

Heurísticas para Reserva e Negociação de Recursos Intra-Domínio em WebLabs com Suporte a DiffServ

Lucio Agostinho¹, Luis F. Faina²,
Eliane G. Guimarães³, Paulo R. S. L. Coelho¹, Eleri Cardozo¹

¹ Faculdade de Engenharia Elétrica e de Computação - FEEC
Universidade Estadual de Campinas - UNICAMP

² Faculdade de Computação - FACOM
Universidade Federal de Uberlândia - UFU
38400-902 - Uberlândia - MG - CP 593

³ Centro de Pesquisas Renato Archer
13083-970 - Campinas - SP

Abstract. The performance evaluation in networks is necessary to ensure end-to-end QoS for applications. The experiments in WebLabs place a large strain on internal network and the competition between several streams reduces the overall QoS offered for the experiments. The case studies show that the DiffServ use is not enough to provide appropriate QoS for the network. Therefore, the allocation resources monitoring by Bandwidth Broker (BB) is essential for efficient control of bandwidth distribution with QoS between several PHB. This article proposes the use of heuristics in BB to optimize the network use and help in resources allocations control for experiments with DiffServ support.

Keywords: DiffServ, SOA, QoS, Bandwidth Broker, WebLabs.

1 Introdução

Uma rede de computadores permite que diversos usuários utilizem aplicações para a troca de mensagens, acesso remoto a outros computadores e demais nós da rede, para *upload* e *download* de arquivos, entre muitos outros serviços. A Internet é uma rede que possui usuários que utilizam aplicações que exigem diferentes recursos da rede, mas a maioria dessas aplicações estabelece a comunicação informando o endereço IP e a porta remota sem levar em consideração outras necessidades do aplicativo e do usuário. Por muito tempo, a garantia de acesso à Internet com a velocidade contratada era o quesito suficiente para atender às necessidades da maioria desses usuários. Os esforços em pesquisa e desenvolvimento da arquitetura Internet se preocuparam em tornar a plataforma robusta e flexível para as aplicações [1].

A Qualidade de Serviço (*Quality of Service* - QoS) da rede é um fator importante para atender às necessidades das aplicações. QoS pode ser definida como um conjunto de requisitos garantidos para as aplicações quanto ao transporte de fluxos na rede [2].

Dentre os requisitos mais comumente aceitos para descrever a QoS são citados o atraso, perda de pacotes, *jitter* e taxa de erros. A Internet atual não faz distinção entre os tipos de tráfego e seus requisitos. O serviço de encaminhamento de pacotes oferecido é do tipo melhor-esforço, ou seja, não há garantias quanto à entrega de pacotes fim-a-fim. Os pacotes são encaminhados pelos roteadores de acordo com o algoritmo FIFO (*First-in, First-out*) com o simples descarte dos pacotes que excederem o buffer das interfaces de rede nos *hosts* intermediários. Para as aplicações de tempo-real, multimídia, ensino à distância, acesso e controle de instrumentos remotos, visualização científica, entre outros, são exigidos mais recursos da rede para garantir a qualidade fim-a-fim da comunicação [3]. Nesse contexto, a diferenciação dos serviços fornecidos otimiza o uso da rede para atender às necessidades de QoS das aplicações. Um quesito importante para administradores de rede e que também está relacionado à QoS diz respeito ao compartilhamento de largura de banda para diferentes classes de tráfego. Cada uma dessas classes pode oferecer diferentes garantias para o encaminhamento dos pacotes que trafegarem por elas. Essa solução é uma alternativa para prover o uso mais eficiente da rede, ainda que a capacidade do enlace seja fundamental para oferecer serviços de qualidade.

A abordagem DiffServ (*Differentiated Services*) ganhou destaque nos últimos anos por permitir o controle de fluxos de rede com uma escalabilidade maior se comparada às demais propostas [3]. No entanto, a avaliação das configurações fica a cargo do administrador de redes que pode escolher as ferramentas que julgar mais adequadas para avaliar os resultados. A diferenciação de serviços reduz o desperdício de uso do enlace e é uma alternativa para otimizar a distribuição de recursos na rede [4].

