

**SUPER NÓS EM SISTEMAS P2P DE
DISTRIBUIÇÃO DE MÍDIA AO VIVO**

JOÃO FERREIRA D'ARAÚJO E OLIVEIRA

**SUPER NÓS EM SISTEMAS P2P DE
DISTRIBUIÇÃO DE MÍDIA AO VIVO**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

ORIENTADOR: SÉRGIO VALE AGUIAR CAMPOS

Belo Horizonte

Março de 2010

© 2010, João Ferreira D'Araújo e Oliveira.
Todos os direitos reservados.

Oliveira, João Ferreira D'Araújo e.

O48s Super nós em sistemas P2P de distribuição de mídia
ao vivo. / João Ferreira D'Araújo e Oliveira. — Belo
Horizonte, 2010.

xxiv, 67 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de
Minas Gerais. Departamento de Ciência da Computação.

Orientador: Prof. Sérgio Vale Aguiar Campos.

1. Redes de computadores - Teses. 2. Redes P2P - Teses.
3. Mídia contínua - Teses. 4. Super nós - Teses.
I. Orientador. II. Título.

CDU 519.6*22(043)



UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Super Nós em sistemas P2P de distribuição de mídia ao vivo

JOÃO FERREIRA D' ARAÚJO E OLIVEIRA

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:


PROF. SÉRGIO VALE AGUIAR CAMPOS - Orientador
Departamento de Ciência da Computação - UFMG


PROF. ALEX BORGES VIEIRA
Instituto de Informática - PUC-MG


PROF. ANTÔNIO MARINHO PILLA BARCELLOS
Instituto de Informática - UFRGS


PROFA. JUSSARA MARQUES DE ALMEIDA
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 18 de junho de 2010.

Dedico este trabalho especialmente aos meus pais, João e Telma, que sempre estiveram ao meu lado e me permitiram alcançar todos os destinos que persegui.

Agradecimentos

Gostaria primeiro de fazer um agradecimento metafísico: à vida... à existência, ao ar, ao éter, às idéias, aos sonhos, à massa e à energia. Todas essas coisas que fazem o mundo ser o que o mundo é assim como tudo que tem dentro e fora dele e que nos permitem ser o que somos e o que ainda não somos dado que um dia seremos de novo o mundo (se já não o somos).

Agradeço especialmente aos meus pais que durante décadas me nutriram de apoio, carinho, amor e cuidaram para que um dia eu pudesse abandonar o ninho e descobrir meu próprio mundo. Um salve especial para toda pessoa que sente falta dos pais, seja por que motivo for.

Agradeço muito à minha fiel escudeira, gatíssima, linda, simpática, inteligente, divertida e fofa, Nana. Sem você do meu lado, amor, nesses últimos dois anos eu talvez tivesse feito um mestrado mais rápido (um pouquinho só, tá?) mas infinitamente menos feliz. E o importante mesmo nessa vida é ter felicidade, todos os nossos objetivos apontam para esse máximo!

Agradeço ao meu orientador, Sérgio Vale Aguiar Campos, pelos puxões de orelha, pelas dicas fantásticas, pela dedicação, honestidade, paciência e incentivo e especialmente por compartilhar os ensinamentos sobre essa nossa vida acadêmica. Deixo a certeza de que muito foi ensinado (por ele!) e muito foi aprendido (por mim!). Estendo o agradecimento a Alex, Marcus, Pedro e Mantini, um grupo de pessoas que geram impacto para o Brasil só de existirem. Espero poder continuar trabalhando com vocês num futuro próximo.

Agradeço imensamente as componentes da minha banca, Profa. Jussara Almeida, Prof. Marinho Barcellos e Prof. Alex Borges (de novo!), que contribuíram com críticas fundamentais para aumentar a organização e clareza do trabalho. Podem ter certeza que as anotações foram significativamente úteis e sempre consideradas.

Agradeço aos amigos que estiveram aqui na UFMG, em BH, e que dividiram comigo do pior ao melhor, dos anseios às baladas. Guilherme, Alex, Dudu, Flávio, Paolo, Pedro, Celso, Manú, Vânia, Henrique, Eder, Famube, Leandro, Heitor, Biel,

Narlla, Thaís, Létch!

Agradeço aos velhos, mas nunca velhos de mais, amigos de Salvador: Edú, Danilo e Diego! E mando um salve pro grande Lechuga que sofreu em paralelo os desafios do mestrado.

Agradeço aos amigos virtuais sempre presentes do World of Warcraft! Infelizmente eu não tenho como citar em pouco mais que uma página as centenas de pessoas que conheci nesse meio então segue uma frase significativa para este grupo: É nós que voa bruxão!

Agradeço ao DCC e à UFMG pela grande universidade e pela oportunidade. Foi SEMPRE um prazer poder me dedicar à ciência apoiado pelo Brasil. Feliz sejam os pagadores de impostos (se bem que poderiam ser um pouquinho menos, né?)!

“La disciplina es la parte más importante del éxito.”

(Capote)

Resumo

Distribuição de mídia contínua através da Internet tem se mostrado nos últimos anos uma aplicação extremamente interessante. A utilização de redes par-a-par (P2P) para difusão desse conteúdo traz vantagens acerca da escalabilidade do número de usuários, tolerância a falhas e diminuição de custos, contudo, problemas relativos à segurança e qualidade tanto da rede quanto do conteúdo contribuem para uma ainda baixa adoção da tecnologia no cotidiano. Neste trabalho o estudo das redes P2P de distribuição de mídia contínua levou à identificação de um elemento especialmente importante na rede, o super nó. Com foco nessa característica realizam-se discussões, estudos e proposições acerca do potencial do super nó, da sua formação e de como aproveitá-lo a favor de uma topologia que reduza os efeitos negativos do uso de P2P. Os resultados aqui apresentados demonstram, especialmente, a extensa influência dos super nós sobre a segurança e qualidade de serviço da rede. Experimentos em ambiente real mostraram que um único super nó pode, através somente dos dados que envia, influenciar mais de 30% da banda de *download* da rede e mais de 50% dos pares. Essa característica associada à poluição de conteúdo pode ser desastrosa para o sistema. Em simulação foi verificado que a composição da rede em duas camadas (sendo uma de usuários comuns e outra de super nós confiáveis, presentes e robustos), comparado ao esquema original, é capaz de reduzir a latência e elevar a continuidade da mídia em situações de alto *churn* e reduzir em mais da metade a quantidade média de poluição trafegada na rede para um ataque combinado de 10% dos participantes. Acredita-se que a arquitetura proposta pode ser aplicada com sucesso à nova geração de sistemas P2P de distribuição de mídia contínua, agregando valor à mesma.

Abstract

In the past few years live streaming networks have become a killer application over the Internet. Use peer-to-peer (P2P) networks to broadcast live content has some advantages such as scalability in number of users, fault tolerance and cost reduction, however, problems relating to safety and quality of both the network and the content contribute to low usage of that technology in a daily basis. In this work the study of P2P live streaming networks has led to identification of a particularly important element in the network, the super node. Focusing on this feature we held discussions, studies and proposals over the potential of the super node, its appearance and how to use this in order to create topologies which reduces the negative effects of using P2P. The results presented here especially show the extensive influence of the super nodes on the safety and quality of service of the overlay. Experiments in real environment showed that a single super node can influence network's download bandwidth over 30% and over 50% of all users. This characteristic associated with content pollution would be disastrous for the system. In simulation it was observed that a two-layered topology, being one composed of preset super nodes, is able to reduce latency and to increase the stream continuity in situations of high churn. Besides, it is able to more than halve the average amount of network's pollution traffic for a combined attack of 10% of participants compared to the original layout. Thus, we believe that the proposed architecture can be valuable and successfully applied to next generation P2P live streaming systems.

Lista de Figuras

1.1	Tendência de busca para sistemas populares de P2PTV nos últimos 5 anos	2
2.1	Organização de sistemas em árvore	17
2.2	Organização de sistemas em malha	18
3.1	Tendência de busca para sistemas populares de P2PTV em 2009	24
3.2	Distribuição Cumulativa da Taxa Média de Upload dos Pares	28
3.3	Organização de sistemas baseados em malha considerando super nós	29
3.4	Percentual de banda de <i>upload</i> fornecida em função do percentual de pares de maior <i>upload</i> controlados	31
4.1	Distribuições Relativas aos Pacotes Marcados	36
4.2	Uso de Banda da Rede	37
4.3	Parcerias do Nó Marcador	38
4.4	Parceiros na Rede	39
5.1	Correlação entre ranks de Grau e Upload.	43
5.2	Correlação entre ranks de Closeness e Upload.	44
5.3	Correlação entre ranks de Distância e Upload.	44
5.4	Correlação entre ranks de (Distância,Grau) e Upload.	45
5.5	Correlação entre ranks de Betweenness e Upload.	45
5.6	Correlação entre ranks de Betweenness para Fonte e Upload.	46
6.1	Proposta de topologia em duas camadas	48
6.2	Medidas de continuidade. Rede representando o sistema: (a) original com diferentes modelos de geração de <i>churn</i> . (b) onde o comportamento de <i>churn</i> dos usuários é similar a um momento onde um evento de interesse está sendo transmitido variando a quantidade de super nós determinados.	55

6.3	Medições de latência para os três tipos de <i>churn</i> . Rede representando o sistema: (a) original; (b) com a presença de 1% de super nós determinados; (c) com a presença de 5% de super nós determinados.	56
6.4	Medições de poluição para ataque combinado de 10% dos nós comuns nas redes original (sem super nós) e em duas camadas (com 5% de super nós).	57

Lista de Tabelas

5.1	Coeficientes de Correlação de Pearson.	43
6.1	Sumário de Parâmetros do Simulador	52

Lista de Siglas

CDF	Função de densidade acumulada
CDN	<i>Content Distribution Network</i>
CES	<i>Consumer Electronics Show</i>
CPU	<i>Central Processing Unit</i>
DHT	<i>Distributed Hash Table</i>
IP	<i>Internet Protocol</i>
IPTV	<i>Internet Protocol Television</i>
NAT	<i>Network Address Translation</i>
P2P	Par-a-Par ou <i>Peer-to-Peer</i>
P2PTV	<i>Peer-to-Peer Television</i>
TV	Televisão
VOD	Vídeo sob demanda

Sumário

Agradecimentos	ix
Resumo	xiii
Abstract	xv
Lista de Figuras	xvii
Lista de Tabelas	xix
1 Introdução	1
1.1 Motivação	1
1.2 Objetivos	4
1.3 Resumo da Metodologia	5
1.4 Trabalhos Relacionados	6
1.5 Organização do Trabalho	9
2 Fundamentos Teóricos	11
2.1 Redes Par-a-Par	11
2.1.1 Definição quanto à Estrutura	12
2.1.2 Definição quanto à Centralização	14
2.2 Distribuição de Mídia Ao Vivo em P2P	15
2.2.1 Sistemas baseados em Árvore	16
2.2.2 Sistemas baseados em Malha	17
2.3 Redes Complexas	20
2.3.1 Propriedades dos Grafos	20
2.3.2 Modelos de Redes	20
2.3.3 Métricas de Centralidade	21
3 Conceituação do Super Nó	23

3.1	Ferramental para Captura de Dados	23
3.1.1	SopCast	23
3.1.2	PlanetLab	25
3.1.3	Simulador: OMNeT++ e OverSim	25
3.2	Metodologia de Captura de Dados	26
3.3	Distribuição de Taxas de Upload Médio	28
3.4	Super Nó	29
4	A Influência do Super Nó	33
4.1	Ataques de Poluição de Conteúdo	33
4.2	Metodologia de Marcação de Pacotes	34
4.3	Avaliação da Influência de Super Nós	35
4.3.1	Análise Preliminar	35
4.3.2	Análise do Cenário Realista	36
5	Caracterização de Super Nós através de Cálculos de Centralidade	41
5.1	Metodologia de Análise dos Dados	41
5.2	Análises de Correlação	43
6	Protocolo para Determinação de Super Nó	47
6.1	Contextualização	47
6.2	Simulador de P2PTV baseado em malha	49
6.2.1	Alterações nos algoritmos	50
6.3	Análise Experimental	51
6.3.1	Sumário dos Experimentos	51
6.3.2	Resultados	54
7	Conclusões e Trabalhos Futuros	59
	Referências Bibliográficas	61

Capítulo 1

Introdução

1.1 Motivação

Ao longo do último século o avanço da tecnologia vem gerando contínuas melhorias nos meios de transmissão de informações. Essa linha do tempo abrange telégrafo, rádio, televisão e, mais recentemente, computadores e a Internet.

Desde o seu surgimento, a Internet é utilizada para tráfego de dados entre computadores em escala global. Com o passar dos anos, a computação foi ficando mais complexa e poderosa, sua evolução natural tornou os objetos e mídias trafegados na rede maiores e, ao mesmo tempo mais populares à medida que a tecnologia foi se tornando mais acessível.

Atualmente, é possível observar uma migração intensa do conteúdo audiovisual, antes apenas presente no rádio e televisão, para esse meio. Contudo, a evolução da rede mundial de computadores não tem acompanhado suficientemente rápido as demandas atuais. Essa deficiência é marcada por diversos fatores, entre eles, um crescimento insatisfatório do tempo de resposta ao tentar resolver requisições de mídia ou diminuição da qualidade da mesma a fim de alcançar as restrições temporais.

Uma parcela significativa do problema está associada ao modelo de transmissão conhecido como cliente-servidor. Nesse cenário, onde existem muitos usuários interessados em mídias densas, como vídeos de alta qualidade, esse modelo se destaca negativamente pela sua baixa escalabilidade, onde os servidores de conteúdo acabam se tornando gargalos que afetam brutalmente o tempo de resposta.

De maneira alternativa, se cada interessado no conteúdo pudesse inicialmente consumi-lo como um cliente e, posteriormente assumir o papel de servidor, disponibilizando o objeto de desejo para novos interessados, seria possível distribuir a carga total entre todos esses "parceiros". Surge assim o termo que define esse tipo de transmis-

são: par-a-par (P2P), que se popularizou a partir de 1999 com sistemas como Napster [Shirky, 2001], Gnutella [Ripeanu, 2001], Emule [Kulbak & Bickson, 2005], BitTorrent [Cohen, 2003] ou Skype [Guha et al., 2006].

Muitas redes P2P com diversas motivações podem ser formadas como: compartilhamento de arquivo, bate-papo, telefonia e jogos [Cohen, 2003; Strauss & Schmidt, 2003; Knutsson et al., 2004; Guha et al., 2006]. Em especial, esse trabalho está direcionado ao estudo das redes P2P que realizam difusão de mídia contínua ao vivo, ou *live streaming*. Esta categoria permite, por exemplo, que uma partida de futebol seja transmitida na Internet no exato momento em que ocorre.

Redes P2P de distribuição de mídia contínua ao vivo, ou P2PTV, vêm se destacando ao longo dos últimos cinco anos pela sua ascensão e pelos números que movimentam sobre a Internet. Na figura 1.1 é possível observar esse comportamento através de estatísticas geradas no Google Trends¹ mostrando o volume relativo de busca entre alguns aplicativos dessa natureza. São diversos os sistemas que detêm diariamente centenas de milhares de usuários conectados, em média, a cada instante [Hei et al., 2007; Wu et al., 2007; Huang et al., 2008b; Sentinelli et al., 2007]. Muitos desses consumidores são provenientes de uma migração dos serviços de radiodifusão comuns para a Internet, um câmbio tecnológico natural. Alguns enxergam maior valor agregado no serviço como: maior variedade de escolha de conteúdo, potencial de interatividade e mobilidade a preços inferiores [CNET, 2010] e outros pertencem a uma nova classe de usuários que enxerga nesse contexto a possibilidade de criar e produzir programação e conteúdo próprios com baixo custo de distribuição [Sentinelli et al., 2007; Silva, 2009].

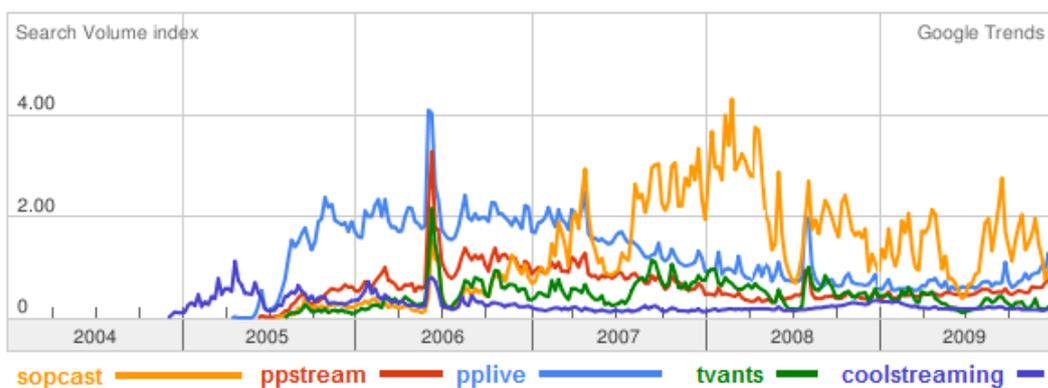


Figura 1.1: Tendência de busca para sistemas populares de P2PTV nos últimos 5 anos

Em vista do fato de que essa migração carece de um modelo de transferência suficientemente capaz de lhe dar suporte, a escalabilidade do sistema quanto ao número

¹<http://www.google.com/trends/>

de usuários e as taxas de transferência alcançáveis através da tecnologia de P2P, são peças fundamentais para aliviar a carga e despesas com servidores dedicados. Esta abordagem não é a única; existem outras formas de tratar um tráfego tão intenso e tais demandas por conteúdo [Hei et al., 2007], por exemplo, através da contratação de redes de entrega de conteúdo (*content delivery networks*, ou CDNs), como Akamai² e Limelight³, replicação de servidores e distribuição de carga. Verizon⁴, Comcast⁵ e Yahoo! Live⁶ são algumas das grandes empresas do setor que se encaixam nesse perfil tecnológico de distribuição de mídia. Todavia, essas alternativas demandam custos elevados para sua instalação e manutenção. Por exemplo, segundo Yen [2008], há 2 anos estimava-se que o YouTube⁷ gastava por volta de 1 milhão de dólares por dia para pagar somente pela banda utilizada na distribuição dos vídeos.

Entretanto, não só benefícios são oferecidos pelo uso de parcerias. Essa solução traz consigo deficiências que, se não tratadas corretamente, atrapalham a popularização do serviço. Alguns dos principais desafios estão relacionados com segurança, gerenciamento e desempenho, como o longo tempo de espera para iniciar a exibição, a rotatividade dos parceiros (*churn*), baixa resiliência a ataques e suscetibilidade a poluição de conteúdo. Melhorar a qualidade em distribuição P2P de mídia contínua ao vivo passa pelo refinamento desses problemas latentes.

De acordo com o painel *The Next Big Thing* proferido na *Consumer Electronics Show 2010* (CES 2010) [CNET, 2010], distribuição de conteúdo audiovisual pela Internet é uma tendência atual. É estimado por Greeson [2009] do *The Diffusion Group* que mais de dois terços dos consumidores de televisão paga (cabos e satélite) estão dispostos a utilizar a TV pela Internet, seja complementando o serviço atual ou substituindo-o integralmente. Por ser um mercado que movimentava 120 bilhões de dólares só nos Estados Unidos, entre propaganda e assinatura, esse é um cenário que não há como ser ignorado. Ainda, em Sentinelli et al. [2007], análises da indústria apontam que haverá cerca de 80 milhões de usuários em sistemas de distribuição de conteúdo audiovisual pela Internet no mundo todo por volta de 2011.

Esse câmbio tecnológico vem para mudar a maneira como os usuários escolhem, compartilham, recomendam e experimentam conteúdo. O desejo dos usuários e a tecnologia convergem para melhoria dos sistemas através da implementação de: mídias de alta definição, interatividade do espectador com o conteúdo e do conteúdo com o

²<http://www.akamai.com/>

³<http://www.limelightnetworks.com/>

⁴<http://www22.verizon.com/>

⁵<http://www.comcast.com/>

⁶<http://live.yahoo.com/>

⁷<http://www.youtube.com/>

ambiente e mobilidade para assistir a programação onde e quando quiser. Com isso, é previsível que o crescimento do consumo de banda torne-se em breve um problema para os servidores de conteúdo.

Além disso, já existem indícios de tecnologias futuras que irão multiplicar ainda mais esse consumo como a possibilidade de assistir mais de um *stream* ao mesmo tempo. Por exemplo, a Conviva⁸, uma empresa voltada para *Internet Protocol Television* (IPTV) com pouco mais de dois anos, prevê que em pouco tempo será possível assistir uma corrida automotiva dos ângulos de várias câmeras ou assistir um *stream* ao vivo ao mesmo tempo em que se assiste um *replay*, uma atividade certamente útil no cenário esportivo.

Nesse cenário onde há uma demanda crescente por banda, o uso de uma rede de distribuição P2P é uma alternativa interessante por diminuir sensivelmente a carga dos provedores. Por esse motivo é de grande interesse buscar soluções que diminuam os efeitos dos problemas anteriormente citados, como suscetibilidade a ataques de poluição de conteúdo ou diminuição da qualidade frente a alto *churn*.

