PREDIÇÃO DO NÍVEL DE COOPERAÇÃO EM SISTEMAS PAR-A-PAR DE VÍDEO AO VIVO A PARTIR DE MÉTRICAS DE CENTRALIDADE

GLAUBER DIAS GONÇALVES

PREDIÇÃO DO NÍVEL DE COOPERAÇÃO EM SISTEMAS PAR-A-PAR DE VÍDEO AO VIVO A PARTIR DE MÉTRICAS DE CENTRALIDADE

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.

Orientadora: Jussara Marques de Almeida
Coorientador: Alex Borges Vieira

Belo Horizonte

Julho de 2012

Gonçalves, Glauber Dias.

G635p Predição do Nível de Cooperação em Sistemas Par-a-Par de Vídeo ao Vivo a partir de Métricas de Centralidade. / Glauber Dias Gonçalves. — Belo Horizonte, 2012.
xiv, 71 f. : il. ; 29cm

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

Orientadora: Jussara Marques de Almeida.
Coorientador: Alex Borges Vieira.

1. Computação - Teses. 2. Videodigital – Teses.
3. Sistemas de transmissão de dados – Teses.
I. Orientador. II. Coorientador. III. Título.

CDU 519.6*84(043)

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

FOLHA DE APROVAÇÃO

Predição do nível de cooperação em sistemas par-a-par de vídeo ao vivo a partir de métricas de centralidade

GLAUBER DIAS GONÇALVES

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

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

PROF. ALEX BORGES VIEIRA - Co-orientador
Departamento de Ciência da Computação - UFJF

PROF. ANA PAULA COUTO DA SILVA
Departamento de Ciência da Computação - UFJF

PROF. ÍTALO FERNANDO SCOTÁ CUNHA
Departamento de Ciência da Computação - UFMG

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

Belo Horizonte, 13 de julho de 2012.

A DEUS, pela fé e renovação das minhas forças a cada dia, e à minha noiva Gabriela, pelo amor, carinho e incentivo, sabendo administrar com firmeza os muitos momentos difíceis durante a realização deste trabalho, dedico.

Agradecimentos

Aos meus pais Hilton e Jucéia, pela presença diária em minha vida, mesmo estando distantes.

Aos meus irmãos Glaucio e Glaucia, pelo apoio e incentivo.

Aos meus familiares em Belo Horizonte, pelo carinho e acolhimento.

A Anna Guimarães ("minha aluna de iniciação científica"), pela ajuda e paciência em boa parte desse mestrado.

Aos meus colegas de laboratório (VoD) João, Flávio, Henrique, Kênia, Éder e Fabiano, pelo auxílio e bons momentos compartilhados.

À professora Dra. Jussara Almeida, pela orientação, pela compreensão durante estes dois anos e meio de convivência e pelos valiosos ensinamentos passados.

Ao professor Dr. Alex Borges, pela coorientação e incentivo nos momentos difíceis.

Ao professor Dr. Ítalo Cunha, pelo valioso auxílio técnico e conselhos.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), pela concessão da bolsa.

"A tarefa não é tanto ver aquilo que ninguém viu,
mas pensar o que ninguém ainda pensou sobre aquilo que todo mundo vê."
(Arthur Schopenhauer)

Resumo

A arquitetura P2P vem sendo utilizada com sucesso para diminuir os custos e aumentar a escalabilidade dos sistemas de distribuição de vídeo ao vivo pela Internet. Nesses sistemas, os participantes em uma transmissão ao vivo (pares) trocam segmentos do vídeo entre si e cooperam para a distribuição do conteúdo. Alguns desses sistemas adotam mecanismos de incentivo à cooperação dos pares, provendo uma qualidade de serviço diferenciada aos pares mais cooperativos. Para medir o nível de cooperação dos pares, esses sistemas utilizam apenas as taxas de *upload* e *download* coletadas periodicamente. Contudo, tais medidas podem ser susceptíveis a falsificação por parte de pares maliciosos. Esta dissertação investiga meios alternativos para prever o nível de cooperação dos pares sem confiar especificamente nas taxas de *upload* e *download*. Em particular, é investigado o potencial de utilizar propriedades topológicas da rede sobreposta para predizer o nível de cooperação dos pares com precisão razoável. As métricas de centralidade grau, *betweenness* e proximidade foram utilizadas para a predição, pois elas indicam a importância relativa de um par na rede, considerando a sua posição na topologia ou número de parcerias. Foram realizados experimentos para coletar dados de uma das mais populares aplicações P2P de vídeo ao vivo, SopCast, usando um grande número de máquinas do PlanetLab. A partir desses dados, foi mostrado que as métricas de centralidade de um par têm uma correlação alta com o seu nível de cooperação, medido pela razão da taxa de *upload* por *download*, durante uma janela de tempo preestabelecida. A métrica grau de saída, especificamente, foi a que mostrou maior correlação. Além disso, essa métrica se mantém razoavelmente estável ao longo de janelas de tempo consecutivas. Tendo essas informações como motivação, foram desenvolvidos modelos baseados em regressão para predizer o nível de cooperação de um par na janela de tempo seguinte, dado o seu grau de saída coletado na última janela de tempo. Os dados coletados foram utilizados para avaliar os modelos e um modelo polinomial de grau quatro foi o mais preciso. As propriedades topológicas da rede sobreposta ainda foram exploradas para a detecção de pares maliciosos que agem em conluio para aumentar a sua centralidade.

Abstract

The P2P architecture has been used successfully to reduce costs and increase the scalability of Internet live streaming systems. In a P2P live transmission, users (peers) exchange video chunks among themselves and cooperate with the system to distribute the media content. Some P2P streaming systems adopt incentive mechanisms, providing a differentiated quality of service for more cooperative peers. To measure the level of cooperation of peers, these systems typically use only the upload and download rates, collected periodically from peers. However, such measures may be susceptible to malicious peers that lie about their cooperation. In this work, we investigate alternative methods to predict the level of cooperation of peers without relying specifically on their upload and download rates. In particular, we assess the potential benefit of exploiting topological properties of the P2P overlay network to predict, with reasonable accuracy, the level of cooperation of peers. To that end, we use the centrality metrics degree, betweenness and closeness, as they capture the relative importance of a peer in the network, considering its position in the topology or number of partnerships. Our study relies on data collected from one of the currently most popular P2P live applications, i.e., SopCast, using a large number of Planetlab machines. We first show that there is a high correlation between the centrality of a peer and its level of cooperation, which is defined as the ratio of the total upload to the total download traffics the peer exchanged with its partners. Specifically, the out-degree metric has the highest correlation. Furthermore, this metric remains reasonably stable over consecutive time windows. Motivated by these findings, we then develop regression-based models to predict the level of cooperation of a peer in a time window given its out-degree collected in the previous window. Using our collected data, we evaluate the models, finding that a fourth degree polynomial model is the most accurate one. We also exploit topological properties of overlay network to detect malicious peers that collude to increase their centrality.

Lista de Figuras

Lista de Tabelas

Sumário

Capítulo 1

Introdução

Neste capítulo serão abordados brevemente sistemas de distribuição de vídeo na Internet. A seguir, são apresentados as principais motivações e objetivos desse trabalho, assim como as contribuições obtidas.

1.1 Sistemas de Distribuição de Vídeo

Distribuição de vídeo é atualmente um dos serviços mais populares na Internet. Relatórios recentes mostram que 44% de todo tráfego transmitido em 2010 foi de vídeo e as projeções para 2015 é que essa proporção aumente para 58% [Cisco, 2010]. Logo, vídeo pode ser o conteúdo mais consumido na rede mundial de computadores nos próximos anos. Essa grande demanda impõe desafios para que a distribuição desse conteúdo possa ser feita eficientemente considerando os custos, a escalabilidade e a qualidade de serviço.

A distribuição de vídeo na Internet pode ser categorizada em vídeo sob demanda ou vídeo ao vivo. Na distribuição sob demanda, o conteúdo é armazenado no servidor e os usuários podem assiti-lo a qualquer momento, como ocorre por exemplo nos sistemas Youtube e Vimeo[1]. Na distribuição ao vivo, o vídeo é distribuído em tempo real, e os usuários que estão assistindo devem estar sincronizados com servidor do vídeo, como ocorre por exemplo nos sistemas Justin.tv e SopCast[2]. Ambas as categorias requerem largura de banda alta para distribuir vídeo a muitos usuários simultaneamente o que aumenta os custos, podendo diminuir a escalabilidade desses sistemas. Essa demanda é particularmente alta na distribuição ao vivo devido às condições restritas de entrega de conteúdo em tempo real.

[1] `http://www.youtube.com`, `http://www.vimeo.com`
[2] `http://www.justin.tv`, `http://www.sopcast.com`

A arquitetura clássica de serviços de distribuição de vídeo, cliente-servidor, tem se mostrado não escalável o suficiente para atender a demanda desse serviço na Internet [Silverston et al., 2009; Ullah et al., 2011]. Isso porque, nessa arquitetura, cada cliente estabelece uma conexão exclusiva com o servidor de vídeo o que leva a um consumo excessivo de banda e aumento de custos lineares com o número de usuários simultâneos. Uma arquitetura alternativa é baseada em redes de distribuição de conteúdo (CDN)[Liu et al., 2008]. Nessas redes, a carga central é distribuída em múltiplos servidores instalados em localizações geográficas estratégicas na Internet. Assim, requisições dos usuários são redirecionadas para os servidores mais próximos, diminuindo o atraso na visualização do vídeo e a sobrecarga no servidor central. Todavia, a infra estrutura de uma CDN têm custos altos para instalação e manutenção.

A arquitetura Par-a-Par (P2P) vem sendo utilizada como uma alternativa para distribuir vídeo pela Internet com baixo custo e escalabilidade. Essa arquitetura é empregada tipicamente na distribuição ao vivo, porque o fato dos usuários estarem sincronizados na mesma parte do vídeo favorece o compartilhamento dos fragmentos, ou *chunks* do vídeo entre eles [Shen et al., 2011]. Assim, os sistemas P2P de vídeo ao vivo utilizam a capacidade de *upload* dos usuários (pares) para ajudar na distribuição do conteúdo. Quando esses pares acessam o sistema, eles estabelecem parcerias, organizando-se em uma rede virtual, sobreposta à rede física. Um par pode solicitar aos seus parceiros *chunks* da mídia, liberando o servidor central da responsabilidade e dos custos associados de atender todos os clientes. Alguns sistemas de distribuição de vídeo ao vivo que utilizam essa arquitetura, tais como SopCast e PPLive[3], são muito populares atualmente na Internet, contando com milhares de usuários registrados [Borges et al., 2012].

1.2 Motivação

O bom funcionamento de sistemas baseados na arquitetura P2P e a qualidade de serviço provido dependem do comportamento dos pares envolvidos na distribuição do conteúdo, o que pode tornar esses sistemas susceptíveis a problemas de segurança e cooperação. A segurança diz respeito à integridade do conteúdo. Por exemplo, Liang et al. [2005] destacam o problema de poluição de conteúdo onde pares maliciosos alteram ou forjam o conteúdo sendo compartilhado, tornando-o inútil para os demais pares da rede. A cooperação diz respeito ao comportamento conhecido como *free riding*, termo nomeado por Adar & Huberman [2000] para denominar pares que usufruem do serviço, mas não

[3]http://www.sopcast.com, http://www.pptv.com.

contribuem em um nível aceitável, ou seja, recebem conteúdo dos seus parceiros, mas não o reenviam para outros pares.

Os problemas acima mencionados já foram bem estudados em sistemas P2P de compartilhamento de arquivos [Walsh & Sirer, 2005; Costa & Almeida, 2007; Levin et al., 2008; Xia & Muppala, 2010]. Uma das formas de proteção contra poluição é a verificação da autenticidade dos *chunks* do arquivo em conjunto com sistemas de reputação para identificar e eliminar os pares que espalham chunks poluídos. Em sistemas P2P de distribuição de vídeo ao vivo a poluição de conteúdo pode ser tratada com técnicas similares àquelas utilizadas em sistemas P2P de arquivo, com as devidas adaptações para o ambiente de transmissão ao vivo [Borges et al., 2008; Wang et al., 2010].

O problema de cooperação entre os pares em sistemas P2P de compartilhamento de arquivos vem sendo tratado com a aplicação de contribuição bilateral, e.g. Tit-for-tat[Cohen, 2003], onde o par aumenta a sua probabilidade de receber dados dos parceiros à medida em que ele fornece dados. Contudo, Tit-for-Tat pode não ser eficiente em sistemas P2P de vídeo ao vivo, pois os *chunks* têm utilidade durante um intervalo curto de tempo. Isso implica em menos chances de troca de *chunks* diferentes entre os pares [Silverston et al., 2008]. Além disso as oportunidades de compartilhamento são diferentes para os pares: por exemplo, os pares próximos ao servidor de vídeo tem *chunks* mais novos e interessantes do que os pares mais distantes, cujos *chunks* são mais antigos e perdem a utilidade rapidamente [Piatek & Krishnamurthy, 2010].

Logo, a cooperação entre os pares em sistemas P2P de vídeo ao vivo vem sendo tratado por abordagens mais específicas. Algumas delas focam em detectar pares pouco-cooperativos [Guerraoui et al., 2010; Azzedin, 2010] e removê-los do sistema. Outras focam em explorar os pares mais cooperativos e oferecer algum benefício em troca da sua contribuição [Piatek et al., 2010; Chatzidrossos et al., 2010]. Em ambos os casos, esses métodos precisam estimar periodicamente o nível de cooperação dos pares, porque ela varia ao longo do tempo por motivos diferentes, tais como pares entrando e saindo da rede (*churn*) [Stutzbach & Rejaie, 2006] e falhas no protocolo que causam descoordenação entre os pares [Picconi & Massoulié, 2008; Liang et al., 2008].

Para estimar o nível de cooperação do par, os métodos citados acima utilizam métricas baseadas em medições das taxas de *upload* e *download* dos pares. Alguns deles ainda empregam mecanismos de segurança para se tornarem robustos a pares maliciosos que reportam medições falsas [Jin et al., 2006; Jin & Chan, 2010]. Por exemplo, no mecanismo Contracts [Piatek et al., 2010], os pares precisam apresentar recibos criptografados para comprovar o volume de *chunks* fornecido aos parceiros,

enquanto que no mecanismo LiFTing [Guerraoui et al., 2010], os pares monitoram os seus parceiros, verificando se eles repassam cada *chunk* recebido. Esses mecanismos de segurança aumentam a sobrecarga do sistema com processamento (codificação e decodificação de recibos) e comunicação (envio de mensagens aos parceiros indiretos).

Dado que estimar a cooperação do par apenas com medidas de *upload* e *download* acarreta em aumento de custos para sua verificação, é interessante propor alternativas para estimar a cooperação que não dependam exclusivamente dessas medidas. Uma motivação especial para isso é que além dessas medidas, sistemas P2P de vídeo coletam periodicamente outros dados sobre os pares como a qualidade de vídeo recebida e parcerias recentes [Wu et al., 2007; Li et al., 2008a]. Em particular, a coleta periódica das parcerias permite a reconstrução da rede sobreposta em um ponto centralizado da rede, por exemplo o *tracker* ou o servidor de logs. Isso torna possível explorar propriedades topológicas da rede P2P para obter mais informações sobre os pares.

Há uma série de trabalhos que caracterizaram propriedades topológicas da rede sobreposta em sistemas P2P [Stutzbach et al., 2008; Wu et al., 2008; Tang et al., 2009], utilizando métricas de redes complexas como centralidade, agrupamento e mundo pequeno (*small worlds*) [Newman, 2003]. Dentre esses, há alguns trabalhos que indicaram métricas promissoras para predizer o nível cooperação de um par. Por exemplo, Oliveira et al. [2010] e Gkorou et al. [2011] utilizam métricas de centralidade para identificar pares importantes em uma rede P2P, respectivamente, super pares e pares fazendo o papel de conectores da rede (*hubs*).

Esta dissertação tem o propósito de avançar no uso de propriedades topológicas da rede P2P para predizer o nível de cooperação dos pares. Essa informação é importante para os mecanismos de incentivo em sistemas P2P de vídeo ao vivo, pois ela serve para identificar tanto os pares muito cooperativos como os pares pouco cooperativos (*free riders*). Nesse contexto, as métricas de centralidades serão exploradas, pois elas informam a importância relativa de um par em uma rede [Freeman, 1979]. Dado essas métricas, esse trabalho busca responder a seguinte pergunta:

- As métricas de centralidade de um par podem ser exploradas para predizer com precisão razoável o seu nível de cooperação em sistemas P2P de distribuição de vídeo ao vivo?

1.3 Objetivos

Este trabalho tem por objetivo principal investigar o uso de propriedades da rede P2P sobreposta, especificamente métricas de centralidade dos pares, para predizer o nível de

cooperação dos pares durante a distribuição de vídeo ao vivo. Esse objetivo principal pode ser delineado nos seguintes objetivos específicos:

- Propor modelos para predizer o nível de cooperação de um par utilizando propriedades da rede P2P sobreposta. O nível de cooperação resulta da utilização da largura de banda disponível no par pelo sistema P2P de vídeo, e as propriedades topológicas focam em métricas de centralidade do par.
- Utilizar propriedades da rede P2P sobreposta buscando mitigar a ação pares maliciosos que venham a atacar o sistema fornecendo dados falsos ou agindo em conluio.

1.4 Contribuições

As principais contribuições dessa dissertação são discutidas a seguir. Algumas delas foram publicadas nos seguintes trabalhos [Gonçalves et al., 2011, 2012a,b].

- **Correlação de métricas de centralidade com o nível de cooperação do par**.

  Foi investigada a correlação entre a centralidade de um par na rede sobreposta e o seu nível de cooperação em janelas de tempo sucessivas durante transmissões de vídeo ao vivo em um sistema P2P. O nível de cooperação de um par é estimado pela razão entre o volume de bytes cedidos (*upload*) pelo volume de bytes recebidos (*download*) na troca de dados com os parceiros. As métricas de centralidade consideradas foram o grau, *betweenness* e proximidade [Freeman, 1979]. Essas métricas, juntamente com o nível de cooperação dos pares, foram obtidas por meio de experimentos com o sistema P2P de vídeo ao vivo SOPCast e computadores do PlanetLab[Chun et al., 2003], como é discutida na seção 3.2. O grau de saída foi a métrica mais correlacionada, apresentando coeficiente de correlação de Spearman [Kendall & Gibbons, 1975] maior que 0.8 em 90% das janelas de tempo analisadas. A proximidade de saída e *betweenness* foram, respectivamente, a segunda e a terceira métrica mais correlacionadas. A seção 4.1 traz as distribuições das correlações e uma discussão mais detalhada sobre essa contribuição.

- **Modelo de predição do nível de cooperação de um par usando métricas de centralidade**.

  Foram propostos vários modelos de regressão não lineares para estimar o nível de cooperação de um par na janela de tempo seguinte, dada sua centralidade na última janela. Para construir e avaliar esse modelo foi utilizada a métrica de centralidade mais correlacionada com a cooperação, ou seja, o grau de saída. Primeiramente foi mostrado que essa métrica se mantem razoavelmente estável ao longo de janelas de tempo consecutivas de 60 segundos. A seguir o modelo foi construído e avaliado utilizando os dados coletados nos experimentos com o sistema SOPCast. Um modelo polinomial de grau quatro foi o que obteve melhores resultados, pois apresentou erros absolutos menores ou similares aos demais modelos avaliados. O modelo desenvolvido produz predições razoáveis do nível de cooperação dos pares, como é mostrado na Seção 4.2.2.

- **Abordagem para identificar conluio de pares maliciosos usando uma métrica de agrupamento**.

  A topologia da rede sobreposta foi ainda explorada com o objetivo de detectar suspeitas de pares agindo em conluio na rede. Nesse caso específico, um conluio ocorre quando um grupo de pares agem coordenadamente para formar parcerias falsas e parecerem mais cooperativos do que eles realmente são. A métrica de agrupamento condutância [Leskovec et al., 2008] foi utilizada com esse propósito e para avaliar a sua utilização foram analisados cenários de conluio. Essa métrica se mostrou útil para distinguir os pares maliciosos dos pares cooperativos legítimos, em alguns cenários, como, pares maliciosos utilizando identidades sintéticas para aumentar o seu grau de saída, ou ainda, utilizando algumas parcerias legítimas para dissimular a maioria de parcerias falsas.

1.5 Organização da Dissertação

O restante desta dissertação está organizado da seguinte forma. O Capítulo 2 apresenta uma contextualização sobre a arquitetura P2P para distribuição de vídeo, a cooperação entre os pares em sistemas P2P e métricas de redes complexas. O Capítulo 3 descreve a metodologia de coleta de dados no SopCast e analisa o nível de cooperação dos pares nesse sistema. O capítulo 4 apresenta o modelo de predição do nível de cooperação dos pares, e uma abordagem para identificar pares maliciosos em conluio. Por fim, o Capítulo 5 apresenta as conclusões e possíveis direções para trabalhos futuros.

Capítulo 2

Contextualização

2.1 Arquitetura P2P de Distribuição Vídeo ao Vivo

Os custos para distribuir vídeo ao vivo na Internet usando a arquitetura cliente/servidor clássica ou redes de distribuição de conteúdo (CDN) são altos [Shen et al., 2011]. Isso motivou a busca por uma arquitetura que distribui a carga do sistema, especialmente a banda de rede, entre os usuários assistindo o vídeo. Assim, a arquitetura P2P (*Peer-to-Peer* ou Par-a-Par) aparece como uma alternativa viável para a distribuição de vídeo ao vivo, já que o seu propósito fundamental é fazer com que os recursos computacionais sejam compartilhados diretamente entre os usuários (pares), com o mínimo de suporte ou intermédio de um servidor central [Ullah et al., 2011].

Na arquitetura P2P, os pares são organizados em uma rede lógica sobre as conexões físicas, comumente chamada de rede sobreposta. Nessa rede, há um participante especial que produz o video ao vivo, o servidor de vídeo. Ele particiona o vídeo em pedaços, chamados *chunks*, e os distribui entre os pares para exibição. À medida em que os pares obtêm os *chunks* e visualizam o vídeo, eles são habilitados a compartilharem seus *chunks* com os demais pares do sistema. Isso contribui de modo significativo para aliviar a carga do servidor, assim como da rede em geral. Esse modelo de distribuição pode utilizar tipos diferentes de rede sobreposta: árvore ou malha.

Os sistemas baseados em árvore formam uma rede sobreposta bem estruturada, com o servidor de vídeo na raiz dessa árvore, como é ilustrado na figura 2.1. Cada par recebe *chunks* de um par pai e os retransmite aos seus pares filhos automaticamente. Essa arquitetura é conhecida como *tree-push* [Liu et al., 2008]. Dessa forma, os dados são transmitidos em uma única direção, o que contribui para diminuir atrasos na reprodução do vídeo. Porém, uma das preocupações que devem ser levadas em consideração na construção e manutenção dessa árvore é a sua altura. É necessário

alocar mais pares em largura do que em altura para que os pares nos últimos níveis não tenham muito atraso na recepção dos dados, comparado aos pares que estão nos primeiros níveis. Exemplos de sistema que adotam a arquitetura *tree-push* são ESM [Chu et al., 2000], Zigzag [Tran et al., 2004] e mais recentemente TURINstream [Magnetto et al., 2010].

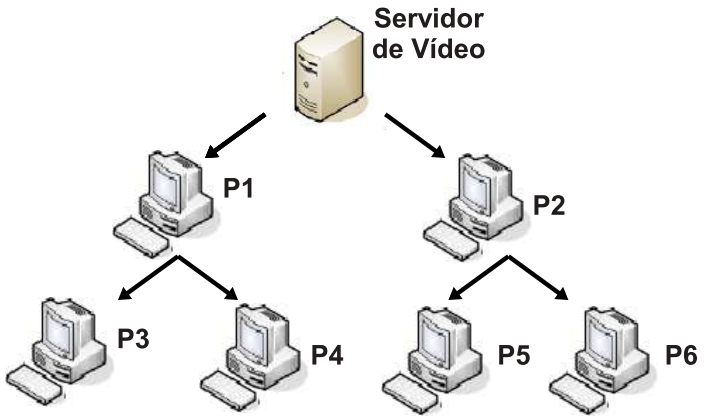


Figura 2.1: Arquitetura P2P baseada em árvore

Um problema da arquitetura baseada em árvore é que ela não utiliza a largura de banda dos pares em sua base (nodos folhas), uma vez que eles não contribuem para o sistema efetivamente fazendo *upload*. Uma proposta para lidar com esse problema é utilizar múltiplas árvores. Com essa arquitetura o servidor divide a transmissão do vídeo em subfluxos e os transmite em árvores diferentes. Assim, um par se conecta em mais de uma árvore para assistir o vídeo, como é mostrado na figura 2.2. Assim, os nodos folhas em uma árvore podem se tornar nodos internos em outra, fazendo melhor uso da largura de banda disponível de todos os pares. Exemplos clássicos de sistemas que utilizam múltiplas árvores para distribuição de vídeo ao vivo são CoopNet [Padmanabhan et al., 2003] e Splitstream [Castro et al., 2003].

