Análise Quantitativa baseada em Medições de Sistemas P2P para Video Streaming

José Alexandre Macedo[1], Ivan de Oliveira Nunes[1], Ebenezer Nogueira da Silva[1], Alan Silva da Paz Floriano[1], Roberta Lima Gomes[1], Magnos Martinello[1]

[1] Departamento de Informática – Universidade Federal do Espírito Santo (UFES)
Av. Fernando Ferrari, S/N, 29060-970 – Vitória – ES, Brasil

{jamacedo, ionunes, ensilva, aspfloriano, rgomes, magnos}@inf.ufes.br

Resumo. *O uso de sistemas P2P Video Streaming tem se tornado cada vez mais popular na Internet. Esta popularidade decorre de diversas vantagens apresentadas por estes sistemas como economia na banda de transmissão nos servidores e ganhos em escalabilidade. Entretanto, comprovar efetivamente os ganhos em sistemas reais é um desafio importante e que poucos trabalhos têm dado atenção na literatura. A proposta desse artigo é analisar dois sistemas de P2P Video Streaming baseado em medições efetuadas experimentalmente de modo a quantificar i) a economia de banda no servidor e ii) a qualidade de experiência percebida pelos usuários. Com este propósito, os sistemas SopCast e Goalbit foram escolhidos, implantados e avaliados no ambiente configurado. Os resultados obtidos mostram economia de banda elevada nos dois sistemas, sendo que o Goalbit obteve o melhor resultado para poucos clientes, enquanto o SopCast foi superior com um número elevado de clientes. A qualidade de experiência foi sutilmente superior no Goalbit, porém em uma rede congestionada o SopCast obteve melhor desempenho.*

Abstract. *The use of P2P Video Streaming systems has increasingly becoming popular. This popularity stems from several advantages presented by this systems as economies in servers transmission bandwidth and improvements in scalability. However, effectively demonstrate the gains, in real systems, is a major challenge that few studies have given attention in the literature. The objective of this paper is to analyze two P2P Video Streaming systems based on experimental measurements performed in order to quantify i) the bandwidth economy in the server ii) the quality of experience perceived by users. With this purpose, SopCast and Goalbit systems were chosen, deployed and evaluated in the configured environment. The obtained results show high bandwidth economy in both systems. Goalbit obtained the best result for few clients while SopCast was best with a high number of clients. The quality of experience was slightly higher in Goalbit, but in a congested network SopCast performed better.*

1. Introdução

A transmissão de vídeo na Internet vem se tornando mais popular a cada dia. Estima-se que em 2012 mais de 50% do tráfego total na Internet irá corresponder à distribuição de conteúdo multimídia (Cisco, 2011). O crescimento do número de usuários associado ao aumento da demanda por vídeos com qualidade cada vez mais elevada tem inviabilizado o modelo cliente-servidor para distribuição de vídeo (Silverston et al., 2009).

Uma solução para viabilizar a distribuição de vídeo em larga escala na Internet baseia-se em redes P2P tanto para transmissões ao vivo quanto para vídeo sob demanda. Atualmente aplicações como PPLive[1], CoolStreaming[2], PPStream[3] e SopCast[4] utilizam a tecnologia P2P para transmitir diversos canais simultaneamente atingindo milhares de usuários. O SopCast, particularmente, tem adquirido grande popularidade alcançando, nos últimos anos, maior volume de tráfego se comparado com outras aplicações (Google, 2012).

A medição e comparação de sistemas *P2P Video Streaming* vêm sendo alvo de diversas pesquisas nos últimos anos (Hei et al. (2007), Lu et al. (2009), Sentinelli et al. (2007)). Os estudos, de uma forma geral, costumam avaliar atrasos de reprodução, estrutura de distribuição, tamanho do buffer, dinâmica da rede, taxas de download e upload, entre outros aspectos. Existem ainda esforços concentrados na compreensão do funcionamento dos protocolos fechados presentes na maioria destes sistemas (Gonçalves et al., 2011). Entretanto, esses estudos de medição não necessariamente analisam os sistemas P2P para *Video Streaming* reproduzindo todas as condições reais, sem implantar o sistema em uma infraestrutura de rede existente. A abordagem adotada neste trabalho, parte da implantação de dois sistemas P2P em uma rede institucional, tendo como foco investigar o desempenho dos principais componentes considerando:

- Economia de banda no servidor em função do número de clientes na transmissão de um determinado canal ao vivo. Essa métrica concentra-se na perspectiva do servidor justificando-se pelo fato que os pares da rede P2P auxiliam o servidor na distribuição de conteúdo;
- Qualidade do vídeo recebido através de métricas de qualidade de experiência do usuário (QoE). A perspectiva do cliente P2P é avaliada a partir de medidas como *Peak Signal-to-Noise Ratio* (PSNR), quando um sinal é comparado antes e depois de um determinado processo de degradação (i.e. distribuição do vídeo). Os experimentos conduzidos no trabalho envolvem a captura do vídeo recebido pelos usuários, e a geração de tráfego para avaliar a sensibilidade ao congestionamento na rede.

Ressalta-se também que os sistemas P2P existentes para *Video Streaming* são fechados, o que leva os trabalhos de medições a observarem de modo passivo os compromissos nesses sistemas. No entanto, recentemente um sistema *P2P Video Streaming* de código aberto, batizado de Goalbit, foi desenvolvido oferecendo a possibilidade de analisar mais detalhes do sistema explorando seu código fonte. Por essa razão, o Goalbit foi escolhido para ser o sistema base que será comparado face a um sistema proprietário como o SopCast.

O artigo será apresentado de acordo com a seguinte estrutura: na Seção 2 são citados alguns trabalhos relacionados descrevendo as contribuições em Sistemas P2P de *Video Streaming*. Na Seção 3 é abordado o funcionamento do SopCast e do Goalbit, levando-se em consideração as características, arquiteturas e protocolos de comunicação. Na Seção 4 descreve-se a metodologia utilizada nos experimentos. São descritas as ferramentas utilizadas e a infraestrutura montada para a realização dos experimentos. Em seguida, na Seção 5, são descritos os experimentos realizados, são analisados os

[1] http://www.synacast.com/en/

[2] http://www.coolstreaming.us/

[3] http://www.pps.tv/

[4] http://www.sopcast.com/

resultados obtidos e o desempenho das duas aplicações é confrontado. Por fim, na Seção 6, são exibidos os principais resultados e as perspectivas de trabalhos futuros.

2. Trabalhos Relacionados

Com objetivo de compreender o que vem sendo estudado, são apresentadas a seguir as contribuições dos principais trabalhos relacionados.

