

ANÁLISE DE QUALIDADE DE SERVIÇO NA INTERNET USANDO RESERVA DE RECURSOS

Ana L.P. Schmidt e Rosa M.M. Leão
UFRJ, Coppe/Sistemas,
Cx.P. 68.511, Rio de Janeiro, RJ 21945-970, Brazil

Resumo— Aplicações de voz e vídeo são conhecidas por possuírem requisitos de tempo real. A rede Internet atualmente oferece um único tipo de serviço: *best effort*, onde nenhuma garantia é dada em termos de parâmetros temporais. A utilização da Internet como rede de suporte para aplicações de voz e vídeo fez com que fossem propostas novas arquiteturas de forma a prover um serviço com garantias. Nestas propostas, está previsto que os roteadores terão como novas funções fazer a classificação e o escalonamento de pacotes segundo a Qualidade de Serviço (QoS) requisitada pela aplicação e um mecanismo para fazer a gerência de recursos deve ser implementado na rede. O objetivo deste trabalho é realizar uma série de testes em um ambiente onde o protocolo RSVP atua como gerenciador de recursos e os roteadores implementam duas políticas de escalonamento de pacotes: Class Based Queue (CBQ) e Weighted Fair Queue (WFQ). Utilizamos sequências de vídeo com características diferentes e a alocação de recursos foi feita tanto baseada na taxa média quanto na taxa de pico da aplicação. Os resultados dos experimentos mostram que uma QoS diferenciada pode ser obtida na presença do ambiente estudado.

I. INTRODUÇÃO

A rede Internet atualmente oferece um único tipo de serviço: *best effort*. Neste serviço nenhuma garantia é dada ao usuário em termos de requisitos temporais (retardo ou jitter máximo garantido). O aparecimento de aplicações com requisitos de tempo real, como a voz e o vídeo, fez com que fossem propostas novas arquiteturas para os serviços a serem disponibilizados na Internet. Duas soluções foram propostas: os Serviços Diferenciados (DiffServ) [1] e os Serviços Integrados (IntServ) [2].

Na arquitetura definida para os Serviços Diferenciados, os roteadores são classificados em roteadores da periferia de uma região e internos a uma região. Os roteadores da periferia são responsáveis pela classificação dos pacotes segundo a QoS solicitada, devendo manter informações por fluxo. Já os roteadores internos à região realizam o escalonamento dos pacotes baseados em um campo do cabeçalho que indica a classe do pacote. Desta forma não é necessário que os roteadores internos à região armazenem/processem informações de QoS por fluxo, o que aumenta a escalabilidade desta proposta.

O trabalho de A.L.P. Schmidt foi apoiado por bolsa de mestrado do CNPq.

Este trabalho é parcialmente suportado pelo CNPq/ProTeM.

Mecanismos para fazer o gerenciamento de recursos, ou seja, alocar os recursos e realizar o controle de admissão e policiamento do tráfego ao longo de um determinado caminho, ainda são um tópico aberto para pesquisa. Em [3] é feita uma proposta de um mecanismo para gerenciamento de recursos para esta arquitetura. O mecanismo é hierárquico: um primeiro nível é responsável pelos recursos dentro de um domínio administrativo (ou *Autonomous System (AS)*) e um segundo nível deve gerenciar os recursos entre domínios. A reserva de recursos e o controle de admissão dentro de um domínio são feitos para cada fluxo individualmente. O protocolo sugerido para o gerenciamento de recursos dentro de um domínio é o *Resource Reservation Protocol (RSVP)* [4].

Na arquitetura proposta inicialmente para IntServ [2] é previsto que cada roteador ao longo do caminho entre a origem e o destino, deva manter as informações de QoS de cada fluxo e deva realizar a classificação e o escalonamento dos pacotes por fluxo. Esta solução pode trazer problemas de escalabilidade em uma rede de dimensões da Internet. Uma possível solução para tornar a proposta IntServ mais escalável é apresentada em [5]. Naquele trabalho é sugerido que a rede seja dividida em regiões e os roteadores sejam classificados em roteadores de entrada/saída da região e roteadores internos à região. A classificação de pacotes, o controle de admissão e o policiamento do tráfego são funções realizadas somente pelos roteadores de entrada/saída da região. A gerência de recursos é realizada de forma distinta para os tráfegos *unicast* e *multicast* usando o protocolo RSVP.