1.2 Objetivos

O objetivo proposto neste trabalho é o de estudar os nós que se destacam pelas suas altas taxas de *upload* em relação aos demais nós de uma rede P2P de distribuição de mídia contínua ao vivo organizada em malha (*mesh*) e com mecanismo de troca de dados *pull*. Pela importância de tais nós, lhes foi atribuído o conceito de super nós para este tipo de rede. A opção pelo foco na categoria *mesh-pull* é justificada por que esta classifica a maioria dos sistemas comerciais de sucesso (e.g. SopCast⁹).

Os tópicos específicos abordados por este estudo estão listados logo abaixo. Uma descrição detalhada da evolução do trabalho pode ser encontrada na sequência.

- discussão e estabelecimento do conceito de super nó no cenário P2P *mesh-pull* de distribuição de mídia contínua ao vivo;
- demonstração da influência de um super nó sobre os dados trafegados e sobre os participantes das redes P2PTV através de experimentos no SopCast;
- caracterização de super nós através da correlação das taxas de *upload* de cada nó da rede com métricas de centralidade, oriundas da disciplina de redes complexas;

⁸<http://www.conviva.com/>

⁹<http://www.sopcast.org/>

- proposição de alterações nos algoritmos descritos na literatura para o sistema CoolStreaming [Zhang et al., 2005] que estimulem ou forcem o estabelecimento de um super nó numa posição desejada da rede.

1.3 Resumo da Metodologia

As análises são realizadas através de dados capturados de um sistema real estabelecido e popular, o SopCast, e através de simulação do protocolo do CoolStreaming, implementada sobre os *frameworks* OMNet++¹⁰ e OverSim¹¹ [Varga et al., 2001; Baumgart et al., 2007]. No decorrer do trabalho serão descritas tais ferramentas assim como os detalhes de metodologia referentes a cada etapa do trabalho. Acredita-se que as comparações entre SopCast e CoolStreaming são válidas na medida em que há uma base teórica semelhante, da categoria *mesh-pull* do protocolo, que é o foco das observações e generalizações.

Inicialmente, é observado nas capturas realizadas no SopCast que há uma disparidade ante a contribuição de cada par na rede. Enquanto uma grande parcela de nós auxilia pouco ou nada no envio do *stream*, um pequeno conjunto (menos de 5%) assume o papel de servir boa parte da entrega de conteúdo (mais de 50%). À medida que se evidencia essa característica, é associado o conceito do super nó aos participantes que realizam a maior parte de todo o envio de dados da rede. Finalmente, para conduzir esse trabalho, é colocado como hipótese que o comportamento imprevisível desses nós, como eles se organizam e quão disponíveis eles estão, seria o principal responsável por dois dos problemas supracitados: diminuição de qualidade da exibição frente a um cenário de alto *churn* e suscetibilidade a ataques de poluição de conteúdo.

Um estudo acerca da influência de um super nó sobre a rede é conduzido no SopCast através de um conjunto de experimentos onde os pacotes enviados por esse nó especial eram marcados e seu fluxo era identificado na rede sobreposta (*overlay*). Os experimentos coletaram o tráfego de aproximadamente 400 nós numa *overlay* isolada que representa um canal fechado de difusão no sistema. A marcação procedeu manipulando o conteúdo de pacotes de dados, trocando o conjunto de *bytes* em uma dada região por uma assinatura específica. Os experimentos mensuram medidas como taxas de *upload*, *download* e quantidade de parcerias de cada nó da rede. Além disso, essa marcação permite acompanhar e analisar o caminho dos pacotes na rede.

Ademais, é realizado um trabalho de caracterização dos nós da rede quanto às suas taxas de *upload* e métricas de centralidade em cada nó como: grau, *betweenness*,

¹⁰<http://www.omnetpp.org/>

¹¹<http://www.oversim.org/>

closeness e duas variações das últimas. A partir dessas medições é construída uma análise de correlação entre *ranks* de *upload* e de centralidade.

Finalmente, essa pesquisa propõe pensar a rede P2P de distribuição de mídia contínua ao vivo como uma rede *overlay* de duas camadas: a camada de super nós, preferencialmente ligados entre si e ao servidor da mídia, e a camada de usuários tradicionais. É implementado em simulador um protocolo genérico de rede P2P *mesh-pull* para distribuição de vídeo baseado no CoolStreaming. A partir dessa base são realizadas três alterações no protocolo implementado para permitir a determinação de quais nós vão se ocupar das funções de super nó e promover essa separação em camadas. As alterações impactam o algoritmo de escalonamento de *chunks*, a manutenção de parcerias e o mecanismo de busca por novos parceiros. Os experimentos realizados através de simulação mensuram a continuidade do *stream*, o atraso em relação ao servidor (latência) e a resiliência a ataques de poluição de conteúdo.

1.4 Trabalhos Relacionados

As discussões sobre a especialização e destaque de nós em redes P2P são anteriores a redes de distribuição de mídia ao vivo. Em Yang & Garcia-Molina [2003] já é possível observar, em cenários onde as redes par-a-par são para distribuição de arquivo (como a Gnutella ou o Kazaa), redes híbridas compostas de clientes comuns e super nós. A proposta mostra que esse arranjo melhora a eficácia do mecanismo de busca da rede por combinar a eficiência do modelo cliente-servidor com a autonomia, o balanceamento de carga e a robustez da rede P2P, além de tirar maior proveito da heterogeneidade da capacidade de cada par.

De acordo com Lei et al. [2009], no contexto de P2P *live streaming*, o sistema Joost¹² já fez uso de super nós com o objetivo de descentralizar o mecanismo de ingresso à rede (*bootstrapping*). Contudo, é em Wu et al. [2008] que se percebe um entendimento sobre super nó mais próximo do encontrado neste trabalho. Ao analisar cargas do sistema de P2PTV UUsee Inc.¹³, eles identificam um valor maior em nós de mais alta taxa de *upload*. Além disso, eles conjecturam que tais nós não buscam se aproximar uns dos outros, criando *clusters*, e que eles emergem em pontos aleatórios da rede. Em publicação subsequente [Liu et al., 2009], há uma alteração no entendimento do que é um par superior para o UUsee Inc., incluindo a variável tempo de vida do par. Na mesma publicação, são levantados meios de priorizar a qualidade dos pares que

¹²<http://www.joost.com/>

¹³<http://www.uusee.com/>

mais contribuem e que mais tempo permanecem na rede como forma de incentivar ou prolongar a participação deles.

Diferente dessa abordagem mista, este trabalho foca no super nó enquanto par de maior taxa de saída relativa. As análises encontradas nesta pesquisa demonstram a influência oriunda desse destaque e propõem a existência de controle sobre a formação dos super nós, assim como uma organização da rede em duas camadas (super nós e nós comuns). Sugere-se que o provedor de conteúdo participe da rede não só com a máquina servidora da mídia mas, também, suprindo a rede com nós confiáveis, presentes, robustos e de maior importância (super nós), sacrificando custos de transmissão para diminuir o impacto na qualidade causada pelo alto *churn* dos participantes e mitigar os efeitos de ataques de poluição de conteúdo.

Relativo a introdução de nós presentes e confiáveis na rede P2P, é importante destacar os trabalhos que envolvem arquiteturas híbridas de P2P com CDNs pela semelhança com a separação em camadas. Esse tipo de proposta está comumente associado a trazer benefícios somente para CDNs como a diminuição de carga e consequente redução de custos [Xu et al., 2006; Huang et al., 2008a; Cahill & Sreenan, 2004]. Xu et al. [2006] aponta o potencial de redes P2P para aliviar a carga de servidores da CDN iniciando a transmissão nestes e passando a responsabilidade de difusão para os clientes até ser possível eliminar o servidor da CDN da rede P2P. Já Huang et al. [2008a] estuda os percentuais de redução da carga dado que os pares da rede P2P contribuem com um terço ou toda sua banda de saída.

Sobre a caracterização de carga em redes P2P de *live streaming*, existem diversos trabalhos na área como: Ali et al. [2006], Dhungel et al. [2007], Hei et al. [2007], Huang et al. [2008b], Sentinelli et al. [2007], Wu et al. [2007], Yang et al. [2008], Tran et al. [2004] e Silverston & Fourmaux [2007], e que revelam traços importantes sobre diversos aspectos dos sistemas de P2PTV. Sobre o SopCast, por exemplo, encontra-se o limiar intuitivo de tamanho entre pacotes de controle e de dados. Entretanto, nesses trabalhos, os dados capturados ficaram basicamente restritos à experiência dos coletores (participantes) e pouco abordam sobre visões mais abstratas da rede como um todo. Em Borges et al. [2009], os autores fazem uma extensa caracterização do comportamento dos usuários em canais com diferentes públicos e temáticas (esportes, notícias, entretenimento).

No que tange ataques de poluição de conteúdo destacam-se os trabalhos de Dhungel et al. [2007], Yang et al. [2008], Liao et al. [2006], Borges et al. [2008] e Oliveira et al. [2009]. Em Dhungel et al. [2007], tais ataques ao aplicativo PPLive¹⁴

¹⁴<http://www.pplive.com/>

são observados na ótica particular de dois pares da rede: um nó geograficamente distante do poluidor e outro ligado diretamente. Seu ataque é realizado de maneira que o poluidor cria pacotes falsos e de alta demanda, numa taxa maior do que a própria taxa de apresentação da mídia, o que força uma disseminação rápida. Já Yang et al. [2008] mostra modelagens e simulações de um ataque de poluição considerando características do Anysee¹⁵ [Liao et al., 2006], outro sistema de P2PTV. A publicação de Oliveira et al. [2009] é uma parte do presente trabalho e foca em um conjunto de ataques realizados sobre um canal fechado no sistema SopCast. Em Borges et al. [2008], discorre-se sobre um modelo de reputação capaz de reduzir o impacto do problema ao isolar os poluidores. Tanto Dhungel et al. quanto Yang et al. concordam em algoritmos de criptografia de dados como uma possível solução para o problema sem, no entanto, demonstrar a viabilidade dos mesmos em vista dos requisitos temporais e espaciais dos sistemas P2PTV. Já Coelho et al. [2010] realiza um estudo preliminar comparativo de tais algoritmos apontando os diversos problemas ainda presentes como: decaimento do nível de segurança em função do tempo, sobrecargas de processamento, comunicação e armazenamento, que diminuem a eficácia e eficiência destes mecanismos de defesa na prática.

Na aplicação de métricas de centralidade em sistemas de distribuição ao vivo P2P, destaca-se Wu et al. [2008] que avaliou a correlação de taxa de *upload* e distribuição de graus entre todos os nós de um sistema. Tal análise foi replicada e estendida por este trabalho para a rede SopCast. Métricas de centralidade são amplamente aplicadas em redes sociais tanto na Internet (Flicker¹⁶, Last.FM¹⁷, Facebook¹⁸ e outras como YouTube¹⁹ [Santos et al., 2009; Kumar et al., 2006]), quanto fora dela em cenários de resolução de problemas em grupo, política, desenvolvimento urbano e projetos de organizações como descrito no trabalho de Freeman [1979].

Outro recente trabalho que trata da melhoria de aspectos da transmissão P2P de vídeo ao vivo é a dissertação de Silva [2009]. O autor realiza uma caracterização das sessões dos usuários e propõe uma hibridização entre a arquitetura cliente-servidor e par-a-par onde um cliente inicialmente descarrega a mídia diretamente do servidor enquanto estabiliza suas conexões com a rede P2P para então se desligar do servidor e depender somente dos seus parceiros. A abordagem visa atacar o problema do longo tempo de espera inicial para preenchimento do *buffer* do usuário e conseqüentemente da exibição do vídeo. Os objetivos e propostas apresentados em ambas as dissertações

¹⁵<http://www.anysee.com/>

¹⁶<http://www.flickr.com>

¹⁷<http://last.fm>

¹⁸<http://www.facebook.com>

¹⁹<http://www.youtube.com>

são complementares na medida em que atacam problemas e trazem soluções diferentes, apesar da mesma intenção de melhorar a experiência dos usuários dos sistemas de transmissão de vídeo ao vivo.

1.5 Organização do Trabalho

Este trabalho está organizado da seguinte forma. No capítulo 2 são apresentados conhecimentos úteis para a compreensão da proposta e dos resultados: as redes P2P e de distribuição de mídia ao vivo como descritos na literatura e conhecimentos da disciplina de Redes Complexas. O capítulo 3 introduz o ferramental utilizado para captura de dados, descreve a metodologia dessa captura e, após analisar a distribuição de taxas de *upload* dos nós nas capturas, apresenta o conceito de super nó para redes P2P de distribuição de mídia ao vivo baseados em malha. O capítulo 4 identifica quantitativamente a influência de um único super nó sobre a rede ao analisar os resultados da marcação de pacotes transmitidos por ele. O capítulo 5 apresenta a caracterização de super nós através de cálculos de centralidade da rede considerando-a como uma rede complexa. São discutidas a metodologia utilizada na análise dos dados e a correlação das métricas de centralidade com taxas de *upload*. O capítulo 6 define a proposta da separação dos nós da rede em duas camadas e discorre sobre a eficácia das alterações propostas no protocolo original, através do valor agregado do uso das alterações nos quesitos impacto do *churn*, continuidade e latência do *stream* e resiliência a ataques. Finalmente, o capítulo 7 apresenta as conclusões e os trabalhos futuros.

Capítulo 2

Fundamentos Teóricos

Neste capítulo serão abordados aspectos das redes P2P, dos sistemas de distribuição de mídia ao vivo que utilizam P2P e das métricas em redes complexas.

2.1 Redes Par-a-Par

As redes par-a-par (P2P) têm inúmeras definições na literatura devido a sua terminologia abrangente. Definições estritas de uma rede P2P pura referem-se a ela como um sistema distribuído auto-organizado, com simetria de papéis entre os participantes e ausência de controle centralizado [Androutsellis-Theotokis & Spinellis, 2004; Lua et al., 2005]. Contudo, tal definição deixa de englobar diversos sistemas que são popularmente considerados como P2P. Uma mais abrangente seria:

"Redes par-a-par são sistemas distribuídos consistindo de nós interconectados capazes de se auto-organizar em topologias de rede com o propósito de compartilhar recursos como conteúdo, ciclos de CPU, armazenamento e banda, capazes de se adaptar a falhas e acomodar populações transientes de nós enquanto mantêm conectividade e desempenho aceitável sem requerer a mediação ou suporte de um servidor ou autoridade global centralizado."(traduzido de Androutsellis-Theotokis & Spinellis [2004])

Apesar dessas definições, existe uma gama de sistemas considerados P2P que englobam mecanismos centralizados como: a busca do Napster [Shirky, 2001], o *tracker* do BitTorrent [Cohen, 2003] ou o *bootstrap* de alguns sistemas P2P de distribuição de mídia [Zhang et al., 2005; SopCast, 2010].

As motivações para o uso de arquiteturas P2P derivam de sua habilidade de funcionar, escalar e se auto-organizar sobre um cenário que prevê entrada e saída intensa de nós (alto *churn*), falhas e ausência de servidor centralizado. A administração, manutenção, responsabilidade pela operação e até mesmo noções de propriedade em sistemas P2P são distribuídas pelos usuários ao invés de serem associadas a uma entidade ou indivíduo, o que os tornam geralmente resistentes à censura (bloqueio de conteúdo). Tais arquiteturas têm o potencial de acelerar os processos de comunicação e diminuir os custos de colaboração dado que melhoram o acesso aos recursos [Androutsellis-Theotokis & Spinellis, 2004]. Comparadas com o modelo cliente-servidor, elas podem eliminar pontos de contenção e reduzir fortemente os custos com servidor. Isso ocorre por que o compartilhamento de recursos computacionais é realizado por troca direta entre os participantes.

Algumas dessas características podem ser encaradas como desvantagens desse tipo de rede. Por exemplo, a descentralização contribui para diversos fatores negativos como: o surgimento de conteúdo poluído, a incapacidade de encontrar algum conteúdo existente ou a repentina ausência de algum conteúdo devido a saída de nós da rede. Situações como essas dificilmente seriam encontradas propositalmente num modelo centralizado, uma vez que há um interesse de quem centraliza em garantir recurso íntegro e disponível.

Existem diversas formas de categorizar as arquiteturas P2P, orientada a seu grau de centralização, à sua estrutura ou quanto a seu mecanismo de busca. Abaixo serão comentados as definições quanto à estrutura e quanto à centralização.

2.1.1 Definição quanto à Estrutura

A classificação pela estrutura trata da organização topológica do sistema, da sua criação e manutenção. São duas formas distintas, uma que envolve regras específicas para estruturar as redes e outra onde os novos participantes e conteúdos são organizados de forma não determinista.

2.1.1.1 Rede P2P Estruturada

A rede P2P estruturada é aquela cuja organização topológica da rede sobreposta é controlada de forma que o conteúdo (ou índices de conteúdo) não esteja espalhado aleatoriamente, e sim em nós específicos da rede a fim de tornar as buscas mais eficientes [Androutsellis-Theotokis & Spinellis, 2004; Lua et al., 2005]. Nesses sistemas há um mapeamento entre conteúdo e localização na forma de tabelas de roteamento distribuídas, por exemplo, tabelas *hash* distribuídas (DHTs). Cada par mantém uma

tabela parcial de roteamento consistindo dos identificadores dos vizinhos e o endereço IP dos mesmos. Mensagens de busca são endereçadas e encaminhadas de maneira progressiva através dos participantes da rede sobreposta até chegarem ao nó específico responsável pelo conteúdo desejado. A principal desvantagem dessa categoria de sistemas é a difícil manutenção da topologia requerida para o eficiente roteamento das mensagens dado um cenário de alto *churn*, especialmente porque há a necessidade de armazenar cópias ou ponteiros de cada conteúdo no par responsável pelo índice dele. Outra desvantagem menor está na ineficiência no suporte a buscas complexas (e.g. sentenças) [Lv et al., 2002; Androutsellis-Theotokis & Spinellis, 2004].

Em suma, a rede sobreposta atribui índices ao conteúdo e organiza-os nos pares a fim de formar um grafo que mapeia um conjunto de índices para cada par. Essa estrutura de grafo permite a descoberta eficiente de conteúdo através dos seus índices. As diversas maneiras de se montar essa política de casamento índice-par são o que caracterizam os maiores exemplos desse tipo de rede: Content Addressable Network (CAN) [Ratnasamy et al., 2001], Tapestry [Zhao et al., 2004], Chord [Stoica et al., 2001], Pastry [Rowstron & Druschel, 2001] e Kademia [Maymounkov & Mazières, 2002].

2.1.1.2 Rede P2P Não-Estruturada

A rede P2P não estruturada é caracterizada pela localização do conteúdo estar completamente desassociada da topologia da rede sobreposta, sendo necessário, em geral, realizar uma busca pouco eficiente para recuperá-lo [Androutsellis-Theotokis & Spinellis, 2004]. A rede pode ser composta de pares tanto homogêneos quanto heterogêneos. Mecanismos de busca abrangem métodos que vão da inundação de mensagens na rede ao uso de *random walks* [Pearson, 1905] e índices durante esse roteamento. Apesar de técnicas baseadas em inundação serem efetivas na localização de itens bastante replicados e de serem resilientes ao *churn*, elas dificilmente têm sucesso em busca de conteúdo raro e, mais importante, não escalam bem dado que a carga em cada participante da rede cresce linearmente com o número total de buscas e com o tamanho do sistema [Lua et al., 2005]. Enfim, as abordagens utilizadas nesses mecanismos têm claras implicações na disponibilidade, escalabilidade e persistência.

Segundo Androutsellis-Theotokis & Spinellis [2004] sistemas não estruturados são geralmente mais apropriados para cenários com alto *churn*. Alguns exemplos desses sistemas são: Freenet [Clarke et al., 2001], Napster [Shirky, 2001], Gnutella [Ripeanu, 2001], BitTorrent [Cohen, 2003], eMule [Kulbak & Bickson, 2005] e GIA [Chawathe et al., 2003].

2.1.2 Definição quanto à Centralização

Por definição, redes P2P são totalmente descentralizadas. Todavia, como visto anteriormente alguns outros sistemas com maior ou menor grau de centralização são também considerados P2P. São três as categorias, puramente descentralizada, híbrida descentralizada e parcialmente centralizada.

2.1.2.1 Rede P2P Puramente Descentralizada

A arquitetura puramente descentralizada tem uma das características que mais se aproxima da definição de Androutsellis-Theotokis & Spinellis [2004]. São redes onde todos os nós exibem comportamento e funções iguais e, além disso, que não dependem de qualquer unidade centralizadora. Essas duas características implicam um sistema altamente escalável e tolerante a falhas já que não existem pontos de contenção ou de falha localizado e a saída de qualquer participante pode ser compensada pelos demais. Os problemas relativos a esse tipo de arquitetura são a lentidão na descoberta de informações, a baixa garantia quanto à qualidade do serviço e a impossibilidade de previsão de comportamento do sistema, dada a ausência de visão globalizada [Pourebrahimi et al., 2005].

