

Avaliando a Eficácia e a Eficiência da Busca Par-a-Par por Conteúdo

Eduardo Colaço, Eder Martins,
Jussara Almeida, Marcos Gonçalves

¹Instituto de Ciências Exatas – Departamento de Ciência da Computação
Universidade Federal do Minas Gerais – (UFMG)
Belo Horizonte– MG – Brasil

{eduardo, ederfm, jussara, mgoncalv}@dcc.ufmg.br

Abstract. *Content-Based search on decentralized architectures, such as Peer-to-Peer (P2P) Digital Libraries (DL), differs from centralized search in many aspects, including the dynamic behavior of peers and its heterogeneity in resources capacity. Evaluation of the effectiveness and efficiency of such systems is still an unresolved problem, as most of prior work either focus only on effectiveness or considers ideal scenarios. This paper evaluates the compromise between efficiency and effectiveness on P2P content-based search considering several practical scenarios. It also presents and evaluates a replication strategy which increases effectiveness of P2P search.*

Resumo. *Busca por conteúdo em arquiteturas descentralizadas, como em bibliotecas digitais Par-a-Par (P2P), difere da busca centralizada em vários aspectos, incluindo o comportamento dinâmico dos pares e a heterogeneidade dos recursos disponíveis. A eficiência e a eficácia destes sistemas, frente a estas, é um problema ainda em aberto, pois a maioria dos trabalhos anteriores enfatiza apenas um aspecto (particularmente eficácia) ou considera cenários idealizados. Este artigo apresenta uma avaliação dos compromissos entre eficiência e eficácia da busca P2P por conteúdo, considerando vários cenários realistas. Ele ainda apresenta e avalia uma estratégia de replicação de conteúdo que contribui para melhorar a qualidade da busca P2P.*

1. Introdução

Sistemas Par-a-Par (P2P) têm se mostrado atrativos em diversos cenários dado seu potencial de escalabilidade, ausência de um ponto único de falha e baixo custo de implantação. Isso tem motivado diversos estudos sobre a extensão ou adaptação de tecnologias P2P para diversas aplicações que potencialmente podem se beneficiar dessas propriedades.

Dentre essas aplicações encontram-se as bibliotecas digitais distribuídas e as máquinas de busca P2P. Uma máquina de busca P2P é composta de computadores independentes e autônomos (i.e., pares), cada um contendo parte dos documentos da coleção. Cada par envia consultas conforme interesse dos usuários e responde a consultas (locais e remotas) com uma lista ordenada dos documentos relevantes presentes em sua coleção local. A busca P2P por conteúdo difere dos mecanismos de busca comuns em redes Par-a-Par (p. ex: KAD, Gnutella), pois a busca é realizada sobre o conteúdo do objeto (p. ex: texto completo do documento), e não em metadados como título e *tags*. Além disso ela

tem como objetivo recuperar e ordenar os documentos mais *relevantes* para uma consulta em vez de simplesmente retornar todos os objetos que casam com a consulta.

A aplicação de uma arquitetura P2P para recuperação de informação pode não apenas amenizar restrições de escalabilidade [Wu et al. 2005], mas também permitir, por exemplo, a construção de bibliotecas digitais comunitárias e colaborativas. Estas bibliotecas estariam distribuídas em um grande número de pares independentes e autônomos, cada um correspondente a um usuário (ou sub-comunidade) com interesse em compartilhar seus documentos e de se beneficiar de serviços (p.ex: busca) oferecidos pelo sistema. Nesse ambiente, os custos de gerência e manutenção do repositório são distribuídos entre os usuários, evitando a necessidade de uma autoridade mantenedora centralizada.

Contudo, os benefícios de arquiteturas P2P para máquinas de busca não vem sem consequências. Dentre estas, podemos citar o comportamento dinâmico dos pares e a limitação e heterogeneidade dos recursos disponíveis em cada par (e.g., banda e espaço em disco). Outra característica essencial é que os pares não fornecem qualquer garantia de qualidade de serviço (QoS), e podem abandonar a rede a qualquer momento. Sendo assim, a eficiência e a eficácia destes sistemas, frente a estas características, é um problema que merece investigação. Entretanto, a maioria dos trabalhos anteriores enfatiza apenas um aspecto (particularmente eficácia) ou considera cenários idealizados, deixando, portanto, várias questões em aberto. Por exemplo, uma grande parte desses trabalhos foca em obter uma eficácia (i.e., recuperação dos documentos mais relevantes) próxima da obtida em máquinas de busca centralizadas [Bender et al. 2005a, Podnar et al. 2007, Tang et al. 2003], sem considerar as limitações de recursos previamente mencionadas. Além disso, em geral, assume-se um cenário ideal em que os pares possuem comportamento pouco dinâmico.

No entanto, estudos de aplicações P2P de compartilhamento de arquivos que analisaram a dinamicidade (*churn*) dos pares concluíram que esses são muito dinâmicos, entrando e saindo várias vezes da rede [Stutzbach and Rejaie 2006]. No contexto específico de máquinas de busca que servem a bibliotecas digitais P2P, não é esperado que todos os usuários estejam presentes na rede o tempo todo. De fato, um usuário pode não obter os documentos mais relevantes para sua consulta se esses pertencerem a pares ausentes no momento em que a consulta for submetida. Logo, a dinamicidade dos pares afeta a disponibilidade de conteúdo e, portanto, a eficácia da busca P2P por conteúdo.

Neste contexto, este artigo apresenta uma avaliação dos compromissos entre eficiência e eficácia da busca P2P por conteúdo, considerando cenários mais realistas que trabalhos anteriores [Bender et al. 2005a, Podnar et al. 2007, Tang et al. 2003]. Nosso ambiente de simulação possui centenas de pares, que saem e entram na rede seguindo o comportamento dinâmico característico de redes Par-a-Par [Stutzbach and Rejaie 2006]. Em nossa avaliação utilizamos cargas reais compostas por documentos e consultas da coleção TREC-08 [Voorhees and Harman 2000]. Nós também consideramos diversas configurações de uma máquina de busca P2P que serve uma biblioteca digital descentralizada, variando o número de documentos replicados e de pares contatados por consulta, o tamanho das respostas enviadas por cada par, e a quantidade de recursos (banda e disco) disponível em cada par. O nosso modelo inclui ainda a simulação do processo de roteamento e troca de mensagens na rede sobreposta, realizada através do *framework* de simulação OverSim [Baumgart et al. 2007]. Nosso trabalho ainda avalia uma estratégia

de replicação de conteúdo que contribui para melhorar a qualidade da busca P2P.

