AVALIANDO A EFICÁCIA E A EFICIÊNCIA DA BUSCA PAR-A-PAR POR CONTEÚDO

EDUARDO JOSÉ MOREIRA COLAÇO

AVALIANDO A EFICÁCIA E A EFICIÊNCIA DA BUSCA PAR-A-PAR POR CONTEÚDO

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

Orientador: Jussara Marques de Almeida
Co-orientador: Marcos André Gonçalves

Belo Horizonte
31 de novembro de 2008

Moreira Colaço, Eduardo José
D1234p Avaliando a Eficácia e a Eficiência da Busca Par-a-Par por Conteúdo / Eduardo José Moreira Colaço. — Belo Horizonte, 2008
xiv, 72 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de Minas Gerais
Orientador: Jussara Marques de Almeida
Co-orientador: Marcos André Gonçalves

1. k1 k2 k3. I. Título.

CDU 519.6*82.10

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

FOLHA DE APROVAÇÃO

Avaliando a eficácia e a eficiência da busca Par-a-Par por conteúdo

EDUARDO JOSÉ MOREIRA COLAÇO

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

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

PROF. MARCOS ANDRÉ GONÇALVES - Co-orientador
Departamento de Ciência da Computação - UFMG

PROF. DORGIVAL OLAVO GUEDES NETO
Departamento de Ciência da Computação - UFMG

PROF. EDLENO SILVA DE MOURA
Departamento de Ciência da Computação - UFAM

Belo Horizonte, 14 de junho de 2010.

Agradecimentos

Agradeço primeiramente a minha família, que sempre foi meu porto seguro e fonte de apoio incessante. Agradeço especialmente a minha mãe, cujo cuidado e carinho sempre estiveram presentes ao longo destes 25 anos. Agradeço também ao meu pai, irmãos e sobrinha, que completam esse time valoroso e precioso que chamo de família.

Agradeço aos meus amigos, a minha segunda família. Amigos que tenho de infância e amigos recentes, todos que me apoiaram e com quem compartilhei felicidades e dificuldades. Tenho muito a agradecer à família que ganhei na "Casa do Seu João", aos grandes parceiros que conheci no laboratório VoD e todos os outros que surgiram na minha vida durante o mestrado. Nós construímos juntos esta etapa da minha vida.

Agradeço aos meus orientadores, Jussara e Marcos, que me guiaram e ensinaram muito durante este trabalho. Agradeço por toda paciência e dedicação, e por estarem sempre dispostos a ensinar e ajudar. Por estes motivos eu me orgulho de chama-los de professores.

Agradeço a Daiana, que me dedicou um amor imensurável ao longo de 6 anos. Não há agradecimento suficientemente grande para tanto carinho.

Agradeço a A UFMG e ao CNPQ, que me forneceram apoio logístico durante o mestrado. Agradeço aos professores e funcionários do DCC, que formam uma grande equipe da qual tenho orgulho de ter recebido o apoio.

Por fim, mas não menos importante, agradeço todos que tornaram este momento possível, direta ou indiretamente.

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 essas características, é um problema ainda em aberto, pois a maioria dos trabalhos anteriores enfatiza apenas um aspecto (particularmente eficácia) ou considera cenários idealizados (p. ex: em que os pares nunca abandonam a rede, ou não possuem limitação de banda). Nesta dissertação, nós apresentamos uma avaliação da eficácia, eficiência, bem como dos compromissos entre estes dois aspectos na busca P2P por conteúdo, considerando características práticas de bibliotecas digitais P2P, tais como a topicidade das coleções de documentos, alta dinamicidade, heterogeneidade e limitação de recursos (particularmente armazenamento e largura de banda) dos pares. Também avaliamos uma estratégia de replicação de conteúdo e uma estratégia de mesclagem de respostas, que contribuem para melhorar a qualidade da busca P2P.

Abstract

Content-Based search on decentralized architectures, like 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 unsolved problem, as most of prior work either focuses only on effectiveness or considers ideal scenarios (e.g., pairs never leave the network or have unlimited bandwidth). In this dissertation, we present an evaluation of efficiency, effectiveness and the tradeoffs between these two aspects, on P2P content-based search considering several practical P2P DL aspects, such as document collection topicality, the dynamics of peer participation (churn) and peer heterogeneity (with respect to bandwidth and storage capacity). We also evaluate a document replication strategy as well as a query response merging strategy which increase the effectiveness of P2P search.

Lista de Figuras

Lista de Tabelas

Sumário

Capítulo 1

Introdução

1.1 Motivação

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 pela possibilidade de distribuição dos custos de manutenção do sistema entre seus usuários sem a necessidade de uma organização centralizada. 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 e as máquinas de busca por conteúdo. Bibliotecas digitais (BD) são sistemas que preservam, gerenciam e fornecem acesso a uma coleção de objetos digitais. Há hoje um grande número de bibliotecas digitais [2, 45] que possuem repositórios gerenciados e mantidos por autoridades centralizadas. Em geral, esta autoridade é responsável por todos os custos de preservação dos objetos e manutenção dos serviços providos pela biblioteca digital. Tradicionalmente, as BDs fornecem serviços de busca por conteúdo em que o usuário expressa sua necessidade de informação através de uma consulta e recebe, como resposta, uma lista de objetos ordenados por sua similaridade à consulta . Esta busca por conteúdo em uma BD 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, nome do arquivo e *tags*.

Há, no entanto, cenários em que este modelo centralizado de BD e máquina de busca não é adequado. A necessidade de uma entidade que arque com todos os custos e gerência do repositório pode dificultar a formação e preservação de BDs para coleções de objetos pouco populares. Em alguns casos, mesmo coleções de documentos populares podem ter sua preservação interrompida, devido a falta de rentabilidade ou

de interesse por parte da entidade mantenedora [46]. Um exemplo deste último caso é a interrupção do serviço *GeoCities*[1], apesar deste ter atraído mais de 177 milhões de visitantes no ano de 2008 [17]. Uma possível solução para estes problemas é a utilização de uma arquitetura par-a-par (P2P) na construção de bibliotecas digitais (e de suas respectivas máquinas de busca). Uma biblioteca digital P2P é composta de computadores independentes e autônomos (isto é, pares), cada um contendo parte dos documentos da coleção. Cada par da BD faz parte também da máquina de busca P2P, e envia e responde a consultas (locais e remotas) com uma lista de documentos ordenados por sua relevância à consulta. Através da arquitetura P2P, os custos de gerência e manutenção do repositório da BD são distribuídos entre seus usuários, evitando-se assim a necessidade de uma autoridade mantenedora centralizada.

Busca par-a-par por conteúdo é uma área de pesquisa e desenvolvimento ainda muito recente. Grande parte dos esforços foca em obter uma eficácia (isto é, recuperação dos documentos mais relevantes) próxima da obtida em máquinas de busca centralizadas [10, 35, 43], sem considerar a heterogeneidade dos pares quanto a recursos como banda e espaço em disco bem como a restrição dos mesmos. Além disso, em geral assume-se um cenário idealizado em que os pares nunca saem da rede ou possuem comportamento pouco dinâmico. No entanto, vários estudos sobre a dinamicidade dos pares (ou *churn*) em aplicações P2P de compartilhamento de arquivos concluíram que pares tendem a ser muito dinâmicos, entrando e saindo várias vezes da rede [41]. Por exemplo, no contexto de bibliotecas digitais P2P, não é esperado que todos os usuários da biblioteca estejam presentes na rede o tempo todo. Nesse cenário instável, 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 ao sistema. Logo, a dinamicidade dos pares afeta diretamente a disponibilidade de conteúdo e, consequentemente, a eficácia da busca P2P por conteúdo.

Em um trabalho publicado recentemente[6], foi demonstrado, através de simulações, que a alta dinamicidade dos pares por si só já reduz significativamente a eficácia de uma máquina de busca P2P. Naquele trabalho foi demonstrado que a alta dinamicidade dos pares degrada a precisão da máquina de busca mesmo em cenários que esses apresentam uma alta disponibilidade. Dois cenários foram analisados, um consistindo de uma rede P2P altamente colaborativa e outro com uma rede com baixa colaboração entre os pares. No primeiro, os pares possuem conhecimento de estatísticas globais relativas ao número total de documentos na rede e a frequência de um determinado termo

[1] O GeoCities era um serviço de hospedagem de sítios Web com conteúdo produzido por seus usuários, que se organizavam em cidades e vizinhanças. É importante notar que o GeoCities não é tecnicamente uma BD, mas apenas um repositório de documentos gerados por usuários.

em todos os documentos. Estas estatísticas são utilizadas por máquinas de busca para determinar o grau de relevância de um objeto para uma consulta. No segundo cenário os pares possuem apenas conhecimento de estatísticas da coleção local de documentos. Os resultados desse trabalho apontam que, mesmo em cenários em que os pares possuem uma disponibilidade média de 75% (isto é, um par permanece no sistema em média 75% do tempo), considerada alta para redes par-a-par, a precisão média da máquina de busca P2P é degradada em até 22%, se comparada à de uma máquina busca centralizada, quando os pares possuem conhecimento de estatísticas globais dos documentos (redes altamente colaborativas), e é degradada em 54% em redes em que os nós possuem apenas conhecimento de estatísticas locais (redes com baixa colaboração entre os pares).

Ainda em [6], é proposto um mecanismo de replicação por similaridade para reduzir os efeitos da alta dinamicidade dos pares. Neste mecanismo, um par replica localmente os documentos retornados em resposta à sua consulta que possuem maior similaridade. Esta estratégia apresentou um incremento significativo na eficácia da máquina de busca P2P em cenários que os pares possuem conhecimento das estatísticas globais da coleção documentos. Entretanto, em cenários que os pares possuem apenas conhecimento de estatísticas de suas coleções locais, o mesmo não ocorre. É sugerido que, como a replicação por similaridade aumenta a *topicidade*[2], os pares passam a possuir documentos com muitos termos em comum. Como a máquina de busca proposta utiliza o modelo vetorial *Term Frequency - Inverse Document Frequency* (TF-IDF) [8] (que utiliza as estatísticas mencionadas acima), o par que possui uma coleção com alta topicidade, ao utilizar apenas estatísticas locais, retorna documentos com uma similaridade muito menor. Isto ocorre porque o componente IDF é amplamente reduzido para termos muito frequentes em uma coleção [6].

Os resultados apresentados em [6] demonstram que o mecanismo de replicação por si só não é suficiente para tornar uma máquina de busca P2P eficaz. Faz-se necessário também um mecanismo que supere o problema da *topicidade*. É importante notar que a topicidade pode surgir mesmo em situações em que não há o mecanismo de replicação por similaridade. Os usuários podem ser detentores de coleções extremamente especializadas, isto é, que possuam *naturalmente* uma alta topicidade.

Como já dito anteriormente, a topicidade afeta diretamente o cálculo do IDF, que depende de estatísticas da coleção global de documentos. O erro no cálculo da similaridade surge quando as estatísticas utilizadas pelo par diferem muito das estatísticas globais. A manutenção destas estatísticas em um ambiente P2P não é uma tarefa

[2]Topicidade é definido em [6] como a cobertura, por parte de um par, de um conjunto de documentos sobre um determinado tópico de interesse.

simples, dado que o número de termos e documentos de uma coleção pode se tornar muito grande, tornando algumas soluções inviáveis [26] .

Vários trabalhos apresentados propõem a utilização de uma rede *overlay* estruturada para a manutenção (no todo ou parcialmente) destas estatísticas globais [32, 38, 43, 42, 18, 16]. Estes trabalhos, no entanto, não consideram várias características de redes P2P reais, como a alta dinamicidade dos pares, heterogeneidade de capacidade (banda e espaço) e saídas abruptas, e portanto, apresentam resultados incompletos quanto à eficácia e à eficiência da busca P2P por conteúdo.

Portanto, não há na literatura uma avaliação quantitativa do impacto de características de sistemas P2P reais sobre a eficácia e a eficiência da busca P2P por conteúdo. A *topicidade*, identificada em [6], também continua um problema em aberto. A ausência de estudos que consideram estas características põe em dúvida a viabilidade de máquinas de buscas e bibliotecas digitais par-a-par. Por este motivo, a avaliação da eficácia e da eficiência da busca P2P por conteúdo frente à *topicidade* e a características práticas de redes par-a-par, tais como a alta dinamicidade, heterogeneidade e limitação de capacidade dos pares (tanto de armazenamento quando em largura de banda) é de grande importância para a viabilidade e desenvolvimento de bibliotecas digitais e máquinas de busca P2P.

1.2 Objetivos

O principal objetivo deste trabalho é realizar uma avaliação quantitativa da eficácia (qualidade dos resultados), eficiência (utilização de recursos como banda e espaço em disco) e dos compromissos existentes entre estes dois aspectos para a busca P2P por conteúdo, considerando o problema da *topicidade* e características de redes P2P realistas, tais como a alta dinamicidade, heterogeneidade e limitação de capacidade dos pares (tanto de armazenamento quanto de largura de banda). Avaliamos a *eficácia* de uma máquina de busca par-a-par comparando a qualidade seus resultados aos de uma máquina de busca centralizada. Avaliamos a *eficiência* medindo quantidade média de recursos utilizados por cada par da máquina de busca para atingir um determinado nível de eficácia. Também é objetivo deste trabalho avaliar duas técnicas previamente propostas, a *mesclagem Kirsch* (proposta em [22] para o contexto de busca por conteúdo com repositórios distribuídos e aqui aplicada, pela primeira vez, no contexto de busca P2P) e *replicação por similaridade* [6], que podem atenuar os efeitos da *topicidade* e dinamicidade dos pares sobre a eficácia da busca P2P. Portanto, podemos elencar os seguintes objetivos para este trabalho:

1. Quantificar o potencial de eficácia da máquina de busca P2P, frente à *topicidade* e à *dinamicidade* dos pares.

2. Avaliar o impacto da *topicidade*, da *dinamicidade* e da *limitação de capacidade* dos pares sobre a eficácia e a eficiência da busca P2P por conteúdo.

3. Avaliar o impacto da *replicação por similaridade* e *mesclagem Kirsch* sobre a eficácia e a eficiência da busca P2P por conteúdo.

4. Avaliar os compromissos existentes entre eficácia e eficiência na busca P2P por conteúdo.

1.3 Contribuições

A principal contribuição deste trabalho está na avaliação quantitativa, via simulação, tanto da eficácia quanto da eficiência da máquina de busca par-a-par. 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 [41], bem como a limitação e heterogeneidade quanto a recursos como banda [3] e espaço disco. Além disso, avaliamos uma estratégia de *replicação por similaridade* de documentos e *meclagem de respostas Kirsch* [22] no contexto de busca P2P por conteúdo, e quantificamos o compromisso existente entre eficácia da busca (qualidade dos resultados) e eficiência (utilização de recursos como banda e espaço em disco) da busca P2P por conteúdo. De maneira objetiva, podemos elencar as principais contribuições deste trabalho a seguir:

- Desenvolvimento de dois simuladores de eventos discretos seguindo dois modelos distintos:

  - O *modelo simplificado* de simulação de máquina de busca P2P, que toma como base o simulador desenvolvido [7, 6] e é utilizado neste trabalho com o intuito de quantificar o potencial de eficácia da busca P2P por conteúdo. Este modelo ignora limitações de banda dos pares, falhas de comunicação e localização de conteúdo na rede, mas em contrapartida, permite a avaliação do potencial de eficácia da máquina de busca P2P em cenários de maior escala, com um grande número de pares (dezenas de milhares) e documentos (milhões).