Nas diversas propostas [6], [3], [5] apresentadas como solução para oferecer um serviço com QoS na Internet, o protocolo RSVP é um forte candidato a se tornar o padrão para realizar o gerenciamento de recursos. Encontramos alguns trabalhos na literatura onde o protocolo RSVP é testado ou avaliado em conjunto com políticas de escalonamento. Em [3] são avaliados, através de simulação, os mecanismos de reserva propostos e a QoS obtida pelas aplicações. Dois tipos de tráfego são usados nos experimentos: fontes ON/OFF e fontes FTP. A política usada pelos roteadores para o escalonamento dos pacotes é fila com prioridade. Em [7] são realizados diversos testes com o protocolo RSVP cu-

jos objetivos são: avaliar o tempo para que uma reserva seja feita, avaliar o overhead devido às mensagens do protocolo e o overhead devido ao escalonamento de pacotes. Resultados de testes com o objetivo de verificar a correção da implementação do protocolo realizada pela Intel, MCI e CISCO podem ser encontrados em [8]. Dos trabalhos encontrados na literatura, não temos conhecimento de nenhum que avalie, através de testes, a QoS obtida por aplicações de vídeo quando estas usam o RSVP em conjunto com políticas de escalonamento implementadas nos roteadores.

O objetivo deste trabalho é apresentar resultados de experimentos realizados em um ambiente onde o protocolo RSVP atua como gerenciador de recursos e os roteadores implementam duas políticas de escalonamento: Class Based Queue [9] e Weighted Fair Queue [10].

Nos nossos testes consideramos como aplicações sequências de vídeos com características diferentes. Experimentos foram realizados considerando que os recursos foram alocados baseados na taxa média e na taxa de pico da aplicação. Foi gerado um tráfego *melhor esforço* bastante alto para competir pelos recursos juntamente com as aplicações de vídeo.

Cabe ressaltar que o objetivo maior do nosso trabalho foi estudar a qualidade de serviço obtida para aplicações de vídeo em um ambiente com gerência de recursos, onde variamos os seguintes aspectos: tipo de alocação de recursos e políticas de escalonamento. Não pretendemos aqui discutir a aplicabilidade de IntServ e DiffServ na Internet, o que pretendemos mostrar em relação às duas abordagens, é que em ambas está previsto um ambiente semelhante ao que usamos nos nossos testes. Os resultados dos experimentos realizados mostraram que o ambiente estudado oferece uma qualidade diferenciada para as aplicações com requisitos de tempo real.

Este artigo está organizado como descrito a seguir. Na seção II apresentamos as principais funcionalidades do protocolo RSVP. Descrevemos o ambiente utilizado para a realização dos testes na seção III e apresentamos os principais resultados dos experimentos realizados na seção IV. A seção V resume os resultados do trabalho, apresenta algumas conclusões e indica possíveis trabalhos futuros.

II. DESCRIÇÃO GERAL DO PROTOCOLO RSVP

O objetivo do protocolo RSVP é transportar as informações de QoS requisitadas por um fluxo de tráfego e realizar a reserva de recursos ao longo de um determinado caminho na rede. Para que uma reserva seja efetuada é necessário o estabelecimento de uma sessão RSVP entre a fonte e um destino (sessão *unicast*) ou diversos destinos (sessão *multicast*).

Para fornecer a QoS requisitada pelo fluxo, o RSVP atua em conjunto com os módulos apresentados na figura 1. O classificador de pacotes determina a classe de cada pacote segundo a QoS requisitada pelo fluxo. O escalonador implementa uma política para servir os pacotes na fila de saída. Durante o estabelecimento de uma reserva, o módulo de controle de admissão é consultado para verificação da existência de recursos disponíveis de forma a fornecer a QoS desejada pelo fluxo. O módulo de roteamento fornece informações sobre as melhores rotas entre as fontes e os destinos da rede assim como informações sobre mudanças na topologia.