Nossos resultados indicam que que replicação por similaridade pode incrementar a eficácia da seleção de pares em até 222% e o processamento de consultas em até 166%, por um custo em banda médio inferior à 34.2Kbps para cada par. Verificamos também que ajustes simples na máquina de busca, como aumentar o número de pares contatados e o tamanho da resposta de cada par, podem oferecer um melhor compromisso entre eficácia e eficiência que a replicação por similaridade. Por fim, verificamos que a eficácia do mecanismo de replicação não é degradada mesmo quando os pares dedicam para replicação um espaço em disco limitado ao equivalente à apenas 500 documentos.

2. Trabalhos Relacionados

Há um recente esforço no desenvolvimento de máquinas de busca P2P eficazes e eficientes. Em [Bender et al. 2005a, Podnar et al. 2007, Tang et al. 2003] são apresentadas máquinas de busca que utilizam redes Par-a-Par estruturadas (*Distributed Hash Tables* ou DHTs) para estimar e armazenar estatísticas da máquina de busca P2P. Todos estes trabalhos focam em aumentar a eficácia da busca P2P. Entretanto, em nenhum deles o mecanismo de busca é avaliado em cenários de alta dinamicidade dos pares, e em geral, não consideram a heterogeneidade e limitação dos pares quanto a recursos como banda e espaço em disco.

A alta dinamicidade é uma característica intrínseca às redes P2P [Stutzbach and Rejaie 2006] que dificulta a qualidade dos serviços que utilizam esta arquitetura. No contexto de busca P2P, em [Atalla et al. 2008] foi demonstrado que a alta dinamicidade dos pares reduz significativamente a eficácia de uma máquina de busca P2P, mesmo em cenários que os pares apresentam uma alta disponibilidade média.

Em [Li et al. 2003] foi realizado um estudo sobre a viabilidade do uso de redes P2P estruturadas no desenvolvimento de máquinas de busca par-a-par para coleções de documentos com tamanho semelhante a Web. O estudo analisa apenas a viabilidade da busca por conteúdo P2P, considerando o consumo de banda global e o espaço em disco individual de cada par. As conclusões deste estudo foram que a utilização do diretório distribuído para armazenar diretamente as listas invertidas da máquina de busca torna o processamento de consultas P2P inviável por causa do excessivo uso de banda, e que mesmo com a aplicação das várias melhorias sugeridas no artigo, os consumo de banda necessário ainda permaneciam uma ordem de magnitude acima do viável (pelos critérios do estudo).

A principal contribuição deste artigo está na **avaliação quantitativa** tanto da eficácia quanto da eficiência da máquina de busca P2P. Tomamos como caso de uso a busca por conteúdo em uma biblioteca digital P2P, em que cada par da rede possui uma parte da coleção de documentos. Diferentemente de trabalhos anteriores, consideramos cenários em que os pares possuem comportamento altamente dinâmico [Stutzbach and Rejaie 2006] e limitação de recursos como banda [Akamai 2009] e espaço disco. Por fim, adaptamos a estratégia de replicação de documentos *owner's replication* [Lv et al. 2002] ao contexto de busca por conteúdo P2P, e avaliamos o compromisso existente entre eficácia da busca (qualidade dos resultados) e eficiência (utilização de recursos como banda e espaço em disco) busca por conteúdo P2P.

3. Máquina de Busca por Conteúdo P2P

O serviço de busca é construído sobre uma rede sobreposta estruturada formada pelos pares. Cada par fornece dois serviços básicos: (1) *manutenção de estatísticas* (i.e., sumários das coleções locais) no diretório distribuído (semelhante a uma *Distributed Hash Table*). (2) *processamento local de consultas* sobre a coleção de documentos mantida pelo par. Estes serviços são utilizados pela máquina de busca P2P durante as etapas de *seleção de pares* para processar uma dada consulta e *processamento federado de consulta* pelos pares pré-selecionados, respectivamente. Por fim, após o processamento da consulta, nossa máquina de busca inicia a etapa de *replicação por similaridade*, em que os documentos mais similares à consulta são replicados em uma cache local do par que iniciou a consulta. O nosso modelo de máquina de busca, incluindo serviços e etapas de processamento, é detalhado a seguir.

3.1. Seleção de Pares

Vários trabalhos tratam da seleção de repositórios para processamento federado de consultas [Callan 2000, Bender et al. 2005b]. Em [Bender et al. 2005b] foi demonstrado que, em uma máquinas de busca P2P que não há muitos documentos repetidos (há poucos documentos replicados entre os pares), o método de seleção CORI fornece os melhores resultados. Como assumimos um cenário pessimista em que há, inicialmente, apenas uma cópia de cada documento na rede, este foi o método selecionado para nosso estudo.

O Seletor de Pares CORI calcula *scores* para a coleção de cada par p , em relação a uma consulta q , dado por $score(p, q) = \sum_{i \in q} \alpha - (1 - \alpha)T(i, p)I(i, p)$. As estatísticas $T(i, p)$ e $I(i, p)$ são calculadas por:

$$I(i, p) = \frac{\frac{\log(n+0.5)}{pf(i)}}{\log(n+1)} \quad T(i, p) = \frac{df(i, p)}{(df(i, p) + 50 + 150 \frac{V(p)}{V_{avg}}} \quad (1)$$

Onde n é o número de pares na rede, $pf(i)$ o número de pares que possuem ao menos um documento com o termo i , $df(i, p)$ o número de documentos do par p que possuem o termo i , $V(p)$ o número de termos distintos na coleção de p e V_{avg} a média desse valor para todos os pares.

Estas estatísticas são mantidas disponíveis pela própria rede P2P: cada par fica responsável por manter estatísticas de um determinado conjunto de termos, fornecendo um serviço de diretório distribuído, de estrutura semelhante a uma DHT. O par responsável pelas estatísticas do termo i é aquele cujo identificador da rede sobreposta possui menor distância ao valor do *hash* $SHA1^1$ do termo. Tal par mantém uma lista de pares (identificador e IP) que possuem o termo i e suas respectivas estatísticas ($df(i, p)$, $V(p)$).

