

Observando o Universo BitTorrent Através de Telescópios

Rodrigo Brandão Mansilha, Alan Mezzomo, Giovani Facchini,
Luciano Paschoal Gaspary, Marinho Pilla Barcellos

¹Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brasil

Resumo. *Trabalhos recentes na literatura indicam que o BitTorrent é o protocolo de compartilhamento de arquivos com maior popularidade, sendo responsável por mais de 45-78% de todo tráfego par-a-par, dependendo da localização geográfica. Apesar da importância dessas estatísticas e o crescente interesse por “redes baseadas em enxames”, ainda pouco se sabe sobre a dinâmica do ecossistema BitTorrent, principalmente devido à natureza limitada – em termos de cobertura, riqueza de detalhes e exatidão – dos métodos de monitoramento empregados para este fim. Para atender a esta demanda, propõe-se um modelo de informações de gerenciamento e uma arquitetura de monitoramento que permite a combinação flexível de diferentes estratégias para observar o “Universo BitTorrent”. Para avaliar a viabilidade conceitual e técnica, implementou-se um protótipo da arquitetura, e o mesmo foi usado para instanciar tarefas de monitoramento que combinam diferentes subconjuntos de estratégias.*

Abstract. *Recent analysis of the latest peer-to-peer (P2P) trends worldwide shows that BitTorrent is the most popular file sharing protocol, responsible for more than 45-78% of all peer-to-peer P2P traffic, depending on geographical location. Despite the importance of these statistics and the growing interest for swarm-based networked systems, very little is known about the dynamics of the BitTorrent “ecosystem”, mainly due to the limited nature – in terms of coverage, accuracy and richness of detail – of the monitoring methods designed to this end. To tackle this issue, we propose a management information model and a monitoring architecture that allows a flexible combination of strategies to observe the “BitTorrent Universe”. To show concept and technical feasibility, we have implemented a prototypical implementation of the architecture, and as an example instantiated monitoring tasks combining a different subset of monitoring strategies.*

1. Introdução

Redes *Peer-to-Peer* (P2P) permitem que recursos sejam compartilhados entre usuários de forma eficiente e escalável. Dentre as classes de aplicações P2P, a de compartilhamento de arquivos é provavelmente a mais popular atualmente [Karakaya et al. 2009]. Incontestáveis exemplos de sucesso na Internet, entre eles o BitTorrent, têm sido acompanhados com grande interesse por parte da comunidade científica.

Apesar de nos últimos anos terem sido publicados muitos estudos sobre redes BitTorrent ([Izal et al. 2004, Pouwelse et al. 2005, Guo et al. 2007, Dale and Liu 2007, Zhang et al. 2009, Zhang et al. 2010]), ainda pouco se sabe sobre o real funcionamento delas e padrões de comportamento de seus usuários. Informações sobre compartilhamento de conteúdo em redes BitTorrent são úteis por uma série de razões. Por exemplo,

Este trabalho foi desenvolvido com suporte da Rede Nacional de Ensino e Pesquisa (RNP).

comercialmente, podem amparar campanhas de *marketing* baseadas em popularidade de conteúdo, estimar perdas financeiras com cópias ilegais [Chow et al. 2007] e auxiliar *Internet Service Providers* na investigação de métodos que minimizem o custo do tráfego BitTorrent. Também podem auxiliar instituições da justiça na detecção de certos tipos de ataques efetuados contra redes BitTorrent e no combate à pedofilia. Apesar desses benefícios potenciais, os métodos existentes para observação de redes BitTorrent deixam a desejar. As informações divulgadas frequentemente são pobres em detalhe, cobertura (quantidade de redes e usuários) e/ou exatidão. Além disso, a inexistência de um modelo de informações do BitTorrent dificulta intercâmbio de dados entre a comunidade científica.

Três fatores explicam a atual falta de conhecimento sobre a dinâmica do *universo BitTorrent*. Primeiro, sua escala planetária, onde milhões de usuários operam em nível de aplicação. A dimensão, complexidade, heterogeneidade e incerteza de tal universo provêm oportunidades limitadas de observação constituindo obstáculos para monitoração efetiva. Segundo, em contraste com outras redes P2P de compartilhamento de arquivos como Gnutella e Kazaa, no BitTorrent a rede é composta por milhões de redes sobrepostas (*overlays*) desconexas menores denominadas “enxames”, sendo o primeiro desafio “chegar” a cada enxame de interesse. Por último, o protocolo BitTorrent nasceu de uma implementação (por Cohen) que foi sendo incrementada, estendida e testada por usuários, ao invés do uso de uma metodologia formal ou especificação. Isto levou à criação de diversas implementações distintas de agentes de usuário e extensões ao protocolo. Consequentemente, para monitorar o Universo BitTorrent efetivamente é necessário contemplar os diversos agentes e protocolos e acompanhar suas mudanças.

Portanto, antes que as informações sobre as redes BitTorrent possam ser extraídas consistentemente e confiavelmente, desafios de pesquisa devem ser superados. Neste contexto, este trabalho apresenta como contribuição principal a proposta de uma arquitetura flexível para monitoramento eficiente do “universo BitTorrent”. Foram identificadas e acopladas à arquitetura diversas estratégias de observação, que flexivelmente podem ser combinadas de acordo com as informações desejadas, visando a obtenção eficiente de resultados.

O restante deste trabalho está organizado da seguinte forma. A Seção 2 apresenta os elementos do universo BitTorrent e discute estratégias de monitoramento. As mesmas servem de base para o projeto de uma arquitetura, descrita na Seção 3. A Seção 4 resume aspectos práticos de monitoramento mais relevantes, enquanto a Seção 5 detalha a metodologia experimental usada (com enxames reais na Internet) e discute os resultados. A Seção 6 apresenta trabalhos relacionados a este, e a Seção 7 encerra o artigo com considerações finais e propostas para trabalhos futuros.

2. Estratégias para Monitoramento do Universo

Esta seção define o Universo BitTorrent (Subseção 2.1) e identifica estratégias para monitoramento dos seus componentes (Subseção 2.2). A seguir, discute os custos das estratégias (Subseção 2.3) e argumenta sobre vantagens e desvantagens de cada uma (Subseção 2.4).

