

Um Sistema de Engenharia de Tráfego Auto-Gerenciável para Transmissão de Conteúdos Multimídia em Redes IP

Pedro D. O. Carvalho¹, John Lennon O. Couto², Orlewilson B. Maia¹, Álvaro C. Lemos Neto², Felipe O. Guimarães², Luciano de Errico¹

¹Programa de Pós-Graduação em Engenharia Elétrica – ²Univ. Federal de Minas Gerais
Av. Antônio Carlos, 6627 - CEP 31270-901 – Belo Horizonte – MG – Brasil

{pedrodoc, johnlcouto}@gmail.com, omaia@ufmg.br, alvarolemos@gmail.com,
felipeoguimaraes@hotmail.com, l.errico@cpdee.ufmg.br

Abstract. *The rapid growth of video and VoIP traffic in corporate and ISP networks weakens QoS provision for this class of applications. This paper presents a Self-Managing Traffic Engineering approach that, without human intervention, is capable of sustaining the agreed QoS level for each different communication flow. Experiments conducted in the ns-2 simulator, with different video streams and abrupt changes in network conditions, confirmed the self-managing operation.*

Resumo. *O rápido crescimento do tráfego de vídeo e VoIP nas redes corporativas e de provedores de acesso compromete o fornecimento de QoS para estas aplicações. Este artigo apresenta uma abordagem de Engenharia de Tráfego Auto-Gerenciável, capaz de, sem intervenção humana, manter o nível de QoS acordado para cada diferente fluxo de comunicação. Experimentos realizados no simulador ns-2, com diferentes fluxos de vídeo e mudanças abruptas na rede, comprovaram o funcionamento auto-gerenciável.*

1. Introdução

A tendência atual é o aumento significativo do tráfego multimídia nas redes, em especial tráfego de vídeo como VoD (*Video on Demand*), WebTV e Televisão sobre IP (*IP TeleVision* – IPTV). Entretanto, a rede IP originalmente oferece apenas um nível de serviço (*best effort*), enquanto que o tráfego de vídeo exige diferentes níveis de Qualidade de Serviço (QoS). As abordagens existentes para a oferta de QoS em redes IP incluem: o superdimensionamento da rede, que pode acabar estimulando o aumento da demanda; o IntServ (*Integrated Services*), que exige demais da rede para atender cada aplicação individualmente; e o DiffServ (*Differentiated Services*), que não é eficaz quando muitas aplicações da mesma classe disputam os recursos da rede.

A proposta deste artigo é permitir um funcionamento “inteligente, dinâmico e autônomo” da rede, proporcionado por um Sistema de Engenharia de Tráfego (*Traffic Engineering* – TE) Auto-Gerenciável. Este sistema, atuando nos roteadores de borda e controlando o roteamento e a admissão de aplicações, deve ser capaz de, sem intervenção humana e de modo dinâmico, manter o nível de QoS acordado para cada fluxo de comunicação, mesmo em face de mudanças abruptas nas condições da rede. As redes alvo são as redes IP corporativas com domínio MPLS [ROSEN et al., 2001].

2. Trabalhos Relacionados

Em Majd e Yaghmaee (2006) é proposto um algoritmo nebuloso de roteamento por QoS, capaz de encontrar rotas obedecendo a restrições de vazão, atraso e perda. Santos e Mateus (2009) descrevem uma alocação dinâmica de rotas com otimização multi-objetivo para prover QoS. Mawass et al. (2012) apresentam uma distribuição *offline* de *Labels* MPLS com algoritmo genético multi-objetivo distribuído. Guo et al. (2013) detalham o uso de serviço de fila diferenciado para conferir adaptabilidade de QoS em redes *wired-wireless*.

3. Sistema de Engenharia de Tráfego Proposto

No sistema de TE proposto, uma aplicação solicita alocação de recursos da rede (vazão, atraso, *jitter* e perdas) a um módulo otimizador, responsável por definir um caminho no domínio do MPLS que atenda as necessidades da aplicação. Se não for encontrado um caminho que atenda as necessidades da aplicação, a admissão é negada. Caso contrário, a aplicação é admitida e seu fluxo será transmitido pelo caminho estabelecido. Um módulo de medição periodicamente atualiza um registro do estado de cada enlace, o que permite monitorar a QoS oferecida. Quando é detectada uma QoS abaixo do acordado, o módulo de otimização é acionado para encontrar um novo caminho para o fluxo. Aplicações que não solicitem reserva de recursos serão consideradas como fluxos não prioritários, sendo transmitidos por meio de *best effort*.

O experimento de teste do sistema de TE proposto está ilustrado na Figura 1, onde são mostrados seus cinco elementos: (1) *arquivo descritor da simulação*; (2) *interpretador*; (3) *simulação*; (4) *medidor de QoS*; e (5) *otimizador de rotas*.

