

UNIVERSIDADE FEDERAL DO AMAZONAS  
INSTITUTO DE COMPUTAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

JHONATHAN ARAÚJO OLIVEIRA

DISPONIBILIDADE DE CONTEÚDO EM SISTEMAS CDN  
ASSISTIDOS POR REDES P2P

Manaus/AM  
2013

JHONATHAN ARAÚJO OLIVEIRA

DISPONIBILIDADE DE CONTEÚDO EM SISTEMAS CDN  
ASSISTIDOS POR REDES P2P

Dissertação apresentada ao Programa de Pós-graduação em Informática da Universidade Federal do Amazonas como requisito parcial para obtenção do grau de Mestre em Informática.

**Área de concentração:** Redes de computadores e telecomunicações.

**Orientador:**

Prof. Dr. César Augusto Viana Melo

Manaus/AM

2013

*Dedico esta dissertação a minha mãe Maria da Conceição Araújo Oliveira.*

## AGRADECIMENTOS

Meus sinceros agradecimentos:

- a Deus, por todas as bênçãos concedidas, só tenho a lhe agradecer pai;
- aos meus pais, em especial, minha mãe Maria da Conceição Araújo Oliveira, que mesmo distante sempre me incentivou e empregou todos os esforços possíveis para que eu conseguisse chegar até o final. Obrigado por tudo mãe. te amo !!!
- ao meu orientador Prof. César Melo (o chefe), não só pela orientação, mas também pelo apoio, incentivo, confiança, constante disponibilidade e compartilhamento de conhecimento (não apenas profissional, mas também pessoal). Obrigado por tudo professor;
- aos Professores da UEA Antonio Cauper, Ângela Timóteo e Andreza Mourão que desde a graduação sempre me incentivaram a buscar novas conquistas. Vocês foram pessoas que contribuíram para que eu ingressasse ao mestrado, meu muito obrigado;
- a todos os professores e demais funcionários do ICOMP/UFAM, pelo empenho em fazer esse instituto sempre melhor;
- ao Prof. Nelson L. da Fonseca (UNICAMP) por ter participado e contribuído em nossas publicações;
- à Fundação de Amparo à Pesquisa do Estado do Amazonas e a Universidade Federal do Amazonas, é claro, pelo suporte financeiro, sem esse incentivo seria impossível me manter;
- a todos os meus colegas e amigos, com os quais passei a maior parte do tempo nesses dois anos de mestrado. Herbert, Rainer, Nilmara, Rallyson, Joyce, Caio, Éfren,

Bruno, Juan, Mayara, Odette, Moyséis, Leandro, Antonio, Elda, Janainny, Thiago, Alex, Marcio Palheta, Viviane e Alesson (só para citar alguns). Você são pessoas que merecem todo o meu respeito tecnológico ;), verdadeiros irmãos. Obrigado pela companhia e apoio de todos.

– Agradeço ao meu amigo João da Mata, com o qual sempre pude contar com a sua ajuda. Você é para mim um verdadeiro irmão. Obrigado por tudo João.

A escalabilidade e a alta demanda por recursos são os principais desafios que os provedores de conteúdo enfrentam na viabilização de aplicações multimídia baseadas em redes. No YouTube, por exemplo, um dos mais populares sistemas de distribuição de vídeo sob demanda, são enviadas 100 horas de vídeo a cada minuto aos seus servidores e mais de quatro bilhões de horas de vídeo são assistidas a cada mês. Sistemas CDN-P2P têm sido apontados como uma alternativa escalável para distribuição de conteúdo multimídia na Internet. Nesses sistemas, os pares da rede par-a-par (P2P) compartilham seus recursos, diminuindo as demandas sobre a infraestrutura da rede de distribuição de conteúdo (CDN). Por outro lado, os servidores da CDN garantem a disponibilidade de conteúdo quando as contribuições dos pares são limitadas pelo *churn*, ou quando o conteúdo for inédito aos pares da rede P2P. Contudo, sistemas CDN-P2P, por si só, não garantem a efetividade dos serviços, visto que a saída de pares que são os únicos detentores de um determinado conteúdo pode gerar congestionamento ao redor do servidor da CDN e degradar a qualidade de experiência dos usuários. Nesta dissertação investiga-se a contribuição de pares estáveis para disponibilidade de conteúdo na parte P2P de um sistema CDN-P2P concebido para distribuir vídeos similares aos distribuídos pelo YouTube. Para isso, dados reais foram coletados do site YouTube, explorando-se o potencial de usuários que acessam playlists para caracterizar a estabilidade dos pares no sistema. A suposição acerca da efetividade dos tocadores de playlists na disponibilidade de conteúdo deve-se ao maior tempo de permanência desses usuários no sistema e à possível popularidade dos conteúdos por eles compartilhados. Verificou-se que quando um número grande de pares tocadores de playlists passam longas sessões conectados, a melhoria na disponibilidade de conteúdo foi de 60%. Adicionalmente em cenários de baixa participação dos tocadores de playlists a melhoria superou 20%. Em seguida, avaliou-se de que forma políticas de formação da malha impactam o sistema de distribuição estudado quando os pares são agrupados e identificados como estáveis e comuns. Estas políticas estruturam a porção P2P do sistema através de critérios que são empregados na chegada, na manutenção e gerência das conexões dos pares, diminuindo assim as exigências sobre o servidor da CDN.

**Palavras-chave:** Disponibilidade de Conteúdo, Redes de distribuição de conteúdo, Redes par-a-par, Sistemas CDN assistidos por redes P2P, Pares estáveis, Políticas de formação da malha e Vídeo sob demanda.

Scalability and high demand for resources are the main challenges that content providers face in multimedia applications based on networks. For instance, YouTube is one of the most popular delivery systems video on demand, the users send 100 hours of video every minute to its servers and more than four billion hours of video are watched every month. The CDN-P2P systems is widely recognized as a scalable alternative for multimedia content delivery in the Internet. In these systems, the peers from a peer-to-peer network (P2P) share their resources thus reducing the demands on the network infrastructure for content delivery (CDN). Moreover, the CDN server to guarantee the availability of content when the peers contributions are limited by the churn, or when the content is is unprecedented to the peers of the P2P network. However, CDN-P2P systems alone do not guarantee the effectiveness of the services, since the peers output that are the only holders of a particular contents can generate congestion around the CDN server and degrade the quality of users experience. This dissertation investigates the contribution of stable peers for availability content on the on the P2P slice form a CDN-P2P system designed to distribute videos similar to those distributed by YouTube. In this way, the data were collected from the real YouTube Web site, exploring the potential of the users that have been access the playlists to characterize the stability of the peers in the system. The assumption about the effectiveness of playlists viewers int the availability content, due to the increased of the stay connected of these users in the system and the possible popularity of contents shared by them. It was found that when a large number of the peers players pairs of the playlists spend long sessions connected, the improvement of the availability of content was in 60%. Additionally, in scenarios of low participation of players in playlists, its was the improvement outperformed in 20%. Furthermore, we evaluated how the build policies of the mesh impact the distribution system when the peers are grouped and identified as ordinaries and stable. These policies structure the portion of the P2P system through the criterias that are employed on arrival, maintenance and management of the connections of the peers, thus reducing the demands on the CDN server.

**Keywords-** Availability of Content, Content Distribution Networks, Peer-to-Peer Networks, CDN systems assisted by P2P networks, Stable peers, and Video-on-Demand.

## LISTA DE FIGURAS

- 2.1 Soluções para distribuição de conteúdo na Internet. . . . .
- 2.2 Redirecionamento de requisições em redes CDN's. . . . .
- 2.3 Arquitetura P2P. . . . .
- 2.4 Distribuição de conteúdo baseada em malha. . . . .
- 2.5 Redes sobrepostas semânticas. . . . .
- 2.6 Arquitetura CDN-P2P. . . . .
- 3.1 Método de coleta para as coleções Top-Feed. . . . .
- 3.2 Metadados de um vídeo do Youtube. . . . .
- 3.3 Metodologia de coleta para as playlists. . . . .
- 3.4 Metadados de uma playlist do Youtube. . . . .
- 3.5 Modelando a popularidade dos vídeos das coleções Top-Feed e playlist de música esporte. . . . .
- 3.6 Modelando a popularidade dos vídeos das coleções Top-Feed e playlist das coleções de comédia, educação, entretenimento e filmes&animações . . . . .
- 4.1 Exemplo de uma playlist de vídeos do YouTube. . . . .
- 4.2 Arquitetura do sistema CDN-P2P avaliado. . . . .
- 4.3 PDF's real e sintética da popularidade dos vídeos. . . . .

4.4	Taxa de acerto de pares comuns e tocadores de playlist com três duração de sessão DOWN para a coleção de vídeos de música. . . . .
4.5	Taxa de acerto de pares comuns e tocadores de playlist com três duração de sessão DOWN para a coleção de vídeos de entretenimento. . . . .
4.6	Taxa de acerto em usuários comuns e tocadores de playlist com sessão DOWN de 24 horas. . . . .
4.7	Impacto de tocadores de playlists impacientes na disponibilidade de conteúdo (coleções de entretenimento e música). . . . .
4.8	Impacto de tocadores de playlists na disponibilidade de conteúdo requisitado por usuários comuns. . . . .
5.1	Matrizes de conectividade dos pares para as duas políticas de formação da malha. . . . .
5.2	Disponibilidade de conteúdo provida pelo tocadores de playlists aos pares comuns. . . . .
5.3	Proximidade média a fonte de conteúdo para as duas políticas de formação da malha. . . . .
5.4	Disponibilidade de conteúdo global do sistema para as duas políticas de formação da malha. . . . .
5.5	Tamanho da Vizinhança dos pares ao longo da simulação. . . . .

## LISTA DE TABELAS

- 3.1 Estatísticas das coleções Top-Feed e Playlist. . . . .
- 4.1 Configuração dos agrupamentos P2P para simulação do Sistema CDN-P2P.

# CAPÍTULO 1

## INTRODUÇÃO

Aplicações multimídia são a realidade da Internet atual. Devido ao seu alto consumo de banda e elevado número de acessos, a viabilização dessas aplicações representa um constante desafio aos provedores de conteúdos que precisam disponibilizar seus serviços de forma eficiente. Diante dessa realidade, diversos estudos (Cf. (PASSARELLA, 2012)) têm sido feitos na tentativa de prover soluções que tornem sistemas de distribuição escaláveis, com menor custo financeiro e que ofereçam boa Qualidade de Experiência (*Quality of Experience - QoE*) aos usuários.

Redes de Distribuição de Conteúdo, do inglês *Content Distribution Network* (CDNs), têm sido utilizadas para lidar com a demanda de recursos dos provedores de conteúdo multimídia por oferecer alta disponibilidade e pequeno atraso. No entanto, a escalabilidade de tais redes e o seu alto custo de implantação têm dificultado sua utilização em larga escala (HAREESH; MANJIAH, 2011).

Sistemas Par-a-Par (*Peer-to-Peer - P2P*) têm se mostrado uma abordagem promissora no provimento de serviços de vídeo em larga escala pela Internet uma vez que, por natureza, estes sistemas mostram alta escalabilidade e baixo custo de implantação. No entanto, em sistemas P2P, a disponibilidade de conteúdo está diretamente relacionada com o número de pares conectados simultaneamente, ou seja, o tamanho do sistema. Quando tais sistemas apresentam tamanho reduzido, seus usuários podem experimentar uma baixa QoE com as aplicações sofrendo longo atraso e frequente perda de continuidade.

A fim de lidar com as limitações das redes CDN e P2P, os Sistemas CDN Assis- tidos por Redes P2P (CDN-P2P) têm sido implantados para explorar as vantagens de

ambas as tecnologias. Nesses sistemas, a rede P2P apoia a tarefa de distribuição, compartilhando recursos de seus pares (i.e., capacidade de armazenamento e largura de banda), diminuindo assim as exigências sobre a infraestrutura de servidores da CDN. Por outro lado, os servidores CDN garantem a disponibilidade de conteúdo quando a contribuição dos pares é limitada por distúrbios como o *churn*, isto é, os ciclos de entrada e saída dos pares, ou quando o conteúdo for inédito dentro da rede P2P.

Entretanto, sistemas CDN-P2P, por si só, não garantem a efetiva distribuição de conteúdo. Por exemplo, a saída de pares que são os únicos detentores de um determinado conteúdo pode gerar congestionamento ao redor do servidor da CDN e degradar a Qualidade de Experiência dos usuários. Consequentemente, identificar funcionalidades que maximizem a disponibilidade de conteúdo, na parte P2P de tais sistemas, tende a reduzir a carga de trabalho submetida aos servidores da CDN.

Nesta dissertação investiga-se a contribuição de pares estáveis para disponibilidade de conteúdo na parte P2P de um sistema CDN-P2P concebido para distribuir vídeos similares aos distribuídos pelo YouTube. Para isso, dados reais foram coletados do site YouTube, explorando-se o potencial de usuários que acessam *playlists* para caracterizar a estabilidade dos pares no sistema. Essas *playlists* são coleções de vídeos criadas e mantidas pelos seus donos. A hipótese acerca da efetividade dos tocadores de *playlists* na disponibilidade de conteúdo deve-se ao maior tempo de permanência desses usuários no sistema e a possível popularidade dos conteúdos compartilhados por tais usuários.

A avaliação em torno da efetividade dos tocadores de *playlists* na disponibilidade se dá a partir de sistemas CDN-P2P com características distintas. Primeiramente, avalia-se o potencial dos tocadores de *playlists* em um sistema onde o servidor da CDN auxilia a malha P2P na descoberta de pares e na busca por conteúdo. Depois, estuda-se de que forma políticas de formação da malha P2P impactam o sistema de distribuição estudado quando os pares são agrupado como estáveis e comuns. Essas políticas atuam na construção e na gerência da malha P2P, diminuindo assim as demandas sobre o servidor da CDN.

## 1.1 Principais Contribuições

As principais contribuições deste trabalho são apresentadas a seguir, na ordem em que aparecem nesta dissertação:

- Apresentação do estado da arte de sistemas de distribuição de conteúdo na Internet, descrevendo as principais arquiteturas desses sistemas e abordando as vantagens e

desvantagens de cada uma;

- Implementação de um coletor de vídeos e *playlists* do YouTube. Esse coletor possibilitou a utilização de metadados reais no sistema de distribuição de conteúdo estudado.
- Análise do padrão de popularidade das coleções de vídeos e *playlists* do YouTube utilizadas nos experimentos numéricos. Essa análise contribui para avaliar a influência dessa popularidade na disponibilidade de conteúdo provida pelos dois agrupamentos de usuários que formam o sistema de distribuição considerado neste trabalho;
- Avaliação do impacto de tocadores de *playlists* na disponibilidade de conteúdo em sistemas CDN-P2P. Os resultados apresentados neste estudo mostram que, em alguns casos, a presença de tocadores de *playlists* no sistema melhora a disponibilidade de conteúdo em até 60%;
- Implementação e avaliação de políticas de formação e gerência da malha P2P em um sistema CDN-P2P que possui pares estáveis. Nesse estudo avalia-se o desempenho do sistema quando os pares não dependem exclusivamente de uma entidade centralizadora para construir e gerenciar as suas conexões.

## 1.2 Artigos Publicados

- MELO, C. A. V.; OLIVEIRA, J. A.; FONSECA, N. L. S. Promotion of Content Availability by Playlist Viewers in CDN-P2P Systems. In: *IEEE International Conference on Communications (ICC)*, 2013.
- OLIVEIRA, J. A.; MELO, C. A. V.; FONSECA, N. L. S. Disponibilidade de Conteúdo Promovida por Tocadores de Playlists em Sistemas CDN-P2P. in: *WP2P+ SBRC*, 2013.

## 1.3 Estrutura da Dissertação

Esta dissertação está dividida em 6 capítulos. O capítulo 2 apresenta uma revisão do estado da arte de arquiteturas de sistemas de distribuição de conteúdo na Internet. Neste Capítulo apresenta-se uma visão geral sobre os sistemas de distribuição, destacando-se as arquiteturas, CDN e P2P, e os sistemas CDN assistidos por redes P2P (CDN-P2P).

O Capítulo 3 apresenta o estudo das coleções de vídeos e *playlists* do YouTube utilizadas nos experimentos numéricos realizados nesta dissertação. Analisa-se o padrão de popularidade de tais coleções, e na forma como essa popularidade influencia os usuários do YouTube na montagem de suas *playlists*. O estudo desses dados permite identificar estatísticas importantes que podem influenciar na disponibilidade de conteúdo provida pelos dois tipos de usuários que compõem o sistema de distribuição estudado.

No Capítulo 4 é avaliado o impacto da presença de pares estáveis (tocadores de *playlists*) na disponibilidade de conteúdo em sistemas CDN-P2P. As principais contribuições apresentadas nesse Capítulo são duas. Primeiro, avalia-se a contribuição para disponibilidade de conteúdo provida pelos tocadores de *playlists* em vários cenários de rede. Depois avalia-se o quanto os tocadores de *playlists* respondem a requisições geradas por pares comuns.

No Capítulo 5 avalia-se de que forma políticas de formação e gerência da malha influenciam o sistema de distribuição estudado quando os pares são agrupados como estáveis e comuns. Estas políticas utilizam diferentes critérios para estruturarem a parte P2P do sistema, minimizando a demanda sobre os servidores da CDN.

Finalmente, no Capítulo 6 são apresentados comentários gerais sobre o estudo realizado, resultados obtidos e as conclusões sobre os resultados. Além disso, são apresentadas as limitações e possíveis extensões desta pesquisa.

## CAPÍTULO 2

# DISTRIBUIÇÃO DE CONTEÚDO NA INTERNET

As novas tecnologias de rede de acesso têm impulsionado a geração e o compartilhamento de conteúdos na Internet por parte dos usuários. Pesquisas revelam (GEBERT et al., 2012) (SANDVINE, 2011) que uma parte significativa do tráfego gerado na Internet atual provém de aplicações onde conteúdos (ou parte deles), em sua maioria, são gerados e mantidos pelos próprios usuários. No YouTube, por exemplo, um dos mais populares sistemas de distribuição de vídeo sob demanda (*Video-on-Demand (VoD)*), são enviadas 100 horas de vídeo a cada minuto aos seus servidores e mais de quatro bilhões de horas de vídeo são assistidas a cada mês (YOUTUBE, 2013). Esse novo contexto apresenta desafios de escalabilidade aos sistemas de distribuição de conteúdo, exigindo que novas soluções sejam implantadas de forma a minimizar o impacto sobre as infraestruturas de rede e oferecer boa Qualidade de Experiência (*Quality of experience - QoE*) aos usuários finais.



Figura 2.1: Soluções para distribuição de conteúdo na Internet.

A Figura 2.1 ilustra as principais soluções que têm sido propostas para distribuição de conteúdo na Internet. Essas soluções incluem as Redes de Distribuição de Conteúdo (*Content Delivery Networks - CDNs*), as Redes Par-a-Par (*Per-to-Per Networks - P2P*) e os Sistemas CDN assistidos por Redes P2P (CDN-P2P), também conhecidos como sistemas híbrido. Nas seções seguintes são abordadas as principais características de cada uma destas soluções. Na Seção 2.1 apresenta-se o conceito de redes de distribuição de conteúdo. Na Seção 2.2 descreve-se sobre as redes Par-a-Par e na Seção 2.3 discute-se os sistemas CDN assistidos por redes Par-a-Par (CDN-P2P). Por fim, na Seção 2.4 apresentam-se as considerações finais deste capítulo.

## 2.1 Redes de Distribuição de Conteúdo (CDNs)

Redes de Distribuição de Conteúdo (PATHAN et al., 2008), do inglês *Content Distribution Networks* (CDNs), consistem em um conjunto de servidores distribuídos, interconectados pela Internet, que atuam de forma cooperativa para a distribuição de conteúdo. Tais redes (VAKALI; PALLIS, 2003) caracterizam-se por empregar replicação de conteúdos como forma de reduzir a carga de trabalho do servidor principal, melhorar a acessibilidade e oferecer boa QoE aos usuários finais.

Provedores CDN, mantêm um conjunto de servidores substitutos (ou réplicas) em diferentes localizações geográficas (PASSARELLA, 2012). A ideia é replicar conteúdos do servidor principal nos servidores substitutos, de modo que requisições por conteúdos sejam redirecionadas para algum servidor (principal ou substituto) localizado próximo ao cliente requisitante. A Figura 2.2 ilustra a dinâmica geral de requisição em uma CDN. Quando um usuário requisita algum conteúdo, sua requisição é direcionada ao servidor principal. Uma vez recebida a requisição de um conteúdo, o servidor principal é responsável de encaminhar essa requisição para o servidor substituto mais próximo do cliente.

Empresas como Akamai (AKAMAI, 2013), Edgecast (EDGECAST, 2013) e Level3 (LEVEL3, 2013) especializaram-se em prover e comercializar arquiteturas CDNs para empresas que precisam distribuir seus conteúdos, como o Youtube.

Segundo Passarella (2012), o serviço de distribuição de conteúdo em uma CDN precisa lidar com três questões fundamentais: identificar a melhor localização para posicionamento dos servidores substitutos (1); identificar que conteúdo replicar (2); identificar como replicar o conteúdo (3).

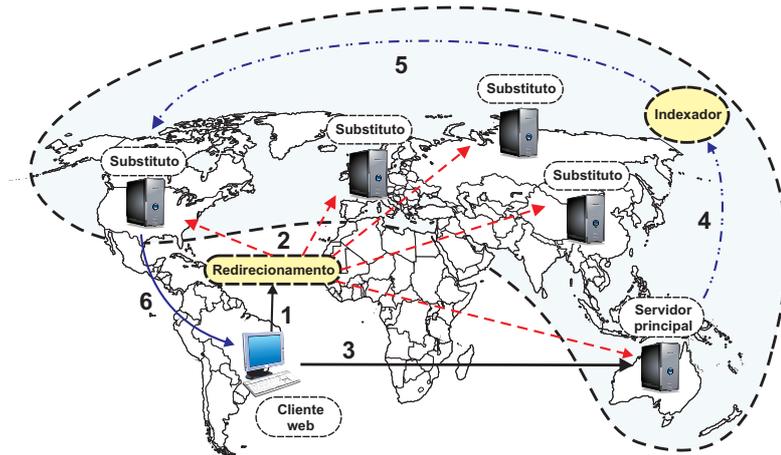


Figura 2.2: Redirecionamento de requisições em redes CDN's.