A seleção de pares funciona como a seguir. Para cada termo i da consulta q , o par requisitante determina o identificador do par responsável pelas estatísticas do termo i e envia uma mensagem a ele, a qual será roteada pela rede sobreposta, requisitando a lista de pares com o termo i e suas respectivas estatísticas. Após receber a resposta, o par requisitante computa os *scores* de cada par, e seleciona top_p pares para processar a

¹SHA1 é uma função *hash* segura que retorna identificadores de 160 bits.

consulta. Note que o cálculo exato de N e V_{avg} exige conhecimento de estatísticas globais da rede. Por este motivo, utilizamos como estimador para N o número de pares com ao menos um termo da consulta, e como estimador de V_{avg} o valor médio do número de termos distintos da coleção de cada um destes pares.

3.2. Manutenção de Estatísticas no Diretório Distribuído

Como visto na Seção anterior, a manutenção de estatísticas sobre as coleções de cada par no diretório distribuído é determinante para a seleção de pares, afetando portanto, a eficácia da máquina de busca P2P. Por outro lado, ela também afeta a sua eficiência, uma vez que requer utilização de recursos (banda e espaço) dos pares.

Adotamos em nosso modelo um diretório distribuído semelhante ao apresentado em [Bender et al. 2005a]. Neste modelo, o diretório distribuído fornece um mapeamento entre um termo i e a lista de *posts* dos pares que possuem o termo i em sua coleção local. Um *post* contém informação de contato do par (identificador na rede sobreposta e endereço IP), sumários estatísticos da coleção ($df(i, p)$ e $V(p)$) e um prazo de validade desta informação (*time-to-live* ou *tll*).

Ao ingressar na máquina de busca P2P, o par deve enviar ao diretório distribuído *posts* para cada termo presente em mais que $DF_{threshold}$ documentos em sua coleção local. O *post* de um termo i é enviado ao par que possui identificador mais próximo ao seu *hash SHAI*. Os *posts* são reenviados periodicamente (a cada *tll* segundos). *Posts* de pares que abandonaram a rede são portanto removidos do diretório distribuído após expirado o período de reenvio.

Quando um novo par entra na rede, todos os seus vizinhos são notificados. De maneira semelhante a uma DHT, cada vizinho do novo par verifica a sua lista de *posts*, repassando ao par ingressante aqueles que ele seja o novo responsável, isto é, aqueles para os quais o identificador do par ingressante seja mais próximo ao *hash SHAI* do termo. Este evento de *post-forward* é importante, pois cria réplicas das listas de *posts* e mantém a consistência do mapeamento entre termos e *posts* no diretório distribuído.

3.3. Processamento Federado de Consultas

Após a seleção dos pares mais relevantes à consulta (Seção 3.1), o par requisitante repassa a consulta, que é processada localmente por cada par selecionado sobre sua coleção de documentos (vide Seção 3.4). A seguir, as respostas são enviadas ao par que iniciou o processamento da consulta e mescladas em uma lista unificada de documentos, antes de ser apresentada ao usuário.

O mecanismo de mesclagem de respostas é responsável por reunir as respostas de todos os pares selecionados pelo Seletor de Pares em uma única lista de documentos, ordenados pela similaridade à consulta. Em nosso modelo, utilizamos um mecanismo de mesclagem simples, em que as respostas dos pares são agregadas utilizando os valores de similaridade reportados pelo processamento de consultas local de cada par. Mecanismos de mesclagem mais sofisticados são propostos em [Larkey et al. 2000], e são recomendados para coleções dos pares com alta *topicidade*, isto é, coleções em que os documentos de cada par estão em torno de um mesmo tópico. Como este não é o caso da coleção analisada, deixamos a avaliação destes mecanismos para trabalhos futuros.

3.4. Processamento Local de Consultas

O processador de consultas local de cada par é baseado no modelo vetorial. Neste modelo, cada consulta q e documento j são representados como vetores w -dimensionais, em que w é o número de termos únicos da coleção, e cada componente dos vetores corresponde ao peso de um termo no documento (ou consulta) que o contém. Tradicionalmente, os pesos usados no modelo de espaço vetorial são baseados em estatísticas *Term-Frequency - Inverse Document Frequency* (TF-IDF) [Baeza-Yates et al. 1999]. O componente TF de um termo i em um documento j é dado pela frequência em que i ocorre em j . O IDF de um termo i é dado por $idf(i) = \lg(\frac{|D|}{d_i})$, onde D é o número de documentos da coleção e d_i é o número de documentos da coleção com o termo i . Portanto, o peso de um termo i em um documento j é dado por $W_{ij} = tf(i, j) \times idf(i)$. A similaridade entre um vetor de consulta q e o vetor de um documento j é tradicionalmente computada pelo cosseno do ângulo entre os vetores.

Entretanto, este método tradicional possui a desvantagem de exigir o recálculo das normas de todos os vetores da coleção cada vez que é adicionado ou removido um documento [Lee et al. 1997]. Um cálculo alternativo para a similaridade é proposto em [Lee et al. 1997]. A métrica de similaridade proposta troca as normas dos vetores por um fator de normalização igual ao número de termos no documento, j , isto é, $sim(q, j) = \frac{q \times j}{\sum_{i=0}^w freq(i, j)}$.

Logo, diferentemente da métrica tradicional, a métrica proposta por [Lee et al. 1997] utiliza uma norma estática e independente da coleção. Apesar disto, [Lee et al. 1997] mostrou que sua eficácia é comparável à do método tradicional para várias coleções de testes. Por esta razão, esta métrica de similaridade é mais adequada para coleções dinâmicas em que documentos são adicionados e removidos constantemente, e portanto, mais adequada ao nosso modelo de máquina de busca P2P.