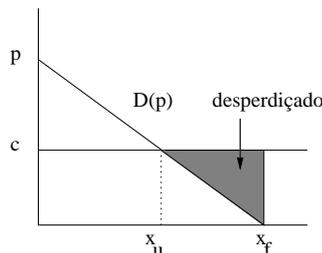


Fig. 1. Custo por Demanda de *Bandwidth* [4]

A Fig. 1 ilustra um cenário onde o ISP (*Internet Service Provider*) atribui um custo fixo c para o uso x_u da banda. p representa o valor cobrado proporcional à unidade de uso, medida em Megabytes (MB) de dados transferidos ou minutos de tempo de conexão. A demanda do usuário é modelada como uma função $D(p) = x_u$ e o custo absoluto relativo ao uso de recursos pode ser calculado como a área cx_f , onde x_f é a situação onde o usuário não consome recursos da rede ($D(0) = x_f$). No entanto, a demanda do usuário não é constante: para as oscilações que ocorrem no intervalo entre x_u e x_f o custo efetivo é o resultado da soma de todos os custos das n oscilações em um determinado período de conexão, ou seja, $\sum_{i=1}^n c(x_{fi} - x_{ui})$. Esse custo é inferior ao custo fixo cobrado pelo ISP. A conclusão é que mesmo não utilizando os recursos, o custo associado à demanda permanece constante, com um desperdício da alocação de recursos.

Esse artigo utiliza um WebLab de redes que permite a realização de diversos experimentos DiffServ respeitando heurísticas para gerenciar e negociar a provisão de QoS para as aplicações. É importante destacar que essas heurísticas não são de caráter exclusivo para experimentos em WebLabs, de forma que podem ser estendidas para quaisquer outros domínios. O controle dos valores mínimos e máximos de cada porção do enlace evita que uma sobrecarga seja admitida no domínio, ou seja, os fluxos inadimplentes são descartados antes mesmo de serem processados pelas disciplinas de fila ou entrarem no domínio. As heurísticas são políticas de controle de admissão implementadas pelo BB, mas o administrador do domínio precisa orientar o uso desse controle. Dessa forma, os limites de banda para cada tipo de fluxo não são fixados, mas o BB mantém o controle de como a configuração proposta é criada. Isso é necessário para que a reserva de largura de banda não ultrapasse a capacidade do enlace e nem interfira nas outras distribuições de banda já realizadas.

Para que os nodos do laboratório reflitam as mesmas configurações diante da submissão de tráfego, um Bandwidth Broker [5] foi utilizado juntamente com um experimento que exhibe em tempo-real o resultado das configurações DiffServ submetidas a diferentes tipos de fluxos agregados.

O artigo é apresentado da seguinte forma: a Seção 2 descreve os trabalhos relacionados sobre DiffServ e WebLabs. A Seção 3 descreve as heurísticas para reserva e negociação de recursos que foram utilizadas nos experimentos. A Seção 4 faz a avaliação do uso das heurísticas propostas. Finalmente, a Seção 5 faz as considerações finais.

2 Trabalhos Relacionados

Os trabalhos correlatos foram divididos em dois grupos porque abordam os principais trabalhos que orientaram a integração dos experimentos DiffServ no WebLab de experimentação remota NetLab WebLab [5], utilizado neste trabalho.

1) WebLabs de Interação com Dispositivos Físicos: Um WebLab é um laboratório remoto controlado através da Internet. Os WebLabs de experimentação remota permitem que estudantes acessem via TCP/IP o equipamento de hardware e programas [6]. Esses laboratórios diferem dos WebLabs de simulação e de experimentação virtual que utilizam exclusivamente softwares para representar os recursos e os resultados dos testes. Dentre os vários projetos de WebLabs, o projeto iLab [7] do MIT (*Massachusetts Institute of Technology*) se destaca na proposição de laboratórios reais acessíveis através da Internet ao propor um *toolkit* de módulos reutilizáveis, um conjunto de protocolos padronizados e Web Services. Todo esse conjunto formou um *framework* de software conhecido *iLab Shared Architecture*. Por outro lado, o WebLab-Deusto [8], que é um laboratório de microeletrônica, destaca-se ao evoluir de soluções de software proprietárias para soluções de software livre e de código-fonte aberto. O NetLab WebLab aproveita parte das propostas anteriores, mas segue a arquitetura orientada a serviços do WebLab GigaBOT [9]. Este é um laboratório para educação robótica móvel que utiliza soluções de software livre e de código-fonte aberto. Na arquitetura, os experimentos do laboratório são formados