### 2.1.1 Posicionamento de Servidores CDN

O posicionamento (PASSARELLA, 2012) de servidores CDNs em locais estratégicos é um problema de otimização pertencente a classe dos problemas NP-Difíceis. A seguir, são apresentados trabalhos que propõem heurísticas para esse problema.

Li et al. (1999) abordam o problema de posicionamento de servidores CDN como um problema de programação dinâmica. Os autores apresentam um algoritmo que considera a topologia das redes subjacentes como árvores. Esse algoritmo, divide uma árvore  $T$  em diversas sub-árvores  $T_i$ , de modo que o melhor posicionamento dos servidores  $t > 1$  na árvore  $T$  é posicionar o servidor  $t'_i$  em cada árvore  $T_i$ , onde  $\sum_i t'_i = t$ . O critério de posicionamento utilizado pelos autores foi o nível de tráfego gerado em cada servidor.

Qiu et al. (2001) apresentam diversas heurísticas para o problema de posicionamento de servidores CDN, e avaliam seus desempenhos em uma rede com topologias sintética e real derivadas de tabelas de roteamento BPG (*Border Gateway Protocol*). Entre as heurísticas propostas, a que obteve os melhores resultados utiliza uma abordagem gulosa (*greedy Algorithm*). Nesta abordagem, a cada iteração calcula-se o custo local para os  $N$  servidores, de acordo com os acessos de clientes, para então escolher o servidor que produza um menor custo combinado com o servidor já escolhido anteriormente.

Wu e Ravindran (2009) estudam a aplicação de algoritmos genéticos para o problema de posicionamento de servidores CDN. Em essência, um algoritmo genético gera uma população de possíveis repostas para o problema tratado para depois submetê-la ao processo de evolução, constituído por etapas tais como, avaliação, seleção, cruzamento e mutação. O desempenho do algoritmo genético foi comparado com o algoritmo guloso apresentando em (QIU et al., 2001). Os resultados numéricos mostram que para um

número de servidores disponíveis maior que dois, o algoritmo genético obtém melhores resultados que o algoritmo guloso.

### 2.1.2 Replicação de Conteúdo em Redes CDNs

Esta seção apresenta algumas abordagens propostas para o problema de replicação de conteúdos em CDNs. Essas abordagens envolvem duas questões principais: *i*) Em que servidor replicar o conteúdo? *ii*) Como replicar o conteúdo?

Tenzekhti et al. (2002) propõem duas abordagens para o problema de posicionamento das réplicas, uma centralizada e outra distribuída. As duas abordagens consideram taxas calculadas com base nos custos de comunicação, processamento e na frequência de acesso às réplicas. Na abordagem centralizada, um servidor central recolhe dados e executa um algoritmo guloso de posicionamento das réplicas. Na abordagem distribuída, as decisões são tomadas localmente de acordo com as informações armazenadas em cada servidor. A abordagem distribuída obteve melhores resultados em relação a abordagem centralizada.

Wauters et al. (2005) apresentam duas heurísticas para o problema de posicionamento das réplicas. Para tais heurísticas, os autores consideraram conceitos de redes P2P, fazendo com que os servidores se comuniquem entre si, trocando informação sobre a carga e tráfego. Apenas os resultados da primeira heurística foram apresentados e comparados com outras heurísticas expostas na literatura, mostrando que a heurística proposta gerou melhores resultados.

Fujita et al. (2004) mostram uma heurística para resolver o problema da replicação completa e parcial de conteúdo. Nesta heurística, os conteúdos do servidor de origem são agregados em múltiplos grupos e a replicação é realizada por granularidade. De acordo como os grupos que se formam, a heurística classifica-os como sendo de replicação estática e de replicação dinâmica. Na replicação estática, os conteúdos são previamente agregados de acordo com alguma característica estática, como o tamanho dos objetos. Na replicação dinâmica, os conteúdos são agregados dinamicamente com base em características dinâmicas, como a carga do servidor. Tal abordagem obteve melhores resultados em relação ao esquema de grupos de replicação estática.

Spanoudakis e Hadjiefthymiades (2009) apresentam dois esquemas de replicação parcial de conteúdo que visam otimizar o espaço em disco nos servidores CDN. O primeiro esquema efetua o particionamento dos conteúdo com base na estrutura local do

site utilizando métricas locais, como a interconectividade da página *web* e a “distância” (o número de *hiperlinks* que o usuário precisa seguir a fim de ir de uma página para outra). O segundo esquema baseia-se no comportamento experimentado por um usuário ao acessar uma página, a fim de determinar a relatividade de conteúdos. Esse esquema foi modelado como processos de Markov. Os resultados numéricos mostram que os dois esquemas apresentam resultados similares.

## 2.2 Redes Par-a-Par (P2P)

Com a disseminação da Internet e da computação distribuída, soluções para distribuição de conteúdo multimídia em larga escala baseadas em redes P2P têm ganho atenção da comunidade (PASSARELLA, 2012) (VU et al., 2010) (LIU et al., 2008). Essas redes têm como principais características a robustez e a escalabilidade. A sua escalabilidade é provida através da capacidade de armazenamento disponível em cada par, e sua robustez, da replicação de conteúdo nos pares que formam a rede .

Em uma rede P2P, os recursos compartilhados são distribuídos entre os pares que a compõem, tendo esses a capacidade de servir e de consumir recursos do sistema. Esse arranjo com os pares desempenhando duplo papel contrasta com a arquitetura cliente-servidor tradicional, onde os papéis de cliente e servidor encontram-se bem definidos. A Figura 2.3 mostra a arquitetura de uma rede P2P. Nesta arquitetura, um par não somente pré-carrega dados da rede utilizando sua banda de recepção, mas também os fornece para outros usuários utilizando sua banda de transmissão.

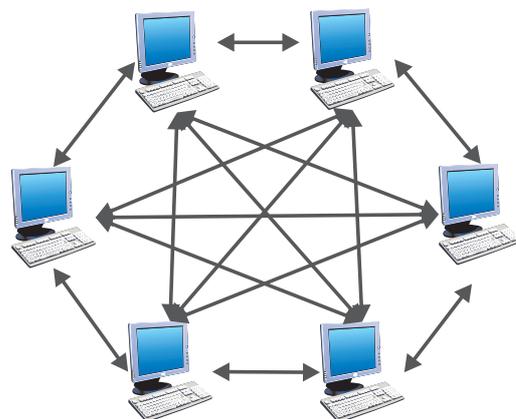


Figura 2.3: Arquitetura P2P.

Atualmente grande parte das soluções para disseminação de conteúdo (i.e., compartilhamento de arquivo, fluxo de vídeo ao vivo) que utilizam a arquitetura P2P, como o BitTorrent (BITTORRENT, 2013) e o PPLive (PPLIVE, 2013), empregam o conceito

de *swarms* (caracterizado na Seção 2.2.1). Os *swarms* são conjuntos de usuários interessados em baixar ou compartilhar um determinado conteúdo, que pode ser um arquivo ou um conjunto de arquivos. O emprego de *swarms* é uma das principais características da denominada distribuição de conteúdo baseada em malha, que é descrita na Seção 2.2.1. Além disso, na Seção 2.2.2 apresentam-se algumas abordagens de distribuição de conteúdo baseadas em semântica de conteúdo, onde são propostas derivações para a distribuição baseada em malha.

## 2.2.1 Distribuição Baseada em Malha

Nesta abordagem, os pares são organizados em uma malha, formando uma rede de sobreposição totalmente distribuída. Os conteúdos distribuídos são fragmentados em segmentos de tamanho fixo chamados blocos. Esses blocos são trocados entre os pares que formam o *swarm* em torno de um determinado conteúdo. A Figura 2.4 ilustra a arquitetura de um sistema de distribuição de conteúdo baseada em malha. Um par conecta-se e desconecta-se com seus vizinhos dinamicamente, além de poder baixar e enviar conteúdo de/para diversos pares ao mesmo tempo. Assim, quando um par abandona a rede, os demais podem continuar acessando o conteúdo dos demais pares que ainda continuam conectados, sem que haja desconexão na rede.

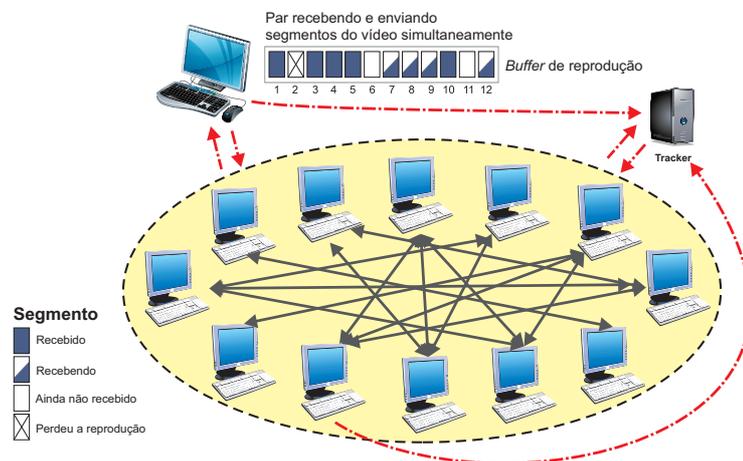


Figura 2.4: Distribuição de conteúdo baseada em malha.

Dentre os sistemas que adotam a distribuição baseada em malha destaca-se as redes do tipo BitTorrent (BITTORRENT, 2013). Nestas redes, o processo de construção e interação dos pares no *swarm* se dá a partir de três dinâmicas principais que são: a chegada do par no sistema, a manutenção das conexões, onde são empregados os mecanismos de incentivo, e o acesso aos conteúdos, onde são implementados mecanismos de escalonamento.

A seguir, estas dinâmicas são descritas considerando-se um sistema para compartilhamento de arquivos e as adaptações propostas para o contexto de vídeo sob demanda (VoD). Ressalta-se que variações em tais dinâmicas têm sido propostas para outros contextos, como no compartilhamento de fluxo de vídeo ao vivo (i.e (MONTAZERI; B.AKBARI, 2010) (HECHT; STILLER, 2010)).

### **Chegada de Pares no Sistema**

Quando um novo par chega à rede e realiza a requisição de um determinado conteúdo, esse envia ao *tracker* suas informações, isto é, endereço IP e o número de sua porta de comunicação. O *tracker* coordena o *swarm*, sendo responsável por manter uma lista com todos os pares que participam do *swarm*. Logo, quando o *tracker* recebe uma consulta sobre pares ativos no *swarm*, é devolvida uma lista de endereços de pares selecionados aleatoriamente. Em seguida, o novo par tenta estabelecer um certo número de conexões com os pares nessa lista, para então se juntar ao *swarm* e iniciar a troca de fluxo de conteúdo com seus vizinhos. Quando um par abandona à rede normalmente, esse informa a ocorrência de tal evento ao *tracker* e aos seus vizinhos, de forma que seu endereço seja removido da lista de pares ativos.

### **Mecanismos de Incentivo**

Quando um par deseja novos blocos de um determinado conteúdo, esse envia mensagens comunicando seu interesse aos vizinhos que possuem blocos que ele ainda não tem. Entretanto, antes que o par em questão comece a solicitar esses blocos, ele precisa aguardar o recebimento de uma autorização, concedida através da mensagem de desbloqueio (*unchoke*). O número de desbloqueios que um par concede simultaneamente (ou número de *uploads* simultâneos) é limitado. É necessário, portanto, que cada par escolha, dentre os vizinhos que estão interessados, aqueles para os quais irá fazer *upload*. Tal escolha se dá segundo um algoritmo de seleção de pares, onde são empregados mecanismos de incentivo.

Os mecanismos de incentivo avaliam os pares da rede quanto à sua contribuição para o sistema. Para isso, é necessário que um par selecione periodicamente um subconjunto de vizinhos para trocar blocos de conteúdos baseado em algum critério que expresse a participação dos pares na rede, tais como, as taxas de *upload* e *download* (JOUNG; HUANG, 2012a) (XIA; MUPPALA, 2010) (MENASCHE et al., 2010). No BitTorrent, as avaliações em torno da participação dos pares são implementadas pelo mecanismo de

incentivo *Tit-For-Tat* (TFT) e suas extensões. A seguir apresenta-se o *Tit-For-Tat* e algumas de suas extensões.

- *Tit-For-Tat*: Nesse mecanismo, um par prioriza pares que são capazes de lhe oferecer melhores taxas de *upload*. Em um determinado momento, o par troca blocos de conteúdo com um número fixo de vizinhos, geralmente no máximo quatro (XIA; MUPPALA, 2010), com os pares que lhe oferecem as melhores taxas de *upload*, e realiza um bloqueio (*choker*) nos outros pares. A cada 10 segundos, o par reavalia as taxas de *upload* dos outros vizinhos que estão enviando dados a ele. Se outro par lhe oferece melhor taxa de *upload* em relação ao pares com quem ele atualmente está trocando blocos de conteúdos, esse é desbloqueado (*unchoking*) e o par com menor taxa *upload* é bloqueado. Desse modo, os pares são incentivados a contribuir, aumentando a disponibilidade global de conteúdos no sistema, além de tornar o *swarm* mais resistente a pares egoístas, também conhecidos como caronas (*free riders*), que são pares que não contribuem com seus vizinhos.
- *Optimistic Unchoking*: No mecanismo TFT, os pares priorizam pares que oferecem melhores taxas de *upload*. Como isso, um par pode ter grandes chances de ficar trocando blocos sempre com os mesmos pares e não teria como descobrir outros com os quais poderia conseguir taxas de *upload* e *download* maiores. Para possibilitar a descoberta de pares potenciais, o BitTorrent incorpora um mecanismo denominado desbloqueio otimista (*optimistic unchoking*). Com esse mecanismo, o par a cada 30 segundos desbloqueia algum dos seus vizinhos de forma aleatória, independente da sua taxa de *upload*. Este mecanismo, além de propiciar a um par a descoberta de outros pares potenciais, pode também ser bastante útil para que novos pares no sistema tenham oportunidades de acesso a blocos de conteúdo.
- *Anti-snubbing*: Em algum momento, um par pode ter sido bloqueado por todos os pares com quem ele está conectado, gerando assim, um natural empobrecimento nas suas taxas *upload* e *download*. Neste caso, o par precisará de certo tempo até que o *optimistic unchoking* encontre novos pares com os quais possa trocar conteúdos. Para lidar com essa situação, o BitTorrent implementa um mecanismo denominado *anti-snubbing*. Esse mecanismo considera que quando um par fica por mais de um minuto sem receber blocos de algum vizinho, esse está sendo esnobado (*snubbed*) pelo mesmo. Nessa situação, o mecanismo *anti-snubbing* permite que o par eleja, temporariamente, mais de um par para receber dados via *optimistic unchoking*. Este mecanismo pode ajudar o par a se recuperar mais rapidamente da situação de não

ter com quem trocar conteúdo.

## Mecanismos de Escalonamento para Acesso a Conteúdos

Como descrito anteriormente, no sistema BitTorrent, os pares divulgam e recebem informações sobre a disponibilidade de blocos de conteúdos entre seus vizinhos. O conjunto de blocos disponíveis para solicitação depende da disponibilidade entre os vizinhos que lhe autorizam *download*. Uma política é empregada para escolha de quais blocos serão solicitados primeiro. Tal política (XIA; MUPPALA, 2010) deve ser eficiente do ponto de vista global do *swarm*, de forma maximizar a chance de interação entre os pares e consequentemente a utilização das suas capacidades.

Para isso, é fundamental manter um equilíbrio na distribuição de blocos, uma vez que é essencial evitar que um bloco torne-se raro a ponto de degradar o desempenho do *swarm* (JOUNG; HUANG, 2012a). Em um caso extremo, um bloco pode tornar-se extinto, ocasionando a falha do *swarm*. Por outro lado, é necessário priorizar a conclusão de um *download*, em detrimento ao *download* de um novo bloco. A seguir, são apresentadas algumas políticas de escalonamento para acesso a conteúdos utilizadas no BitTorrent.

- *Strict Priority*: No BitTorrent, os pares se concentram no *download* de um bloco em andamento, antes de solicitar outro bloco. Assim, se um sub-bloco é requisitado, os sub-blocos subsequentes do mesmo bloco serão solicitados preferencialmente. O objetivo é aumentar rapidamente a quantidade de blocos disponíveis e consequentemente as chances de o par torna-se mais interessante aos seus vizinhos, ou seja, com blocos que eles não possuem.
- *Rarest First*: Pares muitas vezes preferem baixar blocos que são mais raros entre os seus vizinhos, pois em tese esses pedaços são mais difíceis de se obter. Essa estratégia é chamada de *Rarest First* e funciona como segue. Cada par mantém uma lista de blocos que cada um dos seus vizinhos possuem. Tal lista é atualizada a cada vez que um bloco torna-se disponível a partir dos seus vizinhos. Com base nesta lista, o par constrói um conjunto de blocos raros entre os seus vizinhos, para então requisitar o *download* do bloco mais raro entre os seus vizinhos. Obter o bloco mais raro, naturalmente ajuda a aumentar a sua disponibilidade.
- *Random First Piece*: No mecanismo *rarest first* existe uma exceção para que um par solicite um bloco mais raro. Tal exceção se dá quando um par se junta pela primeira vez no *swarm* e não possui um limiar de blocos, geralmente 4. Nesta situação, o par

não possui informação completa da sua vizinhança e por isso não tem como montar um mapa da disponibilidade de blocos. Além disso, o par não possui blocos para retribuir no mecanismo TFT, o que aumenta a necessidade de completar o quanto antes seus primeiros blocos. Por isso, o Bittorrent aplica o mecanismo *random first piece*, para que o par obtenha seus primeiros blocos de maneira aleatória. O mecanismo é aplicado até o par obter o limiar de blocos definido.

- *Endgame Mode*: Esse mecanismo é adotado na fase final do *download* de um par. Se um bloco é requisitado a um par que naquele momento possui baixa taxa de transferência, isso conseqüentemente pode gerar um prolongamento do *download*. Para lidar com esse problema, o par realiza uma solicitação de forma redundante para todos os seus vizinhos para blocos que ele ainda não recebeu. Uma vez que um bloco é obtido, o par cancela as requisições feitas para esse bloco aos seus vizinhos, de modo a minimizar o desperdício de largura de banda devido a *downloads* redundantes.
- *Window-based*: Nesta abordagem é empregada uma janela deslizante dentro da qual os blocos são escolhidos. A janela avança desde o início até o final do conteúdo de acordo com a sequencialidade e o progresso do *download*. Normalmente, a janela começa com o primeiro bloco ainda não baixado. Dentro da janela (ACUNTO et al., 2013) é aplicado o mecanismo *Rarest First*, como o objetivo de equilibrar o *download* na ordem de reprodução e a necessidade de manter a diversidade de blocos. Naturalmente, uma janela menor possibilita a reprodução mais próxima da sequencialidade dos blocos, mas pode reduzir a diversidade e, conseqüentemente, as oportunidades de interação entre os pares. Por outro lado, uma janela maior permite mais possibilidades de interação entre os pares, porém aumenta as chances de atrasos de inicialização e a oportunidade de baixar blocos antes da sua reprodução. Assim sendo, o tamanho da janela (em blocos) representa o parâmetro de sequencialidade desta política.
- *Probabilistic-based*: Na seleção de blocos probabilística, os blocos são escolhidos segundo uma distribuição de probabilidade, geralmente com um viés para os primeiros blocos, ou seja, com um peso maior para tais blocos. Especificamente em (CARLSSON et al., 2009), onde é utilizada a distribuição de probabilidade Zipf, a probabilidade de um par selecionar um bloco para baixar é proporcional a  $(k + 1 - k_0)^\theta$ , onde  $k_0$  é o índice do primeiro bloco que o par ainda não baixou. Semelhante ao tamanho da janela da política *Window-based*,  $\theta$  representa o parâmetro de sequencialidade desta política.

As políticas de seleção de blocos *window-based* e *probabilistic-based* são derivações específicas para sistemas de vídeo sob demanda, visto que nesses sistemas algumas características devem ser consideradas tais como, a interação do usuário (ie., pausar, avançar e retroceder) durante a reprodução do vídeo .

## 2.2.2 Distribuição Baseada em Semântica de Conteúdo

Na distribuição de conteúdo baseada em malha o processo de construção da vizinhança dos pares ocorre de maneira aleatória, ou seja, não há regras relacionadas a formação da malha. Este processo pode isolar pares que possuem interesses similares e afetar a eficácia das consultas, uma vez que essas possuem um tempo de vida útil (*Time to live - TTL*). Diante disso, algumas abordagens têm sido propostas no intuito de explorar o “perfil” dos conteúdos que são acessados e compartilhados pelos pares. No que segue, algumas destas abordagens são descritas.

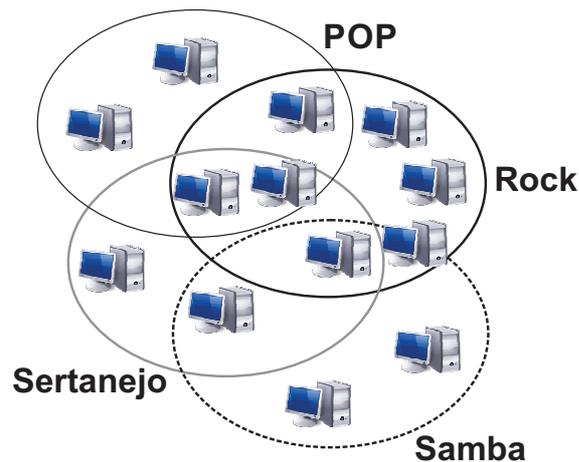


Figura 2.5: Redes sobrepostas semânticas.