3.5. Replicação de Documentos por Similaridade

Para dissociar a disponibilidade de um determinado conteúdo ao par que o hospeda, incorporamos à máquina de busca P2P uma estratégia de replicação de dados. Utilizamos uma adaptação da estratégia *owner replication* [Lv et al. 2002], proposta anteriormente para aplicações P2P de compartilhamento de arquivos. Em nosso modelo, o par, após receber as respostas à sua consulta, replica os r documentos de maior similaridade com a consulta em uma *cache* local. Esta estratégia visa aumentar a disponibilidade dos documentos mais relevantes de forma alinhada aos interesses do par. Em outras palavras, considerando que consultas realizadas por um usuário exibam alguma localidade em termos de documentos mais relevantes, a replicação local dos documentos similares a uma consulta aumenta a chance que consultas futuras encontrem documentos relevantes na rede, a despeito da possível indisponibilidade dos pares que originalmente hospedavam o conteúdo.

Em nosso modelo, cada par tem um espaço disponível (*cache*) para armazenar, localmente, um número limitado C de documentos. Quando um novo documento deve ser inserido na cache e esta se encontra cheia, a política de reposição de documentos é executada. Dado que na coleção de teste TREC-8 não existe uma distribuição de popularidade entre as consultas, nenhuma política de reposição de documentos é especialmente preferível. Portanto, optamos por uma política que remove documentos selecionados uniformemente entre os presentes.

Tabela 1. Parâmetros do Simulador da Máquina de Busca P2P

Parâmetro	Descrição
n	número de pares total que compõem a máquina de busca P2P (<i>online</i> e <i>offline</i>).
t_q	Tempo médio entre chegadas das consultas de um par.
λ_{on}, ρ_{on}	Parâmetros da distribuição Weibull dos tempos de sessão dos pares.
$\lambda_{off}, \rho_{off}$	Parâmetros da distribuição Weibull dos tempos de sessão <i>offline</i> dos pares.
C	Espaço disponível, em número de documentos, na cache local de um par para replicação por similaridade.
$ Q_{set} $	Tamanho, em número de documentos, da resposta de um par a uma consulta.
$ top_p $	Número máximo de pares selecionados pelo Seletor de Pares CORI.
r	Número de documentos replicados a cada consulta.
t_{ll}	Tempo para invalidação de <i>posts</i> dos pares no diretório distribuído.
$DF_{threshold}$	Limiar inferior para submissão <i>posts</i> no diretório distribuído.
α	Parâmetro de ajuste do Seletor de Pares CORI.

4. Metodologia de Avaliação

Este trabalho visa avaliar o compromisso entre qualidade (eficácia) e eficiência (uso de recursos dos pares) da busca por conteúdo P2P. A eficácia é medida através da degradação da qualidade dos resultados em relação aos de uma máquina de busca centralizada equivalente (linha de base). Para analisar eficiência, medimos o uso de banda e espaço em disco dos pares que compõem a máquina de busca P2P. A nossa avaliação é feita via simulação. De forma a capturar as características essenciais, nosso modelo de simulação possui três camadas: a camada de rede física, a camada da rede sobreposta P2P e a máquina de busca Par-a-Par executada sobre ela (já apresentada na Seção 3).

As próximas seções descrevem as principais considerações e componentes de nosso modelo de rede (Seção 4.1), e apresentam a coleção de testes (Seção 4.2) e métricas de avaliação de eficiência e eficácia da máquina de busca P2P (Seção 4.3). A Tabela 1 sumariza os parâmetros do nosso modelo de simulação.

4.1. Modelo de Rede Física e Sobreposta

Como o foco primordial desta avaliação é o compromisso entre eficácia e eficiência da máquina de busca P2P em condições mais realistas, faz-se necessário a utilização de um modelo de rede física e sobreposta detalhado, capaz de simular roteamento, troca de mensagens, atrasos de transmissões, falhas de comunicação, limitação de banda, entre outras características encontradas em sistemas P2P reais.

Para tanto, foi utilizado o *framework* de simulação OverSim [Baumgart et al. 2007]. O *framework* fornece dois modelos de rede física (rede IP e rede IP simplificada) e permite a simulação de diversas redes sobrepostas. As aplicações interagem com a rede sobreposta através da *common API* [Dabek et al. 2003], que define um conjunto de serviços comuns a várias redes sobrepostas estruturadas, tais como Chord, Kademlia e Pastry.

Outra característica importante do modelo de rede é o comportamento dinâmico dos pares, ou *churn*. A dinamicidade dos pares em máquinas de busca P2P ainda não foi caracterizada na literatura, embora vários trabalhos tenham investigado este aspecto em aplicações P2P de compartilhamento de arquivos. Em particular, uma caracterização [Stutzbach and Rejaie 2006] de três sistemas populares mostra que a duração da sessão dos pares (*tempo online*) é bem modelada por distribuições Weibull² com parâmetros de forma $\rho \approx 0.44$ e de escala $\lambda \approx 35.20$ unidades de tempo. Embora relevante, esse

²Função de Densidade de Probabilidade dada por $f(x) = \frac{\rho x^{\rho-1}}{\lambda \rho}$

estudo não apresenta dados similares para os tempos de indisponibilidade (*tempo offline*), outro aspecto de suma importância para a eficácia da busca em P2P. Devido à falta de caracterização da disponibilidade em aplicações de busca P2P assim como sua caracterização completa em outros tipos de aplicação P2P, optamos por modelar os tempos *online* e *offline* dos pares por distribuições Weibull com parâmetros de forma ρ_{on} e ρ_{off} , e parâmetros de escala λ_{on} e λ_{off} , respectivamente.

4.2. Coleção de Teste

Utilizamos a coleção de teste TREC-8 ad-hoc (ou TREC, no decorrer do texto). A TREC [Voorhees and Harman 2000] é constituída por um total de 528.125 documentos, extraídos de jornais e conferências do governo dos EUA, e é comumente utilizada no desenvolvimento e avaliação de sistemas de recuperação de informação. A coleção também fornece um conjunto de 50 consultas manualmente elaboradas por especialistas. Assumimos, em nosso modelo, que os pares submetem consultas de forma independente, com tempo entre consultas exponencialmente distribuídos com média t_q .