Uma vez que os pares podem entrar ou sair da rede a qualquer momento, comportamento dinâmico também conhecido como *churn* [Stutzbach & Rejaie, 2006], manter a estrutura da árvore torna-se uma tarefa não trivial. Quando um par sai da rede, ele causa uma interrupção na transmissão dos seus pares filhos até que a árvore seja reconstruída. As técnicas de reconstrução disponíveis ainda não tornam os sistemas citados acima robustos o suficiente a *churn* [Magharei et al., 2007], porque o vídeo pode ter perdas consideráveis de qualidade com as frequentes reconstruções da árvore [Liu et al., 2008].

A estrutura baseada em malha aparece como uma solução para tornar o sistema mais robusto a *churn* [Hei et al., 2008], pois nessa estrutura não existe uma hierarquia

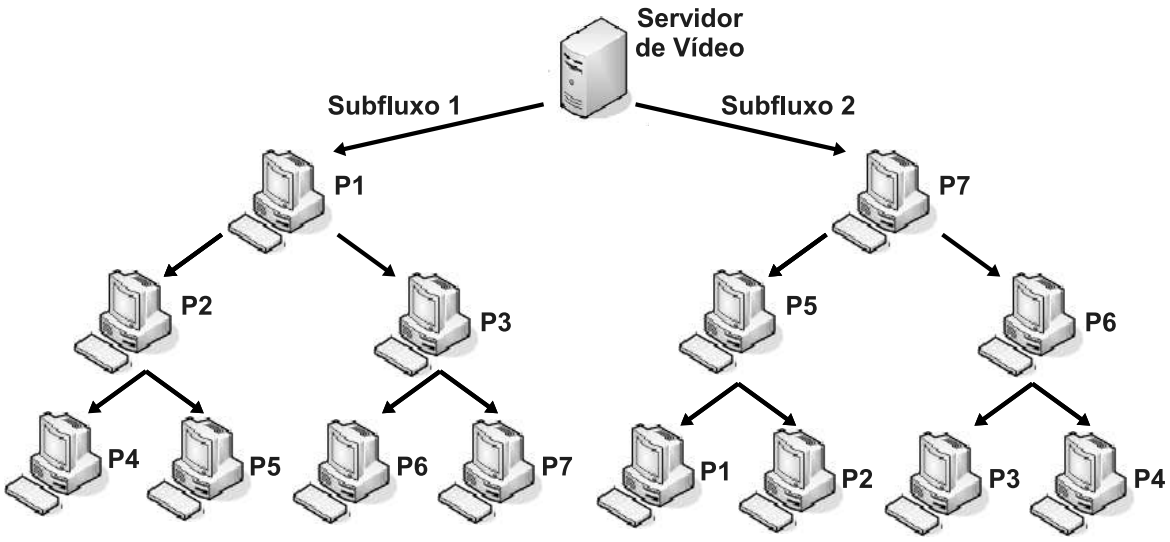


Figura 2.2: Arquitetura P2P baseada em múltiplas árvore com dois subfluxos

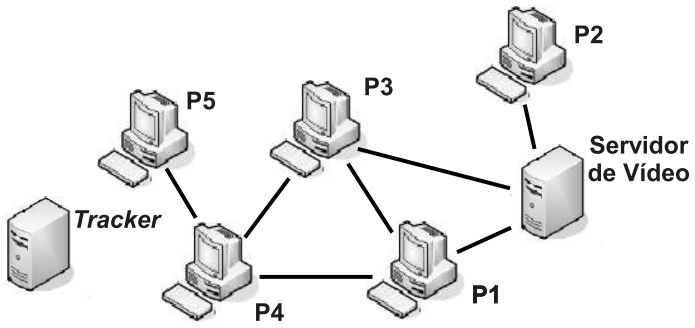


Figura 2.3: Arquitetura P2P baseada em malha

clara entre os pares da rede. Um par é servido por múltiplos pares, como é ilustrado na figura 2.3. Assim, se um de seus parceiros sai do sistema abruptamente, esse par tem outras fontes.

A formação da rede P2P baseada em malha tipicamente ocorre de modo a suportar o dinamismo dos pares. Quando um novo par ingressa no sistema, ele recebe uma lista com um conjunto de pares ativos do *tracker* da rede P2P. Em seguida, o par tenta interagir com os pares dessa lista, estabelecendo parcerias para troca de *chunks*. Dessa forma, quando um par deixa a rede, seus parceiros continuam recebendo *chunks* de outros vizinhos, evitando interrupções na transmissão do vídeo. Contudo, na estrutura de malha, os *chunks* podem chegar fora de ordem no par dado que eles são recebidos via diferentes caminhos. Logo, é necessário reordená-los e armazená-los em uma porção de memória (*buffer*) antes de exibi-los. Os pares mantém mapas de *chunks* descrevendo os *chunks* que eles têm armazenados e os que eles ainda precisam

e trocam esses mapas entre si de modo que os pares sabem a quem requisitar os *chunks* faltantes.

Na rede P2P com estrutura de malha, os *chunks* são requisitados entre os pares tipicamente usando requisição explícita de dados, arquitetura também conhecida como *mesh-pull*. Com essa arquitetura o par solicita *chunks* a determinados parceiros orientado pelos mapas de *chunks* recebidos. O *chunk* a ser requisitado pode ser escolhido por diferentes critérios, por exemplo, pode ser adotada a escolha de *chunks* raros, ou seja, disponíveis em poucos parceiros, ou a escolha de *chunks* na ordem sequencial de utilização. A escolha de *chunks* mais raros é comumente adotada no sistema BitTorrent para ajudar a replicar esses *chunks* na rede. Porém dadas as restrições de tempo real na transmissão ao vivo, os *chunks* tendem a ser requisitados na ordem de utilização na maioria dos sistemas P2P de vídeo ao vivo práticos existentes [Hei et al., 2007].

Arquiteturas híbridas, também conhecidas como *push-pull*, que combinam as vantagens das estruturas de árvore e de malha vêm sendo propostas. Os sistemas híbridos comumente usam uma rede baseada em malha para se tornarem resistentes ao *churn* e encaminhamento automático de um grande volume de *chunks* para diminuir o atraso do vídeo. Nesse caso, requisições explícitas e mapas de *chunks* continuam sendo utilizados para um par requisitar a um determinado parceiro o início do encaminhamento automático, assim como o fim desse quando o *buffer* de vídeo for preenchido. Exemplos de sistemas que adotam essa arquitetura são GridMedia [Zhang et al., 2005, 2007] e o novo Coolstreaming [Li et al., 2008a].

Embora existam abordagens híbridas como mostrado acima, a arquitetura *mesh-pull* é adotada na maioria dos sistemas comerciais existentes [Hei et al., 2008]. Alguns desses sistemas como PPLive, UUSee e SopCast[1] são populares na Internet e também vem despertando a atenção dos pesquisadores [Sentinelli et al., 2007; Hei et al., 2007; Wu et al., 2007; Tang et al., 2009]. As pesquisas sobre esses sistemas visam principalmente entender seu funcionamento e avaliar seu desempenho para propor arquiteturas P2P de distribuição de vídeo ao vivo mais eficientes. Nesta dissertação há o interesse em estudar especialmente o sistema SopCast, porque além de ser muito popular, ele disponibiliza um cliente para sistemas operacional Linux, facilitando assim experimentação em larga escala na plataforma de testes Planetlab [Chun et al., 2003]. A Seção 3.1 apresenta os detalhes sobre o funcionamento do sistema SopCast, os experimentos realizados e as informações obtidas a partir desses.

[1] `http://www.sopcast.com`, `http://www.synacast.com`, e `http://www.uusee.com`. SopCast, PPLive e UUsee, respectivamente.

2.2 Cooperação em Sistemas P2P

Esta seção apresenta alguns estudos sobre a cooperação de pares em sistemas P2P e sobre mecanismos de incentivo à cooperação. A seção 2.2.1 foca nos sistemas P2P de compartilhamento de arquivo, onde o referido tema já foi amplamente estudado em aplicações populares como BitTorrent[2]. Em seguida, na seção 2.2.2 são discutidos trabalhos mais recentes que tratam de cooperação nos sistemas P2P de distribuição de vídeo ao vivo.

2.2.1 Sistemas P2P de Compartilhamento de Arquivos

A cooperação entre os pares foi originalmente analisada no sistema de compartilhamento de arquivos Gnutella. Adar & Huberman [2000] rastrearam esse sistema por um período de 24 horas e observaram que 70% dos usuários não compartilhavam arquivos e que apenas 25% dos usuários atendiam 99% de todas as consultas no sistema. A partir desse trabalho o problema de baixa cooperação em sistemas P2P ganhou notoriedade, tendo sido definido o termo *free riding*, para indicar o padrão de comportamento em que os pares usufruem do serviço P2P, mas não contribuem para a rede ou para os outros pares em um nível aceitável [Karakaya et al., 2009].

Desde então, surgiram soluções variadas para mitigar o *free riding* em sistemas P2P. Dentre elas, o mecanismo de incentivo *"tit-for-tat"* (TFT) do protocolo BitTorrent [Cohen, 2003] é uma das soluções mais conhecidas e avaliadas [Sirivianos et al., 2007; Konrath et al., 2007; Liu et al., 2010]. O mecanismo TFT incentiva a cooperação bilateral (reciprocidade) entre os pares. Seu funcionamento consiste basicamente em cada participante avaliar, em um determinado intervalo de tempo, os parceiros que lhe forneceram mais dados e retribuí-los, no intervalo de tempo seguinte, atendendo às suas requisições. Especificamente, TFT implementa um algoritmo para medir a contribuição dos parceiros em intervalos de tempo a fim de determinar para quais parceiros um dado par deve fazer *upload* ("*unchoked*") e para quais ele deve suprimir o *upload* ("*choked*"). No intervalo seguinte o algoritmo também determina mais um parceiro para *upload* desconsiderando a sua contribuição ("*optimistic unchoked*"), com o intuito de encontrar novos parceiros com taxas de *upload* e *download* melhores que os atuais. Embora TFT seja um mecanismo simples e prático para desencorajar *free riding* em sistemas P2P, ele possui algumas vulnerabilidades que foram analisadas nos trabalhos a seguir.

[2]http://www.bittorrent.com

Em [Sirivianos et al., 2007] foi proposto um experimento para caracterizar o comportamento de *free-riders* e mensurar a degradação de desempenho que eles causam em sistemas P2P de compartilhamento de arquivos. Os autores modificaram um cliente do protocolo BitTorrent de modo que ele tivesse uma visão privilegiada do sistema, podendo assim explorar o *optimistic unchoke* do mecanismo TFT e se conectar a todos os pares ao seu alcance. A seguir, eles inseriram um número determinado de clientes modificados para atuarem como *free-riders*. Foram realizados experimentos em uma rede controlada no PlanetLab e em ambiente aberto da Internet. Assim, em ambos os casos foi observado que pares *free-riders* (os clientes modificados) tiveram taxas de *download* superior aos pares honestos e a taxa média de *download* no sistema sofreu um declínio significativo com o aumento desses pares privilegiados na rede.

Em [Konrath et al., 2007] também foi demonstrado por meio de simulações que o mecanismo TFT não é suficiente para manter a eficiência do BitTorrent com a presença de pares mentirosos. Nesse caso, foi explorado um ataque onde um bando de pares mentirosos anunciam *chunks* que eles não possuem para torná-los escassos no sistema. Um outro ataque explorado nesse trabalho foi o uso de identidades sintéticas (*sybils*). Essas identidades são pares que não realizam *download* nem *upload*, mas têm a função apenas de aumentar o número de atacantes na rede. Cada um desses ataques provocou atrasos em todos os *downloads* da rede e falhas quando metade dos pares na rede efetuava o ataque.

Os trabalhos de Konrath et al. [2007] e Sirivianos et al. [2007] mostraram vulnerabilidades no mecanismo TFT que podem ser exploradas por pares maliciosos, ou seja, participantes que estudam o funcionamento do sistema para realizar ataques. Independente dessas vulnerabilidades discutidas, existem aspectos para incentivar a cooperação entre os pares que não são considerados em TFT, como mostram os trabalhos a seguir.

Liu et al. [2010] argumentam que o mecanismo TFT foca no relacionamento entre pares que têm interesses em comum. Logo, pares que compartilham conteúdo que não são de grande interesse para os demais, não têm oportunidade de contribuir para o sistema. Para essa situação os autores propuseram um mecanismo de incentivo onde os pares formam uma rede social: quando o participante A fornece algum conteúdo para o participante B, ambos pertencentes à mesma rede social, A obtém crédito de B e dos amigos de B, ou seja, pares que relacionam com B. Meulpolder et al. [2009] observaram que TFT incentiva a cooperação entre os pares apenas enquanto eles estão fazendo o *download* de um arquivo. Então os autores propuseram o mecanismo de reputação chamado BarterCast com o objetivo de funcionar conjuntamente com TFT e prolongar o período de cooperação dos pares. Dessa forma, duas políticas de cooperação são

incluídas em TFT: o par com maior reputação recebe *upload* prioritariamente através do *optimistic unchoke* e os pares com reputação baixa (menor que um determinado limiar) não recebem nenhum tipo de *upload*.

O mecanismo BarterCast baseia-se em propriedades topológicas da rede P2P sobreposta, assim, o seu funcionamento será discutido na seção 2.3, juntamente a outros trabalhos que utilizam tais propriedades.

2.2.2 Sistemas P2P de Vídeo ao Vivo

Os sistemas P2P de distribuição de vídeo ao vivo mais populares como PPLive e SopCast não provêem incentivos aos pares que mais cooperam [Piatek & Krishnamurthy, 2010; Tang et al., 2009; Hei et al., 2007], bem como não imputam punições aos pares pouco cooperativos, como é analisado na seção 3.3. Esses sistemas confiam no altruísmo de pares que disponibilizam parte de sua banda de *upload* para a rede P2P, sem obter garantias de retorno em qualidade de vídeo. Tal fato pode se tornar um problema pois havendo uma redução em larga escala na cooperação dos pares o sistema pode entrar em colapso. Existem propostas na literatura de mecanismos que buscam mitigar esse problema. Discute-se a seguir os aspectos mais importantes desses mecanismos.

FlightPath [Li et al., 2008b] é uma proposta de sistema P2P de vídeo ao vivo que impõem restrições aos pares tais como troca balanceada de dados entre parceiros, número pequeno de parceiros (2 a cada ciclo de 2 segundos), fornecimento limitado de *chunks* a cada parceiro por ciclo e limitações na seleção de parcerias. Em FlightPath as parcerias de um par estão restritas a um grupo pequeno de pares em cada ciclo. Essa restrição é para evitar principalmente parcerias estratégicas de pares maliciosos para atacar o sistema. Os n pares ativos no sistema são particionados em $log(n)$ grupos e a cada ciclo um par deve escolher um parceiro de um grupo. Esse grupo é determinado pelo número identificador do par e do ciclo corrente e os outros pares da rede verificam o grupo de onde um par escolheu um parceiro. FlightPath ainda introduz uma porcentagem de relaxamento (10%) nas trocas de dados para ajudar os pares incapazes de contribuir de forma totalmente balanceada.

As restrições acima utilizadas pelo mecanismo FlightPath oferecem condições para uma análise mais formal desse protocolo quanto às possibilidades de comportamento bizantino dos pares (*free riding* e ataques). Todavia, o protocolo pode ter pouca eficiência em redes onde a capacidade de *upload* ou *download* dos pares é heterogênia (cenários reais), como se argumenta a seguir.

Silverston et al. [2008] e Piatek & Krishnamurthy [2010] enfatizaram que

balanceamento na troca de dados não é apropriado para o cenário de distribuição ao vivo porque: (1) devido às restrições de tempo real, *chunks* tem utilidade em um intervalo de tempo curto, o que implica em menos oportunidades de troca de *chunks* diferentes entre os pares; (2) trocas de *chunks* estritamente balanceadas entre os pares podem deixar pares com taxa de *upload* abaixo da média da rede sem *chunks* suficientes para assistir o vídeo; (3) relaxar o balanceamento (FlightPath) ainda tem efeito similar ao balanceamento estrito, ambas as formas dependem da existência de *chunks* de mútuo interesse entre dois pares, o que coloca pares distantes da fonte de vídeo em desvantagem dado que seus *chunks* tendem a perder utilidade mais rapidamente; (4) a taxa de *download* é limitada pela produção de *chunks* no servidor de vídeo (taxa de vídeo), o que implica que pares com taxa de *upload* acima desse limiar não poderão ser recompensados com taxa de *download* similar, como ocorre em BitTorrent.

Considerando que o balanceamento na troca de dados não é efetivo em sistemas P2P de vídeo ao vivo, Silverston et al. [2008] propuseram um mecanismo para incentivar a cooperação dos pares que não depende apenas da distribuição de dados. Nesse mecanismo, um par impossibilitado de servir *chunks* aos parceiros indica pares capazes de lhes oferecerem novos *chunks*. As indicações podem ser feitas a partir dos mapas de *chunks* que um par recebe dos seus parceiros, indicando os *chunks* disponíveis e faltantes em cada parceiro. Logo, as informações contidas nesses mapas podem ser utilizadas para indicar potenciais parcerias. Assim, um par menos favorecido na rede, por exemplo, um par distante do servidor de vídeo, incentiva os seus parceiros a continuar lhe provendo dados em troca de informação sobre novas parcerias.

A proposta de Silverston et al. [2008] é interessante porque preserva a reciprocidade entre os pares na rede P2P, como proposto pelo mecanismo TFT, mas não há garantias de melhoria na qualidade de serviço para os pares mais cooperativos. Logo, a maioria das propostas para incentivar a cooperação em sistemas P2P de vídeo ao vivo se baseiam em medições diretas da cooperação do par. Isso tipicamente é feito coletando as taxas de *upload* e *download* dos pares. Assim, podem ser aplicadas penalizações para os pares que redistribuem poucos *chunks* (taxa de *upload* baixa) tais como remoção da rede ou menor prioridade de serviço, ou podem ser providos benefícios para os pares com volume de *upload* alto, tais como conexão direta com o servidor de vídeo.

Server Guaranted Cap (SGC) é um exemplo de mecanismo que provê incentivo aos pares mais cooperativos, proposto por Chatzidrossos et al. [2010]. Nesse mecanismo, uma fração da capacidade do servidor de vídeo é reservada para manter a qualidade de serviço dos pares altruístas, ou seja, aqueles que fornecem *chunks* para pares impossibilitados de retribuírem na mesma proporção fornecida. Os autores destacam

que tal mecanismo é importante para pares com capacidade de *upload* limitada, por exemplo usuários com dispositivos móveis. Uma das premissas desse mecanismo é que um par cooperativo requisita dados ao servidor de vídeo apenas quando os seus parceiros não podem atendê-lo. Logo, esse mecanismo não considera a existência de pares mentirosos, ou seja, pares que reportam dados falsos para estarem permanentemente conectados ao servidor de vídeo. Tais pares teriam como benefício uma qualidade de serviço diferenciada dos demais pares, por exemplo menor atraso. A seguir é apresentado um mecanismo que provê benefício semelhante aos pares cooperativos, porém propõe meios para verificar a cooperação.

O mecanismo de incentivo Contracts [Piatek et al., 2010] utiliza estimativas do nível de cooperação dos pares para reestruturar a rede sobreposta. Assim, pares cooperativos são trazidos para perto da fonte do vídeo, recebendo uma qualidade de serviço melhor, enquanto pares pouco cooperativos ficam mais distantes da fonte. A cooperação de um par é estimada pelo volume de *chunks* que ele fornece aos parceiros em um determinado instante de tempo e essa estimativa é incrementada por um fator de cooperação efetiva[3] caso esses parceiros sejam cooperativos. As estimativas de cooperação do par podem ser calculadas no *tracker* a partir de dados coletados dos pares periodicamente, ou pelos seus vizinhos a partir de dados espalhadas na rede pelos próprios pares (*flooding*). No primeiro modo, o *tracker* reestrutura a rede determinando novos parceiros aos pares; no segundo, os pares reestruturam a rede lentamente estimando a cooperação dos vizinhos para selecionar suas parcerias. Em ambos os casos, recibos criptografados são utilizados para verificar as informações coletadas, i.e. o volume de *chunks* fornecido pelos pares, isso leva ao aumento do custo computacional nos pares ou no *tracker* devido ao constante processamento de recibos[4].

Há outros mecanismos que focam em detectar os pares pouco cooperativos ou *free-riders* na distribuição de vídeo. Diferentemente de SGC e Contracts, esses mecanismos atuam de modo coercivo; baseando-se na premissa de que imputar punições é uma forma eficiente para desencorajar baixa cooperação para o sistema. Neste caso, esses mecanismos assumem que muitos pares pouco cooperativos têm recursos para contribuir para o sistema, mas deliberadamente não contribuem. Azzedin [2010] propôs a detecção de pares pouco cooperativos em sistemas multimídia em geral através de um mecanismo de reputação. Os pares espalham pela rede mensagens de recomendação dos parceiros cooperativos. Então alguns pares monitores do sistema

[3] A cooperação efetiva é uma métrica adicional utilizada pelo mecanismo Contracts para beneficiar um par que fornece *chunks* a pares muito cooperativos. Essa métrica corresponde a soma dos volumes individuais fornecidos a cada parceiro, onde cada volume é ponderado pela capacidade de *upload* desse parceiro.

[4] Os recibos usam criptografia de chave pública RSA com chaves de 1024 bits.

coletam essas mensagens e determinam se um par é *free-rider* comparando o número de recomendações recebidas em uma janela de tempo do passado com o presente. Contudo, as recomendações não são verificadas. Logo *free-riders* poderiam fornecer recomendações falsas deles próprios ou de seus parceiros para não serem detectados.

Guerraoui et al. [2010] apresentaram LiFTinG, um protocolo para identificar *free riders* e removê-los do sistema P2P de vídeo com verificação das contribuições. Em LiFTing, cada par avalia seus parceiros por meio de verificações diretas e cruzadas: o primeiro tipo de verificação foca na proporção de requisições de *chunks* atendidas pelo parceiro e o segundo tipo verifica se o parceiro fornece a outros pares os *chunks* que lhe foram fornecidos. O par reporta a falta de cooperação dos parceiros aos monitores da rede que têm a permissão para banir do sistema os pares que ultrapassam determinado escore de queixas. Embora o propósito de LiFTing seja não sobrecarregar o sistema com esquemas de autenticação, a verificação cruzada eleva o número de mensagens entre os pares. Por exemplo, uma rede onde cada par fornece *chunks* a pelo menos n parceiros, o custo de mensagens para realizar a verificação cruzada é $O(n^2)$, para cada par. A configuração ideal desse mecanismo seria fazer a verificação cruzada para cada *chunk* o que aumentaria muito a taxa de detecção de *free riders*. Entretanto, essa configuração leva a um custo ainda maior.

Dos mecanismos de incentivo para sistemas P2P de vídeo ao vivo estudados acima, apenas três deles tratam o problema de pares maliciosos agindo em conluio para beneficar uns aos outros com parcerias falsas entre si. São eles os sistema P2P de vídeo ao vivo FlightPath [Li et al., 2008b], os mecanismos de incentivo LiFTinG [Guerraoui et al., 2010] e Contracts [Piatek et al., 2010].

O sistema FlightPath restringe a escolha de parceiros de um par conforme a estratégia mostrada anteriormente, o que diminui efetivamente as possibilidades de conluio entre pares maliciosos. O mecanismo de incentivo LiFTing [Guerraoui et al., 2010] assume que os sistemas P2P de vídeo tipicamente adotam a política de seleção de parcerias aleatórias. Nessa política os pares recebem periodicamente do sistema um subconjunto de potenciais parceiros escolhidos de modo aleatório e uniforme. Logo, esse mecanismo analisa o histórico de parcerias dos pares para detectar parcerias tendenciosas. Assim, os pares que têm os mesmos parceiros com uma determinada frequência são suspeitos de conluio.