2.1. Universo BitTorrent

O universo BitTorrent é composto por **enxames**, **pares**, **rastreadores** e **conteúdos**. Um par é um agente de usuário que executa o protocolo e participa de um ou mais enxames, de acordo com o conteúdo que deseja compartilhar. Um par é nomeado semeador, quando

possui conteúdo completo, ou sugador, caso contrário. Para ingressar em um enxame, o par tipicamente contata o rastreador e como resposta recebe uma lista de IPs de pares (*peer list*) que participam do enxame. Portanto, o rastreador atua como ponto de encontro. Alternativamente, existem duas extensões do protocolo que vêm sendo amplamente adotadas. A primeira, denominada *Peer Exchange* (PEX), permite que pares façam o intercâmbio de lista de pares, e a segunda, geralmente chamada de “*trackerless*”, permite que os pares se encontrem através de tabelas *hash* distribuídas (DHT). No âmbito deste trabalho, por “conteúdo” subentende-se “conteúdo digital”, que é diferente de “conteúdo informacional”. Enquanto o primeiro significa um conjunto específico de arquivos digitais, o segundo significa a informação disponibilizada nos arquivos. Por exemplo, dois torrents podem se referir à um mesmo conteúdo informacional, porém a conteúdos digitais diferentes, como no caso de torrents de uma mesma música codificada em duas taxas de compressão diferentes. O conteúdo compartilhado é organizado em peças. Uma descrição detalhada do protocolo pode ser encontrada em [Konrath et al. 2007].

Para participar de um enxame, um agente de usuário utiliza os metadados disponíveis no respectivo arquivo torrent. Esse arquivo contém informações sobre as peças (para cada peça, seu *hash* e tamanho) que formam o conteúdo e arquivos (nomes e tamanhos). Para distribuir um conteúdo, um par deve gerar um torrent e torná-lo público. Os torrents estão tipicamente disponíveis em sítios dedicados a promover o compartilhamento de arquivos em redes BitTorrent, os quais chamamos de “comunidades”. Um exemplo dessas comunidades é a IsoHunt. Além de publicar torrents, algumas comunidades disponibilizam rastreadores. As comunidades podem ser do tipo “aberta” ou “fechada”, sendo que neste último caso o acesso é restrito a membros cadastrados. Algumas comunidades atuam primariamente como agregadoras de torrents, indexando informações para permitir buscas e apontando para torrents em outras comunidades.

A Figura 1 ilustra o universo BitTorrent, apresentando três cenários diferentes para demonstrar os tipos de formação possíveis com conteúdos, rastreadores e pares. Primeiro, *enxame 1*, que compartilha *conteúdo 1*, mostra o cenário em que um determinado conteúdo é compartilhado por apenas um enxame. Segundo, *enxame 2* e *enxame 3* ilustram o caso em que enxames possuem pares em comum. Note-se que esses enxames são completamente independentes entre si. Por último, *enxame 4* e *enxame 5* exemplificam o caso em que dois ou mais enxames distintos compartilham o mesmo conteúdo. Este cenário pode surgir em dois casos: (a) quando o tamanho das peças é diferente entre os enxames; ou (b) quando o tamanho das peças é igual, porém os conjuntos de rastreadores empregados são mutuamente exclusivos.

2.2. Estratégias

Há diferentes estratégias para extrair informações do universo BitTorrent. Pode-se dividi-las em três grupos de acordo com a fonte de informação.

Comunidades. De maneira geral, as comunidades BitTorrent existem para fornecer arquivos torrent e permitir a interação entre usuários com os mesmos interesses. Comunidades apresentam variados tipos de informação relacionados aos conteúdos, rastreadores e quantidade de pares. Trabalhos como [Andrade et al. 2005, Pouwelse et al. 2005] são exemplos que aplicam a estratégia de monitorar comunidades para obter uma visão ampla do universo.

Rastreadores. Uma outra estratégia é monitorar os rastreadores, que disponibilizam informações mais atualizadas e precisas que as comunidades, assim como in-

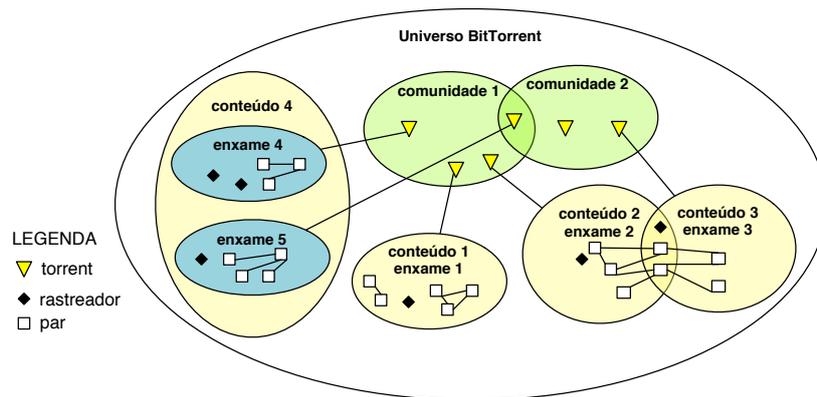


Figura 1. Exemplo de componentes e relações no universo de redes BitTorrent

formações sobre si e sobre os pares (potencialmente apenas um conjunto de endereços IP). Na literatura, são encontradas três variações desta estratégia. A primeira é baseada na análise de *logs* do rastreador, quando disponíveis; um exemplo é encontrado em [Izal et al. 2004]. A segunda variação da estratégia é observar continuamente o conteúdo das páginas Web dinamicamente geradas pelos rastreadores, quando disponíveis; um exemplo é [Andrade et al. 2005]. A terceira variação é registrar-se nos rastreadores usando agentes modificados; um exemplo é encontrado em [Pouwelse et al. 2005]. A quantidade de informação disponibilizada pelo rastreador de uma comunidade dependerá do tipo de conteúdo que está sendo compartilhado e se a comunidade é aberta ou fechada.