Hei et al. (2007) realizaram um estudo de medição sobre o aplicativo PPLive. Como o protocolo do PPLive é proprietário, os autores criaram um *crawler* capaz de simular um *peer* para receber informações que permitiriam compreender o funcionamento do sistema. Com o *crawler,* eles analisaram a dinâmica dos usuários e tiveram uma visão global da rede. Os resultados mostraram que o comportamento dos usuários de P2P IPTV é similar ao dos usuários da TV tradicional. Outra observação é que alguns *peers* do PPLive atuam como *super peers* contribuindo significantemente para *uploading* dos dados de vídeo. Entre os problemas verificados estão os atrasos e *lags* no início das transmissões que variam de alguns segundos até alguns minutos.

Sentinelli et al. (2007) apresentam um conjunto de experimentos sobre o SopCast e diretrizes para implementação de sistemas *P2P Video Streaming*. Os estudos descobriram, por exemplo, que um cliente se conecta geralmente de 2 a 5 *peers* para baixar os pedaços do vídeo. Com relação ao tamanho do buffer, os autores afirmam que este equivale a 1 minuto de vídeo. Os atrasos para inicialização dos vídeos apresentaram um tempo médio de 56 segundos para uma rede universitária e 111 segundos para uma rede residencial. Os autores ainda realizaram uma comparação de desempenho entre o SopCast e os resultados encontrados por trabalhos sobre o PPLive, CoolStreaming e Anysee.

Gonçalves et al. (2011) também avaliaram o SopCast, porém sob uma perspectiva das propriedades dinâmicas da rede. A fim de encontrar perfis de nós mais comuns, os autores submeteram os resultados encontrados ao algoritmo de agrupamento k-means, o qual localizou três perfis de nós. As alterações das centralidades dos nós ao longo do tempo foram mapeadas através da técnica de modelagem *Customer Behavior Model Graph* (CBMG). Com essa técnica, foi observada que a probabilidade de um nó permanecer em um mesmo perfil é bem alta. Já a probabilidade de um nó diminuir sua centralidade é maior que a probabilidade dele aumentar. Com relação à estrutura da rede, foi observado que o diâmetro e o caminho mínimo se mantêm estáveis, enquanto grau máximo e coeficiente de agrupamento não. Os nós tendem a manter em média 70% das parcerias estabelecidas ao longo do tempo.

Lu et al. (2009) relatam um estudo focado na qualidade de experiência de usuários do SopCast. Neste trabalho foram obtidas respostas sobre: qualidade do vídeo, bloqueios, sincronização de áudio e vídeo, nível de sincronização entre os nós e tempo para troca de canal. Verificou-se, por exemplo, que existe uma diferença média de 210 ms no tempo entre o áudio e o vídeo do SopCast, valor que excede a recomendação de 185 ms da ITU. Para medir a qualidade do vídeo assistido pelos clientes foi utilizada uma ferramenta chamada *Video Quality Metric* (VQM) que é capaz de medir deficiências no vídeo como, desfoque, movimentos irregulares e ruídos. O trabalho também contou com uma análise subjetiva realizada através de questionários aplicados a 22 pessoas voluntárias. Nesta análise a avaliação média final definiu como boa a qualidade dos vídeos.

Bertinat et al. (2009) propuseram o aplicativo Goalbit como um sistema P2P para *streaming* de vídeo. Além disso, os autores realizaram medições tanto do lado do servidor quanto do cliente. Uma das observações diz respeito à taxa de download e upload, cujos valores médios foram, respectivamente, 318 Kbps e 64 Kbps. A diferença entre os valores

é justificada por restrições de conectividade. Para a latência (período de tempo entre a requisição de um pedaço e sua recepção) notou-se um tempo médio de 10 segundos. Com relação a perdas de pedaços do vídeo, os valores encontrados nas medições foram inferiores a 0.03%, porcentagem possivelmente relacionada às configurações de ambiente adotadas. Os autores ainda compararam o consumo de banda do sistema com o consumo em uma arquitetura cliente-servidor. Os resultados mostraram uma economia de até 90% na distribuição de um streaming de 300 kbps para 62 usuários.

Existem também trabalhos que realizam simulações que permitem explorar novos algoritmos em redes P2P. Manola et al. (2011) utilizaram simulações para avaliar os ganhos no uso de uma política de pré-busca em sistemas P2P de vídeo sob demanda. O estudo buscou um cenário com a simulação completa da pilha TCP/IP incluindo tráfego de fundo, atrasos e perdas nos enlaces. Nos resultados, os autores verificaram que é possível obter uma economia de até 3,8 vezes com o uso da pré-busca. No que diz respeito ao impacto do tráfego de fundo, foi percebido que houve pouca influência do mesmo. Já o número de interrupções no vídeo assistido pelos clientes diminui bastante com o uso da política de pré-busca, o tempo decorrido entre vídeo parado até voltar a tocar, foi em geral menor que 0,2 segundos.

O presente trabalho se difere dos demais ao se utilizar de medições para determinar ganhos confrontando-se sistemas reais tais como Sopcast e Goalbit considerando i) a economia de banda no servidor e ii) a qualidade de experiência percebida pelos usuários. Além disso, a nosso conhecimento, não existem relatos na literatura de pesquisas que comparem os sistemas SopCast e Goalbit. Partindo do Goalbit, que é um sistema de código aberto, é interessante analisá-lo face à um sistema proprietário como o SopCast, para entender as decisões do sistema explorando seu código fonte.

3. Aplicações P2P de Transmissão de Vídeo ao Vivo

As aplicações P2P de transmissão de vídeo possuem grande importância devido à economia da banda do servidor que elas são capazes de proporcionar. Dentre esses sistemas, um dos que se transformou em sucesso e passou a ser utilizada por milhões de pessoas ao redor do mundo é o SopCast. O SopCast é um aplicativo proprietário no qual os clientes podem atuar assistindo os canais disponíveis ou criando seus próprios canais. Existem muitos estudos na literatura que buscam compreender seu desempenho assim como seu protocolo fechado. Outra opção de aplicação P2P para vídeo é o Goalbit. O Goalbit é um sistema disponibilizado em 2009, portanto, mais recente que seus concorrentes.

Nesta seção apresenta-se as principais características do SopCast e do Goalbit, como arquitetura, políticas e protocolos. Com isso, será possível entender como estas aplicações funcionam e quais são seus componentes básicos.