A desvantagem dos dois mecanismos acima é que eles limitam a aplicação de outras abordagens para seleção de parcerias nos sistemas P2P. Por exemplo, abordagens que exploram o potencial para priorizar parcerias entres pares no mesmo provedor de Internet (ISP), como é proposto por Shen & Zimmermann [2009] e Xie et al. [2008], ou abordagens que priorizam parcerias entre pares com o mesmo nível de cooperação,

como é proposto no mecanismo Contracts [Piatek et al., 2010]. As parcerias dentro do mesmo ISP contribuem para diminuir os custos dos provedores de Internet e também a distância na conexão física entre os pares, o que implica em menor atraso na comunicação. As parcerias entre pares com nível de cooperação similares contribui para melhorar a qualidade de serviço prestado aos pares mais cooperativos, servindo de incentivo à cooperação nos sistemas P2P.

O mecanismo Contracts [Piatek et al., 2010] analisa se o volume de *download* dos pares está de acordo à taxa de vídeo do canal. Isso porque os pares maliciosos que reportam volumes de *chunks* falsos entre si e assistem o vídeo com *chunks* providos por pares honestos, terão taxa de *download* superior à taxa de vídeo. Dessa forma, essa abordagem limita a formação de um conluio entre pares maliciosos que assistem ao vídeo e ainda reportam o recebimento de dados falsos entre si para inflacionar as suas taxas de *upload*. Por outro lado, ela não é totalmente eficaz caso seja utilizado identidades sintéticas (*sybils*) no conluio, ou seja, pares que não assistem ao vídeo verdadeiramente, mas apenas reportam recebimento de dados falsos para favorecer alguns pares maliciosos, como é discutido na Seção 4.4.

Enfim, os trabalhos discutidos nesta seção mostram que identificar o nível de cooperação dos pares é uma tarefa essencial para incentivar a cooperação nas redes P2P de vídeo. Contudo, os métodos existentes confiam apenas em medições do volume de *chunks* trocados entre os pares cuja verificação pode incorrer em sobrecarga para o sistema, por exemplo, um número excessivo de mensagens entre os pares como em LiFTinG [Guerraoui et al., 2010] ou processamento intensivo de dados como em Contracts [Piatek et al., 2010]. Esta dissertação investiga novos aspectos para estimar e predizer a cooperação dos pares explorando propriedades topológicas da rede P2P sobreposta. Ela investiga também como essas propriedades podem ser utilizadas para verificar a cooperação dos pares com menor custo e também identificar pares agindo em conluio.

As propriedades topológicas de redes vem sendo amplamente exploradas em diferentes tipos de sistemas que funcionam sobre a Internet [Easley & Kleinberg, 2010], como as redes sociais virtuais, emails, *hyperlinks* e também os sistemas P2P, embora os trabalhos estudados nesta seção não exploram tais propriedades. As redes modeladas nesses sistemas são comumente chamadas de *redes complexas* pelo fato de serem grandes (milhares de vértices e conexões entre eles) e apresentarem propriedades diferentes de modelos de redes teóricos. A próxima seção discute as propriedades de redes complexas exploradas nesta dissertação.

2.3 Métricas de Redes Complexas

Segundo Newman [2003], a pesquisa em redes está testemunhando uma nova tendência nos últimos anos, com seu foco se movendo do estudo de pequenos grafos ou de propriedades de vértices e arestas específicos para o estudo baseado na análise estatística e em larga escala do grafo como um todo. Essa mudança está sendo possível devido, principalmente, ao aumento do poder computacional, aprimoramento dos métodos estatísticos e ao crescimento das redes de informações, que permitem obter e analisar dados em uma escala cada vez maior. No entanto, mesmo com o auxílio dessas ferramentas, ainda é um desafio responder questões a cerca da estrutura de uma rede quando ela possui milhões ou bilhões de vértices. Nesse contexto, as métricas de redes complexas assumem um papel importante, pois elas possibilitam caracterizar a estrutura de diferentes redes, sejam elas de computadores, sociais ou biológicas, a fim de entender o comportamento dos sistemas que sobre elas funcionam.

Uma análise importante em redes complexas é a identificação dos vértices mais centrais ou mais influentes na rede. Para isso, métricas de centralidade [Freeman, 1979] podem ser utilizadas. Essas métricas expressam a importância relativa de um vértice na rede e grau, *betweenness* e proximidade são as principais métricas que capturam a centralidade desse vértice. Essas métricas são definidas a seguir, onde a rede é modelada como um grafo G não direcionado com N vértices.

(1) Grau de um vértice i é medido pelo número de vértices j adjacentes ou "vizinhos" no grafo, dado por:

$$G(i) = \sum_{j \in G, i \neq j} a(i,j)$$

onde $a(i,j) = 1$ se os vértices i e j estão conectados por uma aresta e $a(i,j) = 0$ caso contrário.

(2) Proximidade de um vértice i é medida pela distância relativa desse vértice para os demais no grafo, dado por:

$$P(i) = \frac{N}{\sum_{j \in G, i \neq j} d(i,j)}$$

onde $d(i,j)$ é a distância geodésica entre os vértices i e j, ou seja, o número de arestas entre esses vértices considerando o caminho mais curto.

(3) *Betweennness* de um vértice i é medido da seguinte forma. Para cada par de vértices jk, o número de caminhos mais curtos entre j e k que passam por i, representado $g_{jk}(i)$, é dividido pelo número total de caminhos mais curtos entre j e k,

representado por g_{jk}. Isso somado sobre todas as combinações de pares de vértices jk em V.

$$B(i) = \sum_{j,k \in G, i \neq j \neq k} \frac{g_{jk}(i)}{g_{jk}}$$

As métricas acima podem ter algumas modificações dependendo do contexto em que a rede é modelada. Por exemplo, uma rede também pode ser vista como um grafo com arestas direcionadas, logo haverá grau de saída e entrada, assim como proximidade de saída e entrada e *betweenness* direcionado. Adicionalmente, o grau e o *betweenness* podem ser expressos relativos ao número de vértices no grafo. Assim, quando houver alguma modificação dessas métricas ao longo dessa dissertação, uma nova definição formal da métrica será apresentada. A seguir são descritos alguns trabalhos que modelaram grafos a partir de sistemas reais e aplicaram algumas dessas métricas para obter informações relevantes sobre esses sistemas.

Chen et al. [2004] analisaram as propriedades topológicas do sistema P2P Gnutella[5] de compartilhamento de arquivos utilizando as três métricas de centralidade mostradas acima. Os autores reconstruíram a topologia coletando dados de clientes desse sistema na Internet com busca em largura por meio de um rastreador (*crawler*) próprio. Os autores observaram que *betweenness* cresce proporcionalmente com o grau do vértice. Contudo existem pares com *betweenness* alto e grau baixo, o que pode levar à perda de desempenho ou conectividade na rede caso esses pares deixem o sistema. Por outro lado, não há uma correlação forte entre grau e proximidade, indicando que pares com graus bastantes diferentes podem ter uma proximidade similar, ou seja, eles podem alcançar outros pares na rede com a mesma facilidade. Os autores também observaram que a distribuição do grau segue uma lei de potência similarmente a outras redes complexas como redes sociais, biológicas e a Internet [Boccaletti et al., 2006].

Contudo, Stutzbach et al. [2008] realizaram um novo estudo das propriedades topológicas do Gnutella e observaram que o grau dos pares não seguem uma distribuição de lei de potência. Isto porque há um pico em torno do grau 30, indicando que o cliente Gnutella tenta manter 30 vizinhos por par. Os autores ainda afirmaram que o uso de coletores lentos levam a dados inconsistentes sobre a topologia de sistemas P2P, como foi reportado em trabalhos anteriores. Então, utilizando um rastreador mais veloz que rastreadores de trabalhos anteriores, e.g. [Chen et al., 2004], foi obtido novas informações sobre a estrutura topológica do sistema Gnutella. Por exemplo, foi observado que pares mais antigos no sistema formam um componente conectado central com um grau de agrupamento alto, enquanto pares mais novos formam grupos de menor conectividade, mas ligados ao componente central. Com essa organização,

[5] `http://rfc-gnutella.sourceforge.net`

a rede é extremamente resiliente à saída de pares, pois, mesmo se 50% dos pares com maior grau forem removidos da rede, cerca de 75% dos pares restantes continuam conectados.

Gkorou et al. [2011] utilizaram a métrica de *betweenness* para melhorar o desempenho do mecanismo de reputação Bartercast, descrito na Seção 2.2.1. Nesse mecanismo cada par mantém localmente um grafo onde vértices representam os pares e as arestas são ponderadas pelo volume de dados trocados entre eles. Um par identifica a reputação dos outros pares calculando o fluxo máximo de dados a partir dele para os demais pares na rede e vice-versa (algoritmo de fluxo máximo). Contudo, os autores perceberam que as reputações são mais exatas se o par calcular o fluxo máximo a partir do vértice com maior *betweenness* ao invés dele próprio, isso porque valores de *betweenness* alto indicam pares com carga de comunicação alta. A dificuldade em implantar essa proposta no mecanismo Bartercast é que os grafos locais de um par têm uma visão parcial da rede ao passo que *betweenness* é uma propriedade global do par. Logo, a precisão do método é maior quanto mais completa for a visão da rede, então ele seria mais preciso em um ambiente com uma visão global da rede, ou seja, uma abordagem centralizada, como proposta nesta dissertação.

Em Onnela et al. [2007] os autores analisaram a rede de comunicação formada a partir das ligações de telefones celulares de aproximadamente 20% da população dos Estados Unidos. No grafo construído, os vértices correspondiam aos usuários dos celulares e uma aresta era inserida entre dois vértices se os usuários correspondentes realizaram chamadas reciprocamente no período de 18 semanas. Foi observada a existência de um único componente conectado que incluía cerca de 84% dos vértices da rede. Além disso, foram identificadas arestas com um *betweenness* alto ligando pequenos componentes conectados no grafo (*comunidades*). Os autores observaram que a remoção dessas arestas causavam o desaparecimento do componente conectado gigante, o que indica a importância dessas arestas[6]. Além disso, foi observado que o grau dos vértices seguem uma distribuição próxima de uma lei de potência, indicando que muitos usuários se comunicavam com poucos indivíduos ao passo que uma minoria se comunicava com dezenas.

Tang et al. [2009] estudaram algumas propriedades topológicas do sistema SopCast. Eles realizaram experimentos com computadores do PlanetLab conectados a um canal privado do SopCast e reconstruíram a topologia completa da rede P2P a partir de traços da comunicação entre os computadores. O estudo focou apenas no grau dos pares, mas foi considerado grau de entrada e de saída, ou seja, a quantidade

[6] *Betweenness* de arestas é calculado da mesma forma que de vértices. $B(e) = \sum_{j,k \in N, j \neq k} \frac{g_{jk}(e)}{g_{jk}}$, onde e representa uma aresta e j e k representam vértices.

de parceiros dos quais o par recebe e para os quais o par envia vídeo, respectivamente. Foi observado que o grau de entrada segue uma distribuição normal com média em torno de duas parcerias e a distribuição do grau de saída segue uma lei de potência, o que implica em poucos pares com grau de saída muito alto e uma maioria de pares com grau de saída baixo. Uma questão deixada em aberto nesse trabalho foi o tamanho adequado de uma janela de tempo para estudar a dinamicidade dos pares. Nos experimentos realizados não foi considerado *churn* e foram utilizados janelas de tempo muito curtas (2 segundos). Nessa dissertação realizamos experimentos com o sistema SopCast considerando *churn* e analisamos os efeitos da dinamicidade dos pares em janelas de tempo com tamanhos diferentes, conforme discutido na seção 3.3.

Wu et al. [2008] caracterizaram as propriedades topológicas do sistema P2P de vídeo ao vivo UUSee com traços da rede coletados nos servidores de *log* desse sistema, conseguindo assim reconstruir a topologia inteira da rede. A propriedade topológica mais explorada foi o grau dos pares. Assim como [Stutzbach et al., 2008], os autores observaram que a distribuição do grau não segue uma lei de potência, pois há picos em graus específicos de acordo com a hora do dia. Também foram analisadas as distribuições do grau de entrada e de saída separadamente. Observou-se que a distribuição do grau de saída segue aproximadamente uma lei de potência de dois segmentos no sistema UUSee, ao passo que a aproximação para uma lei de potência com segmento único foi mais precisa no sistema SopCast [Tang et al., 2009]. Outra análise interessante foi a correlação entre o grau de entrada/saída e as taxas de dados recebida/enviada. Não foi observada uma correlação entre o grau de entrada do vértice e taxa de dados recebida pelo par porque o primeiro varia significativamente enquanto o segundo permanece estável em torno da taxa de vídeo. Por outro lado, observou-se uma correlação positiva entre grau de saída e a taxa de envio.

Dada essa correlação, os autores também exploraram a base de dados do sistema UUSee para predizer a taxa de envio de um par, especificamente o *throughput*, em função do seu grau de saída [Wu et al., 2007]. Contudo, o fluxo de *chunks* entre os pares foi analisado considerando a hora do dia e o provedor de Internet (ISP) de um par. Foram propostos modelos diferentes para cada hora do dia e para cada tupla de ISP formada pelo par fornecedor e par receptor de *chunks*. Esses modelos são lineares com dois parâmetros, intercepção e inclinação da reta, estimados com regressão linear. A intenção dos autores é que os modelos sejam utilizados por um par para selecionar parceiros baseado na predição do *throughput*. Porém, mesmo considerando padrões de tráfego entre ISPs e horários do dia, as predições de *throughput* pelos modelos lineares têm erros altos e elas foram utilizadas apenas para ordenar os pares de acordo a sua cooperação (*ranking*). O modelo proposto nesta dissertação não foca apenas em um

ranking de pares, mas busca prever uma estimativa do nível de cooperação mais precisa, que pode ser utilizada para produzir um *ranking* também.

Oliveira et al. [2010] exploraram propriedades topológicas do sistema SopCast para caracterizar super pares na rede P2P. Eles correlacionaram as métricas de centralidade grau, proximidade e *betweenness* com a taxa de *upload* dos pares. Assim como em Tang et al. [2009], os autores analisaram a topologia completa da rede P2P, obtida a partir de experimentos com computadores do Planetlab e um canal privado do SopCast. Foi observada uma correlação positiva alta entre as métricas de centralidade e a taxa de *upload*, com coeficiente de correlação de Pearson em cerca de 0,85 para as 3 métricas de centralidade. As correlações foram calculadas entre o índice ordenado (*ranking*) das duas medidas sendo correlacionadas.

O trabalho desenvolvido nesta dissertação iniciou analisando a correlação entre as métricas de centralidade e o nível de cooperação dos pares, estimado pela taxa de *upload* por *download* de dados. Mas diferente do trabalho de Oliveira et al. [2010], o foco não foi direcionado apenas para o *ranking*, mas também para o relacionamento das medidas diretas dessas métricas. Isso permitiu inferir relações interessantes que levaram para o projeto de um modelo de predição do nível de cooperação. Similarmente, o modelo proposto por Wu et al. [2007] também foca em *ranking* dos pares, pois ele é um modelo linear simples e a sua construção é baseada em dados sobre os pares sumarizados ao longo de dias diferentes. O modelo proposto nesta dissertação, ao contrário, é não linear e busca predizer o nível de cooperação de um par em valores absolutos com a maior precisão possível. Para isso, ele é construído dinamicamente em intervalos de tempos suficientes para capturar as constantes mudanças da rede P2P. O capítulo 4, especificamente a seção 4.2, descreve os detalhes sobre a construção desse modelo.

Capítulo 3

Análise da Cooperação no SOPCast

Neste capítulo será dada uma visão geral da arquitetura e do funcionamento do sistema SopCast, que é a aplicação P2P de vídeo ao vivo utilizada como base de dados dessa dissertação. Em seguida será mostrada a metodologia aplicada para coletar dados dessa aplicação em uma transmissão de vídeo ao vivo com pares se comportando de modo semelhante a participantes reais de um canal transmitindo um evento. Por fim, é apresentada uma análise do nível de cooperação dos pares nesses dados coletados.

3.1 O Sistema SopCast

SopCast é um sistema P2P de distribuição de vídeo ao vivo que é muito popular na Internet. Segundo as medições do serviço Google Trends[1], o volume de buscas por esse sistema nos últimos doze meses (07/2011 a 06/2012) é superior ao de outros sistemas P2P populares como PPLive e UUSee. O sistema SopCast mantém vários canais públicos, cujo acesso é irrestrito, mas também permite que se crie um canal ao vivo privado que transmita conteúdo para um conjunto restrito de clientes. Cada canal transmite conteúdo ao vivo através de sua própria rede P2P sobreposta, independente de quaisquer outros canais mantidos pela aplicação.

O sistema SopCast usa arquitetura *mesh-pull* [Hei et al., 2008], logo a rede sobreposta de cada canal é baseada em malha com pedido explícito de dados. Nessa arquitetura, um servidor, que gera o conteúdo ao vivo, divide a mídia em pedaços, chamados *chunks*, e os distribui na rede P2P para posterior exibição. Para receber o conteúdo ao vivo, um par faz um pedido explícito de todos os chunks de mídia necessários aos seus parceiros, isto é, aos seus vizinhos na rede sobreposta. Os pares usam o protocolo UDP (User Datagram Protocol) para realizar todo tipo de

[1] `http://www.google.com/trends/?q=sopcast,+pplive,+uusee&ctab=0&geo=all&date=all&sort=0`.

comunicação no sistema SopCast. Essa comunicação compreende pacotes de controle e pacotes de dados. Os pacotes de controle correspondem às mensagens de sinalização entre os pares de acordo a arquitetura *mesh-pull* e os pacotes de dados são mensagens transmitindo *chunks*.

Visto que o foco desta dissertação é na cooperação estimada pelos pacotes de dados, é necessário identificar esse tipo de mensagem na comunicação entre os pares. Contudo, o sistema SopCast é um protocolo proprietário e a identificação de pacotes de dados não é trivial. A forma comumente utilizada para identificar esses pacotes é pelo seu tamanho e os padrões correspondentes de entrega [Hei et al., 2007]. Nesse propósito, a comunicação entre os pares no sistema SopCast foi analisada nesta dissertação. A base para essa análise foram os trabalhos de Sentinelli et al. [2007] e Tang et al. [2009] que investigaram o funcionamento desse protocolo. Foram utilizados também traços de tráfego de redes em uma transmissão de vídeo ao vivo, obtidos de alguns computadores conectados a um canal popular do SopCast. Assim, para identificar os pacotes de dados foi estudado detalhadamente como os pares estabelecem parcerias e retransmitem os dados, como é descrito a seguir.

Quando um par ingressa no sistema, ele recebe uma lista de pares ativos na rede (subconjunto de todos os pares) do *tracker* e inicia imediatamente a trocar pacotes de controle com esses pares para estabelecer parcerias. Dado um conjunto de parceiros, um par faz pedidos explícitos de *chunks* a alguns deles. Após receber o pedido, o parceiro entrega uma série de pacotes de dados para o par requisitante. No caso desse par necessitar de mais *chunks* um novo pedido deve ser feito, por conseguinte, uma nova série de pacotes de dados entregue. De acordo as estimativas de Tang et al. [2009], cada *chunk* no SopCast tem um tamanho de 10 KBytes. Então, o parceiro atende um pedido segmentando os *chunks* requisitados em pacotes com tamanhos menores, seguindo os princípios de fragmentação do protocolo IP. O tamanho máximo de um pacote de dado estabelecido no SopCast é 1362 *bytes*. Logo, a entrega de *chunks* é realizada por uma sequência de pacotes de dados com tamanho máximo e é finalizada tipicamente por um pacote menor portando fragmentos de *chunks*.

A Figura 3.1 mostra histogramas do tamanho de pacotes trocados por três computadores que participaram de uma transmissão de vídeo ao vivo no sistema SopCast por uma hora. Eles são representativos dado que foram computados histogramas para outros 284 computadores e os resultados foram similares. O eixo x representa o tamanho dos pacotes agrupados em classes com intervalos de 50 bytes e o eixo y representa a porcentagem de pacotes de uma classe específica que o par enviou e recebeu durante a transmissão de vídeo. Como pode ser observado, os histogramas mostram três picos: dois picos em pacotes menores que 150 bytes (mais à esquerda) e

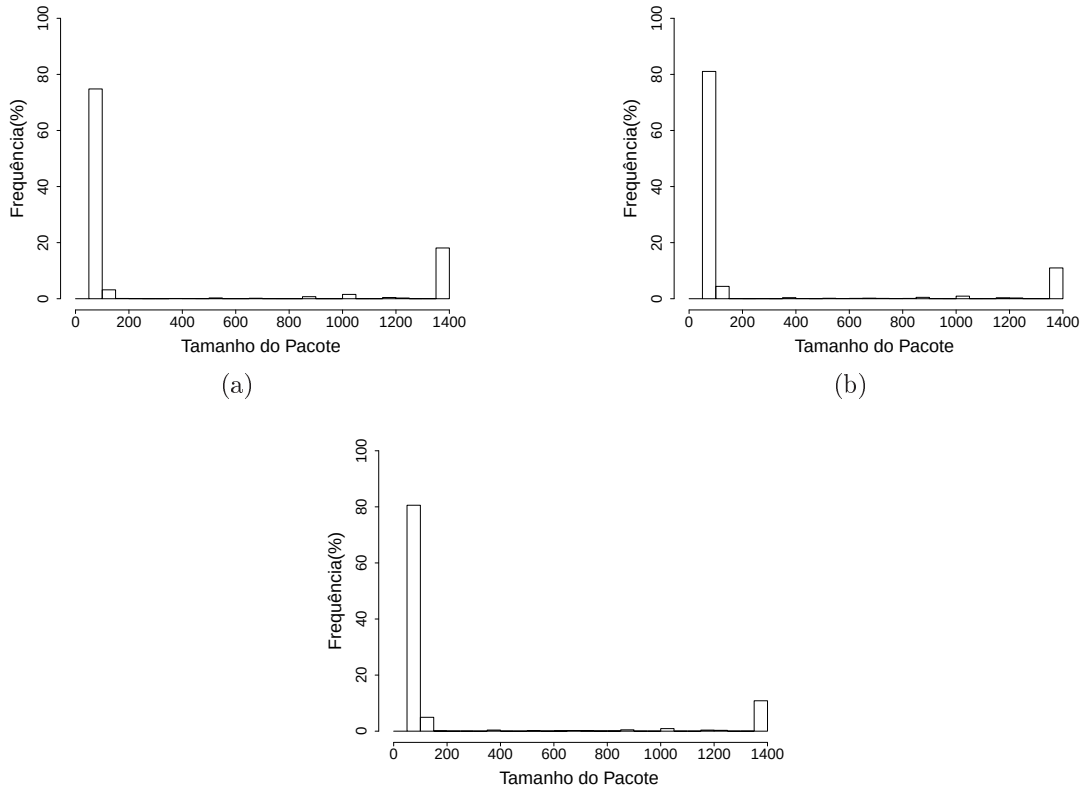


Figura 3.1: Histogramas do tamanho de pacotes recebidos e enviados por três computadores em uma transmissão de vídeo no sistema SopCast durante uma hora.

um pico em pacotes maiores que 1300 bytes (à direita). Para quantificar a porcentagem desses picos em todos computadores, foram computadas as suas médias, que são discutidas a seguir. Os erros dessas médias foram aproximadamente zero ($\pm 0.4\%$) com 95% de confiança, mostrando que os histogramas são muito similares .

Os dois picos à esquerda dos histogramas representam em média 87% dos pacotes trocados por um par e se referem aos pacotes de controle, que são emitidos com maior frequência. Por exemplo, um par emite pacotes do tipo *keep-alive* (84 bytes) aproximadamente a cada segundo para manter a comunicação com os parceiros e pacotes do tipo *hello* para iniciar e confirmar uma conexão (respectivamente 94 e 122 bytes) sempre que ele obtem um novo parceiro [Tang et al., 2009]. O pico à direita representa em média 10% dos pacotes trocados por um par e se refere aos pacotes de dados, que são aqueles de tamanhos maiores emitidos pelos pares, como foi explicado acima.

Todos os pacotes com tamanhos entre 150 e 1300 bytes representam em média 3% dos pacotes trocados por um par e podem se referir tanto aos pacotes de controle

ou pacotes de dados (fragmentos de *chunks*). Não foi possível diferenciá-los, pois eles aparecem finalizando uma sequência de pacotes de dados, como também aparecem em mensagens isoladas que um par troca com a maioria dos parceiros. Seguindo o estudo de Sentinelli et al. [2007], nesta dissertação serão considerados efetivamente pacotes de dados apenas os pacotes maiores que 1300 bytes, que representam a porcentagem de pacotes mais significativa (pico à direita na Figura 3.1) quando são removidos os pacotes de controle.

