSVP túnel localizada entre o ponto $R_{entrada}$ e o ponto R_{saida} fornece reserva de recursos para um conjunto de sessões RSVP individuais configuradas no túnel.

As mensagens *Path* e *Resv* de uma sessão fim-a-fim que chegam nos pontos de extremidade do túnel são mapeadas para a sessão túnel correspondente à classe de tráfego agregada. Elas são encapsuladas e desencapsuladas na mesma maneira que pacotes IP, sendo acrescidas do cabeçalho externo especificando o ponto de entrada do túnel como fonte e o ponto de saída como destino. A figura 1 mostra o mapeamento de uma sessão individual, mas podem existir outras sessões fim-a-fim associadas à sessão túnel.

A associação entre as sessões individuais e a sessão túnel é feita através do objeto SESSION_ASSOC, inserido na mensagem *Path* individual no roteador $R_{entrada}$ ligando a sessão individual à sessão túnel. O objeto SESSION_ASSOC contém informações sobre a sessão individual (endereço destino, protocolo de transporte e número da porta de destino) e informações sobre a sessão túnel (endereço e porta do roteador de entrada do túnel).

A sequência de configuração de reservas no túnel é realizada através dos seguintes passos:

1. o roteador $R_{entrada}$ recebe uma mensagem *Path* individual e cria o estado do caminho (*Path*) da sessão RSVP fim-a-fim;
2. o roteador $R_{entrada}$ envia a mensagem *Path* fim-a-fim encapsulada sobre o túnel para o roteador R_{saida};
3. o roteador R_{saida} recebe a mensagem, desencapsula, cria o estado do caminho da sessão RSVP fim-a-fim e encaminha a mensagem original ao destino;
4. quando uma mensagem *Resv* fim-a-fim correspondente à sessão RSVP fim-a-fim chega no roteador R_{saida}, o roteador cria o estado de reserva, encapsula e envia a mensagem para o roteador de entrada;
5. o roteador de entrada recebe a mensagem *Resv*, desencapsula, cria o estado de reserva e mapeia a sessão RSVP fim-a-fim para uma sessão RSVP túnel;
6. após o mapeamento, o roteador de entrada envia uma mensagem PATH túnel correspondente à sessão RSVP túnel;
7. o roteador $R_{entrada}$ envia uma mensagem *Path* fim-a-fim encapsulada contendo o objeto SESSION_ASSOC associando à sessão RSVP fim-a-fim com à sessão RSVP túnel. O roteador R_{saida} registra a associação e remove o objeto antes de encaminhar a mensagem para o destino;
8. o roteador R_{saida} recebe a mensagem PATH túnel e envia uma mensagem RESV túnel com o objeto RESV_CONFIRM para o interior da região de agregação, solicitando reserva de recursos;
9. se algum roteador interior não aceitar a reserva, então ele envia uma mensagem RESV_ERROR para o roteador de saída. Assim, quando a mensagem RESV túnel chega no roteador $R_{entrada}$, significa que o pedido foi aceito pelos roteadores interiores;
10. depois de receber a mensagem RESV túnel, o roteador de entrada envia uma mensagem RESV_CONFIRM túnel para o roteador R_{saida} como confirmação da reserva no túnel.

Após a configuração da reserva, o roteador de saída recebe periodicamente as mensagens *Resv*. Entretanto, pode ocorrer que uma mensagem *Resv* fim-a-fim possa causar uma mudança nos parâmetros do estado de reserva individual e na sessão túnel associada. Se a mudança é para aumentar a reserva, então o R_{saida} envia uma mensagem RESV túnel, incluindo o objeto RESV_CONFIRM, e o processamento é igual ao do recebimento de uma mensagem *Resv* de uma nova sessão fim-a-fim. Se a mudança é para diminuir a reserva de recursos, então uma mensagem RESV túnel é enviada sem o objeto RESV_CONFIRM, com objetivo de atualizar os estados de reservas nos roteadores interiores e os roteadores de entrada.