3.1. SopCast

O SopCast é um sistema para transmissão P2P de vídeo que nasceu dentro da Universidade Fundan na China. Ele é um aplicativo amplamente estudado na literatura, sendo sua popularidade (Google, 2012) um dos principais motivos deste interesse. Contudo, apesar de ser um software disponibilizado gratuitamente, seu protocolo fechado dificulta a compreensão do seu funcionamento. Isso fez com que diversos pesquisadores focassem seus estudos na compreensão deste protocolo (Sentinelli et al., 2007; Tang et al., 2009; Tang, 2010). A primeira observação levantada diz respeito à arquitetura do SopCast, baseada em malha.

Segundo Tang et al. (2009), o protocolo do SopCast utiliza conexões UDP (*User*

Datagram Protocol) para transportar os pedaços de um vídeo. A comunicação entre *peers* sempre é iniciada por um par de pacotes com 52 e 80 bytes, os quais são responsáveis por começar uma parceria. Os pacotes *keep-alive*, capazes de verificar se um *peer* ainda está participando da distribuição, possuem 42 bytes e são transmitidos com uma alta frequência. As requisições pelos pedaços de um vídeo são feitas com pacotes de controle de 46 bytes. Os vídeos podem ser divididos em *chunks* ou blocos com tamanho de 10 kbytes.

A estrutura do SopCast é dividida em três camadas: a camada provedora de conteúdo (PC), a primeira camada de *peers* e a segunda camada de *peers* (Tang, 2010). A camada provedora de conteúdo contém apenas o nó que possui o vídeo e que irá transmiti-lo. Esse nó se conecta com um número limitado de *peers* que são os presentes na primeira camada. Os demais *peers* localizados na segunda camada se comunicam apenas com os *peers* da primeira camada e entre eles mesmos.

O primeiro passo para o SopCast iniciar o funcionamento é feito através de um contato entre cliente e *tracker*. Neste contato, o *tracker* repassa uma lista de nós com os quais o cliente poderá se comunicar para a troca de pedaços. Sentinelli et al. (2007) inferiram que o protocolo do SopCast é dirigido a pedaços (*chunk-driven*), assim como no BitTorrent. Estes pedaços possuem um tamanho de 1362 bytes, valor observado através de gráficos com a distribuição de tamanho dos pacotes. Sentinelli et al. (2007) também notaram que, quando o transmissor do vídeo é colocado em uma rede de alta velocidade, os clientes exploram sua capacidade realizando mais download diretamente dele, a fim de reduzir os atrasos.

No SopCast os usuários são capazes de consumir e produzir conteúdo. Para consumir basta que o cliente se conecte a um canal e aguarde as trocas de mensagens para localização de *peers* e, em seguida, o carregamento do *buffer*. A produção de conteúdo é feita a partir da criação de um canal. Os clientes podem submeter vídeos nos formatos: Windows Media Video (wmv), Real Media Variable Bitrate (rmvb), Real Media (rm) e Advanced Streaming Format (asf) (Lu et al., 2009).

3.2. Goalbit

O Goalbit é a primeira opção de sistema P2P para transmissão de vídeo com código aberto. O aplicativo permite que os vídeos sejam capturados utilizando diferentes tipos de aparelhos, assim como aceita codificações em diversos formatos: mpeg2/4-avc, vc-1, acc, vorbis, mpga e wma.

A arquitetura do Goalbit é baseada em quatro componentes. São eles: o *brodcaster*, o *peer*, o *super-peer* e o *tracker*. O *broadcaster* é uma peça que desempenha um papel crítico na arquitetura. Ele é o responsável por receber, processar e disponibilizar um vídeo na rede P2P, e qualquer problema nele causa o fim da transmissão. O *super-peer* é um *peer* com capacidade de upload elevado e alta disponibilidade. Ele se conecta diretamente ao *broadcaster* com o objetivo de auxiliar no início da distribuição do conteúdo. O *peer* é o usuário final da rede, o qual apresenta um comportamento variado com alta frequência de chegada e saída. Por fim, o *tracker* é o responsável por manter a lista de *peers* que estão participando de determinada transmissão. Toda vez que um *peer* se conecta em um canal, ele se registra no *tracker* que por sua vez retorna para ele a lista dos *peers* conectados ao canal. O papel deste componente é fundamental na arquitetura do Goalbit, já que sem ele torna-se impossível a entrada de novos *peers* na rede.

O processo para transmissão de vídeos no Goalbit passa inicialmente por um encapsulamento definido pelo *Goalbit Packetized Stream* (GBPS) (Bertinat et al., 2009). O GBPS especifica o conteúdo e a estrutura de cada parte do vídeo. Além da encapsulamento, o Goalbit precisa definir como as partes do vídeo serão transmitidas.

Para isso o protocolo *GoalBit Transport Protocol* (GBTP) foi implementado. Este protocolo é baseado no *BitTorrent Protocol*, com modificações para otimizar a transmissão de vídeo. Com o GBTP, os pedaços do vídeo são identificados por um número único chamado de ID. Este número é utilizado na troca de pedaços entre os *peers* e seus valores variam de 0 até 2^{31} (Barrios et al., 2011). O protocolo também define que cada *peer* tenha uma janela deslizante. Os pedaços do vídeo contidos nesta janela podem ser baixados e compartilhados pelo *peer*. Existem ainda outros conceitos relacionados à janela deslizante, entre eles, o índice de execução, o *buffer* ativo e o índice do *buffer* ativo. O índice de execução registra o ID do próximo pedaço que será requisitado pelo player. O *buffer* ativo é definido pela sequência de pedaços consecutivos que um *peer* possui, iniciando do índice de execução. Por fim, o índice do *buffer* ativo registra o ID do último pedaço de vídeo dentro do *buffer* ativo.

O GBTP também determina como as comunicações entre *peer* e *tracker* e entre *peer* e *peer* devem ocorrer. A comunicação *tracker-peer* é baseada em dois tipos de mensagens (inicial e periódica) através das quais os *peers* recebem listas de outros *peers* com quem podem se conectar. Nestas listas o *tracker* procura unir *peers* geograficamente próximos. A comunicação *peer-to-peer* utiliza mensagens para trocar informações sobre o contexto de cada *peer* e mensagens para trocar pedaços do vídeo. As conexões são estabelecidas de acordo com o tipo do *peer* e sua janela deslizante. A fim de manter os *peers* informados, são trocadas mensagens sempre que chegam novos pedaços ou quando a janela deslizante é atualizada. Quando um *peer* para de receber mensagens de outro, ele manda uma mensagem de verificação. Caso o par não responda, a conexão é encerrada. As trocas de pedaços de vídeo iniciam após um par demonstrar interesse por determinada parte. Além disso, existe a possibilidade do *peer* realizar o pedido de um bloco (*slices*).