Pares. A terceira e última estratégia é monitorar os pares diretamente. Existem duas abordagens, uma ativa e outra passiva. A abordagem passiva para monitorar redes P2P é usada em [Saroiu et al. 2002, Horng et al. 2006], e consiste em capturar e processar pacotes que trafegam nos *enlaces* de rede utilizados pelos pares observados. Em contraste, a abordagem ativa se baseia em instrumentar agentes de usuário que se conectam com pares e extraem informações. Há três variações para a abordagem ativa, todas com o mesmo requisito para implementação e utilização: receber como entrada informações sobre os enxames, ou seja, o endereço IP dos pares do enxame ou, alternativamente, do rastreador, a partir do qual pode-se obter endereços de pares. As três variações diferenciam-se quanto ao nível de mensagens trocadas: na primeira ([Pouwelse et al. 2005, Iosup et al. 2006]), o agente conecta com pares remotos, efetua o *handshake*, troca o mapa de peças e encerra a conexão em seguida; na segunda, mantém conexão com os pares, mas não efetua troca de dados; e na terceira ([Legout et al. 2007]), um par participa ativamente do enxame, inclusive compartilhando dados.

2.3. Custo das Estratégias

As estratégias apresentam diferentes custos e possibilidades de configuração. O custo pode ser mensurado, por exemplo, em termos de utilização de memória, carga de CPU e recursos de rede.

O custo de cada combinação de estratégias é estimado como o produto do custo das escolhas feitas em relação a cobertura, riqueza de detalhe e exatidão. A **cobertura** refere-se ao número de elementos a serem observados, à localização geográfica (por exemplo, todos os pares da Alemanha), intervalo de tempo (por exemplo, todos os torrents publicados no mês) ou tipo de conteúdo (por exemplo, somente livros). A **riqueza de detalhe** corresponde ao número de atributos de interesse, isto é, às variáveis a serem

coletadas; alguns exemplos são o número de pares envolvidos, número e taxas de *downloads*, porção de pares que têm suporte a uma dada característica como criptografia ou PEX, etc. Por último, a **exatidão** representa a qualidade das informações extraídas das redes BitTorrent. A exatidão é afetada por dois parâmetros principais: frequência (isto é, período em que amostras estão sendo coletadas) e tamanho da amostra (relativo ao percentual da população consultada).

2.4. Escolha de Estratégias

Limitar o monitoramento às comunidades traz como vantagem mensurável a economia de recursos de rede, além da não intrusividade no enxame, pois o custo de rede associado a este monitoramento refere-se apenas ao *download* de um pequeno número de páginas html, resultando em poucos kilobytes. Entre as desvantagens está a ausência de informações específicas sobre os pares, que as demais estratégias são capazes de obter. Mesmo que a quantidade de pares que compartilham determinado torrent seja mostrada, nada pode ser presumido, visto que a frequência de atualização dessa informação não é publicada (está potencialmente desatualizada).

A vantagem de monitorar rastreadores em comparação com o monitoramento de comunidades é a possibilidade de identificar a população de pares, o que aumenta o nível de riqueza significativamente. Em comparação com monitoramento de pares, as vantagens são a menor intrusividade e o menor custo para obtenção de informações. Entre suas desvantagens, há a ausência de informações relativas ao conteúdo compartilhado e informações detalhadas sobre o estado atual dos pares e das peças disponíveis. A exatidão é relativamente baixa, pois o tempo com que o rastreador é periodicamente contatado por um par é tipicamente quinze minutos, podendo passar de uma hora.

A maior vantagem de se monitorar os pares em comparação com as outras estratégias é alcançar o maior nível de riqueza de detalhe possível. Entretanto, monitorar pares é a estratégia que implica o maior custo. Como existem duas variantes desta estratégia, monitoração passiva e ativa, elas são comparadas a seguir. A principal vantagem de se usar monitoramento passivo é a menor intrusividade no enxame; além disso, o único custo de rede vem da transferência de logs para a “estação de gerenciamento”. As desvantagens são necessitar poder computacional suficiente para inspecionar todos os dados gravados, a impossibilidade virtual de identificação de tráfego criptografado e a dificuldade para se instanciar pontos de monitoramento passivos em todas as sub-redes desejáveis. Logo, o monitoramento passivo é mais adequado a ambientes menores e controlados, como dentro de um sistema autônomo. Para todas as outras situações, o uso de agentes de usuário modificados é provavelmente mais apropriado.

Dentre as estratégias, a primeira e a segunda demandam apenas acesso aos sítios e ao rastreador, o que tipicamente não apresenta dificuldades. Por outro lado, seus resultados podem deixar a desejar em riqueza de detalhes. A terceira estratégia, monitorar diretamente pares, tende a ser a mais rica em informações, porém é em princípio a menos escalável por demandar o estabelecimento de conexões com pares. Individualmente, elas representam uma parte do problema atacado neste trabalho, pois tipicamente limitam-se a partes específicas do universo. Se combinadas, as estratégias podem fazer parte de uma solução mais completa.

O TorrentU combina estratégias que possibilitam a extração das informações do universo BitTorrent. O ponto principal que difere este trabalho dos demais (mais detalhes na Seção 6, de trabalhos relacionados) é a flexibilidade com respeito à cobertura, riqueza

de detalhe e exatidão das informações, permitindo que peças de informação sejam obtidas através da combinação ideal de um conjunto de estratégias.

3. Observação do Universo com TorrentU

Esta seção oferece uma visão geral do TorrentU. A Subseção 3.1 apresenta o modelo de informações projetado para gerenciar todos os dados do BitTorrent dentro do universo, e a Subseção 3.2 descreve a arquitetura conceitual da solução proposta.

3.1. Modelo de Informações

Um modelo de informações do universo BitTorrent foi criado para estruturar, ligar e racionalizar as informações sobre comunidades, torrents, rastreadores, pares, enxames, etc. Ele é baseado em um subconjunto do Common Information Model (CIM), definido pelo Distributed Management Task Force (DMTF), e no modelo par-a-par, proposto por Doyen *et al.* [Doyen et al. 2004]. Até onde se tem conhecimento, o presente artigo é o primeiro trabalho com intuito de formalizar e organizar, através de um modelo de informações, os vários conceitos associados ao sistema BitTorrent, compreendendo uma contribuição adicional deste trabalho.