A próxima seção descreve experimentos que foram realizados no sistema SopCast e os pacotes de dados serão utilizados para investigar o nível de cooperação dos pares e a sua relação com as propriedades topológicas da rede.

3.2 Metodologia de Coleta de Dados no SopCast

Para coletar dados do SopCast, foram realizados experimentos em um ambiente controlado utilizando um canal de vídeo privado e computadores do Planetlab [Chun et al., 2003] que atuavam como pares da rede P2P. Nesta seção é descrito a configuração dos pares nos experimentos, a coleta e o processamento dos dados e por fim a modelagem da rede P2P sobreposta.

Seis experimentos foram executados em Novembro de 2011 utilizando de 350 a 450 pares estáveis no Planetlab. A premissa deste trabalho é analisar o nível de cooperação dos pares resultantes do protocolo SopCast. Logo, os pares não tiveram suas larguras de bandas limitadas, ou seja, a largura de banda disponível em cada computador foi administrada pelo protocolo.

Também, dado o foco em investigar o benefício de utilizar métricas de centralidade para predizer o nível cooperação de pares, optou-se por coletar dados de um canal privado, restringido a população aos pares do Planetlab. Ao fazer isso, foi possível coletar uma visão completa da rede sobreposta e computar medidas exatas de centralidade e o nível de cooperação de cada par. Para construir um canal privado do SopCast, foi projetado um servidor para codificar e transmitir um vídeo de 280kbps de 60 minutos de duração, que é justamente o tempo de cada experimento.

Antes de iniciar cada experimento, certificou-se que os pares tinham relógios sincronizados, para garantir que diferenças de horário entre eles pudessem ser negligenciáveis (menos que 1 segundo). Embora a administração do Planetlab adote o protocolo NTP[2] para sincronizar os relógios dos computadores, ainda existem computadores dessincronizados nessa plataforma, como pode ser observado

[2] `http://www.ntp.org`

na ferramenta de monitoramento CoMon [Park & Pai, 2006]. Assim, foi realizada uma rotina para selecionar apenas computadores sincronizados.

Nessa rotina, um computador do laboratório, com horário sincronizado, conectava ao computador do Planetlab, registrava o horário remoto (Planetlab), desconectava, e imediatamente registrava o horário local. Em seguida os horários, local e remoto, eram comparados e computadores que apresentavam uma diferença maior que 1 segundo não eram utilizados nos experimentos. O registro do horário e a desconexão de um computador do Planetlab tipicamente ocorria em menos de 1 segundo. Logo, a rotina executada possibilitou comparar os horários local e remoto para garantir computadores sincronizados nos experimentos.

Os experimentos realizados assumiram saídas e reingressos de pares no canal[3] (*churn*) de acordo o modelo de comportamento dos pares determinado em Borges et al. [2012]. Esse modelo define o comportamento de um par representado por dois estados: o estado ON e o estado OFF. O estado ON representa o tempo em que o par está ativo na rede, que é modelado por uma distribuição Weibull com parâmetros $\alpha = 2.032$ and $\beta = 0.233$. O estado OFF representa o tempo em que um par está fora da rede ou sem atividade, que segue uma distribuição exponencial com parâmetro $\lambda = 0.054$[4]. As distribuições desse modelo foram obtidas com a caracterização do comportamento de participantes reais do SopCast em várias transmissões de canais públicos.

O modelo de comportamento definido acima foi aplicado em todos os computadores do Planetlab (pares) da seguinte forma. Em cada experimento, todos os pares ingressavam no canal do SopCast ao mesmo tempo e permaneceram conectados por um instante inicial de 5 minutos. A partir desse período, foi iniciado o comportamento dinâmico de cada par na rede. Então o par alternava entre estados ON e OFF em intervalos de tempo obtidos aleatoriamente segundo as distribuições de probabilidade do modelo. A alternância de estados permanecia até esse par eventualmente sair do sistema ou o experimento chegar ao fim.

Enquanto estavam conectados ao canal, os pares utilizaram Wireshark[5] (*tcpdump*) para coletar todos os pacotes (controle e dados) trocados com os parceiros. As informações sobre esses pacotes eram armazenadas em um arquivo de *log* local no par. Essas informações consistiam de: o horário (sob a granularidade de 1 segundo) em que cada pacote foi enviado ou recebido, o par de origem, o par de destino e o

[3]Inicialmente foram realizados experimentos sem *churn* e a correlação entre propriedades topológicas e a centralidade do par foram diferentes dos resultados mostrados na Seção 4.1. Como em cenários reais os pares exibem churn, essa configuração foi mantida porque ela é mais realista.

[4] As funções de densidade de probabilidade são: $p_X(x) = \alpha\beta x^{\beta-0}e^{-\alpha x^{\beta}}I_{(0,\infty)}(x)$ para Weibull, e $p_X(x) = \lambda e^{-\lambda x}$ para Exponencial.

[5]http://www.wireshark.org

tamanho do pacote.

Ao final da transmissão de vídeo em cada experimento, os *logs* dos pares foram recuperados para um computador local no laboratório. Então esses *logs* foram combinados em um único *log* e processados da seguinte forma. Como o foco dessa dissertação é cooperação entre os pares, foram selecionadas apenas os pacotes de dados, ou seja, pacotes maiores que 1300 bytes, como foi estudado na seção anterior. Em seguida, o *log* foi ordenado de acordo com o horário dos pacotes e foram descartados os primeiros 5 minutos da transmissão, considerando apenas o período em que os pares exibiam *churn*. Esse período foi dividido em intervalos de tempo consecutivos, onde cada intervalo correspondia a uma janela de tempo predefinida de duração W. Assim, reconstruiu-se a rede sobreposta dinâmica do SopCast como uma sequência de fotografias, tiradas a cada W segundos, cada uma refletindo o estado da rede nesse intervalo de tempo.

A rede P2P sobreposta foi modelada para cada janela de tempo na forma de um grafo onde os pares representam vértices e os pacotes de dados trocados entre eles representam arestas. Foram utilizados grafos direcionados e não direcionados. No grafo direcionado, cada aresta informa a direção do fluxo de vídeo, ou seja, par de origem e par de destino. No grafo não direcionado, cada aresta informa apenas a parceria estabelecida entre dois pares para troca de vídeo.

A etapa seguinte no processamento dos *logs* corresponde às medições das taxas de *upload* e *download* dos pares para computar o nível cooperação. Essa computação também foi realizada durante janelas de tempo. Na sessão seguinte, são analisados os níveis de cooperação dos pares para todas as janelas de tempo nos seis experimentos realizados.

3.3 Cooperação de Pares no SopCast

Nesta seção, são analisados os níveis de cooperação de pares no SopCast durante os experimentos realizados. Define-se o nível de cooperação do par i durante a janela de tempo t como

$$NC(i,t) = \frac{upload(i,t)}{download(i,t)} \tag{3.1}$$

onde $upload(i,t)$ e $download(i,t)$ correspondem respectivamente ao volume total de dados que i enviou e recebeu durante t.

Uma questão importante é a escolha da duração W da janela de tempo em que os dados são analisados. É desejável um valor que suavize grandes variações observadas em escalas de tempo muito pequenas, mas que capture as propriedades da dinâmica entre os pares durante uma transmissão. Tang et al. [2009] propuseram o uso de uma janela de tempo de 2 segundos para analisar propriedades topológicas da rede sobreposta no SOPCast. Contudo, nesta dissertação foi verificado que esse intervalo de tempo leva a variações grandes do nível de cooperação (NC) dos pares. A Figura 3.2 mostra o NC de um dado par medido em janelas consecutivas durante o intervalo de 5 minutos de um experimento, para valores de W iguais a 2 e 30 segundos. O NC medido de um par apresenta grande variação para W=2, enquanto para W=30, a curva apresenta um comportamento menos errático. Esta variação pode ser quantificada pelo coeficiente de variação CV (razão do desvio padrão pela média) de todos os valores de NC obtidos para o dado par. O CV é 1.35 para W=2 mas apenas 0.46 para W=30.

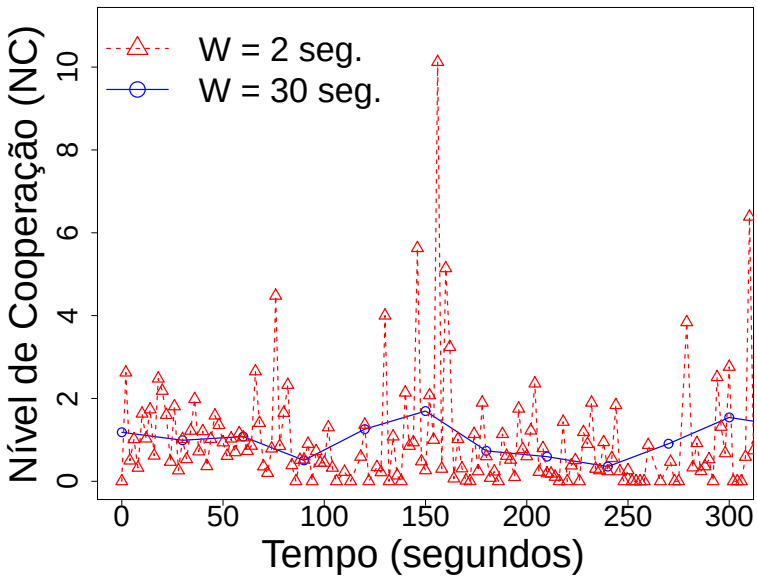


Figura 3.2: NC medido em um determinado par com janelas de tempo W igual a 2 e 30 segundos

Buscando analisar o valor de W mais adequado para expressar o NC dos pares, os coeficientes de variações dos NCs de cada par foram computados para todos os experimentos utilizando janelas de tempo com W igual a 2, 30, 60 e 90 segundos. A Figura 3.3 mostra a distribuição de probabilidade acumulada de todos os coeficientes de variações computados. Claramente, janelas de tempo com $W = 2$ tendem a uma variação maior do NC (coeficientes de variações altos), ao passo que as janelas de tempo com W igual a 30, 60 e 90 possuem distribuições similares com coeficientes de variações menores, mostrando que elas incorrem em variações menores dos níveis de cooperação de cada par. Visto a grande variação do NC medido nos pares considerando $W = 2$, o foco desse trabalho será para valores de W iguais a 30, 60 e 90.

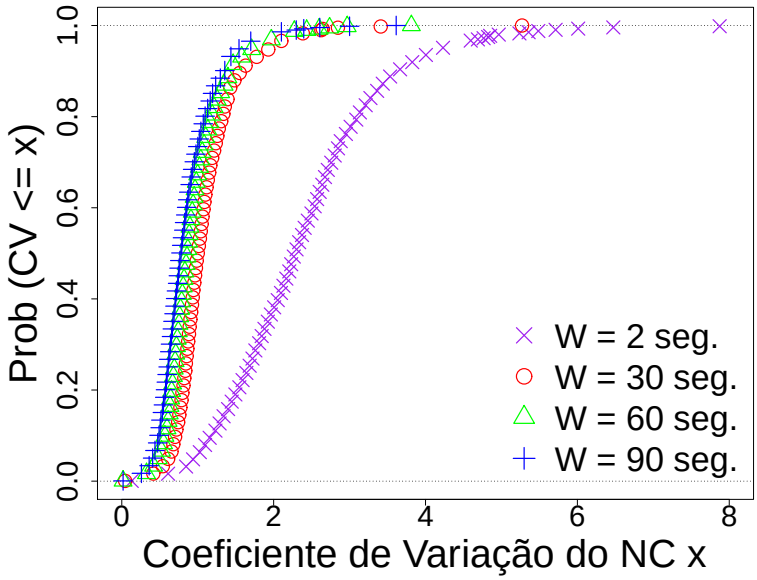


Figura 3.3: Distribuição do coeficiente de variação do nível de cooperação dos pares em janelas de tempo diferentes

Tendo escolhido o valor de W, analisa-se então os valores de NC de pares do SopCast. Isso é feito analisando a distribuição dos valores de NC de todos os pares durante todas as janelas de tempo dos experimentos. A Figura 3.4 mostra os valores de NC em escala logarítmica no eixo x. Analisando as distribuições desses valores, observa-se dois comportamentos extremos nos pares. De um lado, existem pares muito cooperativos que transmitem muito mais dados do que recebem. De um outro lado, há pares muito pouco cooperativos, que transmitem muito menos do que recebem. Por exemplo, a Figura 3.4 mostra que, considerando janelas de tempo de $W = 60$ segundos de duração, cerca de 34% de todos os valores de NC medidos são menores ou iguais a 0.01, valor que corresponde a pares muito pouco cooperativos que, apesar de receberem o conteúdo de vídeo completo, não retransmitiram nem mesmo um centésimo dos

segmentos recebidos para outros pares. Estes resultados ilustram quão desbalanceada é a distribuição de carga do SopCast, dado que cada par poderia utilizar toda a largura de banda disponível nos computadores do Planetlab.

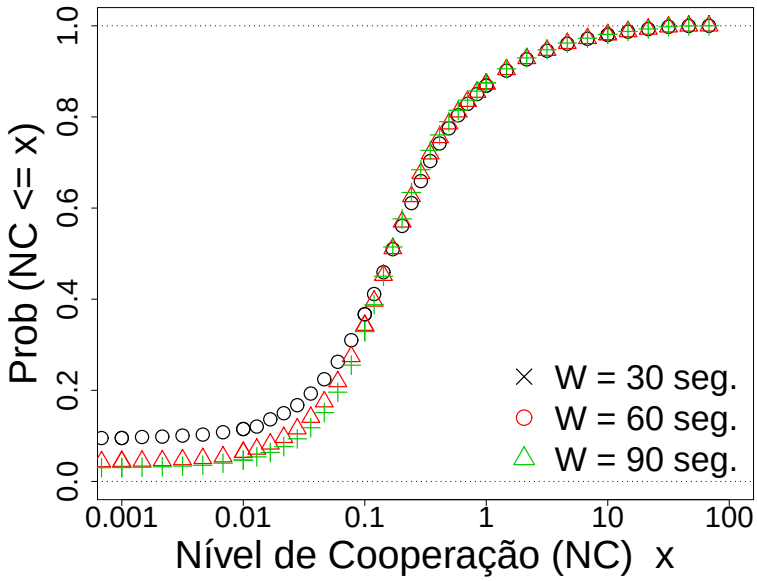


Figura 3.4: Distribuição do nível de cooperação dos pares em janelas de tempo com durações (W) diferentes

Considerando como pouco cooperativo um par que realiza mais *downloads* do que *uploads* para seus parceiros (como em Locher et al. [2006]), isto é, pares com $NC < 1$, a Figura 3.4 mostra que a rede P2P contém cerca de 87% de pares pouco cooperativos para as três durações de janelas de tempo. Assumindo uma definição menos rígida e considerando como pouco cooperativo um par com NC menor que 0.5 (0.2), essa fração cai para 79% (56%). Estas observações indicam que é provável que o SopCast não implemente nenhum mecanismo de balanceamento de carga entre pares, o que também foi observado anteriormente no PPLive [Piatek et al., 2010]. A falta de tais mecanismos gera uma preocupação, pois pares cooperativos (particularmente os muito cooperativos) podem se sentir desencorajados a participar da rede devido ao alto consumo de banda. Além disso, a grande presença de pares pouco cooperativos pode causar atrasos e interrupções de transmissão, podendo levar até mesmo ao colapso do sistema devido à redução na contribuição de pares.

Capítulo 4

Predição do Nível de Cooperação dos Pares

Neste capítulo é investigada a correlação entre o nível de cooperação dos pares e as métricas de centralidade grau, proximidade e *betweenness*. A seguir, é proposto um modelo de predição do nível de cooperação de um par a partir da métrica de centralidade mais correlacionada. Também são discutidas algumas considerações práticas para que esse modelo seja efetivamente utilizado em aplicações P2P de vídeo ao vivo. Por fim, é investigada a exploração de algumas propriedades topológicas da rede P2P sobreposta para mitigar a ação de pares maliciosos em conluio.

4.1 Nível de Cooperação e a Centralidade do Par

Nesta seção investiga-se a relação entre a centralidade de um par e sua importância relativa numa rede. Mais precisamente, é estudada a relação entre a centralidade de um par e o seu nível de cooperação (NC) em uma rede P2P. O objetivo é investigar o benefício de se utilizar métricas de centralidade de um par para prever seu nível de cooperação. Mais especificamente, deseja-se verificar a precisão relacionada à utilização das métricas de centralidade do par i durante a janela de tempo t para prever seu nível de cooperação na janela $t+1$. Há duas hipóteses por trás desta ideia: (1) a centralidade de um par está relacionada ao seu nível de cooperação; e (2) a centralidade de um par permanece suficientemente estável ao longo de janelas consecutivas. Primeiramente, investiga-se se essas hipóteses são verdadeiras no conjunto de dados do estudo.

Consideram-se as três métricas de centralidade comumente utilizadas, introduzidas na Seção 2.3, grau, *betweenness* e proximidade. Contudo, a aplicação dessas métricas será de acordo com o contexto específico em estudo. Em particular, a

métrica *betweenness*, que é computada sobre o grafo da rede completo, será expressa em valores relativos. Isso permite a comparação dessa métrica para redes de tamanhos diferentes em cada janela de tempo, o que ocorre devido ao *churn* dos pares (ver Seção 3.2). Os valores de *betweenness* são normalizados pelo maior valor possível para essa métrica na janela de tempo sob análise. Nesse caso, dada a quantidade de pares N na janela de tempo em questão, o *betweenness* máximo é expresso por $b_{max}(N) = \frac{(N)^2-3N+2}{2}$ [Freeman, 1979]. A métrica proximidade já foi definida em valores relativos visto que ela mede a distância relativa de um par para os demais pares na rede. Por outro lado, a métrica grau, que expressa a atividade local do par e não é calculada sobre o grafo completo, será expresso em valores absolutos.

O modelo de grafo direcionado será utilizado para as métricas grau e proximidade de modo que elas expressem o fluxo de vídeo entre os pares. Logo, essas métricas serão aplicadas na forma de grau de saída e proximidade de saída. Apenas a métrica *betweenness* será computada sobre o grafo não direcionado pelo seguinte motivo. Pares que têm o grau de saída muito maior que o grau de entrada teriam poucos caminhos mínimos que os cruzam e acabariam por ter um *betweenness* baixo. Contudo, tais pares têm um nível de cooperação alto na maioria das vezes. Logo, *betweenness* será aplicada de acordo a sua definição clássica em [Freeman, 1979].

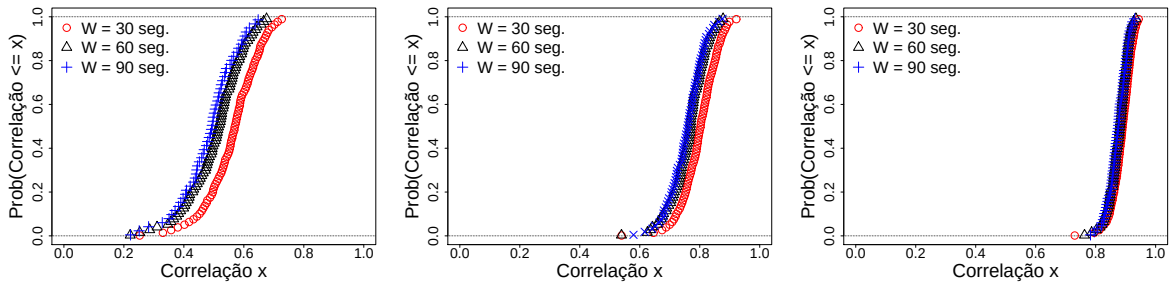
No contexto específico de sistemas P2P de transmissão de vídeo ao vivo, as métricas grau de saída, proximidade de saída e *betweenness* podem ser interpretadas da seguinte maneira. O grau de saída expressa a atividade local de um par na rede, ao passo que, a proximidade de saída indica os pares com capacidade de distribuir dados na rede mais rapidamente. Valores de *betweenness* alto podem refletir pares que conectam diferentes grupos de pares na rede ("pontes" ou "*gate keepers*"). Esses pares podem ter uma carga alta de comunicação por estarem situados em pontos estratégicos da rede sobreposta [Gkorou et al., 2011]. Sumarizando, as métricas *betweenness* e proximidade de saída capturam a importância relativa de um par de acordo a sua posição na rede, enquanto o grau de saída captura a importância de um par pela sua atividade local, ou seja, a quantidade de parceiros atendidos em uma janela de tempo.

A correlação entre cada métrica de centralidade e o nível de cooperação dos pares é analisada a partir do coeficiente de correlação de Spearman, que é uma medida não paramétrica da dependência estatística entre duas variáveis [Kendall & Gibbons, 1975]. Esse coeficiente é adequado para indicar relações não lineares entre duas variáveis, pois ele utiliza a ordenação das variáveis (*ranking*) ao invés dos seus valores diretos. A vantagem desse coeficiente é que ele também captura relações lineares. Assim, para cada janela t analisada, calcula-se a correlação de Spearman ($\rho(t)$) entre a centralidade do par i, de acordo com cada métrica considerada, e seu NC durante t, como mostra

a equação abaixo:

$$\rho(t) = \frac{\sum_i [x(i,t) - \bar{x}(t)][y(i,t) - \bar{y}(t)]}{\sqrt{\sum_i [x(i,t) - \bar{x}(t)]^2 \sum_i [y(i,t) - \bar{y}(t)]^2}} \tag{4.1}$$

onde $x(i,t)$ e $y(i,t)$ representam a posição do par i nos *rankings* de pares criados utilizando os valores da métrica de centralidade e do NC, respectivamente, na janela de tempo t. Os valores $\bar{x}(t)$ e $\bar{y}(t)$ representam respectivamente as médias das posições x e y para todos os pares i em t. Quando há empates nos valores das métricas de centralidade ou nos valores de NC de múltiplos pares em uma janela t, as posições $x(i,t)$ ou $y(i,t)$ para os pares empatados recebem a média das posições que eles teriam, caso não houvessem empates[1]. Por exemplo, o *hanking* de valores $Z = \{0.1, 0.3, 0.7, 0.7, 0.9\}$, possuem posições $z = \{1, 2, 3.5, 3.5, 5\}$ [Myers & Well, 1995].



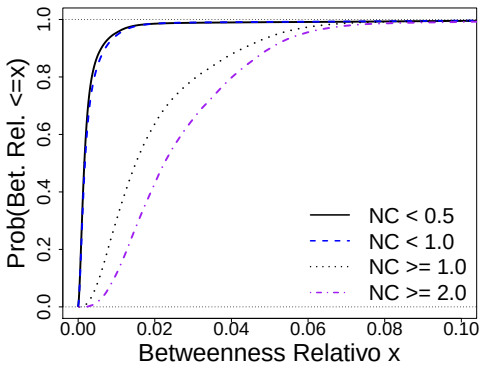
(a) NC e *Betweenness* (b) NC e Proximidade de Saída (c) NC e Grau de Saída

Figura 4.1: Distribuição dos coeficientes de correlação de Spearman entre medidas de centralidade e valores de NC obtidos em todas as janelas de tempo

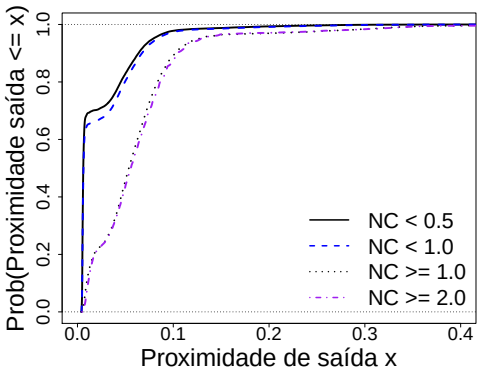
A Figura 4.1 mostra, para cada métrica de centralidade e duração W da janela, as distribuições dos coeficientes de Spearman $\rho(t)$ considerando todas as janelas t nos seis experimentos realizados. Observe que embora as três métricas sejam razoavelmente bem correlacionadas com o NC (correlações tendem a ser maior que 0,5 na maioria das vezes), a distribuição para o grau de saída notavelmente tende a valores maiores que as distribuições para *betweenness* e proximidade, independentemente de W. Isso indica que o grau de saída e o nível de cooperação de um par, de fato, são fortemente correlacionados, dado que em cerca de 99% dos dados observados o coeficiente de correlação de Spearman é maior que 0,8. A principal conclusão que pode ser obtida a partir desses resultados é que o posicionamento do par na rede sobreposta, dado pelas métricas de proximidade ou *betweenness*, provê menos informação sobre o seu nível de cooperação do que a sua atividade local, dada pelo grau de saída.