  - O *modelo detalhado* de simulação de máquina de busca P2P, utilizado para avaliar a eficácia, eficiência e o compromisso entre estes aspectos na busca P2P por conteúdo. Este simulador possui um modelo de rede física e sobreposta detalhado, capaz de simular roteamento de mensagens na rede sobreposta, limitação de banda e outras características encontradas em sistemas P2P reais. Devido ao maior nível de detalhamento, as avaliações realizadas com este simulador são limitadas a cenários com um menor número de pares (centenas).

- Confirmação de resultados anteriores [7, 6], os quais apontam que a *dinamicidade dos pares* e a *topicidade* podem degradar em muito a eficácia da busca P2P por conteúdo (utilizando o *modelo simplificado* de simulação).

- Avaliação do *potencial* de eficácia da máquina de busca P2P e dos mecanismos de *replicação por similaridade* e *mesclagem Kirsch* (utilizando o *modelo simplificado* de simulação).

  - Em cenários que não há topicidade, verificamos que a *replicação por similaridade* pode reduzir drasticamente a degradação da máquina de busca P2P causada pela dinamicidade dos pares, incrementando a eficácia da busca P2P a até 99% da eficácia de uma máquina de busca centralizada equivalente.
  - Em cenários com alta topicidade, a *replicação por similaridade* por si só não é suficiente para manter a eficácia da máquina de busca, permanecendo uma degradação na eficácia da busca P2P de pelo menos 72,5% mesmo quando se utiliza um alto nível de replicação. A *mesclagem Kirsch* (em conjunto com a *replicação por similaridade*) apresentou-se como mecanismo potencialmente eficaz nestes cenários, podendo incrementar a eficácia da busca P2P a 95% da eficácia de uma máquina de busca centralizada equivalente.

- Avaliação quantitativa da eficácia, eficiência e do compromisso entre estes dois aspectos na busca P2P em condições mais realistas que trabalhos anteriores [32, 38, 43, 42, 18, 16, 34, 6]. Nesta avaliação consideramos características práticas redes P2P, tais como a heterogeneidade e limitação dos pares quanto a banda e espaço em disco, bem como seu comportamento dinâmico e topicidade das coleções de documentos. Nesse contexto mais realista, nossos resultados (obtidos utilizando o *modelo detalhado* de simulação) indicam que:

  - Em cenários de baixa topicidade, a *replicação por similaridade*, por si só, pode incrementar a qualidade da busca P2P em até 166% quando comparada

a cenários em que não há replicação. No entanto, os resultados também apontam que a replicação por similaridade oferece um pior compromisso entre eficácia e eficiência quando comparada com ajustes mais simples da busca P2P, como aumentar o número de pares contatados a cada consulta ou o tamanho da resposta de cada par.

- Em cenários de alta topicidade, a *mesclagem Kirsch*, por si só, forneceu um modesto ganho de 5% a 10% na qualidade da busca, mas demonstrou ser um mecanismo eficiente, provendo uma melhoria na eficácia da busca P2P proporcionalmente maior que seu impacto no consumo de banda (que foi de apenas 3% no pior caso).
- A replicação por similaridade é eficiente quanto à utilização de espaço em disco dos pares. Verificamos que não é necessário uma *cache* local muito grande para garantir a eficácia da máquina de busca, bastando cada par dedicar o espaço em disco equivalente a 500 documentos.
- Após a melhoria da eficácia da máquina de busca P2P proporcionada pela *replicação por similaridade* e pela *mesclagem Kirsch*, a maior parte da degradação da eficácia remanescente ocorre já durante a etapa em que máquina de busca seleciona os pares que irão processar a consulta, o que aponta um novo desafio na busca P2P por conteúdo.

Em suma, a contribuição deste trabalho está na apresentação de novos desafios e avaliação quantitativa de soluções para o desenvolvimento da busca par-a-par por conteúdo. A metodologia aqui adotada e os resultados obtidos fornecem um arcabouço mais sólido que trabalhos anteriores [32, 38, 43, 42, 18, 16, 34, 6] por considerar características práticas de máquinas de busca par-a-par previamente ignoradas. Essa avaliação mais detalhada pode favorecer o desenvolvimento e implantação de serviços de busca por conteúdo em aplicações de compartilhamento de documentos e bibliotecas digitais P2P.

1.4 Organização do Texto

O texto está organizado da seguinte forma: O Capítulo 2 apresenta os trabalhos relacionados à bibliotecas digitais e busca por conteúdo em redes par-a-par. O Capítulo 3 apresenta nosso modelo de máquina de busca P2P. O Capítulo 4 discute nossa metodologia de avaliação de eficácia e eficiência de uma máquina de busca P2P por conteúdo. No Capítulo 5 apresentamos uma avaliação do potencial de eficácia da máquina de

busca P2P por conteúdo frente à dinamicidade e topicidade. No Capítulo 6 realizamos uma avaliação detalhada da eficácia e da eficiência da busca P2P por conteúdo, bem como do compromisso existente entre estas duas métricas. Por fim, elencamos as conclusões, considerações finais e trabalhos futuros no Capítulo 7.

Capítulo 2

Trabalhos Relacionados

2.1 Sistemas Par-a-Par

A arquitetura par-a-par tem se tornado cada vez mais popular na engenharia de aplicações em redes. Uma aplicação é considerada par-a-par quando seus participantes (pares) são tanto provedores quanto consumidores do serviço. Por esse motivo as aplicações possuem um potencial quanto à escalabilidade, já que a demanda cresce junto à oferta do serviço.

Os sistemas par-a-par podem ser classificados quanto à organização de seus nós em redes sobrepostas não estruturadas, redes semi-estruturadas e redes estruturadas. Os sistemas que não possuem distinção alguma entre os pares e que não forçam alguma organização na rede sobreposta são considerados não estruturados. A rede Gnutella [1] de compartilhamento é não estruturada, e a recuperação de informação na rede é realizada através de *alagamento* ou *random walks* [1].

Arquiteturas par-a-par semi-estruturadas foram propostas como melhoria às redes não estruturadas. Sistemas como KaZaA [27] e propostas de melhoria para a Gnutella [15] propõem uma distinção entre pares comuns e superpares (*nodes* e *supernodes*). Esta distinção leva em consideração a heterogeneidade dos pares, alocando carga maior aos pares de melhor capacidade e disponibilidade, diminuindo os efeitos da alta dinamicidade dos nós e aumentando a escalabilidade do sistema.

Por fim, os sistemas em que o protocolo obriga uma organização rígida na rede são ditos de arquitetura par-a-par estruturada. Os maiores expoentes destes sistemas são os diretórios distribuídos ou *Distributed Hash Tables* (DHT) como Chord, CAN, Kademlia e Pastry [40, 36, 30, 37]. Estes sistemas possuem protocolos próprios para controle de entrada, saída e falha dos nós, de forma a manter a estrutura lógica da rede sobreposta. Além disso, estes sistemas fornecem uma interface de diretório

distribuído, em que as chaves são mapeadas de maneira uniforme entre os pares. Em geral, localizar o par responsável por uma chave tem custo $O(log(n))$, em que n é o número de pares na rede.

2.2 Recuperação de Informação Textual Par-a-Par

Há hoje vários trabalhos desenvolvidos sobre recuperação de informação textual par-a-par, em especial sobre máquinas de busca *Web* par-a-par, tanto em arquiteturas estruturadas quanto em arquiteturas não estruturadas. Em [26] foi realizado um estudo sobre a viabilidade do uso ingênuo (*sic*) destas arquiteturas no desenvolvimento de máquinas de busca par-a-par. Por uso ingênuo da de um arquitetura P2P estruturada, entenda-se a utilização de uma DHT para armazenamento na íntegra de todas as estatísticas de uma coleção de documentos (e.g. listas invertidas para o processamento de consultas utilizando modelo TF-IDF [8]). Já o uso ingênuo de uma arquitetura P2P não estruturada consiste na utilização de alagamento para execução de consultas ou construção de índices da máquina de busca. Esse trabalho avaliou a viabilidade de tais máquinas de busca considerando apenas o consumo de banda global dos pares[1] e o espaço em disco individual dos pares. As conclusões deste estudo foram que o uso ingênuo da arquitetura P2P, tanto estruturada quanto não estruturada, tornava a máquina de busca inviável, e que, mesmo com a aplicação de várias melhorias sugeridas, os custos ainda permaneciam pelo menos uma ordem de magnitude acima do viável.

Em [18] é sugerido um mecanismo de busca em que os pares possuem o conhecimento global dos documentos da rede. Os pares se organizam em uma rede não estruturada, e as informações sobre os documentos são compartilhadas entre todos os nós através de um mecanismo de fofoca (*gossiping*). O estudo propõe que o mecanismo se adequa apenas a pequenas comunidades de pares, já que algoritmo apresenta problemas de escalabilidade para um número suficientemente grande de pares.

Em [32, 10, 39, 38, 35, 4, 43, 42] são utilizadas utilizadas redes estruturadas (DHTs) para armazenar estatísticas globais com o objetivo de dar suporte à máquina de busca global. O ODISSEA [42] utiliza uma DHT para manter um índice invertido global. A função *hash* da DHT divide os espaço de termos entre os pares[2], portanto o par associado a chave k é responsável pela lista invertida do termo k. É sugerido a

[1] Somatório do consumo de banda de cada par individual.

[2] A partição se dá através do mapeamento de um termo a um par da DHT com uma função hash seguro, e.g. SHA-1.

utilização de *bloom filters* [BLOOM] para reduzir os requisitos de banda necessários para a manutenção e utilização desse índice global. De maneira semelhante funciona o Minerva [10], diferenciando-se por utilizar uma lista invertida com granularidade no nível de par, e não de documento. É sugerido a utilização de *hash sketches*[20] para estimação do freqüência global dos termos e redução dos custos de comunicação. Em ambos os trabalhos não é levado em conta a dinamicidade, heterogeneidade e limitação de capacidade dos pares, além de que não é feito um estudo de como a heterogeneidade da popularidade dos termos e consultas podem afetar o sistema.

Com o intuito de diminuir a carga no diretório global e observando a heterogeneidade da popularidade dos termos nos documentos e consultas, o ALVIS [28, 35] utiliza uma técnica de chaves altamente discriminantes (*highly discriminative keys*). Esta técnica gera índices para combinação de termos que possuem uma freqüência maior que que um certo limiar (parâmetro do modelo proposto). Os resultados mostraram um aumento linear no número de chaves e um incremento na qualidade da busca, porém muitas das chaves geradas nunca eram utilizadas. Em [4] é proposto um mecanismo de indexação dirigido a consultas. Neste modelo são indexadas apenas as chaves que possuem maior frequência, atendendo qualquer outra consulta por alagamento. Esta técnica evita o armazenamento de consultas pouco utilizadas na DHT, em contrapartida é necessário a utilização de um mecanismo de alagamento caso a consulta não esteja entre as mais frequentemente utilizadas. Ambos trabalhos não levam em conta a dinamicidade, heterogeneidade e limitação de capacidade dos pares.

Em [38] é realizada uma combinação da utilização de chaves altamente discriminantes [28, 35] e indexação dirigida a consultas [4], criando um sistema sensível a co-ocorrências de termos em consultas. Neste trabalho é demonstrado que o tráfego gerado pelo sistema na rede é linear ao número de pares na rede. Em um outro trabalho[32], é apresentada uma outra técnica que também leva em consideração a co-ocorrência de termos em consultas. Neste último são utilizados *hash sketches* para determinar frequência global dos termos (da mesma maneira que [10]). Através de operações de união de *hash sketches* o sistema é capaz de calcular a probabilidades condicionais de utilização de dois termos em uma mesma consulta, e a partir daí, inferir a correlação do uso dos dois termos. São propostos neste trabalho dois mecanismo, que baseados na correlação entre dois termos são capazes de retornar uma busca com melhor qualidade que trabalhos anteriores. Apesar dos bons resultados, tanto em [38] quanto em [32] não é considerada a distribuição da carga entre os pares. Esta distribuição será claramente não uniforme, dado que a popularidade dos termos nas consultas e documentos não o é[8], e portanto, não é suficiente para afirmar que o sistema é *viável* ou *eficiente*. E novamente, ambos os trabalhos desconsideram a heterogeneidade,

limitação de capacidade e dinamicidade dos pares.

O PSearch [43] propôs uma rede que se organiza semanticamente, utilizando *Latent Semantic Indexing*. Esta técnica reduz em muito a quantidade de dados armazenados no diretório global e a comunicação entre os pares, além de aumentar a qualidade dos resultados das buscas. O trabalho ainda propõe uma técnica para melhorar a distribuição dos pares no diretório global, de maneira que a vizinhança das chaves mais populares permaneçam mais povoadas. A principal limitação do estudo é exigência de uma amostra significativa da coleção de documentos para computação anterior à entrada de par no sistema. Além disso, não há garantias que o sistema apresente os mesmos resultados com a evolução da coleção de documentos indexados. O estudo desconsidera a dinamicidade e heterogeneidade dos pares, pois considera que a máquina de busca deve permanecer em um subconjunto de pares estáveis e com boa conectividade.

Em [33] é proposto um algoritmo de *Pagerank* distribuído. A convergência do algoritmo para o valor global é comprovado matematicamente e experimentalmente. Os experimentos realizados possuem uma quantidade muito pequena de pares (no máximo 10) e não consideram heterogeneidade e dinamicidade. Além disso, o algoritmo apresentado é restrito a coleções de documentos hiper-ligados (como a *Web*), o que limita a sua utilização no desenvolvimento de bibliotecas digitais P2P.

Em [24, 14, 22] são apresentados mecanismos de mesclagem de respostas para máquinas de busca distribuídas. Em [22], em particular, é apresentado um mecanismo de mesclagem de respostas (referenciado neste trabalho como *mesclagem kirsch*) em que cada nó que processa a consulta envia estatísticas de suas coleções locais junto a suas respostas, fornecendo informação suficiente para que a mesclagem das das respostas de um processador de consultas distribuído se aproxime do equivalente centralizado. Em [14] é sugerido o uso de um mecanismo alternativo de normalização durante a mesclagem das respostas, porém, resultados apresentados em [24] indicam que esta técnica pode não ser eficaz se os documentos forem distribuídos entre os nós da máquina de busca por tópicos, isto é, se houver topicidade entre os nós. Nenhum destes trabalhos analisa os mecanismos de mesclagem de respostas em um ambiente com pares dinâmicos e com limitação de banda e espaço em disco.

Por fim, no trabalho apresentado em [6] é realizado um estudo de como as entradas e saídas frequentes dos pares afetam a qualidade da recuperação de informação textual par-a-par. Neste estudo foi demonstrado que a alta dinamicidade dos pares degrada significativamente a eficácia da máquina de busca P2P, mesmo para redes altamente colaborativas, em que há conhecimento por todos os pares de estatísticas globais. Neste trabalho é proposto um mecanismo de replicação por similaridade, que

reduz significativamente a degradação da eficácia da máquina de busca P2P, porém possui o efeito colateral de aumentar a *topicidade* dos pares, reduzindo a qualidade das respostas dos pares que se tornam super-especializados. O estudo é realizado através de simulação e leva em consideração a alta dinamicidade dos pares, mas desconsidera a heterogeneidade e limitação dos pares quanto a capacidade (banda e espaço) e a heterogeneidade da frequência dos termos em consultas e documentos. Detalhes da infra-estrutura da rede P2P, tais como a organização dos pares na rede e roteamento de mensagens entre os pares, também foram desconsiderados neste trabalho.

Capítulo 3

Busca por Conteúdo em Bibliotecas Digitais P2P

3.1 Descrição

A máquina de busca por conteúdo em uma biblioteca digita P2P é composta de computadores independentes e autônomos denominados pares, cada um contendo partição dos documentos da coleção global. Nosso modelo do serviço de busca P2P por conteúdo considera uma arquitetura de três camadas: a camada de rede física, a camada da rede sobreposta par-a-par e a camada de aplicação, na qual é executada a máquina de busca máquina de busca par-a-par propriamente dita.