O protocolo GBTP é capaz de aplicar duas estratégias de seleção (Bertinat et al., 2009). Estas estratégias são a Seleção de Peer e a Seleção de Pedaço. A Seleção de Peer define uma técnica para escolher quais *peers* estarão aptos a fazer download. A técnica, chamada *tit-for-tat*, concede a preferência aos *peers* que mais forneceram pedaços. Os *peers* recém-chegados não são totalmente prejudicados porque existe uma política para integrá-los a rede. A Seleção de Pedaços diz respeito à estratégia responsável por definir qual pedaço do vídeo será requisitado. Nesta estratégia o Goalbit combina duas técnicas, o *rarest-first* com o *greedy*. O *buffer* é dividido em três faixas: *urgent*, *usual* e *future*. Dessa forma, a preferência será pela busca de pedaços da faixa *urgent*. Se todos os pedaços desta faixa foram baixados, a seleção do próximo pedaço é realizada através de uma amostragem de variável aleatória exponencial que dará um peso maior para a faixa *usual* sem descartar a possibilidade de selecionar pedaços da faixa *future*.

4. Metodologia

Com o objetivo de avaliar e comparar o desempenho dos aplicativos SopCast e Goalbit, a metodologia adotada baseia-se em uma rede experimental semi-controlada composta por um servidor e 36 computadores clientes. A rede é definida como semi-controlada porque ela faz parte de uma infraestrutura institucional do departamento de informática da UFES que compartilha recursos competindo com o tráfego de produção dos usuários. A estrutura física da rede pode ser observada na Figura 1.

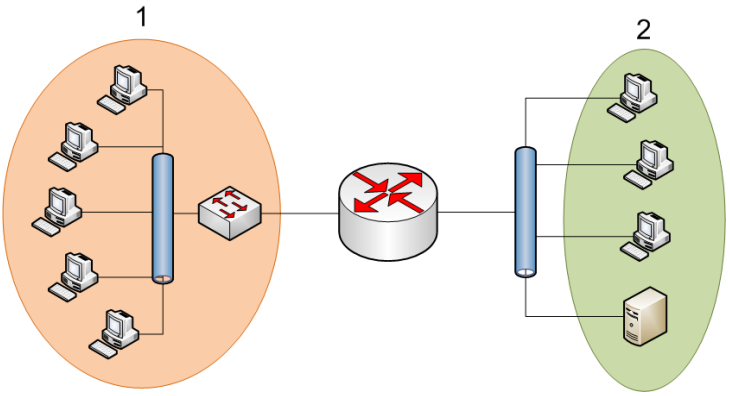


Figura 1. Estrutura física da rede experimental

As máquinas foram fisicamente alocadas em dois laboratórios que se comunicam através de um roteador central em uma topologia *Butterfly*. No Laboratório 1 estão distribuídas 28 máquinas, enquanto no Laboratório 2 estão o servidor e mais 8 máquinas. A capacidade do enlace da rede dos laboratórios é de 100 Mbps.

Antes de iniciar os experimentos foram necessárias algumas configurações nas máquinas. Dessa forma, foram instalados em todos os computadores clientes e no servidor os aplicativos: SopCast (Versão 3.4.0), Goalbit (Versão 0.7.7) e a ferramenta de análise de tráfego de rede, Wireshark (Orebaugh et al., 2006). No caso dos clientes, o Wireshark auxilia no diagnóstico de possíveis problemas. Já no servidor seu papel é registrar informações sobre o consumo médio da banda de upload. Para alcançar este objetivo ele foi pré-configurado.

A tarefa de avaliar e comparar os sistemas de transmissão P2P de vídeo pode ser feita de diversas maneiras. Neste artigo, o foco da análise busca verificar as diferenças tanto do ponto de vista da economia de banda no servidor quanto da qualidade dos vídeos assistidos pelos clientes. Além disso, verifica-se o impacto de um tráfego de fundo intenso na qualidade de experiência dos usuários.

A avaliação da economia de banda no servidor pode ser medida porque um ambiente semi-controlado permite administrar o número de *peers* conectados. Além disso, os dois aplicativos comparados possuem a opção de criar canais para *streaming* de vídeo. Dessa forma, para analisar o consumo de banda foi definido no servidor um canal para cada aplicativo. Isso possibilitou realizar testes de modo a acompanhar a variação da banda consumida com o aumento do número de clientes conectados ao canal. Assim é possível obter uma comparação justa dos dois *software* em questão.

A qualidade de experiência dos usuários pode ser analisada através de métricas objetivas, como por exemplo, *Peak Signal-to-Noise Ratio* (PSNR), *Mean Sum Absolute Difference* (MSAD), Delta, *Video Quality Metric* (VQM) e *Mean Squared Error* (MSE) (Bovik (2000), Wang (2006), Wang & Bovik (2009), MSU (2012)). Uma das técnicas amplamente adotadas na literatura é o PSNR (Winkler & Mohandas, 2008). O PSNR analisa o maior valor do sinal em relação à potência do ruído quando comparados um sinal antes e depois de um determinado processo de degradação. A unidade de medida utilizada por esta técnica é o decibel (dB). Winkler & Mohandas (2008) apresentam algumas razões da popularidade desta técnica, entre elas, está a sua formula simples de ser implementada e o cálculo rápido dos resultados.

Neste artigo, para avaliar a qualidade, foi escolhido uma variação desta métrica chamada APSNR (*Average* PSNR). A vantagem dela é a precisão alcançada quando os *frames* do vídeo possuem complexidade semelhante. O APSNR é calculado da seguinte forma:

$$APSNR = \frac{1}{N}\sum_{t=1}^{N} 10log\left(\frac{255^2}{mse_t}\right) \quad (1)$$

$$mse_t = \frac{1}{W.H}\sum_{x=1,y=1}^{W,H}\left(F'_{t,x,y} - F_{t,x,y}\right)^2 \quad (2)$$

Sendo:

N = Número de frames
W = Largura do frame
H = Altura do frame

O APSNR dos vídeos foi calculado utilizando a ferramenta *Elecard Video Quality Estimator* (Elecard, 2012). Esta ferramenta disponibiliza também outras métricas para análise de qualidade de experiência, como: *New Quality Index* (NQI), VQM, *Structural Similarity* (SSIM) e Delta. A Figura 2 apresenta um exemplo desta ferramenta em funcionamento.