Alguns exemplos dessa arquitetura são as redes sobrepostas da Gnutella [Ripeanu, 2001], Freenet [Clarke et al., 2001], Chord [Stoica et al., 2001] e CAN [Ratnasamy et al., 2001].

2.1.2.2 Rede P2P Parcialmente Centralizada

A arquitetura parcialmente centralizada se caracteriza pela heterogeneidade dos participantes. Alguns participantes atuam de maneira especial, mais importante, tornando-se elementos centralizadores para facilitar ou a troca de recursos ou a localização de participantes. Geralmente esses nós são chamados super nós [Yang & Garcia-Molina, 2003] e, em geral, são dinamicamente escolhidos entre os participantes. Este tipo de rede melhora questões relativas à latência nas buscas por informação, dado que criam elementos centralizados na rede perdendo pouca escalabilidade e sendo resiliente a falhas pelo fato de atuarem em grupo e serem dinamicamente escolhidos (podendo ser substituídos a qualquer momento).

Alguns exemplos dessa arquitetura são observados em sistemas como a nova Gnutella [Yang & Garcia-Molina, 2003], o Emule [Kulbak & Bickson, 2005] ou o Joost [Joost, 2010].

2.1.2.3 Rede P2P Híbrida Descentralizada

A arquitetura híbrida descentralizada é caracterizada pela presença de um servidor centralizado responsável por armazenar metadados sobre recursos compartilhados ou participantes da rede. Todavia, os dados continuam sendo trafegados entre os pares. A função desse servidor de fato é facilitar o acesso a alguns tipos de informação que podem ser do tipo: "qual par tem qual conteúdo" ou "quem está participando da rede", reduzindo a latência nas buscas inerente às redes descentralizadas. Contudo, a centralização introduz alguns problemas como: baixa escalabilidade, vulnerabilidade a ataque e censura, ponto central de falha ou informação possivelmente desatualizada [Pourebrahimi et al., 2005].

O exemplo mais importantes dessa arquitetura é o sistema Napster [Shirky, 2001]

2.2 Distribuição de Mídia Ao Vivo em P2P

Serviços de difusão de mídia contínua estão se espalhando rapidamente pela Internet. Essa tecnologia associada a redes P2P tem potencial para reduzir significativamente os requisitos de largura de banda da fonte (equivalente à emissora na radiodifusão comum) ao fazer com que seus usuários sirvam parte do conteúdo, ou *stream*, para outros usuários interessados no mesmo dado. Essa solução torna o conteúdo muito mais disponível e aumenta a capacidade de transferência da mídia, viabilizando idéias inovadoras como um cliente capaz de receber múltiplos *streams*.

Ao comparar essa tecnologia com compartilhamento de arquivos e com vídeo sobre demanda é possível observar características únicas e bastante peculiares a cada um dos cenários.

Quando se trata de compartilhamento de arquivos, os conteúdos e pedaços (ou *chunks*) de conteúdo já existem previamente e podem estar disponíveis e/ou replicados nos diversos nós da rede. Do ponto de vista da transmissão, o dado é útil enquanto permanecer no nó, todavia, o conteúdo só se torna usável a partir do momento que todos os pedaços forem integrados. Os problemas desse tipo de rede estão fracamente ligados ao tempo. Um *chunk* que não foi encontrado num dado momento pode aparecer um tempo relativamente longo depois. Apesar disso, esse cenário apresenta o risco de não existirem mais cópias completas do arquivo.

Já em sistemas de vídeo sob demanda, ou VOD (*video-on-demand*), a garantia de qualidade depende de que a taxa de *download* mínima seja maior ou igual à taxa de reprodução da mídia e a premissa de integralidade da informação é relaxada, sendo que

os pedaços mais importantes passam a ser os mais próximos da frente de exibição do vídeo que, contudo, ainda não foram recuperados da rede [Huang et al., 2008b]. Nesse ambiente já existem requisitos temporais, ainda que fracos. A mídia deve ser exibida a medida em que cada trecho é recuperado. Todavia, como em arquivos armazenados, os usuários estão mais inclinados, e até acostumados pelos serviços existentes como o YouTube¹, a aceitar um tempo de espera significativo.

Mídia contínua ao vivo, no entanto, é um cenário demandante de requisitos de tempo real, já que os dados são consumidos à medida que vão sendo produzidos. Os *chunks* úteis do *stream* têm um tempo de vida muito curto e devem sempre ser entregues ao usuário a tempo de serem apresentados pelo tocador de mídia [Zhou et al., 2007; Sentinelli et al., 2007]. Somando a isso, é desejável que a percepção do usuário sobre a latência de exibição seja a menor possível para potencializar a adoção da tecnologia. Isso significa que a exibição para um espectador não pode estar perceptivelmente atrasada em relação aos outros (independente da plataforma de exibição). Ademais, é um sistema que tem um ponto central de falha associado à fonte da mídia e, atualmente, baixa resiliência à poluição [Oliveira et al., 2009].

Quanto à sua topologia, redes P2P de *streaming* podem ser classificadas de duas formas: baseadas em árvore e em malha [Sentinelli et al., 2007].

2.2.1 Sistemas baseados em Árvore

Nas redes baseadas em árvore [Deshpande et al., 2002; Tran et al., 2004; Castro et al., 2003; Kostić et al., 2003] os pares formam uma estrutura hierárquica que se assemelha a uma árvore ou a múltiplas árvores (floresta). O conteúdo então é transmitido pelo pai pra filho, fonte dos dados, para os seus parceiros filhos. A transmissão dá-se sempre de pai pra filho, sem necessidade de requisições dos filhos para tanto, e é conhecido como mecanismo *push*. Este é um termo utilizado para indicar que o dado trafega da origem até o destino sem que haja requisição direta do destino. Esse comportamento, que é realizado por todos os pares da rede, garante a transferência do conteúdo até as folhas.

As maiores vantagens das árvores são os fluxos de tráfego previsíveis e ausência de pedidos explícitos de *chunks*. Apesar disso, essa abordagem tem algumas limitações: a qualidade entregue individualmente aos pares é limitada pela menor banda entre as conexões de saída da fonte e o potencial de transmissão de grande parte dos pares folhas da árvore é desperdiçado com esse mecanismo de entrega de conteúdo [Picconi & Massoulié, 2008; Magharei & Rejaie, 2006]. Outra desvantagem que se destaca é a alta vulnerabilidade a ataques ou até mesmo ao *churn*, que sobrecar-

¹<http://www.youtube.com/>

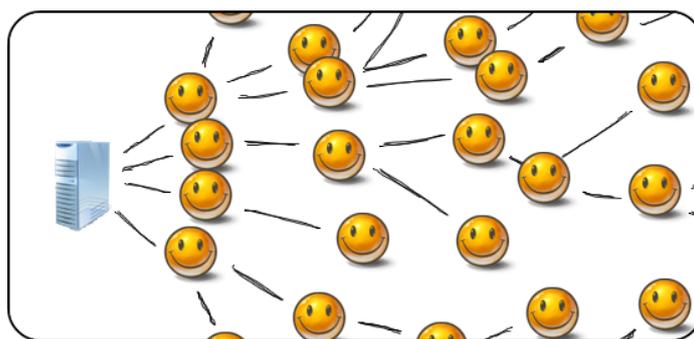


Figura 2.1: Organização de sistemas em árvore

regam a rede com solicitações de reorganização da estrutura. Como exemplo de sistema baseado em árvore existem o Overcast [Jannotti et al., 2000] e o NICE [Banerjee et al., 2002].

2.2.2 Sistemas baseados em Malha

Ao tentar explorar as restrições das redes em árvore surgiram as redes baseadas em malha (*mesh*) [Hefeeda et al., 2003; Zhang et al., 2005; Magharei & Rejaie, 2006]. Estas incorporam técnicas de enxame para distribuição de conteúdo, inspiradas em mecanismos como os do BitTorrent [Cohen, 2003], permitindo que a maioria dos pares contribua ativamente com sua banda de saída. Na construção da rede em malha os participantes formam um entrelaçado de conexões aleatórias gerando topologias não determinísticas. Essa relação dinâmica faz com que a distribuição de vídeo se torne imprevisível. Cada nó mantém múltiplos parceiros e, para que a entrega de conteúdo se torne viável, deve trocar periodicamente com seus pares conhecimento de que recurso cada qual possui.

Geralmente, requisições são enviadas para um vizinho que conhecidamente detém os dados que o par deseja, utilizando-se do mecanismo conhecido como *pull*. Este é um termo utilizado para indicar que o dado trafega da origem até o destino orientado por requisições explícitas do destino. Outra abordagem possível para a troca de dados é baseada em *push*. Dado que os vizinhos conhecem as necessidades alheias eles passam a enviar dados sem que sejam geradas requisições. É uma técnica que diminui a sobrecarga ocasionada por mensagens de controle (requisições) mas que tem alto potencial para gerar redundância de tráfego na rede, pois mais de um dos parceiros pode simultaneamente prover um determinado dado.

Algumas vantagens são: a construção e manutenção da rede são menos complexas e custosas, um alto grau de conectividade dos pares reduz a probabilidade de haver

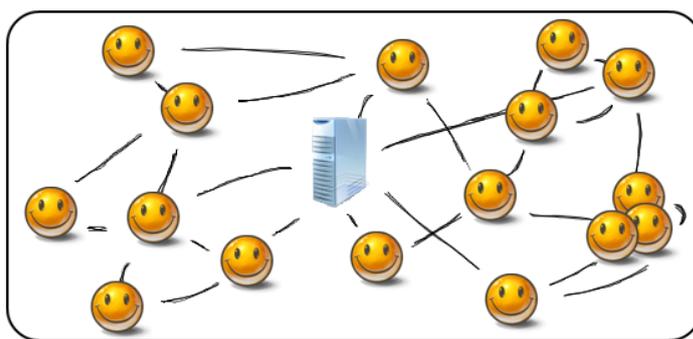


Figura 2.2: Organização de sistemas em malha

gargalos de desempenho e banda, e a resistência ao *churn* torna-se alta. Por outro lado, durante o escalonamento dos pedaços a buscar, os pares tipicamente decidem por diferentes vizinhos para recuperação de cada pacote e, como resultado, os caminhos por pacote e os tempos de entrega são altamente variáveis [Picconi & Massoulié, 2008]. Ainda assim, em [Magharei et al., 2007] é mostrada que uma estrutura baseada em malha tem um desempenho superior que uma estrutura baseada em múltiplas árvores.

2.2.2.1 Esquema de funcionamento de redes malha-pull

A maioria dos sistemas comerciais populares como: PPLive², PPStream³, SopCast⁴, TVAnts⁵ e Joost⁶, são sistemas baseados em malha e utilizam mecanismo de troca de mensagens *pull*. Devido à natureza comercial, suas especificações e códigos-fonte não estão disponíveis abertamente. Todavia, através de estudos de medição e análise é possível extrair informações sobre os seus possíveis meios de funcionamento e entender genericamente suas implementações [Sentinelli et al., 2007; Ali et al., 2006; Yang et al., 2008; Huang et al., 2008b; Hei et al., 2007; Wu et al., 2007; Dhungel et al., 2007].

Sobre a construção da rede, a alternativa mais simples de formar uma rede em malha é manter um servidor centralizado de *bootstrapping* acessível a qualquer par na rede para manter a lista de pares participantes. Cada participante que necessite de novos parceiros pode requisitar um conjunto aleatório de participantes da lista do *bootstrap*. Ou seja, esse serviço beneficia novos pares ou pares que estejam com poucos vizinhos realizando assim a manutenção da malha. Outra maneira possível de se fazer essa manutenção é através de *gossiping* [Voulgaris et al., 2003], uma técnica que des-

²<http://www.pplive.com/>

³<http://www.ppstream.com/>

⁴<http://www.sopcast.org/>

⁵<http://www.tvants.com/>

⁶<http://www.joost.com/>

creve que um participante pode escolher um vizinho qualquer para trocar informações. Ou seja, cada par troca com todos os seus parceiros as listas totais ou parciais de vizinhança a fim de que todos os nós renovem suas parcerias. O primeiro método contudo reduz a chance de particionamento da rede.

O servidor da mídia produz o conteúdo organizando-o em pedaços. Cada par mantém um *buffer* para organizar tais trechos antes de exibi-los. As requisições pelos pedaços para o preenchimento desse *buffer* se dão seguindo alguma regra que os classifica em: dados urgentes, os primeiros *chunks* ou mais próximos de serem exibidos; dados de fácil aquisição, os últimos pedaços ou que acabarem de ser produzidos; e dados comuns, o resto. Os nós trocam mensagens de controle entre si, na forma de um mapa de *bits*, que indicam sua própria disponibilidade de *chunks*. Ao receber uma mensagem desse tipo um par pode determinar quais são os conjuntos de nós e pedaços que têm potencial de troca comparando o mapa recebido com seu próprio *buffer* local. Dessa forma, observa-se que a transmissão não fica restrita a uma estrutura específica da rede, e sim à disponibilidade do dado, que guia a direção do *stream* [Zhang et al., 2005].

É importante observar que esse esquema de transmissão implica uma característica: para cada *chunk* disseminado é formado um grafo de distribuição em árvore com raiz no servidor de mídia. Assim, os dados são injetados na rede através da fonte de conteúdo. Os vizinhos da fonte podem requisitar tais dados assim que souberem que eles estão disponíveis, depois os vizinhos dos vizinhos e assim por diante. Apesar disso, não deve ser confundido com a forma de organização da rede. Nas redes em árvore essa disseminação é forçada pela estrutura de conexão dos vizinhos, enquanto que nas baseadas em malha-pull ela emerge por não necessitar que um nó obtenha mais de uma vez um dado pedaço do *stream* [Biskupski et al., 2008] e o caminho percorrido pelo *chunk* não é necessariamente o menor.

Grande parte dos mecanismos foi extraída das bem estudadas tecnologias de compartilhamento de arquivo, em especial do BitTorrent [Hei et al., 2007]. Parte do sucesso é baseada na idéia de que o dado original é fragmentado para aumentar a flexibilidade da transmissão e maximizar o uso da banda de entrada dos nós. Além disso, ao ter a liberdade para escolher descarregar diferentes pedaços dos melhores parceiros aumenta-se a disponibilidade e o número de cópias do dado e evitam-se perdas por conta do *churn*, atrasos e congestionamento na rede.

2.3 Redes Complexas

Parte deste trabalho fundamenta-se em várias métricas de teoria dos grafos e no estudo de propriedades topológicas das redes complexas. Nesta seção são apresentados os conceitos destas áreas que são determinantes para compreensão do texto.

2.3.1 Propriedades dos Grafos

A rede *overlay* de um sistema de P2PTV pode ser modelada por um grafo, onde os vértices representam os participantes da transmissão, e as arestas as parcerias entre eles. Dois vértices são chamados vizinhos, ou, no caso da rede P2P, parceiros, se há uma aresta que os conecte. O grau de um vértice é o número de arestas que incidem sobre ele, ou neste caso, sua quantidade de parcerias. O grau médio é a média aritmética do grau de todos os participantes na rede. Note que este valor é igual a duas vezes o número de arestas, dividido pelo número de vértices, uma vez que cada aresta contribui para o grau de dois vértices. O valor do grau médio é um dos indicativos do nível de conectividade do grafo.

O caminho em um grafo é uma sequência contígua de arestas e vértices que conecta um par de vértices. Seu tamanho é definido como o número de arestas percorridas. O caminho mínimo entre dois vértices é a menor sequência de arestas possível entre eles e seu tamanho é chamado distância geodésica. O diâmetro é a maior distância geodésica entre quaisquer dois vértices distintos no grafo. Tanto o diâmetro quanto o caminho mínimo médio dão indícios quanto à organização da rede.

O coeficiente de agrupamento local é uma métrica que representa o grau de interconexão entre os vizinhos de um vértice. Seu valor é definido como a razão entre o número de vizinhos de um vértice que também são vizinhos entre si ($V_i(v)$), pelo número máximo possível de vizinhos de v que também são vizinhos entre eles ($V_{i_{max}} = Combinacao(grau(v), 2)$). O coeficiente de agrupamento médio de um grafo mostra a tendência dos seus vértices a se organizarem em comunidades, indicando a densidade de triângulos formados pelos vértices e arestas do grafo. Abaixo está a forma de cálculo.

$$Coeficiente_{agrup.}(v) = \frac{V_i(v)}{V_{i_{max}}} = \frac{2|V_i(v)|}{grau(v)(grau(v) - 1)}$$

2.3.2 Modelos de Redes

O estudo de redes complexas define alguns modelos de redes baseadas nos conceitos apresentados. O primeiro deles é representado por um grafo gerado de forma puramente

aleatório, também chamado de *Erdos-Renyi* [Erdos & Renyi, 1960]. Nele, o número de vértices é fixo, e há uma probabilidade única $p(i, j)$ de haver uma aresta entre dois vértices v_i e v_j quaisquer. Este tipo de grafo apresenta um baixo coeficiente de agrupamento médio e caminho mínimo médio pequeno.

Existe uma classe especial de redes aleatórias que são representadas por um grafo onde a maioria dos vértices não são vizinhos entre si, mas dois vértices quaisquer podem alcançar o outro em um pequeno número de passos. Além disso, seus vértices tendem a se organizar em comunidades. O modelo que descreve tais redes chama-se *Watts Strogatz* [Watts & Strogatz, 1998], mais conhecido também como mundo pequeno (*small world*). Tais redes são caracterizadas por caminho mínimo médio e diâmetro pequenos e coeficiente de agrupamento médio mais alto do que a do grafo aleatório. Muitas destas redes podem ser observadas empiricamente.

2.3.3 Métricas de Centralidade

A teoria dos grafos e análise de rede definem várias medidas de centralidade de um vértice em um grafo que permitem determinar sua importância estrutural relativa. Três métricas, utilizadas nesse trabalho, são: a centralidade de grau, o *closeness* e o *betweenness* [Freeman, 1979; Koschützki et al., 2005]. Elas foram criadas para analisar qualquer tipo de grafo. Contudo, para avaliar o cenário de distribuição par-a-par de mídia contínua ao vivo, foram criadas algumas variações das mesmas. Por exemplo, todos os dados se originam de um único nó, a fonte ou servidor de mídia, e, por isso, foram calculadas duas variações das métricas de *betweenness* e *closeness*, onde somente consideraram-se os caminhos mínimos de todos os vértices para o nó fonte.

A primeira métrica, centralidade de grau, é uma medida relativa ao número de arestas que cada vértice tem. Uma alta centralidade de grau está normalmente associada a uma maior chance de qualquer dado que trafega na rede passar por determinado vértice. Além disso, a centralidade de grau pode ser calculada para grafos não direcionados e direcionados, onde, nesse último caso, são considerados o grau de entrada (arestas que chegam no vértice) e grau de saída (arestas que saem do vértice). Já para os experimentos desta pesquisa, onde a rede forma um grafo não direcionado, temos, segundo [Newman, 2003], que a métrica pode ser calculada como o grau de cada vértice sobre o número n de vértices do grafo menos um:

$$C_{Grau}(v) = \frac{grau(v)}{n - 1}$$

A métrica de *closeness* [Sabidussi, 1966], é uma medida de proximidade espacial

entre nós do grafo. Ela define quão perto um vértice está de todos os outros através das conexões estabelecidos no grafo e é descrita como a distância geodésica (caminho mínimo) média entre o vértice v e todo vértice t alcançável a partir de v ($t \in V \setminus v$).

$$C_{Closeness}(v) = \frac{1}{\sum_{t \in V \setminus v} d_G(v, t)}$$

Outra métrica, baseada no *closeness*, levou em consideração que o único vértice alcançável que importa é a fonte, logo, desconsiderando a variação de t . Dado que t é sempre a fonte, a medida original se tornou somente o inverso do caminho mínimo até o servidor de mídia, portanto, foi considerado somente como distância, para simplificar.

$$C_{Distancia}(v, fonte) = \frac{1}{d_G(v, fonte)}$$

O *shortest-path betweenness* [Brandes, 2001], que neste texto será tratado apenas como *betweenness*, é uma medida que pode ser vista como a probabilidade de um nó v estar envolvido na comunicação entre um nó s e outro t . O valor dele para um vértice v é a quantidade de caminhos mínimos entre quaisquer pares de nós s e t que passam por v . Na fórmula a seguir σ_{st} é o número de caminhos mínimos de s a t , e $\sigma_{st}(v)$ é o número de caminhos mínimos de s a t que passam pelo vértice v :

$$C_{Betweenness}(v) = \sum_{\substack{s \neq v \neq t \in V \\ s \neq t}} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

Por fim, essa fórmula sofreu uma alteração para formar uma nova métrica e t passou a ser percebido como um único vértice específico, a fonte de mídia. Assim, foi calculado um *betweenness* especial, partindo de todos os nós somente até o servidor que passem por v , como:

$$C_{B.Fonte}(v, fonte) = \sum_{s \neq v \neq fonte \in V} \frac{\sigma_{s fonte}(v)}{\sigma_{s fonte}}$$

Capítulo 3

Conceituação do Super Nó

Este capítulo apresenta inicialmente o conjunto de tópicos que tratam da captura do dados e culmina na descrição dos super nós para a rede P2PTV. A primeira seção demonstra as ferramentas utilizadas na captura de dados durante toda a dissertação, sua importância e motivação do uso. Na seção 3.2 é descrita a metodologia da captura de dados dos experimentos. Já na seção 3.3 é colocado em foco a distribuição de taxas de *upload* médio dos nós da rede a fim de evidenciar a existência de um nó especial na rede, o super nó. Por fim, a seção 3.4 discute a existência de super nós nas redes P2P de distribuição de mídia ao vivo.