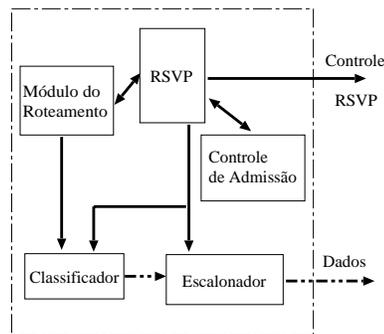


Fig. 1. Arquitetura dos roteadores que implementam o RSVP

O RSVP utiliza duas mensagens principais para que uma sessão seja estabelecida com uma certa QoS: a mensagem *PATH* e a mensagem *RESV*. A mensagem *PATH*, enviada pela fonte, é transmitida para todos os destinos. Esta mensagem contém a QoS requisitada pela fonte. Uma mensagem *RESV* é enviada por cada um dos destinos para a fonte utilizando uma transmissão ponto-a-ponto. Ela contém a QoS requisitada pelo receptor. Quando uma mensagem *RESV* é recebida por um roteador, o módulo de controle de admissão é consultado para verificar se é possível realizar a reserva. Se a reserva é realizada com sucesso em todos os roteadores ao longo do caminho, quando a mensagem *RESV* chega na fonte, a sessão é estabelecida.

O estado de uma determinada reserva é mantido pelos roteadores através do *soft state*. Quando ocorre uma mudança de membros de um grupo *multicast*, uma alteração em uma determinada rota ou o caminho para alcançar um destino não existe mais, o módulo RSVP dos roteadores deve ser informado para atualizar o *soft state*. A forma como o RSVP mantém atualizado o *soft state* é através do envio periódico de mensagens *PATH* e *RESV* pela fonte e o destino, respectivamente. Se após um certo período de tempo um roteador não receber, por exemplo, uma mensagem *RESV* de um certo destino, ele libera os recursos que haviam sido alocados para aquele destino.

III. AMBIENTE PARA A REALIZAÇÃO DE TESTES

Nos nossos experimentos configuramos algumas máquinas com sistema BSD UNIX com a arquitetura apresentada na Figura 1. Estas máquinas atuaram como os roteadores da rede. Nesta arquitetura, o módulo que implementa o protocolo RSVP atua em conjunto com outros quatro módulos: o módulo de roteamento, o classificador, o escalonador e o controle de admissão. Para executar a função dos últimos três módulos, usamos um pacote de software chamado ALTQ [11].

O pacote ALTQ define um novo módulo para fazer o controle de saída de pacotes no *kernel* do sistema operacional BSD UNIX. Tradicionalmente, o módulo que faz o controle da saída, implementa a disciplina FIFO. O pacote ALTQ foi especificado para dar suporte a uma variedade de disciplinas de atendimento a filas com componentes tais como: estratégias de escalonamento, mecanismos para descarte de pacotes e para gerência de buffers. No escopo do nosso trabalho utilizamos dois escalonadores que encontram-se implementados no ALTQ: CBQ e WFQ.

O escalonador CBQ é baseado na criação de classes que possuem uma hierarquia. Cada classe possui a sua própria fila para a qual é atribuída uma prioridade e uma parte da capacidade da banda. O CBQ, além do escalonador de pacotes, possui um módulo estimador da capacidade sendo usada por cada classe. A capacidade usada por cada classe é estimada periodicamente. Caso uma classe ultrapasse o limite alocado para ela, ela é marcada como *sobrecarregada*. O escalonador determina o próximo pacote a ser servido a partir da prioridade e do estado das classes (*sobrecarregada* ou não). Classes com mesma prioridade são escalonadas segundo a disciplina *Weighted Round Robin*.