Quando o roteador de entrada recebe os pacotes de dados, ele verifica se existe reserva de recursos para o fluxo de dados correspondente à sessão RSVP fim-a-fim contida no pacote. Se não existir reserva para o pacote, então o pacote é encapsulado e enviado para o roteador de saída. Caso tenha reserva para o fluxo, o pacote recebe o encapsulamento IP e o encapsulamento UDP (*User Datagram Protocol*), pois as sessões individuais são diferenciadas pela porta UDP

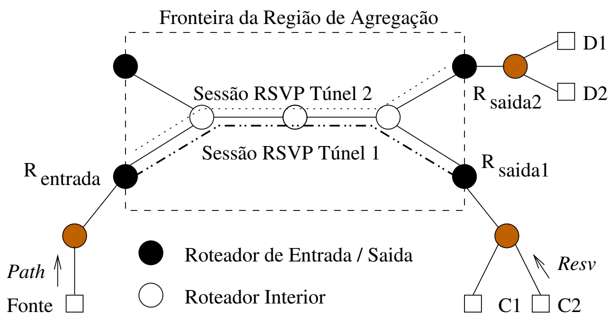

Figura 2. Mecanismo de túneis RSVP *multicast*.

da fonte. Outros componentes da sessão túnel, como endereço IP da fonte e destino, porta UDP destino são iguais para todas as sessões individuais.

A porta UDP fonte é escolhida pelo roteador de entrada quando é estabelecido o estado do caminho inicial para uma nova sessão túnel. A porta UDP fonte associada com a nova sessão é então transportada para o roteador de saída através do objeto SESSION_ASSOC.

Os roteadores intermediários não conhecem as mensagens RSVP fim-a-fim, e manipulam somente as mensagens RSVP geradas pelos pontos finais do túnel. Portanto, existe uma redução estados nos roteadores intermediários que gerenciam somente sessões agregadas por classe de tráfego.

IV. MECANISMO DE AGREGAÇÃO

O mecanismo de agregação de estados de fluxos *multicast* apresentado neste artigo é aplicado ao serviço de carga controlada e agrega o tráfego em trânsito na região de agregação usando a técnica de tunelamento.

O mecanismo é uma extensão do modelo de túneis [1] e as principais mudanças introduzidas foram o mapeamento de sessões RSVP individuais em sessões RSVP túneis, a difusão de uma mensagem *Path* para vários roteadores de saída e a marcação dos pacotes reservados através do campo TOS (*Type of Service*) ou o campo DS (*Differentiated Services*) do cabeçalho externo do pacote IP.

A associação de uma sessão RSVP *multicast* para várias sessões RSVP túneis está ilustrado na figura 2. A figura mostra uma árvore *multicast* composta de uma fonte e quatro receptores (C1,C2 no domínio C e D1, D2 no domínio D), todos pertencentes à sessão *multicast*. No exemplo foram criadas duas sessões RSVP túneis localizadas entre o roteador $R_{entrada}$ e os roteadores R_{saida1} e R_{saida2} para fornecer a reserva de recursos sobre cada túnel à sessão *multicast* RSVP fim-a-fim.

O roteador $R_{entrada}$ recebe uma mensagem *Path*, verifica através do protocolo BGP (*Border Gateway Protocol)* [15] para qual roteador ou quais roteadores de saída a mensagem *Path* encapsulada será enviada. Tendo o conhecimento do(s) roteador(es) de saída, o roteador de entrada inicia o processo de configuração da reserva em cada túnel composto pelas seguintes etapas:

1. o roteador $R_{entrada}$ envia uma mensagem *Path* fim-a-fim encapsulada para o roteador R_{saida1} e o roteador R_{saida2};
2. cada roteador de saída recebe a mensagem, a desencapsula, cria o estado da sessão RSVP fim-a-fim e encaminha a mensagem original ao destino;
3. quando uma mensagem *Resv* correspondente à sessão RSVP fim-a-fim chega no roteador R_{saida1} ou R_{saida2}, o roteador a encapsula e envia para o roteador de entrada;
4. o roteador de entrada, ao receber uma mensagem *Resv*, faz a associação da sessão RSVP individual com a(s) sessão(ões) RSVP túnel(eis) correspondente(s) à mensagem *Resv* recebida;
5. o roteador de entrada envia uma mensagem PATH túnel para cada sessão RSVP túnel associada à sessão RSVP individual;
6. o roteador $R_{entrada}$ envia uma mensagem *Path* fim-a-fim contendo o objeto SESSION_ASSOC associando à sessão RSVP fim-a-fim à sessão RSVP túnel. O roteador R_{saida} registra a associação e remove o objeto antes de encaminhar a mensagem para o destino;
7. cada roteador R_{saida} recebe a mensagem PATH túnel e envia uma mensagem RESV túnel, com o objeto RESV_CONFIRM solicitando reserva de recursos nos roteadores interiores da região de agregação;
8. se algum roteador interior não aceitar a reserva, então ele envia uma mensagem RESV_ERROR para o roteador de saída. Assim, quando a mensagem RESV túnel chega no roteador $R_{entrada}$, significa que o pedido foi aceito pelos roteadores interiores;
9. o roteador de entrada envia uma mensagem RESV_CONFIRM túnel para o roteador R_{saida}, após ter recebido a mensagem RESV túnel.

Os pacotes de dados referentes à sessão *multicast* que chegam ao ponto $R_{entrada}$ e que possuem reservas de recursos associada às duas sessões túneis podem ser marcados com a classe de tráfego no campo TOS (*type of service*) do cabeçalho externo do pacote IP, ou pelo campo DS (*Differentiated Services*). O campo DS é obtido pela renomeação do campo TOS, no caso do IPv4, ou do campo *Traffic Class*, no caso do IPv6 [16]. Após a marcação, são geradas cópias (uma para cada túnel), sendo cada pacote é encapsulado e enviado para o endereço do roteador de saída. Quando chegam nos roteadores borda de saída são desencapsulados e enviadas para o destino.

V. RESULTADOS

Nesta seção são apresentados os resultados obtidos por simulações do mecanismo de agregação *multicast*. O objetivo da simulação é testar a funcionalidade do mecanismo e comparar a quantidade de estados indi-

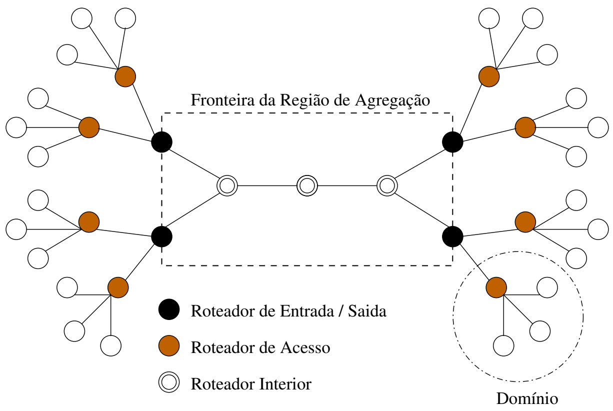

Figura 3. Topologia utilizada.

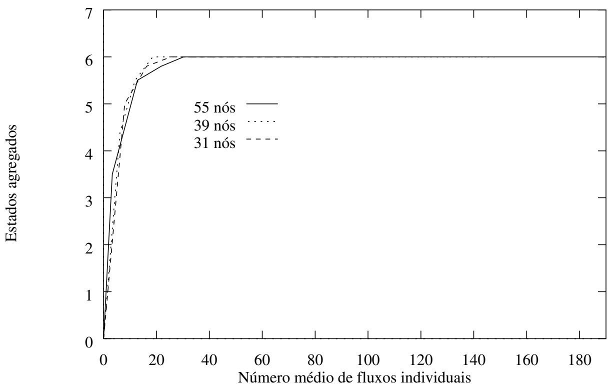

Figura 4. Estados agregados X Estados individuais.

viduais do RSVP clássico com a quantidade de estados agregados nos roteadores de entrada e saída da região de agregação.