3.1 Ferramental para Captura de Dados

Abaixo, serão descritos os sistemas e ferramentas utilizadas no processo de obtenção de dados, reais e simulados, deste trabalho.

3.1.1 SopCast

Atualmente, existem diversos sistemas de transmissão de mídia ao vivo par-a-par, entre eles: PPLive¹, PPStream², SopCast³, TVAnts⁴ e Joost⁵. Todos esses compartilham algumas características como distribuição gratuita e código fechado, sendo cada qual popular em uma dada região do planeta.

¹<http://www.pplive.com/>

²<http://www.ppstream.com/>

³<http://www.sopcast.org/>

⁴<http://www.tvants.com/>

⁵<http://www.joost.com/>

Para esse trabalho será utilizado como estudo de caso o SopCast, a fim de coletar dados de um ambiente real. A escolha desse sistema se justifica na sua popularidade, que, de acordo com o Google Trends (figura 3.1), no ano de 2009 foi o sistema mais buscado no Google em comparação com os demais.

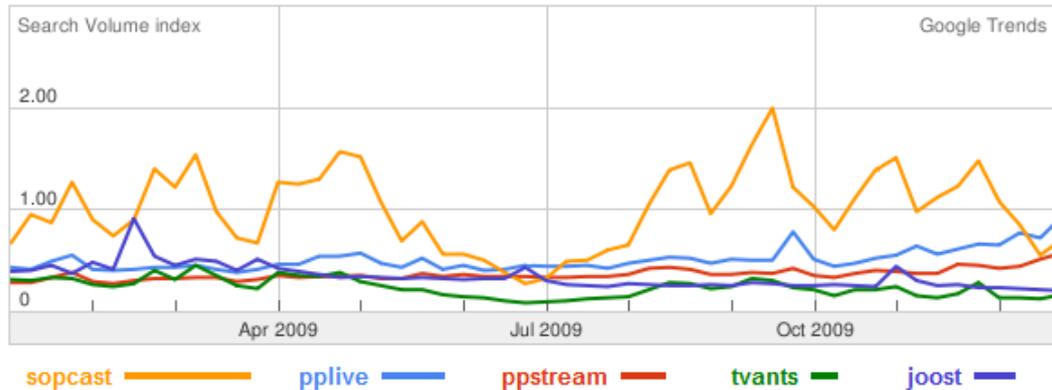


Figura 3.1: Tendência de busca para sistemas populares de P2PTV em 2009

O SopCast é um aplicativo gratuito para apresentação e distribuição de vídeo e áudio sobre a Internet. O sistema adota tecnologia P2P e o seu núcleo é um protocolo de comunicação, o `sop://`, ou tecnologia SoP (*Streaming over P2P*), produzido por um grupo chinês. Através dele é possível tornar-se um emissor de conteúdo sem a necessidade de arcar com os custos de um servidor poderoso ou consumo de muita largura de banda. De acordo com informações do próprio *site*, é possível construir uma estação televisiva doméstica com o mínimo de recursos. Com um computador pessoal e conexão à Internet residencial (entre 256 kbps e 512 kbps) uma pessoa pode servir conteúdo para outros milhares de espectadores ao usar o SopCast. Contudo, para uma distribuição estável é necessária uma conexão cinco vezes maior que a taxa de reprodução da mídia.

Para começar a servir uma mídia no SopCast é preciso inicialmente registrar um canal no *site* do SopCast definindo nome, visibilidade, linguagem, descrição e local de origem do sinal. Recebido um número de identificação do canal é possível utilizá-lo na aplicação do SopCast para transmitir mídias armazenadas (arquivos numa pasta) ou uma fonte de *stream* ao vivo gerado por um servidor ou codificador de mídia. Assim, observa-se que o sistema pode ser usado para VOD. Os protocolos de transferência permitidos são o `mms://` e `http://`, codificáveis através do Windows Media Encoder 9, que é distribuído gratuitamente. Tanto este último quanto a aplicação do SopCast para montar servidores só funcionam no sistema operacional Windows. A construção de um canal é o início da formação de uma rede sobreposta onde o servidor de mídia é

o nó inicial e os espectadores passam a buscar a mídia uns dos outros montando assim a rede de vizinho, P2P.

3.1.2 PlanetLab

O Planet Lab⁶ é um consórcio mundial de instituições de pesquisa que mantém um ambiente global para o desenvolvimento e testes de aplicações distribuídas. Cada instituição mantém um ou mais nós na Internet que operam como servidores de máquinas virtuais. Uma conta de acesso ao PlanetLab é chamado de *slice*. Se uma instituição mantiver pelo menos um nó em operação lhe é concedido o direito de criar *slices* e cada *slice* tem o poder de controlar um conjunto de máquinas virtuais em outros nós da rede.

Um usuário com acesso a um *slice* é capaz de criar uma máquina virtual, com um conjunto inicial de aplicativos minimalista e espaço restrito, em cada nó disponível. Esses recursos devem ser configurados e gerenciados individualmente, o que cria um problema na medida em que qualquer atividade sobre os mesmos deve considerar que suas características de *hardware* e rede são heterogêneas e que eles não estão sempre disponíveis.

Todavia, o uso do PlanetLab traz muitas vantagens no que se refere à aquisição de dados. A mais óbvia é a disponibilidade de pouco mais de mil nós, o que auxilia na recuperação de dados mais representativos e confiáveis. Outra vantagem é que os nós do PlanetLab são dispersos tanto em redes distintas quanto geograficamente, evitando que aspectos de localização mascarem o comportamento da rede. Por fim, o fato dos nós do PlanetLab possuírem IPs reais e não sofrerem filtragem de pacote evita a necessidade de tratar o problema do NAT (*Network Address Translation*) [Bellovin, 2002], onde participantes da transmissão que estão atrás do mesmo *firewall* são identificados como um único participante.

3.1.3 Simulador: OMNeT++ e OverSim

O OMNeT++⁷ [Varga et al., 2001] é um ambiente de simulação de eventos discretos. Ele foi criado para simular redes de comunicação mas, devido a sua arquitetura genérica e flexível, é possível utilizá-lo em simulações de sistemas computacionais complexos, redes de filas ou em projetos de hardware. O que o OMNeT++ provê na verdade é uma arquitetura de componentes para modelos. Os componentes, chamados módulos, são programados em C++ e posteriormente montados em componentes maiores e modelos

⁶<http://www.planet-lab.org/>

⁷<http://www.omnetpp.org/>

através de uma linguagem própria de alto nível. Algumas vantagens extras são que: esse processo permite a construção de modelos reusáveis e, devido a sua arquitetura modular, mesmo o código da simulação tem grande potencial de ser integrado à uma futura aplicação. Apesar dele não ser um simulador de redes, atualmente vem ganhando bastante popularidade na comunidade científica como plataforma de simulação de rede.

Já o OverSim⁸ [Baumgart et al., 2007] é um *framework*, baseado no OMNeT++, para simulação de redes *overlay* e P2P. O simulador conta com diversos modelos já prontos de sistemas P2P e protocolos de *overlay* estruturados (Chord, Kademia, Pastry) e não estruturados (GIA) [Stoica et al., 2001; Maymounkov & Mazières, 2002; Rowstron & Druschel, 2001; Chawathe et al., 2003]. Além disso, oferece abstrações de *churn*, canais de transmissão, largura de banda, *bootstrapping* e, inclusive, roteadores e *backbones* para construção da rede física (*underlay*), se necessário.

Utilizando o OverSim, o modelo baseado no Coolstreaming [Zhang et al., 2005] e influenciado por experimentos no SopCast foi implementado. Essa escolha foi feita por que o Coolstreaming implementa um sistema baseado em malha com mecanismo de troca *pull*, como o SopCast, tendo muitas características similares. São diversos algoritmos, explorados em detalhes em 6.2, para o projeto de um sistema de distribuição de conteúdo P2P como: gerenciamento de participantes, representação e troca do mapa de *bits* do *buffer* e escalonamento de troca de *chunks*.

3.2 Metodologia de Captura de Dados

Em redes de P2PTV comumente observam-se populações de milhares de usuários com comportamentos extremamente distintos e dependentes de diversas variáveis. Isso, aliado ao fator de que se trata de uma área de pesquisa ainda bastante nova e de pouca compreensão especialmente quanto ao fluxo dos pacotes, conduziu à opção por capturas em ambiente real. Ainda assim, este trabalho se utiliza de abordagem real e simulada a fim de melhorar a validade dos resultados e explorar características atualmente não presentes nos sistemas.

Para as capturas no SopCast foram conduzidos uma série de experimentos com o sistema SopCast contando com o auxílio do Planet Lab, do Wireshark e um aplicativo criado para efetuar a manipulação e marcação de pacotes de redes. Aproximadamente 400 pares heterogêneos e dispersos geograficamente (oriundos do PlanetLab) foram inseridos na rede de distribuição de mídia contínua para participar e monitorar o tráfego, com auxílio da ferramenta de análise de rede Wireshark.

⁸<http://www.oversim.org/>

Para uma rodada de experimento, todos os nós, munidos de uma versão Linux do cliente SopCast, ingressam ao mesmo tempo na rede a fim de assistir um canal específico e saem ao final da rodada, que tem duração de 5 minutos. A hipótese de que os pacotes tem tempo de vida curto (da ordem de alguns segundos) explica a duração da rodada. Cada par realiza uma coleta dos *logs* da camada de rede usando Wireshark já filtrando o tráfego relativo pela porta da aplicação analisada.

O canal citado acima é um canal privado no SopCast, transmitido especialmente para a execução dos experimentos. Por canal privado entende-se que a transmissão só seria acessada por usuários que conhecessem o seu número de identificação, ou seja, ele não é divulgado na lista normal de distribuição de canais do software e, portanto, supõe-se um ambiente isolado de agentes externos (sem usuários reais assistindo a mídia, somente *crawlers* ou *bots*). Essa decisão teve como intenção não causar nenhum prejuízo real ao sistema dada a independência entre canais, além de não ter sido o foco do trabalho identificar o comportamento de usuários reais sob ataque. O canal transmite uma mídia codificada através do Windows Media Encoder ⁹ a uma taxa de aproximadamente 120kbps.

A opção por experimentos fechados, em especial, onde os nós ingressam e deixam a rede simultaneamente conduzem a um cenário de experimentação ausente da rotatividade de participantes natural às redes par-a-par (*churn*). Apesar do isolamento do fator *churn* externo, a saída e entrada de pares do canal, o SopCast apresenta uma alta rotatividade interna, observado na forma de parcerias bastante promíscuas. Dado que um par não tem conhecimento sobre a permanência na rede de um vizinho que se desconecta dele, a visão de cada par é semelhante para cenários fechados e abertos. Apesar disso, deve ser considerado o fato de que nós antigos na rede, mesmo com novas parcerias, mantêm *chunks* potencialmente interessantes para os novos parceiros, ao contrário de um nó novo.

Para garantir a correteza, coerência e aumentar a significância dos resultados, o experimento foi repetido 22 vezes. Entretanto, apesar da aparente configuração igual para cada repetição, durante as capturas observou-se que a influência de um participante da rede era variável e produziam resultados que iam de quase nenhuma transmissão de pacotes marcados à vasta contribuição (e.g. 30%-70%) do nó à rede. Os motivos serão melhor explicados na próxima seção. Dentre todas as capturas, os 5 experimentos onde o nó marcador teve as maiores taxas de *upload* foram selecionados para efetuar as análises do capítulo 4.

Através dessa metodologia foi possível entender diversas características que

⁹<http://www.microsoft.com/windows/windowsmedia/pt/9series/encoder/>

variam ao longo do tempo dessa *overlay*. Os dados obtidos foram utilizados das seguintes formas:

- nesse capítulo será dada ênfase à distribuição das taxas de *upload* médio dos pares, característica que independe da manipulação dos pacotes;
- no capítulo 4 serão apresentados os resultados relativos à pervasividade dos pacotes marcados. Serão traçadas relações entre o tempo em que o nó permaneceu manipulando pacotes, o tempo de degradação e recuperação da rede, alcance e tempo de vida médio dos pacotes marcados, através de medições das taxas de *download*, *upload* e quantidade de parcerias;
- no capítulo 5 serão correlacionadas as taxas de *upload* dos nós a diversas métricas de centralidade da rede reconstruída a partir dos *logs* de cada nó.

3.3 Distribuição de Taxas de Upload Médio

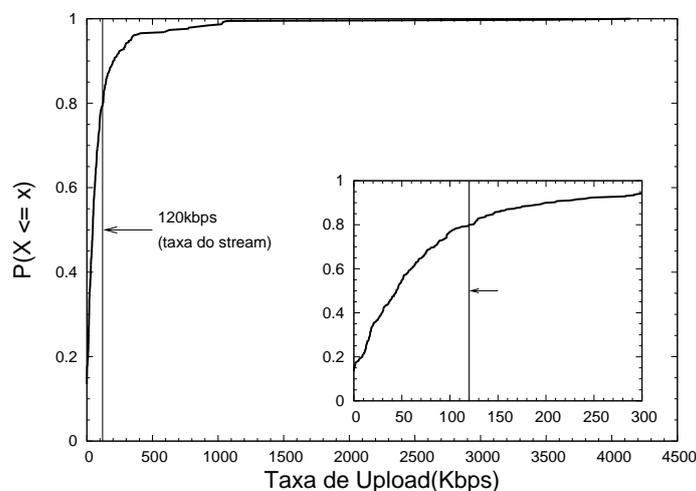


Figura 3.2: Distribuição Cumulativa da Taxa Média de Upload dos Pares

A figura 3.2 introduz uma distribuição comum, obtida no cenário real, de taxas de *upload* média dos pares. A figura interna é um *zoom* do início da curva (até 300 kbps). O resultado mostra que nem todos os nós contribuem da mesma forma para a transmissão. Na maioria dos experimentos, entre 70 e 80% dos nós tiveram taxas médias de *upload* inferior à taxa original de transmissão do vídeo. Mais importante, existe uma pequena margem de nós (menos de 5%) que detêm uma carga de transmissão significativamente superior aos demais, chegando a taxas de envio mais de 30 vezes

maiores do que a taxa do *stream*. As taxas de *download* para todos os nós tendem a ser próximos da taxa de transmissão.

Aos nós responsabilizados por concentrar a maior parte do *upload*, influenciando assim uma grande parcela dos dados trafegados na rede, foi atribuído o título de super nós. Com essa definição, a figura 3.3 trás um novo entendimento de como se organiza uma rede em malha de distribuição de mídia contínua exibindo uma distinção entre dois tipos de nós: os comuns e os super nós.

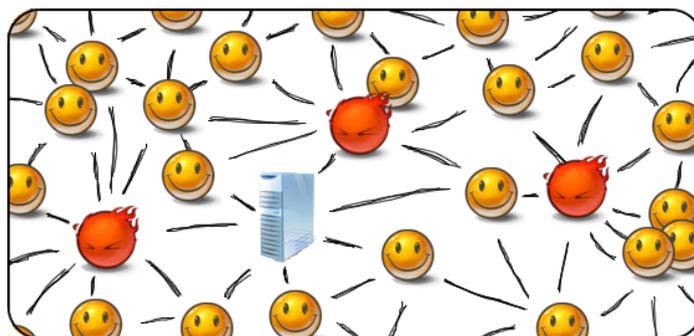


Figura 3.3: Organização de sistemas baseados em malha considerando super nós

3.4 Super Nó

Um super nó numa rede par-a-par é um nó que, por alguma característica, quebra a homogeneidade dos papéis dos pares na rede. Ele deve ter algum destaque ou uma função especial como, por exemplo, ser responsável por indexar os arquivos do seu grupo de vizinhos. Nesse trabalho, são considerados super nós aqueles que se destacam pela taxa elevada de *upload* entre todos os participantes da *overlay*. Concordante com essa definição, Wu et al. [2008] os apresentam como "pares de alto *throughput* de envio" enquanto demonstra-se a existência deles no sistema UUSEE Inc.¹⁰. Pelas semelhanças no sistema de troca de dados, acredita-se que essa característica pode ser encontrada em outros sistemas P2P de distribuição de mídia contínua baseados em malha.

O conceito de nós especiais na rede não é algo inovador. Na pesquisa de Yang & Garcia-Molina [2003], são descritas redes P2P híbridas de distribuição de arquivo com a presença de clientes comuns e super nós. As vantagens apresentadas no trabalho são relativas ao mecanismo de busca das redes P2P que ganham eficiência

¹⁰<http://www.uusee.com/>

através da capacidade centralizadora do super nó. Outro sistema notoriamente conhecido pelo uso de super nós é o Skype [Guha et al., 2006], que os utiliza para rotear diversos tipos de solicitações (por exemplo, chamadas e transferências de arquivo).

Há diferenças essenciais na abordagem que se quer dar ao super nó aqui e em cenários de distribuição de arquivo como o de Yang & Garcia-Molina. Como destacado na seção 2.2, distribuição de arquivos e de mídia contínua são sistemas com características distintas. Em especial, neste trabalho nem estão em questão mecanismos de busca, dado que a localização de vizinhos úteis é comumente feita através do servidor de *bootstrap* ou das parcerias existentes e que a localização de *chunks* é feita através dos vizinhos imediatos (se esses não têm o *chunk* provavelmente estão também o procurando nos seus vizinhos e assim por diante). O papel assumido pelo super nó no cenário de mídia contínua é contrário a algumas boas práticas das redes P2P como a de distribuição da carga (vantagem associada ao super nó de Yang & Garcia-Molina), já que ele tende a concentrar o envio de *chunks*.

Experimentalmente, observa-se que a rede de distribuição de mídia ao vivo é injusta já que alguns nós servem mais que os outros. No capítulo 4, ao analisar a influência do super nó, é possível perceber que eles realizam a maior parte do *upload* e que têm um controle muito grande da rede, da sua segurança e estabilidade. Sendo assim, pode ser interessante para um gerador de conteúdo ter autoridade sobre esses nós, protegendo-os, mantendo-os sempre ativos e impedindo que seu conteúdo seja deturpado, minimizando os impactos na rede.

Obviamente, o controle de mais nós implica em mais custos, especialmente com banda, dada a premissa de que esses nós tenham *upload* mais alto que todos os demais nós da rede. Todavia, ainda assim é mais econômico do que manter-se no modelo cliente-servidor. A figura 3.4 traz uma perspectiva sobre o fator custo, dado que seja considerado somente o uso de banda com envio. No eixo X está apresentado o percentual acumulativo dos nós ordenados de maneira decrescente pela taxa de *upload*, ou seja, dos participantes que mais contribuíram ao que menos contribuíram, da esquerda pra direita. Já no eixo Y é colocado o percentual acumulativo da taxa de envio média ordenado também de forma decrescente, da maior taxa de *upload* média à maior taxa de *upload* média, de baixo para cima. Ao analisar o gráfico podemos concluir que, por exemplo, controlando 5% dos nós com maior *upload* controla-se aproximadamente 50% de todo *upload* realizado pela rede. Assim, o risco de problemas com *churn* e ataques é reduzido na medida em que diversos super nós são gerenciados por quem tem mais interesse com a qualidade do *stream*.

Finalmente, o super nó desse tipo de rede P2P é uma abstração para indicar os nós que mais realizam *upload* na rede. Contudo, tanto a análise anterior quanto a

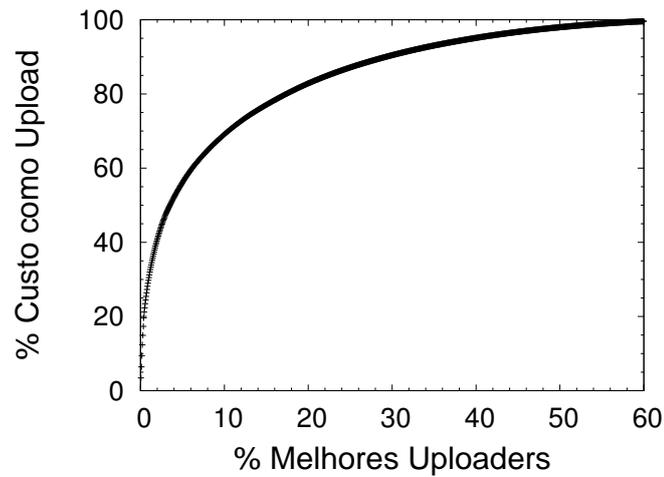


Figura 3.4: Percentual de banda de *upload* fornecida em função do percentual de pares de maior *upload* controlados

da figura 3.2 não evidenciam um limiar claro para determinar essa característica, ou seja, o ponto que separa nós comuns de super nós não é concreto. Considerando os parâmetros de *upload* e custo, apresentados em tais figuras, sugere-se justamente que um quantidade próxima de 5% do total de nós da rede sejam considerados super nós. Outro critério possível, consonante com o anterior, é considerar super nós aqueles que realizam *upload* a uma taxa superior a duas ou três vezes a taxa do *stream*.