Crespo e Garcia-Molina (2005) apresentam um esquema para a criação de redes sobrepostas para compartilhamento de músicas, onde comunidades se formam de acordo com os estilos musicais. No esquema proposto, sempre que um par chega a rede, esse se conecta aleatoriamente com algum par disponível. Quando o par requisita algum conteúdo, ele envia uma solicitação para algum dos seus vizinhos com quem ele está conectado. O vizinho que recebeu a solicitação, por sua vez, devolve uma lista de pares interessados no mesmo estilo musical do conteúdo que o par está requisitando para que esse tente se conectar com algum dos pares da lista. Os pares que formam a comunidade compartilham conteúdos que são de interesse da comunidade a qual eles pertencem, e também podem participar de mais de uma comunidade, como ilustrado na Figura 2.5. Os autores compararam o esquema proposto com uma formação de rede sobreposta totalmente ale-

at6ria, utilizada na 6poca dos experimentos pelo sistema Gnutella (GNUTELLA, 2013). Os resultados mostraram que abordagem proposta reduz significativamente os custos de comunica76o em rela76o 6 forma76o aleat6ria.

Anwar et al. (2005) avaliam uma rede de sobreposi76o baseada nos padr6es de similaridades existentes entre amigos da rede social Orkut. Os autores utilizaram o conceito de DHT (*Distributed Hash Table*) para cria76o de hiperlinks entre os usu6rios. Em uma DHT, as chaves (informa76es) s6o distribu6das entre os n6s participantes, de forma que as buscas possam ser roteadas a partir das informa76es armazenadas em cada par. Os resultados indicam que a lat6ncia na comunica76o em grupo 6 reduzida em compara76o com outras t6cnicas de *multicast*.

Mekouar et al. (2010) estudam o impacto de fun76es de similaridade em um esquema de recomenda76o de conte6do em uma rede P2P parcialmente descentralizada. Neste esquema, sempre que um par requisita um conte6do, esse dirige sua requis76o a um super-par (definido com base no tamanho da cache dos pares), informando o conte6do requisitado e o conjunto de conte6dos que o par j6 possui em sua cache. Em seguida, o super-par comunica-se com outros super-pares e recebe destes uma lista de pares que possuem o conte6do requisitado juntamente com os outros conte6dos pertencentes a cada um pares. De posse dessas informa76es, o super-par aplica uma fun76o de similaridade para ent6o indicar pares que possuem o conte6do que foi requisitado. Os resultados mostram que as m6tricas de similaridade que n6o consideram ocorr6ncia negativa (conte6dos n6o pertencentes a nenhum dos pares) apresentam os melhores resultados.

## 2.3 Sistemas CDN Assistidos por Redes P2P (CDN-P2P)

Aplica76es baseadas em redes CDN e P2P tem se mostrado promissoras no provimento de servi76os de v6deo em larga escala pela Internet. No entanto, o alto custo de implanta76o e a escalabilidade das CDNs t6m dificultado seu uso em larga escala. Por outro lado, aplica76es baseadas em redes P2P requerem um n6mero significativo de pares para alavancar o processo de distribu76o. Em compara76o com servidores CDN, um par geralmente oferece banda de *upload* muito menor, exigindo, portanto, v6rios pares para transmitir com boa QoE uma m6dia a um par solicitante. Sendo assim, nenhuma das arquiteturas (CDN e P2P), por si s6, fornece uma solu76o eficiente e escal6vel para distribu76o de fluxo de m6dia na Internet (JIANG et al., 2008).

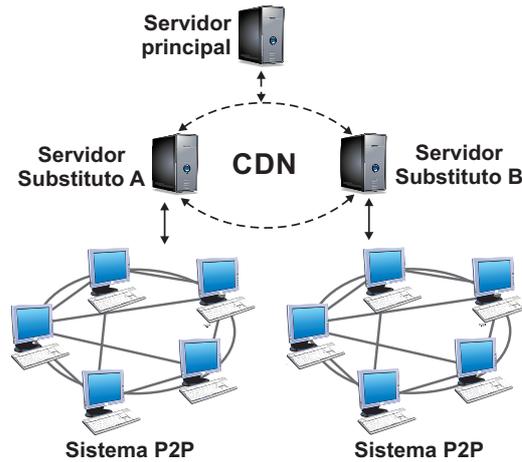


Figura 2.6: Arquitetura CDN-P2P.

Sistemas CDN assistidos por redes P2P (CDN-P2P) surgem como forma de oportunizar as vantagens de ambos, CDN e P2P (PASSARELLA, 2012) (YIN et al., 2009) (HUANG et al., 2008). A Figura 2.6 ilustra a arquitetura CDN-P2P. Nessa arquitetura as duas tecnologias se complementam. Quando um conteúdo precisa ser distribuído para uma comunidade de pares, o arquivo é primeiramente distribuído para o servidor CDN e uma porção da capacidade de armazenamento do servidor é reservada para esse arquivo.

Quando um par requisita o arquivo, esse se torna uma semente. Juntos, o servidor e os pares sementes servem o fluxo de mídia com maior capacidade do que o servidor sozinho. Quando o sistema P2P possui uma quantidade de pares sementes que conseguem atender aos pedidos a determinado arquivo, esses assumem as requisições ao arquivo e, assim, diminuem a carga de trabalho que chega ao servidor. Quando a aplicação P2P assume as requisições a esse arquivo, a capacidade de armazenamento ocupada pelo arquivo no servidor CDN é liberada, economizando assim custo para o provedor.

## 2.4 Considerações Finais do Capítulo

Este Capítulo apresentou as principais abordagens para distribuição de conteúdos na Internet. Foram discutidas as redes CDNs, as redes P2P e os sistemas CDN assistidos por redes P2P (CDN-P2P).

Os sistemas CDN-P2P têm sido apontados como promissores na distribuição de conteúdo multimídia em larga escala, uma vez que tais sistemas capitalizam vantagens de ambos, CDN e P2P. No entanto, a arquitetura CDN-P2P, por si só, não garante a distribuição efetiva de conteúdo, uma vez que a saída dos pares na parte P2P do sistema, que são os únicos detentores de determinados conteúdos, pode gerar congestionamento ao

redor do servidor da CDN, fazendo com que usuários experimentem baixa Qualidade de Experiência.

Desta maneira, identificar funcionalidades que possam maximizar a disponibilidade de conteúdo na parte P2P do sistema ainda é um problema a ser estudado. Os Capítulos 4 e 5 desta dissertação tratam essa questão explorando o potencial na disponibilidade de conteúdo gerado pela presença pares que permanecem por mais tempo conectados ao sistema.

## CAPÍTULO 3

# COLEÇÕES DE VÍDEOS E PLAYLISTS DO YOUTUBE

Neste capítulo apresentam-se as coleções de vídeos utilizadas nos estudos (Capítulos 4 e 5) conduzidos ao longo desta dissertação. As coleções são agrupadas em duas classes: coleções de vídeos *Top-Feed* e coleções de *playlists*. Ambas as coleções foram obtidas do site YouTube e incluem metadados dos vídeos e *playlists* disponibilizados para acesso dos usuários.

No decorrer desta pesquisa utilizou-se mais de dois milhões de vídeos envolvendo coleções de seis categorias de vídeos e *playlists* do YouTube (música, esporte, entretenimento, filmes&animações, comédia e educação). Tais coleções foram estudadas, tendo-se extraído dessas coleções as seguintes estatísticas: duração dos vídeos e das *playlists*, padrão de popularidade dos vídeos e *playlists*. Analisou-se também de que forma a popularidade dos vídeos influencia a montagem das *playlists*.

O estudo destas coleções permite identificar características existentes nos vídeos da coleção *Top-Feed* e de *playlist*, que influenciam na disponibilidade de conteúdo provida pelos dois agrupamentos de usuários considerados no sistema de distribuição de conteúdo estudado neste trabalho. O primeiro agrupamento é composto por usuários que acessam vídeos da coleção *Top-Feed*, e o segundo por usuários que acessam vídeos a partir das *playlists*. Os usuários que acessam vídeos da coleção *Top-Feed* têm comportamento mais dinâmico, permanecendo-os por um curto período conectados no sistema. Por outro lado, os usuários que acessam vídeos a partir das *playlists*, permanecem conectados por longas sessões.

## 3.1 Metodologia de Coleta

Nesta seção, apresentam-se as metodologias adotadas para a coleta dos vídeos e *playlists* utilizados nos experimentos numéricos desta dissertação. Os dados foram obtidos a partir da implementação de um coletor que permitiu extrair metadados reais dos vídeos e *playlists*. Esse coletor utiliza a API de dados do YouTube (GOOGLE, 2012) que viabiliza o acesso por aplicações externas, ao conteúdo disponibilizado nas página do Site YouTube. Ambos, coletor e coleções, estão disponíveis para a comunidade mediante solicitação<sup>1</sup>. A metodologia adotada para obtenção das coleções vídeos Top-Feed e de *playlist* são apresentadas nas seções 3.1.1 e 3.1.2, respectivamente.

### 3.1.1 A Coleta dos Vídeos Top-Feed

Os metadados da coleção *Top-Feed* foram obtidos por um coletor construído a partir do método bola-de-neve (*snowball sampling*) (LEE et al., 2006). Este método implementa uma busca em largura sobre as componentes conexas de um grafo ( $G$ ) onde se encontram os vídeos obtidos nos *feeds*, chamados de vídeos sementes. Para as coleções *Top-Feed* as sementes foram definidas pelos vídeos associados aos seguintes *feeds* da página principal do YouTube: Mais qualificados, Favoritos, Mais Vistos, Mais Populares, Mais Discutidos e Mais Respondidos. Esses *feeds* podem ser filtrados por país e por categoria de vídeo. Dado que o sistema de distribuição, descrito nesta dissertação (ver Capítulo 4 e 5), leva a localização e o contexto em consideração, os conteúdos oferecidos aos brasileiros foram coletados e filtrados por seis das dez categorias listadas no YouTube. A Figura 3.1 mostra o esquema geral implementado pelo coletor.

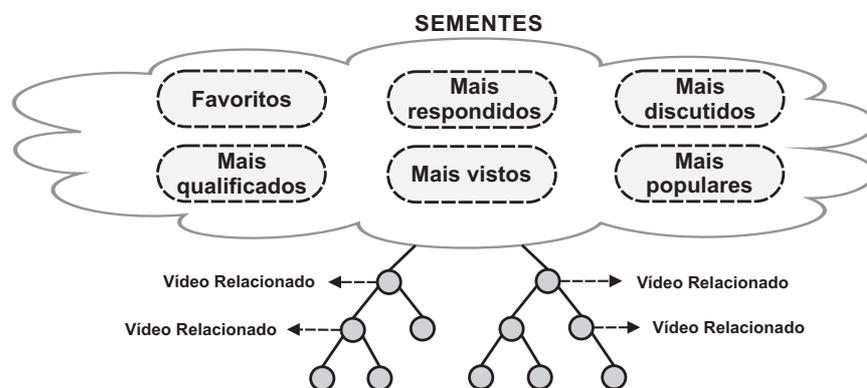


Figura 3.1: Método de coleta para as coleções Top-Feed.

Considerando a busca em largura implementada no método bola-de-neve, uma fila

<sup>1</sup>e-mail: jhonathan@icomp.ufam.edu.br

foi construída e alimentada com informações dos vídeos coletados dos *feeds* padrões. Para cada vídeo desenfileirado, os seguintes dados foram coletados (ilustrados na Figura 3.2): ID do vídeo, duração, visualizações, qualificação, autor, data de publicação, e um conjunto de 25 vídeos relacionados, que é gerado pelo sistema de recomendação do YouTube. Os vídeos desse conjunto são enfileirados e usados no prosseguimento da busca em largura, desde que não tenham sido previamente visitados.

<b>ID</b> 3GR3HvriWtY		<b>PROPRIETÁRIO</b> Carlos JSB	
<b>RATING</b> 5,00	<b>NÚMERO DE VISUALIZAÇÕES</b> 778999	<b>DURAÇÃO</b> 481	
<b>DATA/UPLOAD</b> 2012-10-14 T21:25:17.000Z		<b>CATEGORIA</b> MUSIC	

Figura 3.2: Metadados de um vídeo do Youtube.

A coleção do YouTube foi explorada até um nível previamente estabelecido, permitindo que um número máximo de vídeos relacionados possam ser acessados. O limiar do nível foi estabelecido com base na duração média de uma sessão de navegação no YouTube (YOUTUBE, 2013), que é de 15 minutos. Assim, em cada coleta, definiu-se um limiar para o nível de exploração da busca em largura de tal forma que o número máximo de vídeos relacionados pudesse ser acessado em uma sessão de conexão.

### 3.1.2 A Coleta das Playlists

O outro conjunto de dados, chamado de coleção de *playlists*, é composto por vídeos que estão em *playlists* do YouTube. Toda *playlist* criada no YouTube tem um dono e os vídeos em uma *playlist* são acessados seguindo a sequência estabelecida pelo dono.



Figura 3.3: Metodologia de coleta para as playlists.

Usando entradas da coleção *Top-Feed*, um conjunto  $S$  composto por pares  $(V_i, D_i)$ , que definem o Dono  $D_i$  do Vídeo  $V_i$  foi criado. Usando  $S$ , coletou-se todas as *playlists* do

dono  $D_i$  que possuem o vídeo  $V_i$ . As *playlists* foram processadas para que se tornassem únicas na coleção, uma vez que dois ou mais vídeos de um mesmo dono podem pertencer à mesma *playlist*. A Figura 3.3 ilustra o esquema geral da coleta.

<b>ID</b> PL1202D675158800FE		<b>PROPRIETÁRIO</b> escoladeinteligencia
<b>DURAÇÃO</b> 4661	<b>N° VÍDEOS</b> 10	<b>DATA/UPLOAD</b> 2010-07-16 T19:41:42.000Z
<b>CONJUNTO DE VÍDEOS</b> SGZXVZ0MeZW pbiUBN0gzfc fRWSzdtLL4  8opsH64jUA zkACP7c3t54 TIOE9sAixY  NXO3yq2wGvo pGFQZGfv2_0 w9SmUZDy3A  j4WBiv-PV9A		

Figura 3.4: Metadados de uma playlist do Youtube.

A Figura 3.4 apresenta uma amostra dos metadados extraídos para cada *playlist* do YouTube. Esses metadados são compostos pelo ID da *playlist*, autor, duração, número de vídeos, data de publicação, e os ID's dos vídeos que compõem a *playlist*. Na época em que foram realizadas as coletas, o YouTube permitia que seus usuários criassem *playlists* com até 200 vídeos, e um ilimitado número de *playlists* por usuário.

## 3.2 Caracterizando os Vídeos das Coleções Top-Feed e Playlist

Nesta seção, as coleções de vídeos *Top-Feed* e as coleções de *playlist* são analisadas. Primeiro, descreve-se os padrões de duração dos vídeos e das *playlists*, posteriormente verifica-se o padrão de popularidade dos vídeos *Top-Feed* e das *playlists* para em seguida, identificar-se o papel de tal popularidade na montagem das *playlists*.

Na Tabela 3.1 são apresentados dados estatísticos das coleções *Top-Feed* e *playlist*. Dentre os dados apresentados, mostra-se o padrão de duração dos vídeos *Top-Feed* e das *playlists*. Observa-se que para as coleções *Top-Feed* os vídeos são de curta duração e, em média variam entre 3 e 7 minutos. A coleção da categoria educação é a que possui maior média de duração dentre as coleções. Tal comportamento é explicado pela característica de alguns vídeos dessa coleção, que são do tipo vídeo-aulas, ou seja, os usuários integram vários vídeos em apenas um. Analisando o coeficiente de variação, conclui-se que a coleção da categoria filme apresenta a maior dispersão na duração dos vídeos, com coeficiente de variação 0,83, enquanto que, a categoria música é a menos dispersa, com coeficiente de variação 0,38. Para as coleções de *playlists*, verifica-se que a média de duração varia entre 27 e 79 minutos, sendo a coleção da categoria educação é a que apresenta a maior média.

A descrição relatada anteriormente na coleção *Top-Feed* da categoria educação, também explica esse padrão na duração das *playlists* de educação.

Tabela 3.1: Estatísticas das coleções Top-Feed e Playlist.

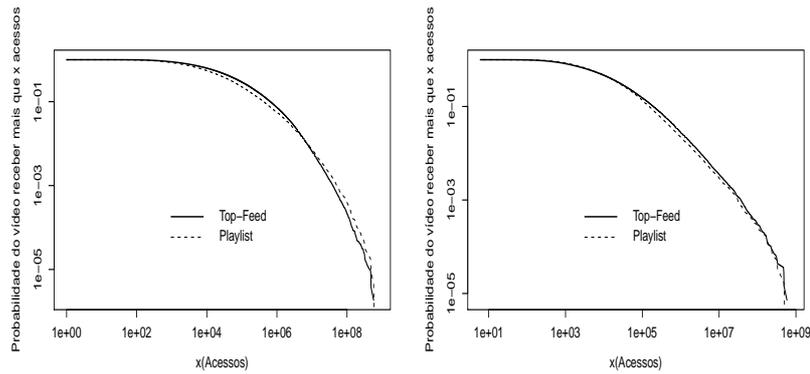
Categoria	Coleção Top-feed				
	Princípio de Pareto	Nro de Vídeos	Duração(seg.)		Data da Coleta
			Média( $\mu$ )	Varição( $\sigma$ )	
Música	80% - 7%	535,687	222.6	0.38	2012-07-27
Esporte	80% - 10%	226,043	169.2	0.76	2012-08-01
Entretenimento	80% - 7%	413,524	219.3	0.79	2012-08-03
Filmes&Animações	80% - 12%	535,687	250.0	0.83	2012-11-10
Comédia	80% - 12%	448,591	235.5	0.80	2012-11-12
Educação	80% - 10%	330,843	433.0	0.64	2012-11-08

Categoria	Coleção Playlist				
	Princípio de Pareto	Nro de Playlists	Por Playlist (média)		Data da Coleta
			Duração em seg.	Nro de Vídeos	
Música	80% - 4.0%	22,550	2,607.7	9.80	2012-07-27
Esporte	80% - 2.0%	10,602	1,647.2	10.07	2012-08-01
Entretenimento	80% - 2.4%	14,969	2,190.6	12.03	2012-08-03
Filmes&Animações	80% - 3.2%	16,205	2,254.2	12.33	2012-11-15
Comédia	80% - 4.0%	19,105	2,028.1	11.20	2012-11-17
Educação	80% - 2.0%	16,788	4,762.6	12.24	2012-11-20

Na Figura 3.5 apresentam-se os gráficos da distribuição dos acessos nas coleções gerados através da função de distribuição cumulativa complementar, do inglês, *Complementary Cumulative Distribution Function* (CCDF). Para as coleções de música e esporte as curvas mostram que ambas as coleções (*Top-Feed* e *playlist*) compartilham um padrão de popularidade similar. Tal padrão pode ser resumido da seguinte forma: a probabilidade de um vídeo receber um grande número de visualizações cai rapidamente à medida em que esse vídeo não está nas primeiras posições do ranking. Em outras palavras, em ambas as coleções, um subconjunto de vídeos recebe a maioria das visualizações.

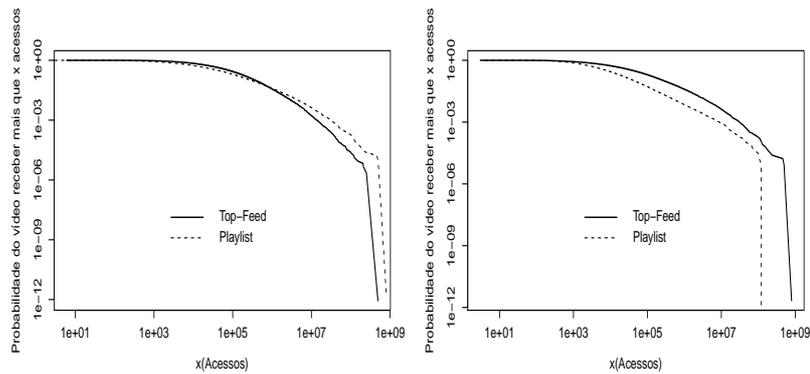
Para as outras coleções (comédia, educação, entretenimento e filmes&Animações), este padrão também é observado (ver Figura 3.6). Entretanto, verifica-se que existe um distanciamento entre as curvas das coleções *Top-Feed* e *playlist*. Esse distanciamento é explicado pela dinâmica de construção das *playlists*. Como os usuários são livres para editarem suas *playlists*, podem ocorrer situações onde vídeos classificados pelo YouTube como sendo de uma determinada categoria sejam inseridos em uma *playlist* de outra categoria. Tal comportamento é mais frequente na categoria de entretenimento visto que,



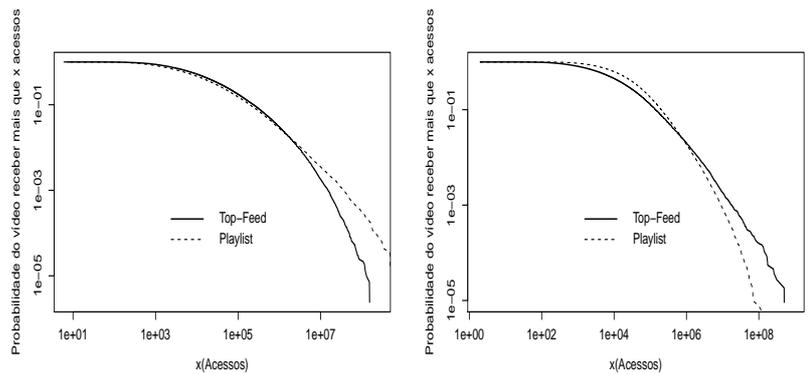
(a) CCDF para coleção de música (b) CCDF para coleção de esporte

Figura 3.5: Modelando a popularidade dos vídeos das coleções Top-Feed e playlist de música esporte.

por natureza, os conteúdos dessa categoria tendem envolver diversas outras categorias.