A Figura 2 ilustra uma visão simplificada do modelo. As classes em branco contornadas com linhas sólidas e trastejadas designam, respectivamente, classes originadas do CIM e do modelo de Doyen *et al.*. Por outro lado, as classes em cinza são aquelas propostas para expressar os conceitos de BitTorrent e seus relacionamentos. Devido a restrições de espaço, não é possível detalhar as relações ilustradas na figura.

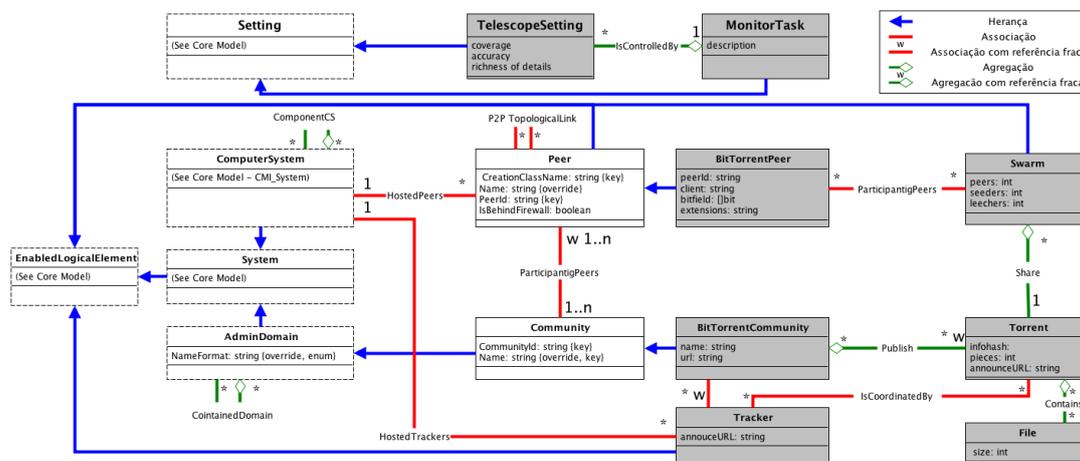


Figura 2. Modelo de Informações do BitTorrent

3.2. Arquitetura

A arquitetura é composta por dois componentes principais, denominados TorrentU Observer e TorrentU Telescope, que interagem entre si, com o usuário e com o universo, conforme a Figura 3.

TorrentU Observer. Este componente faz o papel de *front-end*, isto é, gerente da aplicação, permitindo que o operador configure o sistema e observe os dados coletados em tempo real (assim como o histórico dos dados). Além disso, o Observer deve determinar um conjunto de estratégias e parâmetros a serem utilizados em função de um conjunto objetivo de dados a serem coletados e um conjunto restritivo de recursos disponíveis.

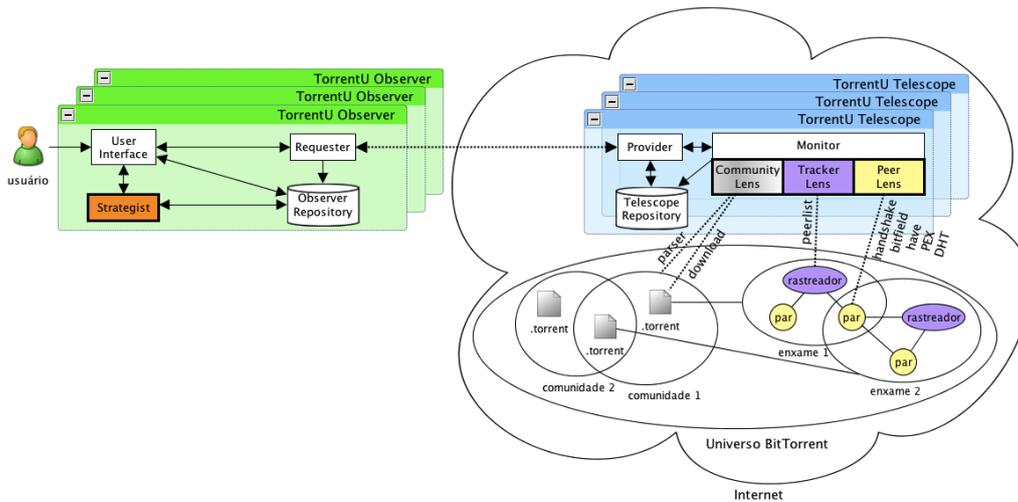


Figura 3. Arquitetura TorrentU

O Observer é composto por quatro subcomponentes: User Interface, Requester, Strategist e Observer Repository. O User Interface é o subcomponente responsável pela interação com o usuário. O Requester é a interface de comunicação com o Telescope. O Strategist ajuda a determinar o melhor conjunto de estratégias e atributos, dada uma certa tarefa de monitoramento. Por fim, o Observer Repository armazena, seguindo o modelo de informações apresentado na subseção anterior, os resultados obtidos pelo Telescope, mantendo-os disponíveis para o User Interface. Esse repositório é particularmente importante para dados de séries temporais, pois “fotografias” são periodicamente criadas e precisam ser armazenadas para posterior recuperação e análise dos dados.

TorrentU Telescope. Este componente é responsável pelo monitoramento do universo BitTorrent e pelo retorno de resultados de acordo com as requisições enviadas pelo Observer. Ele é composto por três subcomponentes, denominados Provider, Telescope Repository e Monitor. O Provider é a interface de comunicação com o Observer. O Telescope Repository é responsável pelo armazenamento dos dados coletados no Telescope. A exemplo do que ocorre com o Observer Repository, ele também adota o modelo de informações apresentado na Subseção 3.1 como forma de persistir, de forma estruturada, os dados coletados. Por fim, o Monitor é o subcomponente que, de fato, contata os elementos do BitTorrent (portais de comunidades, rastreadores e pares). O Monitor é subdividido em três partes, denominadas “lentes” (*lens*), sendo cada uma responsável por monitorar um grupo diferente de elementos do universo: *Community Lens*, *Tracker Lens* e *Peer Lens*. Essa modularização permite que as lentes existentes possam ser substituídas, assim como novas possam ser facilmente incorporadas na arquitetura (sem modificação de seus componentes essenciais).