Capítulo 4

A Influência do Super Nó

No capítulo 3 foi observado que a rede se organiza de tal forma que alguns nós se destacam pela quantidade de *upload* que realizam. Nesse capítulo será abordada a influência que um único super nó pode alcançar num sistema. A influência dos super nós pode impactar os sistemas P2P de distribuição de mídia ao vivo *mesh-pull* de diversas formas como em casos de falhas ou de ataque de poluição de conteúdo. Para verificar essa característica são realizados experimentos que efetivam uma marcação dos pacotes de rede a fim de conhecer o fluxo dos dados que são originados de um super nó. Em especial, serão observados aqui os experimentos onde o nó responsável por manipular pacotes foi sempre um super nó, esteve entre os quatro nós de mais alto *upload*, o que corresponde a 1% dos pares da rede utilizada que é de aproximadamente 400 nós. Os resultados principais dos experimentos indicam que 30% a 70% de todo conteúdo descarregado na rede estava marcado e 50% a 70% de todos os participantes receberam este tipo de pacote, e mostram que um único super nó pode afetar significativamente o sistema P2P de distribuição de mídia contínua ao vivo.

Parte dos dados deste capítulo foram previamente publicados em Oliveira et al. [2009].

4.1 Ataques de Poluição de Conteúdo

Uma das consequências do grau de influência dos super nós sobre a rede é a suscetibilidade da mesma à ataques de poluição de conteúdo e, por isso, esse tema será abordado na seção. Os danos causados pela poluição em sistemas P2P já são bem conhecidos [Yang et al., 2008], especialmente para sistemas de compartilhamento de arquivo. Em geral, os sistemas P2P são sujeitos a poluição por uma característica essencial: os clientes participam ativamente do sistema. Como eles são, individualmente, pequenos

servidores de conteúdo, podem modificar as requisições respondidas sem que o sistema, na sua totalidade, reconheça a ação maliciosa. Isso ocorre não só com distribuição de mídia contínua ao vivo, mas em qualquer tipo de dado de aplicação P2P, basta que seja conhecida a forma com a qual os dados são manipulados na comunicação, isto é, o seu protocolo.

Apesar de ser um tema interessante e pertinente, existem poucas publicações sobre ataques reais de poluição de conteúdo nas redes de mídia contínua ao vivo, o que torna a bibliografia escassa até onde se tem conhecimento. Em parte, isso ocorre porque os protocolos dos sistemas atuais mais populares são proprietários e, supostamente, mais difíceis de serem penetrados. Por esse mesmo motivo, o funcionamento interno dos mesmos é desconhecido e figura como tema de grande interesse acadêmico. Contudo, com a popularização da tecnologia e um número crescente de usuários, espera-se concomitantemente um crescimento do interesse e do empenho de usuários maliciosos a fim de descobrir meios de adulterar os sistemas para atender suas necessidades.

Esse tipo de ataque degrada o nível de disponibilidade de dados corretos alterando maliciosamente o conteúdo original do sistema alvo [Yang et al., 2008; Dhungel et al., 2007]. Sua ocorrência é representada pelo aparecimento de conteúdo corrompido, ausência de conteúdo, propagandas ou pornografia no meio de uma transmissão normal, o que provavelmente induz os usuários comuns a mudarem de canal ou desligar o sistema, se distanciando da informação que não lhes é de interesse. Em caso de degradação prolongada ou conteúdo psicologicamente nocivo, como pornografia ou imagens chocantes, o consumidor pode inclusive alimentar uma aversão pelo sistema, deixando de utilizá-lo por completo. Por isso, conhecendo essa ameaça, sua aplicabilidade e seu poder destrutivo, é importante gerar soluções e estudos que impeçam a sua propagação.

4.2 Metodologia de Marcação de Pacotes

Anteriormente, na seção 3.2, está descrita a metodologia de captura de dados. Nessa seção serão explicitados os detalhes da manipulação e marcação dos pacotes de rede realizados por um nó. Para alcançar os objetivos do estudo do capítulo foi definido, dentre os pares inseridos na rede, um nó diferenciado que marca todos os pacotes que envia durante um intervalo de tempo. Dado que cada rodada de experimento dura 300 segundos, o intervalo inicia no instante $t = 110$ e cessa em $t = 240$ segundos. Esses valores visam observar dentro do período total de experimento a perturbação causada pela marcação de pacotes a partir de uma rede estável até a re-estabilização após o final

da marcação. Para todas as rodadas o nó manipulador de pacotes foi sempre único e o mesmo.

A forma de marcação protagonizada nesse trabalho envolve a alteração de um conjunto de *bytes* no pacote correspondente ao que seria a parte da mídia codificada. Análises anteriores (refeitas para as capturas deste trabalho) mostram que pacotes de dados e de controle tem tamanhos significativamente diferentes. Dados tendem a ocupar todo o quadro de transmissão enquanto que mensagens de reconhecimento e controle aparentemente não chegam a mais de 100 *bytes* [Ali et al., 2006]. Para garantir a identificação dos pacotes de dados foi acrescido uma margem de segurança a esse limiar. Portanto, todos os pacotes acima de 160 *bytes* foram considerados pacotes de dados. Em uma região supostamente contendo o *stream*, são modificados 11 *bytes* de forma a assinar o pacote com uma sequência específica. Ainda, ao lado da sequência são colocados 2 *bytes* como um identificador numérico autoincrementado de criação, garantindo que cada pacote manipulado transmitido terá uma assinatura única composta por uma parte constante e um número. Para interceptar um pacote de rede que está saindo desse nó com capacidade de manipulação utiliza-se uma característica do *framework* de filtragem de pacote presente no *kernel* do Linux 2.6.x, o Netfilter¹. Ao criar uma regra no *software* iptables capaz de filtrar todos os pacotes saindo da máquina local relativos ao SopCast e enviá-los a uma fila no espaço de usuário, a aplicação pode marcar os pacotes e devolvê-los à rede.

Esta operação resulta para os nós que receberam o pacote manipulado na errônea decodificação do fluxo de vídeo, criando uma imagem distorcida e irreconhecível em comparação com a original. É importante lembrar que os experimentos foram feitos repetidas vezes com um número elevado de nós e que apenas cinco experimentos, onde o marcador tem as maiores taxas de *upload*, foram selecionados para as análises. Apesar da análise dos resultados serem relativas a todos os cinco experimentos, os gráficos apresentados a seguir ilustram apenas um deles, selecionado para descrever visualmente o grupo. A metodologia permitiu a identificação do caminho dos pacotes marcados e a obtenção dos resultados a seguir.

4.3 Avaliação da Influência de Super Nós

4.3.1 Análise Preliminar

O primeiro resultado obtido demonstra que apesar de existirem técnicas conhecidas de verificação de integridade do dado na literatura sua utilização pelo sistema estu-

¹<http://www.netfilter.org/>

dado não é clara. Em um experimento inicial menor utilizando somente quatro pares, sendo um nó marcador e um servidor, foram identificados pacotes manipulados sendo trocados entre os dois participantes ingênuos, retransmissores passivos de dados marcados. Dessa forma, sem a identificação da manipulação, o pacote não é marcado como ruim/corrompido. Conseqüentemente, não é descartado, não são realizados pedidos de retransmissão e, por esse motivo, não foram observadas alterações nas taxas de *download* e *upload* de cada nó decorrentes do esquema de marcação.

4.3.2 Análise do Cenário Realista

Os experimentos a seguir foram realizados conforme a metodologia de captura descrita em 3.2 e a metodologia de marcação presente em 4.2. A rede é composta por aproximadamente 400 nós no PlanetLab, que se conectam simultaneamente ao canal do experimento criado no SopCast e, caso não falhem por qualquer motivo, abandonam também a rede sobreposta todos juntos após cinco minutos de coletas de *logs* da camada de rede com uso do Wireshark.

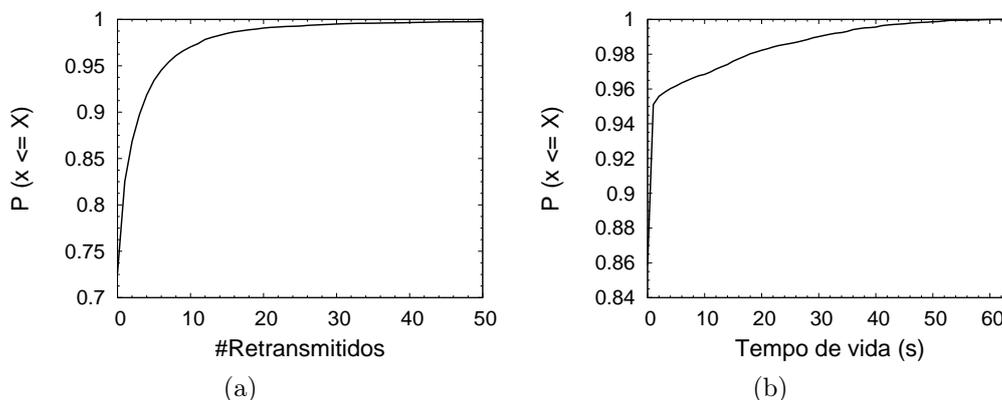


Figura 4.1: Distribuições Relativas aos Pacotes Marcados

A figura 4.1 apresenta relações acerca dos pacotes assinados. O gráfico 4.1a mostra a distribuição de retransmissões de pacotes manipulados, isto é, quantas vezes cada um desses pacotes trafegou pela rede. É importante lembrar que cada qual tem identificação única (só é emitido uma vez pelo marcador), ou seja, um *chunk* solicitado por dois vizinhos distintos gera dois pacotes marcados distintos. Dessa forma o gráfico revela o alcance e a pervasividade de um nó com alta oferta de *upload* e, por consequência, a possibilidade de impacto negativo do super nó. Pode-se perceber que a maioria dos pacotes marcados são passados adiante poucas vezes. Todavia, mesmo poucas retransmissões são significativas considerando que a rede tem um diâmetro pe-

queno. Em aproximadamente 97% das vezes eles foram repassados pelos demais nós menos do que 15 vezes, enquanto um pequeno percentual é altamente pulverizado pela rede sendo retransmitido mais de 80 vezes.

A figura 4.1b é uma distribuição do tempo de vida desses pacotes, considerando o tempo de vida a diferença temporal entre a primeira e a última aparições em todos os *logs* do pacote assinado. Esse tempo de vida pode ser encarado como o tempo de utilidade do pacote para a rede. Apesar da granularidade da medição ser de segundo a segundo foi possível extrair as seguintes informações a partir dos dados: em 95% dos casos, o pacote marcado foi útil por um tempo extremamente curto, menor ou igual a dois segundos. Esse percentual é afetado especialmente por uma alta incidência de pacotes que não foram encaminhados nenhuma vez; o tempo máximo entre um pacote ter sido gerado e sua última retransmissão foi um pouco maior que um minuto; e, finalmente, não há uma coincidência entre pacotes mais retransmitidos e de maior tempo de vida. Esses resultados relativos à figura 4.1 demonstram uma concordância e reforço da necessidade de controle sobre super nós, especialmente dado que a recepção de dados marcados na rede ocorre em mais de 70% das vezes por transmissão direta e não por retransmissão.

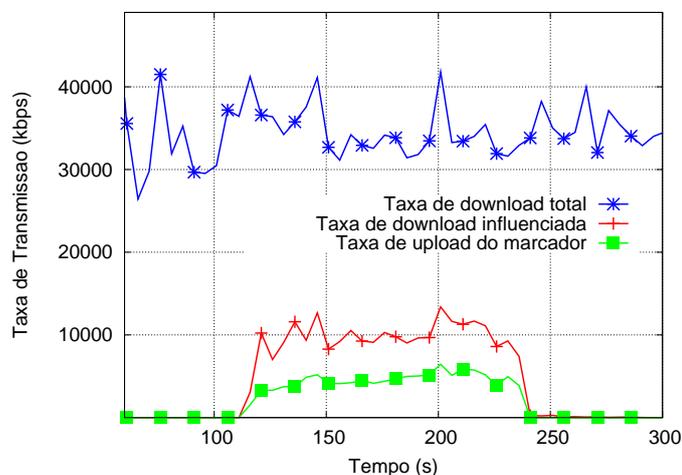


Figura 4.2: Uso de Banda da Rede

A figura 4.2 evidencia as taxas de *upload/download* médio da rede dando ênfase nas taxas relacionadas com a marcação de pacotes. Como é possível observar, a contaminação dos pares é praticamente instantânea, em poucos segundos após a inicialização do esquema de marcação ($t = 110$) a rede está tomada de pacotes modificados, o que demonstra a suscetibilidade do sistema. Acrescentando à análise, a descontaminação ($t = 240$) foi similarmente rápida, depois do fim do ataque a rede retransmite os últimos

pacotes marcados produzidos pelo nó manipulador. Durante as capturas a descontaminação completa foi praticamente instantânea em todas as vezes, corroborando com a análise anterior sobre o tempo de vida do pacote. Acredita-se que o tempo máximo de descontaminação corresponde ao tamanho do *buffer* de vídeo do cliente SopCast, supostamente de 60 segundos.

O percentual de influência sobre a banda total consumida pela rede é também observado na mesma figura 4.2. Os dados mostram que a rede consumiu pacotes assinados numa taxa entre duas e três vezes maior que a taxa de geração, ou de *upload* do nó manipulador de pacotes. De acordo com os dados obtidos das capturas, a taxa de *download* total da rede não é influenciada pela quantidade de pacotes marcados inseridos ou transmitidos, fortalecendo a conclusão de que os nós ignoram o conteúdo alterado na retransmissão desses pacotes. Em todos os experimentos a taxa de *upload* do nó marcador foi praticamente a mesma, não dependeu do tamanho da rede ou da taxa total de *download*. Coincidente com essa análise, o grau de parcerias no nó marcador foi similar em todas as capturas e aparentaram saturar em 50 nós. Para o grupo de experimentos, a quantidade de pacotes marcados recebidos variou sua ocupação aproximadamente entre 30 e 70% do total de pacotes recebidos por toda a rede. Com isso é possível observar o poder de transmissão de um super nó, de forma análoga a um cenário sem manipulação de pacotes, lembrando que a diferença entre pacotes normais e assinados é mínima e não influencia na difusão dos pacotes de vídeo.

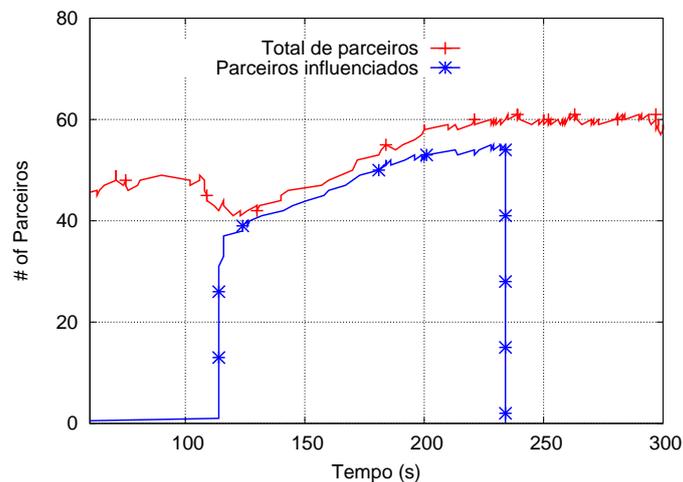


Figura 4.3: Parcerias do Nó Marcador

Na figura 4.3 observa-se a quantidade atual de sessões de parcerias do nó manipulador, assim como as parcerias com pares que receberam pacotes marcados recentemente (considerados influenciados). As sessões de parceria são consideradas desfeitas quando se passa mais do que três minutos sem comunicação entre os pares. Considera-se

que um nó está sob influência de pacotes marcados quando recebe um pacote modificado e permanece nesse estado até o pacote não servir mais para ser retransmitido. De acordo com a análise do gráfico 4.1b, a sessão de influência foi assumida como 30 segundos ignorando aproximadamente 1% de chance dele ainda manter um pacote assinado. Se nesse intervalo um novo pacote for recebido o temporizador é reiniciado. Ainda, em alguns experimentos nota-se que apesar do grau de parceiros influenciados ser próximo de 50 pares houve situações onde o marcador mantinha mais parceiros e um alto *churn* no conjunto de pares influenciados, ou seja, o conjunto de parceiros possíveis era maior que o conjunto instantâneo de influenciados que varia a cada instante, apesar de manter um tamanho fixo. Essa observação foi possível graças aos limites temporais diferentes entre uma sessão de parceria e de influência.

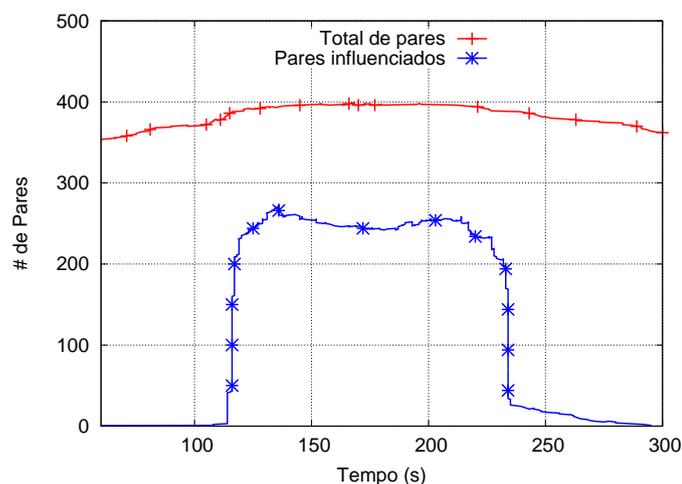


Figura 4.4: Parceiros na Rede

O percentual de contaminação de pares a partir da visão da rede variou entre aproximadamente 50 e 75%, enquanto o número de pares contaminados na rede ficou entre 3 e 5 vezes maior que o número de pares contaminados diretamente pelo nó manipulador de pacotes. A figura 4.4 exibe quantos nós da rede foram atingidos pelo mesmo ataque mostrado na figura anterior.

Como conclusão, pelo percentual elevado de contaminação de parceiros e de banda, percebe-se que um ataque que visa a degradação da qualidade do sistema como um todo é possível de ser obtido a partir de um único nó marcador, dado que este seja suficientemente solicitado, ficando caracterizado como super nó. Mesmo que a taxa de dados marcador recebida por um nó seja menor que a taxa do vídeo, a chegada de alguns poucos pacotes danificados já resultaria na má decodificação, afetando a percepção do usuário. Além disso, supõe-se que essa degradação, mesmo que num conjunto pequeno de nós, geraria um efeito em cadeia, onde cada usuário que saísse do

sistema aumentaria a probabilidade de um usuário ainda não influenciado pela mídia corrompida ser contaminado.

Capítulo 5

Caracterização de Super Nós através de Cálculos de Centralidade

O objetivo deste capítulo é a caracterização de super nós de um sistema P2P de transmissão de mídia ao vivo baseado em malha que usa mecanismo de transmissão *pull* usando métricas de centralidade. Como visto na seção 2.3, e com base na forma com a qual o conteúdo é disseminado através da rede, supõe-se, então, que tais métricas podem prover subsídios para posterior detecção da forma com a qual protocolos dessa categoria influenciam no aparecimento de super nós, dado que a função das métricas de centralidade é determinar a importância relativa de um vértice num grafo (no caso, um nó numa rede).

Os dados deste capítulo foram previamente publicados em Oliveira et al. [2010].

5.1 Metodologia de Análise dos Dados

Para estudar as diversas métricas de centralidade na *overlay* de sistemas P2PTV utilizou-se três cenários de dados experimentais: o real, cujas amostras foram capturadas para as análises dos últimos capítulos, baseadas na metodologia da seção 3.2, e dois simulados, no OverSim, representando redes onde os pares têm uma distribuição de largura de banda heterogênea e homogênea. Cada cenário conta com cinco amostras. A opção por dois cenários de simulação foi embasada em dois fatores: no desconhecimento da largura de banda dos nós da rede real e nas diferenças comportamentais desse tipo de protocolo no que diz respeito ao escalonador de *chunks*. Dado que na implementação feita para o simulador o algoritmo de escalonamento de *chunks* atua de forma a solicitá-los do parceiro mais rápido, se todo nó tem a mesma velocidade, o escalonamento realiza uma seleção aleatória entre as possíveis origens para cada *chunk*,

caso contrário é muito provável que o algoritmo selecione sempre os mesmos vizinhos durante as requisições. O objetivo de trazer cenários simulados foi tentar comparar e validar o que foi observado no cenário real.