3.2 Camada de Rede Física

A camada de rede física define a interconexão e comunicação entre os computadores de uma rede. Em nosso modelo, nos referimos a todos os serviços providos pelas camadas de rede e transporte do modelo OSI como serviços da camada de rede física (em contraponto aos serviços providos pela camada de rede sobreposta).

Nosso modelo de máquina de busca P2P utiliza o serviço de transporte de mensagens indiretamente (através da rede sobreposta) ou diretamente, enviando mensagens em datagramas UDP. Em ambos os casos, as mensagens são roteadas sobre um rede IP até o *host* destino. É importante notar que, tanto a camada de rede (IP) quanto a camada de transporte (UDP) não fornecem garantias da ordem, tempo ou mesmo da chegada de mensagens [Kurose], fato que pode impactar tanto na eficácia quanto eficiência da máquina de busca P2P.

3.3 Camada de Rede Sobreposta Par-a-Par

Dá-se o nome de rede sobreposta a uma rede computadores construída sobre outra rede. Em nosso caso, consideramos como camada de rede sobreposta a rede de enlaces lógicos, construídos sobre a camada de rede física em uma arquitetura par-a-par. Como discutido nos Seção 2.1, as redes sobrepostas podem ser classificadas como redes sobrepostas não estruturadas (não forçam organização alguma na formação dos enlaces lógicos), redes sobrepostas semi-estruturadas (distinção simples dos pares em duas classes: nós e super-nós) e redes sobrepostas estruturadas (organização rígida com critérios rígidos para formação e manutenção dos enlaces lógicos).

Nosso modelo de máquina de busca utiliza uma rede sobreposta estruturada com suporte a operações de *roteamento de mensagens* e *notificação de entrada e saída de pares vizinhos*. Muito embora o conceito de vizinho varie para cada rede sobreposta, em geral, os vizinhos de um par são definidos como os pares da rede que possuem identificadores da rede sobreposta mais próximos ao seu. Esta característica é aproveitada na replicação de conteúdo que serão localizados na rede estruturada através de algum identificador, em nosso caso, as estatísticas necessárias na etapa de seleção de pares para o processamento de uma consulta são localizadas utilizando *hash* dos termos da consulta como identificadores da rede sobreposta (vide Seções 3.4.1 e 3.4.2). Tanto a operação de *roteamento de mensagens* quanto a *notificação de entrada e saída de pares vizinhos* estão definidas na *common API*[1] [19] e podem ser mapeadas ou implementadas facilmente em redes sobrepostas tais como Chord, Kademlia, CAN e Pastry[40, 30, 36, 37].

3.4 Camada de Aplicação

O máquina de busca P2P por conteúdo é construído na camada de aplicação sobre a rede sobreposta estruturada formada por pares (computadores independentes e autônomos que agem como cliente e provedores da máquina de busca P2P). Cada par fornece dois serviços básicos: (1) *manutenção de estatísticas* (isto é, 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* e *processamento federado de consulta*, respectivamente. Por fim, após o proces-

[1] A *common API* [19] define um conjunto de serviços comuns a várias redes redes sobrepostas estruturadas. O objetivo da *common API* é fornecer uma abstração da rede sobreposta, desacoplando aplicação da implementação da rede sobreposta estruturada.

samento federado da consulta, nossa máquina de busca inicia as etapas de *mesclagem de respostas*, na qual as respostas de diversos pares são unificadas em uma lista ordenda de documentos, e *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.

3.4.1 Seleção de Pares

Vários trabalhos tratam da seleção de repositórios para processamento federado de consultas [13, 11]. Em [11] foi demonstrado que em uma máquina de busca P2P onde 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, dados por:

$$score(p,q) = \sum_{i \in q} \alpha - (1-\alpha) T(i,p) I(i,p) \tag{3.1}$$

$$I(i,p) = \frac{\frac{log(n+0{,}5)}{pf(t)}}{log(n+1)} \qquad T(i,p) = \frac{df(i,p)}{(df(i,p) + 50 + 150\frac{V(p)}{Vavg}} \tag{3.2}$$

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.

Em nosso modelo, estas estatísticas são mantidas disponíveis pela própria rede P2P - cada par fica responsável por manter estatísticas de um determinado subconjunto de termos da coleção, 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*[2] do termo. Tal par mantém uma lista de pares (identificador e IP) que possuem o termo i e suas respectivas estatísticas ($df(i,p)$ e $V(p)$).

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

[2]SHA1 é uma função *hash* segura que distribui os identificadores de maneira uniforme em um espaço de 160 bits.

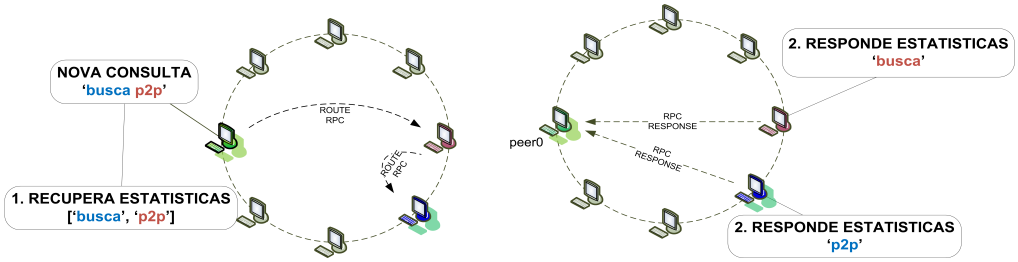


Figura 3.1. Seletor de Pares CORI, obtendo estatísticas para os termos *'busca'* e *'p2p'* no diretório distribuído.

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 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.

3.4.2 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 [10]. 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 TTL).

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 *hash SHA1*. Os *posts* são reenviados periodicamente (a cada *ttl* segundos, em nossos experimentos). *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*,

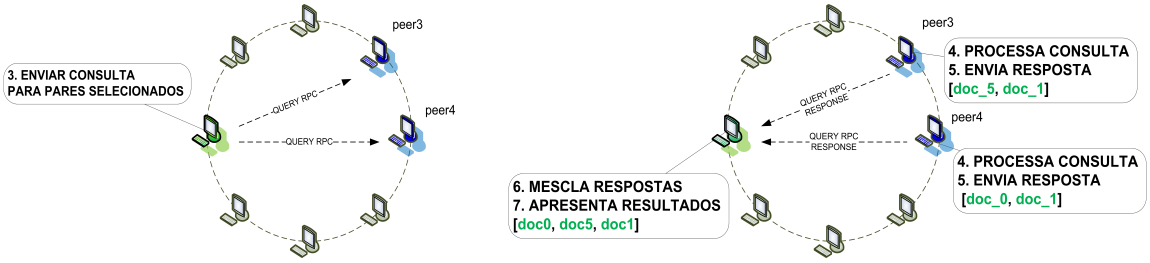


Figura 3.2. Processamento federado de consultas na máquina de busca P2P.

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 SHA1* 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.

3.4.3 Processamento Federado de Consultas

Após a seleção dos pares mais relevantes à consulta (Seção 3.4.1), o par requisitante envia a consulta aos pares selecionados, que então é processada por cada par sobre sua coleção local de documentos de maneira independente (vide Seção 3.4.4). Em seguida, as respostas são enviadas ao par que iniciou o processamento da consulta e mescladas em uma lista unificada de documentos, para finalmente 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. Utilizamos dois mecanismos de mesclagem de respostas em nosso modelo, o mecanismo de *mesclagem simples* e o mecanismo de *mesclagem Kirsch*[22], os quais são discutidos em detalhes na Seção 3.4.5.

3.4.4 Processamento Local de Consultas

Cada par da máquina de busca possui um processador de consultas local e sua respectiva coleção de documentos. Utilizamos um modelo de máquina de busca vetorial, no qual cada consulta q e documento j são representados como vetores T-dimensionais,

$$\vec{q} = \{W_{0q}, W_{1q}, W_{2q}, W_{wq}\} \tag{3.3}$$

$$\vec{j} = \{W_{0j}, W_{1j}, W_{2q}, W_{wj}\} \tag{3.4}$$

em que T é o número de termos únicos da coleção, e cada componente dos vetores W_{ij} (W_{iq}) corresponde ao peso do termo i no documento j (consulta q) que o contém. O peso de um termo i em uma consulta q é dado por:

$$W_{iq} = \begin{cases} 1, & \text{se o termo } i \text{ está presente na consulta } q, \\ 0, & \text{caso contrário.} \end{cases} \tag{3.5}$$

Tradicionalmente, o peso de um termo i em um documento j no modelo de espaço vetorial calculado a partir do produto das estatísticas *Term-Frequency e Inverse Document Frequency* (TF e IDF) [8]. 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

$$idf(i) = \lg(\frac{|D|}{d_i}) \tag{3.6}$$

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 W_{ij} de um termo i em um documento j é dado por

$$W_{ij} = tf(i, \vec{j}) \times idf(i) \tag{3.7}$$

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

$$sim(\vec{q}, \vec{j}) = \cos\theta = \frac{\vec{q} \cdot \vec{j}}{||\vec{q}|| \times ||\vec{j}||}, \tag{3.8}$$

entretanto, este método possui a desvantagem de exigir o recálculo das normas de todos os vetores da coleção cada vez que é adicionado ou removido um documento [25]. Um cálculo alternativo para a similaridade é proposto em [25]. 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:

$$sim'(\vec{q}, \vec{j}) = \frac{\vec{q} \cdot \vec{j}}{\sum_{i=0}^{w} tf(i, \vec{j})} \tag{3.9}$$

Logo, diferentemente da métrica tradicional, a métrica proposta em [25] utiliza uma uma norma estática e independente da coleção. Apesar disto, [25] 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 dinâmica em que documentos são adicionados e removidos constantemente, e portanto, mais adequada ao nosso modelo de máquina de busca P2P.

3.4.5 Mesclagem de Respostas

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 do usuário. Definimos em nosso modelo dois mecanismo de mesclagem de respostas: o 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, e o mecanismo de *mesclagem kirsch*, em que os valores de similaridade reportados são recomputados pelo par que enviou a consulta, utilizando-se de um agregado de estatísticas de par que processou a consulta.

O mecanismo de mesclagem simples, embora atraente por sua simplicidade, pode levar a uma ordenação de documentos incorreta quando os pares possuem coleções de documento com uma alta topicidade (vide o conceito de topicidade na Seção 1.1). O mecanismo de *mesclagem Kirsch* foi proposto em [22] no contexto de recuperação de informação em repositórios distribuídos, em especial para situações em que as estatísticas do repositório locais divergem significativamente das estatísticas globais (como é o caso em cenários que há topicidade). Entretanto, é importante notar que tanto a eficácia quanto a eficiência desta técnica no contexto de máquinas de busca P2P ainda são questões em aberto, principalmente considerando cenários realistas com um número potencialmente grande (milhares) de pares, exibindo alta dinamicidade e limitações de banda e espaço.

O mecanismo de *mesclagem kirsch* implica no envio por cada par que processa a consulta de algumas estatísticas da sua coleção local, para que o par que recebe as respostas agregue as estatísticas e recalcule a similaridade dos documentos, aproximando o valor ao equivalente global. No caso de nosso processador de consultas, essas estatísticas são os valores de $TF(i)$ de cada termo i da consulta, o número total de documentos do par (D_p) e os valores de $DF_p(i)$ de cada termo (número de documentos do par p que possuem o termo i). Portanto, quando o mecanismo de busca par-a-par por conteúdo estiver utilizando a *mesclagem Kirsch*, a resposta típica de um par contém uma tupla com os valores de DF de cada termo da consulta

$$\langle\ D_p,\ DF(q_0),\ ..., DF(q_i),\ ...,\ DF(q_Q)\ \rangle \tag{3.10}$$

e uma lista de tuplas (uma para cada documento na resposta) do tipo:

$$\langle\ docUID(j),\ sim(q,\ j),\ TF(q_0,\ j), ...,\ TF(q_{0i},\ j),\ ...,\ TF(q_Q,\ j)\ \rangle \tag{3.11}$$

em que $docUID(x)$ uma função que retorna um identificador único do objeto x, j um

documento coleção, q uma consulta, $sim(q,\ j)$ o cálculo de similaridade entre a consulta q e um documento j, Q o número de termos na consulta e $TF(q_i,\ j)$ a frequência do i-ésimo termo da consulta no documento j. Com estas informações, o par, ao receber as respostas, agrega as estatísticas e recalcula a similaridade de todas as respostas ajustando o fator IDF de cada termo q_i utilizando a Equação 3.12. É importante perceber que só é possível recalcular o *idf* desta forma por que utilizamos uma norma estática em nosso modelo, que é independente da composição de documentos da coleção local do par (vide Seção 3.4.4).

$$idf'(q_i) = \lg(\frac{\sum_p D_p}{\sum_p DF_p(q_i)}) \tag{3.12}$$

Note que, no caso mesclagem simples, a resposta de um par consiste apenas de uma lista de tuplas do tipo:

$$\langle\ docUID(j),\ sim(q,\ j)\ \rangle \tag{3.13}$$

Portanto, é fácil perceber que a mesclagem Kirsch implica no envio de uma maior quantidade de informação para cada tupla (ou documento) presente na resposta de cada par a uma consulta. Abaixo mensuramos a quantidade de informação (em número de bytes) tipicamente retornada por cada par para cada um dos mecanismos de mesclagem, com o objetivo de estimar o *overhead*, em termos de consumo de largura de banda de rede, da mesclagem Kirsch. O cálculo deste *overhead* é importante para analisar tanto a viabilidade do mecanismo, quanto possíveis compromissos entre uso de recursos e qualidade de resultados.

Sejam $size(x)$ o número de bytes necessários para representar o parâmetro x, $R_{simples}$ a resposta de um par a uma consulta quando utilizada a mesclagem simples e R_{kirsch} quando utilizada a mesclagem Kirsch, n o número documentos contidos nesta resposta, $docId$ o identificador universal único do documento, sim o seu julgamento de similaridade e $\bar{q}$ o número médio de termos por consulta, temos que:

$$size(R_{simples}) = n * (size(docId) + size(sim)) \tag{3.14}$$

$$size(R_{kirsch}) = n * (size(docId) + \bar{q} * size(TF)) + \bar{q} * size(DF) + size(D) \tag{3.15}$$

Considerando que um *docId*, por ser um identificador universal único, possui 16 *bytes* (número de bytes de um *hash MD5*, por exemplo), que números em representação de ponto flutuante (como o valor de similaridade *sim*) e ponto fixo (como o TF,D e DF) possuem 4 *bytes*, e que o número médio de termos de uma consulta ($\bar{q}$) é seis (valor pessimista), obtemos que o tamanho de cada uma das respostas é:

$$size(R_{simples}) = n * (16 + 4) \tag{3.16}$$

$$size(R_{simples}) = n * 20 \text{ bytes} \tag{3.17}$$

$$size(R_{kirsch}) = n * (16 + 6 * 4) + 6 * 4 + 4 \tag{3.18}$$

$$size(R_{kirsch}) = n * (40) + 28 \tag{3.19}$$

$$size(R_{kirsch}) \approx n * 40 \text{ bytes} \tag{3.20}$$

Portanto, a resposta do mecanismo de *mesclagem kirsch* possui aproximadamente o dobro do tamanho quando comparado ao mecanismo de mesclagem simples (assumindo-se o número médio de termos de uma consulta $\bar{q}$ igual a seis). É importante notar que a estimativa de um número médio de 6 termos consultas é pessimista, pois há na literatura argumentos que o número médio de termos de uma consulta do usuário é entre 2 e 3 termos. Neste caso, o aumento na quantidade de bytes retornada por cada par é de somente 20% e 40%, respectivamente.