4.3. Métricas de Avaliação

Utilizamos revocação relativa para quantificar a qualidade (eficácia) da busca P2P. A revocação é definida como a fração de documentos relevantes à consulta que é retornado pela máquina de busca, isto é, dada uma consulta q , um conjunto de documentos relevantes à consulta $relevantes(q)$ e um conjunto de documentos retornados pela máquina de busca $retornados(q)$, definimos Revocação de uma máquina de busca com relação a q como $r(q) = \frac{|relevantes(q) \cap retornados(q)|}{|retornados(q)|}$. A Revocação Relativa é utilizada para medir o quanto a máquina de busca P2P se aproxima da máquina de busca centralizada, e possui a mesma definição, mas considerando o conjunto de documentos relevantes como o conjunto de documentos retornados pela máquina de busca centralizada.

Note que a revocação, como definida, acima, aplica-se diretamente aos resultados do processamento federado de consultas, mas não captura a eficácia da seleção de pares. Para tanto, definimos a revocação da seleção de pares, $r_{selecao}(q)$, como a fração de documentos relevantes à consulta que se encontram presentes nos pares selecionados. Em outras palavras, $r_{selecao}(q)$ é definida como acima, trocando $retornados(q)$ por $selecionados(q)$, que representa o conjunto união dos documentos dos pares selecionados para a consulta q : $r_{selecao}(q) = \frac{|relevantes(q) \cap selecionados(q)|}{|selecionados(q)|}$. Assim como no caso da Revocação Relativa para uma consulta, definimos a Revocação Relativa da Seleção de Pares para avaliar a eficácia do Seletor de Pares considerando o conjunto de documentos retornados pela máquina de busca centralizada como o conjunto de documentos relevantes.

Para avaliar a eficiência de uma máquina de busca, medimos a quantidade de recursos média por par participante da máquina de busca necessários para atingir um determinado nível de eficácia. Em nosso modelo, os pares possuem banda e espaço em disco limitados, e portanto estes são os recursos medidos.

5. Resultados

5.1. Cenários Avaliados

Consideramos diversos cenários, variando os parâmetros do simulador, para avaliar o compromisso entre eficácia e eficiência da busca P2P por conteúdo. Em todos os cenários,

cada par é responsável por um subconjunto de documentos da coleção TREC. Os documentos são distribuídos de maneira uniforme entre os pares, uma vez que não há caracterização da distribuição de documentos para nosso cenário de uma biblioteca digital P2P. Todavia, é importante notar que essa distribuição uniforme é pessimista tanto ao Seletor de Pares, já que é mais difícil distinguir os pares mais relevantes para uma determinada consulta, quanto à disponibilidade dos documentos na rede, uma vez que, inicialmente, há apenas uma única cópia de cada documento no sistema.

Para avaliar o impacto de diferentes níveis de dinamicidade dos pares, fixamos a distribuição dos tempos *online* dos pares com uma média de $\mu_{on}=91.84$ minutos, dada pelos parâmetros Weibull ρ_{on} e λ_{on} iguais a 0.44 e 35.20 respectivamente (ver Seção 4.1). Além disso, fixamos $\rho_{off}=\rho_{on}$ enquanto variamos λ_{off} , produzindo diferentes tempos médios *offline* μ_{off} . Finalmente, de forma a capturar a estabilidade do sistema, define-se disponibilidade do par, A , como sendo o percentual do tempo (simulado) durante o qual o par está disponível na rede, em média, ou seja $A=\frac{\mu_{on}}{\mu_{on}+\mu_{off}}$. Em nossas simulações, utilizamos valores de A iguais a 25% e 75%.

A distribuição de banda dos pares foi feita com base na distribuição observada em *hosts* dos EUA, reportada em [Akamai 2009]. Consideramos uma banda mínima de 256kbps, já que *hosts* com menor capacidade, em especial aqueles com conexões *dial-up*, muitas vezes não possuem a banda necessária para contribuir na manutenção do diretório distribuído. De qualquer forma, estes *hosts* compõem uma parcela pequena dos *hosts* observados (5.8%). Além disso, devido à sua baixa capacidade, estes *hosts* tendem a se comportar como clientes da máquina de busca e não como provedores de serviço.

Com o intuito de avaliar o compromisso entre eficiência e eficácia da busca P2P, variamos o tamanho da resposta de cada par ($|Q_{set}|=100, 1000$ documentos), o número de documentos replicados por consulta ($r=0, 250, 500$), e o espaço disponível replicação na cache local ($C=500, 1000$ e ilimitado ou ∞). Outros parâmetros do nosso modelo foram fixados. Utilizamos uma rede com total de $n = 1000$ pares, e um tempo médio entre consultas de um mesmo par $t_q=1000s$. Este valor de t_q é uma escolha pessimista, representando usuários que utilizam a máquina de busca com uma alta frequência, uma vez que analisando os *logs* da máquina de busca AOL³ [AOL 2006] encontramos que o tempo médio entre consultas dos usuários em um mesmo dia de 3190.1s. Logo, estes valores representam a máquina de busca sob uma carga proporcionalmente alta para seu número de usuários. Os valores de limiar para publicação no diretório distribuído ($DF_{threshold}=10$) e o parâmetro de ajuste do seletor de pares ($\alpha=0.4$) foram fixados tomando como base trabalhos anteriores [Bender et al. 2005b]. Fixamos ainda o tempo de vida dos *posts* no diretório distribuído ($t_{ll} = 400s$) e focamos em Chord como infraestrutura de rede P2P.

Os resultados reportados nas próximas seções são médias de no mínimo 4 experimentos, com desvio padrão inferior a 9% da média.

5.2. Eficácia e Eficiência da Seleção de Pares

Nesta seção, avaliamos a eficácia e a eficiência do Seletor de Pares CORI em nosso modelo de máquina de busca, variando o número de pares selecionados a cada consulta (top_p), o número de documentos replicados após o processamento de cada consulta (r) e disponibilidade média dos pares (A). Inicialmente, consideramos um cenário otimista em que

³Nossa utilização dos *logs* de consulta está dentro dos termos de uso exigidos pela AOL.