De cada experimento foram calculados ou observados, para cada nó, as seguintes métricas: taxa de *upload*, grau, *closeness*, distância ao servidor de conteúdo, *betweenness* normal e dos nós à fonte. Foi feita uma ordenação dessas medidas a fim de formar *ranks*, ou seja, é associado o número 1 ao nó mais importante para uma dada métrica, ao segundo mais importante, 2, e assim por diante. Para a taxa de *upload*, grau, *closeness* e ambos os *betweenness*, os nós mais próximos da primeira posição são aqueles com maiores valores para a medida, enquanto que para distância, são os que têm os menores valores. Por exemplo, para distância, o nó mais perto da fonte é possivelmente o mais valioso por que receberá a mídia primeiro portanto sua posição no *rank* terá um valor mais próximo de um.

A criação dos *ranks* foi feita de forma a não ter um critério de desempate. Se mais de um par tiver o mesmo valor para uma dada métrica, a posição deles no *rank* será igual e a posição do próximo nó com valor de métrica diferente será incrementado do número de nós presentes na posição anterior do *rank*. Essa metodologia alcança resultados interessantes para quase todas as medidas, tendo em alguns poucos casos resultado visualmente em patamares nos gráficos de correlação. Um exemplo disto foi distância dos participantes à fonte, uma medida teve pouca variabilidade, o que implica muitos nós empatados numa mesma posição do *rank*. Por isso, a fim de melhorar o resultado dessa medida específica, foi criada uma métrica baseada na ordenação do par de métricas (distância, grau), tornando o grau o critério de desempate.

Para gerar as análises foram remontados os grafos da rede a partir dos *logs* de troca de fluxo de mídia, ignorando a parte inicial da coleta a fim de eliminar instabilidades típicas da inicialização da rede. Neste capítulo, as métricas descritas anteriormente (em 2.3) foram calculadas a cada 5 segundos sobre um grafo temporalmente dinâmico com base no intervalo entre 100 e 300 segundos dos *logs*. Os *ranks* são obtidos sobre o valor médio de cada métrica em cada nó. Para apresentação dos resultados foram calculadas as médias entre os *ranks* das métricas de cada rodada do experimento. Finalmente, os *ranks* de centralidade são traçados contra o *rank* de *upload* e o coeficiente de correlação de Pearson[Jain, 1991] é calculado para cada cenário.

	Grau	Close.	Distância	(Dist.,Grau)	Betw.	B.Fonte
Real SopCast	0.8893	0.8869	0.7994	0.8345	0.8319	0.7562
Sim. Heterogêneo	0.6864	0.6925	0.2960	0.5725	0.7204	0.6991
Sim. Homogêneo	0.1683	0.2316	0.7378	0.7213	0.1236	0.9446

Tabela 5.1: Coeficientes de Correlação de Pearson.

5.2 Análises de Correlação

A tabela com coeficientes de Pearson (tabela 5.1) sumariza os resultados e mostra que grau e *closeness* resultam em *ranks* sistematicamente mais correlatos com o *rank* de taxas de *upload* que as demais métricas. Os resultados da simulação heterogênea validam esse resultado indicando ainda que o *betweenness* é também uma métrica com forte correlação, apesar de não ter sido tão interessante para esse conjunto de dados reais. O cenário real e simulado heterogêneo são os que se equivalem e que geram conclusões mais correlatas entre si. Finalmente, foi observado que o *betweenness* dos nós somente para a fonte alcançou a melhor correlação entre todas dado o cenário onde a rede tem uma característica homogênea de pares.

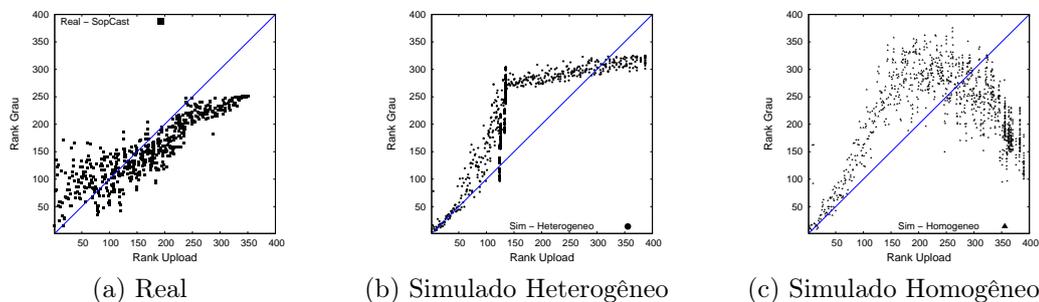


Figura 5.1: Correlação entre ranks de Grau e Upload.

A primeira medida analisada foi centralidade de grau, representada pela figura 5.1. A imagem 5.1a é uma das que apresentam melhores correlações visuais entre as métricas para os dados reais. A tabela 5.1 também confirma esse dado numericamente através do coeficiente de correlação de Pearson entre taxa de *upload* e grau. É possível observar que a dispersão dos pontos dos experimentos reais no início do *rank* (até 100) tem menos precisão do que o resto dos pontos e baixa exatidão comparado com o resultado esperado, que era de que relação linear perfeita entre as variáveis. Apesar disso, em 80% dos experimentos o nó de maior taxa de *upload* teve também o maior grau. Os dados simulados da rede heterogênea aparentam ter dois crescimentos diferentes por

que alguns experimentos tiveram pouca variação da taxa de *upload* e muitos nós (mais de 50%) não fizeram *upload* portanto saturando o *rank*. No entanto, a correlação no início da curva (até 50) é visualmente linear. Este comportamento é melhor explicado pelo modelo de disseminação em árvore: um nó que possui mais parceiros tem grandes chances de emergir como um super nó. Além disso, pode ser inferido que a forma de disseminação supera os modelos de organização da *overlay* e da *underlay*. Finalmente, na distribuição de *ranks* de graus da simulação homogênea da figura 5.1c observamos uma forma de sino indicando uma correlação quase inexistente entre as métricas para esse caso.

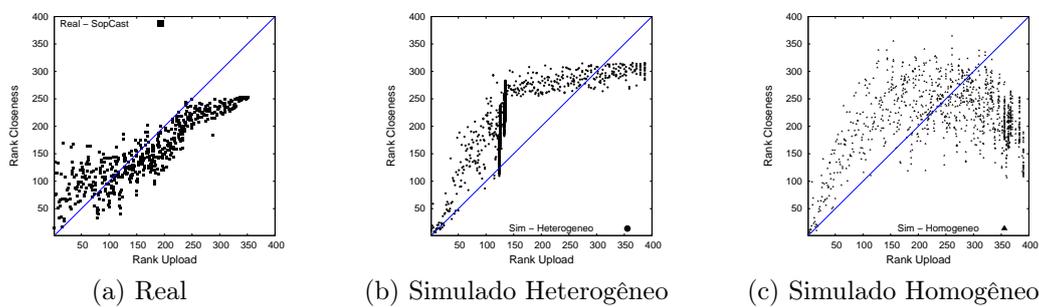


Figura 5.2: Correlação entre ranks de Closeness e Upload.

Os resultados para *closeness* (figura 5.2) obteve resultados similares aos para grau, inclusive no que diz respeito a análises mais finas como: o par de maior *closeness* ser o de maior taxa de *upload*. Numericamente é possível observar na tabela de coeficientes que essa métrica é similar à métrica de grau.

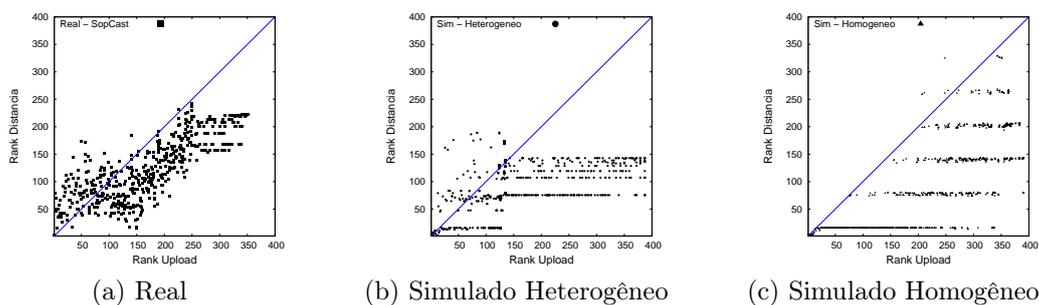


Figura 5.3: Correlação entre ranks de Distância e Upload.

A terceira métrica, distância, presente na figura 5.3, apresentou a pior correlação visual apesar de posar numericamente como uma correlação pouco mais fraca que as últimas para o caso real. No gráfico 5.3a a distribuição real teve poucos patamares por conta da média dos experimentos contudo geralmente esta métrica teve pouca variação,

assim, aglomerando a análise em níveis. De toda forma, isso permitiu deduzir que a árvore de disseminação tenta se manter curta mas com carga baixa na raiz, uma vez que, para distância do nó-fonte > 1 , *ranks* baixos (menores distâncias) concentram mais pontos. Ademais, distância não aparenta ser uma boa métrica neste cenário onde as parcerias entre nós variam com o tempo e, apesar de numericamente a simulação homogênea encontrar um bom resultado, é possível observar graficamente que os *ranks* a taxa de *upload* não correspondem com alta exatidão aos *ranks* dessa métrica.

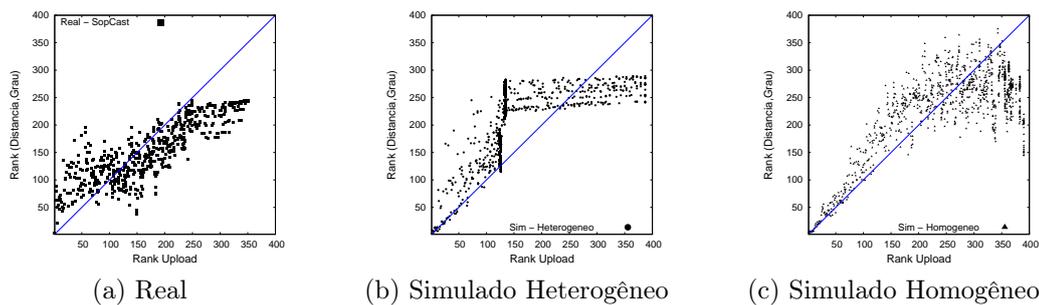


Figura 5.4: Correlação entre ranks de (Distância,Grau) e Upload.

Na figura 5.4, é mostrada a métrica que usa o par (distância, grau) ordenado para formar o *rank*. Essa medida foi capaz de melhorar a variação de valores do eixo Y, especialmente sobre a média dos experimentos, mas o resultado geral não melhorou muito. Visualmente alguns dos níveis observados na figura 5.3 são ainda um pouco presentes na figura 5.4. Numericamente a métrica composta alcançou um resultado intermediário comparado com seus componentes.

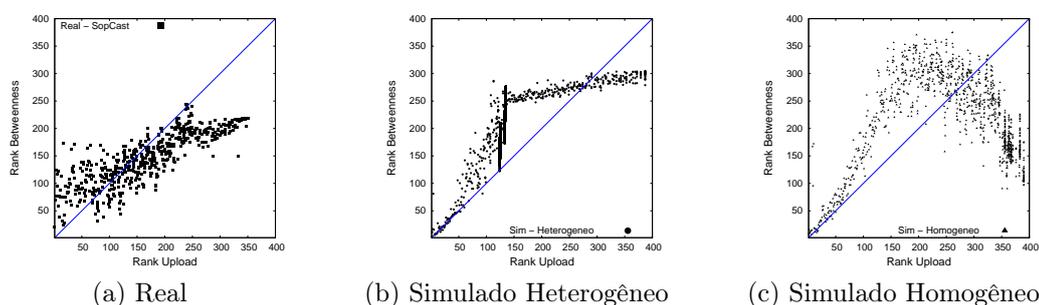


Figura 5.5: Correlação entre ranks de Betweenness e Upload.

O gráfico do *rank* de *betweenness* pode ser visto na figura 5.5. A análise é novamente similar à das métricas de *closeness* e grau, trazendo na prática poucas diferenças. Todavia, de acordo com a tabela 5.1 essa métrica é menos efetiva que as citadas para

o caso real apesar de ser até melhor que as demais de acordo com a simulação heterogênea.

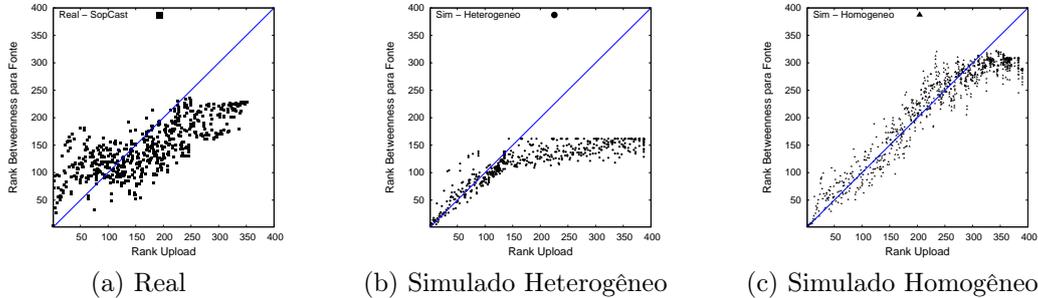


Figura 5.6: Correlação entre ranks de Betweenness para Fonte e Upload.

Finalmente, a última métrica analisada foi o *betweenness* de todos os nós somente até a fonte, que deu resultados bem diferentes para os três cenários, como mostra a figura 5.6 e a tabela 5.1. Essa exceção era esperada justamente por conta da diferença de funcionamento do escalonador. Nos cenários heterogêneos é mais provável que os nós mais distantes do servidor façam nenhum ou pouco envio e que sejam aqueles que têm os menores valores para essa métrica. Além disso, os caminhos de distribuição de dados são mais fixos que no homogêneo. A rede analisada é remontada a partir dos *logs* de distribuição, e nos cenários heterogêneos a métrica tem menor variabilidade já que ao longo do tempo o grafo sobre o qual se faz as análises muda pouco. Ou seja, como podemos verificar através da figura 5.6c, essa métrica criada é muito boa para identificar o *rank* de *upload* dos nós num cenário de seleção de *chunks* de pares aleatório, mas não para o cenário real que é heterogêneo e o escalonamento leva banda em consideração.

No capítulo mostra-se que tanto a centralidade de grau quanto o *closeness* oferecem as melhores correlações com taxa de *upload* entre todas as métricas de centralidade testadas (grau, *closeness*, distância, o par (distância, grau), *betweenness* e *betweenness* dos nós para a fonte). Através do conjunto de dados da simulação com pares heterogêneos foi possível validar os resultados obtidos e induzir ao entendimento de a influência do modelo de disseminação é superior ao do modelo de *overlay* e de *underlay*, ou seja, o algoritmo de escalonamento de *chunks* influencia mais na correlação do que a organização das redes sobreposta e física. Além disso, através do *betweenness* dos nós para a fonte, uma modificação da métrica de *betweenness* original, foi possível encontrar a melhor correlação num ambiente de pares homogêneos ou no qual o algoritmo de escalonamento escolha parceiros aleatórios que, minimamente, tenham o *chunk* desejado.

Capítulo 6

Protocolo para Determinação de Super Nó

Esse capítulo apresenta o estudo proveniente das alterações do protocolo das redes de P2PTV baseadas em malha que permitem determinar quais nós da rede se tornarão super nós. Essas modificações permitem a montagem de redes com duas topologias: a normal, criada pelo algoritmo original e sem definição conhecida da posição dos super nós, e uma de duas camadas, onde os super nós são estrategicamente posicionados. O capítulo traz ao final os resultados e melhorias possíveis obtidos através do controle e da gestão topológica dos super nós.

6.1 Contextualização

Ao longo dos últimos capítulos foi possível observar o potencial de um super nó, sua importância relativa e como essa característica se relaciona com as métricas de centralidade da rede. Entretanto, na prática, escolher ou determinar os super nós é mais vantajoso do que esperar estatisticamente que eles estejam distribuídos e comportem-se da forma desejada.

Os esforços contidos neste trabalho estão direcionados ao entendimento, exploração e o aproveitamento dessa característica emergente da rede. Dado que super nós existem, eles podem ser protegidos, dimensionados, imunizados ou reposicionados na rede trazendo benefícios para a qualidade da transmissão ao tornarem-se mais confiáveis, disponíveis e ricos em recursos. Acredita-se que, com isso, é possível reduzir impactos do *churn*, da poluição de conteúdo, melhorar a capacidade de atendimento ou diminuir o tráfego da rede física. Nesse capítulo, ao invés de buscar formas de iden-

tificar a emergência natural de super nós, é modificado o protocolo a fim de permitir a determinação de super nós dentre nós comuns.

Acredita-se que o controle e gestão dos super nós pode atrair o interesse dos provedores de conteúdo, que participariam da rede não só com a máquina servidora da mídia mas, também, com os nós mais importantes, aumentando custos de transmissão para diminuir o impacto de problemas como a alta rotatividade dos participantes e possíveis ataques. Em outro cenário, analisando um modelo de negócio onde existe uma entidade que intermedia o serviço de publicação do *stream*, é de grande utilidade fornecer funções de reorganização da rede atrelados a diminuição de gargalos originados nos espectadores ou a melhoria de qualidade, especialmente num cenário de grande dinamismo dos usuários como descrito em Borges et al. [2009].

Para traduzir esse entendimento sobre os benefícios e verificar tais hipóteses é definida uma proposta da separação dos nós da rede em duas camadas. A implementação do modelo original da rede prescreve comportamento homogêneo para qualquer nó que não seja o servidor. Apesar disso, observa-se que os nós se diferenciam por características de transmissão. Através de algumas modificações nos algoritmos originais (descritas na seção 6.2.1) objetiva-se diferenciar os super nós por indicação, ou seja, será possível determinar ou apontar um nó para que o mesmo se destaque. Com isso, quer se criar um cenário onde exista uma camada de super nós permanentes, que não se tornem poluidores e que apoiem diretamente o servidor de mídia. Participantes tão idôneos, disponíveis e úteis quanto o próprio servidor da mídia que, de maneira abstrata, atuariam como um servidor distribuído do *stream*. A segunda camada de nós é representada pelos demais nós, os espectadores comuns do sistema.

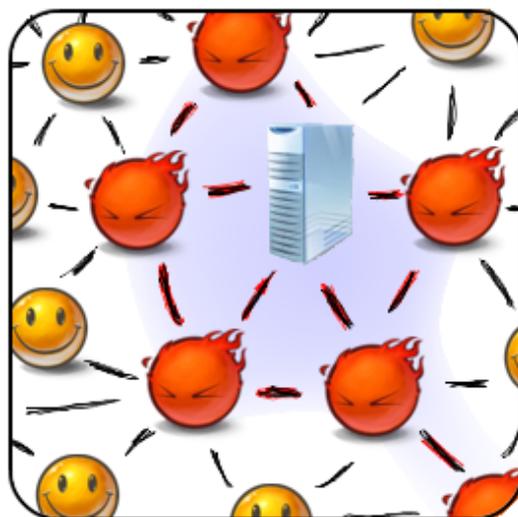


Figura 6.1: Proposta de topologia em duas camadas

Apesar das modificações propostas, a determinação de super nós não elimina a característica natural da rede de surgimento de super nós. Ela é, contudo, burlada. Ao definir um super nó, ele passa a atuar de tal forma a ocupar as vagas naturais de super nós disponíveis na rede. Ou seja, um grupo de super nós indicados tende a ocupar as posições estatísticas de super nós naturais garantidas pela distribuição da taxa média de *upload*, presente na figura 3.2 (a parte mais alta da curva). Portanto, ao definirmos que todos ou nenhum dos nós da rede são super nós o esquema original é retomado.

6.2 Simulador de P2PTV baseado em malha

Para realização dos experimentos utilizou-se ambiente simulado. Como pontuado na seção 3.1, foi implementado um sistema P2P de transmissão de mídia contínua ao vivo no OverSim baseado no artigo do CoolStreaming [Zhang et al., 2005] aliado a conhecimentos adquiridos através das observações dos experimentos com o SopCast. Tanto CoolStreaming quanto SopCast são exemplos de sistemas P2P da geração atual baseados em malha e mecanismo de transmissão *pull*.

A construção e posterior manutenção da malha depende de um nó especial chamado servidor de *bootstrap*. Sempre que o nó necessitar de novos parceiros ele requisita uma lista de possíveis vizinhos ao *bootstrap*. Essas requisições são lançadas em dois casos: quando a quantidade de parceiros estiver abaixo do mínimo e quando os requisitos de continuidade não estiverem sendo alcançados. Cada nó tem limiares inferior e superior da quantidade de parcerias desejadas. Parceiros são descartados em duas situações: quando passam alguns segundos sem se comunicar com o nó ou quando um nó já no seu limiar superior de quantidade de parcerias aceita adquirir um novo parceiro. Nesse último caso, desfaz-se a parceria menos útil, aquela que fez menos envios e recebimentos de *chunks*.