Figura 2. Exemplo de análise comparativa entre dois vídeos na ferramenta *Elecard Video Quality Estimator*

Para fazer a análise com a ferramenta, foram gravados os vídeos assistidos pelos clientes durante os experimentos. Após a gravação selecionou-se amostras para serem comparadas com o vídeo original. Por fim, este mesmo teste de QoE foi repetido com tráfego de fundo intenso na rede, gerado através do *software* Iperf (Iperf, 2012). Dessa forma, foi possível realizar a avaliação da resiliência dos sistemas P2P, mediante as condições adversas.

Na próxima seção são apresentados os resultados obtidos durante os experimentos e discutidos os valores encontrados.

5. Experimentos

Nesta seção apresentam-se os resultados obtidos a partir dos experimentos realizados. Dessa forma, as informações foram divididas em duas subseções. Na subseção 5.1 discute-se resultados relacionados à banda de *upload* no servidor. Já na subseção 5.2 avalia-se a qualidade de experiência do usuário para os vídeos transmitidos pelas aplicações estudadas.

5.1. Banda de *upload* no servidor

Os testes de banda de *upload* no servidor foram realizados com vídeos codificados em duas taxas comumente encontradas na distribuição de vídeos online, 300 e 600 kbps. Para cada medição, depois de conectado o devido número de clientes ao canal, aguarda-se um intervalo de tempo para que a conexão se tornasse estável com os clientes exibindo os

vídeos. O consumo de banda foi analisado variando o número de clientes de 1 a 36, sendo 36 o número máximo de máquinas existentes nos laboratórios. Foram realizadas três campanhas de medições para as duas qualidades de vídeo.

As Figuras 3 e 4 apresentam, para cada curva, a média das três campanhas de medições realizadas. Pode-se observar que ambos os *software* propiciam uma grande economia de banda de *upload* no servidor em relação ao modelo cliente-servidor. Para calcular a economia de banda, supondo uma arquitetura hipotética Cliente-Servidor, a banda total agregada seria de 36x600 kbps = 21,6 Mbps na Figura 3, e 36x300 kbps =10,8 Mbps na Figura 4.

Considerando o sistema Goalbit com 36 usuários conectados, a banda foi de 7 Mbps (Figura 3) e 2,8 Mbps (Figura 4), o que corresponde a aproximadamente 32% da banda para o vídeo de 600 kbps e 25% da banda para o vídeo de 300 kbps, ou basicamente uma redução de 3,12 vezes e 3,85 vezes respectivamente.

É possível inferir a partir dos gráficos, que o Goalbit apresenta um aumento do consumo de banda aproximadamente linear conforme aumenta o número de clientes conectados ao canal de *streaming*. O fato do aumento ser linear, sugere uma concentração do envio de informações de controle, dado que a distribuição de conteúdo é baseada no TCP. Ainda assim esse crescimento é significativamente menor que o esperado de uma distribuição de vídeo utilizando o modelo cliente-servidor, o que permite uma elevada economia de banda percentual conforme aumenta o número de clientes.

O SopCast apresentou pouca economia de banda enquanto havia menos de 12 clientes conectados ao canal. Porém, acima de 12 clientes, o aumento na banda consumida cessa. Este resultado demostra um sistema descentralizado de controle da rede, não apresentando *overhead* de fluxo de controle no servidor. O SopCast promoveu uma economia de 2,84 vezes para o vídeo de 300 kbps (banda agregada de 3,8 Mbps Figura 4) e 3,32 vezes para o vídeo de 600 kbps (banda agregada de 6,5 Mbps Figura 3), em relação ao modelo cliente-servidor.

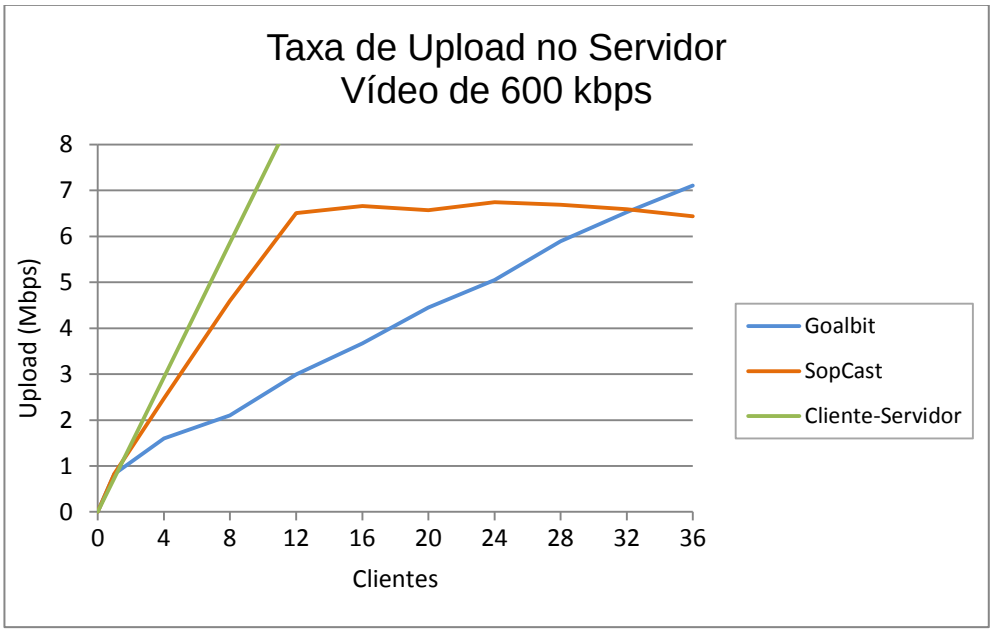


Figura 3. Banda de Upload no servidor para um vídeo de alta qualidade (600kbps)