Suponha que tenhamos três classes definidas. Se todas as classes estiverem recebendo tráfego, aquela que exceder a capacidade reservada, terá seus pacotes descartados, caso o buffer esteja cheio. Se uma determinada classe parar de receber pacotes, a capacidade do canal que ficar ociosa, será dividida igualmente entre as duas outras classes.

O princípio de funcionamento da política WFQ é baseado na criação de filas por classe de serviço onde as filas são servidas em *round-robin*. O tempo alocado a cada uma das filas é proporcional ao peso atribuído a ela.

O classificador de pacotes implementado no ALTQ, após o estabelecimento com sucesso de uma sessão RSVP, atribui uma classe ao fluxo da sessão (o que determina a fila de saída dos pacotes) baseado na qualidade de serviço recebida do módulo RSVP e nas informações recebidas do módulo de roteamento.

Uma implementação bastante simples do controle de ad-

missão é encontrada no pacote ALTQ. Cada vez que uma nova sessão é requisitada pelo RSVP, é verificado se a banda disponível (banda total menos a banda alocada para as outras sessões) é suficiente para atender a nova demanda.

IV. RESULTADOS DOS EXPERIMENTOS

Nesta seção estão reunidos os resultados dos experimentos realizados com aplicações de vídeo usando o ambiente de reserva definido nas seções precedentes.

Utilizamos uma aplicação fonte localizada no laboratório LAM (Coppe/UFRJ) e as aplicações destino ficavam no laboratório LAND (NCE/UFRJ). Estes dois laboratórios estão interconectados através de um anel FDDI. A arquitetura da Figura 1 foi implementada na máquina que faz o roteamento dos pacotes que saem do laboratório LAM (roteador *lamrouter*).

Foi desenvolvida uma aplicação *multicast* para enviar pacotes a partir de uma máquina definida como a fonte e receber/analisar pacotes nas máquinas destino. A função da aplicação fonte é gerar pacotes a partir de sequências de vídeo codificadas em MPEG. A aplicação no destino faz a coleta de dados e a geração de estatísticas relacionadas a dois parâmetros: o jitter e a perda de pacotes. Para o jitter são calculadas a sua distribuição, média e variância. Em relação a perda de pacotes calculamos a proporção de pacotes perdidos e a função de probabilidade de massa (FPM) do número de pacotes consecutivos perdidos (ou seja a probabilidade do número de pacotes consecutivos perdidos ser igual a um valor x). Esta última medida sendo a FPM da variável aleatória que indica o número de pacotes consecutivos perdidos entre duas chegadas. O motivo para o cálculo desta última medida é que através dela podemos avaliar se as perdas ocorrem em rajadas ou não, o que não é possível de ser analisado com a fração de pacotes perdidos. Com estas duas medidas podemos caracterizar de forma bastante precisa o processo de perdas na rede.

De forma a criar uma situação de tráfego bastante intenso no roteador onde a reserva é efetuada, foi usada uma ferramenta para geração de tráfego desenvolvida no nosso grupo de pesquisa [12]. Com isso, o roteador que implementa os mecanismos para garantir a QoS, tinha sempre um tráfego *best effort* muito alto em concorrência com o tráfego que observamos a QoS obtida.

A informação passada pelo RSVP ao CBQ, no estabelecimento da sessão, é a banda requisitada pela aplicação. Outros parâmetros de configuração do CBQ, como prioridade, devem ser configurados de forma estática nos roteadores. Configuramos o CBQ com uma classe *vídeo* onde foi alocada 20% da banda total e maior prioridade que a classe *default* que possui o restante da banda. A

classe *RSVP* é criada dinamicamente pelo CBQ, herdando as mesmas características da classe *vídeo*, a partir do estabelecimento de uma sessão RSVP. A banda alocada é correspondente a QoS requisitada pela mensagem RSVP.