através da composição dos serviços oferecidos pelo WebLab, reduzindo o esforço para a criação de novos experimentos. O modelo de referência permite que diversos WebLabs sejam integrados. Os elementos centrais são o participante, que pode ser um usuário individual ou um grupo, e o WebLab. Para utilizar um WebLab o participante deve possuir credenciais e estabelecer uma ou mais sessões com o WebLab.

2) Implementação da Arquitetura DiffServ: Dentre as abordagens sugeridas para contornar o problema da reserva de recursos na Internet destacam-se as que utilizam o componente *Bandwidth Broker* (BB) para centralizar as requisições de parâmetros de QoS, distribuir e gerenciar o uso recursos no domínio. O NetLab WebLab baseou-se na implementação de uma arquitetura multi-camadas [10] onde o BB interage com os elementos presentes em outras camadas lógicas no domínio.

Outra proposta de implementação de um domínio DiffServ descreve um mecanismo de incorporação de qualidade de serviço em aplicações que necessitam de largura de banda assegurada e baixo *jitter* para aplicações telemáticas [11].

Finalmente, diversas abordagens [12] [13] [14] utilizam o software de simulação NS-2 (*Network Simulator-2*) para avaliar diversos parâmetros de QoS em um domínio DiffServ. As simulações com NS-2 estão fora do escopo desse trabalho, mas esses estudos contribuíram ao descrever um conjunto de metodologias que poderiam ser realizadas em um ambiente de testes “real” com dispositivos físicos gerenciáveis.

3 Heurísticas para Reserva e Negociação de Recursos

A concorrência que ocorre entre fluxos reduz a qualidade de encaminhamento de pacotes, resultando em maior atraso na entrega, perda de pacotes e aumento do jitter. Entre fluxos de mesmo PHB (*Per Hop Behavior*) é desejado que exista uma distribuição equitativa de banda de forma que cada fluxo possua uma quantidade mínima de banda. Entre fluxos de diferentes PHBs é desejado que exista a garantia mínima de banda sem que essa garantia implique na penalização excessiva de fluxos de outros PHBs. A garantia de recursos é uma garantia estatística, ou seja, em virtude da capacidade do enlace será realizado um tratamento diferenciado do tráfego com a distribuição dos recursos entre os diferentes tipos de serviços de encaminhamento. Diante dessas considerações, as seguintes heurísticas foram propostas:

1. $\sum_{i=1}^n \max_{PHB\ i} \leq \text{enlace}$: Cada PHB possui uma configuração com determinada quantidade de banda do enlace. Quando a banda estiver com baixa utilização faz sentido permitir que os fluxos utilizem mais recursos da rede. No entanto, se muitos fluxos forem admitidos, as limitações impostas por disciplinas de fila hierárquicas, tais como HTB (*Hierarchical Token Bucket*) não são suficientes para descartar adequadamente os pacotes inadimplentes de forma a promover uma correta preempção de recursos, com perdas reduzidas, para os fluxos de maior prioridade, porque os fluxos já foram admitidos. Ou seja, para alguns poucos PHBs de alta prioridade, os fluxos de menor prioridade serão severamente penalizados e podem nem mesmo conseguir uma quantidade mínima de recursos. O inverso também é válido: quando muitos fluxos de baixa prioridade são admitidos estes serão severamente penalizados para permitir a garantia mínima de recursos para alguns poucos fluxos de alta prioridade.