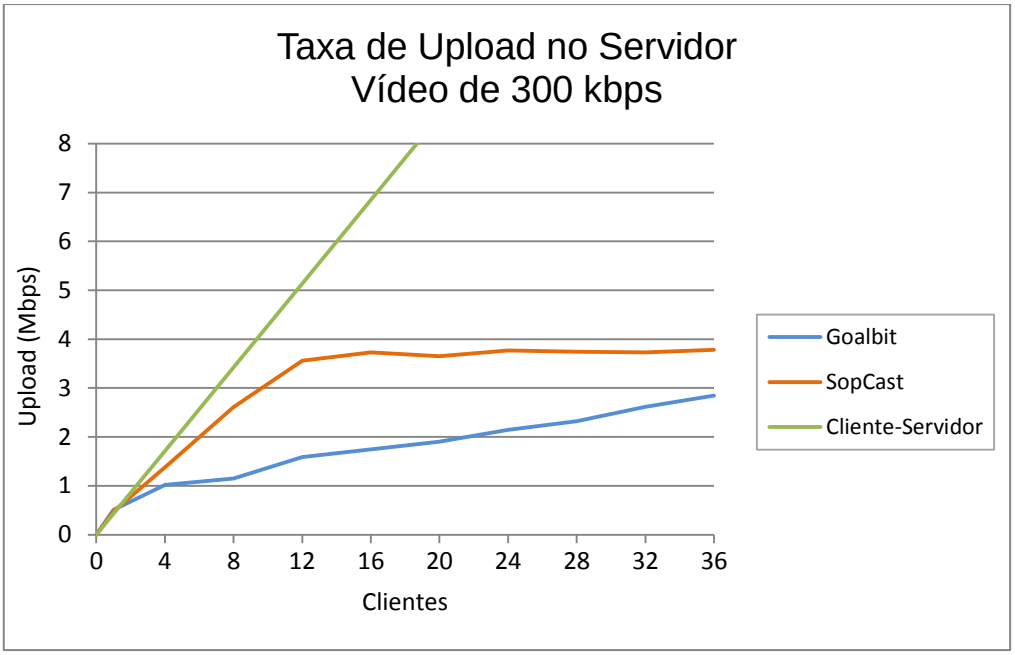


Figura 4. Banda de Upload no servidor para um vídeo de baixa qualidade (300kbps)

Pode-se concluir que, com relação à economia de banda no servidor, as duas aplicações apresentaram um resultado satisfatório. O Goalbit se mostrou mais eficiente em uma transmissão para poucos clientes. Conforme aumenta o número de clientes conectados ao canal, o Goalbit passa a demandar mais banda de *upload* que o SopCast. Isso se deve ao fato do SopCast atingir um *upload* aproximadamente constante após um certo número de clientes conectados. No experimento da Figura 4, com uma taxa de vídeo de 300 kbps, o SopCast não chega a consumir menos banda que o Goalbit. Mesmo assim é visível que, aumentando ainda mais o número de clientes, o SopCast tende a ser mais econômico.

5.2. Qualidade do vídeo recebido

Os testes para a verificação da qualidade do vídeo recebido foram feitos utilizando 36 clientes conectados aos canais do SopCast e do Goalbit. Para análise, selecionou-se uma amostra dos vídeos recebidos por quatro clientes. Dessa forma, foram 16 gravações, levando em conta a avaliação com e sem tráfego de fundo nas duas aplicações. A taxa do vídeo usado nos testes foi de 600 kbps e a quantidade de tráfego de fundo gerado foi de 90 Mbps. Os resultados obtidos são apresentados nos gráficos das Figuras 5 e 6. Na Figura 5, são apresentados os valores médios da métrica APSNR considerando os quatro clientes e sem tráfego de fundo. No caso da Figura 6, são apresentados os resultados com tráfego de fundo. A comparação considerou a média dos quatro vídeos capturados pelo SopCast, entretanto, no caso do Goalbit, devido a limitações da aplicação discutidas a seguir, são apresentados apenas os dados referentes a um dos clientes da amostra.