[1]Para as poucas janelas t onde não ocorrem empates para centralidade ou NC, a correlação de Spearman é calculada simplesmente por $\rho(t) = 1 - \frac{6\sum[(x(i,t)-y(i,t)]^2}{n(n^2-1)}$ [Myers & Well, 1995]

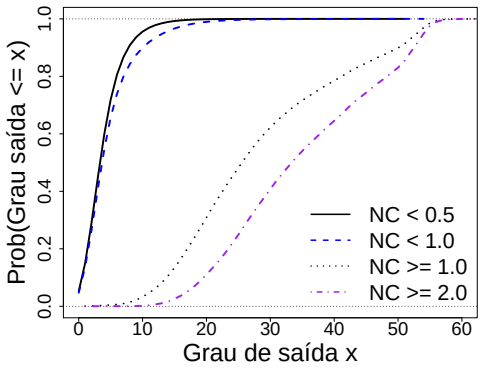
A capacidade de cada métrica de centralidade distinguir pares com diferentes níveis de cooperação também é analisada. A Figura 4.2 mostra as distribuições de grau de saída, proximidade de saída e *betweenness* para pares com diferentes valores de NC, considerando todas as janelas de duração $W = 60$ segundos. Resultados para outros valores de W são semelhantes e portanto são omitidos. Todas as três métricas apresentam distribuições bem distintas para cada grupo de pares: pares pouco cooperativos tendem a ter centralidade reduzidas. Contudo, o grau de saída novamente se destaca como a métrica mais discriminativa para o nível de cooperação dos pares. Esses resultados são evidências de que utilizar métricas de centralidade, em particular grau de saída, pode levar a boas estimativas do nível de cooperação de um par.



(a) Betweenness



(b) Proximidade de saída



(c) Grau de saída

Figura 4.2: Distribuição das medidas de centralidade por pares agrupados em diferentes níveis de cooperação ($W = 60$ segundos)

O foco agora é voltado para a segunda hipótese, isto é, de que pares mantêm valores de centralidade razoavelmente estáveis em janelas de tempo consecutivas. Em particular, analisa-se o módulo da diferença entre a centralidade de cada par em duas janelas t e $t + k$, para um k positivo. Por exemplo, $k = 1$ implica em analisar o módulo da diferença em duas janelas consecutivas. A análise é focada na métrica grau de saída, já que, dentre as três métricas estudadas, ela apresentou maior correlação e maior poder discriminativo entre os pares com diferentes níveis de cooperação. A Figura 4.3 mostra as distribuições das diferenças absolutas do grau de saída para diferentes valores de k, dados por $|g(i, t + k) - g(i, t)|$, considerando $W = 60$. Observa-se que para valores de k pequenos ($k \leq 4$), o grau de saída tende a variações pequenas: 80% das variações representam incrementos ou decrementos no grau de saída menores que 5. Logo, assume-se que o grau de saída dos pares, em geral, permanece suficientemente estável em pelo menos uma janela consecutiva $(t + 1)$.

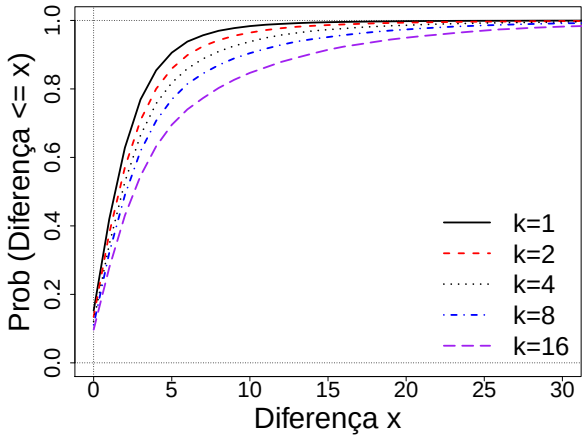


Figura 4.3: Distribuição do módulo da diferença entre o grau de saída dos pares em janelas de tempo t e $t + k$ ($W = 60$ segundos)

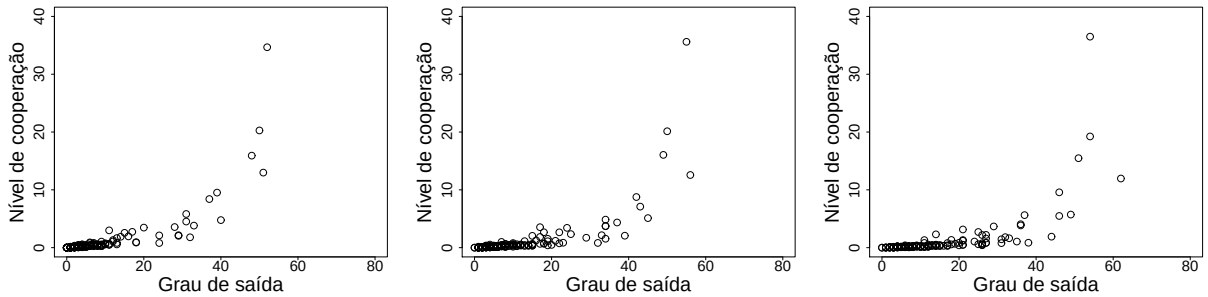
Retomando as duas hipóteses apresentadas no início dessa seção, observou-se que (1) a centralidade do par, dada especificamente pelo grau de saída, tem uma correlação alta com o seu nível de cooperação; e (2) essa métrica permanece suficientemente estável ao longo de janelas consecutivas. Dadas essas observações, este trabalho utilizará a métrica grau de saída para o desenvolvimento de um modelo de predição do nível de cooperação dos pares, na próxima seção.

4.2 Predição do Nível de Cooperação dos Pares

Nesta seção são propostos e avaliados modelos para prever o nível de cooperação de um par na próxima janela de tempo $(t+1)$ usando o seu grau de saída na janela atual (t). Incialmente são apresentados os modelos investigados. Em seguida é descrita a metodologia de avaliação e por fim são discutidos os resultados.

4.2.1 Modelos de Predição

Antes de propor modelos de predição foi realizado uma inspeção visual dos dados sendo estudados, como é recomendado por Jain [1991]. Especificamente, foram inspecionados vários gráficos de pontos em diferentes janelas de tempo que mostram a relação entre o grau de saída e o nível de cooperação dos pares. A Figura 4.4 mostra exemplos representativos de gráficos de pontos para janelas de tempo com W igual a 30, 60 e 90 segundos. Cada gráfico mostra a relação típica existente entre o grau de saída e o nível de cooperação. Pode-se verificar que essa relação exibe, notavelmente, uma curva com crescimento não linear. O fato de grau de saída ser a métrica mais correlacionada com o nível de cooperação era de certa forma esperado, já que um par que tem muitos parceiros tende a fornecer mais dados. Entretanto, é interessante notar que essa correlação é não linear, conforme a figura 4.4, o que poderia não ser óbvio no primeiro momento e essa é uma das principais contribuições dessa dissertação.



(a) $W = 30$ segundos (b) $W = 60$ segundos (c) $W = 90$ segundos

Figura 4.4: Relação entre o grau de saída e a cooperação em janelas de tempo típicas.

Para descrever a relação entre o grau de saída e o nível de cooperação, foram propostos e avaliados os seguintes modelos não lineares

$$\widehat{NC}(i,t+1) = a_0 + a_1 g(i,t) + a_2 e^{g(i,t)} \tag{4.2}$$

$$\widehat{NC}(i,t+1) = a_0 + a_1 g(i,t) + a_2 g(i,t)^2 + a_3 g(i,t)^3 + a_4 g(i,t)^4 \tag{4.3}$$

$$\widehat{NC}(i,t+1) = a_0 + a_1 g(i,t) + a_2 g(i,t)^2 + a_3 g(i,t)^3 \quad (4.4)$$

$$\widehat{NC}(i,t+1) = a_0 + a_1 g(i,t) + a_2 g(i,t)^2 \quad (4.5)$$

onde $g(i,t)$ representa o grau de saída do par na janela de tempo t e $\widehat{NC}(i,t+1)$ representa a predição para o nível de cooperação do par na janela $t+1$. Os parâmetros a_j são computados com minimização da soma do quadrado dos erros, como é utilizado tipicamente nas estratégias de regressão [Jain, 1991]. Nesse caso foi utilizado a implementação disponível no *software* livre de análises estatísticas GNU R[2].

Os gráficos de pontos da Figura 4.4 mostram que um modelo linear, como foi proposto por Wu et al. [2007], descreve apenas a relação entre grau de saída e nível de cooperação (NC) para os pares pouco cooperativos, i.e., pares com o nível de cooperação (NC) entre zero e um. Para os pares com o nível de cooperação maior que um essa relação exibe uma curvatura mais acentuada, que pode ser descrita por um modelo quadrático (Equação 4.5) ou exponencial (Equação 4.2). Isso é intuitivo visto que as equações desses modelos são suficientes para descrever uma curva monotonicamente crescente. Contudo, dado o dinamismo da rede P2P, algumas janelas de tempo podem apresentar uma relação com um padrão menos definido que os mostrados nas janelas de tempo representativas da Figura 4.4. Logo, faz-se necessário também examinar modelos polinomiais de mais alta ordem, como os mostrados nas Equações 4.4 e 4.3. Tais modelos podem se adequar melhor às possíveis variações na relação entre o grau de saída e o nível de cooperação dos pares ao longo de transmissão ao vivo.

Os modelos mostrados acima serão avaliados com o objetivo de determinar o modelo mais preciso. A precisão do modelo é quantificada pelo erro da predição, como é descrito na próxima seção.

4.2.2 Metodologia de Avaliação

A avaliação dos modelos é realizada por meio de estimativas produzidas em janelas de tempo sucessivas de modo que a *treino* e o *teste* dos modelos sejam feitos em janelas de tempo separadas. O treino ou calibração do modelo significa computar os parâmetros a_j utilizando regressão, onde a variável preditora (x) é o grau de saída e a variável resposta (y) é o nível de cooperação. Ambas as variáveis são extraídas da mesma janela de tempo. O teste significa computar o erro do modelo calibrado em um conjunto de dados diferente do treino, e.i. outra janela de tempo, para examinar a precisão do mesmo. Visando uma aplicação prática do modelo, neste trabalho, a

[2]http://www.r-project.org/

avaliação foi divida em três fases: (1) treino ou calibração do modelo (2) a coleta de dados de entrada, onde o grau de saída dos pares coletados da rede mais recentemente são introduzidos no modelo para obter uma predição do nível de cooperação, (3) a avaliação da predição, onde o nível de cooperação predito ($\widehat{NC}$) é comparado ao nível de cooperação medido (NC) na janela de tempo seguinte à coleta de dados.

A Figura 4.5 ilustra a metodologia de avaliação dos modelos. Sumarizando, os parâmetros a_j do modelo são computados na janela de tempo $t-1$ (Treino). O modelo é aplicado utilizando o grau de saída em t (Coleta). Por fim, a precisão da predição é avaliada para cada par comparando os valores preditos ($\widehat{NC}$) com os valores medidos (NC) na janela de tempo $t+1$ (Avaliação).

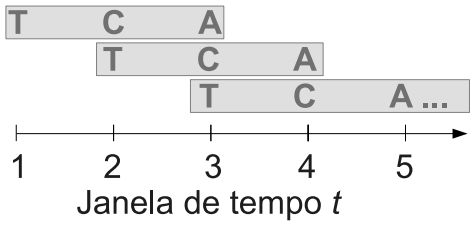


Figura 4.5: Metodologia de avaliação do modelo onde as letras T, C, A significam treino, coleta e avaliação, respespectivamente.

A precisão das predições é quantificada utilizando o erro absoluto do modelo dado por $\widehat{NC}(i, t+1) - NC(i, t+1)$. Caso haja predições com valores negativos, elas são truncadas para 0 visto que um NC negativo não faz sentido.

A precisão dos modelos é analisada pela distribuição dos erros absolutos computados sobre todos os pares e janelas de tempo obtidas nos experimentos. Para melhorar a visualização, os erros dos modelos são agrupados em três categorias de acordo o nível de cooperação (NC) medido. Assim, obteveram-se predições para pares pouco cooperativos ($NC < 1$), pares cooperativos ($1 \leq NC < 10$) e pares muito cooperativos ($NC \geq 10$). Essas categorias são uma escolha *ad hoc* e a precisão de um modelo é considerada razoável quando a sua distribuição do erro absoluto tende a zero, tal que as predições identificam uma porcentagem alta dos pares na categoria correta.

É importante ressaltar que nos experimentos realizados a cooperação do par foi dada pelo protocolo SopCast e não por limitações na largura de banda nos pares. Então um par pode estar em diferentes categorias ao longo do tempo dado o dinamismo da rede.

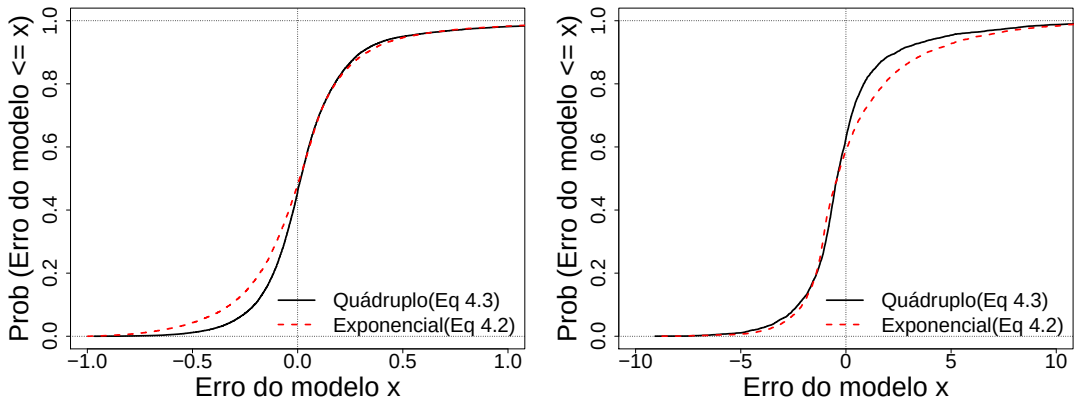
4.2.3 Resultados

Nesta seção, os modelos propostos acima são comparados com o objetivo de determinar prioritariamente o modelo mais preciso e possivelmente mais simples. A precisão do

modelo é quantificada pelo erro da predição, descrito na seção anterior, e o modelo mais simples é aquele que possui menos parâmetros. Com esse propósito, primeiramente são comparados os modelos exponencial (Equação 4.2) e polinomial de mais alta ordem (Equação 4.3). Em seguida, o modelo mais preciso dentre esses dois é comparado com os modelos polinomiais de menor ordem (Equações 4.5 e 4.4).

4.2.3.1 Modelos Exponencial e Quádruplo

Agora são apresentados resultados da avaliação do modelo exponencial e polinomial de mais alta ordem, o modelo quádruplo. As Figuras 4.6a, 4.6b e 4.7 mostram as distribuições acumuladas dos erros para o modelo quádruplo (Equação 4.3) e o modelo exponencial (Equação 4.2). Inicialmente, janelas com $W = 60$ segundos são utilizadas. Em geral, os erros de ambos os modelos tendem a zero nas três categorias, embora a concentração em torno de zero seja menor para a categoria de pares muito cooperativos. Embora essa última categoria seja muito importante para o sistema em termos de cooperação, ela é a mais difícil de prever porque ela compreende apenas 2% dos pares na rede (ver Seção 3.3 Figura 3.4). Isso implica em menos dados para computar os parâmetros a_j ("calibração") de modo que o modelo seja mais adequado para esses pares. A seguir, a precisão dos modelos é analisada com mais detalhes em cada categoria.



(a) Pares pouco cooperativos ($NC < 1$)

(b) Pares cooperativos ($1 \leq NC < 10$)

Figura 4.6: Distribuição dos erros de predição para os modelos exponencial e quádruplo ($W = 60$ segundos).

A Figura 4.6a mostra a distribuição dos erros absolutos para a categoria de pares pouco cooperativos. Observa-se nessa figura que o modelo exponencial é menos preciso porque ele subestima as predições mais do que o modelo polinomial. Isso pode ser visto quando a curva de erro do modelo polinomial cresce mais lentamente que a do modelo

exponencial para valores negativos no eixo x. Dessa forma, o modelo polinomial está mais concentrado em torno de zero. Por outro lado, a distribuição dos erros absolutos para a categoria de pares cooperativos, apresentada na Figura 4.6b, mostra que o modelo exponencial superestima mais as predições tornando-se menos preciso que o polinomial. Nesse caso, a curva de erro desse último modelo está mais próxima de zero.

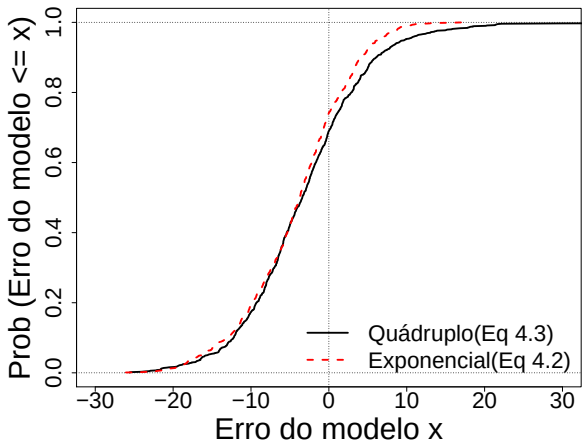


Figura 4.7: Distribuição dos erros de predição para a categoria de pares muito cooperativos com $NC \geq 10$ ($W = 60$ segundos).

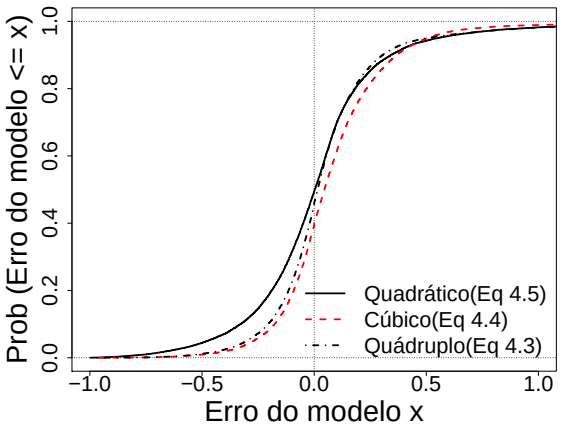
Para a categoria de pares muito cooperativos, cujo erro é mostrado na Figura 4.7, ambos os modelos apresentam erros aproximadamente similares, com o modelo quádruplo superestimando os valores de NC um pouco mais. Por exemplo, 80% dos erros do modelo exponencial se concentram na faixa entre ± 10, enquanto que para o modelo quádruplo 78% dos erros estão nesta faixa.

Considerando esses resultados, este trabalho optou pelo modelo quádruplo porque ele tem predições mais precisas para a categoria de pares pouco cooperativos e cooperativos que são a maioria dos pares na rede (98%). Além disso, as predições dos dois modelos para os pares muito cooperativos têm distribuições do erro muito similares. Tendo observado a superioridade do modelo quádruplo sobre o exponencial, são avaliados outros modelos polinomiais na seção a seguir.

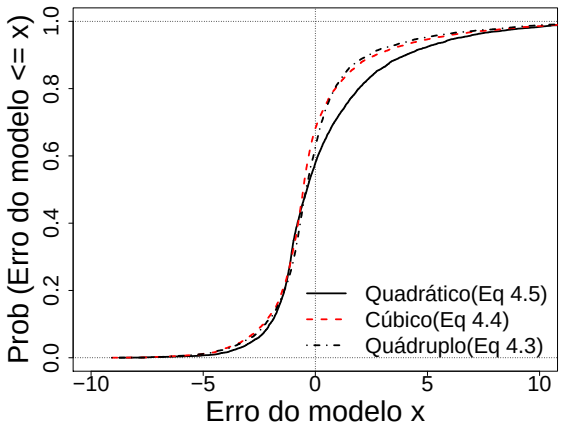
4.2.3.2 Modelos Polinomiais

O objetivo dessa seção é comparar os modelos polinomiais mais simples com o modelo quádruplo, avaliado na seção anterior. Desse modo, esse modelo é comparado aos modelos quadrático (Equação 4.5) e cúbico (Equação 4.4) em janelas de tempo com $W = 60$ segundos. A Figura 4.8 mostra as distribuições acumuladas dos erros dos modelos polinomiais para as três categorias de pares conforme o NC medido.

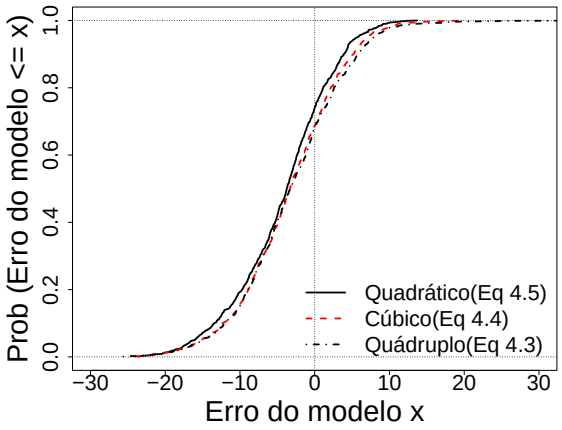
Observa-se na Figura 4.8a que os modelos quadrático e cúbico possuem notavelmente menor precisão para a categoria de pares pouco cooperativos: o modelo quadrático subestima muito os valores de NC ao passo que o modelo cúbico os superestima. A Figura 4.8b mostra que o modelo quadrático também é menos preciso para a categoria de pares cooperativos. Para esses pares, ele tende a superestimar os valores de NC, enquanto o modelo cúbico é tão preciso quanto o quádruplo. Para a categoria de pares muito cooperativos, mostrada na Figura 4.8c, o modelo quadrático ainda tem um erro ligeiramente maior que os modelos cúbico e quádruplo, que se mantêm com precisões similares. Logo, modelos polinomiais não lineares mais simples são menos precisos e não haveria vantagem em utilizá-los para substituir o modelo quádruplo, principalmente para a predição de pares pouco cooperativos.



(a) Pares pouco cooperativos ($NC < 1$)



(b) Pares cooperativos ($1 \leq NC < 10$)



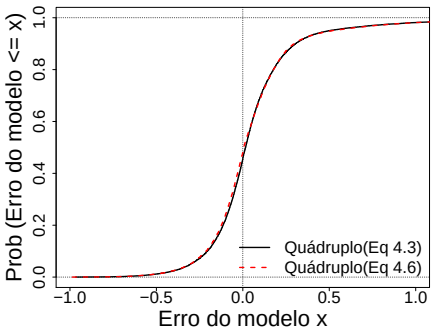
(c) Pares muito cooperativos ($NC \geq 10$)

Figura 4.8: Distribuição dos erros de predição para os modelos polinomiais (W=60seg.)

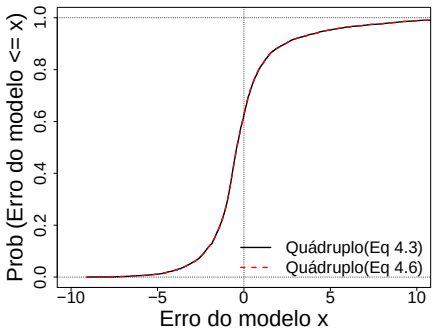
Até esse ponto foi observado que o modelo quádruplo (Equação 4.6) tem precisão superior (ou ao menos equivalente) a todos os modelos examinados nas três categorias de NC. Agora será avaliado se esse modelo ainda pode ser simplificado, desta vez diminuindo a sua quantidade de parâmetros. Com esse propósito, o parâmetro interceptador (a_0) foi removido porque ele é o menos significativo dentre os cinco parâmetros desse modelo. Isso foi verificado pelo teste de hipótese nula que é realizado pelo GNU R durante a computação de cada parâmetro. Esse teste indicou que a_0 é o parâmetro que tem maior probabilidade de ser zero em todas as janelas de tempo, o que é intuitivo desde que a cooperação de um par é zero quando o seu grau de saída é zero. Logo, é proposto o modelo quádruplo sem o parâmetro interceptador a_0 abaixo:

$$\widehat{NC}(i,t+1) = a_1 g(i,t) + a_2 g(i,t)^2 + a_3 g(i,t)^3 + a_4 g(i,t)^4 \tag{4.6}$$

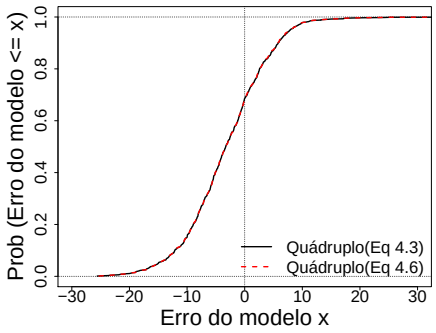
A Figura 4.9 mostra as distribuições acumuladas dos erros dos dois modelos quádruplos propostos para as três categorias de pares conforme o NC medido. Observa-se que a remoção do parâmetro a_0 não provocou mudanças significativas na precisão desse modelo. Logo, o modelo quádruplo (Equação 4.6) será adotado de agora em diante neste trabalho.