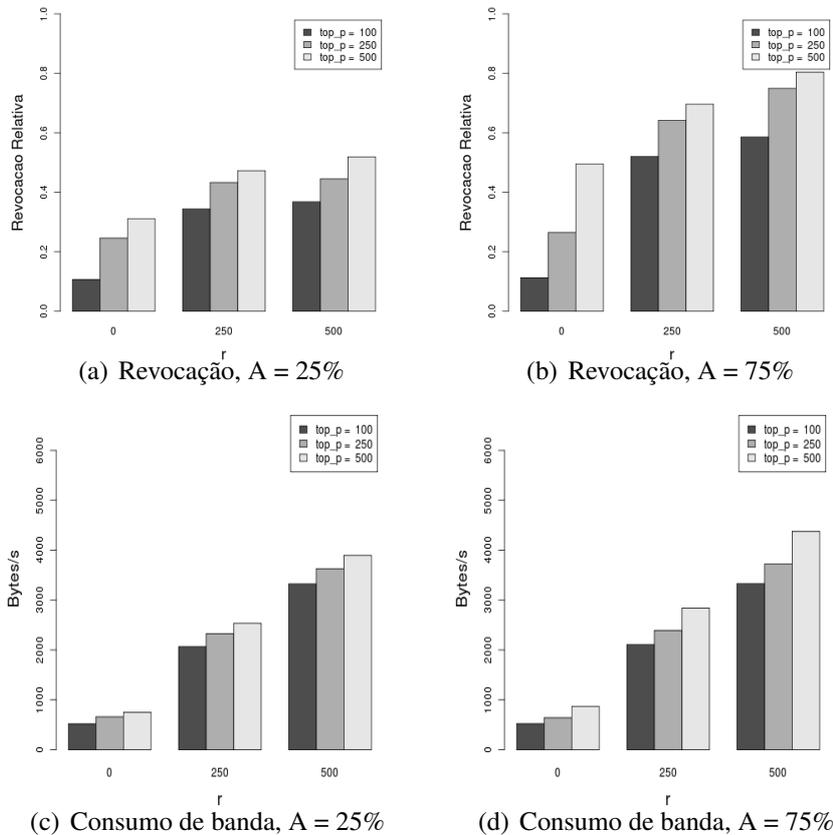


Figura 1. Revocação relativa e consumo de banda da seleção de pares.

não há limitação de espaço em disco de cada par (tamanho da cache $C=\infty$).

As Figuras 1(a-b) mostram os valores da revocação relativa média da seleção de pares, ao final da simulação, em cenários que os pares possuem disponibilidade média (A) igual a 25% e 75% e para diferentes valores de $r = [0, 250, 500]$ e $top_p = [100, 250, 500]$. Estes resultados mostram que, como esperado, a eficácia da seleção de pares é reduzida para disponibilidade média ($A = 25\%$), e que maiores valores para top_p e r levam a uma maior Revocação na seleção de pares. Entretanto, eles demonstram também que, considerando o cenário $top_p=100, r=0$ e $A=75\%$ como ponto de partida, a replicação por similaridade possui um impacto maior na qualidade da seleção de pares que o número de pares selecionados, i.e., o principal fator de degradação da eficácia da seleção de pares é a indisponibilidade dos documentos na rede. Incrementar o número de pares selecionados top_p de 100 para 250 melhora a eficácia da seleção de pares de 130% (para $r=0$), enquanto variar o número de documentos replicados r de 0 para 250 causa um incremento de 222% (para $top_p=100$). Fazendo uma análise de variância [Jain 1991] com esses níveis para top_p e r , e fixado a disponibilidade média $A = 0.25$, observamos que o fator r é responsável por 77% da variação na eficácia, enquanto top_p é responsável apenas por 21% desta variação. Note entretanto que o ganho em replicar mais de 250 documentos é menos significativo. Tomando como base $r=250$ e $top_p=100$, o ganho em aumentar top_p para 250 (26%) ou 500(37%) é mais significativo que aumentar r para 500 (6%).

Já as Figuras 1(c-d) apresentam a banda média (em bytes/s) utilizada por cada par nos mesmos cenários acima citados. Note que, assim como no caso da eficácia, valores

maiores de top_p e r levam a um maior consumo de banda. É importante notar também que há consumo de banda médio ligeiramente maior em cenários que os pares possuem maior disponibilidade ($A = 0.75$), já que, como cada par envia consultas de maneira independente, um maior número de pares na rede implica uma maior número de consultas enviadas e portanto, mais banda consumida. Nossos resultados indicam que a estratégia de replicação possui impacto maior na utilização de recursos que o número de pares selecionados. O incremento no número de pares selecionados top_p de 100 para 250 gera um aumento no consumo de banda, em média, de 9% a 27% (considerando os diferentes valores de r), enquanto aumentar o número de documentos replicados a cada consulta r de 0 para 250 gera um uso de banda 234% a 297% maior (para diferentes valores de top_p). A análise de variância com níveis $top_p=[100, 250]$, $r=[0, 100]$ e $A=0.25$ aponta que o fator r é responsável por 98% variação do consumo de banda.

Para analisar o compromisso entre eficácia e eficiência, observamos a razão entre a revocação relativa e o consumo de banda para cada cenário. Os resultados mostrados na Tabela 2 indicam que $r = 0$ e $top_p = 500$ gera a melhor razão entre eficácia e eficiência em banda, o que indica que é mais eficiente em banda selecionar mais pares que replicar os documentos. Entretanto, é importante notar que mesmo oferecendo um pior compromisso entre eficácia e eficiência, a replicação por similaridade oferece uma eficácia de 62% à 415% maior que os cenários sem replicação, por um consumo de banda médio inferior à 34.2Kbps (para $A=0.25$). Portanto, a utilização de replicação por similaridade é adequada apenas em cenários em que a eficácia da busca seja mais importante que a eficiência. Em outros cenários, principalmente quando os pares possuam pouca banda, utilizar valores de top_p maiores pode levar a um melhor compromisso.

5.3. Eficácia e Eficiência do Processamento Federado de Consultas

Avaliamos agora a eficiência e eficácia do processamento federado de consultas. Para tanto, fixamos o número de pares selecionados ($top_p = 250$), e variamos o tamanho da resposta de cada par (Q_{set}), o número de documentos replicados (r) e a disponibilidade média dos pares (A). Medimos a Revocação Relativa e uso de banda médio de cada par considerando que o espaço em disco em cada par é ilimitado ($C=\infty$).