3.4.6 Replicação de Documentos por Similaridade

Para dissociar a disponibilidade de um determinado conteúdo da disponibilidade do par que o hospeda, incorporamos à máquina de busca P2P uma estratégia de *replicação* de dados. Utilizamos uma adaptação da estratégia *owner's replication* [29], 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 e popularidade do conteúdo. Além disso, considerando que as respostas das 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 do 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. Em nosso modelo, consideramos uma política de reposição *least frequently used* (LFU), em que o documento da cache com menor número de acessos é removido.

3.5 Desafios à Busca por Conteúdo em Bibliotecas Digitais P2P

Há vários desafios que à eficácia e eficiência da busca par-a-par. Em um cenário prático, como uma biblioteca digital P2P, por exemplo, não é esperado que todos os usuários da biblioteca estejam presentes na rede o tempo todo, isto é, espera-se que os pares que compõem a máquina entrem e saiam da rede de maneira dinâmica . Por este motivo, um dos desafios da máquina de busca P2P em um cenário real é o comportamento dinâmico (*churn*) dos pares.

O *churn* pode afetar a máquina de busca par-a-par em todas as suas etapas de processamento da consulta. Vimos na Seção 3.4.2 que os pares mantém um diretório distribuído estruturado contendo as informações necessárias para o funcionamento Seletor de Pares CORI. A Figuras 3.3(a-b) apresentam um cenário em que o *churn* pode afetar a seleção de pares. No exemplo ilustrado na Figura 3.3(a), o par *peer0* está na primeira etapa da seleção de pares (vide Seção 3.4.1), na qual irá recuperar estatísticas dos termos *"busca"* e *"p2p"* (termos presentes na consulta do usuário), no entanto, os pares do diretório distribuído responsáveis por estes termos (*peer3* e *peer4*) estão (temporariamente ou definitivamente) indisponíveis, como ilustrado na Figura 3.3(b). Neste cenário, o par que iniciou a consulta não obterá resultado algum, visto que não há como selecionar pares sem conhecimento das estatísticas dos termos. Os mecanismos de *post-foward* e de reenvio periódico de *posts* (descritos na Seção 3.4.2) têm como objetivo superar esta indisponibilidade dos pares, replicando e repondo, respectivamente, os *posts* no diretório distribuído.

O comportamento dinâmico dos pares pode impactar também o processamento federado de consultas (Seção 3.4.3). A Figura 3.4(a-b) ilustra a etapa de processamento federado de da consulta *"busca p2p"*, a qual ocorre após o processo de seleção de pares. Na ilustração, os pares *peer3* e *peer4* foram escolhidos pelo Seletor de Pares CORI para o processamento federado de consultas baseado em estatísticas (*posts*) dos termos da consulta encontradas no diretório distribuído. Nesta etapa, o par que iniciou a consulta envia os termos da consulta para cada um dos pares selecionados (Figura 3.4(a)). No entanto, é possível que, devido ao *churn*, os pares selecionados estejam indisponíveis

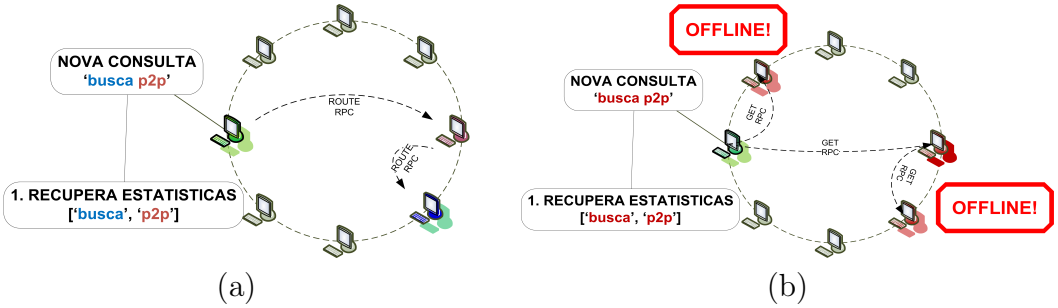


Figura 3.3. *Churn* afetando o processo de seleção de pares. O par que inicia a consulta não consegue obter as estatísticas para os termos 'busca' e 'p2p' no diretório distribuído devido à indisponibilidade do pares responsáveis pelos termos, ou pela indisponibilidade dos pares responsáveis pelo roteamento da mensagem.

(como ilustrado na Figura 3.4(b)) no momento da consulta, e que por causa disso o usuário não receba parte ou todos os resultados de sua consulta.

A indisponibilidade dos pares selecionados pode ocorrer de duas formas: primeiramente, o par pode sair da rede entre o instante da seleção e do envio da consulta; segundo, o diretório distribuído pode conter informações antigas, de pares que há muito abandonaram a rede (de forma temporária ou até mesmo definitiva). Em ambos os casos, o que ocorre é que as informações (*posts*) obtidas no diretório distribuído não correspondem ao estado da rede par-a-par no momento de envio da consulta. O prazo de validade (*time-to-live*) e reenvio periódico dos *posts* no diretório distribuído descritos na Seção 3.4.2 têm também como objetivo de superar este desafio, retirando as informações antigas e atualizando o diretório distribuído periodicamente. Note que quanto menor o período de validade e de reenvio das informações do diretório distribuído, isto é, quanto menor o *time-to-live* dos *posts*, maior será o consumo de banda (maior número de mensagens enviadas) e menor serão as chances de erro na informação obtidas no diretório distribuído devido ao *churn*.

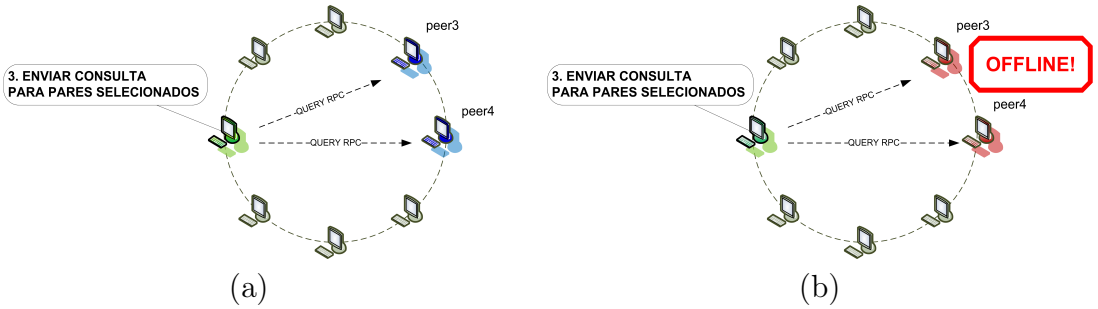


Figura 3.4. *Churn* o processamento federado de consultas. O seletor de pares CORI definiu que o *peer3* e *peer4* eram os melhores pares para o processamento federado da consulta, mas os pares estão indisponíveis.

Outro desafio à eficácia e eficiência da busca P2P é a limitação de recurso dos pares. Os pares são computadores não dedicados à máquina de busca, que possuem limitações de banda e espaço em disco. A limitação de banda dos pares tem maior impacto no tempo de tranmissão e na quantidade de documentos replicados (vide Seção 3.4.6), além de limitar a quantidade de informação que pode ser enviada e recuperada no diretório distribuído (Secões 3.4.1 e 3.4.2), fatos que podem reduzir a eficácia do mecanismo de replicação por similaridade e do seletor de pares, respectivamente. A limitação de espaço dos pares afeta também o mecanismo de replicação, determinando um número máximo de documentos que cada par pode replicar localmente.

Capítulo 4

Metodologia de Avaliação

4.1 Descrição

Como já afirmado anteriormente, o objetivo principal deste trabalho é realizar uma avaliação quantitativa da eficácia e da eficiência (e do compromisso existente entre estes dois aspectos) na busca P2P por conteúdo. 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, nossos modelos de simulação foram decomposto nas três camadas que compõem o modelo de máquina de busca P2P apresentado no Capítulo 3: camada de rede física, camada da rede sobreposta P2P e camada de aplicação.

A avaliação é realizada utilizando dois modelos de simulação distintos: o *modelo simplificado* da máquina de busca P2P, utilizado para analisar apenas o potencial de eficácia da busca P2P; e o *modelo detalhado* da máquina de busca P2P, utilizado para avaliar o compromisso entre eficácia e eficiência da busca P2P por conteúdo. O *modelo simplificado* restringe a avaliação da eficácia máquina de busca a cenários otimistas, pois ignora limitações de banda dos pares, falhas de comunicação e localização de conteúdo na rede, mas em contrapartida, permite a avaliação da máquina de busca em cenários de maior escala, com um grande número de pares (dezenas de milhares) e documentos (milhões). O *modelo detalhado* permite a avaliação tanto da eficácia quanto da eficiência da máquina de busca P2P, já que possui um modelo de rede física e sobreposta detalhado, capaz de simular roteamento mensagem na rede sobreposta e limitação de banda encontradas em sistemas P2P reais. Contudo, o nível de detalhamento deste modelo faz com que a sua avaliação seja limitada a cenários de menor

escala, com um menor número de pares (centenas). As Tabelas 4.1 e 4.2 sumarizam os parâmetros do modelo simplificado e detalhado, respectivamente.

Tabela 4.1. Parâmetros do Modelo Simplificado da Máquinha de Busca P2P.

Parâmetro	Descrição
col	Coleção de teste utilizada (*WBR* ou *TREC-8*).
n	número de pares total que compõem a máquina de busca P2P (*online* e *offline*).
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.
λ_{off}, ρ_{off}	Parâmetros da distribuição Weibull dos tempos de sessão *offline* dos pares.
$\|Q_{set}\|$	Tamanho, em número de documentos, da resposta de um par a uma consulta.
r	Número de documentos replicados a cada consulta.

Tabela 4.2. Parâmetros do Modelo Detalhado de Simulação da Máquinha de Busca P2P.

Parâmetro	Descrição
col	Coleção de teste utilizada (*WBR-TOP1000* ou *TREC-8*).
n	número de pares total que compõem a máquina de busca P2P (*online* e *offline*).
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.
λ_{off}, ρ_{off}	Parâmetros da distribuição Weibull dos tempos de sessão *offline* 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.
α	Parametro de ajuste do Seletor de Pares CORI.
r	Número de documentos replicados a cada consulta.
ttl	Tempo para invalidação de *posts* dos pares no diretório distribuído.
$DF_{threshold}$	Limiar inferior para submissão *posts* no diretório distribuído.

Tanto o *modelo simplificado* quanto o *modelo detalhado* simulam a máquina de busca apresentada no Capítulo 3 com diferentes pressupostos e níveis de abstração. O *modelo simplificado* e o *modelo detalhado* de simulação são descritos nas Seções 4.3 e 4.2. Logo após seguem seções que descrevem o modelo de dinamicidade dos pares (Seção 4.4), coleções de testes (Seção 4.5) e métricas utilizadas para a avaliação da eficiência e eficácia da máquina de busca P2P (Seção 4.6).

4.2 Modelo Detalhado de Máquina de Busca P2P

O *modelo detalhado* da máquina de busca P2P inclui todos os elementos descritos na camada de rede física, sobreposta e aplicação descritos no Capítulo 3. Como seu foco primordial é avaliar o compromisso entre eficácia e eficiência da máquina de busca P2P em condições mais realistas, o modelo de rede física e sobreposta é necessariamente mais detalhado, incluindo o roteamento na rede sobreposta, limitação de banda e outras características encontradas em sistemas P2P reais.

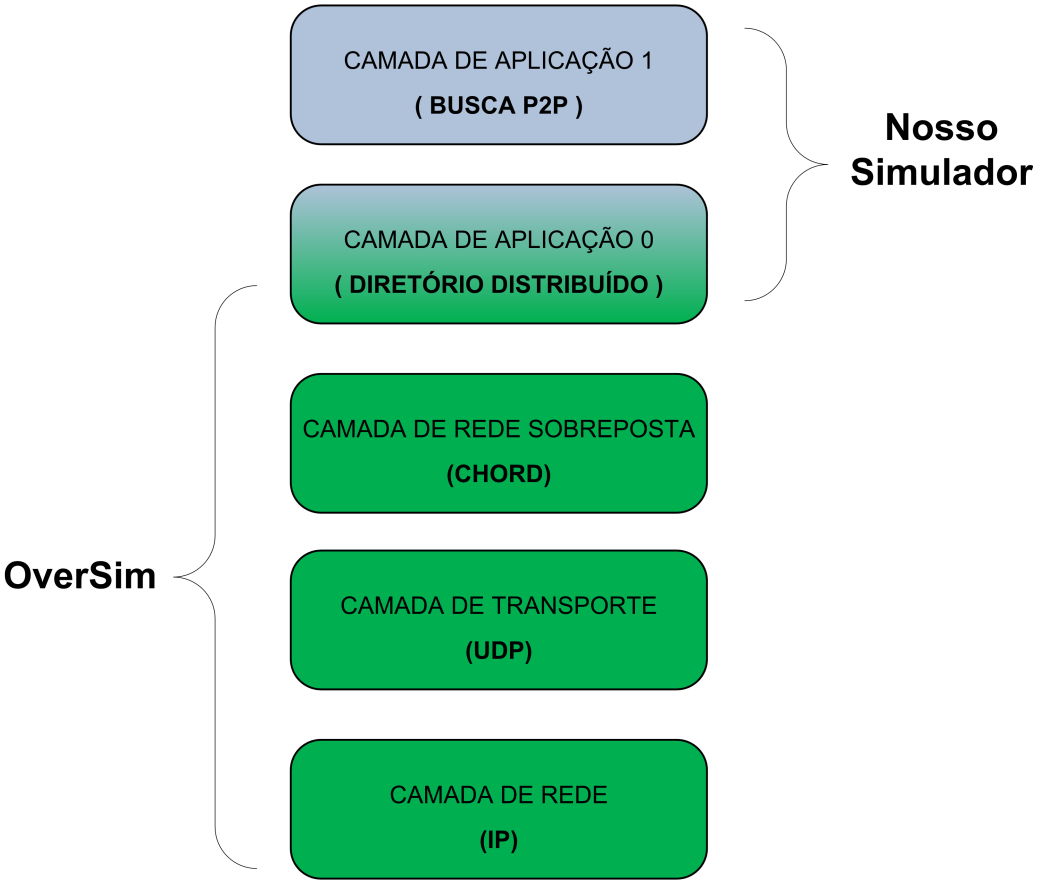


Figura 4.1. Arquitetura em camadas do *modelo detalhado* de simulação construído utilizando o *framework* de simulação OverSim.

Para tanto, foi utilizado o *framework* de simulação OverSim [9]. O *framework* fornece dois modelos de rede física (rede IP e rede IP simplificada, em ambos UDP é o único serviço da camada de transporte disponível) e permite a simulação de diversas redes sobrepostas. A aplicação interage com a rede sobreposta através da *common API*[19], que define um conjunto de serviços comuns a várias redes redes sobrepostas estruturadas, tais como Chord, Kademlia e Pastry. A Figura 4.1 ilustra como o modelo da máquina de busca P2P descrita no Capítulo 3 foi integrada ao simulador *framework* de simulação.

4.3 Modelo Simplificado de Máquina de Busca P2P

O *modelo simplificado* da máquina de busca P2P aqui apresentado é o mesmo utilizado em [6], e corresponde ao modelo descrito no Capítulo 3 levando-se em conta alguns pressupostos e abstrações. Como o objetivo desde modelo é avaliar apenas o *potencial* de eficácia da máquina de busca, e não o seu desempenho ou eficiência, algumas

simplificações são realizadas na camada de rede física, sobreposta e aplicação.