(a) Pares pouco cooperativos ($NC < 1$)



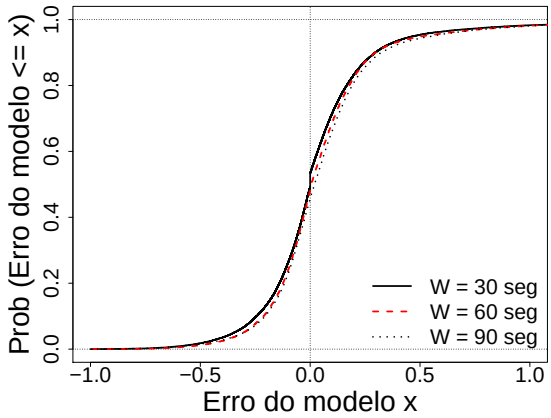
(b) Pares cooperativos ($1 \leq NC < 10$)



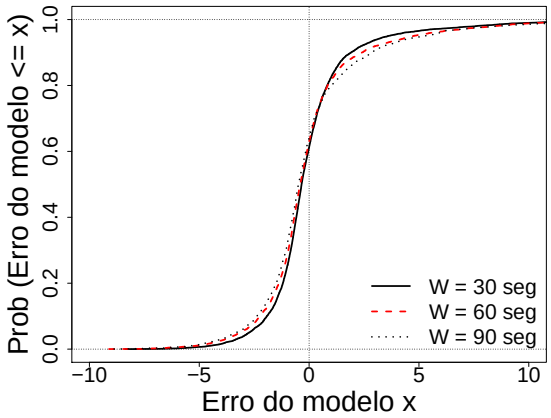
(c) Pares muito cooperativos ($NC \geq 10$)

Figura 4.9: Distribuição dos erros de predição para os modelos quádruplos com o parâmetro a_0 (Equação 4.3) e sem esse parâmetro (Equação 4.6).

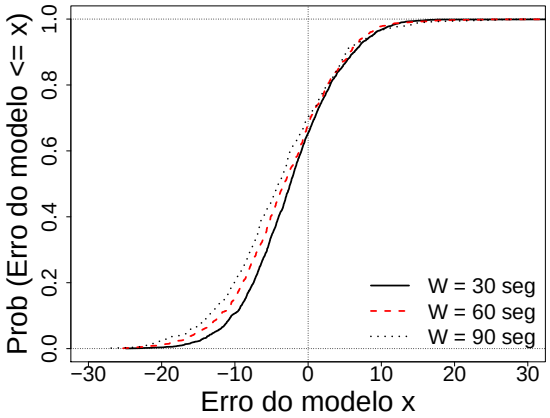
Os tamanhos W das janelas de tempo com 30 e 90 segundos agora são retomados para verificar se eles influenciam na precisão do modelo adotado. Assim, a distribuição dos erros desse modelo é analisada para janelas de tempo com W diferentes. A Figura 4.10 mostra que os erros do modelo se concentram em torno de zero e a escolha do valor de W tem menor impacto nos erros para a categoria de pares pouco cooperativos. Por exemplo, considerando as predições para a categoria de pares pouco cooperativos (Figura 4.10a), cerca de 70%, 72% e 71% das predições estão razoavelmente precisas, com um erro de $\pm$ 0.2 em relação ao NC medido, para W igual a 30, 60 e 90 segundos, respectivamente. De forma semelhante, a maior parte das predições para a categoria de pares cooperativos (Figura 4.10b) estão entre 2 unidades dos valores medidos. Contudo, o modelo tende a perder um pouco de precisão à medida que o valor W aumenta. De fato, 81%, 77% e 73% dos erros do modelo estão dentro desta faixa, para W igual a 30, 60 e 90 segundos.



(a) Pares pouco cooperativos ($NC < 1$)



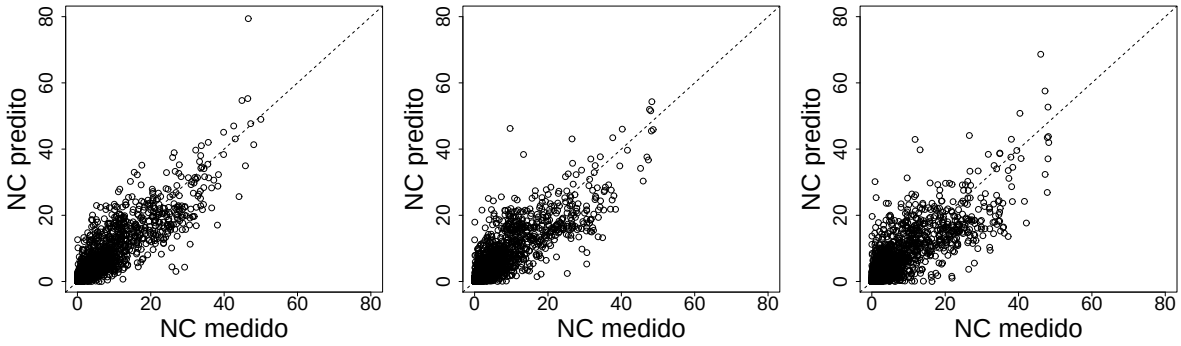
(b) Pares cooperativos ($1 \leq NC < 10$)



(c) Pares cooperativos ($NC > 10$)

Figura 4.10: Distribuição dos erros de predição para janelas de tempo com tamanhos diferentes W.

O tamanho de W tem maior influência nas predições para pares muito cooperativos, como mostra a Figura 4.10c, pois o modelo tende a subestimar os valores de NC medidos para valores maiores de W. Por exemplo, cerca de 52%, 47% e 45% dos erros estão dentro de 5 unidades para W igual a 30, 60 e 90 segundos enquanto que cerca de 86%, 83% e 77% dos erros estão dentro de 10 unidades dos valores medidos, para os valores correspondentes de W. Essa diferença de precisão mais visível para valores de W diferentes ocorre porque os pares muito cooperativos, além de representarem uma proporção muito menor na rede, variam mais seus parceiros, como será discutido a seguir. Assim, janelas de tempo longas, por exemplo 90 segundos, capturam menos essa dinamicidade, pois acumulam muitas parcerias de um par, diminuindo a relação entre grau de saída e NC, descrita pelo modelo. Por outro lado, janelas de tempo de 30 segundos descrevem melhor a dinamicidade dos pares muito cooperativos, tornando as predições do modelo um pouco mais precisas para eles.



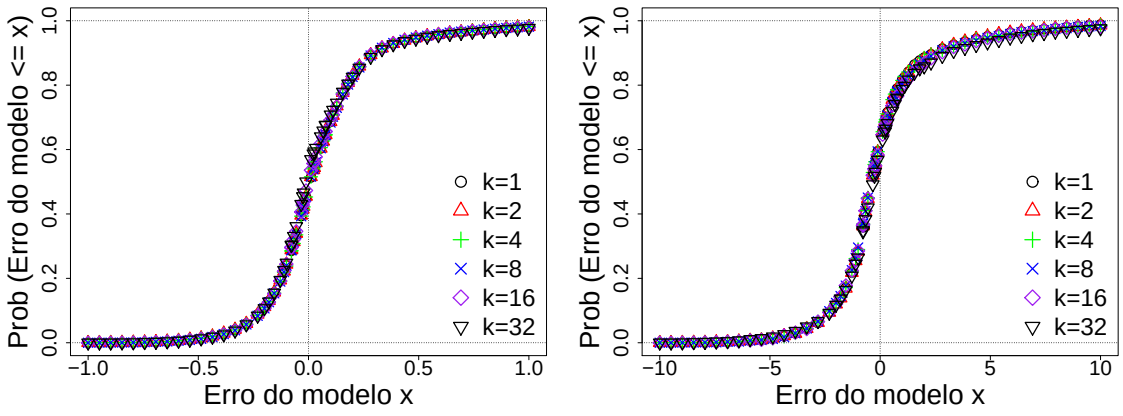
(a) $W = 30$ segundos (b) $W = 60$ segundos (c) $W = 90$ segundos

Figura 4.11: Valores de NC preditos versus medidos

O modelo também é avaliado a partir de um gráfico de pontos com os valores de NC medidos e NC preditos para cada par i e janela de tempo t. A Figura 4.11 mostra os resultados para todos os pares e janelas de tempo com W igual a 30, 60 e 90 segundos. A tendência à linearidade esperada é evidente. De fato, o coeficiente de Pearson de correlação linear ρ [Jain, 1991] é 0.91, 0.88 e 0.86 para valores de W iguais a 30, 60 e 90 segundos, respectivamente, o que indica uma forte correlação positiva. Novamente, é observado que a precisão diminui à medida que o tamanho de W aumenta, como foi discutido anteriormente, mas apesar dos erros absolutos, a maioria das predições são próximas dos valores de NC medidos. Por exemplo, considerando todas as predições para janelas com os três valores de W, o modelo é capaz de identificar 95% dos pares pouco cooperativos ($NC < 1$) na categoria correta. Mesmo apresentando erros maiores

para os pares muito cooperativos, observa-se que, considerando as predições do modelo e a categoria em que o par pertence em função do NC medido, as predições para pelo menos 80% desses pares estão na categoria correta ($NC \geq 10$). Para o restante deles, as predições apontam para a categoria de pares cooperativos ($1 \leq NC < 10$).

Até aqui foi discutida a precisão do modelo de regressão quando os parâmetros a_j são computados a cada janela de tempo $t-1$, e o modelo é aplicado sobre as métricas de centralidade coletadas durante t. Considera-se, agora, o caso em que o modelo é calibrado (isto é, os parâmetros são computados) na janela de tempo t e aplicado sobre as métricas de centralidade coletadas durante a janela k seguinte, isto é, a janela $t+k$. Observa-se que, os valores de NC preditos para cada janela de tempo $t+k$ são comparados aos medidos em $(t+k)+1$. O objetivo é analisar a sensibilidade do modelo à frequência com que ele deve ser calibrado (parâmetros recomputados), enquanto k define o intervalo de tempo entre duas calibrações consecutivas.



(a) Pares pouco cooperativos ($NC < 1$) (b) Pares cooperativos ($NC \geq 1$)

Figura 4.12: Distribuição dos erros de predição em função do período em que o modelo é calibrado dado por kW (W=60).

As Figuras 4.12 e 4.13 mostram as distribuições dos erros do modelo para vários valores de k e W=60 segundos. Mais uma vez, os resultados são mostrados separadamente em três categorias de acordo com os valores de NC medidos. O impacto de k nos erros do modelo mal pode ser percebido, em especial para as categorias de pares com o NC medido menor que 10 (Figura 4.12). Isso implica que o modelo pode ser calibrado uma vez e aplicado repetidamente para, pelo menos, os 32 minutos seguintes.

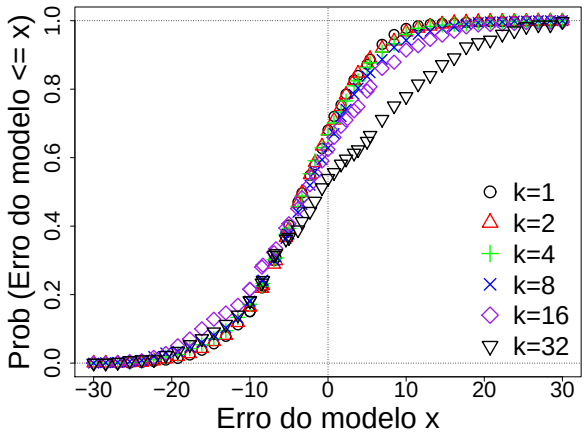


Figura 4.13: Distribuição dos erros de predição em função do período em que o modelo é calibrado dado por kW (W=60 segundos). Categoria de pares muito cooperativos com $NC \geq 10$

Em contraste, o modelo é um pouco mais sensível a longos períodos de calibração para predizer os valores de NC para pares muito cooperativos (Figura 4.13). Para essa faixa, os erros tendem a aumentar a partir de $k = 16$. Isso ocorre porque alguns desses pares passam por frequentes alterações em suas parcerias. Conjectura-se que tais alterações podem ser explicadas pela política adotada na maioria dos sistemas P2P de vídeo ao vivo, onde os pares selecionam novos parceiros aleatórios periodicamente [Guerraoui et al., 2010; Li et al., 2008a]. Provavelmente, um par que perde um de seus parceiros acaba por criar uma nova parceira com pares muito cooperativos, já que estes estão mais receptivos a novas parceiras. Isso provoca maior variação no número de parcerias desses pares em relação aos pares das demais categorias, o que exige uma calibração mais frequente. Entretanto, mesmo para a categoria de pares muito cooperativos, aumentar k até 16 causa pouco impacto na precisão do modelo e pode ainda ser aceitável.

4.3 Considerações Práticas

Os resultados mostrados na seção anterior indicam que métricas de centralidade, especificamente, grau de saída, podem ser usadas para predizer o nível de cooperação dos pares com precisão razoável. Essa predição pode ser utilizada para prover benefícios ou restrições aos pares de acordo o seu nível de cooperação. Por exemplo, a identificação de pares mais cooperativos pode ser explorada em políticas que favoreçam esses pares provendo dados diretamente do servidor de vídeo para melhorar a qualidade de serviço prestado a eles, como proposto pelos mecanismos Contracts [Piatek et al., 2010] e SGC

[Chatzidrossos et al., 2010]. Nesta seção, são discutidos alguns fatores que impactam na utilização prática do modelo desenvolvido. Em particular, discutem-se aspectos relacionados à coleta de dados para aplicação e calibração do modelo, à sobrecarga de processamento no sistema e à robustez da estratégia proposta a pares maliciosos.

Para calibrar e aplicar o modelo, um participante central do sistema precisa coletar periodicamente dados sobre as parcerias dos pares para reconstruir a rede P2P sobreposta e também coletar os valores de nível de cooperação (NC) para calibrar o modelo. A coleta dos valores de NC pode ser feita em intervalos de tempos maiores, conforme discutidos na seção 4.2.3.2. Além disso, ao invés de coletar dados de todos os pares da rede, pode ser coletado uma amostra dos pares, já que a métrica grau de saída não exige o conhecimento completo da rede. De fato, um trabalho a ser explorado no futuro é o desenvolvimento de estratégias de amostragem de pares na rede para calibrar o modelo.

Muitos sistemas P2P de vídeo ao vivo existentes já coletam dados sobre os pares periodicamente utilizando mensagens de controle. Por exemplo, o sistema Coolstreaming[3] possui um servidor de *logs* dedicado para coletar dados de cada par sobre qualidade de vídeo, volume de *download* e *upload*, e parcerias recentes [Li et al., 2008a]. De forma similar, tais informações são coletadas pelos sistemas UUSee [Wu et al., 2007] e PPLive. Este último, em particular, utiliza o *tracker* da rede P2P para realizar tal coleta [Piatek et al., 2010]. Isso mostra o interesse desses sistemas em monitorar tanto o tráfego de dados entre os pares como também as parcerias formadas entre eles ou a organização da rede P2P sobreposta.

No sistema SopCast o *tracker* também comunica com todos os pares por meio de mensagens de controle sincronizadas em intervalos de 2 e 3 minutos, como foi observado nos experimentos realizados neste trabalho, embora não tenha sido possível decodificar o conteúdo dessas mensagens. Caso as mensagens de controle trocadas entre os pares e o *tracker* no sistema SopCast não contenham informações de parcerias, incluí-las incorre em pequena sobrecarga de largura de banda. Isso porque os pares acumulam poucas dezenas de parceiros em uma janela de tempo. De fato, foi observado uma média de 15,65 parceiros em um minuto, e cada parceiro pode ser representado por apenas 4 *bytes*. A título de ilustração, se existem 10.000 pares na rede informando as suas parcerias a cada 2 minutos nas mensagens de controle, a sobrecarga adicional agregada no *tracker* vai ser de 40kbps. Essa sobrecarga é pequena dado que o monitoramento completo da rede, inclusive sua estrutura topológica, é de interesse dos administradores dos sistemas P2P de distribuição de vídeo como foi discutido anteriormente.

[3]Atualmente Roxbeam Inc. `http://www.roxbeam.com`

O segundo ponto importante é a sobrecarga de processamento no sistema. As informações sobre a rede sobreposta devem ser atualizadas à medida que novos dados são coletados dos pares. Nesse processo, as parcerias são verificadas a partir das informações dos dois pares envolvidos na comunicação (o par provedor e par receptor de *chunks*). O parâmetro de entrada do modelo, grau de saída dos pares, não requer computação extra uma vez que a rede sobreposta está disponível. O custo para obter essa métrica por par é fixo, ou seja, $O(1)$. Caso o modelo utilizasse as outras métricas de centralidade estudadas, o custo seria maior. Por exemplo o cálculo do *betweenness* exato tem complexidade $O(mn)$ em grafos não direcionados com m arestas e n vertices [Brandes, 2001]. Já o cálculo da proximidade exato tem o custo $O(nm + n^2 logn)$ [Eppstein & Wang, 2001].

Para calibrar o modelo é necessário obter o nível de cooperação dos pares (NC), que pode ser coletado com um custo extra de 4 bytes na mensagem de controle. Todavia, o NC é necessário apenas em intervalos de tempos mais longos, pois como foi mostrado na seção anterior (Figuras 4.12 e 4.13), a calibração do modelo em intervalos de 16 minutos ainda leva a resultados razoavelmente precisos. A calibração também tem um custo baixo, pois algoritmos de regressão para computar os parâmetros a_j do modelo não são computacionalmente intensivos [Jain, 1991]. Por exemplo, a calibração do modelo com GNU R para um milhão de pontos (variáveis preditoras e respostas) dura em média 3 segundos utilizando um computador com processador de 2 Ghz e memória de 8 GBytes. Possivelmente, a calibração exige menos recursos computacionais do que a utilização de mecanismos de autenticação no processamento de dados coletados dos pares, tal como o mecanismo Contracts [Piatek et al., 2010], como será discutido a seguir.

O terceiro ponto diz respeito à robustez a pares maliciosos, ou seja, pares que informam parcerias falsas para aumentar o seu NC indevidamente. A forma com que a rede sobreposta é reconstruída dificulta que um par malicioso sozinho engane o sistema. Isso porque as parcerias são consideradas quando são reportadas por ambos os pares envolvidos na comunicação. Logo, não é necessária nenhuma computação extra para verificar a autenticidade do grau de saída dos pares.

A verificação do NC de um par tem um custo alto de processamento caso seja adotado um mecanismo de incentivo, como Contracts, que utiliza apenas técnicas de autenticação (recibos criptografados). Em Contracts, cada par coleta recibos dos seus parceiros a cada volume v de *chunks* fornecidos. Ele também reporta esses recibos ao *tracker* para comprovar as suas cooperações ao final de uma janela de tempo. Essa abordagem aumenta significativamente a sobrecarga de processamento no *tracker*. Por exemplo, considerando que um par reporta pelo menos um recibo para cada parceiro

que ele forneceu dados (grau de saída) em um minuto e considerando o grau de saída médio no SopCast de 8,21 parceiros, o *tracker* processa cerca de 80.000 recibos por minuto em uma rede com 10.000 pares. O processamento desses recibos em tempo real exige uma infraestrutura com poder computacional alto, mesmo com implementação otimizada e paralelização de criptografia de chave pública (Algoritmo RSA) como sugerem os autores. Esse é um custo considerável comparado à abordagem proposta nesta dissertação, cujo custo com processamento é baixo.

Existem situações específicas em que recibos criptografados são necessários para comprovar parcerias na rede sobreposta. Tome como exemplo, um ataque coordenado dos parceiros de um par (conluio) para difamá-lo negando as parcerias estabelecidas com ele. Nesse caso, o modelo pode ser aplicado conjuntamente com o mecanismo Contracts, mas diminuindo a quantidade de recibos processados no *tracker*. Dessa forma, o sistema requisita recibos de um par apenas quando uma proporção considerável de suas parcerias não são confirmadas na manutenção da rede sobreposta. Esses recibos servirão como prova para desconectar os pares em conluio da rede.

Contudo, existe outro tipo de conluio mais difícil de ser identificado e o uso de recibos criptografados não é suficiente. Considere um cenário em que pares agem coordenadamente para beneficiar uns aos outros informando parcerias falsas entre si. Nesse caso, o uso de recibos criptografados não é capaz de detectar esses pares porque eles trocam recibos criptografados. Na próxima seção serão exploradas outras propriedades topológicas da rede sobreposta buscando diminuir as possibilidades desse tipo de conluio.

4.4 Resistência a Conluio

Nesta seção é discutida uma abordagem para aumentar a robustez do modelo de predição proposto à ação de pares maliciosos que agem coordenadamente para beneficiar uns aos outros, comportamento conhecido como conluio. Essa abordagem avança no uso de propriedades topológicas da rede sobreposta para identificar o comportamento dos participantes de um sistema P2P. Assim, serão descritos, nesta seção, os cenários de conluio para os quais essa abordagem foi desenvolvida, a sua aplicação e as suas limitações.

No caso estudado nesta dissertação, um conluio acontece quando um grupo de pares maliciosos reportam cooperações falsas entre eles. Desse modo esses pares aumentam os seus graus de saída e consequentemente aumentam os seus níveis de cooperação que é predito pelo modelo mostrado na Seção 4.2. Logo, o *tracker* não

pode identificar esse comportamento malicioso apenas conferindo as parcerias, pois os pares maliciosos podem confirmar as parcerias entre eles, inclusive, utilizando recibos criptografados [Piatek et al., 2010]. Contudo, existem dois cenários em que esse comportamento pode ser mitigado utilizando propriedades topológicas da rede. O primeiro cenário ocorre quando um grupo de pares maliciosos formam parcerias falsas entre si de modo que todos eles, ou a maioria deles, sejam beneficiados (i.e. tenham um grau de saída alto), como mostra a Figura 4.14a. O segundo cenário ocorre quando o conluio visa beneficiar poucos pares, por exemplo, alguns pares maliciosos tem o grau de saída alto enquanto os demais maliciosos apenas colaboram para isso, como mostra a Figura 4.14b.

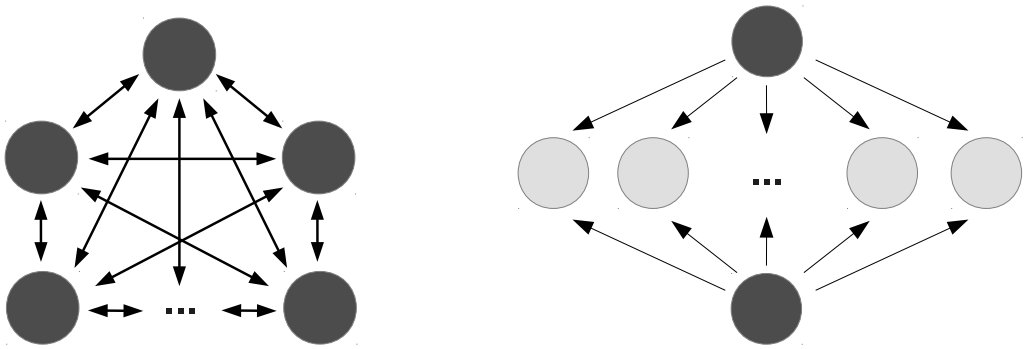


Figura 4.14: Cenários de conluio com pares maliciosos beneficiados (nodos escuros) e pares maliciosos auxiliares (nodos claros). Em (a) a maioria dos participantes do conluio são beneficiados, em (b) apenas dois pares são beneficiados com o conluio e os demais pares apenas auxiliam.

Nos cenários ilustrados na Figura 4.14, os nodos escuros representam os pares maliciosos que obtém um grau de saída alto e estão sendo beneficiados com o conluio. Os nodos claros representam os pares maliciosos que apenas auxiliam o conluio reportando parcerias falsas, mas não são beneficiados. Os pares auxiliares podem ser participantes reais do sistema ou identidades sintéticas. O uso dessas identidades, também conhecido como ataque *Sybil* [Douceur, 2002], no contexto de redes P2P de vídeo, consiste em o par malicioso registrar clientes no sistema que não assistem o vídeo verdadeiramente, mas apenas reportam contribuições falsas para favorecê-lo.