(a) CDF para coleção de comédia (b) CDF para coleção de educação



(c) CDF para coleção de entretenimento (d) CDF para coleção de filmes&animações

Figura 3.6: Modelando a popularidade dos vídeos das coleções Top-Feed e playlist das coleções de comédia, educação, entretenimento e filmes&animações .

Para verificar como as visualizações estão distribuídas, as coleções foram analisadas segundo o princípio de Pareto, também conhecido com regra 80-20%. Essa regra, que

define um padrão geral de uso de recursos pelas demandas, estabelece que 20% das demandas consomem 80% de todos os recursos disponíveis. No estudo realizado, identificou-se a porcentagem de vídeos, de cada coleção, que responde por 80% das visualizações registradas. Na *playlist* da coleção de música 80% das visualizações estão associadas a somente 4% de seus vídeos, enquanto na coleção *Top-Feed* 80% das visualizações são associadas a 7%. Para as coleções de vídeos de Entretenimento esses números são 80% - 2,4% (*Playlist*) e 80% - 7% (*Top-Feed*), e para as coleções de Esporte tem-se 80%-2% (*Playlist*) e 80% - 10% (*Top-Feed*), (ver Tabela 3.1 para as demais coleções). Esses resultados mostram que as popularidades dessas coleções divergem da regra geral 80-20%, especialmente as *playlists*. Tal divergência pode ser explicada pela dinâmica de navegação nos vídeos do YouTube, que em geral é iniciada nos vídeos da página principal, através dos *feeds* padrão, e então desce para vídeos de menor popularidade, seguindo as sugestões do sistema de recomendação do YouTube, e finalizando após 15 minutos de navegação em média (YOUTUBE, 2013).

Em resumo, tem-se que a curta duração das sessões de acesso e a atuação do sistema de recomendação tornam as requisições à coleção de vídeo do Youtube mais concentradas a um subconjunto que é menor que aquele definido pelo princípio de Pareto.

### 3.2.1 A Influência da Popularidade dos Vídeos na Montagem das Playlists

Nesta seção, avalia-se de que forma a popularidade dos vídeos influencia os donos das *playlists* no processo de construção das mesmas. As investigações realizadas apontam que uma quantidade significativa de donos de *playlist* tinham conhecimento do status dos vídeos quando ocorreu a montagem das *playlists*.

*Playlists* podem ser criadas e mantidas inalteradas ou podem ser atualizadas periodicamente. O YouTube não registra a data em que o vídeo foi adicionado a *playlist*, embora exista registro da data em que os vídeos foram disponibilizados, i.e a data de *upload*, e da data em que as *playlists* foram publicadas. Essa informação foi processada e verificou-se que a maioria dos vídeos são mais velhos que as *playlists* em que eles estão listados. Por exemplo, na coleção de música, 71% dos vídeos são mais velhos que as suas *playlists*, para as coleções de Entretenimento, Esporte, Educação, Comédia, Filmes&Animações essas proporções são 67%, 69%, 59%, 63% e 53%, reforçando que vídeos selecionados para *playlists* tinham um lastro de visualizações quando foram incluídos nas *playlists*.

Crane e Sornette(CRANE; SORNETTE, 2008) argumentam que a popularidade de vídeos no YouTube pode ser agrupada em quatro classes. A primeira classe, chamada de neutra, composta por 90% dos vídeos, apresenta pouca flutuação em sua popularidade, e as outras três classes têm vídeos que, após um período de dormência, experimentam um aumento rápido de visualizações. Essas três classes são nomeadas como Viral, Qualidade e Lixo, devido à intensidade das atividades relacionadas a elas em um curto período de tempo. Neste mesmo trabalho (CRANE; SORNETTE, 2008), os autores mostram que o período de dormência não dura mais que 250 dias. Nas coleções de vídeos da *playlist*, 76%, 67% e 77% dos vídeos mais populares nas coleções de Música, Entretenimento e Esporte, respectivamente, já tinham passado o período de dormência quando as *playlists* que eles estão listados foram publicadas. Para as coleções Comédia, Educação e Filmes&Animações verificou-se que 66.7%, 66.1% e 57.7% dos vídeos, respectivamente, já haviam ultrapassado aquele período de dormência.

Ao considerar as estatísticas de popularidade e idade dos vídeos e *playlists*, é razoável inferir que o dono da *playlist* tinha informações suficientes para escolher, entre os vídeos disponíveis nas coleções, os mais relevantes em relação ao tema da *playlist* bem como aqueles que já possuíam um número significativo de visualizações.

### 3.3 Considerações Finais do Capítulo

Neste capítulo foram apresentadas as coleções vídeos e *playlists* utilizadas nesta pesquisa. Primeiramente, apresentou-se a metodologia adotada para a coleta dos vídeos das coleções *Top-Feed* e *playlist*. Em seguida, estudou-se estatísticas relacionadas a ambas as coleções. Tais estatísticas envolveram duração dos vídeos e *playlists*, padrão de popularidade e a influência da popularidade dos vídeos na montagem das *playlists*.

Para análise da popularidade identificou-se que em todas as coleções, há um padrão similar de popularidade entre as coleções de vídeos *Top-Fed* e *playlist*. Esse padrão, define que um pequeno subconjunto dos vídeos recebe a maioria das visualizações. Além disso, verificou-se que a maioria dos vídeos que são inseridos nas *playlists* dos usuários do YouTube já possuíam um certo nível de popularidade. Em outras palavras, a maioria dos vídeos que são inseridos nas *playlists*, já haviam recebido um lastro de visualizações.

## CAPÍTULO 4

# DISPONIBILIDADE DE CONTEÚDO PROMOVIDA POR TOCADORES DE PLAYLISTS EM SISTEMAS CDN ASSISTIDOS POR REDES P2P

Os sistemas CDN assistidos por redes P2P (CDN-P2P) têm sido implantados a fim de explorar vantagens de ambos, CDN e P2P. Nestes sistemas, o conteúdo armazenado nos servidores da rede de distribuição de conteúdo (CDN) é compartilhado pelos pares da rede par-a-par (P2P), diminuindo a demanda sobre a infraestrutura da CDN. Por outro lado, os servidores da CDN garantem a disponibilidade de conteúdo quando a contribuição dos pares é limitada pelo *churn*, que é o efeito gerado pelos ciclos de entrada e saída de pares, ou quando o conteúdo não tiver sido acessado por pares da rede P2P.

Contudo, a arquitetura CDN-P2P, por si só, não garante a distribuição efetiva de conteúdo, uma vez que a saída de pares do sistema que são os únicos detentores de determinados conteúdos pode gerar congestionamento em torno do servidor CDN e degradar a qualidade de experiência dos pares. Um questão a ser investigada é a disponibilidade de conteúdo gerada por pares que passam mais tempo conectados à rede. Em (WANG et al., 2008), os autores estudam a importância de pares estáveis em sistemas P2P de distribuição de fluxo de vídeo ao vivo. O ajuntamento de conteúdo popular e não-popular (*bundling*), para aumentar a disponibilidade de conteúdo, especialmente o não popular, foi estudado em (MENASCHE et al., 2009).

Em um sistema CDN-P2P que distribui conteúdo similar ao disponibilizado pelo YouTube, pares estáveis acessam *playlists* de vídeos, que são coleções de vídeos criadas e mantidas pelos seus donos. A Figura 4.1 ilustra uma *playlist* de vídeos do YouTube.

Uma *playlist* é composta por um conjunto de vídeos, que podem se reproduzidos sequencialmente. Embora a edição dessas *playlists* fique a cargo de seus donos, o número de vídeos que podem ser inseridos nas *playlists* é estabelecido pelos provedores. Por exemplo, usuários do YouTube podem criar *playlists* com até 200 vídeos. A maioria dos vídeos dessas *playlists* são sobre tópicos específicos e agregam vídeos populares e não populares, de programas de TV, grandes eventos de esporte e apresentações de artistas.



Figura 4.1: Exemplo de uma playlist de vídeos do YouTube.

Desta maneira, no presente capítulo investiga-se o impacto de tocadores de *playlists* na disponibilidade de conteúdo em redes CDN-P2P considerando diversos cenários de rede. Os resultados numéricos mostram que mais de 40% das demandas geradas pelos pares podem ser atendidas na parte P2P do sistema, quando o sistema tem uma alta taxa de chegada de tocadores de *playlists*. Em sistemas com baixa taxa de chegada, a disponibilidade é melhorada em pelo menos 20%, indicando que, mesmo em comunidades de pares menos participativos, tocadores de *playlists* podem ajudar a melhorar a disponibilidade de conteúdo em um sistema CDN-P2P.

O restante deste capítulo está organizado da seguinte forma. Na Seção 4.1 apresentam-se trabalhos da literatura que estudam a importância de pares estáveis em redes P2P. Na Seção 4.2 descreve-se o sistema CDN-P2P avaliado. Na Seção 4.2.2 descreve-se o método utilizado para reproduzir o acessos aos vídeos do Youtube usados nas simulações. Posteriormente, na Seção 4.3 apresentam-se os resultados numéricos coletados nos experimentos de simulação. Por fim, na Seção 4.4, as considerações deste capítulo são apresentadas.

## 4.1 Estabilidade de pares em sistemas P2P

A identificação e a importância de pares estáveis ou super-pares em sistemas P2P de larga escala têm sido objeto de diversos estudos (LV et al., 2013)(LI et al., 2011)(WANG et al., 2011)(KWON et al., 2011)(MENASCHE et al., 2009)(WANG et al., 2008)(SACHA et al., 2006). Entre os fatores considerados para caracterizar a estabilidade dos pares destacam-se o tempo de permanência no sistema, largura de banda, capacidade de processamento e armazenamento. No que segue, alguns trabalhos mais diretamente relacionados com o presente estudo são apresentados.

Li et al. (2011) estudam o impacto da formação de grupos homogêneos em uma rede P2P para distribuição de fluxo de vídeo ao vivo utilizando traços reais capturados do sistema Gnutella (GNUTELLA, 2013). Os grupos são compostos por pares de maior quantidade de recurso e outros pares, de menor quantidade de recursos. Os recursos considerados pelos autores foram capacidade de armazenamento e largura de banda disponível em cada par. Um modelo analítico baseado em processos estocásticos foi proposto para modelar a estabilidade ótima do sistema. Os resultados numéricos mostram que o esquema proposto reduz o custo de comunicação e de transferência de dados entre o pares do sistema. Enquanto (LI et al., 2011) avaliam um sistema de fluxo de vídeo ao vivo, e a estabilidade dos pares é definida pela capacidade de armazenamento e largura de banda dos pares, no presente estudo considera-se o fato do par está acessando uma *playlist* para caracterizá-lo como estável, e o sistema considerado é o de fluxo de vídeo sob-demanda do tipo YouTube.

Wang et al. (2011) avaliam como pares estáveis em um sistema P2P podem melhorar uma sessão de fluxo de vídeo ao vivo, ao manterem conteúdo disponível durante parte da sessão. Os autores mostram que a Qualidade de serviço, do inglês *Quality of Service*, de sistemas P2P de fluxo de vídeo pode ser melhorada quando pares estáveis são identificados e organizados em árvores de distribuição melhorando a resiliência do sistema. Nesse sistema, a estabilidade dos pares é medida pela razão entre a participação do par em um fluxo e a duração da sessão de fluxo, classificando-se os pares como estáveis ou instáveis. Resultados numéricos mostram que a taxa de perda de dados, atraso de inicialização e atraso na reprodução são reduzidos nos sistemas P2P quando operam com os algoritmos propostos. Enquanto em (WANG et al., 2011), os autores estudam um sistema de fluxo ao vivo e a estabilidade do par é definida usando diferentes critérios, o estudo no presente trabalho considera o fato do par está acessando uma *playlist* para caracterizá-lo como estável, e o sistema considerado é o de fluxo de vídeo sob-demanda do tipo YouTube.

Em (MENASCHE et al., 2009), os autores estudam o impacto da forma como conteúdos são agregados, chamada *bundling*, na disponibilidade desses conteúdos em um sistema de compartilhamento de arquivos baseado em *swarm*. Essa técnica ajunta vários arquivos com diferentes níveis de popularidade em um único grande *swarm*. Um modelo baseado em uma fila  $M/G/\infty$  foi proposto para modelar o sistema. Os autores mostram que um ajuntamento de  $K$  arquivos é capaz de aumentar a disponibilidade de conteúdo não popular por um fator de  $e^{\Theta(K^2)}$ . O presente estudo mostra que a *playlist* guarda similaridade com o *bundling*, dado que vídeos populares e não tão populares são ajuntados e tocados sequencialmente, se for da conveniência dos usuários. Uma *playlist* é um tipo especial de *bundle*, na qual usuários podem selecionar quais conteúdos irão assistir. Esse tipo de *bundle* é também chamado de *bundle* misto. Enquanto em (MENASCHE et al., 2009) os autores estudam apenas *bundles* puros em sistemas do tipo bitTorrent, no presente estudo, consideram-se *bundles* mistos, em sistemas do tipo YouTube.

Sacha et al. (2006) mostram um algoritmo *Gossip* para seleção de vizinhos baseado no tempo médio de permanência dos pares na rede. Este tempo caracteriza a utilidade do par no sistema. O algoritmo considera a formação de uma topologia gradiente. Nesta topologia, os pares de mais alta utilidade são conectados entre si, formando um núcleo na rede. Tal núcleo é responsável por manter a maior demanda de recursos da rede, enquanto que os pares de menor utilidade são dispostos gradualmente mais distantes do núcleo. Quando um par realiza uma consulta por algum conteúdo, essa é encaminhada em direção ao núcleo. Os experimentos realizados revelam que a busca baseada na topologia gradiente exhibe melhor desempenho do que a busca baseada na exploração de Boltzmann (SUTTON; BARTO, 1998) e a busca aleatória, em termos de número de saltos e a taxa de perda de mensagens. Enquanto (SACHA et al., 2006) avaliam o número de saltos e taxa de perda de mensagens gerados em uma busca, o presente estudo mede a disponibilidade de conteúdo provida por pares que permanecem por mais tempo conectados ao sistema e que são caracterizados por acessarem vídeos a partir de *playlists*.

## 4.2 O Sistema de Distribuição de Conteúdo Simulado

Nesta seção, apresenta-se o sistema de distribuição de conteúdo considerado neste estudo, bem como as diretrizes gerais da distribuição do conteúdo a partir do sistema. Os estudos realizados utilizaram simulação numérica, tendo-se avaliado características do sistema que impactam a distribuição do conteúdo, especificamente a presença de pares que permanecem um maior tempo no sistema.

O sistema de distribuição de conteúdo é um sistema CDN assistido por redes P2P (CDN-P2P) concebido para distribuir vídeos de curta duração similares aos distribuídos pelo YouTube. A Figura 4.2 mostra a arquitetura do sistema avaliado. Nesse sistema, a requisição para um vídeo pode ser atendida por um par da porção P2P ao invés de um servidor da CDN. Para que isso aconteça, vídeos acessados por um par da porção P2P devem ser armazenados no par tornando-o um servidor desses vídeos. Adicionalmente, tem-se que os pares são agrupados em comunidades construídas em torno de um tema, por exemplo, Música, Educação, Esporte, Entretenimento e Jogos, facilitando o compartilhamento de conteúdo. Esse agrupamento, também restringe a operação de busca aos membros da comunidade que tomam conhecimento de seus vizinhos a partir de metadados disseminados pelos servidores da CDN. Embora os pares possam estar em várias comunidades simultaneamente, é muito mais provável que eles permaneçam em uma única comunidade. De fato, estatísticas mostram que sessões do YouTube duram, em média, 15 minutos, período relativamente pequeno para um par acessar dois vídeos com temas completamente diferentes. Além disso, a maioria das visualizações, algo como 60% das requisições para vídeos, vem da lista de vídeos relacionados, construída pelo sistema de recomendação que indica sua relevância (DAVIDSON et al., 2010).

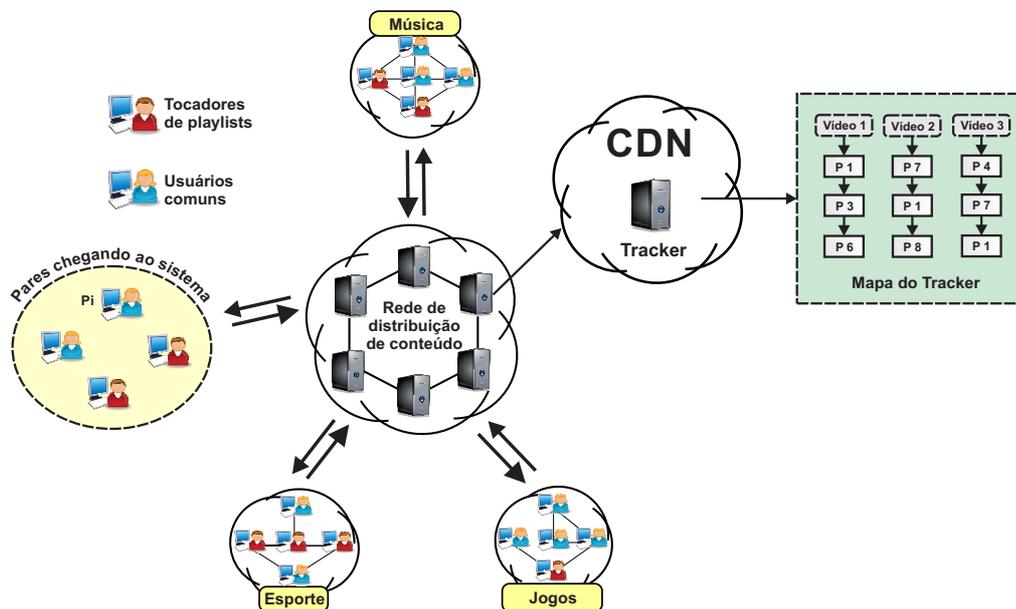


Figura 4.2: Arquitetura do sistema CDN-P2P avaliado.

Nesse sistema, existem dois tipos de usuários: os que acessam os vídeos diretamente da coleção de vídeos e aqueles que acessam os vídeos a partir de uma *playlist*. O primeiro tipo, chamado de usuário comum, investe algum tempo considerando os vídeos recomendados ou mesmo navegando pela coleção de vídeos antes de realizar o acesso a um vídeo. O último, chamado de tocador de *playlist*, acessa os vídeos a partir de uma

*playlist*, seguindo a ordem em que eles se encontram na *playlist*, não investindo tempo em considerações sobre que vídeo deve ser acessado.

O servidor da CDN que atende uma comunidade também atua como um *tracker* (como ilustrado na Figura 4.2), com características similares ao utilizado no BitTorrent (JOUNG; HUANG, 2012b). Este *tracker* possui um mapa do sistema, com informações sobre os conteúdos acessados pelo pares da comunidade. Assim, quando um par chega ao sistema e realiza a requisição de um determinado conteúdo, sua requisição será dirigida ao *tracker* que pode então indicar uma lista de pares que já possuem o conteúdo ou indicar que tal conteúdo deve ser acessado do servidor. Sempre que um par armazena um conteúdo acessado, ou realiza o descarte de um conteúdo previamente armazenado, é sua responsabilidade informar ao *tracker* a ocorrência de tal evento. Neste trabalho considerou-se que o processo de busca e requisição por um conteúdo se dá a partir do vídeo, e não de partes dele. Com isso, o estudo relacionado com mecanismos de escalonamento para o acesso desse conteúdo nos pares está além do escopo desse trabalho, de forma que a presença ou não do conteúdo na porção P2P foi considerada como a medida de desempenho a ser observada.

#### 4.2.1 Modelo de *Churn*

Nesta seção, descreve-se o modelo de *churn* empregado para porção P2P do sistema de distribuição simulado. Tal modelo, é um modelo genérico definido em Yao et al. (2006).

Seja um sistema P2P com  $n$  pares, onde cada par  $i$  está conectado (*UP*) ou desconectado (*DOWN*) em um tempo  $t$ . Esse comportamento pode ser modelado por um processo  $Z_i(t)$  (Equação 4.1) para cada par  $i$ :

$$Z_i(t) = \begin{cases} 1 & \text{se o par } i \text{ está } UP \text{ no tempo } t \\ 0 & \text{caso contrário} \end{cases} \quad (4.1)$$

Para cada processo  $Z_i(t)$  a duração de uma sessão *UP*  $\{L_{i,c}\}_{c=1}^{\infty}$  tem uma distribuição associada  $F_i(x)$  e a duração de uma sessão *DOWN*  $\{D_{i,c}\}_{c=1}^{\infty}$  tem outra distribuição associada  $G_i(x)$ , onde  $c$  representa o número do ciclo. A duração das sessões *UP* e *DOWN* de um par  $i$  é dada pelas variáveis aleatórias  $L_{i,c} > 0$  e  $D_{i,c} > 0$ , respectivamente. O *churn* do sistema P2P avaliado neste trabalho foi modelado pela distribuição exponencial:

$$f(x; \lambda) = 1 - e^{-\lambda x}$$

com média  $\lambda^{-1}$

Com base no modelo acima, as seguintes suposições são feitas:

1. Para capturar a natureza independente dos pares, o conjunto  $\{Z_i(t)\}_{i=1}^n$  é composto por processos de nascimento e morte mutuamente independentes. Assim, os pares se comportam de forma independente uns dos outros e os processos  $\{Z_i(t)\}$  e  $\{Z_j(t)\}$ , para qualquer  $i \neq j$ , são independentes;
2.  $l_i$  é a duração média de uma sessão *UP* e  $d_i$  é a duração média de uma sessão *DOWN*, ambas, são independentes para cada par. Os valores de  $l_i$ , e  $d_i$  considerados no presente estudo são apresentados na Tabela 4.3.