Essa regra garante que cada PHB possua uma configuração com determinada porção da rede, ou seja, a máxima porção do enlace que pode ser alocada para cada PHB ($\sum_{i=1}^n \max_{PHB\ i}$) deve ser menor do que a capacidade do enlace global (enlace). Essa regra limita a quantidade de fluxos logo na entrada do domínio, sem interferir nas demais regras de distribuição de recursos promovidos pelas demais configurações DiffServ, sem penalizar outros fluxos ou ultrapassar a capacidade do enlace.

2. $\sum_{i=1}^n \text{alocado}_{PHB\ i} \leq \max_{PHB}$: Cada configuração DiffServ para um PHB é capaz de definir os limites máximos de uso de porções do enlace (\max_{PHB}). Quando um fluxo é admitido no domínio ele solicita determinada quantidade de recursos ao BB. Como fluxos agregados são mapeados para PHBs faz sentido que a configuração de um PHB possa admitir tantos fluxos mapeados quanto possíveis, desde que o somatório dos recursos do enlace alocados para cada admissão sejam mantidos dentro da porção do PHB ($\sum_{i=1}^n \text{alocado}_{PHB\ i}$).

3. $\min_{PHB} \leq \text{alocado}_{PHB}$: Quando uma parcela de recursos do enlace é alocada para um fluxo mapeado para um PHB (alocado_{PHB}) deve existir a garantia de que uma quantidade mínima de recursos (\min_{PHB}) será mantida para a alocação. No entanto, à medida que o enlace é saturado e mais fluxos são admitidos, cada um deles deve possuir uma quantidade mínima de recursos que justifique a alocação. Para os fluxos de mesmo PHB isso significa que os recursos serão compartilhados igualmente entre os fluxos. Essa garantia implica na determinação da quantidade de fluxos que podem ser submetidos ao mesmo tempo com uma quantidade mínima de recursos.

As regras são divididas em dois grupos: a primeira regra mantém o critério para a definição dos limites de largura de banda; as demais mantêm os critérios para a admissão dos fluxos na rede. Como consequência, $\sum_{i=1}^n \text{alocado}_{PHB\ i} \leq \text{enlace}$, ou seja, cada PHB admite uma determinada quantidade de fluxos com recursos mínimos garantidos para cada um desses fluxos. Quando a porção do enlace atribuída a cada PHB estiver completa é necessário evitar que o conjunto de admissões de um PHB interfira nas admissões de outro. As regras garantem que a alocação de recursos para uma determinada classe de fluxos seja realizada independente da outra, e que todas elas respeitem a mesma largura de banda. Essas regras garantem a integridade das alocações com a definição dos limites de cada PHB, com as restrições dos valores máximos permitidos para cada um deles.

Uma vez que o fluxo é admitido são garantidos mais ou menos recursos dentro dos limites impostos para o PHB, sem que isso interfira nas alocações de outras classes. Na verdade é realizada uma distribuição controlada de recursos para várias redes lógicas em uma mesma rede física. A otimização do uso da banda ocorre de maneira controlada dentro dos limites atribuídos a cada PHB para evitar que um conjunto de admissões inviabilize ou penalize severamente as admissões de outras classes.

4 Avaliação do Uso das Heurísticas

O uso das heurísticas propostas permite um controle mais eficiente da largura de banda do laboratório. O BB precisa manter o controle das alocações de banda para evitar que muitos experimentos sejam iniciados e sobrecarreguem a rede do

laboratório. Uma configuração desse tipo é realizada por um usuário administrativo para que a distribuição de recursos do enlace seja utilizada pelos experimentos. Mas não é permitido que diversos usuários atuem nessa configuração ao mesmo tempo, o que sobrecarregaria o elemento BB.

Nos cenários apresentados são submetidos diferentes agregados à rede, ou seja, um conjunto de pacotes IP com o cabeçalho marcado com o mesmo DSCP (*Differentiated Services Codepoint*) no campo DS (*Differentiated Services*). Para a geração de fluxos e recuperação das estatísticas do tráfego submetido à rede foram utilizados os softwares RUDE (*Real-time UDP Data Emitter*) e CRUDE (*Collector for RUDE*) [15]. Para recolher corretamente os valores da vazão e perda de pacotes criou-se uma nova versão do software Qosplot [16]. A submissão de fluxos foi realizada na rede principal de alta velocidade (1000Mbps) do NetLab WebLab[5].