Quanto a rede física e sobreposta, o modelo simplificado assume que: (1) pares disponíveis nos sistema são sempre visíveis entre si, (2) o conteúdo disponível (isto é, os pares que o contêm) pode ser sempre localizado e (3) o roteamento de consultas e respostas nunca falha. Além disso, como a qualidade das respostas do sistema, e não o desempenho, é a métrica de interesse, os tempos de consulta e de resposta são considerados desprezíveis.

Quanto a camada de aplicação (isto é, a máquina de busca P2P), o modelo simplificado assume durante a etapa de seleção de pares, todo o par que contenha pelo menos um termo da consulta é selecionado (dado que não há restrições de banda, não faz sentido selecionar um subconjunto menor de pares).

É importante ressaltar novamente que o objetivo deste modelo em nosso trabalho é analisar o potencial de eficácia da máquina de busca P2P, e por isso muitos do pressupostos e abstrações do modelo são otimistas. No entanto, ressalvamos que o *modelo simplificado* inclui todos os outros elementos do *modelo detalhado* de máquina de busca P2P, tais como a dinamicidade dos pares, o processamento federado de consultas, mesclagem de respostas e replicação por similaridade, e por este motivo serve como um limite superior da eficácia de nosso modelo de máquina de busca P2P.

4.4 Modelo de Comportamento Dinâmico dos Pares

Uma característica importante em nossa avaliação é o modelo 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 [41] de três sistemas populares mostra que a duração da sessão dos pares (*tempo online*) é bem modelada por distribuições Weibull[1] com parâmetros de forma $\rho \approx 0,44$ e de escala $\lambda \approx 35,20$ unidades de tempo. Embora relevante, esse 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.

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

Tabela 4.3. Sumário da Coleção de Teste

Collection (*col*)	# **Docs.**	# **Consultas**
TREC-8	528.155	50
WBR	5.751.296	50 / 1000
WBR-TOP1000	2.005.340	50 / 1000

4.5 Coleções de Teste

Utilizamos duas coleções de teste: WBR e TREC-8 ad-hoc (ou TREC, no decorrer do texto). A TREC [44] é constituída por um total de 528.125 documentos, submetidos à trilha *ad-hoc task* da conferência TREC-8 para a avaliação de algoritmos de Recuperação de Informação, e incluem textos de jornais e documentos do governo dos EUA. A WBR [12] é composta por 5.751.296 documentos coletados da Web brasileira pela máquina de busca TodoBR. Devido a impossibilidade de utilizar a coleção WBR em nosso *modelo detalhado* de simulação, construímos uma coleção derivada da WBR, denominado WBR-TOP1000, que inclui os 1000 *hosts* (que em nossas avaliações equivalem à um par) com maior número de documentos na coleção, o que corresponde a aproximadamente 35% dos documentos da coleção original.

A Tabela 4.3 sumariza as coleções avaliadas. Cada coleção possui, além de documentos, um conjunto de consultas para avaliação da máquinas de busca. A TREC possui 50 consultas manualmente elaboradas e avaliadas por especialistas. A WBR fornece dois conjuntos de consultas: o primeiro conjunto, de modo semelhante a TREC, contém 50 consultas avaliadas por especialistas; o segundo conjunto contém as 1000 consultas mais populares da máquina de busca e a distribuição de popularidade das mesmas.

4.6 Métricas de Avaliação

Utilizaremos duas métricas para avaliar qualidade da máquina da busca P2P: a *revocação relativa* e a média aritmética da Precisão Média (MAP, *Mean Average Precision*). A *revocação relativa* mensura a fração dos resultados, em relação a uma máquina de busca centralizada, que são apresentados pela máquina de busca P2P. A MAP utiliza o julgamento de relevância fornecidos pelas coleções (WBR e TREC) para medir a qualidade da busca P2P.

Para avaliar a eficiência de uma máquina de busca P2P, medimos a quantidade média de recursos utilizados por cada par da máquina de busca para atingir um determinado nível de eficácia. Em nosso modelo, os únicos recursos limitados dos pares são

banda e *espaço em disco*, e portanto estes são os recursos mensurados.

Ao analisar o compromisso entre eficácia e eficiência da busca P2P, comparamos a razão da variação da eficácia e pela variação da eficiência nos diversos cenários. É importante notar que isto implica que a eficiência é proporcionalmente equivalente (tem mesmo importância) que a eficácia, o que não é necessariamente verdade em um caso de uso real.

4.6.1 Revocação Relativa

A revocação é definida como a fração de documentos relevantes à consulta que é retornado por uma 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:

$$r(q) = \frac{|relevantes(q) \cap retornados(q)|}{|relevantes(q)|} \tag{4.1}$$

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.

$$r_{rel}(q) = \frac{|retornados_{central}(q) \cap retornados_{p2p}(q)|}{|retornados_{central}(q)|} \tag{4.2}$$

Note que a revocação, como definida acima, aplica-se somente ao resultado final da máquina de busca P2P, e por este motivo não distingue a eficácia da *seleção de pares* e do *processamento federado de consultas*. Para tanto, definimos a revocação da seleção de pares , $r_{selecao}(q)$, como a fração de documentos relevantes à consulta que se encontram presentes no conjunto de pares selecionados P. 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 P para a consulta q:

$$r_{selecao}(q, P) = \frac{|relevantes(q) \cap selecionados(P)|}{|relevantes(q)|} \tag{4.3}$$

$$selecionados(P) = \{ \bigcup_{p \in P} documentos(p) \} \tag{4.4}$$

Note que $r_{selecao}(q)$ para uma consulta q define um limite superior para a $r(q)$. 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.

$$r_{selecao\ rel}(q, P) = \frac{|retornados_{central} \cap selecionados(P)|}{|retornados_{central}|} \tag{4.5}$$

A revocação relativa é uma métrica comumente utilizada para mensurar a eficácia de máquinas de busca P2P [32], e possui a vantagem de permitir a avaliação da busca P2P por conteúdo com conjuntos de consultas que não possuem julgamentos de relevância produzido por especialistas (e.g. o conjunto das 1000 consultas mais populares da coleção WBR e WBR-TOP1000, descritas na Seção 4.5).

4.6.2 MAP

A MAP é mais uma métrica utilizada em nosso trabalho para quantificar a qualidade da busca. Em especial, utilizaremos esta métrica para confrontar nossos resultados aos produzido em [6]. Dado o conjunto *relevantes* dos documentos relevantes da coleção por consulta, a lista de Q consultas processadas, a lista dos K documentos mais similares à consulta q de acordo com a máquina de busca, o julgamento binário de relevância $x_{q,i}$ para o i-ésimo documento da lista de documentos respondida a consulta q e a precisão do m-ésimo documento desta lista:

$$p_{q,m} = \frac{1}{m} \sum_{i=1}^{m} x_{q,i} \tag{4.6}$$

, define-se formalmente [23] MAP como:

$$MAP = \frac{1}{|Q|} \sum_{q=1}^{Q} \sum_{i=1}^{K} \frac{x_{q,i} p_{q,i}}{|relevantes_q|} \tag{4.7}$$

Isto é, MAP fornece a média aritmética da precisão média (AP) de cada consulta, que por sua vez fornece a fração de documentos relevantes, ponderados pela sua posição na lista de documentos recuperados. A MAP é uma métrica amplamente utilizada para medir a eficácia de algoritmos de recuperação de informação, incorporando fatores como precisão e revocação sem desprezar a ordem dos documentos na resposta.

Capítulo 5

Avaliando o Potencial de Eficácia da Máquina de Busca P2P

5.1 Descrição

Iniciamos nossa avaliação através do *modelo simplificado* (descrito na seção 4.3) de máquina de busca P2P, que nos permite mensurar o *potencial de eficácia* da busca P2P em cenários de maior escala (maior número de pares e documentos). Esta avaliação nos permite avaliar o a eficácia máxima que uma máquina de busca P2P seguindo o modelo descrito no Capítulo 3 pode atingir frente à dinamicidade dos pares e à topicidade, além de permitir mensurar individualmente o impacto destas características sobre a eficácia da busca P2P por conteúdo.

Para tanto, consideramos cenários com diversos níveis de dinamicidade (A), analisando o impacto para ambas coleções de teste, a TREC e a WBR em sua configuração original ($col = [TREC, WBR]$). Nossos resultados são apresentados como médias da revocação relativa e MAP (vide Seção 4.6) de diversas execuções de um mesmo cenário. Devido a diferenças significativas entre as coleções (ver Tabela 4.3), atribuímos a cada coleção uma distribuição de documentos diferente. Na WBR, cada documento está vinculado a uma URL, o que nos permite fazer uma distribuição direta entre pares e *hosts*: cada uma das n_{wbr} máquinas hospedeiras é associada aleatoriamente a um par e as páginas Web da máquina hospedeira são associadas aos documentos do par correspondente. Por outro lado, como a $TREC$ não possui esta característica *Web*, distribuímos uniformemente os documentos por cada um dos n_{trec} pares. O número de pares para os cenários que utilizam é WBR é igual ao número de *hosts* da coleção ($n_{wbr} = 110912$). O número de pares nos cenários que utilizam a $TREC$ foi escolhido

para efeitos de comparação aos resultados anteriores [6] ($n_{trec} = 880$).

Para avaliar o impacto de diferentes níveis de dinamicidade, fixamos a distribuição dos tempos *online* dos pares com uma média de μ_{on}=91, 84 unidades de tempo, dada pelos parâmetros Weibull ρ_{on} e λ_{on} iguais a 0, 44 e 35, 20 respectivamente (ver Seção 4.4). Além disso, fixamos ρ_{off}=ρ_{on} enquanto variamos λ_{off}, produzindo diferentes tempos médios *offline* (μ_{off}). Finalmente, de forma a capturar a estabilidade do sistema, define-se [31] disponibilidade do par, A, como sendo o percentual do tempo (simulado) que o par participa da rede, em média. Isto é:

$$A = \frac{\mu_{on}}{\mu_{on} + \mu_{off}} \tag{5.1}$$

Utilizamos em nossos cenários disponibilidades de 25%, 50% e 75% e 100%, sendo este último valor utilizado apenas para compor o caso base (*baseline*).

Consideramos, também, cenários em que os pares possuem conhecimento de estatísticas globais TF-IDF (GK) ou apenas estatísticas locais (LK). O cenário GK corresponde a redes onde os pares conseguem, de maneira altamente colaborativa, manter todas as estatísticas TF-IDF acessíveis no diretório distribuído, enquanto o cenário LK corresponde a um cenário que os pares agem de maneira mais independente. Nos cenários em que são utilizadas apenas estatísticas locais surge o problema da *topicidade*, e por este motivo, apenas nestes cenários avaliaremos os ganhos na eficácia obtidos através da utilização do mecanismo de *mesclagem kirsch* (descrito na Seção 3.4.5).

Os resultados reportados neste capítulo são resultados de no mínimo 4 experimentos com coeficiente de variação inferior a 8%.

5.2 Impacto da Dinamicidade dos Pares e da Topicidade na Eficácia Máquina de Busca P2P

Nesta seção mensuramos o impacto da dinamicidade e da topicidade na busca P2P por conteúdo. É importante notar que estes primeiros resultados consistem dos mesmos experimentos executados em [6], mas adaptados ao nosso modelo de máquina de busca apresentado no Capítulo 3. Em [6] foi demonstrado que a dinamicidade dos pares pode , por si só, pode reduzir significativamente a eficácia da máquina de busca P2P. Este trabalho utilizou um processador de consultas baseado na métrica de similaridade tradicional (Equacao 3.8). Nesta seção, nós revisitamos esta análise utilizando o nosso processador de consultas com norma modificada (Equação 3.9).

As Tabelas 5.1 e 5.2 mostram os valores da revocação relativa (MAP en-

tre parênteses) e sua degradação em relação ao equivalente centralizado, respectivamente, para diferentes disponibilidades (A) e para dois níveis de conhecimento, global ($conhecimento = GK$) e local ($conhecimento = LK$). As células centrais apresentam a revocação relativa e a MAP (este último apresentado entre parênteses). Os resultados são apresentados para cenários em que os pares possuem 75%, 50% e 25% de disponibilidade média. Nos cenários em que os pares possuem conhecimento das estatísticas globais ($conhecimento = GK$), toda degradação da eficácia da máquina de busca P2P pode ser atribuída exclusivamente à dinamicidade dos pares, isto é, toda redução da revocação relativa é devido à indisponibilidade dos pares na rede, dado que o processamento da consulta utiliza estatísticas livres de erro. Já nos cenários em que os pares possuem apenas conhecimento local ($conhecimento = LK$), a degradação da eficácia da máquina de busca pode ser atribuída tanto à dinamicidade quanto à topicidade, em outras palavras, a redução da revocação relativa é causada tanto pela indisponibilidade dos pares quanto pelo fato de suas estatísticas locais divergirem das estatísticas da coleção de documentos global. Portanto, com o intuito de mensurar isoladamente o impacto da topicidade na máquina de busca, podemos observar a degradação da eficácia entre cenários LK e GK equivalentes, isolando assim os efeitos da dinamicidade e topicidade.

Tabela 5.1. Valor absoluto da revocação relativa (MAP entre parênteses) da busca P2P por conteúdo sob efeito da dinamicidade e topicidade.

	TREC		**WBR**	
A %	**GK**	**LK**	**GK**	**LK**
centralizado	1,0 (0,0558)		1,0 (0,0973)	
75	0,751 (0,046)	0,747 (0,040)	0,759 (0,076)	0,473 (0,052)
50	0,499 (0,035)	0,498 (0,031)	0,500 (0,057)	0,352 (0,042)
25	0,246 (0,018)	0,246 (0,016)	0,246 (0,038)	0,194 (0,027)

Tabela 5.2. Degradação da revocação relativa (degradação da MAP entre parênteses) da busca P2P por conteúdo devido à dinamicidade e topicidade.

	TREC		**WBR**	
A %	**GK**	**LK**	**GK**	**LK**
centralizado	0 (0)		0 (0)	
75	24,9% (17,0%)	25,3% (28,1%)	24,1% (22,2%)	52,7% (46,1%)
50	50,1% (36,4%)	50,2% (43,7%)	50,0% (41,2%)	64,8% (57,2%)
25	75,4% (68,3%)	75,4% (71,9%)	75,4% (60,4%)	80,6% (72,3%)

Os resultados apresentados nas Tabelas 5.1 e 5.2 confirmam os resultados anteriores que apontam que a alta dinamicidade dos pares, por si só, pode degradar signifi-

cativamente a eficácia da busca P2P por conteúdo [6]. Observando os cenários em que os pares possuem conhecimento global (GK), podemos notar que mesmo quando os pares possuem uma disponibilidade média de 75% (considerada alta para redes par-a-par), a degradação da revocação relativa e da MAP da máquina de busca P2P chega a 24,1% e 22,2%, respectivamente, enquanto em cenários de maior indisponibilidade dos pares (A=25%) a degradação da revocação relativa chega a até 75,4% (68,3% no caso da MAP). Note que a degradação eficácia da busca P2P para estes cenários (GK) é semelhante para as duas coleções de documentos ($col = [TREC, WBR]$), o que sugere que o impacto da dinamicidade na eficácia da busca P2P independe da distribuição dos documentos ou do número de pares na rede. Voltaremos a esta discussão mais tarde (apresentaremos uma análise de variância sobre estes dados).