O emprego da distribuição exponencial no modelo de *churn* considerado no presente trabalho foi baseado no estudo realizado em (MELO et al., 2011b). Nesse estudo, os autores avaliaram diferentes processos renováveis, definidos a partir da combinação das distribuições Pareto e exponencial, na modelagem da autonomia (*churn*) do pares da porção P2P do sistema. O uso da distribuição Pareto na duração das sessões caracterizam uma participação errática dos pares, enquanto que o uso da distribuições exponencial caracteriza pares como uma participação mais próximo do comportamento médio esperado. Como destacado em (MELO et al., 2011b), a qualidade das conclusões sobre a disponibilidade de conteúdo no sistema tem pouca influência do processo renovável definido pela combinação das distribuições exponencial e Pareto e usado para caracterizar o *churn*.

## 4.2.2 Padrão de Acesso aos Vídeos

Nesta seção, apresenta-se o método adotado para simular os acessos aos vídeos no sistema de distribuição avaliado no presente estudo. Os vídeos utilizados nas simulações envolvem seis categorias de vídeos e *playlists* do Youtube e são apresentados no Capítulo 3 desta dissertação.

A popularidade de conteúdos multimídia tem sido modelada pela distribuição Zipf e suas variações, tais como Zipf com corte Exponencial e Mandelbrot-Zipf. A distribuição Exponencial Dilatada, do inglês *Stretched Exponential*, tem também sido considerada como apropriada para modelar a popularidade de vídeos (GUO et al., 2008). Tais conclusões foram derivadas usando diferentes coleções de vídeos, disponibilizadas em uma variedade de sistemas de distribuição. Entretanto, o que existe de comum entre essas abordagens é que elas são todas distribuições que modelam a lei-da-potência, do inglês

*power-law*. A lei-da-potência é uma relação funcional entre duas grandezas. Quando uma das grandezas medidas depende da outra, elevada a certa potência, diz-se que essa grandeza segue uma lei de potência (ou lei de escala) (FALOUSTOS et al., 1999) (ZHOU, 2009).

Lee et al. (2005) mostraram que valores de medições feitas em dados que seguem a lei-de-potência são muito sensíveis à amostragem. Neste estudo, os autores exploraram vários métodos de amostragem, e identificaram que a relação entre a quantidade de amostras e propriedades topológicas das rede tais como, grau de centralidade, coeficiente de clusterização e tamanho médio de caminhos da rede, pode influenciar na geração de traços sintéticos que diferem o comportamento real da rede.

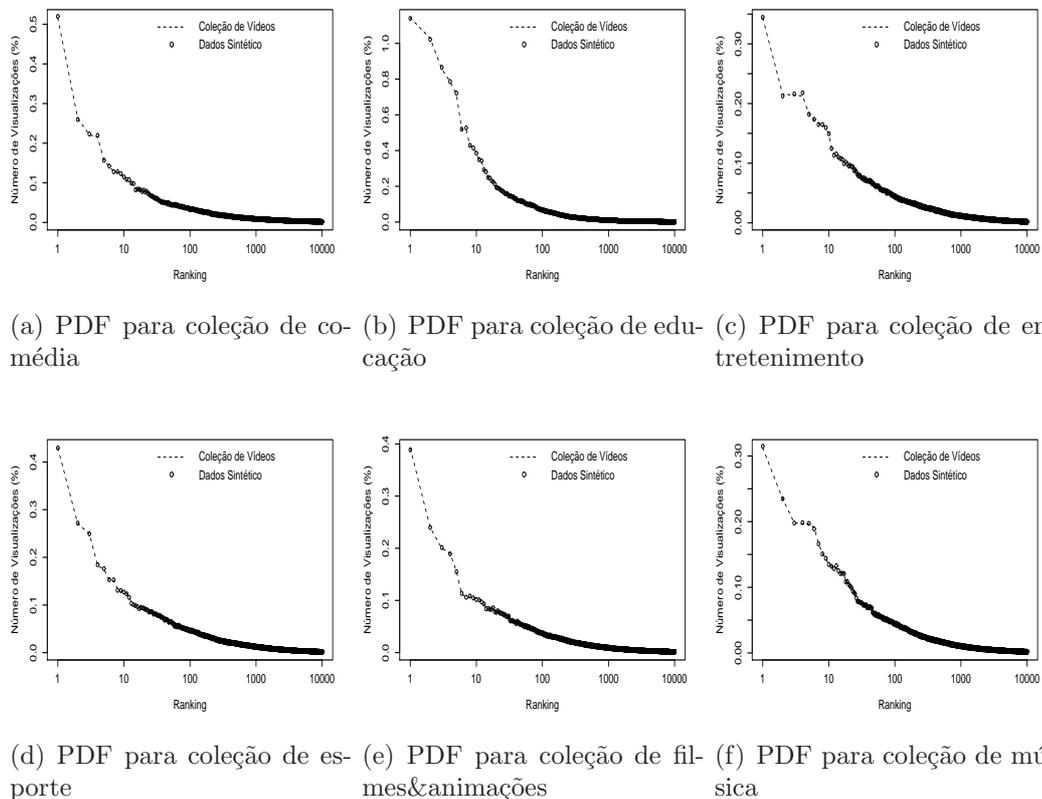


Figura 4.3: PDF's real e sintética da popularidade dos vídeos.

Reconhecendo as dificuldades de modelar popularidade em redes sociais on-line, especialmente usando as distribuições mencionadas, aplicou-se o seguinte procedimento para simular os acessos aos vídeos do sistema:

1. Todas as requisições são guiadas pela combinação da duração do vídeo e do “tempo de avaliação”, ou seja, o tempo no qual o usuário decide qual vídeo irá assistir;

2. Os acessos foram reproduzidos a partir da função densidade de probabilidade extraída da coleção do YouTube utilizando o método de geração de número aleatório aceitação-rejeição (RIPLEY, 1987);

As suposições feitas nesse procedimento são que os vídeos acessados serão vistos por completos e os usuários irão despende algum tempo procurando vídeos relacionados antes de acessar um novo vídeo. A Figura 4.3 mostra a função de distribuição de probabilidade para as coleções de vídeos Top-Feed e o seu similar sintético. Observa-se que os acessos gerados em simulação representam satisfatoriamente o padrão de visualização identificado nas coleções do YouTube. O teste de aderência não paramétrico de Wilcoxon com nível de significância de 5% aceita a hipótese de que os valores gerados pela simulação (passo 2 do procedimento) provêm da mesma função de distribuição extraída das coleções do YouTube.

### 4.3 Resultados Numéricos

Nesta seção, apresentam-se os resultados dos experimentos que mediram a melhoria na disponibilidade de conteúdo em sistemas CDN-P2P que possui pares tocadores de *playlists* em sua parte P2P. Nos experimentos foram usados dados reais coletados do YouTube (Capítulo 3) e considerou-se a disponibilidade de conteúdo gerada pela presença do vídeo requisitado na parte P2P do sistema. Avaliou-se, através de experimentos que simularam o sistema descrito na Seção 4.2, o impacto da taxa de chegada de usuários tocadores de *playlists* em um cenário em que toda a *playlist* é tocada e em um cenário em que a reprodução da *playlist* é interrompida.

No sistema estudado existem dois tipos de usuários: os tocadores de *playlists* e os comuns. Os usuários comuns podem sair ou entrar no sistema em qualquer instante que lhes seja conveniente. Ao entrar no sistema, o par (usuário) inicia uma sessão de conexão (sessão *UP*) que irá durar até a sua partida. Por outro lado, ao deixar o sistema uma sessão de desconexão (sessão *DOWN*) será iniciada, encerrando-se quando ocorrer o retorno do par para o sistema. Para caracterizar a dinâmica gerada pela autonomia dos usuários comuns, utilizou-se o modelo apresentado na Seção 4.2.1, que é baseado em processos renováveis independentes, para definir a duração das sessões *UP* e *DOWN*.

A autonomia dos tocadores de *playlists* foi baseada na ideia geral aplicada aos usuários comuns, com as sessões *UP* e *DOWN*. Entretanto, para caracterizar a duração das sessões *UP* considerou-se que o tocador de *playlists* irá acessar os vídeos listados

previamente, sem realizar considerações sobre a ordem de execução ou interrupções momentâneas na execução dos vídeos na *playlist*. O acesso então ocorrerá de forma integral ou haverá uma interrupção definitiva na reprodução da *playlist*. Os valores da sessão *UP* para os tocadores de *playlists* apresentados na Tabela 4.3 estão na seguinte ordem: Música, Esporte, Entretenimento, Filmes&Animações, Comédia e Educação. Os usuários comuns acessam os vídeos em sessões *UP* que duram 1/4 de hora, em média, conforme estatística extraída de (YOUTUBE, 2013).

Em (MELO et al., 2011a), estudou-se o impacto da capacidade de armazenamento e do número de pares na disponibilidade de conteúdo de um sistema híbrido CDN-P2P. A partir desse estudo definiu-se uma comunidade com 10.000 pares, sendo 2.000 pares, no máximo, formando o agrupamento de tocadores de *playlists* e 8.000 pares, no mínimo, formando o agrupamento de usuários comuns. A capacidade de armazenamento (*buffers*) é reduzida, de apenas 34 vídeos, evitando os ganhos de desempenho que uma capacidade maior de armazenamento propicia, e a duração das sessões *DOWN*, entre 6 e 24 horas. A Tabela 4.3 mostra detalhadamente o conjunto de parâmetros do sistema e as valores usados nos estudos.

Tabela 4.1: Configuração dos agrupamentos P2P para simulação do Sistema CDN-P2P.

Parâmetros do Sistema	Níveis do Parâmetro	Agrupamento de usuários	
		Comum	Tocador de Playlists
Duração Média da Sessão	UP	1/4 (hora)	43.45, 27.45, 36.5, 37.57, 33.8, 79.37 (minutos)
	DOWN	24 (hora)	6, 12 e 24 (hora)
Modelo de Churn	UP	exponencial	PDF empírica
	DOWN	exponencial	exponencial
Tamanho do Buffer (vídeo)	Por Par	34	34

A disponibilidade de conteúdo foi a medida de desempenho usada para avaliar o sistema estudado, tendo-se considerado a taxa definida pelo número de requisições geradas na porção P2P do sistema e o número de requisições que permanecem na malha, i.e que foram atendidas pelos próprios pares que formam a porção P2P do sistema. Essa taxa, chamada de taxa de acerto (TA), é dada por:

$$TA = \frac{RA}{TRT} \quad (4.2)$$

*RA* é o número de requisições atendidas pelos pares e *TRT* é o número total requisições realizada pelos pares da porção P2P.

A política *Greedy-Dual Size Adaptive* (GDSA) (MELO et al., 2012) foi usada para gerenciar o conteúdo em cada par e o sistema foi simulado durante um ano, com tempo de aquecimento da simulação igual a três meses e coleta de dados semanais. Os resultados apresentados são as médias dos valores coletados após o período de aquecimento do sistema, com intervalo de confiança de 95%. Os problemas relacionados à distribuição de carga estão fora do escopo deste trabalho pois requerem a definição de mecanismos de controle de admissão e escalonamento que otimizem a utilização do canal disponível nos usuários.

A Figura 4.4 mostra a taxa de acerto para pares comuns e tocadores de *playlists* em função do número de tocadores de *playlists* no sistema. Nessa figura, a barra sistema representa a disponibilidade de conteúdo que foi calculada somando-se a disponibilidade nos dois agrupamentos de pares, tocadores de *playlist* e pares comuns. A medida que o número de tocadores de *playlists* aumenta, aumentam também suas contribuições para a disponibilidade de conteúdo. A taxa máxima de acerto foi de 45%, para sessão *DOWN* de seis horas e 20% de tocadores de *playlists* no sistema. A contribuição de pares comuns diminuiu de 22% para 1% com o aumento do número de tocadores de *playlists*.

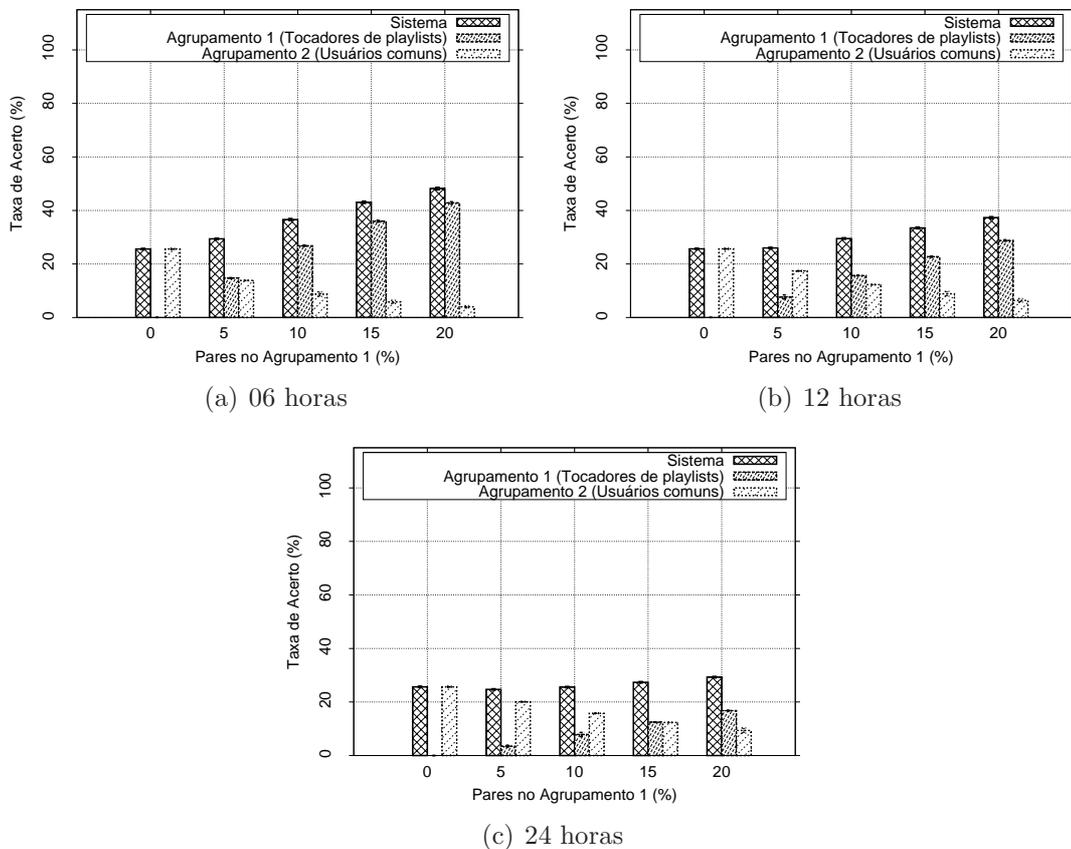


Figura 4.4: Taxa de acerto de pares comuns e tocadores de *playlist* com três duração de sessão *DOWN* para a coleção de vídeos de música.

A rapidez com que os tocadores de *playlists* tornam-se os principais responsáveis pela disponibilidade de conteúdo está relacionada com a frequência com que esses pares estão disponíveis para atender requisições de outros pares. Por exemplo, para uma sessão *DOWN* de 24 horas de duração, os tocadores de *playlists* dominam a taxa de acerto quando eles representam 15% dos pares no sistema, enquanto que, para uma sessão *DOWN* de seis horas, os tocadores tornam-se os maiores responsáveis pela disponibilidade de conteúdo quando são apenas 5% dos pares. A redução do tempo entre sessões *UP* implica em aumentar a taxa de chegada de pares tocadores de *playlists* ao sistema, tornando-os aptos a compartilharem os vídeos das *playlists* que já foram acessados. Além disso, os vídeos que estão nas *playlists* também são populares no contexto da comunidade formada em torno da coleção de vídeos *Top-feed* o que faz com que o conteúdo nas *playlist* esteja mais tempo disponível.

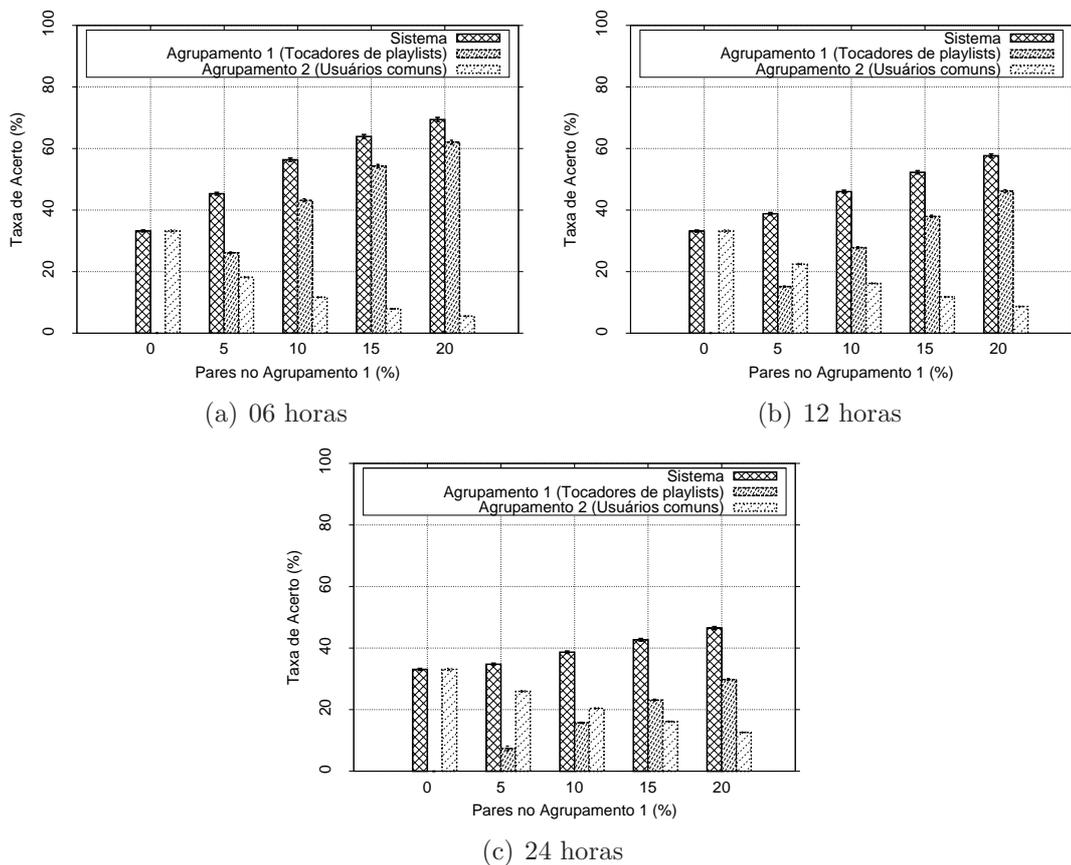


Figura 4.5: Taxa de acerto de pares comuns e tocadores de playlist com três duração de sessão *DOWN* para a coleção de vídeos de entretenimento.

A Figura 4.5 mostra os resultados para a coleção de Entretenimento. Para essa coleção a taxa máxima de acerto foi 79%, para uma sessão *DOWN* de 6 horas e 20% de tocadores de *playlists* no sistema. A contribuição de pares comuns diminuiu de 18% para 5% com o aumento do número de tocadores de *playlists*. Os tocadores de *playlists*

passam a dominar a taxa de acerto quando estes correspondem 5% (10%) (15%) dos pares do sistema, para uma sessão *DOWN* de 6%, 12% e 24% horas. O aumento do número de tocadores de *playlists* e a redução da duração de sessão *DOWN*, gerar uma melhoria na disponibilidade de conteúdo, assim como nos resultados apresentados para coleção de Música.

A Figura 4.6 mostra os resultados da avaliação realizada com as outras coleções: Esporte, Educação, Comédia e Filmes&Animações, quando as sessões *DOWN* duram 24 horas. As conclusões obtidas com as coleções de vídeos de Música e Entretenimento podem ser aplicadas para as outras coleções. Ao aumentar o número de tocadores de *playlists* e reduzir a duração da sessão *DOWN*, ocorre uma melhoria na disponibilidade de conteúdo, embora haja diferença quantitativa em relação ao resultado obtido com as coleções de Música e Entretenimento. Por exemplo, com 10% dos tocadores de *playlists* a contribuição dos pares comuns é maioria no atendimento das requisições na coleção de Música, porém com aquele mesmo número de tocadores de *playlists*, os pares comuns não mostram contribuição similar para as coleções de vídeo de Esporte.

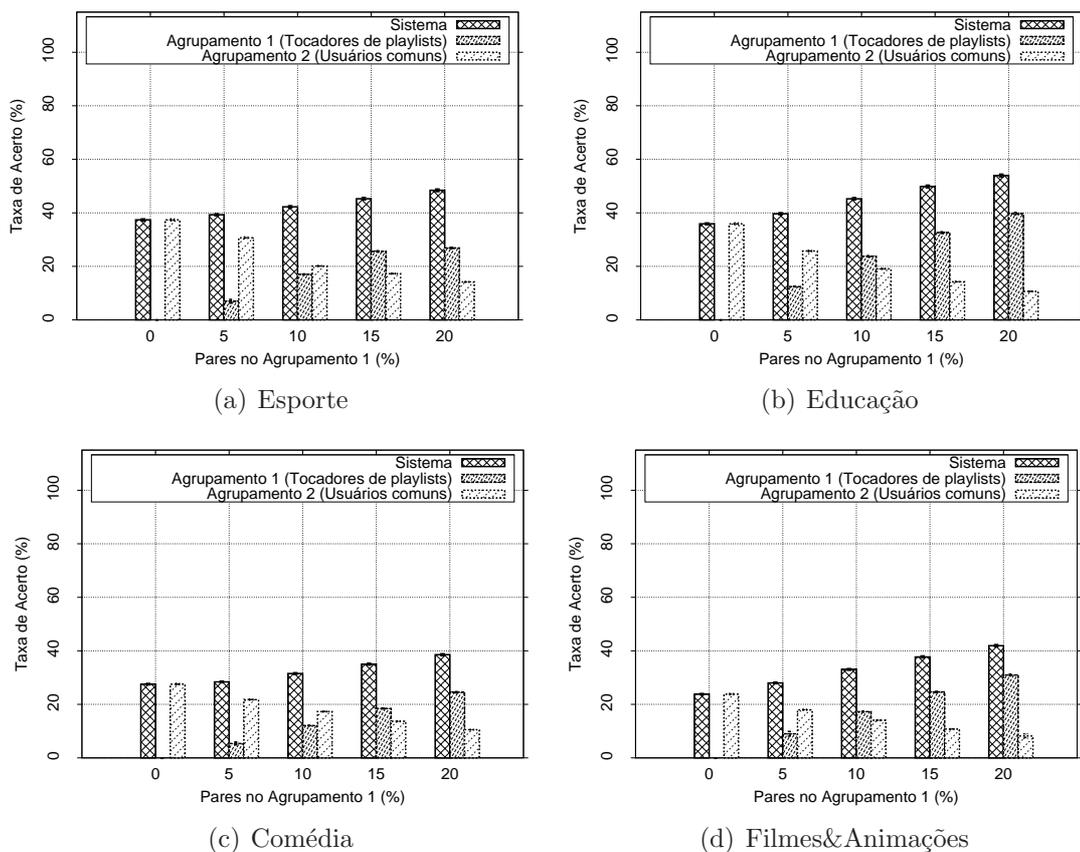


Figura 4.6: Taxa de acerto em usuários comuns e tocadores de playlist com sessão *DOWN* de 24 horas.