4.4.1 Uma Abordagem para Detectar Suspeitas de Conluio

Dados os cenários acima, esta seção mostra como uma métrica de agrupamento pode ser utilizada com o objetivo de identificar os pares agindo em conluio, especificamente os pares que estão sendo beneficiados. Essa identificação é realizada comparando o nível de agrupamento de pares com graus de saída altos com valores de referência analisados no sistema SopCast sem a presença de pares maliciosos. Assim, foi medido o nível de agrupamento dos pares com o maior grau de saída no protocolo SopCast com base nos experimentos realizados. Uma métrica de detecção de comunidades em redes chamada condutância [Kannan et al., 2004] foi utilizada para medir esse nível de agrupamento.

A condutância, no contexto desta dissertação[4], é computada a partir de um grupo de nodos na rede $S(i)$ formado pelo par i e os parceiros diretos para quem ele fornece dados. Seja v o total de parcerias com origem e destino em $S(i)$ durante a janela t, chamadas de parcerias internas, e seja s o total de parcerias com origem em $S(i)$ e destino em $\bar{S}(i)$ durante t, chamadas de parcerias externas, onde $\bar{S}(i)$ denota o complemento de $S(i)$. A condutância $c(S(i), t)$ de um grupo $S(i)$ na janela de tempo t é dado pela razão entre o total de parcerias externas e a soma do total de parcerias externas e internas do grupo $S(i)$ durante a janela t, como mostra a equação abaixo:

$$c(S(i), t) = \frac{s}{s + v} \tag{4.7}$$

Para entender melhor como a condutância é computada, a Figura 4.15 mostra um exemplo ilustrativo de uma rede sobreposta e a condutância é computada para o grupo $S(i)$ formado pelo par i (nodo escuro) e os seus parceiros de *upload* (nodos cinzas). As parcerias internas são representadas por arestas tracejadas e as parcerias externas são representadas por arestas pontilhadas. Logo, o valor da condutância para

[4]A métrica proposta em Kannan et al. [2004] não está definida para grafos direcionados. Nesta dissertação está sendo feita uma adaptação da métrica condutância originalmente proposta para o contexto de grafos direcionados aqui utilizados.

o grupo $S(i)$ na janela de tempo t é dado por $c(S(i), t) = 2/(2 + 11)$, onde os valores 2 e 11 se referem respectivamente ao número de parcerias externas e internas, conforme a Equação 4.7.

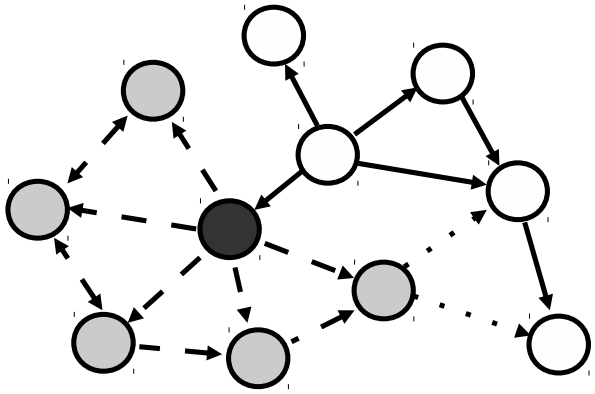
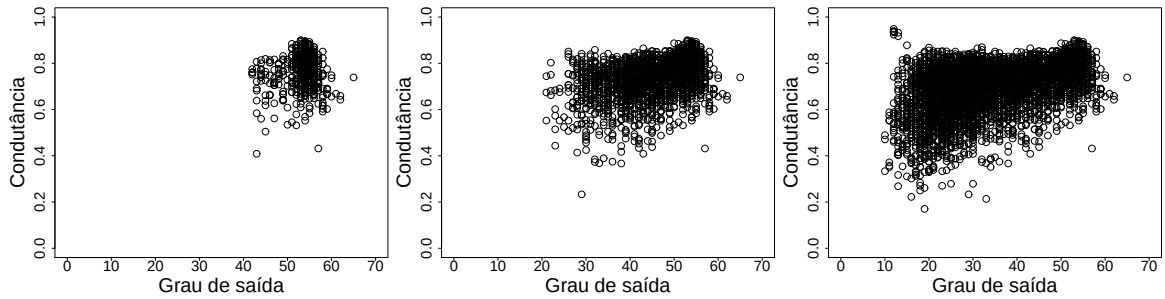


Figura 4.15: Exemplo ilustrativo de uma rede P2P sobreposta para computar condutância $c(S(i), t)$, onde o nodo escuro i e os nodos cinzas formam o grupo $S(i)$, as arestas tracejadas são as parcerias internas e as arestas pontilhadas são as parcerias externas.

Quanto menor é o valor da condutância para o grupo $S(i)$, maior é o seu nível agrupamento. A intuição para utilizar essa métrica é que os parceiros de um par muito cooperativo i tendem a formar menos parcerias internas e mais parcerias externas. Isso porque esses parceiros obtém a maioria dos dados do par i, então é mais provável que eles redistribuam esses dados aos demais pares da rede pertencentes ao conjunto $\bar{S}(i)$. Os pares maliciosos em conluio, ao contrário, buscam aproveitar ao máximo parcerias internas para aumentar os seus graus de saída artificialmente, conforme os cenários mostrados na Figura 4.14. Logo, espera-se que o grupo $S(i)$ formado por um par i com grau de saída alto legitimamente e os seus parceiros de *upload* tenham condutância alta (baixo nível de agrupamento).

Para motivar o uso da condutância, a relação dessa métrica com grau de saída é analisada em todas as janelas de tempo de 60 segundos dos experimentos realizados. Dado que o foco desta seção é detectar suspeitas de conluio entre os pares com grau de saída alto, são analisados os $k\%$ pares com o maior grau de saída em cada janela de tempo t. Para isso são utilizados os gráficos de pontos mostrados na Figura 4.16, onde cada ponto representa um par e $k = \{1, 4, 10\}$. A partir desses gráficos, pode ser observado que os pares com grau de saída alto tendem a ter condutância alta. Isso é observado principalmente na Figura 4.16a onde é exibida a relação para 1% dos pares com maior grau de saída e, como pode ser observado, os pontos estão concentrados no canto superior direito. Por outro lado, à medida em que o valor de k diminui, ou

seja, a porcentagem de pares com maior grau de saída é relaxada, aparecem pares com valores menores de condutância, como mostra a Figura 4.16c.



(a) $k = 1$ (b) $k = 4$ (c) $k = 10$

Figura 4.16: Relação entre as métricas grau de saída e condutância para os grupos de $k\%$ pares com o maior grau de saída na rede

Dadas a definição de condutância e a sua relação com grau de saída, agora é descrito como essa métrica pode ser utilizada para detectar os pares suspeitos de conluio. A detecção considera que o sistema oferece algum incentivo para o grupo de $k\%$ pares com o maior grau de saída. Então, deve ser verificado, com uma determinada confiança, se as parcerias dos pares candidatos a receberem o incentivo são legítimas. Assim, é proposto uma abordagem de verificação que utiliza como referência as distribuições dos graus de saída e condutância, mostradas na Figura 4.17. Essas distribuições foram obtidas do sistema SopCast em janelas de tempo de 60 segundos sem a presença de pares maliciosos, como será discutido a seguir. A abordagem de verificação tem os seguintes passos:

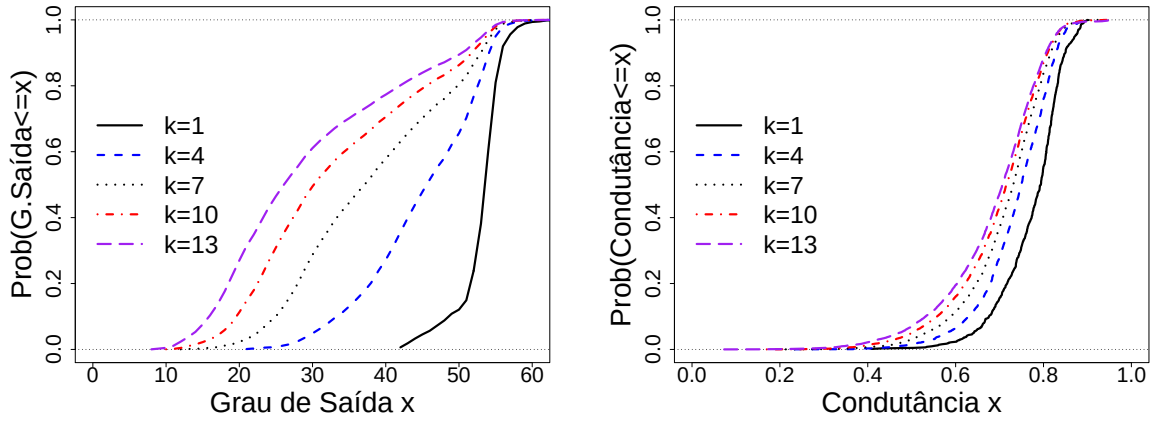
1. Deve ser informado a porcentagem k de pares com o maior grau de saída e uma confiança θ para o qual o sistema deve verificar conluio.

2. São determinados limiares mínimos para grau de saída e condutância, representados por g_m e c_m respectivamente, a partir das distribuições de referência mostradas na Figura 4.17 tal que:

$$prob(g(i,t)) > g_m) = \theta \quad \text{e} \quad prob(c(S(i),t) > c_m) = \theta$$

Logo, espera-se, com probabilidade θ, que um par cooperativo legítimo i tenha grau de saída $g(i,t) > g_m$ e condutância $c(S(i),t) > c_m$. Caso contrário, esse par é suspeito de conluio.

As distribuições de referência para grau de saída e condutância foram computadas para os grupos de $k\%$ pares com maior grau de saída, onde $k = \{1, 4, 7, 10, 13\}$. O valor de k foi limitado em 13% porque a partir dessa porcentagem a condutância já não exibe a relação com grau de saída mostrada na Figura 4.16. Os grupos de $k\%$ pares com maior grau de saída foram selecionados em cada janela de tempo t, nos experimentos realizados, conforme a a Seção 3.2. Esses experimentos focam no comportamento dos pares no sistema SopCast sem a presença de pares maliciosos. Logo, as distribuições computadas podem ser utilizadas como referencial para analisar a condutância dos pares com grau de saída alto nesse sistema. A partir da Figura 4.17b, pode ser observado novamente que quanto menor o valor de k, ou seja, a distribuição é restrita aos pares com o maior grau de saída, maior é a condutância. Isso pode ser visualizado quando as curvas desse gráfico tendem mais à direita à medida que k diminui.



(a) Grau de saída

(b) Condutância

Figura 4.17: Distribuições de referência para as métricas grau de saída e condutância para os grupos de $k\%$ pares com o maior grau de saída

A título de ilustração, considera-se uma verificação de conluio para 4% dos pares com maior grau de saída na rede ($k = 4$) com uma confiança de 99% ($\theta = 0.99$), conforme a abordagem descrita acima. Tomando as distribuições para $k = 4$ na Figura 4.17, 99% dos pares tem grau de saída maior que 26 e condutância maior que 0,47. Esses valores correspondem ao primeiro percentil das curvas dessas distribuições, e eles serão considerados os limiares mínimos $g_m = 26$ e $c_m = 0,47$. Assumindo os cenários de conluio mostrados na Figura 4.14, para um par malicioso ser confundido no grupo de 4% dos pares com o maior grau de saída e receber incentivos do sistema, ele precisa alcançar grau de saída maior que g_m e condutância maior que c_m. Isso é mais difícil de ocorrer nos referidos cenários porque a quantidade de parcerias internas é muito

superior às parcerias externas, ou seja, quando os pares maliciosos aumentam o grau de saída apenas com parcerias mútuas, eles diminuem a condutância. A seguir os cenários de conluio são analisados mais detalhadamente.

4.4.2 Análise da Abordagem para Detecção de Conluio

Nesta seção são discutidas as vantagens e limitações da detecção de conluio com a abordagem proposta na seção anterior que utiliza a métrica condutância.

Nos cenários de conluio ilustrados da Figura 4.14 a condutância dos pares maliciosos é zero pois não existem parcerias externas, o que facilita a detecção com a abordagem proposta. Mas os pares maliciosos podem adicionar parcerias externas no conluio, estabelecendo algumas parcerias legítimas com pares honestos, como ilustra a Figura 4.18, para dificultar a detecção. Dessa forma o par malicioso pode obter um número maior de parcerias externas e ainda manter a quantidade suficiente de parcerias internas para ultrapassar os limiares de grau de saída (g_m) e condutância (c_m) mais facilmente. Configurações de conluio explorando essa possibilidade foram simuladas tendo como base as redes sobrepostas dos experimentos com SopCast, como é descrito a seguir.

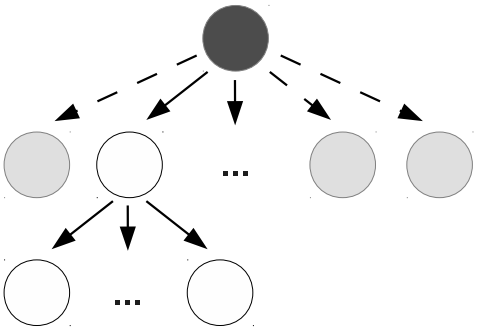


Figura 4.18: Configuração de conluio onde um par malicioso (nodo escuro) realiza algumas parcerias verdadeiras (arestas contínuas) com pares honestos (nodos brancos) e algumas parcerias falsas (arestas tracejadas) com pares auxiliares (nodos claros)

As configurações de conluio foram simuladas da seguinte forma. Primeiramente foram definidos valores de k e θ para obter os limiares mínimos g_m e c_m, tal que o grau de saída legítimo l (proveniente de parcerias legítimas) dos pares maliciosos variou de 0 até g_m+1. Em cada janela t, contendo um conjunto de pares V, foram selecionados dois pares com grau de saída igual a l. Esses pares constituíram o conjunto M de pares maliciosos atuando em cada janela t e o seu número foi restrito a dois porque os experimentos têm um número limitado de pares, podendo existir poucos deles para

alguns valores de grau de saída entre 0 e g_m+1. Um conjunto A de pares auxiliares foi incluído na rede e obteve-se um conjunto final de pares em conluio $C = \{M, A\}$. A seguir, foram criadas parcerias falsas tal que o grau de saída de cada par malicioso $i \in M$ é dado por $g(i,t) = g_m + 1$, ou seja, um grau de saída suficientemente maior que o limiar g_m. Das parcerias que somam o grau de saída do par i, existem l parcerias legítimas com pares do conjunto V e $g(i,t) - l$ parcerias falsas com pares do conjunto A. Observa-se que não foram criadas parcerias mútuas (internas) entre os pares em M ou entre os pares em A para dificultar a detecção do conluio.

O gráfico da Figura 4.19 mostra a porcentagem de detecção (linhas contínuas) e também a média do nível de cooperação (linhas pontilhadas) dos pares maliciosos em todas as janelas de tempo à medida que l é incrementado. Para cada média do nível de cooperação (NC) foi computado o seu erro com 95% de confiança. Esses resultados foram obtidos a partir de simulações com valores de $k = \{4\%, 7\%, 10\%\}$ pares com o maior grau de saída da rede e uma confiança $\theta = 99\%$. Esses valores correspondem aos graus de saída mínimos $g_m = \{26, 18, 14\}$ e condutância mínima $c_m = \{0.47, 0.43, 0.39\}$ para cada grupo k, respectivamente. Para facilitar a comparação das simulações para os três valores de g_m, o grau de saída legítimo (l) foi expresso em porcentagem referente ao grau de saída alvo do par malicioso que é $g_m + 1$.

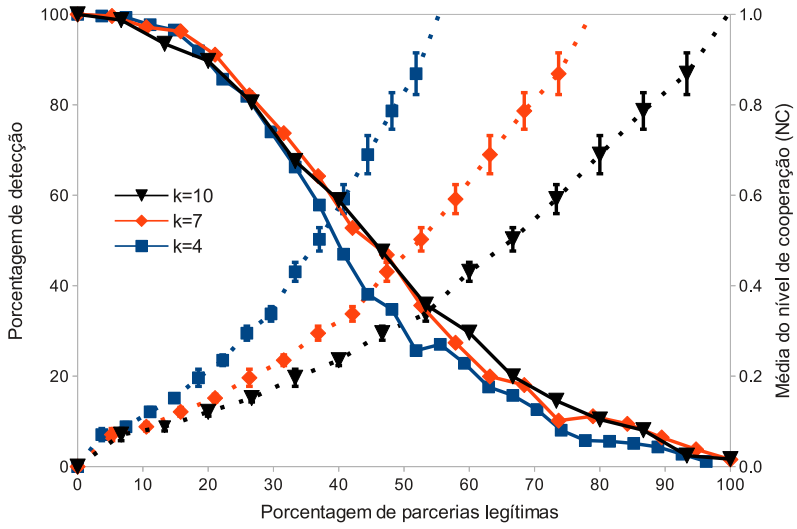


Figura 4.19: Porcentagem de detecção (linhas contínuas) e média do nível de cooperação (NC) dos pares maliciosos com uma confiança de 95% (linhas pontilhadas).

Na Figura 4.19 pode ser observado que a porcentagem de detecção diminui à medida que os pares maliciosos realizam parcerias legítimas, como é esperado. O aspecto positivo a ser destacado é que a abordagem proposta impõe que os pares

maliciosos realizem alguma cooperação para serem considerados no grupo k de pares com o maior grau de saída. Então, para diminuir as chances de detecção esses pares precisam realizar um número maior de parcerias legítimas, o que aumenta os seus níveis de cooperação, tornando o ataque menos nocivo.

Nesse sentido, a porcentagem de detecção dos pares maliciosos, mostrada na Figura 4.19, pode ser considerada eficiente, pois ela é alta quando esses pares têm poucas parcerias legítimas e, consequentemente, um nível de cooperação muito baixo. Por exemplo, cerca de 80% dos pares maliciosos são detectados quando eles realizam cerca de 25% de parcerias legítimas ($NC < 0.3$) para dissimular as demais parcerias falsas. Por outro lado, a detecção diminui quando os pares maliciosos passam a ser cooperativos ($NC \geq 1$). Mesmo assim, a detecção ainda pode ser considerada eficiente, especialmente para valores menores de k, onde os pares maliciosos precisam de um número maior de parcerias legítimas para ultrapassar os limiares g_m e c_m. Por exemplo, os pares maliciosos que buscam ser confundidos no grupo $k = 7$ passam a ser cooperativos a partir de 84% de cooperações legítimas, nesse ponto, a porcentagem de detecção desses pares alcança 9%. Quando esses pares buscam ser confundidos no grupo $k = 4$, eles se tornam cooperativos a partir de 58% de cooperações legítimas, nesse ponto, a porcentagem de detecção ainda alcança 23%.

Os pares maliciosos em conluio ainda podem assumir uma configuração estratégica, como mostra a Figura 4.20, para serem confundidos com os k% pares com o maior grau de saída na rede. Nessa configuração um ou mais pares beneficiados com o conluio (cor escura) precisam utilizar pares auxiliares (cor clara), i.e., pares que não são beneficiados. Nesse caso, o número de parcerias internas é o grau de saída do par beneficiado enquanto os pares auxiliares são divididos em dois grupos para se obter parcerias externas ao grupo $S(i)$. Essa configuração é mais difícil de ser detectada e uma possibilidade para isso seria diminuir a confiança (θ) para aumentar c_m. Assim os pares beneficiados precisam utilizar mais pares auxiliares no conluio para obter condutância maior que c_m. Essa medida pode desestimular a articulação de um conluio, todavia ela aumenta a detecção de pares honestos, i.e., falsos positivos. Assim, o valor de θ deve ser configurado de acordo o melhor compromisso para o sistema entre uma porcentagem alta de detecção e uma porcentagem baixa de falsos positivos.

Uma outra abordagem que poderia ser utilizada para lidar com tais configurações estratégicas seria monitorar as parcerias dos pares em janelas de tempos sucessivas e aqueles que têm o mesmo conjunto de parceiros com muita frequência, podem ser considerados suspeitos de conluio, como proposto em [Guerraoui et al., 2010]. Ainda, uma outra abordagem seria restringir a seleção de parcerias dos pares, como é realizado em [Li et al., 2008b]. Embora essas abordagens ajudam a combater conluio, elas

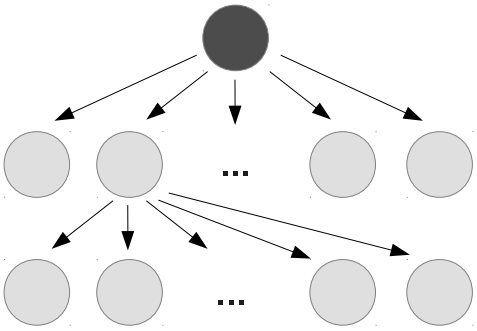


Figura 4.20: Configuração estratégica de conluio entre um par malicioso beneficiado (nodo escuro) e pares maliciosos auxiliares (nodos claros) para aumentar a quantidade de parcerias externas

impedem a formação de parcerias entre pares no mesmo ISP ou com nível de cooperação similares, o que pode impactar negativamente no desempenho dos sistemas de vídeo. No caso dos pares auxiliares serem identidades sintéticas (i.e., um ataque sybil), uma estratégia seria o uso de CAPTCHA [Ahn et al., 2003] juntamente com identificadores duráveis, por exemplo, um código enviado via uma mensagem de SMS, como proposto em [Piatek et al., 2010]. Tais métodos podem limitar a criação de identidades sintéticas.

Enfim, a abordagem apresentada não cobre todos os possíveis cenários de conluio de pares maliciosos. Assim, conluio permanece um desafio para os sistemas P2P [Ciccarelli & Cigno, 2011; Gheorghe et al., 2011], particularmente quando pares maliciosos agem estrategicamente para atacar o sistema. A intenção dessa seção foi discutir possíveis estratégias que poderiam ser adotadas para tornar o modelo de predição do nível de cooperação proposto nesta dissertação mais robusto à ação de pares maliciosos. Uma avaliação mais detalhada dessas estratégias, bem como, extensões da abordagem proposta para detecção de conluio em cenários com configurações estratégicas de pares maliciosos serão deixadas para trabalhos futuros.

Capítulo 5

Considerações Finais

Este capítulo apresenta as principais conclusões obtidas com o trabalho desenvolvido nesta dissertação e também aponta possibilidades de trabalhos futuros.

5.1 Conclusões

Nesta dissertação foi investigado se propriedades topológicas da rede sobreposta podem ser utilizadas para predizer, com precisão razoável, o nível de cooperação dos pares em um sistema P2P de distribuição de vídeo ao vivo. O trabalho desenvolvido pode ser dividido em três etapas: (1) análise da relação entre o nível de cooperação (NC) de um par e a sua centralidade na rede sobreposta; (2) desenvolvimento e avaliação de modelos de predição do nível de cooperação do par; (3) considerações sobre a aplicação do modelo em diferentes cenários incluindo questões relacionadas à segurança do sistema. A seguir é apresentado um sumário sobre as principais conclusões a cerca dessas três etapas.

A relação entre a centralidade de um par e o seu nível de cooperação (NC) foi analisada a partir da correlação entre as métricas grau, *betweenness* e proximidade com os valores de NC. Para isso foi utilizado o coeficiente de correlação de Spearman, e os valores das métricas de centralidade e NC dos pares foram computadas em janelas de tempo com uma duração predeterminada. Foi observado que grau de saída é a métrica mais correlacionada, apresentando uma correlação de Spearman maior que 0,8 em 90% das janelas de tempo de todos os experimentos realizados. As métricas proximidade de saída e *betweenness* foram, respectivamente, a segunda e a terceira métricas mais correlacionadas. A principal conclusão que pode ser obtida a partir desses resultados é que o posicionamento do par na rede sobreposta, dado pelas métricas proximidade ou *betweenness*, provê menos informação sobre o seu nível de cooperação do que a sua

atividade local, dada pelo grau de saída.

A partir da análise de correlação, foi proposto um modelo para predizer o nível de cooperação dos pares utilizando grau de saída como entrada. Foram avaliados diferentes modelos de regressão com crescimento exponencial e polinomial, e um modelo polinomial de grau quatro foi o que apresentou resultados mais precisos. A precisão do modelo foi analisada pela distribuição dos erros absolutos da predição do nível de cooperação dos pares em todas janelas de tempo. O modelo escolhido produz predições razoavelmente precisas, dado que seus erros tendem a valores próximos de zero. Mesmo apresentando alguns erros altos, foi observado que, considerando as predições e a categoria à qual o par pertence em função do NC medido, o modelo identificou 95% de pares com o nível de cooperação menor que 1 (categoria de pares pouco cooperativos) e 80% dos pares com o nível de cooperação maior que 10 (categoria de pares muito cooperativos). Quanto à calibração, foi verificado que o modelo pode ser calibrado em uma janela de tempo e aplicado para os próximos 16 minutos sem perda significativa de precisão.