Para avaliar a submissão de fluxos no domínio, foram inseridas disciplinas de fila nos roteadores de borda do domínio para limitar a taxa de entrada de pacotes, assegurando que os fluxos que ultrapassem a taxa pré-estabelecida serão descartados antes mesmo de entrarem na pilha de comunicação da interface de ingresso. Nesses *hosts* também foi realizado o controle de tráfego com o policiamento, para determinar a taxa de entrada de pacotes no domínio. Dessa forma, os demais *hosts* do domínio, ou seja, os nós de núcleo, irão apenas encaminhar os fluxos previamente filtrados.

4.1 Cenário #1: Vazão dos fluxos ultrapassa a largura de banda global

Neste cenário, a largura de banda da rede é limitada em 10MBps (Megabytes/segundo) e não são utilizadas as heurísticas propostas. A vazão mínima garantida para cada agregado é de 1MBps. A vazão máxima permitida foi distribuída da seguinte forma: AF11: 5MBps, AF21: 4MBps, AF22, BE e EF: 3MBps. A vazão submetida por cada agregado é mostrada na Tab. 1. A Fig. 6 ilustra este cenário.

PHB	Período (segundos)	Pacotes/Segundo	Bytes/Pacote
AF11, AF21	1-5	2000	500
AF11, AF21	5-10	2000	1000
AF11, AF21	10-15	2000	1500
AF11, AF21	15-20	2000	2500
AF11, AF21	20-25	2000	500
AF11, AF21	25-30	2000	1000
AF11, AF21	30-40	2000	1500
AF22, BE, EF	1-40	2000	1500

Tab. 1. Vazão submetida por cada agregado mapeado para um PHB.

A consequência de assegurar a vazão para cada agregado sem respeitar os limites do enlace é a diminuição da QoS para a aplicação que submete os fluxos. Com o intuito de garantir uma vazão para todos os agregados, estes competem entre si e todos são igualmente penalizados porque não há uma distribuição adequada da banda. Na configuração, o fluxo do tipo EF (*Expedited Forwarding*) foi mapeado para uma disciplina de fila FIFO com tamanho de fila reduzido, mas percebe-se que essa opção

não é a ideal para concorrer com os outros fluxos AF (*Assured Forwarding*) em uma disciplina HTB. A preocupação nesse mapeamento EF foi a de oferecer baixo *delay*, sem considerar a preempção que seria, a priori, a mais desejada. O objetivo desse cenário é mostrar que as configurações de disciplinas de fila não privilegiam a redistribuição de pacotes. É possível outra configuração DiffServ que privilegie os fluxos EF mas, a excessiva competição entre os fluxos admitidos reduz a QoS do enlace, com a ocorrência de altos índices de perda. O cenário é baseado em uma configuração que tenta distribuir os recursos disponíveis para cada agregado quando o enlace não estiver completamente utilizado.

O gráfico da vazão representa o agregado AF que define a maior prioridade para o fluxo AF11 e a menor para o fluxo BE (*Best Effort*). À medida que os fluxos AF11 e AF21 aumentam a sua vazão, os demais fluxos são penalizados. Até o instante de 15 segundos, AF22, BE e EF são penalizados. Isso ocorre porque AF11 e AF21 estão inseridos na disciplina GRED (*Generalized Random Early Detection*) com largura de banda maior e alta prioridade de encaminhamento de pacotes em relação aos demais fluxos AF. Os fluxos BE e EF participam da distribuição de banda, mas por estarem em outras classes, apenas lhes é assegurada a vazão mínima garantida em situações de congestionamento. De 15 a 20 segundos, AF11 submete à rede a vazão de 5MBps. Nesse período, AF21 também é penalizado, mas mantém a sua vazão acima da largura de banda mínima assegurada. Como a largura de banda máxima foi limitada, os fluxos utilizam praticamente toda a banda. No instante de 20 segundos, AF11 e AF21 reduzem a sua vazão, o que permite que os demais fluxos ocupem a banda restante.