Ainda observando a Tabela 5.1, notamos que, em cenários que os pares possuem apenas conhecimento de estatísticas de sua coleção local de documentos (LK), esta degradação chega a ser ainda maior, em especial para a coleção WBR. Estes resultados apontam que a topicidade e a dinamicidade (isto é, o erro nas estatísticas locais e a indisponibilidade dos pares) podem degradar a eficácia da máquina de busca P2P em até 25,3% da revocação relativa (28% no caso da MAP) no cenário $col = TREC$ e em até 52,7% da revocação relativa (46,1% no caso da MAP) para $col = WBR$, mesmo em cenários que os pares possuem uma alta disponibilidade ($A = 75\%$). Em cenários de baixa disponibilidade média dos pares ($A = 25\%$) essa degradação chega a até 80,6%. É interessante notar que o impacto da topicidade é muito maior quando $col = WBR$ que quando $col = TREC$.

A diferença dos resultados de eficácia da busca P2P nos cenários LK é devida a maior *topicidade* da coleção de documentos WBR em relação à $TREC$. A *topicidade* foi identificada em [6] no contexto de busca P2P, e pode ser definida como a cobertura, por parte de um par, de um conjunto de documentos sobre um determinado tópico de interesse. Pares com uma alta topicidade possuem documentos com muitos termos em comum, com estatísticas locais muito diferentes da coleção de documentos global, enquanto pares com uma menor topicidade possuem uma distribuição termos e documentos mais similar à coleção global, e portanto, suas estatísticas locais possuem valores mais semelhantes às estatísticas globais.

A coleção WBR possui naturalmente uma alta topicidade, pois seus documentos são distribuídos entre os pares seguindo a distribuição de *hosts* da Web brasileira de 1999. A coleção $TREC$ não possui alta topicidade, pois a distribuição de documentos é uniforme. Entretanto, mesmo com uma baixa topicidade, o erro introduzido pelas estatísticas locais é suficiente para reduzir sensivelmente a eficácia da busca P2P. De

fato, realizando a análise de variância[1] [21] para a revocação relativa com níveis $A = [0,25, 0,75]$ e *conhecimento* $= [LK, GK]$, obtemos que, para a coleção WBR, cerca de 78,7% da variação revocação relativa é devido exclusivamente ao fator A (isto é, devido à dinamicidade dos pares) e 14,3% da variação é devido exclusivamente ao fator *conhecimento* (em outras palavras, devido à topicidade). Para coleção $TREC$ utilizando os mesmo fatores, a análise de variância aponta que o fator A explica 99,996% da variação da revocação relativa, e apenas 0,002% da variação da revocação relativa é devido ao fator *conhecimento*. Resultado semelhante ocorrem analisando a variância da MAP para a WBR e $TREC$. A Tabela 5.3 apresenta todos os resultados desta análise de variância para a revocação relativa e MAP (esta última apresentanda entre parêntesesna Tabela), incluindo ainda a percentagem da variação não explicada.

Tabela 5.3. Percentagem de variação da revocação relativa (MAP entre parênteses) devido ao fator A (dinamicidade dos pares) e ao fator *conhecimento* (topicidade).

	Percentagem Da Variação	
Fator	**TREC**	**WBR**
conhecimento	0,002% (2,3%)	14,3% (20,7%)
A	99,996% (97,1%)	78,7% (76,2%)
Residual	0,002% (0,6%)	6,94% (3,0%)

Esta diferença de impacto da topicidade na $TREC$ e WBR pode ser entendida melhor ao observamos atentamente o modelo de processamento de consultas descrito na Seção 3.4.3. Como a máquina de busca proposta utiliza o modelo vetorial $TF-IDF$ [8], o par que possui uma coleção com alta topicidade, ao utilizar apenas estatísticas locais, retorna documentos com uma similaridade muito menor. Isto ocorre porque o componente IDF é amplamente reduzido para termos muito frequentes na coleção. Como dito anteriormente, a coleção WBR possui naturalmente uma alta topicidade, pois seus documentos são distribuídos entre os pares seguindo a distribuição de *hosts* da Web brasileira de 1999. A coleção $TREC$ não possui alta topicidade, pois a distribuição de documentos é uniforme, e por isso sua topicidade é reduzida.

[1] A análise de variância que aqui nos refereimos é tradicionalmente conhecida como ANOVA Fatorial 2^2 com repetição [21], e é comumente utilizado para analisar o impacto dos fatores em um experimento.

5.3 Impacto da Replicação de Documentos por Similaridade na Eficácia Máquina de Busca P2P

Como vimos na seção anterior, a dinamicidade dos pares, por si só, pode reduzir significativamente a eficácia da máquina de busca par-a-par. Nesta seção, avaliamos o impacto positivo que a replicação por similaridade (apresentada na Seção 3.4.6) apresenta na eficácia da busca P2P, considerando diferentes valores do parâmetro r (números de documentos replicados por consulta). Como o objetivo da replicação por similaridade é amenizar os efeitos da indisponibilidade dos pares, utilizaremos cenários com baixa disponibilidade para verificar sua eficácia (disponibilidade média $A = 25\%$).

Nosso experimento consistiu na execução de um total de 1500 consultas, de um conjunto 50 consultas distintas executadas em *round-robin* e repetidas 30 vezes, para cada coleção. Simulamos cenários em que os pares possuem conhecimento de estatísticas globais (GK) e cenários em que os pares utilizam apenas informação local (LK), utilizando o mecanismo de mesclagem simples apresentado na Seção 3.4.5.

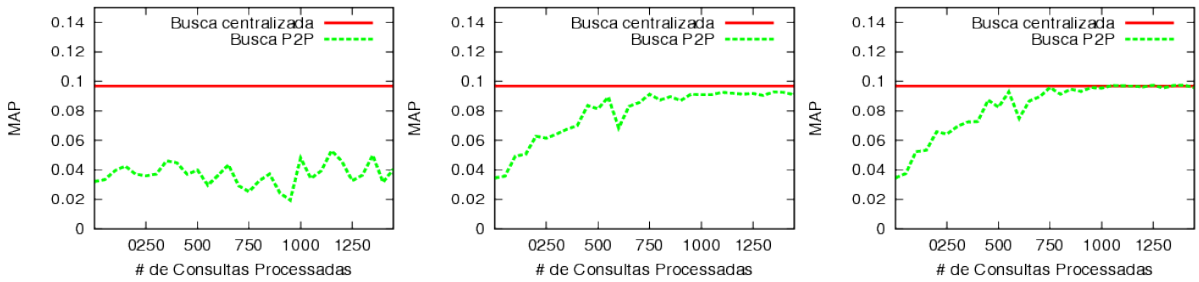
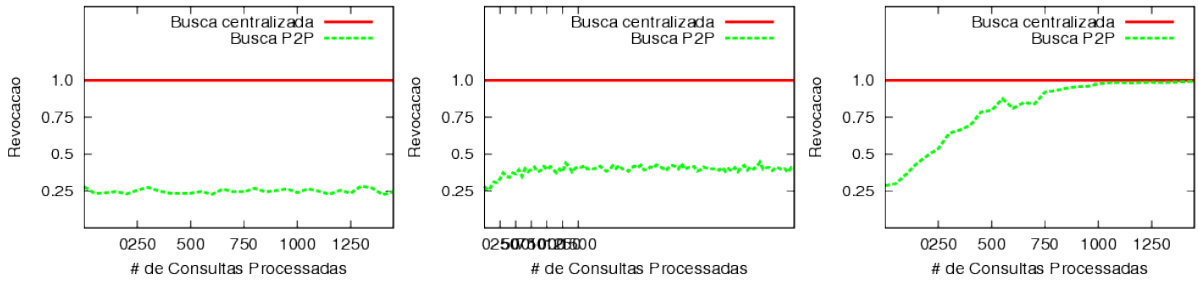


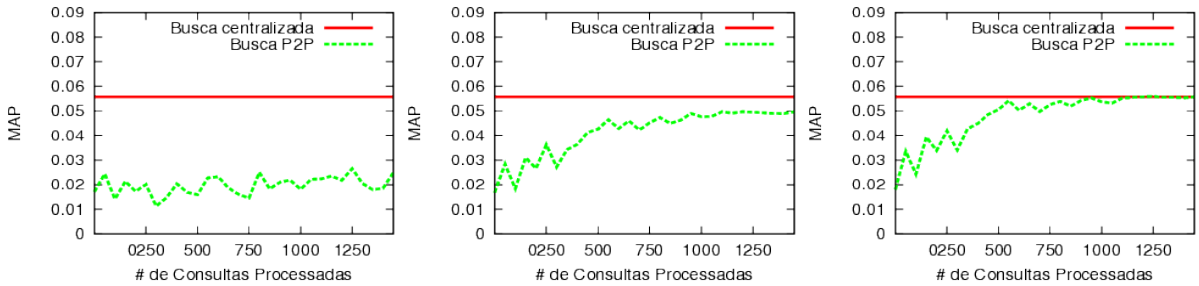
Figura 5.1. *MAP* para a *coleção WBR* no decorrer do tempo, utilizando conhecimento *global* e *mesclagem simples*. (C=∞, $Q_{set} = 1000$)

A Figuras 5.2 e 5.1 apresentam no eixo Y a evolução da revocação relativa e da MAP, respectivamente, e o número de consultas processadas no eixo X, nos cenários em que os pares possuem conhecimento de estatísticas globais (GK) e utilizam mesclagem simples de respostas, para a coleção WBR. Percebe-se que a eficácia da busca P2P oscila muito abaixo do equivalente centralizado nos cenários em que a replicação por similaridade não é utilizada ($r = 0$). Com $r = 1000$ a revocação relativa e a MAP da busca P2P atinge um valor superior a 94% quando comparado ao *baseline*. Valores equivalentes ocorrem para a $TREC$ e são apresentados nas Figuras 5.4 e 5.3.



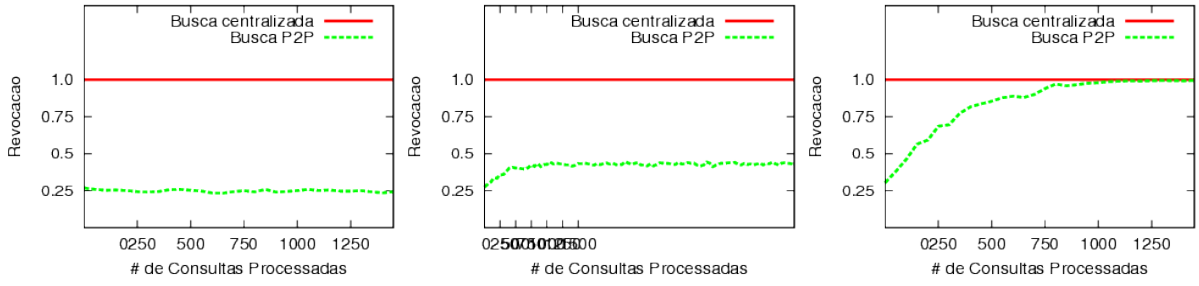
(r=0) (r=100) (r=1000)

Figura 5.2. Revocação Relativa para a *coleção WBR* no decorrer do tempo, utilizando conhecimento *global* e *mesclagem simples*. (C=∞, $Q_{set} = 1000$)



(r=0) (r=100) (r=1000)

Figura 5.3. *MAP* para a *coleção TREC-8* no decorrer do tempo, utilizando conhecimento *global* e *mesclagem simples*. (C=∞, $Q_{set} = 1000$)



(r=0) (r=100) (r=1000)

Figura 5.4. Revocação Relativa para a *coleção TREC-8* no decorrer do tempo, utilizando conhecimento *global* e *mesclagem simples*. (C=∞, $Q_{set} = 1000$)

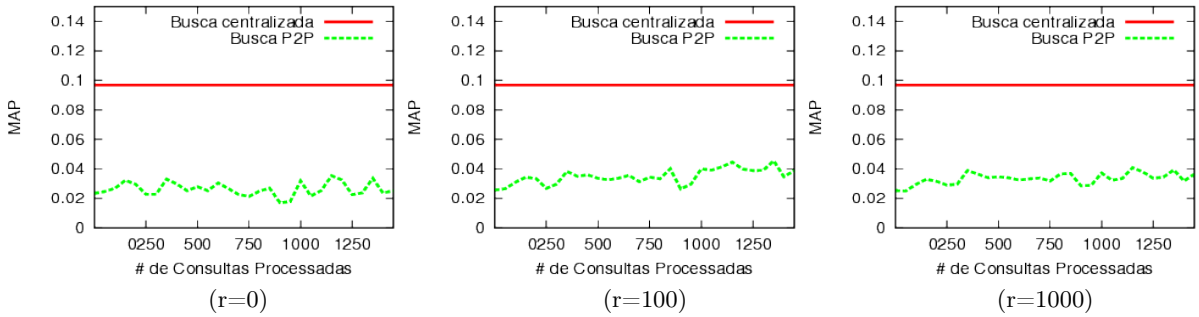
É interessante notar que para $r = 100$, a MAP da busca P2P chega a 94,6% do equivalente centralizado, enquanto a revocação relativa chega à apenas 40%. Isso se deve a diferença fundamental entre as duas métricas: a MAP considera apenas o

conjunto de arquivos relevantes julgados por especialistas fornecido por cada coleção (TREC e WBR), enquanto a revocação relativa utiliza os documentos retornados pela máquina de busca centralizada como conjunto de documentos relevantes. O importante nesta diferença é que o maior conjunto de documentos relevantes selecionados por especialistas possui apenas 60 documentos, enquanto na revocação relativa são considerados relevantes os top-1000 documentos retornados pela busca centralizada. É fácil ver que $r = 100$ é um valor mais que suficiente para replicar os 60 documentos mais relevantes, no caso da MAP, mas pode ser um valor pequeno para replicar os 1000 documentos mais relevantes no caso da revocação relativa. Esse problema se torna pior se mais de r documentos relevantes estiverem em um mesmo par, pois nesse caso, os documentos relevantes restantes não serão replicados nunca, já que pela definição do mecanismo de *replicação por similaridade* apenas os $top - r$ documentos são replicados. Portanto, a eficácia de um determinado valor de r depende da distribuição de documentos em uma determinada coleção.

No geral, os resultados indicam que a replicação por similaridade pode reduzir significativamente a degradação da eficácia da máquina de busca causada exclusivamente pela dinamicidade dos pares. A replicação por similaridade, por sua definição, aumenta a disponibilidade dos documentos mais semelhantes às consultas. Como os pares possuem conhecimento de estatísticas TF-IDF coleção global (GK), o cálculo de similaridade dos documentos de cada par é equivalente ao centralizado, e portanto, os documentos relevantes que estiverem disponíveis na rede serão recuperados pela máquina de busca P2P de modo semelhante ao equivalente centralizado. Depois de replicados, estes documentos possuem mais chances de serem recuperados no futuro, dado que agora possuem uma maior disponibilidade média, e por este motivo, a eficácia da máquina de busca P2P aumenta a cada nova consulta processada.

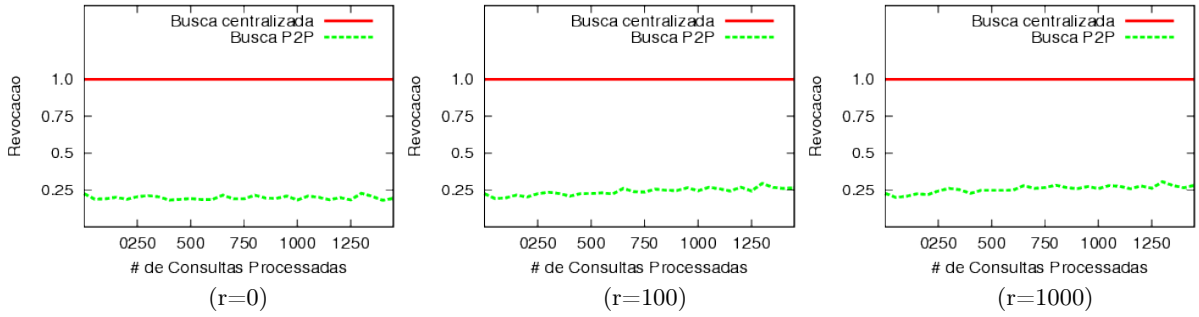
Apesar dos bons resultados apresentados nestes cenários, é importante notar que há na literatura registro que pode ser inviável a manutenção de todas as estatísticas TF-IDF no diretório distribuído [26]. Portanto, uma avaliação mais realista seria mensurar a eficácia da busca P2P por conteúdo em cenários em que cada par utiliza apenas estatísticas locais para calcular a similaridade dos documentos a uma consulta (LK). Como discutido na Seção 5.2, a degradação na eficácia da máquina de busca P2P nestes cenários é causada pela *dinamicidade* dos pares e pela *topicidade* da coleção de documentos de cada par.