O escalonador WFQ do ALTQ só possui um único parâmetro para ser configurado: a banda requisitada pela aplicação. Configuramos o escalonador criando uma classe para o tráfego das sequências de vídeo (onde a banda alocada corresponde à banda requisitada pela aplicação) e uma outra classe para o restante do tráfego. Utilizamos três sequências de vídeo nos testes. As sequências de vídeo usadas tem características diferentes. A primeira sequência, chamada Discovery [13], tem taxa média de 1.4 Mbps, taxa de pico de 3.8 Mbps e uma duração de 30s. A segunda sequência, Tangram2 [13], representa o filme de uma palestra que foi codificado em MPEG no nosso laboratório. A taxa média da sequência é de 986 Kbps, a taxa de pico é de 3.4 Mbps e a duração é de 4min30s. A terceira sequência é o filme Starwars [13], sua taxa média é igual a 468 Kbps, a taxa de pico é de 5.5 Mbps e o filme tem duração de 85min.

Foram realizados para todas as sequências testes em um ambiente onde existiu o controle de QoS e testes em um ambiente onde não foi implementado nenhum controle. No ambiente sem controle o roteador *lamrouter* faz escalonamento FIFO e não existem os módulos para fazer o controle de tráfego. Apresentamos a seguir os principais resultados dos experimentos.

A. Resultados da Sequência Discovery

Os seguintes experimentos foram realizados com esta sequência: (1) Ambiente controlado com escalonamento CBQ onde a banda requisitada corresponde à taxa média da sequência; (2) Ambiente controlado com escalonamento WFQ onde a banda requisitada corresponde à taxa média da sequência; (3) Ambiente controlado com escalonamento WFQ onde a banda requisitada corresponde à taxa de pico da sequência; (4) Ambiente sem controle.

Nas Figuras 2 e 3 temos a distribuição do jitter no caso do ambiente sem controle e quando a política WFQ é usada e a alocação é feita pela taxa de pico. A partir dos gráficos podemos observar que quando a alocação é feita pela taxa de pico, os valores de jitter são sempre inferiores a 0.02 ms. Já para o caso do ambiente sem controle temos valores de jitter que são uma ordem de grandeza superiores aos do ambiente com controle.

Um segundo parâmetro medido foi a função de probabilidade de massa (FPM) do número de pacotes consecutivos perdidos. Podemos observar na figura 4 a FPM para o caso do ambiente sem controle. Neste gráfico

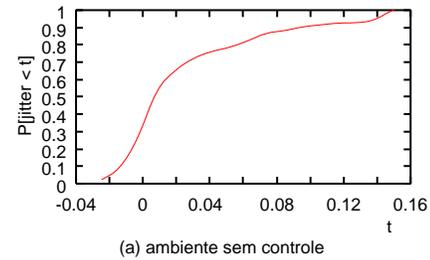


Fig. 2. Distribuição do jitter - ambiente sem controle

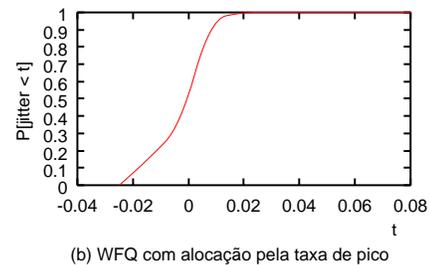


Fig. 3. Distribuição do jitter- WFQ com alocação pela taxa de pico

vemos que podem existir rajadas de perdas que variam de 3 a 13 pacotes com probabilidade da ordem de $10e-02$. No nosso caso usamos a configuração do MPEG onde um quadro I é gerado após 11 quadros P e B. Como temos perdas de até 14 pacotes consecutivos, certamente quadros I seriam perdidos, o que significaria uma qualidade ruim da aplicação no destino. No caso do CBQ e WFQ alocando pela taxa média e WFQ alocando pela taxa de pico, o número total de pacotes perdidos foi insignificante: 7, 2 e 0, respectivamente, não havendo perda de mais de 2 pacotes consecutivos. Com esse número de pacotes perdidos, certamente o usuário no destino teria uma boa qualidade para o vídeo sendo transmitido.