As predições do nível de cooperação dos pares dadas pelo modelo podem ser utilizadas por mecanismos de incentivo que provêem benefícios ou restrições aos pares de acordo o seu nível de cooperação. Uma questão importante para prover incentivo em redes P2P de vídeo é a verificação dos dados reportados pelo par para computar o seu nível de cooperação. Nessa dissertação foi proposto verificar parcerias durante a reconstrução da rede P2P sobreposta a ser utilizada pelo modelo de predição. Tal procedimento pode diminuir a sobrecarga introduzida por técnicas de autenticação para verificar a cooperação dos pares, e.g., recibos criptografados, como foi discutido na Seção 4.3. Essas técnicas exigem do sistema uma infraestrutura com poder computacional alto para processar milhares de recibos em uma janela de tempo.

Foi proposto ainda uma abordagem que explora o grau de agrupamento entre os pares na rede sobreposta para detectar pares maliciosos agindo em conluio com o objetivo de aumentar os seus graus de saída por meio de parcerias falsas. Para isso foi utilizada a métrica de agrupamento condutância. Essa abordagem se mostrou útil em cenários de conluio onde os pares maliciosos formam parcerias falsas entre si, ou parcerias falsas com pares maliciosos auxiliares / identidades sintéticas, ou ainda, realizam algumas parcerias legítmas para dissimular as parcerias falsas. Esse último cenário é mais díficil de ser analisado e foram utilizadas simulações baseadas nos traços do SopCast para avaliar a abordagem proposta. Foi mostrado que a abordagem detecta cerca de 80% dos pares maliciosos com até 1/3 de parcerias legítimas. A detecção diminui à medida que as parcerias legítimas aumentam, entretanto, o par malicioso tende à cooperar mais para o sistema, o que torna o conluio menos nocivo.

5.2 Trabalhos Futuros

A seguir são apontadas algumas possibilidades de trabalhos futuros que estendem a pesquisa desenvolvida nesta dissertação.

A precisão do modelo proposto pode ser melhorada explorando mais métricas de centralidade e novas metodologias para construir e avaliar modelos de regressão. Por exemplo, proximidade de saída foi a segunda métrica mais correlacionada com o nível de cooperação do par, e a métrica *betweenness* com caminhos aleatórios [Newman, 2005] pode prover mais informação sobre o nível de cooperação. Essas métricas poderiam ser utilizadas juntamente com o grau de saída para obter um modelo com maior ganho em precisão. Nesse caso, haverá mais de uma variável preditora o que levará a muitas possibilidades de modelos. Logo, é necessário aplicar outras técnicas para auxiliar a busca por modelos mais precisos mais rapidamente. Para isso, podem ser utilizadas técnicas de aprendizagem de máquina voltadas para modelos de regressão como SVR (Support Vector Regression) [Smola & Schölkopf, 2004].

Além de buscar um modelo mais preciso, outro trabalho que pode ser explorado é o desenvolvimento de estratégias de amostragem de pares para o treino (calibração) desse modelo. Isso consiste em determinar a quantidade de pares suficientes, na rede em geral ou por categorias de NC, para calibrar o modelo e manter sua precisão e tempo de utilização equivalentes à calibração com a rede completa. Tal procedimento implicaria em reduzir os custos no *tracker* com largura de banda para coletar dados dos pares. No entanto, se o modelo utilizar métricas computadas com a rede completa, como *betweenness* e proximidade, seria necessário realizar aproximações dos valores dessas métricas. A aproximação também implica em reduzir custos com processamento, e o método proposto por Gkorou et al. [2011] para computar *betweenness* aproximado de um par em redes P2P poderia ser utilizado. Também, poderiam ser explorados os algoritmos desenvolvidos por Lim et al. [2011] para amostrar e estimar os k pares mais centrais de uma rede.

Uma outra questão importante a ser tratada, que não foi possível realizar nesse trabalho por limitação de tempo, é investigar se propriedades das redes sobrepostas em outros sistemas P2P de vídeo ao vivo se comportam como no sistema SopCast e também podem ser utilizadas para predizer o nível de cooperação dos pares. Infelizmente, dentre os sistemas mais populares, apenas SopCast disponibiliza um cliente para sistema Linux, o que possibilita realizar experimentos em larga escala utilizando o ambiente Planetlab. Contudo, existem projetos de *software* livre para o desenvolvimento de plataformas de distribuição de vídeo ao vivo, e.g., Napa-Wine e Goalbit[1]. Esses

[1] `http://www.napa-wine.eu`, `http://goalbit.sourceforge.net`.

projetos estão em estágio bem avançado e oferecerem códigos fontes de suas aplicações para sistema Linux. Além disso, eles oferecem uma série de ferramentas para o desenvolvimento e análise de sistemas P2P de vídeo. Um sistema de código fonte aberto permite, além de investigar as propriedades da rede sobreposta, implementar e validar o modelo de predição. Outro ponto importante é que pode ser analisado o impacto do retreino no modelo caso sejam realizadas modificações na topologia. Por exemplo, alteração da posição de um par na rede sobreposta para ficar mais próximo do servidor de vídeo como forma de incentivo.

Nesta dissertação o foco foi voltado para vídeo ao vivo e arquitetura P2P, dado o sucesso e a popularidade de sistemas como SopCast, PPLive, UUSee e outros. Contudo, o estudo de propriedades topológicas da rede também abre espaço para trabalhos com sistemas de vídeo sob demanda (VoD). Dado que alguns sistemas VoD funcionam conjuntamente com uma rede social, e.g., Youtube, a relação entre essa rede e a rede sobreposta de um possível sistema P2P VoD pode ser explorada em trabalhos futuros. Uma possível hipótese a ser investigada nesses sistemas é se um usuário com centralidade alta na rede social seria um bom candidato para compartilhar ou servir como um *cache* de *chunks* na rede P2P. Isso porque tal usuário possivelmente tem influência sob os demais (importância relativa na rede) e uma atividade alta no sistema. Logo, os vídeos que ele assiste podem estar sendo assistidos por um número maior de usuários simultaneamente. Enfim, sistemas P2P VoD associados a redes sociais abre espaço para muitas hipóteses interessantes de serem investigadas.

Referências Bibliográficas

Adar, E. & Huberman, B. (2000). Free Riding on Gnutella. *First Monday*, 5(10).

Ahn, L. V.; Blum, M.; Hopper, N. J. & Langford, J. (2003). CAPTCHA: Using Hard AI Problems for Security. In *Proc. of International Conference on Theory and Applications of Cryptographic Techniques (EUROCRYPT'03)*, pp. 294--311.

Azzedin, F. (2010). Trust-based taxonomy for free riders in distributed multimedia systems. In *Proc. of the International Conference on High Performance Computing and Simulation*, HPCS'10, pp. 362--369, Caen, France.

Boccaletti, S.; Latora, V.; Moreno, Y.; Chavez, M. & Hwang, D.-U. (2006). Complex networks: Structure and dynamics. *Physics Reports*, 424(4–5):175--308.

Borges, A.; Almeida, J. & Campos, S. (2008). Fighting pollution in p2p live streaming systems. In *Proc. of the IEEE International Conference on Multimedia and Expo*, pp. 481--484, Hannover, Germany.

Borges, A.; Gomes, P.; Nacif, J.; Mantini, R.; Almeida, J. M. & Campos, S. (2012). Characterizing sopcast client behavior. *Computer Communications*, 35(8):1004--1016.

Brandes, U. (2001). A Faster Algorithm for Betweenness Centrality. *Journal of Mathematical Sociology*, 25.

Castro, M.; Druschel, P.; Kermarrec, A.-M.; Nandi, A.; Rowstron, A. & Singh, A. (2003). Splitstream: high-bandwidth multicast in cooperative environments. In *Proc. of the nineteenth ACM symposium on Operating systems principles*, SOSP '03, pp. 298--313, Bolton Landing, NY, USA. ACM.

Chatzidrossos, I.; Dán, G. & Fodor, V. (2010). Server guaranteed cap: An incentive mechanism for maximizing streaming quality in heterogeneous overlays.

In *Proc. of 9th International IFIP TC 6 Networking Conference*, volume 6091 of *NETWORKING'10*, pp. 315--326, Chennai, India. Springer.

Chen, H.; Jin, H.; Sun, J.; Deng, D. & Liao, X. (2004). Analysis of large-scale topological properties for peer-to-peer networks. In *Proc. of the 2004 IEEE International Symposium on Cluster Computing and the Grid*, CCGRID'04, pp. 27--34. IEEE.

Chu, Y.-h.; Rao, S. G. & Zhang, H. (2000). A case for end system multicast (keynote location). *SIGMETRICS Perform. Eval. Rev.*, 28(1):1--12.

Chun, B.; Culler, D.; Roscoe, T.; Bavier, A.; Peterson, L.; Wawrzoniak, M. & Bowman, M. (2003). Planetlab: an overlay testbed for broad-coverage services. *SIGCOMM Comput. Commun. Rev.*, 33(3):3–12.

Ciccarelli, G. & Cigno, R. L. (2011). Collusion in peer-to-peer systems. In *Elsevier Computer Networks*, volume 55(15), pp. 3517–3532.

Cisco (2010). Cisco digital media systems solution overview. http://www.cisco.com.

Cohen, B. (2003). Incentives Build Robustness in BitTorrent. In *Proc. of 1st Workshop on Economics of Peer-to-Peer Systems*, P2PECON'03, Berkeley, USA.

Costa, C. & Almeida, J. (2007). Reputation systems for fighting pollution in peer-to-peer file sharing systems. In *Proc. of the Seventh IEEE International Conference on Peer-to-Peer Computing*, P2P'07, pp. 53--60. IEEE.

Douceur, J. R. (2002). The sybil attack. In *Revised Papers from the First International Workshop on Peer-to-Peer Systems*, IPTPS'01, pp. 251–260, Cambridge, USA. Springer-Verlag.

Easley, D. & Kleinberg, J. (2010). *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge Univ Press, Cambridge, UK.

Eppstein, D. & Wang, J. (2001). Fast approximation of centrality. In *Proc. of the twelfth annual ACM-SIAM symposium on Discrete algorithms*, SODA'01, pp. 228--229, Washington, USA. Society for Industrial and Applied Mathematics.

Freeman, L. C. (1978–1979). Centrality in social networks conceptual clarification. *Social Networks*, 1(3):215--239.

Gheorghe, G.; Lo Cigno, R. & Montresor, A. (2011). Security and privacy issues in p2p streaming systems: A survey. In *Peer-to-Peer Networking and Applications*, volume 4(2), pp. 75–91.

Gkorou, D.; Pouwelse, J. & Epema, D. (2011). Betweenness centrality approximations for an internet deployed p2p reputation system. In *Proc. of the IEEE International Symposium on Parallel and Distributed Processing Workshops and Phd Forum*, IPDPSW'11, pp. 1627--1634, Shanghai, China.

Gonçalves, G.; Borges, A. & Almeida, J. (2011). Centralidade em redes p2p de transmissao ao vivo. In *Proc. do 11 Workshop de Teses e Dissertações do Simpósio Brasileiro de Sistemas Multimídia e Web*, WEBMEDIA'11, Florianópolis, Brasil. SBC.

Gonçalves, G.; Guimarães, A.; Borges, A. & Almeida, J. (2012a). Predição do nível de cooperação em sistemas par-a-par de vídeo ao vivo a partir de métricas de centralidade. In *Proc. do 30 Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, SBRC'12, Ouro Preto, Brasil. SBC.

Gonçalves, G.; Guimarães, A.; Cunha, I.; Borges, A. & Almeida, J. (2012b). Using centrality metrics to predict peer cooperation in live streaming applications. In *Proc. of 11th International IFIP TC 6 Networking Conference*, volume 7290 of *NETWORKING'12*, pp. 84--96, Prague, Czech Rep. Springer.

Guerraoui, R.; Huguenin, K.; Kermarrec, A.-M.; Monod, M. & Prusty, S. (2010). Lifting: lightweight freerider-tracking in gossip. In *Proc. of the ACM/IFIP/USENIX 11th International Conference on Middleware*, Middleware'10, pp. 313--333, Bangalore, India. Springer-Verlag.

Hei, X.; Liang, C.; Liang, J.; Liu, Y. & Ross, K. (2007). A measurement study of a large-scale p2p iptv system. *IEEE Transactions on Multimedia*, 9(8):1672 –1687.

Hei, X.; Liu, Y. & Ross, K. (2008). Iptv over p2p streaming networks: the mesh-pull approach. *IEEE Communications Magazine*, 46(2):86–92.

Jain, R. (1991). *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. John Wiley & Sons, New York, USA, 1 edição.

Jin, X.; Chan, S.-H.; Yiu, W.-P.; Xiong, Y. & Zhang, Q. (2006). Detecting malicious hosts in the presence of lying hosts in peer-to-peer streaming. In *Multimedia and Expo*, pp. 1537–1540, Toronto, Canada.

Jin, X. & Chan, S.-H. G. (2010). Detecting malicious nodes in peer-to-peer streaming by peer-based monitoring. *ACM Trans. Multimedia Comput. Commun. Appl.*, 6(2):1–18.

Kannan, R.; Vempala, S. & Vetta, A. (2004). On clusterings: Good, bad and spectral. *Journal of the ACM (JACM)*, 51(3):497--515.

Karakaya, M.; Korpeoglu, I. & Ulusoy, O. (2009). Free riding in peer-to-peer networks. *IEEE Internet Computing*, 13(2):92--98.

Kendall, M. & Gibbons, J. (1975). *Rank Correlation Methods*. Griffin, London, UK, 4 edição.

Konrath, M. A.; Barcellos, M. P. & Mansilha, R. B. (2007). Attacking a swarm with a band of liars: evaluating the impact of attacks on bittorrent. In *Proc. of the Seventh IEEE International Conference on Peer-to-Peer Computing*, P2P'07, pp. 37--44, Galway, Ireland. IEEE.

Leskovec, J.; Lang, K. J.; Dasgupta, A. & Mahoney, M. W. (2008). Statistical properties of community structure in large social and information networks. In *Proc. of the 17th international conference on World Wide Web*, WWW'08, pp. 695--704, Beijing, China. ACM.

Levin, D.; LaCurts, K.; Spring, N. & Bhattacharjee, B. (2008). Bittorrent is an auction: analyzing and improving bittorrent's incentives. In *Proc. of the ACM SIGCOMM 2008 conference on Data communication*, SIGCOMM '08, pp. 243--254, Seattle, WA, USA. ACM.

Li, B.; Xie, S.; Qu, Y.; Keung, G.; Lin, C.; Liu, J. & Zhang, X. (2008a). Inside the new coolstreaming: Principles, measurements and performance implications. In *Proc. of the 27th IEEE Conference on Computer Communications*, INFOCOM'08, pp. 1031--1039, Phoenix, USA.

Li, H. C.; Clement, A.; Marchetti, M.; Kapritsos, M.; Robison, L.; Alvisi, L. & Dahlin, M. (2008b). Flightpath: obedience vs. choice in cooperative services. In *Proc. of the 8th USENIX conference on Operating systems design and implementation*, OSDI'08, pp. 355--368, San Diego, California. USENIX.

Liang, C.; Guo, Y. & Liu, Y. (2008). Is random scheduling sufficient in p2p video streaming? In *Proc. of the 28th International Conference on Distributed Computing Systems*, ICDCS '08, pp. 53--60. IEEE.

Liang, J.; Kumar, R.; Xi, Y. & Ross, K. (2005). Pollution in p2p file sharing systems. In *Proc. of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 2 of *INFOCOM'05*, pp. 1174--1185, Miami,USA.

Lim, Y.; Menasche, D.; Ribeiro, B.; Towsley, D. & Basu, P. (2011). Online estimating the k central nodes of a network. In *Proc. of the Network Science Workshop*, NSW'11, pp. 118--122, West Point, USA. IEEE.

Liu, Y.; Guo, Y. & Liang, C. (2008). A survey on peer-to-peer video streaming systems. *Peer-to-Peer Networking and Applications*, 1(1):18–28.

Liu, Z.; Hu, H.; Liu, Y.; Ross, K. W.; Wang, Y. & Mobius, M. (2010). P2p trading in social networks: the value of staying connected. In *Proc. of the 29th conference on Information communications*, INFOCOM'10, pp. 2489--2497, San Diego, USA. IEEE.

Locher, T.; Moor, P.; Schmid, S. & Wattenhofer, R. (2006). Free Riding in BitTorrent is Cheap. In *Proc. of the 5th Hot Topics in Networks*, Irvine, USA.

Magharei, N.; Rejaie, R. & Guo, Y. (2007). Mesh or multiple-tree: A comparative study of live p2p streaming approaches. In *Proc. of the 26th IEEE International Conference on Computer Communications*, INFOCOM'07, pp. 1424--1432, Anchorage, USA.

Magnetto, A.; Gaeta, R.; Grangetto, M. & Sereno, M. (2010). Turinstream: A totally push, robust, and efficient p2p video streaming architecture. *IEEE Transactions on Multimedia*, 12(8):901--914.

Meulpolder, M.; Pouwelse, J. A.; Epema, D. H. J. & Sips, H. J. (2009). Bartercast: A practical approach to prevent lazy freeriding in p2p networks. In *Proc. of the 2009 IEEE International Symposium on Parallel & Distributed Processing*, IPDPS '09, pp. 1--8, Rome, Italy. IEEE.

Myers, J. L. & Well, A. D. (1995). *Research Design & Statistical Analysis*. Routledge, 1 edição.

Newman, M. E. J. (2003). The Structure and Function of Complex Networks. *SIAM Review*, 45(2):167--256.

Newman, M. J. (2005). A measure of betweenness centrality based on random walks. *Social Networks*, 27(1):39--54.

Oliveira, J.; Gomes, P.; Borges, A. & Campos, S. (2010). Centralidade em redes p2p de transmissao ao vivo. In *Proc. do Workshop de Redes Dinâmicas e Sistemas P2P do 28 Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, SBRC'10, Gramado, Brasil. SBC.

Onnela, J.; Saramaki, J.; Hyvonen, J.; Szabo, G.; Lazer, D.; Kaski, K.; Kertesz, J. & A., B. (2007). Structure and tie strengths in mobile communication networks. In *Proc. of the National Academy of Science of the United States of America*, volume 104 of *PNAS'07*, pp. 7332–7336, USA. PNAS.

Padmanabhan, V. N.; Wang, H. J. & Chou, P. A. (2003). Resilient peer-to-peer streaming. In *Proc. of the 11th IEEE International Conference on Network Protocols*, ICNP'03, pp. 16--27, Atlanta, USA. IEEE.

Park, K. & Pai, V. S. (2006). Comon: a mostly-scalable monitoring system for planetlab. *SIGOPS Oper. Syst. Rev.*, 40(1):65–74.

Piatek, M. & Krishnamurthy, A. (2010). Improving the Performance and Robustness of P2P Live Streaming with Contracts. *USENIX;login*, 35(4).

Piatek, M.; Krishnamurthy, A.; Venkataramani, A.; Yang, R.; Zhang, D. & Jaffe, A. (2010). Contracts: practical contribution incentives for p2p live streaming. In *Proc. of the 7th USENIX conference on Networked systems design and implementation*, NSDI'10, pp. 6--6, San Jose, USA. USENIX.

Picconi, F. & Massoulié, L. (2008). Is there a future for mesh-based live video streaming? In *Proc. of the 2008 Eighth International Conference on Peer-to-Peer Computing*, P2P '08, pp. 289--298. IEEE.

Sentinelli, A.; Marfia, G.; Gerla, M.; Kleinrock, L. & Tewari, S. (2007). Will iptv ride the peer-to-peer stream? [peer-to-peer multimedia streaming]. *IEEE Communications Magazine*, 45(6):86--92.

Shen, Z.; Luo, J.; Zimmermann, R. & Vasilakos, A. (2011). Peer-to-peer media streaming: Insights and new developments. *Proc. of the IEEE*, 99(12):2089 –2109.

Shen, Z. & Zimmermann, R. (2009). Isp-friendly peer selection in p2p networks. In *Proc. of the 17th ACM international conference on Multimedia*, MM '09, pp. 869–872, Beijing, China. ACM.

Silverston, T.; Fourmaux, O.; Botta, A.; Dainotti, A.; Pescapé, A.; Ventre, G. & Salamatian, K. (2009). Traffic analysis of peer-to-peer iptv communities. *Computer Networks*, 53(4):470–484.

Silverston, T.; Fourmaux, O. & Crowcroft, J. (2008). Towards an incentive mechanism for peer-to-peer multimedia live streaming systems. In *Proc. of the 8th International Conference on Peer-to-Peer Computing*, P2P'08, pp. 125 –128, Aachen, Germany.

Sirivianos, M.; Han, J.; Rex, P. & Yang, C. X. (2007). Free-riding in BitTorrent Networks with the Large View Exploit. In *Proc. of 6th International Workshop on Peer-to-Peer Systems*, IPTPS'07, Bellevue, USA.

Smola, A. J. & Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222.

Stutzbach, D. & Rejaie, R. (2006). Understanding churn in peer-to-peer networks. In *Proc. of the 6th ACM SIGCOMM conference on Internet measurement*, IMC '06, pp. 189–202, Rio de Janeriro, Brazil. ACM.

Stutzbach, D.; Rejaie, R. & Sen, S. (2008). Characterizing unstructured overlay topologies in modern p2p file-sharing systems. *IEEE/ACM Transactions on Networking*, 16(2):267--280.

Tang, S.; Lu, Y.; Hernández, J. M.; Kuipers, F. & Mieghem, P. (2009). Topology dynamics in a p2ptv network. In *Proc. of the 8th International IFIP-TC 6 Networking Conference*, NETWORKING'09, pp. 326--337, Aachen, Germany. Springer-Verlag.

Tran, D. A.; Hua, K. A. & Do, T. T. (2004). A peer-to-peer architecture for media streaming. *IEEE Journal on Selected Areas in Communications*, 22(1):121–133.

Ullah, I.; Doyen, G.; Bonnet, G. & Gaiti, D. (2011). A survey and synthesis of user behavior measurements in p2p streaming systems. *Communications Surveys Tutorials, IEEE*, PP(99):1--16.

Walsh, K. & Sirer, E. G. (2005). Fighting peer-to-peer spam and decoys with object reputation. In *Proc. of the 2005 ACM SIGCOMM workshop on Economics of peer-to-peer systems*, P2PECON '05, pp. 138–143, Philadelphia, Pennsylvania, USA. ACM.

Wang, Q.; Vu, L.; Nahrstedt, K. & Khurana, H. (2010). Mis: malicious nodes identification scheme in network-coding-based peer-to-peer streaming. In *Proc. of*

the 29th conference on Information communications, INFOCOM'10, pp. 296--300, San Diego, California, USA. IEEE.

Wu, C.; Li, B. & Zhao, S. (2007). Characterizing peer-to-peer streaming flows. *IEEE Journal on Selected Areas in Communications*, 25(9):1612--1626.

Wu, C.; Li, B. & Zhao, S. (2008). Exploring large-scale peer-to-peer live streaming topologies. *ACM Trans. Multimedia Comput. Commun. Appl.*, 4(3):1–23.

Xia, R. & Muppala, J. (2010). Discovering free-riders before trading: A simple approach. In *Proc. of the 16th IEEE International Conference on Parallel and Distributed Systems*, ICPADS'10, pp. 806–811, Shanghai, China.

Xie, H.; Yang, Y. R.; Krishnamurthy, A.; Liu, Y. G. & Silberschatz, A. (2008). P4p: provider portal for applications. *SIGCOMM Computer Communication Review*, 38(4):351–362.

Zhang, M.; Luo, J.-G.; Zhao, L. & Yang, S.-Q. (2005). A peer-to-peer network for live media streaming using a push-pull approach. In *Proc. of the 13th annual ACM international conference on Multimedia*, MULTIMEDIA'05, pp. 287–290, Hilton, Singapore. ACM.

Zhang, M.; Zhang, Q.; Sun, L. & Yang, S. (2007). Understanding the power of pull-based streaming protocol: Can we do better? *IEEE Journal on Selected Areas in Communications*, 25(9):1678--1694.