A Figura 5.6 e 5.5 apresenta a evolução da revocação relativa e MAP, respectivamente, nos cenários que os pares possuem apenas conhecimento de estatísticas locais para a coleção WBR. Como podemos notar, tanto a revocação relativa quanto o MAP oscila muito abaixo do equivalente centralizado, mesmo nos cenários com alta replica-



(r=0) (r=100) (r=1000)

Figura 5.5. *MAP* para a *coleção WBR* no decorrer do tempo, utilizando conhecimento *local* e *mesclagem simples*. (C=∞, $Q_{set} = 1000$)



(r=0) (r=100) (r=1000)

Figura 5.6. Revocação Relativa para a *coleção WBR* no decorrer do tempo, utilizando conhecimento *local* e *mesclagem simples*. (C=∞, $Q_{set} = 1000$)

ção ($r = 1000$). Em todos os cenários, a degradação da revocação relativa oscila entre 72,5% e 80,3% (63,2% e 71,7% no caso da MAP) quando comparado ao centralizado. Estes valores de degradação são altos mesmo quando deixamos de utilizar o a busca centralizada como *baseline* e comparamos os resultados a um cenário equivalente GK. Por exemplo, no cenário LK em que o par replica os 1000 documentos mais similares à sua consulta ($r = 1000$) a revocação relativa foi degrada em 74,6% (65,14% no caso da MAP), enquanto no mesmo cenário GK a degradação da revocação relativa foi de apenas 6% (apenas 1% no caso da MAP). Essa discrepância nos resultados dos cenários GK e LK se deve, novamente, à *topicidade*.

Como já vimos, a *topicidade* faz com que o processador calcule valores errados de similaridade dos documentos com relação às consultas. Como o mecanismo de *replicação por similaridade* utiliza estes valores para escolher quais documentos replicar, a sua eficácia pode ser substancialmente reduzida na presença da topicidade. Isto ocorre no cenário LK da WBR, pois esta coleção possui naturalmente uma alta topicidade.

A coleção $TREC$, no entanto, possui uma distribuição uniforme de documentos, e por este motivo espera-se que os efeitos da topicidade para esta coleção seja reduzido. As Figuras 5.8 e 5.3 apresentam a revocação relativa a MAP, respectivamente, para o cenário LK utilizando a coleção $TREC$ com valores de $r = [0, 100, 1000]$. Como podemos observar nas figuras, a degradação da revocação relativa oscila entre 34,6% e 75,2% (20,9% e 66,5% no caso da MAP). Assim como no cenário da WBR, a degradação da eficácia da máquina de busca é bem superior neste cenário quando comparado ao equivalente GK. No cenário GK utilizando a $TREC$ e $r = 1000$ a degradação da revocação relativa é de apenas 1,5% (1,6% no caso da MAP), enquanto para o cenário equivalente LK a degradação chega a 34,6% (20,9% no caso da MAP). Novamente, esta discrepância na degradação se deve à topicidade.

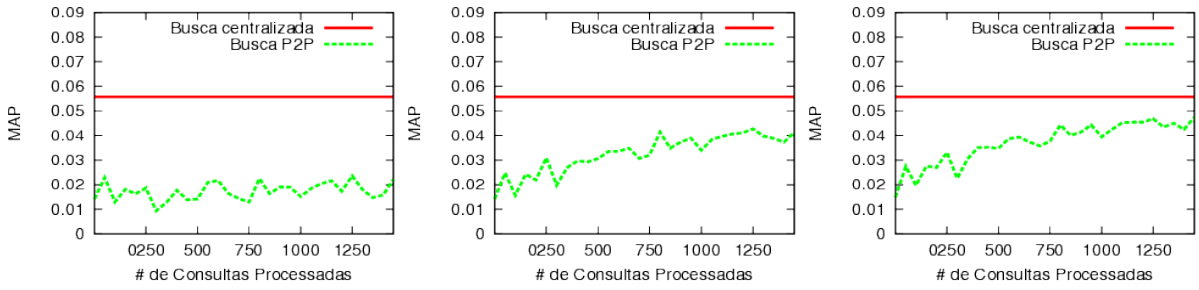


Figura 5.7. *MAP* para a *coleção TREC-8* no decorrer do tempo, utilizando conhecimento *local* e *mesclagem simples*. (C=∞, $Q_{set} = 1000$)

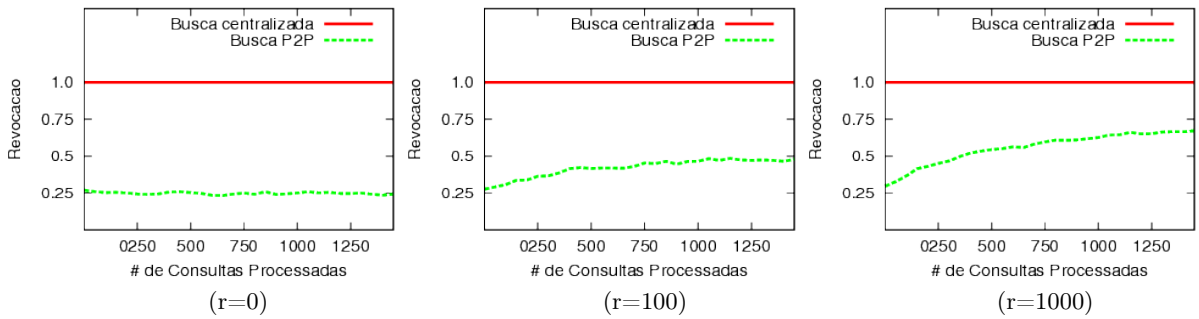


Figura 5.8. Revocação Relativa para a *coleção TREC-8* no decorrer do tempo, utilizando conhecimento *local* e *mesclagem simples*. (C=∞, $Q_{set} = 1000$)

Embora ocorra em menor escala na TREC devido a distribuição uniforme de documentos, a topicidade ainda degrada a a eficácia da busca P2P neste cenário. Isto se

deve principalmente a dois motivos: primeiramente, apesar de uniforme, a distribuição de documentos não é perfeita, de modo que as estatísticas locais dos pares podem diferir substancialmente da estatísticas globais, especialmente para os termos mais raros; segundo, a replicação por similaridade induz uma topicidade em torno dos tópicos das consultas, dado que apenas documentos que possuem alta similaridade com as consultas são replicados. As diferenças entre a *estatísticas locais dos pares* e este *efeito induzido de topicidade* reduzem a da replicação por similaridade no cenário LK da TREC.

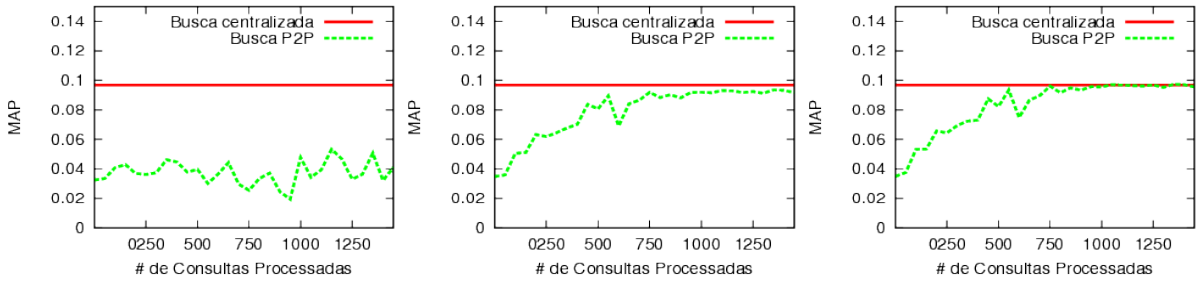
Portanto, o mecanismo de replicação por similaridade, em cenários que os pares possuem apenas conhecimento local, não é suficiente para garantir a eficácia da busca P2P por conteúdo. Veremos na Seção 5.4 que o uso da *meclagem kirsch* pode aproximar os resultados da máquina de busca P2P que utiliza apenas conhecimento local a valores equivalentes aos cenários em que os pares utilizam conhecimento global.

5.4 Impacto da Mesclagem Kirsch na Eficácia Máquina de Busca P2P

Como vimos na seção anterior, a replicação por similaridade apresentou-se como um mecanismo eficiente na redução da degradação da eficácia da máquina de busca P2P causada pela dinamicidade dos pares, porém demonstrou ser uma péssima alternativa nos cenários em que há degradação na eficácia da busca P2P devido à topicidade. Com o intuito de reduzir a degradação causada pela topicidade, adaptamos a *mesclagem Kirsch* [22] ao contexto da busca par-a-par por conteúdo. O funcionamento do mecanismo de mesclagem Kirsch está descrito em detalhes na Seção 3.4.5.

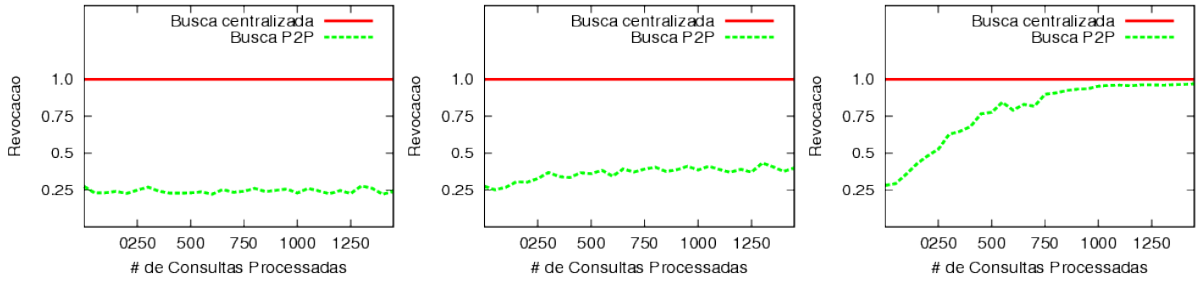
As Figuras 5.10, 5.9), 5.12 e 5.11 apresentam a evolução da revocação relativa e da MAP nos cenários em que os pares possuem conhecimento apenas de estatísticas locais (LK) e utilizam o *mesclagem kirsch*, para as coleções WBR e $TREC$ e $r = [0, 100, 1000]$. Note que estes resultados diferem em no máximo 4% dos cenário em que os pares utilizam informação global (GK) (Figuras 5.1 e 5.3).

A Tabela 5.4 apresenta os resultados da análise de variância para os cenários apresentados nas Figuras 5.9 e 5.11. Como podemos ver, a mesclagem kirsch tem menor importância (explica uma menor parte da variação) nos cenários da coleção $TREC$, e possui maior importância (explica uma maior parte da variação) nos cenários da coleção WBR. É interessante notar que há uma grande parte da variação explicada pela interação entre os dois fatores. Esta interação pode ser explicada pelo fato da meclagem kirsch modificar diretamente o cálculo da similaridade, que por sua vez,



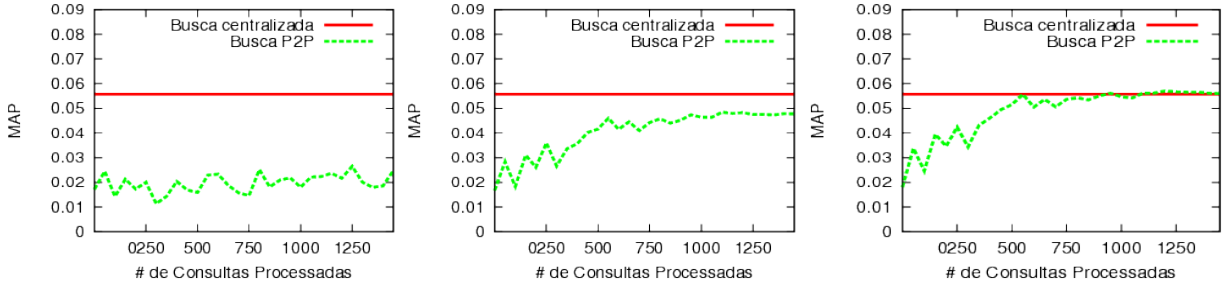
(r=0) (r=100) (r=1000)

Figura 5.9. *MAP* para a *coleção WBR* no decorrer do tempo, utilizando conhecimento *local* e *mesclagem Kirsch.*(C=∞, $Q_{set} = 1000$)



(r=0) (r=100) (r=1000)

Figura 5.10. Revocação Relativa para a *coleção WBR* no decorrer do tempo, utilizando conhecimento *local* e *mesclagem Kirsch.*(C=∞, $Q_{set} = 1000$)



(r=0) (r=100) (r=1000)

Figura 5.11. $MAP(k)$ para a *coleção TREC-8* no decorrer do tempo, utilizando conhecimento *local* e *mesclagem Kirsch.* (C=∞, $Q_{set} = 1000$)

é utilizada pelo mecanismo de replicação para decidir que documentos serão replicar. Estes resultados apontam uma relação sinérgica entre a mesclagem Kirsch e a replicação por similaridade, e que estas técnicas podem elevar a eficácia da máquina de busca P2P

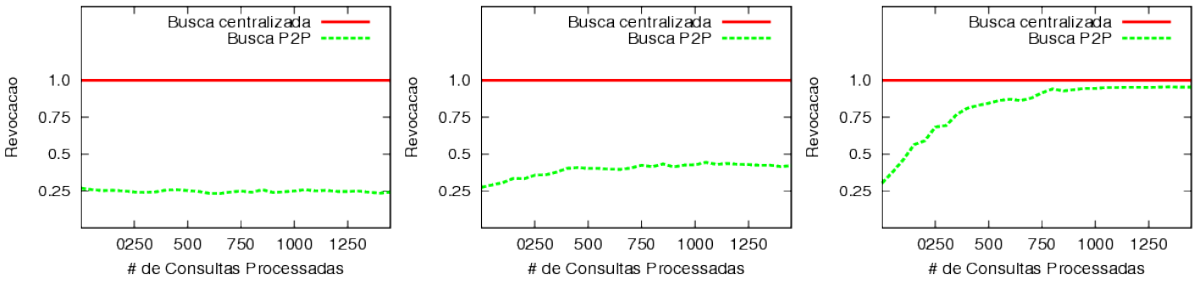


Figura 5.12. Revocação Relativa para a *coleção TREC-8* no decorrer do tempo, utilizando conhecimento *local* e *mesclagem Kirsch*. (C=∞, Q_{set} = 1000)

a um valor equivalente à eficácia da busca centralizada.

Tabela 5.4. Percentagem de variação da revocação relativa (da MAP entre parênteses) devido ao fator r (replicação por similaridade) e ao fator $kirsch$ (estratégia mesclagem kirsch) e interação entre os dois fatores.