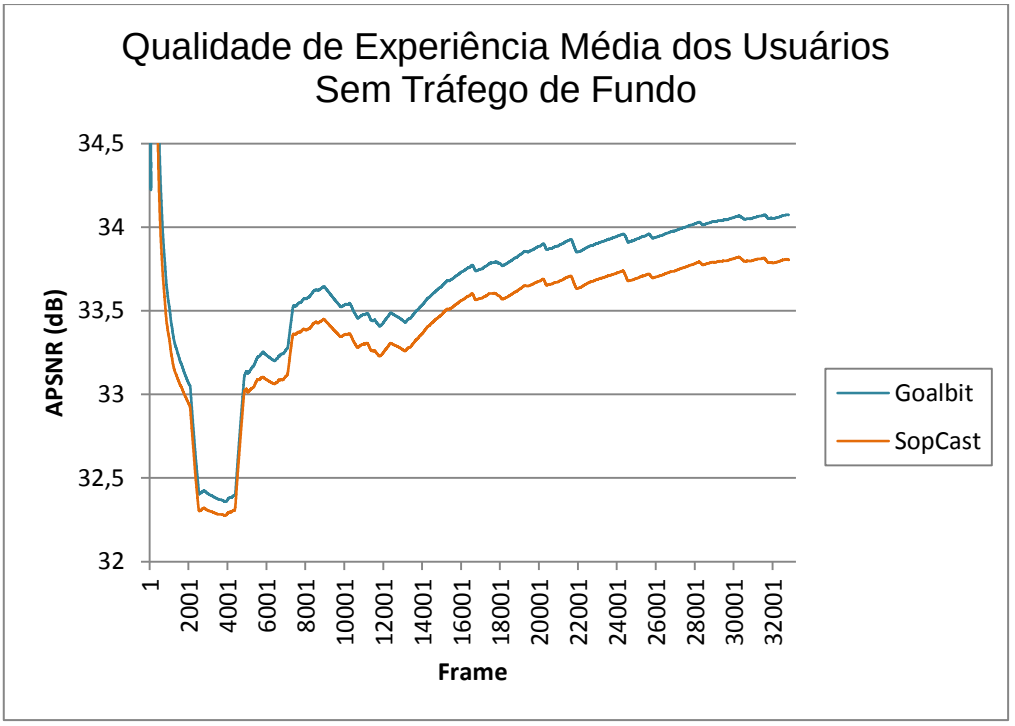


Figura 5. Comparação da qualidade de experiência do usuário sem tráfego de fundo

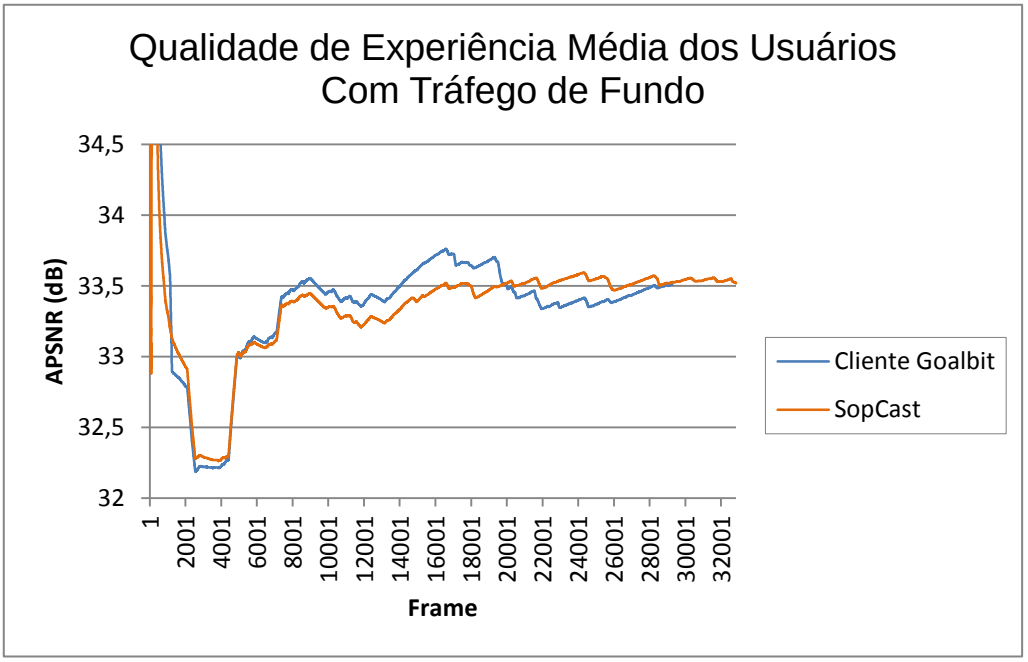


Figura 6. Comparação da qualidade de experiência do usuário com tráfego de fundo

A Figura 5 revela que no Goalbit a qualidade do vídeo recebido pelos clientes foi superior em relação ao mesmo vídeo transmitido pelo SopCast. Zinner et al. (2010) mostraram que valores acima ou igual a 33 dB no PSNR representam boa qualidade de experiência. Nos resultados, tanto o Goalbit quanto o SopCast mantiveram-se acima de 32 dB ao longo de toda a transmissão do vídeo, demonstrando uma qualidade de experiência satisfatória.

No caso do experimento com tráfego de fundo (Figura 6), os dois *software* apresentaram uma queda sutil na qualidade do vídeo transmitido. Apesar dessa queda, em

ambos o volume de dados no *buffer* foi mais afetado do que a qualidade da imagem propriamente dita. Observou-se, por exemplo, que em alguns momentos o *buffer* do SopCast chegou a ficar quase vazio, mas logo voltava a encher.

Outra constatação referente ao SopCast no experimento com tráfego de fundo é que a banda de *upload* requerida do servidor aumentou de 6,5 Mbps para cerca de 8,5 Mbps. Esse aumento é uma estratégia do SopCast para compensar a perda de pacotes com o congestionamento da rede. Como o SopCast opera sobre UDP, o aumento do *upload* no servidor sugere a existência de algum mecanismo de garantia de qualidade no nível da aplicação. O Goalbit, por operar em cima do protocolo TCP foi mais afetado, causando paradas constantes na exibição dos vídeos. O TCP tem a característica de ser um protocolo que busca ajustar a banda de modo igualitário entre os fluxos. Com isso, em situações de congestionamento, o tráfego TCP não utiliza indiscriminadamente a capacidade de encaminhamento disponível, isso porque ela é dividida com outros fluxos de dados. O experimento retratado na Figura 6 mostrou que esta característica prejudicou o Goalbit.

As frequentes paradas na exibição dos vídeos no Goalbit afetaram também suas capturas. Quando um vídeo para de ser exibido, a gravação do mesmo é finalizada. Dessa forma, as amostras foram prejudicadas e apenas um dos vídeos gerados obteve um tempo de gravação próximo do vídeo original. Com isso, ao invés de apresentar a média das amostras do Goalbit na Figura 6, foram inseridos apenas os valores da melhor captura. Nessa captura, entre os *frames* 28 mil e 30 mil, nota-se que a linha indicativa desaparece. Neste momento a gravação foi interrompida, pois o buffer se esgotou e foi necessário ser recarregado. Assim, para redes limitadas, o Goalbit pode ser prejudicado pelo uso do TCP. Apesar do TCP garantir a entrega dos pacotes, ele pode não ser o protocolo mais adequado para transmissão de vídeo ao vivo nesta situação.

Uma última observação está relacionada à compactação dos vídeos no servidor antes de serem transmitidos. Notou-se que o Goalbit adota um mecanismo para reduzir a qualidade do vídeo quando são criados canais com *bitrate* elevados (3000 kbps). Essa característica do Goalbit é interessante quando pensamos na transmissão de vídeo em sistemas de recursos limitados, como *smartphones* e *tablets*. O SopCast não apresentou um mecanismo semelhante e, além disso, não conseguiu transmitir vídeos com alto *bitrate*.

6. Conclusões e Trabalhos Futuros

Neste artigo foram avaliados e comparados dois sistemas de *P2P Video Streaming* com objetivo de analisar a economia de banda no servidor e a qualidade de experiência dos usuários. O trabalho foi motivado pela possibilidade de explorar o desempenho do sistema de *P2P Video Streaming* Goalbit. Para isso, comparou-se ele com um dos aplicativos mais conhecidos e utilizados neste meio, o SopCast.

Em relação ao servidor, foi possível observar que ambos os *software* propiciaram uma boa economia da banda de upload em relação ao modelo cliente-servidor. O Goalbit proporcionou uma redução de 3,12 vezes para o vídeo de 600kbps e de 3,85 para o vídeo de 300kbps. Já o SopCast obteve uma redução de 3,32 vezes para o vídeo de 600kbps e de 2,84 vezes para o vídeo de 300kbps. Os resultados também mostraram que o Goalbit é uma opção mais eficiente nas transmissões de vídeo para poucos clientes (até 32). No entanto, o SopCast tende a ser mais econômico com o aumento do número de *peers*. Contudo, em trabalhos futuros ainda podem ser explorados outros recursos do Goalbit, como os *super-peers*, além de melhorias na sua implementação.

A qualidade do vídeo recebido foi medida utilizando a métrica APSNR. Os valores obtidos mantiveram-se na maior parte do tempo entre 33 e 35 dB, representando um QoE considerado satisfatório. A qualidade dos vídeos recebidos no Goalbit foi sutilmente

superior em relação aos vídeos do SopCast. Congestionando a rede com tráfego de fundo, os dois sistemas sofreram uma queda na qualidade. O Goalbit foi mais prejudicado, neste caso, devido ao uso do protocolo TCP para a transmissão. Além disso, percebeu-se que o Goalbit utiliza um mecanismo de compressão, no caso de vídeos com *bitrate* muito elevado. Esta característica pode ser explorada de modo a estender o Goalbit para equipamentos com recursos limitados. Já o SopCast, com a inserção do tráfego de fundo, apresentou um aumento na banda de *upload* do servidor, buscando compensar a perda de pacotes devido ao congestionamento da rede.