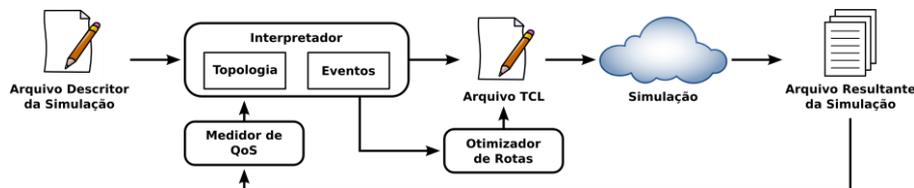


Figura 1. Fluxograma do sistema de engenharia de tráfego proposto

Através do *Arquivo Descritor da Simulação*, o usuário descreve todas as informações necessárias para a simulação do cenário desejado. O *Interpretador* organiza as informações em duas estruturas de dados: *Topologia*, que armazena o estado de todos os enlaces, e *Eventos*, que armazena as ações que deverão ocorrer ao longo da simulação, como início de transmissão e quedas de enlaces. O *Otimizador de Rotas* recebe as novas requisições de fluxos pela estrutura *Eventos* e o estado da rede pela estrutura *Topologia*, definindo a rota de cada fluxo. Em seguida, essas informações são descritas em linguagem *Tool Command Language* (TCL) e enviadas para o EvalVid, um *framework* que possibilita a transmissão de vídeos no simulador de redes *Network Simulator 2* (ns-2) [KLAUE et al. 2003]. O resultado da simulação é analisado pelo *Medidor de QoS*, que extrai informações de vazão, atraso, *jitter* e perdas de pacotes para cada enlace e fluxo estabelecido. Tais informações são utilizadas para atualizar o estado atual da rede, fechando o *loop*. Este processo é repetido até que todos os eventos tenham sido processados.

Dado que diferentes requisitos precisam ser atendidos, o problema de otimização se configura como multi-objetivo. No entanto, devido ao alto custo computacional dos algoritmos de otimização multi-objetivo e a necessidade de uma otimização rápida e dinâmica, decidiu-se transformar o problema em mono-objetivo. Para isso, utilizou-se uma única fun-

ção que considera o custo de todos os objetivos, ponderados por pesos definidos conforme o perfil da aplicação solicitante. Pela agilidade, optou-se por utilizar Algoritmos Genéticos (GA). Cada indivíduo é composto por genes, os quais identificam os caminhos candidatos para cada fluxo a ser alocado na rede. O custo de um gene é baseado no estado de todos os enlaces que formam o caminho candidato e o custo total do indivíduo é a soma dos custos de todos os genes. O GA, baseado em estudos passados de Andrade et al. (2008), foi desenvolvido utilizando cruzamento em dois pontos, mutação aleatória em qualquer gene de qualquer indivíduo e uma variação da seleção utilizando *stochastic remainder sampling* com torneio.

4. Resultados Preliminares

Para avaliar o sistema de TE proposto, foram criadas três aplicações prioritárias: Vídeo 1 (resolução 1280x720 pixels, 60 fps e bitrate de 2000 kbps), Vídeo 2 (resolução 640x360 pixels, 25 fps e bitrate de 1000 kbps) e uma aplicação com fluxo constante (CBR) a 600 kbps. Também foram adicionadas aplicações não prioritárias (FTP), como tráfego de fundo. A topologia utilizada está ilustrada na Figura 2.

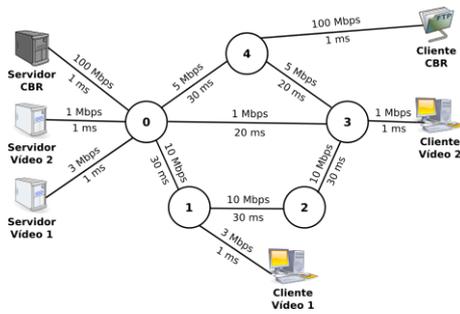


Figura 2. Topologia utilizada nos experimentos

Tabela 1 – Rotas adotadas pelos sistemas sem/com TE proposto, antes/depois da falha

Aplicações Prioritárias	Antes da falha		Após a falha	
	Sem TE	Com TE	Sem TE	Com TE
Video1	0-1	0-1	0-3-2-1	0-4-3-2-1
Video2	0-3	0-4-3	0-3	0-4-3
CBR	0-4	0-4	0-4	0-3-4

A simulação teve duração de 20s e consistiu em avaliar os elementos de QoS (*jitter*, atraso, vazão e perda de pacotes) de cada aplicação prioritária, nos cenários sem e com o uso do TE proposto, conforme os seguintes parâmetros de QoS: reserva de 3Mbps e atraso < 150ms para o Vídeo 1; reserva de 1Mbps e atraso < 150ms para o Vídeo 2; e reserva de 0,7 Mbps e atraso < 150ms para o CBR. As aplicações não prioritárias e prioritárias foram iniciadas nos instantes 1s e 2,3s, respectivamente. No instante 5s ocorre uma queda de enlace entre os nós 0 e 1. A Figura 3 apresenta a vazão e o atraso das aplicações prioritárias. As rotas adotadas estão descritas na Tabela 1.