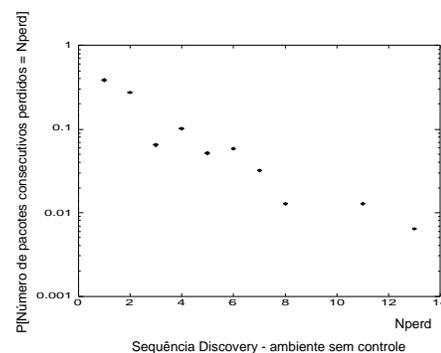


Fig. 4. FPM do número de pacotes consecutivos perdidos

A tabela abaixo apresenta um resumo dos resultados obtidos nos testes com a sequência Discovery. Podemos

observar através dos resultados da tabela que a qualidade de serviço fornecida para a aplicação é bastante superior no caso do ambiente com controle. A média do jitter é diminuída em até 4 ordens de magnitude (WFQ com alocação pela taxa de pico) quando comparada com a média obtida para o ambiente onde nenhuma reserva é feita para a aplicação. Observando o percentual de perda, verificamos uma queda acentuada, podendo-se chegar a uma taxa de perda igual a zero quando a alocação é feita pela taxa de pico.

Par. QoS	CBQ (1)	WFQ (2)	WFQ (3)	S.C.(4)
Méd. jitter(ms)	2.178e-4	1.368e-4	-4.528e-5	1.956e-1
Var. jitter(ms)	6.450e-4	4.916e-4	8.199e-5	1.833e-2
perda(%)	0.77	0.22	0	46.67

B. Resultados da Sequência *Tangram2*

Os seguintes experimentos foram realizados com esta sequência: (1) Ambiente controlado com escalonamento CBQ onde a banda requisitada corresponde à taxa média da sequência; (2) Ambiente controlado com escalonamento CBQ onde a banda requisitada corresponde à taxa de pico da sequência; (3) Ambiente controlado com escalonamento WFQ onde a banda requisitada corresponde à taxa média da sequência; (4) Ambiente controlado com escalonamento WFQ onde a banda requisitada corresponde à taxa de pico da sequência; (5) Ambiente sem controle.

Nos experimentos realizados para o ambiente sem controle tivemos rajadas de perdas de 1 até 14 pacotes como no caso da sequência *Discovery*. Como usamos a mesma configuração do MPEG, certamente perderíamos quadros I, o que tornaria a qualidade do vídeo no destino ruim. Quando usamos o CBQ com alocação da banda usando a taxa média, tivemos perdas consecutivas de até 14 pacotes, não existindo melhora em relação ao ambiente sem controle. Já quando a reserva é feita para a taxa de pico, ainda temos perdas acima de 3 pacotes porém com probabilidade da ordem de 10^{-3} , o que reduz as chances de haverem perdas de quadros I.

Para o caso da política WFQ, quando a alocação é feita pela taxa média temos no máximo 4 pacotes consecutivos perdidos (com probabilidade da ordem de 10^{-3}) e quando a alocação é feita pela taxa de pico temos até 3 pacotes consecutivos perdidos (com probabilidade da ordem de 10^{-3}), o que representa uma melhora significativa em relação ao ambiente sem controle.

As tabelas abaixo mostram um resumo dos experimentos. Podemos observar uma diminuição do jitter de uma ordem de grandeza quando o ambiente com controle é usado. O percentual de perda de pacotes também possui uma redução significativa. Como era esperado, podemos observar que a QoS obtida quando a reserva é feita

usando a taxa de pico é melhor do que quando a reserva é feita usando a taxa média.

QoS	CBQ(1)	CBQ(2)	WFQ(3)	WFQ(4)	SC(5)
Méd.jit.	1.33e-2	5.30e-3	7.15e-3	3.39e-3	3.15e-2
Var.jit.	2.28e-3	3.73e-4	1.06e-3	1.68e-2	2.04e-3
%perda	29.0	13.7	17.0	9.24	48.6

C. Resultados do Filme *Starwars*

Descreveremos os resultados de dois experimentos feitos com esta sequência: (1) Ambiente controlado com escalonamento CBQ onde a banda requisitada corresponde à taxa de pico da sequência; (2) Ambiente sem controle.