O servidor de mídia cria o *stream* periodicamente organizando-o em *chunks*, que têm marcadores temporais indicando sua posição em relação aos demais. Periodicamente todos os nós consomem os pedaços do *stream* na mesma taxa de produção da mídia. A fim de suavizar a exibição da mídia os pares armazenam os trechos num *buffer*. Essa estrutura também serve para anunciar quais são os trechos presentes no nó. As requisições por *chunks* para preencher o *buffer* ocorrem seguindo a regra *earliest deadline first* (EDF) [Burns & Wellings, 2009] onde os blocos mais próximos do prazo de exibição são solicitados. Durante o *bootstrap*, os pares adquirem seus prazos de exibição, que são compostos pelo marcador temporal do pedaço mais recentemente produzido no servidor de mídia somados a um valor de latência (essencial para permitir

o preenchimento dos *buffers* e a subsequente transmissão par-a-par). A requisição de *chunks* é feita periodicamente e em intervalos mais curtos que a taxa de produção. Cada requisição a dado pedaço tem uma validade para ser refeita (isso visa minimizar o recebimento de duplicatas). Para saber o que se pode solicitar de cada vizinho, os pares trocam mapas de bits anunciando sua disponibilidade de *chunks* no *buffer*. Essa mesma mensagem é utilizada como forma de identificar se o vizinho ainda compõe a rede. Ao receber mensagens com mapas de *buffer* dos vizinhos, o nó pode determinar quais *chunks* têm potencial para serem solicitados de que parceiros comparando os mapas alheios com o seu próprio.

Diversos fatores podem fazer um par perder sincronismo com a mídia, por exemplo, muitos vizinhos deixando o sistema ao mesmo tempo. Nessa situação, o par interrompe a exibição, esvazia o *buffer*, requisita novos parceiros e prazos de exibição. É importante ressaltar que, apesar disso, uma vez que a mídia começa a ser exibida, as medições para geração de estatísticas continuam a ser gravadas até que o nó saia da rede.

Periodicamente são capturados em cada participante métricas como: a latência, continuidade, quantidade de poluição, percentual de preenchimento do *buffer*, taxas de *upload* e *download* de mídia e total, tamanho da vizinhança, tamanho da vizinhança especial (super nós e servidor) e desperdícios causados por entregas de *chunk* atrasadas ou duplicadas. Também são medidos parâmetros globais da rede como o latência, continuidade e quantidade de poluição médias, tamanho e tempo médio de vida e de espera antes de retornar dos participantes.

6.2.1 Alterações nos algoritmos

Para promover a determinação de um super nó e a separação em camadas foram feitas três modificações no protocolo proposto pelo CoolStreaming: o escalonamento preferencial, a persistência de parceria e o *bootstrapping* diferencial. Essas alterações permitem parametrizar a simulação de forma a definir quais nós pertencem ao grupo que atua como super nó. Abaixo será descrita cada uma delas individualmente.

Escalonamento preferencial é a propriedade que determina que qualquer nó sempre dá preferência a baixar um *chunk* de um super nó. Originalmente, a seleção da fonte de um determinado pedaço a ser buscado relaciona todos os parceiros que detêm o pedaço-alvo numa lista e seleciona, dentre eles, o vizinho que tem a melhor capacidade de servi-lo. A modificação cria uma segunda lista onde são inseridos somente parceiros que sejam super nós e que tenham o pedaço, se ela não

estiver vazia ela é utilizada na etapa que busca o melhor vizinho; caso contrário, utiliza-se a lista original.

Persistência de parceria é a propriedade que determina que qualquer nó, ao desfazer parcerias, sempre remove preferencialmente vizinhos que sejam nós comuns. Sabe-se que um nó pode desfazer parcerias com vizinhos que são pouco úteis, ou seja, que efetuam poucas ou nenhuma troca de dados com ele. Apesar de ser mais provável que parcerias com nós comuns sejam desfeitas antes das parcerias com super nós, não seria impossível que isso acontecesse. A modificação garante que parcerias com super nós sejam mais duradouras.

Bootstrapping diferencial é a propriedade que determina que, quando um super nó realiza uma consulta por parceiros ao servidor de *bootstrap*, a lista de nós de resposta deve ser composta por um percentual p de outros super nós ou o servidor de mídia. Com essa alteração os super nós são atraídos para perto uns dos outros e do servidor de mídia. A idéia é que a camada de super nós forme um grafo conectado englobando a fonte do *stream* para garantir um caminho rápido e confiável do conteúdo.

Essas alterações combinadas promovem a separação em camadas. Contudo, como todas as modificações são dependentes da indicação explícita de quais nós da rede são super nó, é possível, mesmo com essa implementação, preservar o protocolo original e suas funcionalidades. Portanto, a adoção das modificações é opcional e escalável, na medida em que é possível dinamicamente inserir ou remover um super nó na rede. Além disso, não gera restrições para provedores de conteúdo que não tenham intenção de adotar essa visão dado que eles podem privar-se da implantação de super nós.

6.3 Análise Experimental

6.3.1 Sumário dos Experimentos

Relativo a configurações gerais para todas as rodadas, o simulador cria uma rede com 400 nós iniciais. Cada rodada simula 300 segundos de interação entre pares da rede *overlay*. A organização da camada física é ignorada e não há erros ou falhas na transmissão de pacotes. Apesar disso os nós tem uma largura de banda definida e pacotes podem ser descartados por atraso.

A distribuição de banda segue o relatório da Akamai do segundo semestre de 2009 que descreve a situação da Internet no mundo [Akamai, 2009]. Especificamente,

Tabela 6.1: Sumário de Parâmetros do Simulador

Valor	Descrição
400	Quantidade inicial de pares
300	Limite de tempo da simulação (s)
100	Repetições para cada cenário
8-12	Quantidade de parcerias esperadas no servidor
40-60	Quantidade de parcerias esperadas nos demais nós
6	<i>Timeout</i> de parceria (s)
90	Nível de continuidade esperado (%)
30	Nós com largura de banda de 256Kbps (%)
40	Nós com largura de banda de 2Mbps (%)
30	Nós com largura de banda de 5Mbps (%)
5	Largura de banda do servidor e super nós (Mbps)
345	Taxa de produção/reprodução do <i>stream</i> (Kbps)
7200	Tamanho do <i>chunk</i> (Bytes)
30	Tamanho do <i>buffer</i> (<i>Chunks</i>)
0.5	Período entre requisições de <i>chunks</i> (s)
6	<i>Chunks</i> requisitados por período
1	<i>Timeout</i> de requisição de <i>chunk</i> (s)
1	Período entre produção de <i>chunks</i> (s)
1	Período entre consumo de <i>chunks</i> (s)
1	Período entre troca de mapas de bits do <i>buffer</i> (s)
3-5	Latência inicial de um nó (s)
50	Limiar do <i>bootstrapping</i> diferencial (%)

nele está indicado que 5% dos usuários dos Estados Unidos tem conexão inferior a 256Kbps, 57% superior a 2Mbps e 24% superior a 5Mbps. Para simplificar foi adotado uma distribuição similar onde 30% dos nós tem banda de 256Kbps, 40% de 2Mbps e 30% de 5Mbps. O servidor e super nós, quando definidos, são nós confiáveis e de mais recurso portanto tendo uma banda de 5Mbps.

A quantidade de parcerias esperada pelo servidor está entre 8 e 12 nós enquanto que os demais nós podem ter entre 40 e 60. Se um nó não receber mensagens do seu vizinho por mais de 6 segundos ele o descarta e requisita novas parcerias. Outra forma de requisitar mais parceiros é caso a taxa de continuidade do *stream* esteja abaixo do esperado, que é de 90% dos *chunks* exibidos a cada segundo. O *stream* gerado pelo servidor tem uma taxa de reprodução aproximada de 345Kbps sendo dividido em pedaços de 7200 *bytes*.

A cada segundo *chunks* são produzidos no servidor, consumidos em todos os nós e mensagens de troca de mapa de bits do *buffer* são trocados entre os parceiros. Apesar disso, requisições por novos pedaços são criadas a cada meio segundo e expiram depois

de um segundo sem resposta. A motivação para requisições de mais do que se pode consumir em um período está no preenchimento do *buffer*, que é capaz de armazenar 30 *chunks*, ou 5 segundos. Para simulação de transmissão ao vivo cada nó ao ser iniciado ou reiniciado começa a realizar solicitações sobre *chunks* com 3 a 5 segundos de atraso relativo ao pedaço mais novo produzido pelo servidor.

O parâmetro referente ao *bootstrapping* diferencial foi mantido em 0.5 fazendo com que o servidor de *bootstrap* responda a cada consulta por novos pares originada de um super nó com uma lista onde metade dos nós são super nós ou o servidor.

A tabela 6.1 sumariza os parâmetros gerais das simulações descritos acima.

Foram realizadas duas classes de experimentos: uma relativa a variações na geração de *churn* e uma relativa à resiliência a poluição de conteúdo. A seguir, são descritos os cenários utilizados e parametrização específica. Cada cenário foi executado 100 vezes utilizando sementes distintas. As múltiplas execuções permitem a formação de redes conectadas de formas diferentes, com diferenças no *churn*, na banda dos nós e conseqüentemente no escalonamento de *chunks*, o que implica em uma variância nos resultados.

6.3.1.1 Churn

De acordo com Borges et al. [2009], o comportamento de usuários de uma rede real depende do conteúdo exibido e implica modelos de carga diferentes. O trabalho mede e discute dois cenários: um onde o conteúdo é o comum ou diário (tele jornal, novela) e outro onde o conteúdo é especial, caracterizado por ser um evento incomum de interesse pontual (no caso foi analisado uma partida de final de campeonato de futebol mas poderiam ser conteúdos como as olimpíadas ou a copa do mundo).

O cenário adotado visa avaliar o impacto da determinação de super nós para três modelos de geração de *churn* diferentes: sem *churn*, com *churn* normal e com *churn* ocasionado por eventos. Estes dois últimos padrões, cujas distribuições (de tempo de permanência e de espera fora da rede) serão utilizadas a seguir, estão descritos analiticamente no trabalho de Borges et al. [2009]. Na ausência de *churn* os 400 participantes iniciais não saem da rede até o fim da simulação. Com comportamento de *churn* normal o tempo de vida dos participantes na rede segue uma distribuição Lognormal ($\mu=0.823286$ e $\rho=1.458894$). Já com *churn* ocasionado por eventos essa distribuição torna-se uma Weibull ($\lambda=2.03178$ e $\kappa=0.233279$). Apesar da diferença no tempo de vida, o tempo que um nó aguarda fora da rede para voltar em ambos os casos segue distribuição Exponencial ($\lambda=18.4908$). Esse nó, ao retornar, é tratado como um novo nó por não armazenar parcerias antigas ou ter dados frescos que permita contribuir

com a rede.

Para cada um dos modelos foi variada a quantidade de super nós da rede em: 0%, 1% e 5% por cento de super nós do total de nós iniciais. Dado a suposição de que os super nós são robustos e disponíveis, os determinados (pela parametrização do simulador) não sofrem *churn*. Para verificar as vantagens do uso de super nós mediu-se a continuidade e latência desses sistemas.

6.3.1.2 Resiliência

Para essa classe de experimentos o *churn* foi eliminado para evitar interferência dessa variável e melhor evidenciar os resultados. Nessa classe o que se estuda é a quantidade percentual média de poluição que atinge os usuários. Poluidores, que podem ser um ou mais nós, ao responderem requisições de *chunks* o fazem marcando o dado como poluído. Na simulação admite-se um sistema que não identifica nem age contra dados mal formados, somente os contabiliza, tal qual os experimentos realizados no capítulo 4. Os experimentos foram divididos em três casos.

O primeiro caso é o ataque original. Numa rede sem determinação de super nós são colocados um, um por cento ou dez por cento de nós poluidores aleatórios (não incluindo o servidor de mídia).

O segundo se passa numa rede onde cinco por cento dos nós são explicitamente determinados como super nós por parametrização do simulador. Dado que super nós são confiáveis, nenhum deles inicia o ataque e, como no caso anterior, o ataque combinado é observado a partir dos nós comuns. Todos os nós, inclusive os super nós, repassam poluição ingenuamente.

O último caso equivale ao pior caso possível apesar de ser também mais irreal. Como no caso anterior, cinco por cento dos nós são marcados como super nós. Contudo, os ataques, tanto o simples quanto o combinado, são realizados a partir deles. A improbabilidade do cenário advém da premissa de que esses nós pré-determinados são confiáveis e, portanto, eles não deveriam atuar contra o sistema. Apesar disso, o cenário alerta para o impacto que super nós podem ter no sistema.

6.3.2 Resultados

A seguir estão apresentados os resultados dos experimentos separados por métrica analisada: continuidade, latência e resiliência. Os resultados foram traçados como curvas de função de distribuição acumulativa (CDFs). Cada ponto representa a média entre todos os nós para a métrica observada em uma das 100 execuções do cenário específico de experimento.

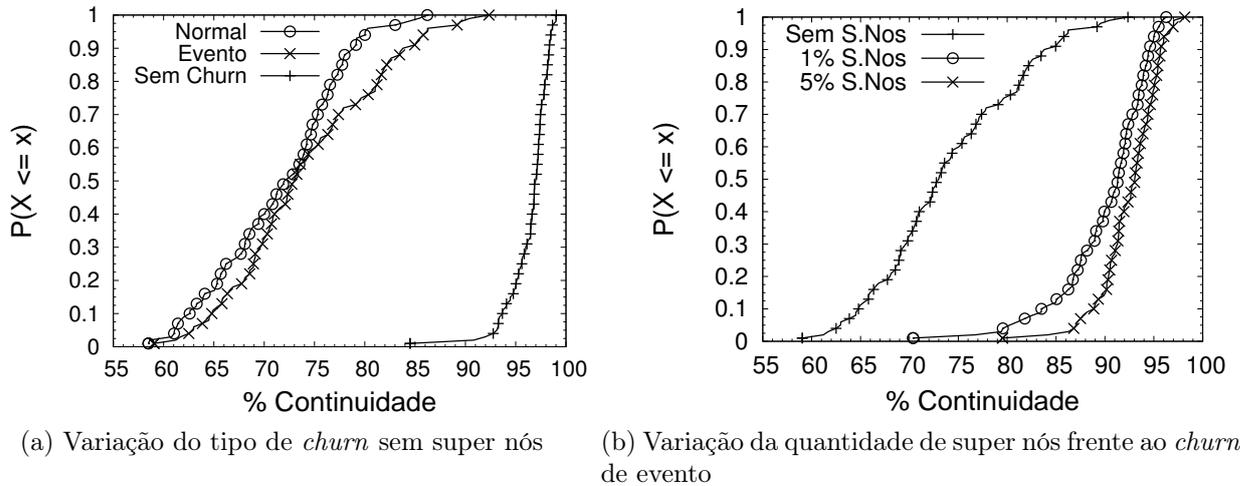


Figura 6.2: Medidas de continuidade. Rede representando o sistema: (a) original com diferentes modelos de geração de *churn*. (b) onde o comportamento de *churn* dos usuários é similar a um momento onde um evento de interesse está sendo transmitido variando a quantidade de super nós determinados.

6.3.2.1 Continuidade

A continuidade significa quanto do *stream* foi recebido pelos pares desde o primeiro segundo de exibição até a saída do par da rede. Uma continuidade baixa indica que muitos *chunks* estão sendo perdidos no momento de exibição criando assim uma sensação para o usuário de mídia entrecortada, com falhas ou travando, o que é geralmente associada à baixa qualidade. Apesar de alguma perda de dados ser aceitável nas transmissões pela Internet, na medida em que essa métrica diminui a mídia passa a ser cada vez menos útil.

A continuidade foi medida como:

$$Continuidade = \overline{\sum_{p \in P} \frac{CC_p}{CE_p}}$$

sendo p o nó, P o conjunto formado por todos os nós que apareceram durante a rodada, CC a quantidade de *chunks* consumidos e CE a quantidade esperada de *chunks*.

A figura 6.2a mostra um problema grave na rede sem super nós, ou seja, o esquema de transmissão ao vivo original. Em casos onde o *churn* é modelado perto do real, a continuidade, que é um indicador de qualidade da mídia em todos os pares, distancia-se do ideal. Para o cenário e parâmetros apresentados, observa-se que a rede tem dificuldade em manter uma boa continuidade da mídia na presença tanto do *churn* normal quanto do de evento. Comparado com uma rede sem *churn*, o *stream* teve, na

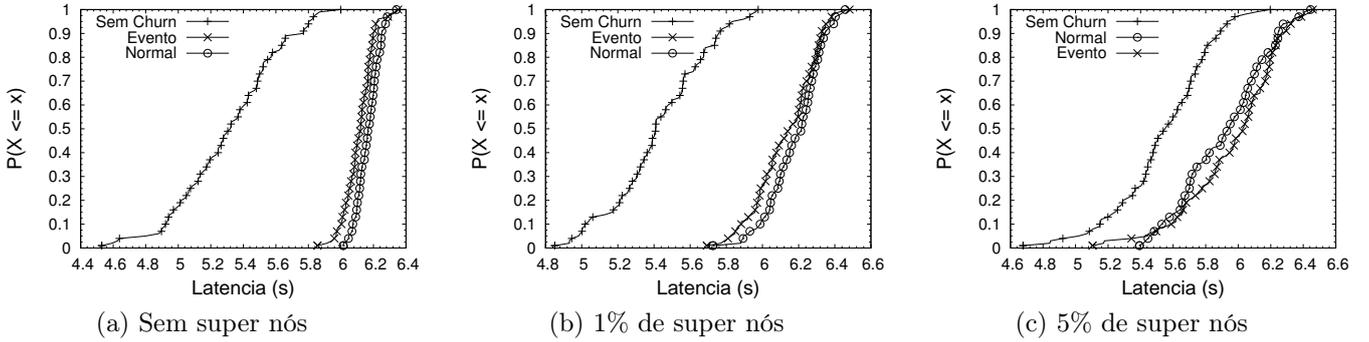


Figura 6.3: Medições de latência para os três tipos de *churn*. Rede representando o sistema: (a) original; (b) com a presença de 1% de super nós determinados; (c) com a presença de 5% de super nós determinados.

média, uma continuidade aproximadamente 23% menor. Outra influência importante no resultado é a escolha de distribuição de banda para os nós. Em experimentos isolados onde todos os nós tinham banda muito superior à taxa do *stream* a continuidade também foi alta (próxima de 100%), contudo, esta não é uma representação satisfatória da realidade.

Apesar disso, na figura 6.2b, nas redes com 1% e 5% de super nós foi possível, na média, melhorar a continuidade original em 16% e 18% considerando o *churn* de evento. Já no cenário de *churn* normal os ganhos foram de 18% e 22% para redes com 1% e 5% de super nós determinados. Com tais melhorias foi possível alcançar resultados em cenários de *churn* tão bons quanto os ambientes sem rotatividade de pares.

6.3.2.2 Latência

A latência é a medida do atraso do que está sendo exibido num nó cliente comparado ao *chunk* mais novo que está sendo produzido no servidor.

$$Latencia = \overline{\sum_{p \in P} LCCT_p - LPCT_s}$$

sendo p o nó, s o servidor, P o conjunto formado por todos os nós que apareceram durante a rodada, $LCCT_p$ o marcador temporal do último *chunk* consumido no par e $LPCT_s$ o marcador temporal do último *chunk* produzido no servidor.

É importante lembrar que cada par já conta com uma latência inicial entre 3 e 5 segundos (uniformemente distribuída entre todos os nós). Por essa razão são observados os ganhos relativos das redes sem super nós, com 1% e 5% deles frente à

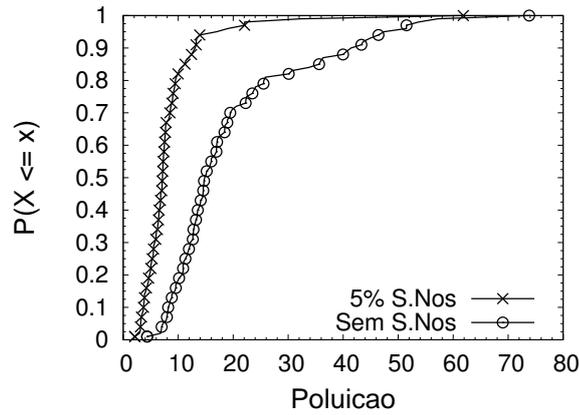


Figura 6.4: Medições de poluição para ataque combinado de 10% dos nós comuns nas redes original (sem super nós) e em duas camadas (com 5% de super nós).

variação do tipo de *churn*.

A figura 6.3 exibe valores de latência para redes sem super nós, com 1% e com 5%, respectivamente. Em cada cenário a influência do *churn* foi medida sobre a latência. Na figura 6.3a, a diferença média entre as latências observadas nas redes com e sem *churn* foi de aproximadamente 0.81 segundos. Já nas figuras 6.3b e 6.3c, essa diferença caiu para 0.725 e 0.4 segundos, respectivamente. Conclui-se com isso que a determinação dos super nós é também benéfica para a diminuição da latência da rede.