No estudo que segue, avalia-se o impacto da saída antecipada de tocadores de

*playlists*, i.e quando ocorre a interrupção da execução da *playlist*, na disponibilidade de conteúdo. Para tal, considerou-se uma configuração do sistema em que as sessões *DOWN* duram Seis e 24 horas, 20% dos pares na comunidade são tocadores de *playlists*, e as saídas prematuras acontecem com taxa  $\beta = 0.25$ ,  $\beta = 0.50$  e  $\beta = 0.75$ . Em outras palavras, em média um tocador de *playlist* irá reproduzir a lista de vídeos ocorrendo a interrupção quando 25%, 50%, e 75% dos vídeos na lista forem reproduzidos. Usou-se uma distribuição geométrica truncada, configurada com os valores de  $\beta$ , para modelar a interrupção das *playlists*. Figura 4.7 mostra o desempenho do sistema na presença de tocadores de *playlists* impacientes quando as coleções de vídeos de Entretenimento e Música foram consideradas.

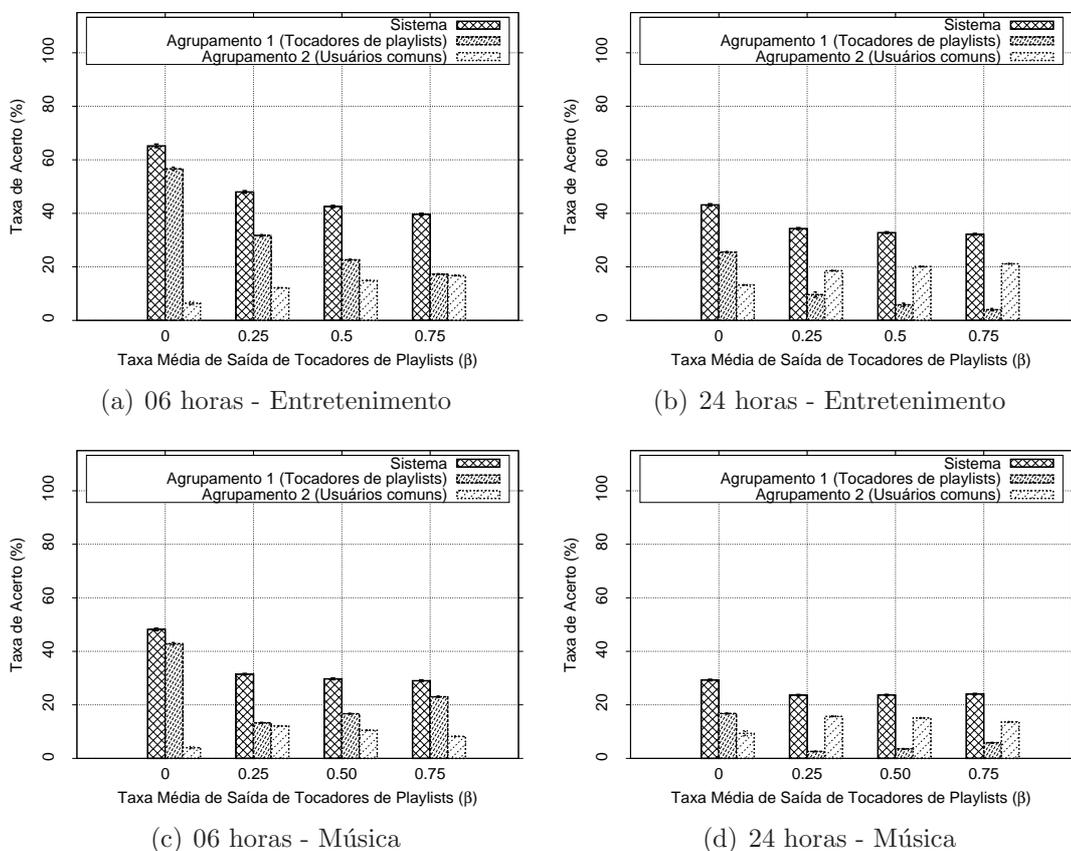


Figura 4.7: Impacto de tocadores de playlists impacientes na disponibilidade de conteúdo (coleções de entretenimento e música).

A conclusão geral é que à medida que as saídas acontecem mais cedo, a disponibilidade de conteúdo é afetada mais fortemente (ver Figura 4.7). Essa redução é mais destacada para o período de desconexão de Seis horas, quando a disponibilidade de conteúdo diminui em 1/3 para as duas coleções avaliadas. A saída prematura dos tocadores de *playlists* do sistema gera a indisponibilidade do conteúdo armazenado somente por eles, tendo-se observado que uma parte significativa dos vídeos mais populares da coleção estão no início das *playlists*. De fato, verificou-se que 8% (6%) dos vídeos que respondem por

80% das visualizações nas coleções *Top-Feed* de Entretenimento e Música encontram-se posicionados no primeiro quarto da *playlist*. Assim, quando o tocador de *playlist* interrompe a reprodução logo após reproduzir o último vídeo do primeiro quarto, todo o atendimento que poderia ser realizado a partir do armazenamento desse conteúdo popular é inviabilizado.

Os resultados apresentados até então consideraram a disponibilidade de conteúdo proporcionada pelos tocadores de *playlists* no sistema como um todo, o que inclui o auto-atendimento e o atendimento de requisições geradas por usuários de ambos os agrupamentos, à medida que o número de tocadores de *playlists* torna-se maior. Embora sejam relevantes para avaliar a sua importância para o sistema, essas medidas não revelam a intensidade dos serviços prestados pelos tocadores de *playlists* aos pares comuns. Para tal, mediu-se a quantidade de requisições geradas pelos usuários comuns e que é exclusivamente servida por tocadores de *playlists*, Figura 4.8, para as coleções de Entretenimento e Música.

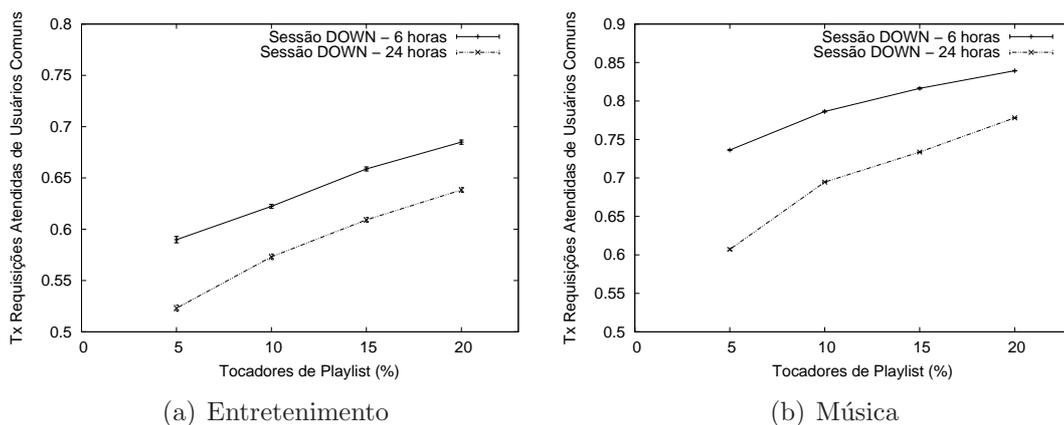


Figura 4.8: Impacto de tocadores de playlists na disponibilidade de conteúdo requisitado por usuários comuns.

Verifica-se que uma parte significativa das requisições do agrupamento de usuários comuns é atendida pelos tocadores de *playlists*. Por exemplo, quando os vídeos de Entretenimento são acessados esse atendimento é de 52% (63%) em um sistema que possui 5% (20%) dos seus pares compondo o agrupamento de tocadores de *playlists* em uma sessão de desconexão de 24 horas. Para a coleção de música, os valores são 60% (75%) quando o sistema possui 5% (20%) dos seus pares compondo o conjunto de tocadores de *playlists*. Esse comportamento é explicado pela intersecção do conjunto dos vídeos mais populares de ambas as coleções *playlist* e *Top-Feed*, que concentram a grande maioria dos acessos (80%), conforme análise apresentada na Tabela 3.2 do Capítulo 3.

Outro fator que influencia o atendimento das requisições dos usuários comuns pelo

agrupamento de tocadores de *playlist*, é o posicionamento dos vídeos mais populares na parte inicial das *playlists*. Por exemplo, verificou-se que 12% dos vídeos que respondem pela maioria dos acessos à coleção *Top-Feed* Entretenimento, 80% dos acessos, estão na primeira metade das *playlists*. Dessa forma, a reprodução da metade final da *playlist* representa a possibilidade de atendimento das requisições pelos tocadores de *playlists*, principalmente aquelas feitas para os vídeos mais populares.

## 4.4 Considerações Finais do Capítulo

Estudos têm demonstrado a importância de pares estáveis no ecossistema de aplicações P2P. Nestes sistemas, a disponibilidade de conteúdo é fortemente afetada pela autodeterminação dos pares sobre a sua participação no sistema. Com isso, pares que se juntam por longas sessões melhoram a disponibilidade de conteúdo.

Neste capítulo, estudou-se a efetividade de pares estáveis (tocadores de *playlists*), na melhoria da disponibilidade de conteúdo em um sistema CDN-P2P que distribui vídeos de curta duração similar aos distribuídos pelo YouTube. A melhoria na disponibilidade de conteúdo foi avaliada em diferentes cenários de rede, i.e. quando um número grande de pares tocadores de *playlists* passam longas sessões conectados, a melhoria na disponibilidade de conteúdo foi de 60% para a coleção de Entretenimento, e de 40% para a coleção de Música. Adicionalmente, em cenários que têm todos os pares com sessão *DOWN* de 24 horas, a melhoria foi de 15% para a coleção de vídeos de Esporte, superou 20% para a coleção de vídeos de Filmes&Animações.

## CAPÍTULO 5

# POLÍTICAS DE FORMAÇÃO DA MALHA EM SISTEMAS CDN-P2P

O estudo realizado no Capítulo 4 mostrou que o ajuntamento de pares estáveis (tocadores de *playlists*) pode melhorar significativamente a disponibilidade de conteúdo em um sistema para distribuição de vídeos similar aos distribuídos pelo YouTube, uma vez que esses pares além da permanecerem por longas sessões conectados no sistema, também acessam e compartilham conteúdos populares.

Tendo-se identificado o potencial dos tocadores de *playlists* na melhoria da disponibilidade de conteúdo, o estudo do presente capítulo investiga de que forma políticas de formação impactam o sistema de distribuição avaliado quando os pares são identificados e agrupados conforme características de participação no apoio à distribuição. Estas políticas constroem e gerenciam a vizinhança dos pares da malha P2P de acordo com critérios previamente definidos, diminuindo assim as demandas sobre o *tracker* da CDN. Em outras palavras, avalia-se qual o desempenho do sistema quando os pares não dependem exclusivamente de uma entidade centralizadora responsável por gerenciar a descoberta de pares, bem como a busca por conteúdo dentro da malha P2P.

### 5.1 Formação e Gerência de Malha P2P

Nesta seção apresentam-se alguns trabalhos expostos na literatura que estudam a influência de mecanismos para formação da malha na parte P2P de sistemas CDN-P2P. Esses mecanismos envolvem o estabelecimento e a gerência das conexões, considerando

diferentes contextos do ecossistema de aplicações P2P.

Em (AL-HAMRA et al., 2009), os autores estudam uma estratégia preemptiva de gerência de vizinhança e comparam seus resultados com a abordagem padrão utilizada em sistemas do tipo BitTorrent. Nesta estratégia, quando um par estabelece o número de máximo de conexões permitidas e recebe um novo pedido de conexão, esse aceita tal pedido desconectando-se de algum de seus atuais vizinhos de forma aleatória, diferentemente da estratégia padrão do BitTorrent onde o pedido de conexão é simplesmente rejeitado. Os resultados numéricos indicam que a estratégia proposta minimiza o nível de clusterização na rede melhorando assim sua robustez e potencializa a convergência para o número máximo de vizinhos dos pares, oportunizando assim maior interação entre eles. Além disso, a estratégia proposta foi capaz de reduzir o diâmetro dentro da malha. Ou seja, seriam necessários poucos saltos para que um conteúdo fosse alcançado em qualquer ponto da rede.

Mais recentemente Joung e Huang (2012c) analisaram de forma mais aprofundada o impacto dos mecanismos de construção e manutenção de topologia em sistemas do tipo do BitTorrent. Os autores avaliaram os diferentes mecanismos dividindo o processo de construção da malha P2P em três fases: seleção do conjunto de pares inicial, atualização do conjunto de pares e gerenciamento das conexões. Além disso, foram propostas e avaliadas abordagens híbridas que integram critérios empregados em diferentes mecanismos. A análise sobre os mecanismos para seleção do conjunto de pares inicial avaliou o impacto gerado pelos diferentes critérios adotados pelo *tracker* para fazer indicações aos pares recém-chegados ao sistema. Tal análise mostrou que a estratégia que adota a aleatoriedade para indicação de pares se apresentou mais eficiente em relação a utilização do enlace, enquanto que a estratégia de indicação baseada na largura de banda disponível nos pares, potencializou a reciprocidade no sistema. No entanto, essa estratégia gerou um maior índice de clusterização no sistema. Sendo assim, os autores avaliaram uma estratégia híbrida onde parte da lista de possíveis conexões indicadas pelo *tracker* é baseada em aleatoriedade e a outra parte baseada na largura de banda. Tal estratégia minimizou o nível de clusterização no sistema, além de manter um nível de reciprocidade satisfatório.

Ainda em (JOUNG; HUANG, 2012c), avaliou-se a performance de diferentes critérios utilizados na renovação e gerência das conexões dos pares. A estratégia híbrida baseada em aleatoriedade e largura banda apresentou melhor eficiência nas taxas de *download* e reciprocidade. Além disso tais estratégias minimizam a carga de trabalho imposta ao *tracker*, uma vez os pares comunicam-se entre si para a escolha de novos vizinhos. Em ambas estratégias, os pares inicialmente solicitam do *tracker* a lista inicial de pares para

interagirem. Essa lista é gerada com base em algum dos critérios escolhidos (aleatoriedade ou largura de banda). Quando os pares precisam atualizar suas vizinhanças, esses também se utilizam de algum do critério previamente definido para então tentar estabelecer novas conexões. Com relação ao gerenciamento das conexões, como observado em (AL-HAMRA et al., 2009), a estratégia preemptiva ajuda a reduzir o diâmetro dentro da malha. No entanto, um pequeno diâmetro não representou a eficiência em relação a utilização de link disponível em cada par.

Em (CHOFFNES; BUSTAMANTE, 2008), os autores consideram a localidade dos pares como critério de indicação de vizinhança. Para tal, um servidor CDN é responsável de coletar informações sobre os pares, para então fazer indicações priorizando pares que estão no mesmo ISP a fim de melhorar a experiência dos usuários e diminuir custos dos ISPs. Os resultados numéricos mostram que a abordagem proposta além de reduzir os custos Inter-ISPs, também é capaz de fornecer aos usuários taxas de *download* satisfatórias. Dentro desse mesmo contexto, (WU et al., 2010), estudam a aplicação de algoritmos genéticos para determinar as melhores conexões para cada par de acordo com alguma métrica, tais como largura de banda de *download* e *upload*.

Moraes e Duarte (2010) estudam a efetividade de um mecanismo de seleção de vizinhos para o contexto de vídeo sob-demanda. Esse mecanismo, chamado de *Lifetime-based Peer Selection*, constrói e monitora a vizinhança baseado no tempo de chegada dos pares. Este assume que pares com tempo de chegada similares estão assistindo à mesma parte do vídeo e, portanto, possuem mais dados em comum para trocar. Os resultados mostram que o LIPS seleciona os melhores vizinhos em relação à eficiência das conexões, garantindo alta continuidade de reprodução do vídeo.

Os trabalhos aqui descritos consideram a heterogeneidade entre os pares da rede P2P através de critérios como largura de banda e tempo de chegada. No entanto, nenhum dos trabalhos avalia de que forma pares que se juntam por longas sessões impactam a estruturação da malha P2P. Ainda, o sistema estudado neste capítulo apresenta uma dinâmica mais intensa, com a grande maioria dos pares participando em sessões de curta duração.

## 5.2 Políticas de Formação da Malha em Sistemas CDN-P2P

Nesta seção, apresentam-se as políticas de formação da malha do sistema de distribuição de conteúdo estudado. Tais políticas estruturam a porção P2P do sistema com base em critérios que são empregados na chegada dos pares no sistema, na construção e na gerência das vizinhanças. Cada par ( $P_i$ ) da porção P2P possui associado a ele um perfil ( $Pf_i$ ), definido a partir dos objetos que ele possui em sua cache, e uma lista de vizinhos ( $L_i$ ), onde  $L_i$  possui um tamanho máximo ( $L_iMax$ ).

---

**Algoritmo 1:** Algoritmo para formação da malha P2P.

---

```
Entrada: ( $P_i$ ,  $StatusP_i$ )
se ( $StatusP_i == Bootstrap$ ) então
     $P_i$  recebe do tracker uma lista com os pares ativos e abre uma tentativa de
    conexão com algum dos pares da lista;
senão
    se ( $StatusP_i == Maintenance$ ) então
         $P_i$  avalia os pedidos de conexão realizados a ele;
        se ( $C_i \leq 390$ ) então
             $P_i$  realiza a busca por um novo vizinho;
            se (existir vizinho disponível) então
                // Par avalia se sua lista de vizinhos está completa
                se ( $L_i == L_iMax$ ) então
                     $P_i$  remove algum de seus vizinhos;
                     $P_i$  estabelece uma conexão com o vizinho escolhido;
```

---

O Algoritmo 1 descreve de forma geral como a malha P2P é formada. Na sua chegada ao sistema o par entra em estado de *Bootstrap*. Nesse estado, o par realiza requisições ao *tracker* solicitando a lista de pares que estão ativos no sistema. De posse dessa lista, o par tenta estabelecer uma conexão com algum dos pares indicados pelo *tracker*. Tendo acessado o sistema, o par utiliza mecanismos de gerência de suas conexões, realizando assim a manutenção (*Maintenance*) da topologia. A manutenção inclui a avaliação dos pedidos de conexões, a busca por novos vizinhos e a avaliação das conexões estabelecidas. A manutenção ocorre em um período ( $C_i$ ), tendo-se para as duas políticas implementadas definido que a descoberta de novos vizinhos ocorre em um intervalo de  $3x C_i$ , sendo  $C_i$  igual a 130 segundos.

### 5.2.1 Política Aleatória

Nessa política (Algoritmo 2) a construção da malha P2P ocorre de maneira aleatória. Ao chegar no sistema o par entra em estado de *Bootstrap* solicitando do *tracker* a lista de pares ativos. De posse dessa lista, o par tenta estabelecer conexão com algum dos pares da lista escolhendo-os de forma aleatória. Essa ideia também é aplicada quando o par entra no estado de manutenção (*Maintenance*) de sua vizinhança e busca por um novo vizinho. Caso seja necessário a substituição de algum de seus atuais vizinhos ( $L_i$ ), o par não realiza qualquer tipo de avaliação em relação a qualidade dos mesmos e escolhe, de forma aleatória, entre aqueles que ele possui conhecimento, quem receberá a sua tentativa de estabelecer uma nova conexão.

---

**Algoritmo 2:** Política aleatória.

---

```
Entrada: ( $P_i$ ,  $StatusP_i$ )
se ( $StatusP_i == Bootstrap$ ) então
  |  $P_i$  recebe do tracker uma lista com os pares ativos e abre uma tentativa de
  | conexão de forma aleatória com algum dos pares da lista;
senão
  | se ( $StatusP_i == Maintenance$ ) então
  | |  $P_i$  avalia os pedidos de conexão realizados a ele;
  | | se ( $C_i == <= 390$ ) então
  | | |  $P_i$  busca por um novo vizinho aleatoriamente dentre os pares ativos;
  | | | se (existir vizinho disponível) então
  | | | | // Par avalia se sua lista de vizinhos está completa
  | | | | se ( $L_i == L_iMAX$ ) então
  | | | | |  $P_i$  remove algum de seus vizinhos de forma aleatória;
  | | | | |  $P_i$  estabelece uma conexão com o vizinho escolhido;
```

---

A intuição por trás desta política é a diversificação da vizinhança dos pares. Tal diversificação (JOUNG; HUANG, 2012a) (XIA; MUPPALA, 2010) impede que os pares mantenham relação sempre com os mesmos vizinhos, impossibilitando a descoberta de pares de maior potencial. Essa estratégia é adotada no mecanismo de incentivo *Optimistic Unchoking* implementado em sistemas BitTorrent.