Para obtenção das medidas foi utilizado o simulador de redes de computadores *ns* (*Network Simulator*) [17]. Uma extensão ao pacote básico do *ns* foi implementada para fornecer o suporte ao mecanismo de agregação sendo constituída de dois módulos: RSVP clássico (fluxos individuais) e RSVP agregado (fluxos agregados). O módulo RSVP clássico foi obtido de [18]. Os dois módulos foram implementados usando as linguagens C++ e OTcl.

O módulo RSVP agregado desenvolvido executa as seguintes funções:

- processamento das mensagens PATH Túnel, RESV Túnel, RESV_ERROR túnel e RESV_CONFIRM Túnel;
- mapeamento das sessões individuais para sessões túneis;
- encapsulamento e desencapulamento dos pacotes;
- manutenção das tabelas de estados agregados.

A topologia da simulação está representada na Figura 3. Os nós de cor preta representam os roteadores de entrada ou saída da região de agregação. O conjunto de um nó de acesso (cor cinza) ligado aos nós finais (cor branca) representam um domínio que utiliza o protocolo RSVP clássico.

Em cada domínio, os nós finais são receptores e fontes de tráfego para grupos *multicast*. Na simulação, um grupo é composto de uma fonte e até 8 (oito) receptores. Uma sessão *multicast* fica caracterizada pelo par (Fonte, Grupo). Os nós na região de agregação são interligados por enlaces de 10Mbps, o enlace entre o nó de acesso e o nó borda tem 2Mbps. Os nós finais estão conectados por enlaces de 1Mbps.

Cada nó final envia fluxos CBR (*Constant Bit Rate*) à taxa de 256Kbps. Antes de um fluxo RSVP ser transmitido, as mensagens *Path* são enviadas em intervalos aleatórios. Quando os receptores recebem as mensagens, eles que disparam automaticamente um pedido de reserva através da mensagem *Resv*. Cada nó fonte envia uma mensagem *Path* para cada grupo à que pertence, e o número de grupos por fonte pode variar de dois a oito. A fonte, o receptor e o número de receptores por grupo são variáveis aleatórias uniformente distribuídas. Os receptores de um grupo não podem pertencer ao mesmo domínio da fonte e nem pertencer ao domínio vizinho do domínio da fonte, pois observando a figura 3, os domínios vizinhos estão ligados ao mesmo roteador de entrada/saída e entre os domínios vizinhos não se formam túneis.

Foram executadas sessões de simulações com topologias de 31, 39 e 55 nós que resultaram em valores médios. Na figura 4 é apresentado um gráfico que relaciona o número médio de estados agregados (equivalente ao número médio de túneis) e o número médio de fluxos individuais sendo reservados por roteador de entrada/saída. Um fluxo representa uma sessão *multicast*. Pode-se observar que o número de estados agregados permanece constante a partir de um determinado instante, enquanto fluxos individuais reservados aumentam. Nas topologias simuladas o número máximo de túneis no roteador de entrada/saída é 6 (seis).

Na comparação entre estados individuais e estados agregados é avaliado o nível de agregação que pode ser alcançado na região de agregação, chamado de Fator Médio de Agregação.

O Fator Médio de Agregação é a relação entre o número médio de estados individuais e número máximo de estados agregados (número de túneis) por roteador de entrada/saída. A Tabela 1 mostra o valor médio do Fator de Agregação para 3(três) topologias. O número de sessões é o número máximo de sessões configuradas com reservas de recursos na região de agregação.

VI. CONSIDERAÇÕES FINAIS

Os mecanismos de agregação de fluxos fornecem uma redução no número de estados necessários para a armazenar as informações das reservas de recursos do protocolo RSVP, aumentando a escalabilidade do seu emprego na *Internet*.