Uma grande desafio por trás do monitoramento do universo BitTorrent é criar planos de monitoramento que gerenciem o compromisso entre os objetivos de observação e recursos computacionais disponíveis, balanceando adequadamente riqueza de detalhe, cobertura e exatidão. Este desafio é tratado pelo Observer. Devido à natureza complexa do problema e restrição de espaço, este trabalho é focado na operação do Telescope e suas lentes, as quais são detalhadas na próxima seção.

4. Lentes do Telescópio

Conforme discutido na seção anterior, o Telescope é formado por três componentes. O Telescope Repository e o Provider são implementações típicas de repositórios e interfaces de serviços web, respectivamente (portanto, maiores detalhes são omitidos). Em contraste, é importante descrever o Monitor, pois ele é específico deste trabalho e implementa as estratégias discutidas na Seção 2. Nas próximas subseções são apresentadas as lentes do Monitor.

4.1. Community Lens

O Community Lens monitora a publicação de arquivos torrents em um sítio da web. Conforme comentado anteriormente, informações de interesse podem ser extraídas de páginas da Internet ou de arquivos torrent. Para monitorar páginas é necessário o desenvolvimento de *parsers* específicos para cada comunidade, como por exemplo IsoHunt. Definiu-se uma estrutura bem clara de maneira a diminuir o custo de elaboração de novos *parsers*.

Um *crawler* focado é utilizado para se obter arquivos torrent de comunidades e sítios que possuem mecanismo de busca de torrents. Normalmente, todos os torrents de um sítio da web seriam gradualmente transferidos, começando pelos publicados mais recentemente até um certo ponto nos mais antigos. Entretanto, para reduzir a quantidade total de torrents a serem carregados de um dado sítio, filtros dessas comunidades são usados para reduzir o conjunto de torrents a serem recuperados. O processo de varredura e transferência dos arquivos torrent ocorre em rodadas. A cada rodada, os dados coletados são adicionados ou atualizados no Telescope Repository.

O Community Lens é capaz de limitar a cobertura utilizando três classes de filtros: conteúdo (tipo), quantidade (número de torrents) e localidade (país da comunidade). Em termos de riqueza de detalhe, o Community Lens é o nível mais superficial em termos de riqueza de detalhes e subdivide-se em dois: no primeiro sub-nível, os sítios são apenas percorridos e listas de torrents, obtidas; no segundo sub-nível, os torrents são de fato carregados.

4.2. Tracker Lens

Os rastreadores são monitorados pelo Tracker Lens. Ele é usado para extrair a quantidade de pares (sugadores e semeadores), endereços IP e portas dos pares, além de informações sobre os próprios rastreadores.

O Tracker Lens se anuncia ao rastreador, que responde com um subconjunto aleatório de sua lista completa de pares. Devido à escalabilidade, os rastreadores impõem um limite à frequência dos pedidos feitos por um mesmo par. Por isso, se a frequência pretendida é maior que a permitida pelo rastreador, é necessário instanciar múltiplas identidades para realizar o anúncio.

O contato com o rastreador pode falhar devido à indisponibilidade do mesmo ou problemas na rede. Além disso, o enxame pode estar sem novos pares ou com zero pares. Portanto, são estabelecidas regras para determinar a frequência dos contatos em função das respostas do rastreador (verificando sua disponibilidade) e do conteúdo retornado (estimando a existência de pares desconhecidos). Dessa forma, o Tracker Lens pode direcionar seus recursos para os rastreadores que podem confiavelmente informar mais pares novos.

4.3. Peer Lens

Como o nome indica, o Peer Lens observa pares. Contatando-os, é possível extrair uma série de informações importantes, como o nível de popularidade de determinado agente de usuário, assim como inferir valores como taxas de *download*.

O Peer Lens é um agente de usuário BitTorrent modificado. Para aumentar sua escalabilidade, o Peer Lens usa um esquema *round-robin*: a cada rodada, a lente tenta estabelecer conexões com p pares; uma vez abertas, as conexões são mantidas por um tempo t , aguardando mensagens `HAVE`¹. Ao término do período, as conexões são encerradas, e o processo reinicia com os próximos p pares da fila circular. Logo que uma conexão é estabelecida, coleta-se informações sobre o par correspondente, como por exemplo versão do agente e disponibilidade e quantidade de peças concluídas.

Pares que não aceitam o estabelecimento de uma conexão remota, devido a *firewall* ou outra forma de proteção, são marcados como inalcançáveis. Para estes casos, o monitor precisa aguardar até que os inalcançáveis iniciem uma conexão com o Peer Lens. Para aumentar a probabilidade disso acontecer, utiliza-se uma estratégia inspirada em *Sybils* [Douceur 2002]. A lente cria e controla múltiplas identidades lógicas, anunciando-as ao rastreador. Dessa forma, aumenta-se a chance de que os inalcançáveis recebam pelo menos um dos IPs correspondentes ao Peer Lens, portanto aumentando a chance de conexão. Entretanto, essa técnica possui um custo associado: ao anunciar falsas identidades para os rastreadores, é possível que um ou mais pares tentem abrir múltiplas conexões com a lente. Essas conexões seriam rejeitadas, mas ainda assim passariam pelos processos de abertura e fechamento.

A cobertura é ajustada no Peer Lens filtrando-se os pares ou limitando-se a quantidade de pares (p). Além da cobertura, o Peer Lens é flexível em termos de exatidão, onde o valor de t utilizado na janela deveria ser ajustado para aumentar ou diminuir a qualidade das informações. No limite inferior, $t = 0$ significa que a conexão é imediatamente fechada depois do *handshake*. No limite superior, $t = \infty$, a conexão é mantida aberta até o par remoto desconectar (ou falhar) ou o Peer Lens ser finalizado. O estabelecimento de conexão em nível de BitTorrent (mensagem `HANDSHAKE`) permite coletar uma série de informações, como por exemplo o agente sendo utilizado e a atual disponibilidade das peças. Após esta etapa, em geral, quanto maior o tempo de conexão, maior a exatidão sobre a disponibilidade de peças nos pares e mais precisa é a estimativa de velocidade de *download* do par. Além disso, se o par remoto implementa a extensão PEX, é possível trocar listas de pares e estimar a conectividade entre eles.