Através da distribuição do jitter, observamos que com a política CBQ o jitter apresenta valores superiores a 0.02 ms com probabilidade muito baixa. Já para o caso do ambiente sem controle temos valores de jitter até 0.16 ms.

A FPM do número de pacotes consecutivos perdidos para o ambiente sem controle está apresentada na Figura 5. Podemos notar que temos perdas de até 30 pacotes consecutivos. Como observado anteriormente, isso aumenta as chances de perda de quadros I, significando uma qualidade ruim da aplicação no destino. No caso do ambiente com controle temos uma perda igual a zero pacotes.

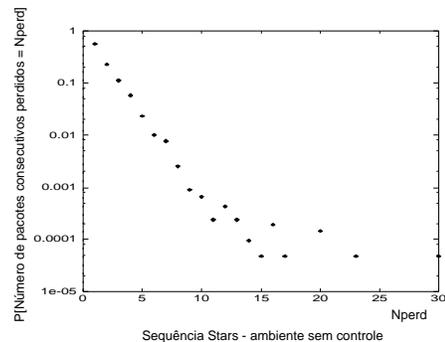


Fig. 5. FPM do número de pacotes consecutivos perdidos

A tabela a seguir apresenta um resumo dos parâmetros medidos. Podemos observar uma diminuição do jitter de três ordens de grandeza e o percentual de perda que atinge o valor zero quando é usado um ambiente controlado.

Par. QoS	CBQ (1)	Sem Controle (2)
Méd. jitter(ms)	1.812e-6	9.771e-3
Var. jitter(ms)	2.017e-4	9.626e-4
Perda(%)	0	22.65

V. RESUMO DOS RESULTADOS

Neste trabalho foi possível avaliar a QoS de aplicações de vídeo em presença de um ambiente onde o protocolo

lo RSVP atua como gerenciador de recursos e onde são implementadas duas políticas de escalonamento: CBQ e WFQ. Dois parâmetros de QoS foram avaliados: o jitter e a perda de pacotes. Para o jitter calculamos a sua distribuição, média e variância. No caso da perda, obtivemos a FPM do número de pacotes consecutivos perdidos e a proporção de pacotes perdidos. Três sequências de vídeo com características diferentes foram consideradas. A primeira delas, Discovery, possui *burstiness* igual a 2.7 e duração de 30s, a segunda, Tangram2, tem *burstiness* igual a 3.4 e duração de 4min30s e a terceira, Starwars, possui *burstiness* igual a 11.7 e tem duração de 85min.

Os resultados dos testes realizados com a sequência Discovery mostraram uma melhora significativa do jitter e da perda de pacotes quando é usado o ambiente com controle. A média do jitter cai até 4 ordens de grandeza e a perda de pacotes pode chegar até zero. Através da FPM do número de pacotes consecutivos perdidos, verificamos que não existem perdas de mais de 2 pacotes consecutivos quando o ambiente é controlado. O fato de obtermos melhora significativa na QoS qualquer que seja a alocação (média ou pico), pode ser consequência da curta duração da sequência.

Para a sequência Tangram2, observamos que quando a política CBQ é usada não obtivemos melhora significativa no número de pacotes consecutivos perdidos. Isto ocorre apesar de haver uma redução de até 35% na proporção de pacotes perdidos, o que mostra que somente este último parâmetro não é suficiente para avaliar a QoS da aplicação no destino. Já quando a política WFQ é usada, a partir da FPM do número de pacotes consecutivos perdidos, podemos verificar que perdas de no máximo 3(4) pacotes ocorrem com probabilidade da ordem de $10e-03$ quando a alocação é feita pela taxa de pico(média).