O mecanismo de agregação *multicast* apresentado neste artigo reduz o número de estados armazenados

TABELA 1
FATOR MÉDIO DE AGREGAÇÃO

Fator Médio de Agregação				
Nr total de nós	Nr de nós por domínio	Nr de sessões	Nr médio de estados individuais	Fator
31	3	120	90	15:1
39	4	200	148	24:1
55	6	240	189	31:1

e o número de mensagens processadas na região de agregação porque os roteadores interiores manipulam somente as mensagens RSVP túneis, geradas pelos pontos finais do túnel. O fator médio de agregação nos roteadores de entrada/saída foi utilizado como parâmetro para mostrar uma significativa redução de estados do protocolo RSVP nas topologias simuladas.

A identificação do tráfego em trânsito com reserva de recursos pelo mecanismo de agregação é importante sob o ponto de vista de gerência de rede, pois o administrador da rede pode avaliar o impacto deste tipo de tráfego na região de agregação.

Como trabalhos futuros podemos citar a realização de simulações em topologias maiores com diferentes tipos de tráfego e uma avaliação do tempo de processamento associado com o encapsulamento e desencapsulamento de pacotes nos pontos finais do túnel.

VII. REFERÊNCIAS

[1] A. Terzis e L. Zhang, "Reservations for Aggregate Traffic: Experiences from an RSVP Tunnels Implementation", *6th International Workshop on Quality of Service (IWQoS'98)*, maio de 1998.

[2] R. Braden, D. Clark e S. Shenker, "Integrated Services in the Internet Architecture: An Overview", *RFC 1633*, julho de 1994.

[3] L. Zhang, S. Deering, D. Estrin, S. Shenker e D. Zappala, "RSVP: a new Resource ReserVation Protocol", *IEEE Network Magazine*, pp. 8-18, setembro de 1993.

[4] J. Wroclawski, "The Use of RSVP with Integrated Services", *RFC 2210*, setembro de 1997.

[5] S. Shenker, C. Partridge e R. Guérin, "Specification of Guaranteed Quality of Service", *RFC 2212*, setembro de 1997.

[6] J. Wroclawski, "Specification of Controlled-Load Network Element Service", *RFC 2211*, setembro de 1997.

[7] O. Schelén e S. Pink, "Resource Reservation Agents in the Internet", *8th International Workshop on Network and System Support for Digital Audio and Video*, 1998.

[8] S. Berson e S. Vincent, "Aggregation of Internet Integrated Services State", *6th International Workshop on Quality of Service (IWQoS'98)*, maio de 1998.

[9] J. Schmitt, M. Karsten, L. Wolf e R. Steinmetz, "Aggregation of Guaranteed Service Flows", *7th International Workshop on Quality of Service (IWQoS'99)*, junho de 1999.

[10] K. Fukuda, N. Wakamiya, M. Murata e H. Miyahara, "On Flow Aggregation for Multicast Video Transport", *6th International Workshop on Quality of Service (IWQoS'98)*, maio de 1998.

[11] R. Braden, L. Zhang, S. Berson e S. Jamin, "Resource Reservation Protocol (RSVP) - Version 1 functional specification", *RFC 2205*, setembro de 1997.

[12] C. Perkins, "IP Encapsulation within IP", *RFC 2003*, outubro de 1996.

[13] C. Perkins, "Minimal Encapsulation within IP", *RFC 2004*, outubro de 1996.

[14] A. Conta e S. Deering, "Generic Packet Tunneling in IPv6 Specification", *RFC 2473*, dezembro de 1998.

[15] T. Bates, R. Chandra, D. Katz e Y. Rekhter, "Multiprotocol Extensions for BGP-4", *Internet-Draft*, janeiro de 1998.

[16] K. Nichols, S. Blake, F. Baker e D. L. Black, "Definition of the differentiated services field (DS field) in the IPv4 and IPv6 headers", *RFC 2474*, dezembro de 1998.

[17] NS, "Network Simulator - NS (version2)", *http://www-mash.cs.berkeley.edu/ns/*.

[18] M. Greis, "RSVP/ns: An Implementation of RSVP for the Network Simulator ns-2", *http://titan.cs.uni-bonn.de/~greis/rsvpns/rsvpns.ps.gz*, 1999.