### 5.2.2 Política Pró-Ativa Baseada em Similaridade

Na política pró-ativa (Algoritmo 3) são propostas três modificações em relação à política aleatória. A primeira modificação estabelece que critérios de similaridade de conteúdo, acessado previamente, sejam usados pelo *tracker* para indicar possíveis vizinhos

ao par recém-chegado. Na política pró-ativa, o par ao chegar no sistema envia seu perfil ( $Pf_i$ ) ao *tracker*, que irá indicar pares com perfis similares. Considerou-se a descrição dos objetos que o par possui em sua cache para caracterizar o seu perfil, e a similaridade ( $Sim$ ) entre os pares é dada por,

$$Sim = \frac{Pf_iA \cap Pf_iB}{Pf_iA \cup Pf_iB} \quad (5.1)$$

onde  $Pf_iA$  é o perfil do par que está em busca de uma nova conexão e  $Pf_iB$  o perfil do par no qual se deseja avaliar a similaridade. Para medir a similaridade ( $Sim$ ) entre os pares utilizou-se o coeficiente de Jaccard (REAL; VARGAS, 1996), que define o grau de interseção entre dois conjuntos.

---

**Algoritmo 3:** Política pró-ativa

---

```

Entrada: ( $P_i$ ,  $StatusP_i$ )
se ( $StatusP_i == Bootstrap$ ) então
    |  $P_i$  recebe lista de pares ativos do tracker ordenada com base em  $Sim$  e abre
    | uma tentativa de conexão com os pares da lista;
senão
    | se ( $StatusP_i == Manutenance$ ) então
    | |  $P_i$  avalia os pedidos de conexão realizados a ele;
    | | se ( $C_i \leq 390$ ) então
    | | |  $P_i$  busca por um novo vizinho enviando seu perfil ( $P_f$ ) para algum de
    | | | seus vizinhos atuais para que este lhe retorne uma lista ordenada com
    | | | base em  $Sim$  ;
    | | | se (existir vizinho disponível) então
    | | | | // Par avalia se sua lista de vizinhos está completa
    | | | | se ( $L_i == L_iMAX$ ) então
    | | | | | par remove algum de seus vizinhos de acordo com o valor de  $Nr_i$ ;
    | | | |  $P_i$  estabelece uma conexão com o vizinho escolhido;
    | | |
    | |
    |

```

---

As outras diferenças entre a política pró-ativa e a aleatória estão na forma como o estado de manutenção (*Maintenance*) é implementado. Ao buscar por um novo vizinho, o par escolhe algum de seus vizinhos atuais, e envia o seu perfil ( $Pf_i$ ) para que esse lhe faça indicações. A Equação 5.1 é usada para construir uma lista com os  $k$  pares mais similares, que é repassada ao par solicitante para que esse tente estabelecer uma nova conexão. Se o par que busca um novo vizinho precisar substituir algum de seus atuais vizinhos( $L_i$ ), ele deverá considerar o nível de satisfação com a sua vizinhança. Tal nível é medido através do número de requisições ( $Nr_i$ ) atendidas por cada um de seus vizinhos. Em outras palavras, o par avalia o nível de contribuição de cada um de seus vizinhos para então remover de sua lista o vizinho que se mostrou menos participativo.

A política pró-ativa tem como principal motivação usar o conceito de similaridade

para aproximar pares que compartilham interesses, o que nem sempre é possível quando a vizinhança se forma de maneira aleatória. Quando pares que possuem interesses similares se juntam (MEKOUAR et al., 2010), o nível de cooperação no sistema pode ser melhorado. Além disso, a taxa de perda de mensagens pode ser reduzida minimizando assim os custos de comunicação (TIRADO et al., 2010).

### 5.3 Resultados Numéricos

Nesta seção, apresentam-se os resultados dos experimentos que avaliaram de que forma as políticas de formação da malha impactam o sistema de distribuição de conteúdo estudado. Analisou-se como as políticas aleatória e pró-ativa estruturam a porção P2P do sistema quando os pares são identificados e agrupados conforme características de participação no apoio a distribuição. Além disso, mediu-se a disponibilidade de conteúdo observando-se a relação entre pares de agrupamentos diferentes e do sistema como um todo.

O sistema avaliado usa uma malha P2P para apoiar a tarefa de distribuição. Neste sistema, existem dois tipos de pares: os pares comuns e os tocadores de *playlist*. Os pares comuns comportam-se de maneira dinâmica, entrando e saindo do sistema quando lhe é conveniente. O outro tipo de par, chamado de tocador de *playlist*, permanece por mais tempo conectados ao sistema reproduzindo um conjunto de vídeos que compõem as *playlists* acessadas por ele. A autonomia dos pares definida pela duração das sessões de conexão (*UP*) e desconexão (*DOWN*) é dada na Tabela 4.3. Neste estudo, considerou-se que a sessões *DOWN* dos tocadores de *playlists* tem duração média de 12 horas. Para os pares comuns a sessão de *UP* dura em média 15 minutos, ou seja, têm-se um sistema bastante dinâmico visto que esses pares representam a maioria da comunidade.

Em todos os cenários avaliados, considerou-se que a comunidade que compõe a malha P2P é formada por 10.000 pares, sendo que cada par pode se conectar a no máximo 30 vizinhos. A capacidade de armazenamento (*buffers*) é de 34 vídeos, o tempo de simulação foi de 5 meses, com tempo de aquecimento de 1 mês, e coleta de dados semanais.

A busca de conteúdo ocorre por inundação. Nessa busca, os pares difundem a requisição por um conteúdo iniciando a exploração a partir de sua vizinhança até uma profundidade previamente estabelecida. As avaliações de disponibilidade de conteúdo consideraram uma inundação de profundidade 4. As coleções de vídeos utilizadas nessas avaliações foram de Entretenimento e Educação.

A Figura 5.1 apresenta as matrizes de conectividade, onde os 2.000 primeiros pares correspondem a tocadores de *playlists*. Cada ponto  $(x,y)$  na matriz representa uma conexão entre os pares  $x$  e  $y$ , sendo que o ponto  $x$  representa uma conexão do tipo cliente e o ponto  $y$  uma conexão do tipo servidor. Uma conexão como cliente, indica que o par requisitou a conexão ao par no qual está conectado. Por outro lado, uma conexão como servidor indica um pedido de conexão aceito.

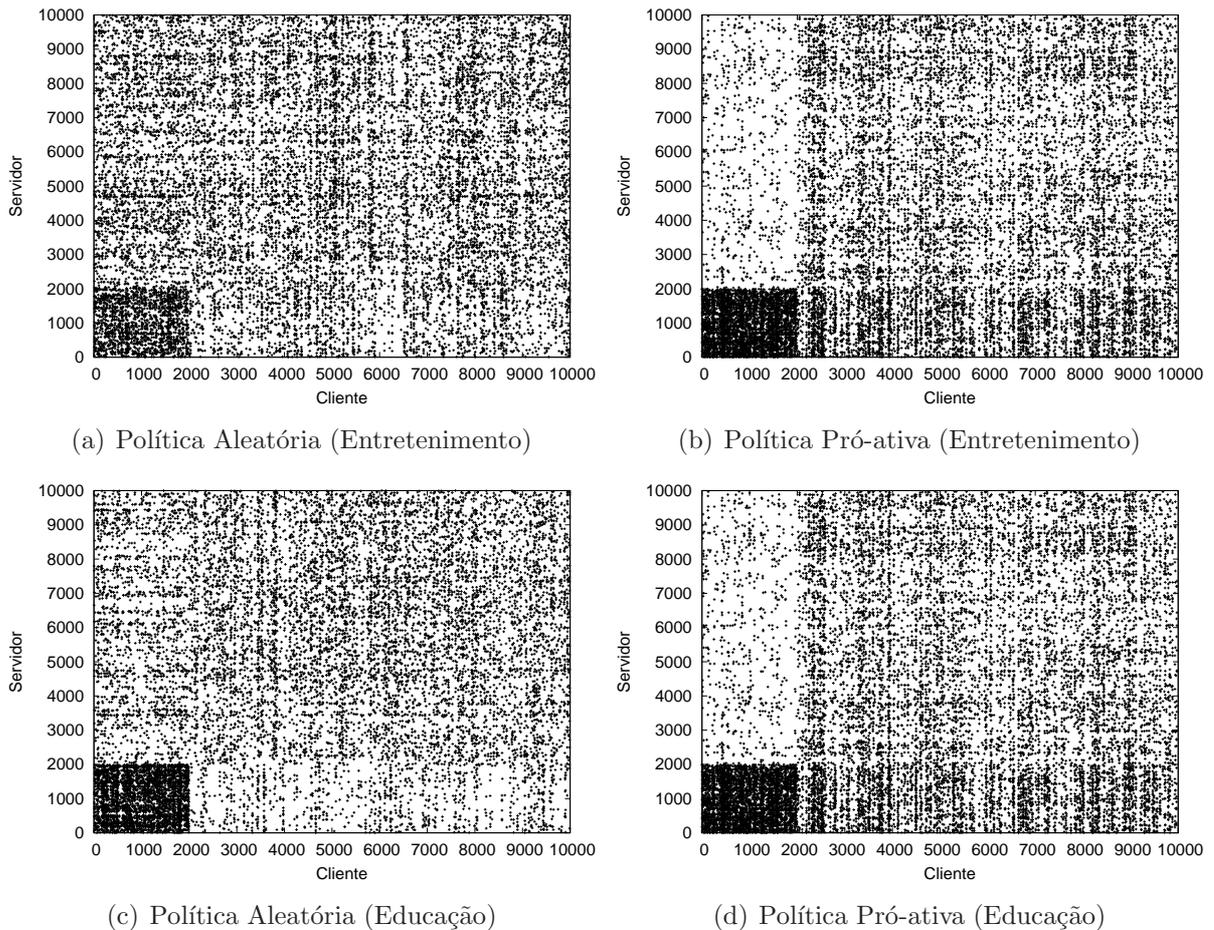


Figura 5.1: Matrizes de conectividade dos pares para as duas políticas de formação da malha.

A análise das matrizes de conectividade mostra que a política aleatória (Figura 5.1(a)) tende a espalhar uniformemente as conexões entre os pares. Tal tendência é explicada pela possibilidade de diversificação da vizinhança dos pares. Como a busca ou desconexão de vizinhos se dá a partir de uma distribuição uniforme, todos os pares envolvidos em ambos os eventos possuem a mesma probabilidade de serem escolhidos.

Observa-se ainda que os tocadores de *playlist* mantêm um maior grau de interação entre si, formando um *cluster* na rede. Este comportamento se justifica pelo fato desses pares permanecerem por mais tempo conectados ao sistema, oportunizando maior

interação entre eles.

Na Figura 5.1(b) é apresentada a matriz de conectividade quando a malha é formada usando a política pró-ativa. Verifica-se que nessa política, o *cluster* formado pelos tocadores de *playlists* tem maior destaque do que na política aleatória. Ou seja, os tocadores de *playlists* se relacionam mais entre eles e estabelecem poucas conexões do tipo cliente com os pares comuns. Por outro lado, observa-se que existe um número mais acentuado de pares comuns que estabelecem conexões, como clientes, dos tocadores de *playlists*. Em outras palavras, a política pró-ativa intensifica a relação entre tocadores de *playlists* e potencializa a relação entre pares comuns e tocadores de *playlists*.

O emprego da similaridade como critério de construção da vizinhança dos pares influencia fortemente na estruturação gerada pela política pró-ativa. Tocadores de *playlists* tendem a manter um maior número de conexões entre eles não apenas devido a sua maior permanência no sistema, mas também pela similaridade do conteúdos compartilhados. Além disso, os tocadores de *playlists* carregam conteúdos populares (como apresentado no Capítulo 3), potencializando a similaridade com pares comuns e oportunizando o estabelecimento de conexões entre os dois agrupamentos de pares.

A baixa interação entre os tocadores de *playlists* como clientes de pares comuns é atribuída: *i*) a ação do *tracker* que prioriza a similaridade no momento da entrada do par no sistema e; *ii*) as avaliações das conexões, que ocorrem periodicamente.

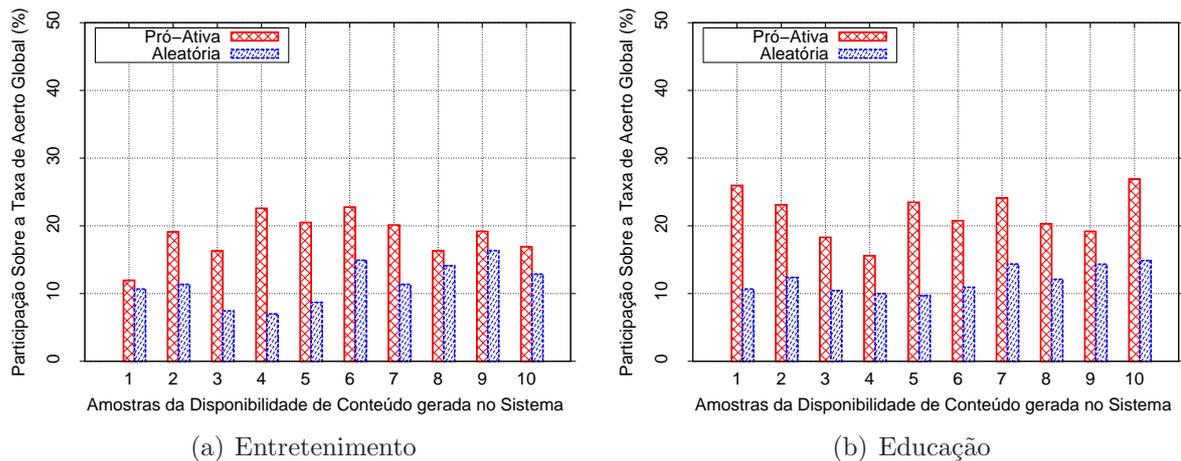


Figura 5.2: Disponibilidade de conteúdo provida pelo tocadores de *playlists* aos pares comuns.

No estudo que segue, avalia-se a influência das políticas de formação no relacionamento de tocadores de *playlists* com os pares comuns. A Figura 5.2 mostra a contribuição na disponibilidade de conteúdo provida pelos tocadores de *playlists* para os pares comuns em relação ao desempenho global do sistema. Os valores de disponibilidade de

conteúdo foram gerados com base em um conjunto de amostras extraídas durante a simulação. Observa-se que a política pró-ativa propicia um maior atendimento dos tocadores de *playlists* para os pares comuns nas duas coleções de vídeos avaliadas. Tal efetividade é decorrente da oportunidade que os pares comuns possuem de manterem alguma relação com pares que têm mais conteúdo popular e permanecem por mais tempo conectados ao sistema. Além disso, o ajuntamento mais intenso dos dois agrupamentos de pares possibilita que consultas de pares comuns que estejam mais distantes dos tocadores de *playlist* alcancem tais pares.

A análise anterior sugere que a aproximação mais intensa dos dois agrupamentos de pares viabiliza que consultas geradas por pares comuns, que se encontram mais distantes cheguem ao *cluster* formado pelos tocadores de *playlists*. De fato, ao observar o número de saltos de consultas geradas por pares comuns que foram atendidas por tocadores de *playlists* verifica-se que elas estão entre 2,5 e 3 saltos para a política pró-ativa (Figura 5.3 (b)), enquanto que na política aleatória estes valores estão entre 2 e 2,5 (Figura 5.3 (a)).

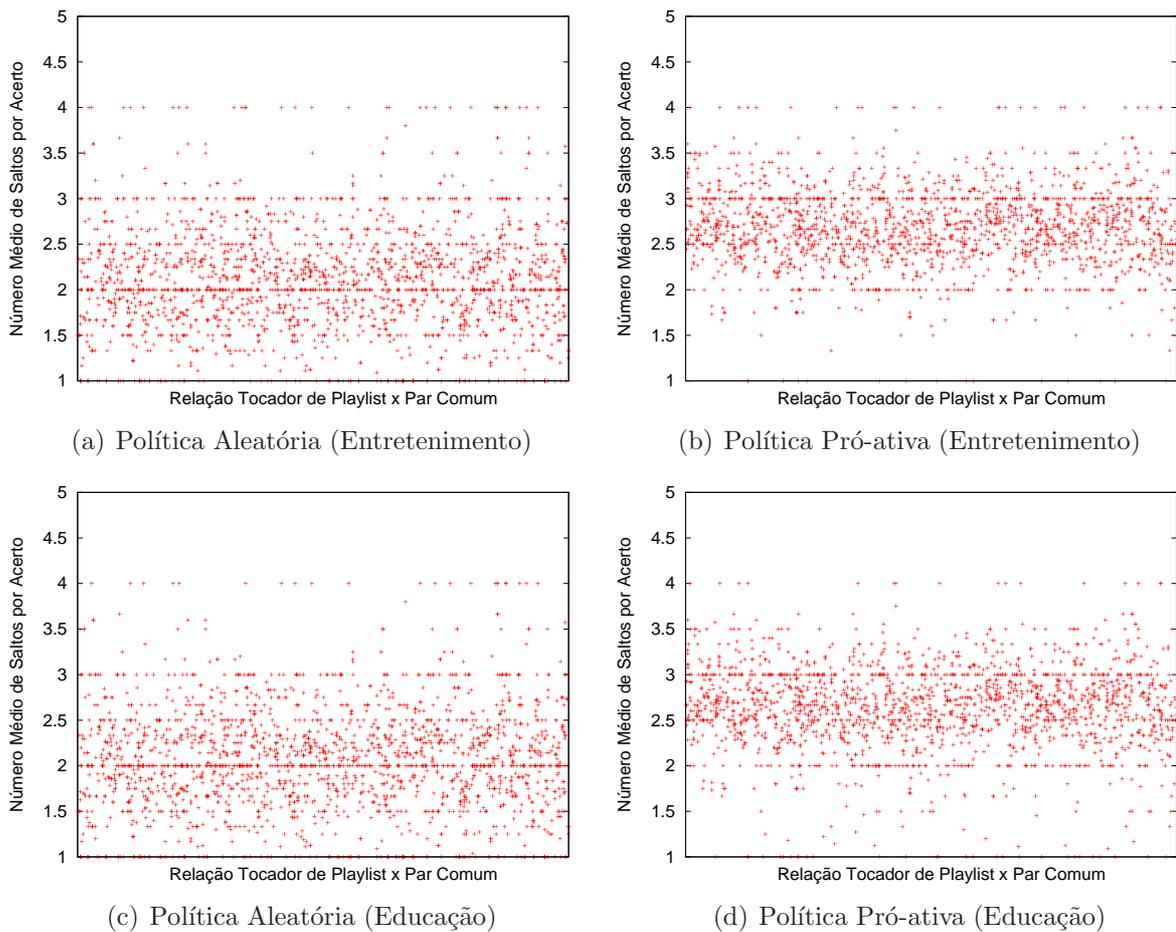


Figura 5.3: Proximidade média a fonte de conteúdo para as duas políticas de formação da malha.

Os resultados apresentados até então consideraram a disponibilidade de conteúdo gerada pela relação entre tocador de *playlist* e par comum, ou seja, o atendimento de tocadores de *playlist* a requisições geradas por pares comuns. No que segue, são descritos os resultados considerando o desempenho global do sistema.

A Figura 5.4 mostra a taxa de acerto global do sistema em função do número de tocadores de *playlists*. Se comparado com o resultado apresentado no Capítulo 4, onde o sistema possui um *tracker* responsável por gerenciar todas as dinâmicas do sistema, incluindo busca por conteúdo, observa-se que o desempenho do sistema decresce como um todo. Tal decréscimo é influenciado diretamente pelo dinamismo do sistema avaliado. Os pares comuns, que representam a maioria da comunidade, permanecem por um curto período conectados, em média 15 minutos. Com isso, os pares possuem pouco tempo para construírem suas conexões de forma a manterem interação com vizinhos que sejam participativos. No entanto, o comportamento gerado pelo aumento do número de tocadores de *playlists* sem mantê-los, ou seja, à medida que o sistema possui mais tocadores de *playlists* a disponibilidade de conteúdo é melhorada.

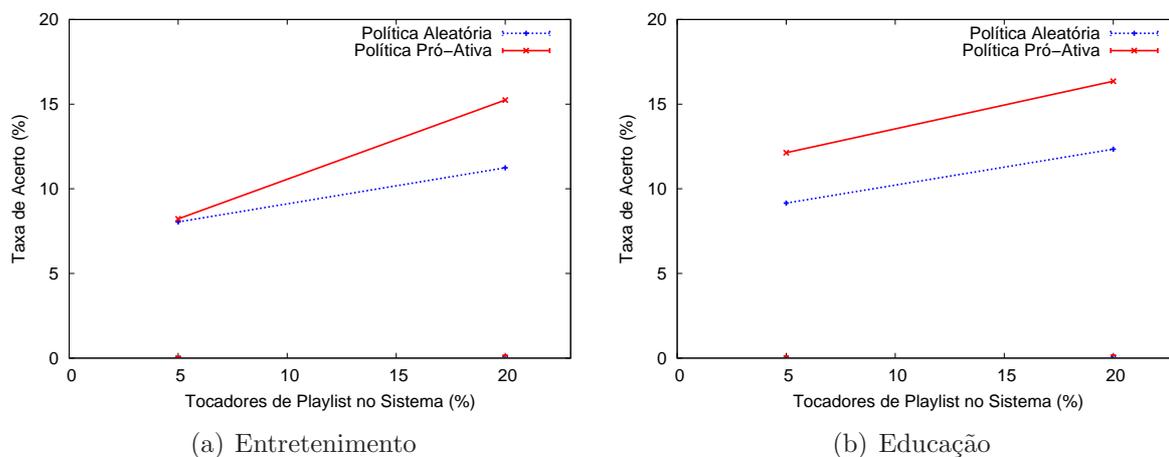


Figura 5.4: Disponibilidade de conteúdo global do sistema para as duas políticas de formação da malha.