Na sequência e à medida que os fluxos AF11 e AF21 aumentam a sua vazão, os demais fluxos são novamente penalizados. Percebe-se que a preempção de recursos ocorre entre os fluxos das classes AF configurados com a disciplina GRED, mas os serviços BE e EF são tratados igualmente e esse comportamento é indesejável. Esse problema ocorre porque é permitida a entrada no domínio de fluxos de até 10MBps. A configuração GRED para as classes AF permite distribuir adequadamente os recursos entre AF11, AF21 e AF22, mas a alta vazão de entrada permitida para BE penaliza o comportamento do agregado EF nessa configuração, ou seja, é necessário um controle mais preciso dos fluxos na entrada do domínio para auxiliar as disciplinas de fila a darem prioridade no encaminhamento dos agregados. É indesejável que agregados de alta prioridade tenham altos índices de perda.

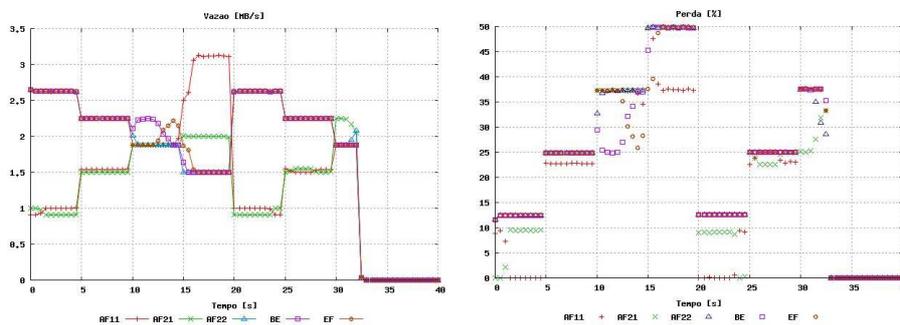


Fig. 6. Vazão e perdas obtidos com a garantia de vazão para cada agregado.

A queda abrupta no final das plotagens ocorre devido à consulta pela atividade da sessão através da rede de retaguarda do NetLab WebLab, que possui um *hub* Ethernet de 10/100Mbps (Megabits/segundo).

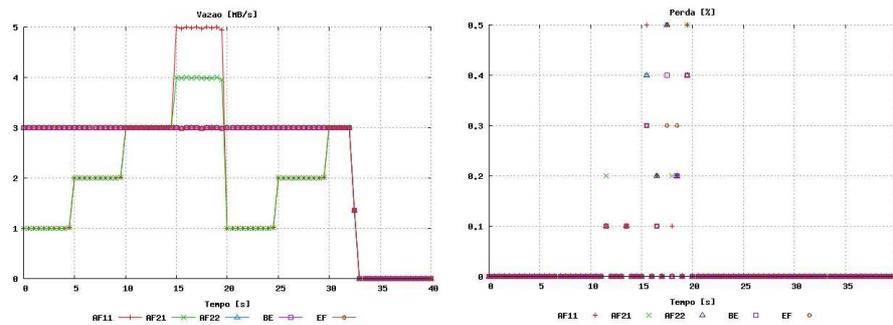


Fig. 7. Vazão e perdas obtidos com o aumento da largura de banda.

Para resolver os problemas desse teste foi proposto o aumento da largura de banda, como mostra a Fig. 7, sem a utilização das heurísticas propostas. Neste cenário, a largura de banda foi incrementada de 10MBps para 20MBps. A mesma vazão dos agregados foi novamente submetida. A consequência direta do aumento da largura de banda é o aumento da QoS de vazão oferecida para as aplicações. No entanto, o custo associado ao aumento da oferta de recursos pode não ser proporcional ao seu uso, resultando em desperdício de utilização. Os fluxos BE de menor prioridade obtiveram a mesma vazão que os fluxos EF de maior prioridade, o que é indesejável.