Apesar disso, comparando as curvas sem *churn* dos três gráficos é possível observar que a concentração do tráfego com o uso de super nós causou a adição inesperada de um *overhead* na latência, dado que a latência média de cada curva ficou em 5.33, 5.43 e 5.54 segundos respectivamente. Ainda assim, essa perda é menos significativa que o resultado anterior.

6.3.2.3 Poluição

Nesse grupo de experimentos, um alto índice de poluição significa que os *chunks* estão sendo manipulados nos atacantes (parâmetro do cenário) e bastante retransmitidos pelos demais nós, dado que não ocorre verificação do conteúdo.

A quantidade de poluição presente na rede foi calculada como:

$$Poluicao = \overline{\sum_{p \in P} \frac{CP_p}{CC_p}}$$

sendo p o nó, P o conjunto formado por todos os nós que apareceram durante a rodada, CP a quantidade de *chunks* poluídos e CC a quantidade de *chunks* efetivamente

consumidos.

A figura 6.4 mostra um cenário de ataque simultâneo com 10% dos usuários como poluidores. Destaca-se esse cenário por corresponder a um ataque bastante possível e capaz de degradar significativamente o sistema. Para os experimentos apresentados 10% são 40 nós, contudo, atualmente, não é difícil obter acesso a um número muito maior de pares através de *botnets*, por exemplo. Comparando a média dos índices de poluição das redes com e sem super nós foi encontrado o valor aproximado de 8.4% e 20% de poluição. Isso significa que em média uma rede composta de 5% de super nós confiáveis é cerca de 2.5 vezes mais resiliente a ataques originados nos nós comuns. Além disso, o ataque lançado a partir dos 5% super nós alcançou uma taxa média de 72% de degradação de todo dado exibido nos pares. Essa curva foi omitida do gráfico por se tratar de um cenário menos factível e se caracterizar como um ataque extremamente "sortudo", já que consegue comprometer todos os super nós de uma só vez.

Conclui-se, através das análises e resultados comparativos, que as redes P2P de distribuição de mídia ao vivo baseadas em malha-pull podem ser bastante melhoradas através da proposta desta arquitetura híbrida com maior controle sobre os super nós. O sumário dos resultados utilizando a abordagem de duas camadas é de que é possível replicar os altos níveis de continuidade de uma rede sem *churn* para redes com *churn*, diminuir a latência média observada e dividir por mais da metade os índices de poluição da rede através da seleção de super nós confiáveis.

Capítulo 7

Conclusões e Trabalhos Futuros

Redes P2P de distribuição de mídia ao vivo baseadas em malha têm uma característica natural configurada pela importância desproporcional dos nós que têm as maiores taxas de *upload* dentre todos os participantes, aqui classificados como super nós. Neste trabalho esta característica foi explorada de três formas: ao analisar a influência de um super nó sobre a rede, através da marcação de seus pacotes enviados e observação do fluxo destes pela rede; ao caracterizar a rede quanto à correlação de taxas de *upload* e métricas topológicas de centralidade de cada nó; e, ao determinar sua existência em meio ao conjunto de participantes a fim de separar a rede em duas camadas, onde a camada composta por super nós (confiáveis, disponíveis e ricos em recursos) fortalece e complementa diretamente a capacidade do servidor. Foram utilizados dados gerados tanto de experimentos reais, através de *logs* de atividade de rede de centenas de nós no PlanetLab participando da difusão de um canal fechado no SopCast, quanto simulados, através da implementação de um protocolo de rede P2PTV em malha com base em especificações do CoolStreaming.

Os resultados aqui apresentados demonstram, especialmente, a extensa influência dos super nós sobre a segurança e estabilidade da *overlay* estudada. Experimentos em ambiente real mostraram que um único super nó pode, através somente dos dados que envia, ter impacto sobre mais de 30% da banda de *download* da rede e mais de 50% dos pares. Entende-se que isso atribuído a poluição de conteúdo é desastroso para sistemas de P2PTV. A identificação, proteção e manutenção de super nós é um requisito chave para melhoria da qualidade na medida em que ataca alguns dos maiores problemas desse tipo de rede, atribuídos a: latência, continuidade e resiliência a ataques ao conteúdo. Através de simulação foi observado que a composição da rede em duas camadas, tendo super nós indicados, é capaz de reduzir a latência nos participantes, elevar a continuidade do sistema em situações de alta rotatividade dos parceiros a níveis

similares à inexistência de *churn* e reduzir em mais da metade a quantidade média de poluição trafegada na rede para um ataque combinado de 10% dos participantes comparado ao esquema original.

Sugere-se como trabalhos futuros os seguintes:

- Sobre o estudo da influência de super nós:
 - explorar a marcação dos pacotes em todos os participantes da rede a fim de entender a influência de cada nó comparando com sua taxa de *upload*;
- Sobre os estudos de centralidade:
 - verificar a influência da largura de banda de um participante ou tempo de permanência na rede na formação de super nós;
 - combinar as métricas destacadas neste trabalho para apresentar, de forma objetiva, uma metodologia que permita identificar se um determinado participante é efetivamente um super nó;
- Sobre a separação em camadas:
 - observar ganhos obtidos considerando a rede física, como: aumento da capacidade de atendimento global da rede e diminuição de tráfego físico ocasionado por posicionamento estratégico de super nós;
 - revisar e propor novas melhorias do protocolo e de disposição dos super nós para produzir resultados ainda mais eficazes como a eliminação do *overhead* na latência ou incrementar a resiliência à poluição através do bloqueio à poluição passiva nos super nós;
 - explorar propósitos paralelos para a camada de nós confiáveis. Alguns objetivos interessantes seriam relacionados com, por exemplo, obter opiniões seguras de reputação ou realizar identificação e repressão de ataques a partir de uma quantidade reduzida de nós.

Referências Bibliográficas

- Akamai (2009). State of the Internet Report. Technical report, Akamai. Available at: <http://www.akamai.com/stateoftheinternet/>.
- Akamai (2010). Akamai. <http://www.akamai.com/>.
- Ali, S.; Mathur, A. & Zhang, H. (2006). Measurement of commercial peer-to-peer live video streaming. In *Proc. of Workshop in Recent Advances in Peer-to-Peer Streaming*.
- Androutsellis-Theotokis, S. & Spinellis, D. (2004). A survey of peer-to-peer content distribution technologies. *ACM Computing Surveys (CSUR)*, 36(4):371.
- Banerjee, S.; Bhattacharjee, B. & Kommareddy, C. (2002). Scalable application layer multicast. In *Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*, p. 217. ACM.
- Baumgart, I.; Heep, B. & Krause, S. (2007). OverSim: A Flexible Overlay Network Simulation Framework. In *Proceedings of 10th IEEE Global Internet Symposium (GI '07) in conjunction with IEEE INFOCOM 2007, Anchorage, AK, USA*, pp. 79–84.
- Bellovin, S. M. (2002). A technique for counting natted hosts. In *IMW '02: Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurement*, pp. 267--272, New York, NY, USA. ACM.
- Biskupski, B.; Schiely, M.; Felber, P. & Meier, R. (2008). Tree-Based Analysis of Mesh Overlays for Peer-to-Peer Streaming. *Lecture Notes in Computer Science*, 5053:126.
- Borges, A.; Almeida, J. & Campos, S. (2008). Fighting pollution in p2p live streaming systems. *Multimedia and Expo, 2008 IEEE International Conference on*, pp. 481–484.
- Borges, A.; Gomes, P.; Nacif, J. A.; Almeida, J. & Campos, S. (2009). Characterizing sopcast client behavior. *Submitted to: Elsevier Computer Communication*.

- Brandes, U. (2001). A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology*, 25(2):163--177.
- Burns, A. & Wellings, A. (2009). *Real-Time Systems and Programming Languages: Ada, Real-Time Java and C/Real-Time POSIX*. Addison-Wesley Educational Publishers Inc, USA.
- Cahill, A. & Sreenan, C. (2004). An efficient CDN placement algorithm for the delivery of high-quality tv content. In *Proceedings of the 12th annual ACM international conference on Multimedia*, pp. 975--976. ACM New York, NY, USA.
- Castro, M.; Druschel, P.; Kermarrec, A.; Nandi, A.; Rowstron, A. & Singh, A. (2003). SplitStream: High-bandwidth content distribution in a cooperative environment. *Proceedings of SOSP*.
- Chawathe, Y.; Ratnasamy, S.; Breslau, L.; Lanham, N. & Shenker, S. (2003). Making gnutella-like p2p systems scalable. In *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pp. 407--418. ACM New York, NY, USA.
- Clarke, I.; Sandberg, O.; Wiley, B. & Hong, T. (2001). Freenet: A distributed anonymous information storage and retrieval system. *Lecture Notes in Computer Science*, pp. 46--66.
- CNET (2010). The next big thing supersession: I want my iptv. Filme-video.
- Coelho, R. V.; Barcellos, M.; Jansch-Pôrto, I. & Gasparly, L. P. (2010). P2p streaming de alta definição: Análise quantitativa de esquemas de assinatura digital para autenticação de conteúdo. In *SBRC 2010*.
- Cohen, B. (2003). Incentives Build Robustness in BitTorrent. In *Workshop on Economics of Peer-to-Peer Systems*, volume 6. Berkeley, CA, USA.
- Comcast (2010). Comcast. <http://www.comcast.com/>.
- Conviva (2010). Conviva. <http://www.conviva.com/>.
- Deshpande, H.; Bawa, M. & Garcia-Molina, H. (2002). Streaming Live Media over a Peer-to-Peer Network. *Stanford database group technical report (2002)*.
- Dhungel, P.; Hei, X.; Ross, K. & Saxena, N. (2007). The pollution attack in P2P live video streaming: measurement results and defenses. In *Proceedings of the 2007*

- workshop on Peer-to-peer streaming and IP-TV*, pp. 323--328. ACM New York, NY, USA.
- Erdos, P. & Renyi, A. (1960). On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci.*, 5:17--61.
- Freeman, L. (1979). Centrality in social networks conceptual clarification. *Social networks*, 1(3):215--239.
- Google Trends (2010). Google trends. <http://www.google.com/trends/>.
- Greeson, M. (2009). Over-the-top, cord-cutting and the consumer. Research report, The Diffusion Group.
- Guha, S.; Daswani, N. & Jain, R. (2006). An experimental study of the skype peer-to-peer voip system. In *Proc. of IPTPS*, volume 6. Citeseer.
- Hefeeda, M.; Habib, A.; Botev, B.; Xu, D. & Bhargava, B. (2003). PROMISE: peer-to-peer media streaming using CollectCast. *Proceedings of the eleventh ACM international conference on Multimedia*, pp. 45--54.
- Hei, X.; Liang, C.; Liang, J.; Liu, Y. & Ross, K. (2007). A Measurement Study of a Large-Scale P2P IPTV System. *Multimedia, IEEE Transactions on*, 9(8):1672--1687.
- Huang, C.; Wang, A.; Li, J. & Ross, K. (2008a). Understanding hybrid CDN-P2P: why limelight needs its own Red Swoosh. In *Proceedings of the 18th International Workshop on Network and Operating Systems Support for Digital Audio and Video*, pp. 75--80. ACM.
- Huang, Y.; Fu, T. Z.; Chiu, D.-M.; Lui, J. C. & Huang, C. (2008b). Challenges, design and analysis of a large-scale p2p-vod system. *SIGCOMM Comput. Commun. Rev.*, 38(4):375--388.
- Jain, R. (1991). *The art of computer systems performance analysis: techniques for experimental design, measurement, simulation, and modeling*. Wiley New York.
- Jannotti, J.; Gifford, D.; Johnson, K.; Kaashoek, M. & O'Toole Jr, J. (2000). Overcast: reliable multicasting with on overlay network. In *Proceedings of the 4th conference on Symposium on Operating System Design & Implementation-Volume 4*, pp. 14--14. USENIX Association Berkeley, CA, USA.
- Joost (2010). Joost. <http://www.joost.com/>.

- Knutsson, B.; Lu, H.; Xu, W. & Hopkins, B. (2004). Peer-to-peer support for massively multiplayer games. In *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 1.
- Koschützki, D.; Lehmann, K. A.; Peeters, L.; Richter, S.; PODEHL, D. T. & Zlotowski, O. (2005). *Centrality Indices*, volume 3418/2005 of *Lecture Notes in Computer Science: Network Analysis*, chapter Part I: Elements, pp. 16--61. Springer Berlin / Heidelberg.
- Kostić, D.; Rodriguez, A.; Albrecht, J. & Vahdat, A. (2003). Bullet: high bandwidth data dissemination using an overlay mesh. *ACM SIGOPS Operating Systems Review*, 37(5):282--297.
- Kulbak, Y. & Bickson, D. (2005). The eMule Protocol Specification. *eMule project*, <http://sourceforge.net>.
- Kumar, R.; Novak, J. & Tomkins, A. (2006). Structure and evolution of online social networks. In *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 611--617, New York, NY, USA. ACM.
- Lei, J.; Shi, L. & Fu, X. (2009). An experimental analysis of joost peer-to-peer vod service. *Peer-to-Peer Networking and Applications*, pp. 1--12.
- Liao, X.; Jin, H.; Liu, Y.; Ni, L. & Deng, D. (2006). Anysee: Peer-to-peer live streaming. In *IEEE INFOCOM*, volume 6. Citeseer.
- Limelight (2010). Limelight networks. <http://www.limelightnetworks.com/>.
- Liu, Z.; Wu, C.; Li, B. & Zhao, S. (2009). Distilling superior peers in large-scale P2P streaming systems. *Proc. of IEEE INFOCOM, Rio de Janeiro, Brazil*.
- Lua, E.; Crowcroft, J.; Pias, M.; Sharma, R. & Lim, S. (2005). A survey and comparison of peer-to-peer overlay network schemes. *IEEE Communications Surveys and Tutorials*, 7(2):72--93.
- Lv, Q.; Cao, P.; Cohen, E.; Li, K. & Shenker, S. (2002). Search and replication in unstructured peer-to-peer networks. In *Proceedings of the 16th international conference on Supercomputing*, pp. 84--95. ACM New York, NY, USA.
- Magharei, N. & Rejaie, R. (2006). Understanding mesh-based peer-to-peer streaming. In *NOSSDAV '06: Proceedings of the 2006 international workshop on Network and operating systems support for digital audio and video*, pp. 1--6, New York, NY, USA. ACM.

- Magharei, N.; Rejaie, R. & Guo, Y. (2007). Mesh or multiple-tree: A comparative study of live p2p streaming approaches. In *IEEE INFOCOM 2007. 26th IEEE International Conference on Computer Communications*, pp. 1424--1432.
- Maymounkov, P. & Mazieres, D. (2002). Kademlia: A peer-to-peer information system based on the xor metric. *Proceedings of IPTPS02, Cambridge, USA*, 1:2--2.
- Newman, M. (2003). The structure and function of complex networks. *Arxiv preprint cond-mat/0303516*.
- Oliveira, J.; Gomes, P.; Borges, A. & Campos, S. (2010). Centralidade em redes p2p de transmissão ao vivo. In *WP2P 2010*.
- Oliveira, J.; Vieira, A. & Campos, S. (2009). Poluição de conteúdo em sistemas p2p live streaming. *Simpósio Brasileiro de Sistemas Multimídia e Web - Webmedia*.
- Pearson, K. (1905). The problem of the random walk. *Nature*, 72(1865):294.
- Picconi, F. & Massoulié, L. (2008). Is there a future for mesh-based live video streaming? *Peer-to-Peer Computing, 2008. P2P '08. Eighth International Conference on*, pp. 289--298.
- Planet Lab (2010). Planet lab. <http://www.planet-lab.org/>.
- Pourebrahimi, B.; Bertels, K. & Vassiliadis, S. (2005). A Survey of Peer-to-Peer Networks. *Computer*.
- PPLive (2010). PPLive. <http://www.pplive.com/>.
- PPStream (2010). PPStream. <http://www.ppstream.com/>.
- Ratnasamy, S.; Francis, P.; Handley, M.; Karp, R. & Schenker, S. (2001). A scalable content-addressable network. In *Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications*, p. 172. ACM.
- Ripeanu, M. (2001). Peer-to-Peer Architecture Case Study: Gnutella Network. In *Proceedings of International Conference on Peer-to-peer Computing*, volume 101. Sweden: IEEE Computer Press.
- Rowstron, A. & Druschel, P. (2001). Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In *IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, volume 11, pp. 329--350. Citeseer.

- Sabidussi, G. (1966). The centrality index of a graph. *Psychometrika*, 31(4):581--603.
- Santos, R.; Rocha, B.; Rezende, R. & Loureiro, A. (2009). Characterizing the YouTube video-sharing community. <http://security1.win.tue.nl/~bpontes/pdf/yt.pdf>, 12.
- Sentinelli, A.; Marfia, G.; Gerla, M.; Tewari, S. & Kleinrock, L. (2007). Will IPTV Ride the Peer-to-Peer Stream? *IEEE COMMUNICATIONS MAGAZINE*, 45(6):86.
- Shirky, C. (2001). Listening to Napster. *Peer-to-Peer: Harnessing the Benefits of a Disruptive Technology*, pp. 19--28.
- Silva, T. H. (2009). Transmissões de vídeo ao vivo geradas por usuários: Caracterização e análise. Master's thesis, Universidade Federal de Minas Gerais.
- Silverston, T. & Fourmaux, O. (2007). Measuring P2P IPTV Systems. In *Proc. of ACM NOSSDAV*.
- SopCast (2010). Sopcast. <http://www.sopcast.org/>.
- Stoica, I.; Morris, R.; Karger, D.; Kaashoek, M. & Balakrishnan, H. (2001). Chord: A scalable peer-to-peer lookup service for internet applications. In *Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications*, p. 160. ACM.
- Strauss, F. & Schmidt, S. (2003). P2P CHAT-a peer-to-peer chat protocol. Technical report, Internet draft, Internet Engineering Task Force, June 2003, Work in progress.
- Tran, D.; Hua, K. & Do, T. (2004). A peer-to-peer architecture for media streaming. *Selected Areas in Communications, IEEE Journal on*, 22(1):121--133.
- TVAnts (2010). Tvants. <http://tvants.en.softonic.com/>.
- UUSee Inc. (2010). Uusee inc. <http://www.uusee.com/>.
- Varga, A. et al. (2001). The OMNeT++ discrete event simulation system. In *Proceedings of the European Simulation Multiconference (ESM)*, pp. 319--324.
- Verizon (2010). Verizon. <http://www22.verizon.com/>.
- Voulgaris, S.; Jelasity, M. & van Steen, M. (2003). A Robust and Scalable Peer-to-Peer Gossiping Protocol. In *The Second International Workshop on Agents and Peer-to-Peer Computing (AP2PC)*. Springer.

- Watts, D. & Strogatz, S. (1998). Collective dynamics of small-world networks. *Nature*, 393(6684):440--442.
- Wireshark (2010). Wireshark. <http://www.wireshark.org/>.
- Wu, C.; Li, B. & Zhao, S. (2007). Characterizing Peer-to-Peer Streaming Flows. *Selected Areas in Communications, IEEE Journal on*, 25(9):1612--1626.
- Wu, C.; Li, B. & Zhao, S. (2008). Exploring large-scale peer-to-peer live streaming topologies. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, 4(3):1--23.
- Xu, D.; Kulkarni, S.; Rosenberg, C. & Chai, H. (2006). Analysis of a CDN-P2P hybrid architecture for cost-effective streaming media distribution. *Multimedia Systems*, 11(4):383--399.
- Yahoo! Live (2010). Yahoo! live. <http://live.yahoo.com/>.
- Yang, B. & Garcia-Molina, H. (2003). Designing a super-peer network. In *Proceedings of the International Conference on Data Engineering*, pp. 49--62. Citeseer.
- Yang, S.; Jin, H.; Li, B.; Liao, X.; Yao, H. & Tu, X. (2008). The Content Pollution in Peer-to-Peer Live Streaming Systems: Analysis and Implications. In *Parallel Processing, 2008. ICPP'08. 37th International Conference on*, pp. 652--659.
- Yen, Y.-W. (2008). Techland - youtube looks for the money clip. <http://techland.blogs.fortune.cnn.com/2008/03/25/youtube-looks-for-the-money-clip/>.
- YouTube (2010). Youtube. <http://www.youtube.com/>.
- Zhang, X.; Liu, J.; Li, B. & Yum, T. (2005). CoolStreaming/DONet: A Data-Driven Overlay Network for Efficient Live Media Streaming. *Proceedings of IEEE INFOCOM*, 3:13--17.
- Zhao, B.; Huang, L.; Stribling, J.; Rhea, S.; Joseph, A. & Kubiawicz, J. (2004). Tapestry: A resilient global-scale overlay for service deployment. *IEEE Journal on selected areas in communications*, 22(1):41--53.
- Zhou, Y.; Chiu, D. & Lui, J. (2007). A Simple Model for Analyzing P2P Streaming Protocols. In *Network Protocols, 2007. ICNP 2007. IEEE International Conference on*, pp. 226--235.