5. Avaliação

A arquitetura descrita na seção anterior foi materializada na forma de um protótipo, implementado em Java 1.6. A biblioteca Java NIO foi empregada para permitir um grande número de conexões em paralelo, assim como comunicação não-bloqueante. Os dados coletados foram armazenados em um SGBD Mysql através de um conjunto de tabelas que mapeiam o modelo de gerenciamento de informações.

Esta seção oferece uma avaliação do TorrentU e está organizada como segue. A Subseção 5.1 apresenta evidências em favor do TorrentU, considerando o atendimento aos requisitos propostos. A Subseção 5.2 discute os resultados obtidos para um estudo de caso.

¹um par envia uma mensagem `HAVE` para todos os pares vizinhos conectados quando uma peça é completada.

5.1. Atendimento aos Requisitos

Foram realizados experimentos com o objetivo de avaliar a **escalabilidade**, através da demanda de recursos em função da quantidade de componentes observados. Em particular, nas Figuras 4(a) e 4(b) apresenta-se, respectivamente, o consumo médio de tráfego (*download*) e carga média de processamento na CPU em função do número médio de pares conectados.

Observa-se que as quantidades de largura de banda de *download* e de uso de CPU podem ser interpoladas por uma função linear do número de pares conectados. Embora omitidos neste trabalho, resultados similares foram obtidos para taxa de *upload* e uso de memória. A partir desses resultados, conclui-se que a quantidade de recursos necessários aumenta linearmente em função do número médio de pares conectados. Portanto, TorrentU é escalável verticalmente mesmo quando a estratégia mais custosa é empregada. Além disso, o TorrentU é escalável horizontalmente, pois é possível dividir a carga entre múltiplos Telescopes executados em paralelo, em máquinas espalhadas pela rede.

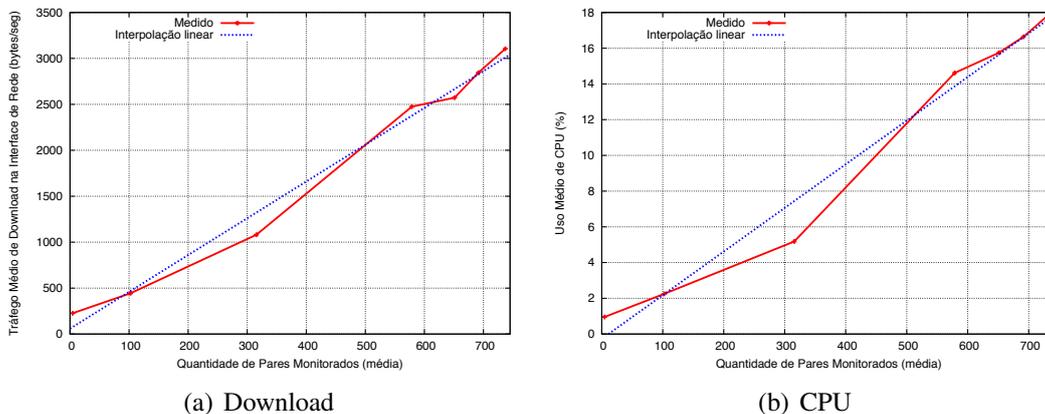


Figura 4. Ocupação de recursos

Em um cenário onde os recursos não podem ser incrementados, uma alternativa para aumentar a cobertura do monitoramento é melhorar sua eficiência diminuindo (até mesmo anulando) o coeficiente angular da função de custo, que é linear. Conforme será visto a seguir, no TorrentU esta possibilidade é explorada ao máximo através da sua flexibilidade.

O TorrentU é **flexível** porque oferece uma gama de parâmetros para riqueza de detalhes, cobertura e exatidão. Em termos de riqueza, existem 4 níveis de custo. Em termos de cobertura, foram definidos 3 filtros de comunidade, 2 filtros de rastreador, 3 filtros de pares, e há a opção de definir novos. Em termos de exatidão, existem 2 parâmetros para cada nível de riqueza: periodicidade (com que os elementos são consultados) e tamanho da amostra (percentual do conjunto observado). Portanto, são 3×2 (*comunidade*) + 3×2 (*torrent*) + 2×2 (*tracker*) + 3×2 (*peer*) = 22 parâmetros que, se multiplicados pelos valores possíveis, resultam em uma infinidade de opções.

Para um determinado objetivo de monitoramento, é possível usar estratégias que sejam mais **eficientes**. Para demonstrar essas possibilidades, foi analisada a quantidade total de *download* necessária para um cenário. Através de medições, foram obtidos os seguintes valores: arquivo torrent, 20 KB; lista com 50 pares a partir do rastreador, 467 bytes; mensagem HAVE de um par, 7 bytes; mensagem HANDSHAKE de um par, 136 bytes, e;

mensagem `BITFIELD` de um par, 180 bytes. Considere o seguinte caso de monitoramento hipotético: quantidade de pares, 100.000, quantidade de torrents, 100 e quantidade de peças por torrent, 500. Neste cenário, caso a pergunta seja “quais os conteúdos compartilhados?”, para a qual o monitoramento das comunidades é suficiente, o custo possível varia entre 2 MB (realizando *download* de torrents apenas) e 384 MB (considerando também todas as mensagens do protocolo). O TorrentU é eficiente porque é flexível no uso de diferentes estratégias, e por isso pode minimizar o custo nas mais diversas situações. No caso citado, apenas o Community Lens seria empregado para realizar o *download* dos torrents, evitando 382 MB de *download* desnecessário. Se por exemplo o monitoramento acima, com 100.000 pares, fosse realizado no intervalo de 2 dias, então as taxas de sobrecarga seriam apenas, respectivamente, 0,09 Kbps e 17,68 Kbps.

5.2. Estudo de Caso

Para realizar o estudo de caso, acompanhou-se a agenda de lançamento de seriados norte americanos, muito populares em comunidades BitTorrent. Os monitoramentos iniciaram com quatro horas de antecedência ao lançamento do conteúdo original. Os resultados apresentados a seguir são referentes ao nono episódio da oitava temporada do seriado denominado “Family Guy” nas comunidades BTJunkie, IsoHunt e TorrentDownloads. A observação foi feita em um período pouco maior que 22 horas.