As Figuras 2(a-b) mostram a revocação relativa média para valores de $r = [0, 250, 500]$, $Q_{set}=[100, 1000]$ e $A=[0.25, 0.75]$. Como esperado, maiores valores de Q_{set} e r levam a uma maior eficácia, enquanto uma menor disponibilidade média dos pares reduz a qualidade da busca. Assim como nos resultados da eficácia da Seleção de Pares, encontramos que o número de documentos replicados a cada consulta tem impacto maior na eficácia que o tamanho da resposta de cada par. Fixado o valor de $A=0.25$, realizamos uma análise de variância com níveis $Q_{set}=[100, 1000]$ e $r=[0, 250]$ e obtivemos que o valor de r , por si só, explica 83% da variação da eficácia, enquanto o tamanho da resposta apenas 8%. Um resultado interessante é que em alguns cenários, em especial no cenário com $r=0$, incrementar o tamanho das respostas dos pares não aumenta, ou aumenta de-

Tabela 2. Razão entre revocação relativa e consumo de banda para seleção de pares.

		A=0.25			A=0.75		
		0	250	500	0	250	500
top_p	r						
	100	0.0002054548	0.0001671035	0.0001112500	0.0002158816	0.0002471428	0.0001743880
	250	0.0003725259	0.0001849922	0.0001234786	0.0004123207	0.0002684268	0.0001938928
500	0.0004178119	0.0001865705	0.0001337773	0.0005731019	0.0002469856	0.0001840872	

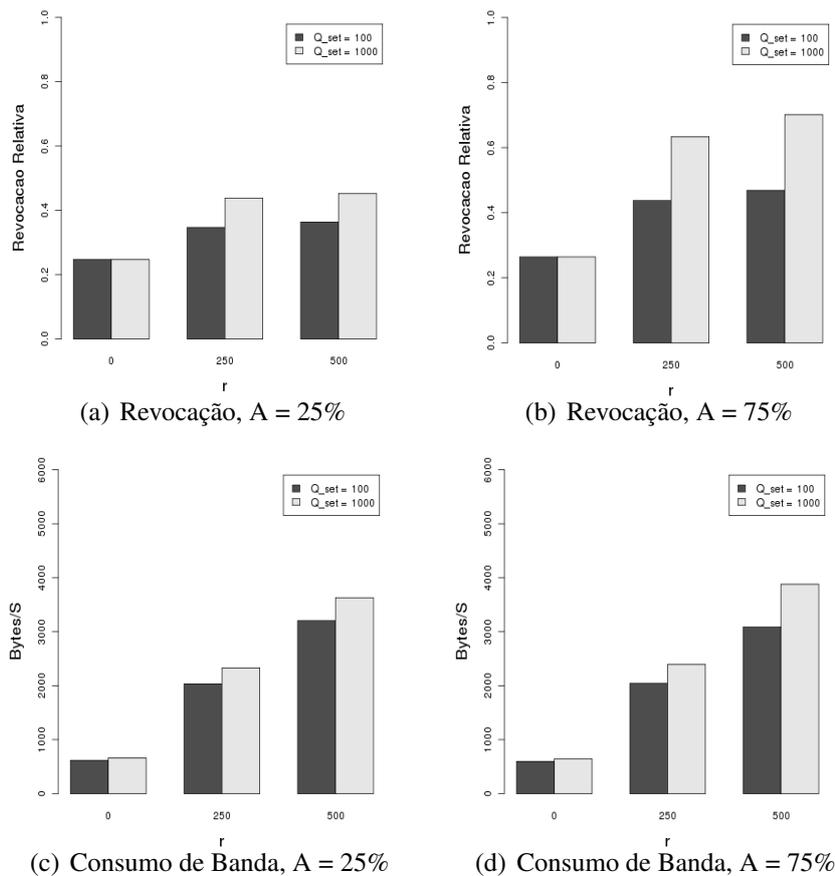


Figura 2. Revocação relativa e consumo de banda do processamento de consultas.

sprezivelmente, a qualidade dos resultados da busca. Acreditamos que isto se deve ao fato dos documentos estarem, inicialmente, distribuídos uniformemente entre os pares, o que implica que os documentos relevantes à consulta também possuem uma distribuição uniforme. Em cenários que a replicação não ocorre, esta uniformidade permanece até o fim da simulação, enquanto que em cenários que $r > 0$, os documentos mais relevantes são replicados nos pares que enviam mais consultas (mais tempo *online* na rede), que passam então a concentrar estes documentos relevantes. Nestes cenários, o tamanho da resposta passa a ter um efeito maior na qualidade da busca.

As Figuras 2(c-d) apresentam a banda média utilizada (em bytes/s) por cada par nos mesmos cenários. Novamente, a replicação de documentos possui um impacto muito maior no consumo de banda que o tamanho da resposta de cada par. A variação do tamanho da resposta de cada par $|Q_{set}|$ de 100 para 1000 gera um aumento no consumo de banda de apenas 16%, enquanto aumentar o valor de r causa um aumento de até 230% no consumo de banda. Portanto, a estratégia de replicação possui um maior impacto tanto na eficácia quanto na eficiência da busca P2P.

A Tabela 3 apresenta a razão entre a revocação relativa e o consumo de banda médio dos pares, para diferentes valores de $|Q_{set}|$, r e fixado ($A = 0.25$). Podemos observar que o melhor compromisso entre eficácia e eficiência é obtido para $|Q_{set}| = 100$ e $r = 0$, e que, com exceção para o cenário em que $r = 0$, incrementar o tamanho das respostas dos pares oferece um melhor compromisso que incrementar o valor de r . Apesar do pior razão entre eficácia e eficiência, a replicação fornece uma eficácia de 40%

a 166% maior que os cenários sem replicação por um consumo de banda relativamente baixo (menor que 28.4Kbps em média), e portanto, é adequada a cenários que a eficácia da máquina de busca P2P é mais importante que sua eficiência.

5.4. Eficiência em Consumo de Espaço em Disco

Por fim, nós avaliamos a eficiência em consumo de espaço em disco da busca P2P medindo a degradação da eficácia devido à limitação de espaço disponível para *cache* local de cada par, C . Para tanto, fixamos valores de $top_p=100$, $|Q_{set}|=1000$ e $r=250$, e variamos a quantidade de espaço em disco ($C=[500, 1000, \infty]$) disponível para replicação, e a disponibilidade média dos pares ($A=[0.25, 0.75]$).