4.2 Cenário #2: Admissão de fluxos respeitando as heurísticas

Este cenário tem o objetivo de se aproximar do teste anterior com uma largura de banda de 10MBps. Para isso, é considerado que a distribuição adequada dos recursos da rede é preferível à competição natural entre os agregados e que deve ser mantida a reserva de recursos para os agregados EF em detrimento dos demais. Os agregados AF devem manter a competição por recursos entre si, mas para BE é preferível uma acentuada penalização em benefício dos demais.

Esse cenário é possível com o uso das heurísticas propostas inseridas nos filtros de ingresso, auxiliando o encaminhamento dos agregados nas disciplinas de fila. A mesma vazão dos cenários anteriores é submetida por cada agregado. No entanto, é promovida uma realocação de banda para respeitar as heurísticas e esse processo foi realizado no BB. A vazão mínima garantida para cada agregado se manteve em 1MBps. A vazão máxima permitida foi redistribuída da seguinte forma: AF11: 3MBps, AF21: 2MBps, AF22 e BE: 1 e EF: 3MBps. Essa realocação é realizada pelo administrador do WebLab para que a distribuição de recursos respeite as necessidades das aplicações. A Fig. 8 ilustra o resultado da submissão neste cenário.

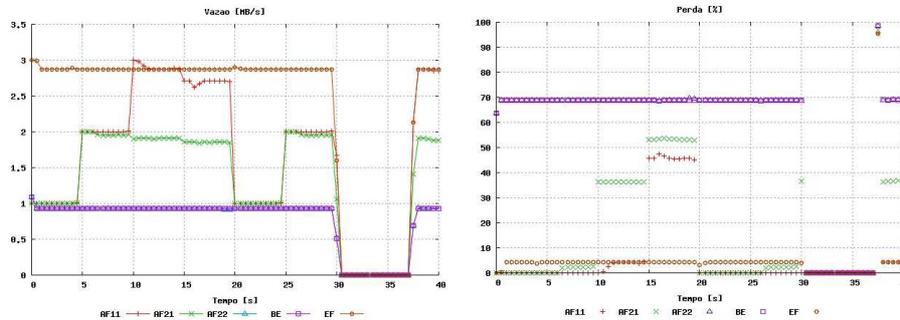


Fig. 8. Vazão e perdas obtidos com a atribuição das heurísticas.

Como consequência direta dessa redistribuição tem-se uma redução significativa da perda de pacotes com a garantia da vazão para os agregados restrita nas porções alocadas de cada PHB. Outra consequência é que se tem uma perda acentuada de pacotes para os agregados BE de menor prioridade, mas essa perda é controlada e tem o intuito de beneficiar os agregados EF de maior prioridade de encaminhamento. Percebe-se que é possível manter uma configuração de disciplinas de fila para EF que apenas assegurem um tamanho de fila reduzido com baixo *delay*. A presença dos filtros de ingresso auxilia a configuração, garantindo uma alta vazão. A redistribuição de recursos reduz significativamente a concorrência entre os agregados associados a PHBs diferentes, o que garantiu a submissão do agregado EF com perdas reduzidas. Os agregados ainda trafegam com a mesma vazão até atingirem o roteador de ingresso no domínio DiffServ.

5 Considerações Finais

Um ambiente DiffServ precisa oferecer serviços consistentes para as aplicações que submetem diferentes tipos de fluxos ao domínio. São válidas quaisquer configurações que permitam diferenciar o tráfego para oferecer QoS para as aplicações, mas a falta de gerência na distribuição de largura de banda é inviável porque reduz qualidade do serviço contratado. O uso das heurísticas propostas otimiza a distribuição de recursos e permite a gerência adequada da rede entre os diversos tipos de usuários que utilizam o domínio. O artigo preocupou-se em definir regras que orientam a distribuição de fluxos no domínio, independente de quais configurações DiffServ sejam utilizadas. Esse é um passo genérico para otimizar o tráfego de pacotes, mas passos específicos como o atraso, *jitter*, pacotes desordenados e erros também devem ser analisados porque são métricas para avaliar a QoS na rede.