Em relação ao desempenho apresentado por cada uma das políticas de formação, verifica-se que quando o sistema possuiu 5% dos pares compondo o agrupamento de tocadores de *playlists* os resultados são similares para coleção de Entretenimento, com taxa de acerto de 8,7% (política pró-ativa) e 8,2% (política aleatória). Porém, quando o sistema possui 20% dos pares compondo o agrupamento de tocadores de *playlists*, a política pró-ativa apresenta uma melhoria superior a 50% em relação à política aleatória. Por outro lado, para coleção de Educação, a política pró-ativa apresenta uma melhoria mais significativa mesmo quando o sistema possuiu apenas 5% dos pares no agrupamento de

tocadores de *playlists*, com taxa de acerto de 12%, enquanto que para política aleatória esse valor foi de 9%. Com 20% dos pares no agrupamento de *playlist* a taxa de acerto de foi 16% (política pró-ativa) e 12% (política aleatória).

A melhoria da política pró-ativa na disponibilidade de conteúdo se justifica pelo fato dessa política gerar um relacionamento mais intenso dos tocadores de *playlists*, fazendo que estes pares construam uma vizinhança mais estável. Além disso, a oportunidade de aproximação dos pares comuns com os tocadores de *playlists* também implica no aumento de desempenho desta política.

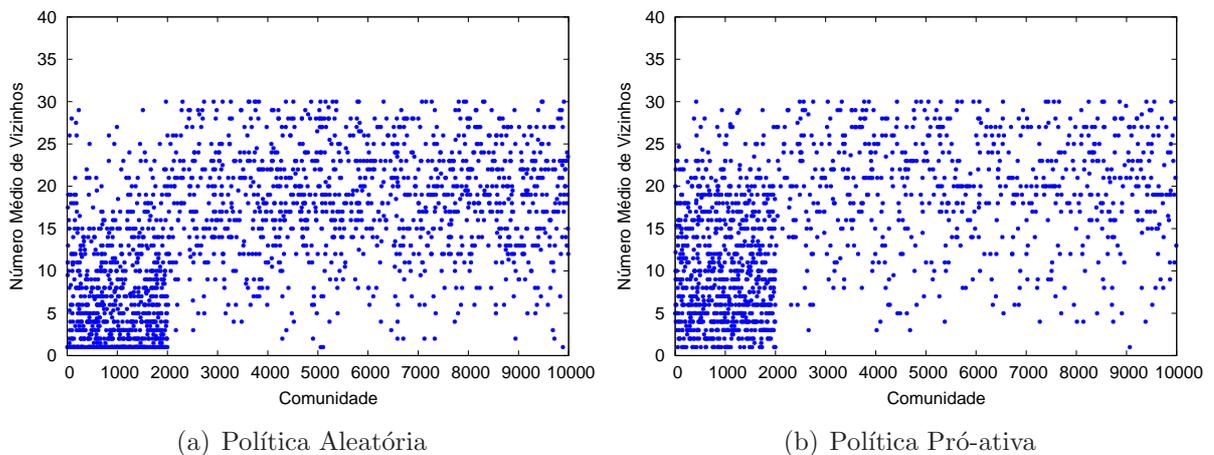


Figura 5.5: Tamanho da Vizinhança dos pares ao longo da simulação.

A maior quantidade de conexões estabelecidas não é reflexo de uma disponibilidade de conteúdo mais efetiva. Na Figura 5.5, apresenta-se o número médio de vizinhos nos quais os pares mantiveram relação ao longo do tempo de simulação. Verifica-se que com a política aleatória, a maioria dos pares concentram suas vizinhanças em torno de 20 a 25 pares, enquanto que para política pró-ativa esses valores estão entre 15 e 20. Assim, avaliar a qualidade dos vizinhos que os pares interagem é um indicativo de que o nível de cooperação pode ser melhorado e, conseqüentemente, o desempenho do sistema como um todo.

## 5.4 Considerações Finais do Capítulo

Neste capítulo estudou-se a influência de políticas de formação da malha em um sistema CDN-P2P quando os pares são caracterizados com estáveis (tocadores de *playlists*) e comuns. Essas políticas estruturam a porção P2P do sistema de acordo com critérios que são empregados na construção e na gerência da vizinhança dos pares, diminuindo assim as exigências sobre o *tracker* da CDN.

Os resultados apresentados mostram que quando os pares são agrupados de acordo com a similaridade de conteúdo que eles compartilham, a disponibilidade de conteúdo pode ser melhorada. No entanto, o dinamismo do sistema estudado gera um impacto significativo na disponibilidade de conteúdo, uma vez que os pares possuem pouco tempo para estruturarem suas vizinhanças de forma a manter relações com vizinhos participativos.

## CAPÍTULO 6

## CONSIDERAÇÕES FINAIS

Aplicações multimídia baseadas em redes impõem aos provedores de conteúdo a provisão de serviços de forma eficiente e com certo grau de escalabilidade. Soluções que utilizam CDN diminuem a latência e melhoram a disponibilidade dos conteúdos, mas têm alto custo financeiro para instalação e manutenção. Soluções baseadas em redes P2P têm sido estudadas, uma vez que são naturalmente escaláveis e robustas. No entanto, a disponibilidade de conteúdo nessas redes é fortemente influenciada pelo número pares ativos, ou seja, o tamanho do sistema. Diante disso, surgiram os sistemas CDN assistidos por redes P2P (CDN-P2P), que capitalizam as vantagens de ambas as tecnologias. Porém, por si só, sistemas CDN-P2P não garantem a efetividade dos serviços, visto que a baixa disponibilidade de conteúdo na parte P2P do sistema, gerada pelo *churn*, pode sobrecarregar o servidor da CDN e, conseqüentemente, degradar a eficiência da distribuição.

Nesta dissertação estudou-se a potencialidade de tocadores de *playlists* na disponibilidade de conteúdo em um sistema CDN-P2P construído para distribuir vídeos similares ao distribuídos pelo YouTube. Para tal, dados reais foram coletados do Site YouTube envolvendo seis categorias de vídeos e *playlists*. Através de análise de tais dados, identificou-se que os vídeos das *playlists* apresentam um padrão de popularidade similar aos vídeos *Top-Feed*, que define que um subconjunto de vídeos representa a maior demanda de recursos. Além disso, percebeu-se que quando os usuários do Youtube montam suas *playlists*, esses já possuem conhecimento suficientes para escolher, entre os vídeos disponíveis nas coleções, os mais relevantes em relação ao tema da *playlist* bem como aqueles que já possuíam um número significativo de visualizações.

Em um sistema onde o servidor da CDN assessora a porção P2P na descoberta

de pares, bem como na busca por conteúdo, a presença dos tocadores de *playlists* gerou uma melhoria na disponibilidade de conteúdo de 60% para a coleção de Entretenimento, e 40% para a coleção de Música. Adicionalmente, em cenários que têm todos os pares com sessão *DOWN* de 24 horas, a melhoria foi de 15% para a coleção de vídeos de Esporte, superou 20% para a coleção de vídeos de Filmes&Animações. Ao avaliar a contribuição dos tocadores de *playlists* para os pares comuns, verificou-se que mais de 50% das requisições geradas por esses pares foram exclusivamente servidas por tocadores de *playlists*, indicando o potencial de tais pares para melhoria na disponibilidade de conteúdo e na redução das demandas sobre os servidores da CDN.

Finalmente, avaliou-se a eficiência de políticas de formação da malha em um sistema CDN-P2P onde os pares são identificados e agrupados como estáveis e comuns. Essas políticas estruturam a malha P2P de acordo com critérios usados na construção e gerência das conexões na qual os pares interagem, minimizando a carga de trabalho sobre o servidor da CDN. Neste sistema, observou-se que a disponibilidade é substancialmente afetada pelo dinamismo gerado pelo curto tempo de sessão do pares que representam o maior conjunto de pares da malha P2P, dificultando assim a interação com vizinhos participativos. A política que emprega a similaridade dos conteúdos compartilhados como critério de construção se mostrou mais eficiente. Em um cenário onde 5% da comunidade que forma a malha P2P é composta por tocadores de *playlists* a disponibilidade de conteúdo foi de 8,7% (política pró-ativa) e 8,2% (política aleatória) para coleção de Entretenimento, enquanto que para coleção de Educação esses valores foram de 12% (política pró-ativa) e 9% (política aleatória). Porém, quando o sistema possui 20% dos pares compondo o agrupamento de tocadores de *playlists*, a política pró-ativa apresenta uma melhoria superior a 50% em relação à política aleatória para as duas coleções avaliadas. Em relação a contribuição entre os dois agrupamentos de pares, verificou-se que na política pró-ativa a contribuição dos tocadores de *playlist* para os pares comuns foi mais intensa, ou seja, os tocadores de *playlists* servem mais os pares comuns se comparado a política aleatória.

## 6.1 Trabalhos Futuros

Como trabalho futuro, relacionado a efetividade dos tocadores de *playlists* na melhoria da disponibilidade de conteúdo, pode-se avaliar se mecanismos de antecipação desenvolvidos para outros contextos, podem ser aplicados em um sistema CDN-P2P. Esses mecanismos tornariam o conteúdo da *playlist* disponível o quanto antes para o par que irá reproduzir a *playlist* e, conseqüentemente, para os outros pares da rede P2P. Além

disso, pode-se estudar mecanismos de incentivo que façam com que os pares permaneçam por mais tempo conectados no sistema.

Em continuidade ao estudo das políticas de formação da malha no desempenho da disponibilidade de conteúdo, outras técnicas de recomendação poderiam ser estudadas, uma vez que filtragem colaborativa mostrou-se incapaz de lidar com a dinâmica do sistema estudado.

## REFERÊNCIAS BIBLIOGRÁFICAS

ACUNTO, L. D. et al. Bittorrent-like p2p approaches for vod: A comparative study. *Computer Networks*, v. 57, p. 1253-1276, 2013.

AKAMAI, T. *Akamai faster forward*. 2013. Disponível em: <<http://www.akamai.com/>>. Acesso em: 12 maio 2013.

AL-HAMRA, A. et al. Swarming overlay construction strategies. *IEEE Proceedings of 18th International Conference on Computer Communications and Networks (ICCCN)*, p. 1-6, 2009.

ANWAR, Z. et al. Leveraging social-network infrastructure to improve peer-to-peer overlay performance: Results from orkut. *Networking and Internet Architecture (cs.NI); Computers and Society*, 2005.

BITTORRENT. *BitTorrent Homepage*. 2013. Disponível em: <<http://www.bittorrent.com/>>. Acesso em: 12 maio 2013.

CARLSSON, N.; EAGER, D. L.; MAHANTI, A. Peer-assisted on-demand video streaming with selfish peers. In: *Proceedings of the 8th International IFIP-TC 6 Networking Conference*. [S.l.: s.n.], 2009.

CHOFFNES, D. R.; BUSTAMANTE, F. E. Taming the torrent: a practical approach to reducing cross-isp traffic in peer-to-peer systems. *Proceedings of the ACM SIGCOMM conference on Data communication*, v. 38, p. 363-374, 2008.

CRANE, R.; SORNETTE, D. Robust dynamic classes revealed by measuring the response function of a social system. *The National Academy of Sciences*, v. 105, p. 15649-1565, 2008.

CRESPO, A.; GARCIA-MOLINA, H. Semantic overlay networks for p2p systems. In: MORO, G.; BERGAMASCHI, S.; ABERER, K. (Ed.). *Agents and Peer-to-Peer Computing*. [S.l.]: Springer Berlin / Heidelberg, 2005, (Lecture Notes in Computer Science, v. 3601). p. 1-13.

DAVIDSON, J. et al. The youtube video recommendation system. In: *Proceedings of the fourth ACM conference on Recommender systems*. New York, NY, USA: ACM, 2010. (RecSys '10), p. 293–296. ISBN 978-1-60558-906-0.

EDGECAST, N. *CDN Solutions*. 2013. Disponível em: <<http://www.edgecast.com/>>. Acesso em: 12 maio 2013.

FALOUSTOS, M.; FALOUTSOS, P.; FALOUTSOS, C. On power-law relationships of the internet topology. *ACM SIGCOMM Computer Communication*, v. 29, 1999.

FUJITA, N. et al. Coarse-grain replica management strategies for dynamic replication of web contents. *Comput. Netw.*, Elsevier North-Holland, Inc., New York, NY, USA, v. 45, p. 19–34, May 2004. ISSN 1389-1286.

GEBERT, S. et al. Internet access traffic measurement and analysis. *ACM Proceedings of the 4th international conference on Traffic Monitoring and Analysis (TMA)*, p. 29–42, 2012.

GNUTELLA. *Gnutella Homepage*. 2013. Disponível em: <<http://www.gnutella.com/>>. Acesso em: 13 maio, 2013.

GOOGLE. *YouTube APIs and Tools*. 2012. Disponível em: <<http://code.google.com/intl/en/apis/youtube/overview.html>>. Acesso em: 09 maio. 2012.

GUO, L. et al. The stretched exponential distribution of internet media access patterns. In: *Twenty-seventh ACM symposium on Principles of distributed computing*. [S.l.: s.n.], 2008. p. 283–294.

HAREESH, K.; MANJIAH, D. H. Design issues and challenges of peer-to-peer video on demand system. In: *First International Conference on Artificial Intelligence, Soft Computing and Applications*. [S.l.]: Technology, 2011. (AIAA-2011, 3), p. 57–67. ISBN 978-1-921987-01-4.

HECHT, F. V.; STILLER, B. Report- and reciprocity-based incentive mechanisms for live and on-demand p2p video streaming. In: *AIMS'10 Proceedings of the Mechanisms for autonomous management of networks and services, and 4th international conference on Autonomous infrastructure, management and security*. [S.l.: s.n.], 2010. p. 81–84.

HUANG, C. et al. Understanding hybrid cdn-p2p: why limelight needs its own red swoosh. In: *Proceedings of the 18th International Workshop on Network and Operating Systems Support for Digital Audio and Video*. New York, NY, USA: ACM, 2008. (NOSSDAV '08), p. 75–80. ISBN 978-1-60558-157-6.

JIANG, W. et al. A new proxy scheme for large-scale p2p vod system. In: *Embedded and Ubiquitous Computing, 2008. EUC '08. IEEE/IFIP International Conference on*. [S.l.: s.n.], 2008. v. 1, p. 512–518.

JOUNG, Y.-J.; HUANG, H.-L. A detailed examination of the overlay construction and maintenance mechanism in bittorrent. *Computer Communications*, v. 35, p. 682–695, 2012.

- JOUNG, Y.-J.; HUANG, H.-L. A detailed examination of the overlay construction and maintenance mechanism in bittorrent. *Computer Communications*, v. 35, p. 682–695, 2012.
- JOUNG, Y.-J.; HUANG, H.-L. A detailed examination of the overlay construction and maintenance mechanism in bittorrent. *Computer Communications*, v. 35, p. 682–695, 2012.
- KWON, O. C.; SONG, H.; UM, T.-W. Overlay multicast tree construction algorithm for stable multimedia service. *EEE Consumer Communications and Networking Conference (CCNC)*, p. 342 – 346, 2011.
- LEE, S. H.; KIM, P.; JEONG, H. Statistical properties of sampled networks. *Physical Review E*, 73, p. 1–7, 2005. Updated in 2009.
- LEE, S. H.; KIM, P.-J.; JEONG, H. Statistical properties of sampled networks. *Phys. Rev. E*, American Physical Society, v. 73, p. 016102, Jan 2006.
- LEVEL3. *Level3 Homepage*. 2013. Disponível em: <<http://www.level3.com/>>. Acesso em: 12 maio 2013.
- LI, B. et al. On the optimal placement of web proxies in the internet. In: *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*. [S.l.: s.n.], 1999. v. 3, p. 1282–1290 vol.3.
- LI, Z. et al. Stability-optimal grouping strategy of peer-to-peer systems. *IEEE Transactions on Parallel and Distributed Systems*, p. 2079–2087, 2011.
- LIU, Y.; GUO, Y.; LIANG, C. A survey on peer-to-peer video streaming systems. *Peer-to-Peer Networking and Applications*, v. 1, n. 1, p. 18–28, mar 2008. ISSN 1936-6450.
- LV, Z.-H. et al. Prose: Proactive, selective cdn participation for p2p streaming. *Journal of Computer Science and Technology*, v. 28, p. 540–552, 2013.
- MEKOUAR, L.; IRAQI, Y.; BOUTABA, R. Impact of peers' similarity on recommendations in p2p systems. *IEEE International Conference on Computer and Information Technology*, p. 393–399, 2010.
- MELO, C. A. V.; LIBÓRIO, J. da M.; FONSECA, N. L. S. da. Object management policies in p2p-cdn networks. *IEEE International Conference on Communications (ICC)*, p. 1996 – 2000, 2012.
- MELO, C. A. V.; VIEIRA, D.; LIBÓRIO FILHO, J. M. Impact of churn on object management policies. In: *Communications (LATINCOM), 2011 IEEE Latin-American Conference on*. [S.l.: s.n.], 2011. p. 1 –6. ISBN 978-1-4673-0277-7.
- MELO, C. A. V.; VIEIRA, D.; LIBÓRIO, J. Impact of churn on object management policies. *IEEE LATINCOM*, 2011.
- MENASCHE, D.; MASSOULIE, L.; TOWSLEY, D. Reciprocity and barter in peer-to-peer systems. *IEEE Proceedings INFOCOM*, p. 1–9, 2010.

- MENASCHE, D. S. et al. Content availability and bundling in swarming systems. *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, p. 121–132, 2009.
- MONTAZERI, A.; BAKBARI. An incentive mechanism for peer-to-peer video streaming. *5th International Symposium on Telecommunications (IST)*, p. 715–720, 2010.
- MORAES, I. M.; DUARTE, O. C. M. B. A lifetime-based peer selection mechanism for peer-to-peer video-on-demand systems. *IEEE International Conference on Communications (ICC)*, p. 1–5, 2010.
- PASSARELLA, A. A survey on content-centric technologies for the current internet: Cdn and p2p solutions. *Computer Communications*, v. 35, n. 1, p. 1–32, 2012. ISSN 0140-3664.
- PATHAN, M.; BUYYA, R.; VAKALI, A. Content delivery networks: State of the art, insights, and imperatives. *Springer - Lecture Notes Electrical Engineering*, v. 9, p. 3–32, 2008.
- PPLIVE. *PPLive Homepage*. 2013. Disponível em: <<http://www.pplive.com>>. Acesso em: 12 maio 2013.
- QIU, L.; PADMANABHAN, V.; VOELKER, G. On the placement of web server replicas. In: *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*. [S.l.: s.n.], 2001. v. 3, p. 1587–1596 vol.3.
- REAL, R.; VARGAS, J. M. The probabilistic basis of jaccard's index of similarity. *SYSTEMATIC BIOLOGY*, v. 45, p. 380–384, 1996.
- RIPLEY, B. D. *Stochastic Simulation*. USA: Wiley, 1987. (Wiley Series in Probability and Statistics). ISBN 0-471-81888-4.
- SACHA, J. et al. Discovery of stable peers in a self-organising peer-to-peer gradient topology. In: *Proceedings of the 6th IFIP WG 6.1 international conference on Distributed Applications and Interoperable Systems*. [S.l.: s.n.], 2006.
- SANDVINE. *Fall 2011 global internet phenomena report*. 2011.
- SPANOUDAKIS, M.; HADJIEFTHYMIADES, S. Adaptive partial cdn replication. *IEEE Symposium on Computers and Communications (ISCC)*, p. 86–91, 2009.
- SUTTON, R.; BARTO, A. Reinforcement learning: An introduction. *MIT Press, Cambridge*, 1998.
- TENZEKHTI, F.; DAY, K.; OULD-KHAOUA, M. Replication algorithms for the world-wide web. In: *16th Annual International Symposium on High Performance Computing Systems and Applications*. [S.l.: s.n.], 2002. p. 84 – 89.
- TIRADO, J. M. et al. Affinity p2p: A self-organizing content-based locality-aware collaborative peer-to-peer network. *Computer Networks*, v. 54, p. 2056–2070, 2010.
- VAKALI, A.; PALLIS, G. Content delivery networks: status and trends. *Internet Computing, IEEE*, v. 7, n. 6, p. 68–74, nov.-dec. 2003. ISSN 1089-7801.

- VU, Q. H. et al. *Peer-to-Peer Computing: principles and applications*. New York, USA: Springer Berlin Heidelberg, 2010. ISBN 978-3-642-03513-5.
- WANG, F.; LIU, J.; XIONG, Y. Stable peers: Existence, importance, and application in peer-to-peer live video streaming. *IEEE INFOCOM*, p. 1364–1372, 2008.
- WANG, F.; LIU, J.; XIONG, Y. On node stability and organization in peer-to-peer video streaming systems. In: *IEEE Systems Journal*. [S.l.: s.n.], 2011. v. 5, p. 440–450.
- WAUTERS, T. et al. Load balancing through efficient distributed content placement. *Next Generation Internet Networks*, p. 99–105, 2005.
- WU, J.; RAVINDRAN, K. Optimization algorithms for proxy server placement in content distribution networks. *IEEE International Symposium on Integrated Network Management-Workshops, 2009. IM '09*, p. 193–198, 2009.
- WU, T.; LI, M.; QI, M. Optimizing peer selection in bittorrent networks with genetic algorithms. *Future Generation Computer Systems*, v. 26, p. 1151–1156, 2010.
- XIA, R.; MUPPALA, J. A survey of bittorrent performance. *IEEE Communications Surveys and Tutorials*, p. 140–158, 2010.
- YAO, Z. et al. Modeling heterogeneous user churn and local resilience of unstructured p2p networks. In: *Proceedings of the Proceedings of the 2006 IEEE International Conference on Network Protocols*. Washington, DC, USA: IEEE Computer Society, 2006. p. 32–41. ISBN 1-4244-0593-9.
- YIN, H. et al. Design and deployment of a hybrid cdn-p2p system for live video streaming: Experiences with livesky. In: *17th ACM international conference on Multimedia*. [S.l.: s.n.], 2009.
- YOUTUBE. *Youtube statistic page*. 2013. Disponível em: <[http://www.youtube.com/t/press\\_statistics](http://www.youtube.com/t/press_statistics)>. Acesso em: 29 abr. 2013.
- ZHOU, S. Power law modelling of internet topology. *Complex Sciences Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, p. 2090–2098, 2009.