Por fim, os resultados apontam o Goalbit como uma alternativa promissora se comparado a um sistema amplamente utilizado, como o SopCast. Mesmo assim, o Goalbit ainda é um aplicativo recente, que pode ser melhorado em aspectos como, por exemplo, a otimização da distribuição da carga do servidor, no caso de um elevado número de clientes. Entre os trabalhos futuros destacam-se:

- Comparar o Goalbit com outros sistemas *P2P Video Streaming*;
- Aplicar outras métricas de QoE tanto objetivas, como o PSNR, quanto subjetivas, onde os usuários avaliam os vídeos assistidos;
- Explorar o código fonte do Goalbit a fim de propor melhorias relacionadas aos pontos fracos do sistema;
- Explorar a capacidade do Goalbit de compactação dos vídeos para entregar conteúdo de acordo com a capacidade dos clientes.

7. Agradecimentos

Agradecemos à CAPES – Brasil pelo financiamento parcial deste projeto (Projeto RH-TVD #225/2008), e ao programa PET (Roberta Lima Gomes é tutora do PET Eng. de Computação/UFES; Alan Floriano, Ivan Nunes e Ebenezer Silva são bolsistas). Esta pesquisa também foi financiada pela FAPES e pela FINEP (Projeto CPID-CIDIG).

7. Referências

Barrios, A., Barrios, M., Vera, D., Rodríguez-Bocca, P., Rostagnol, D. (2011). “GoalBit: A Free and Open Source Peer-to-Peer Streaming Network”. International Conference on Multimedia (ACM MULTIMEDIA 2011), Open Source Software Competition. Scottsdale, Arizona, USA. November 28- December 1, 2011.

Bertinat, M., Vera, D., Padula, D., Robledo, F., Rodríguez-Bocca, P., Romero, P. e Rubino, G. (2009). “GoalBit: The First Free and Open Source Peer-to-Peer Streaming Network”, in: Proceedings of the 5th international IFIP/ACM Latin American conference on Networking (LANC’09), Pelotas, Brazil, 2009.

Bovik, A. (2000). "Handbook of Image and Video Processing", Academic Press, ISBN 0-12-119790-5 (2000).

Cisco (2011). “Cisco Visual Networking Index: Forecast and Methodology, 2010-2015”. Site: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-481360.pdf. Acessado em: 28 de Março de 2012.

Elecard (2012). Elecard Video Quality Estimator. Site: http://www.elecard.com/en/products/professional/analysis/video-quest.html. Acessado em: 8 de Março de 2012.

Gonçalves, K., Vieira, A., Almeida, J., Marques, H., Campos, S. (2011). “Caracterização das Propriedades Dinâmicas da Estrutura da Rede Sobreposta do SopCast”. In: X

Workshop em Desempenho de Sistemas Computacionais e de Comunicação (WPerformance 2011), 2011, Natal. Anais do X Workshop em Desempenho de Sistemas Computacionais e de Comunicação (WPerformance 2011), 2011.

Google. Google Trends. Site: http://www.google.com.br/trends/?q=sopcast,+pplive,ppstream,coolstreaming. Acessado em 17 de Fevereiro de 2012.

Hei, X., Liang, C., Liang, J., Liu, Y. and Ross, K.W. (2007) 'A measurement study of a large-scale P2P IPTV system', IEEE Transactions on Multimedia, Vol. 9, No. 8, pp.1672–1687.

Iperf (2012). Site: http://sourceforge.net/projects/iperf/. Acessado em: 9 de Março de 2012.

Lu, Y., Fallica, B., Kuipers, F., Kooij, R. e Mieghem, P. (2009). "Assessing the quality of experience of sopcast". Journal of Internet Protocol Technology, 4(1):11-23, 2009.

Manola, R., Martinello, M., Gomes, R. L., Marcondes, C. (2011). "Análise de Desempenho no Uso de Pré-Busca Para Distribuição de Vídeo Sobre Redes P2P". In: Workshop de Redes Dinâmicas e Sistemas P2P (WP2P'11), 2011, Campo Grande. Anais do VII Workshop de Redes Dinâmicas e Sistemas P2P. Porto Alegre : SBC, 2011. p. 3-16.

MSU (2012). "MSU Quality Measurement Tool: Metrics information". Site: http://compression.ru/video/quality_measure/info_en.html. Acessado em: 29 de Março de 2012.

Orebaugh, A., Burke, J., Pesce, L., Ramirez, G. (2006). Wireshark & Ethereal Network Protocol Analyzer Toolkit, Syngress Publishing, 2006.

Sentinelli, A., Marfia, G., Gerla, M., Kleinrock, L. e Tewari, S. (2007) "Will IPTV ride the peer-to-peer stream?", Communications Magazine, Vol. 45, No. 6, pp.86–92.

Silverston, T., Fourmaux, O., Botta, A., Dainotti, A., Pescapé, A., Ventre, G., e Salamatian, K. (2009) "Traffic analysis of peer-to-peer iptv communities," Comput. Netw., vol. 53, no. 4, pp. 470-484, 2009.

Tang, S. (2010). "Information Propagation in Peer-to-Peer Networking: Modeling and Empirical Studies". 2010. 170 f. Tese (Doutorado) - Delft University Of Technology, Delft, Holanda, 2010.

Tang, S., Lu, Y., Hernandez, J. M., Kuipers, F. e Mieghem, P. (2009). "Topology Dynamics in a P2PTV Network". In NETWORKING '09, Berlin, Heidelberg.

Wang, Y. (2006). "Survey of Objective Video Quality Measurements", Tech report, Worcester Polytechnic Institute, Jun. 2006.

Wang, Z. e Bovik, A. (2009). "Mean Squared Error: Love It or Leave It?", IEEE Signal Processing Magazine (2009).

Winkler, S. e Mohandas, P. (2008). "The evolution of video quality measurement: From PSNR to hybrid metrics," IEEE Transactions on Broadcasting, vol. 54, no. 3, pp. 660-668, September 2008.

Zinner, T., Abboud, O., Hohlfeld, O. e Hossfeld, T. (2010). "Towards QoE Management for Scalable Video Streaming". In 21th ITC Specialist Seminar on Multimedia Applications - Traffic, Performance and QoE, Miyazaki, Jap, 2010.