As aplicações multimídia são altamente dependentes dos recursos disponíveis na rede e a sua disponibilização no ambiente Internet é um desafio. Essas aplicações dependem significativamente da qualidade do serviço oferecida para transferir os seus dados, do enlace entre os *hosts*, entre outros. A necessidade de um modelo para prover QoS é importante quando o serviço na Internet está sob restrições, por

exemplo, para WebLabs robóticos ou redes de computadores, configurados e/ou manipulados remotamente. A implementação dos experimentos no WebLab justifica a importância da configuração adequada dos quesitos de QoS para os mais diversos tipos de experimentos.

Referências

1. Fernandes, S., Kamienski, C., Mariz, D., and Sadok, D. Avaliação de Técnicas de Agrupamento na Amostragem de Tráfego na Internet. XXIV SBRC (2006).
2. Crawley, E., Nair, R., Rajagopalan, B., and Sandick, H. A Framework for QoS based Routing in the Internet. Technical Report RFC 2386, Network Working Group (1998).
3. Teitelbaum, B., Hares, S., Dunn, L., Neilson, R., Narayan, V., and Reichmeyer, F. Internet2 QBone: Building a Testbed for Differentiated Services. IEEE Network, Vol.13, No.5, pp. 8–16. IEEE Network (1999).
4. Edell, R. J., Varaiya, P. P., Edell, R., and Varaiya, P. Providing Internet Access: What We Learn From INDEX. In IEEE Network, pp. 18–25. (1999).
5. Agostinho, L., Farias, A. F., Faina, L. F., Guimarães, E. G., Cardozo, E. and Coelho, P.R.S.L. Uma Proposta de Arquitetura para Experimentos DiffServ em WebLabs. In XXXIV CLEI (2008).
6. Lopez-de Ipiña, D., Garcia-Zubia, J., and Orduña, P. Remote Control of Web 2.0-Enabled Laboratories from Mobile Devices. In Proc. Second IEEE International Conference on e-Science and Grid Computing e-Science '06, pp. 123. e-Science (2006).
7. Lerman, S. and del Alamo, J. iLab: Remote Online Laboratories. <http://icampus.mit.edu/projects/iLabs.shtml>. (2005).
8. García-Zubia, J., Lopez-de Ipiña, D., Orduna, P., and Hernández-Jayo, U. Experience with WebLab-Deusto. In Proc. IEEE ISIE, Vol. 4, pp. 3190–3195. (2006).
9. Coelho, P.R.S.L., Sassi, R.F., Cardozo, E., Guimarães, E.G., Faina, L.F., Lima, A.Z., Pinto, R.P.: A Web Lab for Mobile Robotics Education. IEEE International Conference on Robotics and Automation, pp. 1387-1386. ICRA (2007).
10. Politis, G. A., Sampatakos, P., Iakovos, D., and Venieris, S. Design of a Multi-Layer Bandwidth Broker Architecture. In Lecture Notes in Computer Science; Vol. 1938. Springer Verlag (2000).
11. Pinto, R. P., Guimarães, E. G., Cardozo, E., and Magalhães, M. F. Incorporação de Qualidade de Serviço em Aplicações Telemáticas. XXI SBRC (2003).
12. Reis, L. and Guardieiro, P.R. An Enhanced Allocation Resource Mechanism for DiffServ Domains. In Proc. International conference on Networking and Services ICNS '06, (2006).
13. Bouras, C., Pappas, I., Primpas, D., and Stamos, K. Using the NS-2 Simulation Environment to Implement and Evaluate Bandwidth Broker Models. In Proc. 2nd Conference on Next Generation Internet Design and Engineering NGI '06. (2006).
14. Lakkakorpi, J., Strandberg, O., and Salonen, J. Adaptive Connection Admission Control for Differentiated Services Access Networks. IEEE JSAC, Vol.23, No.10, pp. 1963–1972. (2005)
15. Laine, J., Saaristo, S., and Prior, R. Rude and Crude. <http://rude.sourceforge.net>. (1999).
16. Ubik, S. Qosplot. <http://www.ces.net/project/qosip>. (2004).