A Figura 5(a) apresenta uma série temporal da evolução da quantidade de pares no enxame segundo as três lentes, apresentada em escala logarítmica no eixo y. Como esperado, observe que há divergências entre os valores observados pelas lentes, conforme explicado a seguir. Primeiro, o número de pares conectados mostra-se muito inferior, o que se deve à parametrização do Peer Lens, configurado com uma amostragem reduzida de pares. Segundo, a lista de pares conhecidos cresce de forma constante, mas consome um certo tempo até convergir para um valor mais próximo àqueles informados pelas outras lentes. Por sua vez, a soma das quantidades de pares retornadas pelos rastreadores possui a limitação de contar múltiplas vezes pares que estão conectados em mais de um rastreador. Além disso, as informações divulgadas nas comunidades podem estar desatualizadas.

A Figura 5(b) apresenta uma série temporal da quantidade de torrents de acordo com duas lentes: Community Lens e Tracker Lens. Observe a quinta hora de monitoramento, aproximadamente 30min após o término da estréia nos E.U.A.: um grande número de torrents é adicionado ao universo. Observe que mesmo 4 horas antes do lançamento do conteúdo, quando o monitoramento foi iniciado, já existiam torrents nas comunidades. Uma possibilidade é que esses torrents estejam associados a tentativas de desestimular a distribuição ilegal de conteúdo, pois verificamos nos sítios das comunidades que 78% das avaliações destes torrents denunciavam conteúdo falso, e que todas as outras, favoráveis, partiram do mesmo usuário.

De forma complementar, a Tabela 1 resume os dados que respondem uma lista de perguntas, de acordo com determinada lente, em três momentos diferentes da história do monitoramento. O MaxMind GeoIP [GeoIP 2010] foi utilizado para mapear endereços IP para países.

6. Trabalhos Relacionados

Em [Stutzbach et al. 2008] é apresentado Cruiser, um *crawler* para rede Gnutella que captura *snapshots* da rede. Cruiser é um típico *crawler* P2P para redes não-estruturadas, que usa técnicas para aumentar a velocidade de captura dos *snapshots*. TorrentU, diferentemente, tem como alvo redes BitTorrent.

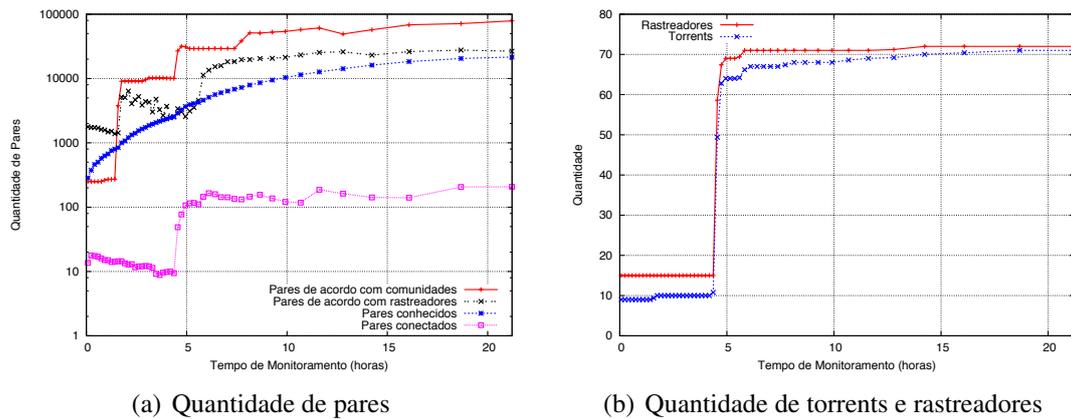


Figura 5. Exemplo de observação do Universo Torrent

Tabela 1. Amostras do monitoramento em diversos horários

Pergunta	Lente	1ª hora	10ª hora	20ª hora
Quantos torrents?	Community	9	68	71
Quantos arquivos (distintos/total)?	Community	21 / 21	249 / 769	258 / 783
Quantos rastreadores ativos?	Tracker	15	71	72
Qual a distribuição geográfica dos pares? (3 países mais populares)	Tracker	23% Espanha 22% Irlanda 14% Brasil	17% EUA 11% Espanha 9% Itália	13% EUA 9% Itália 8% Espanha
Quais agentes são utilizados? (2 agentes mais populares)	Peer	99% Azureus 0,62% UTorrent	66% UTorrent 22% Azureus	68% UTorrent 21% Azureus
Quantos pares são alcançáveis?	Peer	59 sim / 225 não	479 s / 876 n	935 s / 1827 n
Quantos usuários estão compartilhando este conteúdo?	Community Tracker Peer	271 1561 663	53029 20628 10580	78153 26629 21354

Em [Chow et al. 2007] é apresentado BTM, um sistema de monitoramento com foco em detecção de pirataria em BitTorrent. Ele possui um módulo de pesquisa de torrents e outro para análise dos mesmos. Este segundo módulo contata rastreadores e pares para obter as informações relevantes com o objetivo de detectar pares que distribuem conteúdo pirata.

BitProbes, proposto em [Isdal et al. 2007], é um sistema genérico de medição da Internet que usa pares do BitTorrent para monitorar as condições de uma rede. BitProbes também pode ser usado para monitorar enxames do BitTorrent. O sistema instanciado é composto por agentes BitTorrent modificados, que contam com um elemento denominado *shadow tracker* para compartilhamento de *peer list* e divisão da tarefa de monitoramento. Os torrents de entrada são escolhidos em um processo prévio em função do número de pares.

A principal vantagem do TorrentU sobre propostas anteriores é sua flexibilidade na escolha de um conjunto de estratégias que será empregado para extrair a informação desejada. Um conjunto de Telescopes é usado para acompanhar os elementos principais das redes BitTorrent. Cada Telescope têm três lentes, uma para cada tipo de elemento do BitTorrent. Um conjunto distribuído de Telescopes tem melhores chances de observar o universo sem sobrecarregar uma única parte da rede por causa de um monitor ou exceder

os limites de uso de recursos impostos pelas comunidades, rastreadores e pares. Um usuário pode especificar partes de interesse do universo a fim de que o TorrentU “aponte seus Telescópios” para aquela região. A arquitetura é totalmente configurável, permitindo que exatidão e tamanho da amostra sejam escolhidos para que peças do universo possam ser monitoradas de acordo com os recursos disponíveis.