A solução ótima encontrada pelo sistema com TE no momento da admissão dos fluxos prioritários coincide com algumas rotas de menor comprimento (medido em número de saltos), o que contribuiu para que ambos os sistemas apresentassem desempenho semelhante nos instantes iniciais. De fato, a única rota diferente antes da falha é a alocada para o Vídeo 2, que foi o único fluxo que mostrou um desempenho inferior para o caso sem TE (vide o gráfico de atraso). Apesar da rota 0-3 ser mais curta do que a rota 0-4-3, escolhida pelo sistema com TE, ela possui capacidade máxima igual à vazão requerida pelo Vídeo 2, não restando largura de banda para os fluxos de FTP que também disputam pelo acesso ao enlace, elevando o tempo médio de fila desta rota.

A diferença entre os sistemas é evidenciada após a queda do enlace 0-1. Nessa situação, todo o conjunto de rotas de cada sistema é diferente. Os resultados observados para va-

zão e atraso mostram que no instante da falha ambos os sistemas sofrem uma perturbação, principalmente o Vídeo 1 que utilizava o enlace afetado pela falha. No entanto, a solução aplicada pelo sistema com TE estabiliza os fluxos dentro dos padrões de QoS requisitados. Já a solução do sistema sem TE se mostra insuficiente para os fluxos Vídeo 1 e Vídeo 2, permitindo que o atraso de ambos aumente até o congestionamento das rotas, ocasionando perdas de pacotes entre 60% a 100%.

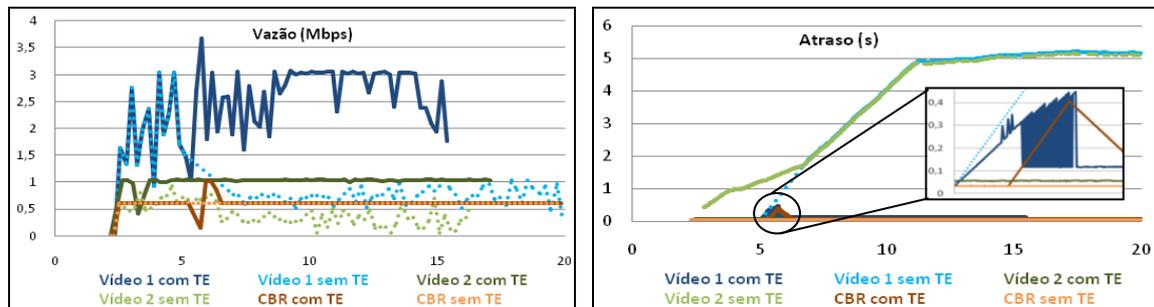


Figura 3. Vazão (à esquerda) e atraso (à direita) das aplicações prioritárias

5. Conclusões e Trabalhos Futuros

O sistema autônomo de engenharia de tráfego proposto apresentou capacidade de auto-otimização e auto-configuração ao identificar e alocar um conjunto de rotas que melhor atendesse a QoS requerida pelas aplicações prioritárias. Também demonstrou auto-cura, reagindo a uma falha de enlace, encontrando uma nova solução e mantendo as necessidades de QoS requeridas. Para trabalhos futuros, o sistema deverá ser capaz de reagir às aplicações que violem o contrato de QoS requisitada e admitida, conferindo ao sistema maior grau de auto-proteção, bem como a realização de experimentos para negação da admissão de uma aplicação que solicite recursos além do que a rede dispõe.

Agradecimentos

Este trabalho foi realizado com o apoio financeiro da FAPEMIG e da CAPES – Brasil.

Referências

- Andrade, A. V., de Errico, L., Aquino, A. L. L., Assis, L. P., and Barbosa, C. H. N. R. (2008). Analysis of Selection and Crossover Methods used by Genetic Algorithm-based Heuristic to solve the LSP Allocation Problem in MPLS Networks under Capacity Constraints. In *EngOpt 2008*, pages 1 – 9.
- Guo, M. et al. (2013). Provisioning of QoS Adaptability in Wired-Wireless Integrated Networks. In *Journals of Communications and Networks*, vol. 15, no. 1, pages 61 – 70.
- Klaue, J., Rathke, B., and Wolisz, A. (2003). EvalVid – A Framework for Video Transmission and Quality Evaluation, In *TOOLS 2003*, pages 255–272.
- Majd, N. E. and Yaghmaee, M. H. (2006). A fuzzy algorithm for QoS-based routing in MPLS network. In *APCC'06*, pages 1 – 5.
- Mawass, N. et al. (2012). Offline Multi-Class Flow Allocation in MPLS Networks Using a Distributed Multi-Objective Genetic Algorithm. In *ICT 2012*, pages 1 – 6.
- Rosen, E., Viswanathan, A., and Callon, R. (2001). MultiProtocol Label Switching (MPLS) architecture. IETF RFC 3031.
- Santos, F. A. and Mateus, G. R. (2009). Otimização multi-objetivo aplicada a alocação dinâmica de rotas LSP em redes MPLS. In *SBRC 2009*, pages 349 – 362.