No caso do filme Starwars, só foram realizados experimentos com a política CBQ onde a alocação é feita usando a taxa de pico. Os resultados mostram melhora significativa no parâmetro jitter e na perda de pacotes. A partir da FPM do número de pacotes consecutivos perdidos obtida para o ambiente sem controle, podemos notar que temos perda de 5 até 30 pacotes consecutivos com probabilidade que varia de $10e-02$ até $10e-05$. Esta perda cai para zero quando o ambiente com controle é usado.

Analisando os resultados dos experimentos podemos observar que o jitter possui valores bem abaixo do mínimo definido em padrões internacionais para aplicações de vídeo [14] tanto para o ambiente sem controle quanto para o ambiente controlado. Isto deve-se ao fato de realizarmos os testes dentro da rede da UFRJ onde as aplicações passam por um número reduzido de roteadores.

Contudo, o que é importante de ser observado é a redução de algumas ordens de grandeza na média do jitter quando o ambiente controlado é usado.

Em todos os experimentos onde usamos as duas políticas, CBQ e WFQ, podemos notar que a política WFQ apresenta um desempenho superior à política CBQ em termos da QoS fornecida para a aplicação. Adicionando-se a esse resultado o fato da política WFQ possuir uma complexidade bastante inferior (no que diz respeito à configuração dos parâmetros e ao algoritmo implementado) à complexidade da política CBQ, isto torna a WFQ uma forte candidata a ser usada em um ambiente com serviços diferenciados.

Como possíveis extensões desse trabalho podemos citar a realização de testes dos diversos estilos de reserva em conjunto com um algoritmo para estimar a capacidade a ser reservada para os fluxos agregados (por exemplo, um algoritmo para cálculo de capacidade efetiva). Este último devendo ser incorporado ao módulo de controle de admissão.

REFERÊNCIAS

- [1] Y. Bernet et al. Differentiated Services, February 1999. Internet Draft - draft-ietf-diffserv-framework-02.txt.
- [2] R. Braden, D. Clark, and S. Shenker. Integrated Services in the Internet Architecture: an Overview, June 1994. Internet RFC 1633.
- [3] A. Terzis, L. Wang, J. Ogawa, and L. Zhang. A Two-Tier Resource Management Model for the Internet. In *Proceedings of Globecom '99*, volume 3, pages 1779–1791, Rio de Janeiro, Brazil, December 1999.
- [4] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin. Resource ReSerVation Protocol - Version 1 Functional Specification, September 1997. Internet draft.
- [5] S. Berson and S. Vincent. Aggregation of Internet Integrated Services State. In *Proceedings of IWQoS 98*, May 1998.
- [6] Y. Bernet et al. A Framework for use of RSVP with DiffServ Networks, September 1999. Internet draft, work in progress.
- [7] A. Neogi and T. Chiueh. Performance Analysis of an RSVP-Capable Router. *IEEE Network*, pages 56–63, September/October 1999.
- [8] M. Baugher and S. Jarar. Test Results of the Commercial Internet Multimedia Trials. *ACM SIGCOMM*, 28(1):16–27, January 1997.
- [9] S. Floyd and V. Jacobson. Link-Sharing and Resource Management Models for Packets Networks. *IEEE/ACM Transactions Networking*, 3(4), August 1995.
- [10] A. Demers, S. Keshav, and S. Shenker. Analysis and Simulation of a Fair Queueing Algorithm. In *Proceedings of SIGCOMM'89*, pages 1–12, Austin, TX, September 1989.
- [11] K. Cho. A Framework for Alternate Queueing: Towards Traffic Management by PC-UNIX Based Routers. In *Proceedings of USENIX'98*, New Orleans, LA, June 1998.
- [12] R. M. M. Leão, E. de Souza e Silva, and S. C. de Lucena. A Set of Tools for Traffic Modelling, Analysis and Experimentation. In *Proceedings of TOOLS'2000*, Schaumburg, IL, March 2000. To appear.
- [13] Sequências de vídeo podem ser encontradas em <ftp://ftp.land.ufrj.br/pub/rsvp/seqs>.
- [14] Raif O. Onvural. *Asynchronous Transfer Mode Networks-Performance Issues*. Artech House, 2 edition, 1995.