7. Conclusão e Trabalhos Futuros

Este trabalho definiu o conceito de “universo BitTorrent” e seus elementos. Estratégias para extrair informações das redes BitTorrent foram identificadas (de acordo com a literatura e implementações). Essas estratégias diferem-se em custo, requisitos e peças de informação que elas são capazes de extrair; além disso, informações são obtidas em diferentes graus de exatidão.

Essas estratégias formaram a base para o projeto de uma arquitetura de monitoramento *flexível* que utiliza “Telescópios” para observar partes do universo BitTorrent. Dado o tamanho deste universo, é impraticável observá-lo inteiramente. Nem mesmo é pragmático acompanhar um grande conjunto de pares com muita riqueza de detalhe. Portanto, o TorrentU permite que um conjunto de estratégias seja escolhido e personalizado a fim de que os elementos de interesse no universo possam ser inspecionados considerando os recursos disponíveis.

Como trabalhos futuros, são consideradas três iniciativas principais. A primeira, projetar e implementar o BitTorrent Observer levando em consideração estratégias de otimização ao determinar a quantidade de Telescopes, para onde os mesmos estarão apontando, e que estratégias serão usadas, para atingir um determinado objetivo de observação. A segunda consiste em estender os experimentos, implementando múltiplas instâncias do protótipo do Telescope e efetuando experimentos de observação de longo prazo na Internet (PlanetLab). A terceira iniciativa é estender a presente arquitetura do TorrentU para que um conjunto de Observers compartilhe um conjunto maior de Telescopes e informações coletadas dos mesmos, seguindo uma filosofia P2P de monitoramento.

Agradecimentos

À Rede Nacional de Ensino e Pesquisa (RNP) pelo suporte e ao PoP-RS/Leandro Márcio Bertholdo pela disponibilização do hardware utilizado na avaliação experimental apresentada neste artigo.

Referências

- Andrade, N., Mowbray, M., Lima, A., Wagner, G., and Ripeanu, M. (2005). Influences on cooperation in bittorrent communities. In *ACM SIGCOMM Workshop on Economics of Peer-to-Peer Systems*, pages 111–115.
- Chow, K. P., Cheng, K. Y., Man, L. Y., Lai, P. K. Y., Hui, L. C. K., Chong, C. F., Pun, K. H., Tsang, W. W., Chan, H. W., and Yiu, S. M. (2007). Btm - an automated rule-based bt monitoring system for piracy detection. *Second International Conference on Internet Monitoring and Protection 2007*, page 2.
- Dale, C. and Liu, J. (2007). A measurement study of piece population in bittorrent. *IEEE Global Telecommunications Conference, 2007. GLOBECOM'07*, pages 405–410.
- Douceur, J. R. (2002). The sybil attack. In *1st International Workshop on Peer-to-Peer Systems*, volume 2429 / 2002, pages 251–260.

- Doyen, G., Festor, O., and Emmanuel Nataf (2004). A cim extension for peer-to-peer network and service management. *11th International Conference on Telecommunications*, pages 801–810.
- GeoIP (2010). Maxmind geoip, http://www.maxmind.com/app/geoip_country.
- Guo, L., Chen, S., Xiao, Z., Tan, E., Ding, X., and Zhang, X. (2007). A performance study of bittorrent-like peer-to-peer systems. *IEEE Journal on Selected Areas in Communications*, 25(1):155–169.
- Hornig, M.-F., Chen, C.-W., Chuang, C.-S., and Lin, C.-Y. (2006). Identification and analysis of p2p traffic - an example of bittorrent. In *First International Conference on Innovative Computing, Information and Control (ICICIC)*, volume 2, pages 266–269.
- Iosup, A., Garbacki, P., Pouwelse, J., and Epema, D. (2006). Correlating topology and path characteristics of overlay networks and the internet. *Cluster Computing and the Grid Workshops, 2006. Sixth IEEE International Symposium on*, 2:10–10.
- Isdal, T., Piatek, M., Krishnamurthy, A., and Anderson, T. (2007). Leveraging bittorrent for end host measurements. *Lecture Notes in Computer Science*, 4427 / 2007:32–41.
- Izal, M., Urvoy-Keller, G., Biersack, E., Felber, P. A., Al-Hamra, A., and Garces-Erice, L. (2004). Dissecting bittorrent: Five months in a torrent’s lifetime. In *5th International Workshop, PAM 2004*, volume 3015 / 2004, pages 1–11.
- Karakaya, M., Korpeoglu, I., and Ulusoy, O. (2009). Free riding in peer-to-peer networks. *Internet Computing, IEEE*, 13(2):92 – 98.
- Konrath, M. A., Barcellos, M. P., Silva, J. F., Gaspary, L. P., and Dreher, R. (2007). Atacando um enxame com um bando de mentirosos: vulnerabilidades em bittorrent. *XXV Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC 2007)*, 2:883–896.
- Legout, A., Liogkas, N., Kohler, E., and Zhang, L. (2007). Clustering and sharing incentives in bittorrent systems. *SIGMETRICS '07: Proceedings of the 2007 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, pages 301–312.
- Pouwelse, J., Garbacki, P., Epema, D., and Sips, H. (2005). The bittorrent p2p file-sharing system: Measurements and analysis. *Lecture Notes in Computer Science*, 3640 / 2005:205–216.
- Saroiu, S., Gummadi, K. P., Dunn, R. J., Gribble, S. D., and Levy, H. M. (2002). An analysis of internet content delivery systems. *Symposium on Operating Systems Design and Implementation (OSDI)*, pages 315–327.
- Stutzbach, D., Rejaie, R., and Sen, S. (2008). Characterizing unstructured overlay topologies in modern p2p file-sharing systems. *Networking, IEEE/ACM Transactions on*, 16(2):267 – 280.
- Zhang, C., Dhungel, P., Wu, D., Liu, Z., and Ross, K. (2010). Bittorrent darknets. *IEEE INFOCOM*.
- Zhang, C., Dhungel, P., Wu, D., and Ross, K. W. (2009). Unraveling the bittorrent ecosystem. pages 1–13.