A Tabela 4 apresenta o valor da Revocação Relativa média da busca P2P ao final da simulação. Podemos notar que, em todos os cenários, não é necessário uma *cache* local muito grande para garantir a eficácia da máquina de busca. Valores de $C=500$, isto é, cada par dedicar a cache local o espaço em disco equivalente a apenas 500 documentos, já é o suficiente para atingir eficácia equivalente a $C=\infty$.

6. Conclusão e Trabalhos Futuros

Neste artigo, avaliamos a eficácia (qualidade) e eficiência (utilização de recursos) de uma máquina de busca P2P, em cenários que os pares possuem comportamento altamente dinâmico e recursos limitados, como espaço em disco e banda, limitados. Utilizamos uma coleção de testes real (TREC-8) e consideramos um modelo de rede sobreposta e física detalhado. Apresentamos também uma estratégia de replicação para o contexto de busca P2P, e avaliamos o compromisso entre qualidade e utilização de recursos na escolha de parâmetros da máquina de busca P2P.

Nossos resultados apontam que a replicação por similaridade pode incrementar a qualidade da busca P2P em até 166% quando comparada a cenários em que não há replicação, a um custo em banda médio inferior à 34.2Kbps por par. Apesar da grande melhoria na eficácia da busca P2P, a replicação por similaridade oferece um menor compromisso entre eficácia e eficiência quando comparada a ajustes mais simples, como aumentar o número de pares contatados a cada consulta ou o tamanho da resposta de cada par. Entretanto, em todos os cenários a replicação forneceu uma eficácia superior de 40% a 166%, o que a indica como uma alternativa para cenários em que a eficácia da busca for mais importante que eficiência. Por fim, verificamos que nosso modelo de replicação mantém sua eficácia mesmo em cenários que os pares possuem relativamente pouco espaço em disco dedicado para replicação (o equivalente a 500 documentos).

Tabela 3. Razão entre revocação relativa e consumo de banda do processamento de consulta.

$r \backslash Q_{set} $	A=0.25			A=0.75		
	0	250	500	0	250	500
100	0.0004007127	0.0001715502	0.0001135608	0.0004420546	0.0002145239	0.0001516828
1000	0.0003736625	0.0001866039	0.0001253251	0.0004117104	0.0002649097	0.0001809713

Tabela 4. Eficácia da máquina de busca para diferentes tamanhos de cache.

$C \backslash A$	0.25	0.75
500	0.4302667	0.5938450
1000	0.4361200	0.6291967
∞	0.4382025	0.6333112

Trabalhos futuros incluem avaliações de diferentes redes sobrepostas e números de pares, além da utilização de coleções de testes com distribuições de documentos entre pares e popularidade de consultas não uniformes.

7. Agradecimentos

Este trabalho foi parcialmente desenvolvido dentro dos projetos INCT-Web (MCT-CNPq 57.3871/2008-6) e InfoWeb (MCT-CNPq 55.0874/2007-0) contando com apoios financeiros do CNPq e da FAPEMIG.

Referências

- Akamai (2009). The state of the internet. Technical report, Akamai.
- AOL (2006). 500k user session collection. Technical report, America Online.
- Atalla, F., Miranda, D., Almeida, J., Gonçalves, M., and Almeida, V. (2008). Analyzing the impact of churn and malicious behavior on the quality of peer-to-peer web search. In *SAC '08*.
- Baeza-Yates, R., Ribeiro-Neto, B., et al. (1999). *Modern information retrieval*. Addison-Wesley Harlow, England.
- Baumgart, I., Heep, B., and Krause, S. (2007). OverSim: A Flexible Overlay Network Simulation Framework. In *IEEE GIS '07*, pages 79–84.
- Bender, M., Michel, S., Triantafillou, P., Weikum, G., and Zimmer, C. (2005a). Minerva: collaborative p2p search. In *VLDB '05*, pages 1263–1266.
- Bender, M., Michel, S., Weikum, G., and Zimmer, C. (2005b). The MINERVA project: Database selection in the context of P2P search. *Datenbanksysteme in Business, Technologie und Web*.
- Callan, J. (2000). Distributed information retrieval. *Advances in information retrieval*.
- Dabek, F., Zhao, B., Druschel, P., Kubiatowicz, J., and Stoica, I. (2003). Towards a common API for structured peer-to-peer overlays. *Lecture Notes in Computer Science*, pages 33–44.
- Jain, R. (1991). *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. Wiley- Interscience, New York.
- Larkey, L., Connell, M., and Callan, J. (2000). Collection selection and results merging with topically organized US patents and TREC data. In *CIKM '00*, pages 282–289.
- Lee, D., Chuang, H., and Seamons, K. (1997). Document ranking and the vector-space model. *Software, IEEE*, 14(2):67–75.
- Li, J., Loo, B., Hellerstein, J., Kaashoek, M., Karger, D., and Morris, R. (2003). On the Feasibility of Peer-to-Peer Web Indexing and Search. *LECTURE NOTES IN COMPUTER SCIENCE*.
- Lv, Q., Cao, P., Cohen, E., Li, K., and Shenker, S. (2002). Search and replication in unstructured peer-to-peer networks. In *ICS '02*, pages 84–95. ACM New York, NY, USA.
- Podnar, I., Rajman, M., Luu, T., Klemm, F., and Aberer, K. (2007). Scalable Peer-to-Peer Web Retrieval with Highly Discriminative Keys. In *ICDE '07*.
- Stutzbach, D. and Rejaie, R. (2006). Understanding churn in peer-to-peer networks. In *ACM SIGCOMM '06*, pages 189–202. ACM New York, NY, USA.
- Tang, C., Xu, Z., and Dwarkadas, S. (2003). Peer-to-peer information retrieval using self-organizing semantic overlay networks. In *SIGCOMM '03*, pages 175–186.
- Voorhees, E. and Harman, D. (2000). Overview of the eighth text retrieval conference (TREC-8). *NIST SPECIAL PUBLICATION SP*, pages 1–24.
- Wu, L., Akavipat, R., and Menczer, F. (2005). 6S: Distributing crawling and searching across Web peers. In *IASTED '05*.