	Percentagem Da Variação	
Fator	**TREC**	**WBR**
r	87,4% (92,5%)	40,0% (34,0%)
$kirsch$	6,3% (5,4%)	34,0% (47,4%)
$r * kirsch$	6,3% (2,0%)	25,9% (18,6%)

5.5 Considerações Finais

Neste Capítulo mensuramos a degradação na eficácia da busca P2P causada pela dinamicidade e a topicidade dos pares. Através de uma análise de variância (Tabela 5.3), verificamos que para coleções de alta topicidade como a WBR, a dinamicidade dos pares (fator A) é responsável por mais 76% da variação da eficácia (revocação relativa e MAP) da máquina de busca P2P, enquanto a topicidade (fator *conhecimento*) explica de 14% a 20% desta variação. Em coleções de topicidade mais baixa como a $TREC$, a variação eficácia da busca P2P pode ser atribuída principalmente à dinamicidade: mais de 94% da variação da eficácia (revocação relativa e MAP) pode ser atribuída ao *churn* (fator A).

Como o intuito de reduzir o impacto da dinamicidade dos pares, implantamos e avaliamos o mecanismo de replicação por similaridade na busca. Nossos resultados apontam que este mecanismo, por si só, pode elevar a eficácia da máquina da busca P2P a até 99% do equivalente centralizado em cenários que não há topicidade (GK). No

entanto, em cenários que há uma alta topicidade (LK, coleção WBR), a degradação da eficácia da máquina de busca P2P chega a até 74,6%, mesmo em cenários com uma alta taxa de replicação. Considerando o mesmo cenário para a coleção $TREC$, a degradação da eficácia ainda ocorre, porém em menor escala devido à baixa topicidade da coleção (degradação de até 34,6% da revocação relativa).

Por fim, avaliamos o mecanismo de *mesclagem Kirsch*, cujo o objetivo é reduzir os efeitos da topicidade. Nossos resultados indicam que este mecanismo, quando empregado em conjunto com a replicação por similaridade, pode elevar a eficácia da busca P2P por conteúdo a 95% do equivalente centralizado, mesmo em cenários de alta topicidade e dinamicidade dos pares (LK, *col*=WBR, A=25%). Apesar disso,é importante notar que estes primeiros resultados não levam em conta características de redes par-a-par, tais como a heterogeneidade e limitação dos pares quanto a banda e espaço em disco, e por este motivo, deve ser entendida como uma análise do **potencial** da *mesclagem Kirsch* e da *replicação por similaridade* frente ao problema da topicidade e dinamicidade dos pares. Em outras palavras, estes resultados apontam a **possibilidade** da criação de uma máquina de busca P2P por conteúdo eficaz mesmo em cenários que os pares possuem uma alta dinamicidade e topicidade.

Capítulo 6

Avaliando a Eficácia e a Eficiência da Máquina de Busca P2P

6.1 Descrição

No Capítulo 5 realizamos uma análise do potencial de eficácia da máquina de busca P2P frente a topicidade e dinamicidade dos pares. Apesar de encorajadores, os resultados supracitados possuem uma contribuição limitada a um valor potencial, pois desconsideram diversas características que podem ter grande impacto na eficácia, eficiência ou até mesmo viabilidade prática de máquina de busca P2P por conteúdo, tais como a heterogeneidade e limitação dos pares quanto a banda, falhas de comunicação, atrasos de mensagens e erros na localização e seleção de pares na rede.

Neste capítulo, avaliaremos a eficácia (qualidade), eficiência (utilização de recursos) e o compromisso existente entre estes aspectos da busca P2P por conteúdo em um cenário mais realista que o capítulo anterior. Nosso objetivo é mensurar o desempenho do da replicação por similaridade e mesclagem kirsch frente dificuldades práticas, como por exemplo, a a heterogeneidade e limitação dos pares quanto a banda. Mensuramos a eficácia da máquina de busca P2P por sua revocação relativa em relação a uma máquina de busca centralizada, e mensuramos a eficiência contabilizando o uso de banda médio, por par.

Para tanto, avaliamos a máquina de busca, utilizando o *modelo detalhado* de simulação de máquina de busca P2P apresentado na Seção 4.2 em diversos cenários. Em todos os cenários avaliados, cada par da máquina é responsável por um subconjunto de documentos da coleção TREC ou WBR-TOP1000. A coleção WBR-TOP1000 (descrita em mais detalhes na Seção 4.5) é um subconjunto da WBR com os 1000 *hosts* de maior

número de documentos, o que corresponde a 35% do número de documentos da coleção original. Esta coleção foi utilizada devido a inviabilidade reproduzir um cenário com o conjunto total de *hosts* (que corresponderiam a 110912 pares) utilizando o *modelo detalhado* de simulação de máquina de busca P2P. Assim como no caso da WBR, cada *host* da WBR-TOP1000 é associado a um par da rede, e as páginas Web por ele hospedadas são associadas como documentos do par correspondente. No caso da TREC os documentos são distribuídos de maneira uniforme entre os pares, uma vez que não há caracterização da distribuição de documentos para este cenário. No entanto, vale lembrar que a distribuição de documentos uniforme é pessimista quanto ao Seletor de Pares (descrito na Seção 3.4.1), dado quando os pares são semelhantes entre si é mais difícil distinguir quais os melhores pares para atender uma consulta.

Assim como na avaliação do potencial de eficácia de busca P2P por conteúdo do Capítulo 5, avaliamos o impacto de diferentes níveis de dinamicidade dos pares, utilizando valores de A iguais a 25% e 75%, apresentando sempre um foco maior aos cenários em que os pares possuem uma maior indisponibilidade ($A = 25\%$), no qual os efeitos da dinamicidade e topicidade sobre a eficácia da busca são mais graves. A distribuição de banda dos pares foi feita com base na distribuição observada em *hosts* dos EUA, reportada em [3]. Consideramos uma banda mínima de 256kbps, já que *hosts* com menor capacidade, em especial aqueles com conexões *dial-up* (56,6kbps), muitas vezes não possuem a banda necessária para contribuir na manutenção do diretório distribuído. De qualquer forma, estes *hosts* compõem uma parte pequena de distribuição de *hosts* observada (5,8%). Além disso, devido à sua baixa capacidade, estes *hosts* tendem a se comportar como clientes da máquina 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 (isto é, não mensuramos a escalabilidade da máquina de busca) 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 AOL[1] [5] encontramos que o tempo médio entre consultas de um mesmo usuário em um mesmo dia é de $3190,1s$. Logo, estes valores representam a máquina de busca sob uma carga proporcionalmente alta para seu número de usuá-

[1] Nossa utilização dos *logs* de consulta está dentro dos termos de uso exigidos pela AOL.

rios. Para a TREC, as consultas são escolhidas uniformemente entre cada uma das 50 consultas, dado que a coleção não fornece qualquer distribuição de popularidade. No caso da WBR-TOP1000, utilizamos a distribuição de popularidade fornecida pela coleção para selecionar qual consulta será submetida, o que nos fornece um modelo bem mais realista de carga para a máquina de busca. Em contrapartida, como não há julgamento de relevância para as 1000 consultas mais populares da WBR-TOP1000, não faz sentido utilizar a métrica MAP para mensurar a eficácia da máquina de busca P2P, e portanto, a avaliação de eficácia ficará restrita à revocação relativa.

Os valores de limiar para publicação no diretório distribuído ($DF_{threshold}$=10) e o parâmetro de ajuste do Seletor de Pares (α=0,4) foram fixados tomando como base trabalhos anteriores [11]. Fixamos ainda o tempo de vida dos *posts* no diretório distribuído ($ttl = 400s$) e focamos em Chord como infraestrutura de rede P2P.

Diferentemente do resultados no Capítulo 5, não há distinção dos cenários entre dois níveis de conhecimento (GK e LK), já que no *modelo detalhado* todas as etapas da máquina de busca são simuladas. Isto quer dizer que em todos os cenários cada par utiliza apenas as informações disponíveis em sua coleção local de documentos ou no diretório distribuído (vide Seção 3.4.2). Portanto, os problemas dinamicidade dos pares e da topicidade ocorrem simultaneamente em todos os cenários, apesar deste último apresentar-se de maneira mais contundente apenas no cenário em que a coleção WBR-TOP1000 é utilizada, devido à distribuição não uniforme de documentos entre os pares.

É importante notar que, ao utilizar o *modelo detalhado* de simulação, avaliamos a eficácia e a eficiência da máquina de busca em duas etapas: durante a etapa de seleção de pares (vide Seção 3.4.1), na qual um subconjunto dos pares é selecionado para processar a consulta, e após o processamento federado da consulta (vide Seção 3.4.3), quando as respostas dos pares são mescladas em uma única lista de documentos e apresentada ao usuário. Esta avaliação em duas etapas é importante pois permite capturar com maior precisão a fonte de *degradação da qualidade* da busca P2P por conteúdo, que pode ocorrer devido a má qualidade da *seleção de pares* (devido à dinamicidade, por exemplo) ou devido a um *processamento federado* de consultas incorreto (devido à topicidade, por exemplo).

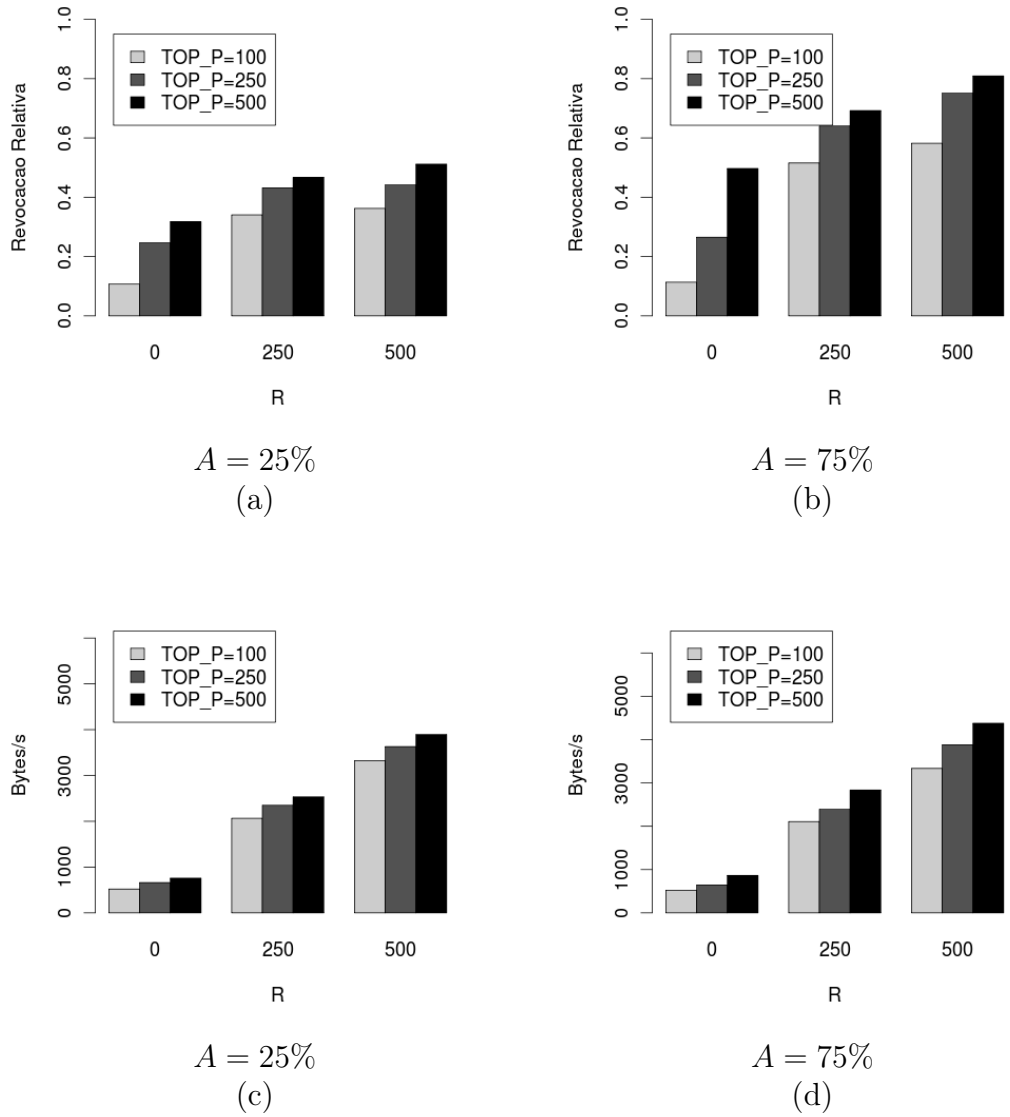
Os resultados reportados neste capítulo são resultados de no mínimo 4 experimentos com coeficiente de variação inferior a 4%.

6.2 Mensurando a Eficácia e a Eficiência da Seleção de Pares em Cenários de Baixa Topicidade (TREC)

Nesta seção, avaliamos a eficácia e a eficiência do Seletor de Pares CORI em nosso modelo de máquina de busca para cenários de baixa topicidade ($col = TREC$), variando apenas os parâmetros que afetam a eficácia e a eficiência da seleção de pares, tais como 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 não há limitação de espaço em disco de cada par (tamanho da cache C=∞).

As Figuras 6.1(a-b) mostram os valores da revocação relativa média da seleção de pares, ao final da simulação, para a coleção $TREC$ 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 menores valores de 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 muito maior na qualidade da seleção de pares que o número de pares selecionados, ou seja, o principal fator de degradação da eficácia da seleção da seleção de pares é a indisponibilidade dos documentos na rede. Incrementar o número de pares selecionados top_p de 100 para 250 melhora e eficácia da seleção de pares em 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 [21] 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. A Tabela 6.1 percentagem de variação da revocação devido aos fatores r e top_p.

Já as Figuras 6.1(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 o consumo de banda médio é 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



$A = 25\%$ (a) $A = 75\%$ (b)

$A = 25\%$ (c) $A = 75\%$ (d)

Figura 6.1. Revocação relativa (a-b) e consumo de banda (c-d) da seleção de pares para a coleção TREC, considerando diferentes níveis de disponibilidade (A), número de pares selecionados (top_p) e número de documentos replicados (r).

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 fixado em 25% aponta que o fator r é responsável por 98% variação do consumo de banda. A Tabela 6.1 apresenta a percentagem de variação do consumo de banda

devido aos fatores r e top_p.

Tabela 6.1. Percentagem de variação da revocação relativa e do consumo de banda da seleção de pares devido ao fator r (replicação por similaridade) e ao fator top_p (número de pares selecionados) e residual. ($A = 0,25$, $col = TREC$, $C{=}\infty$ e níveis de $top_p{=}[100, 250]$ e $r{=}[0, 100]$)

	Percentagem Da Variação	
Fator	**Revocação Relativa**	**Consumo de Banda**
r	76,8%	97,6%
top_p	21,9%	1,6%
Residual	1,2%	0,7%

Portanto, os resultados de eficácia e eficiência (Figuras 6.1 a-d e Tabela 6.1) para coleção TREC apontam que a replicação por similaridade é responsável por uma grande melhoria na eficácia, no entanto, seu impacto na eficiência da máquina busca (consumo de banda dos pares) é ainda maior. Em outras palavras, a replicação por similaridade é um mecanismo eficaz para melhorar a qualidade da máquina de busca P2P, mas não é um mecanismo eficiente. Aumentar o número de pares contatados (maiores valores de top_p) apresenta um melhor compromisso que maiores valores de r.

É importante notar, também, que o ganho em replicar mais de 250 documentos é bem menos significativo. Tomando como base os cenários em que $top_p{=}100$ e $A = 0,25$, observamos nas Figura 6.1(a) que o ganho em aumentar r de 0 para 250 (222%) é bem mais significativo que aumentar r de 250 para 500 (6%). Isso pode se explicado pela própria definição da mecanismo de replicação por similaridade: dado que os documentos mais similares à uma consulta são relevantes (vide a definição da revocação relativa na Seção 4.6), pode-se concluir que os documentos os primeiros 250 documentos retornados pela máquina de busca P2P têm maior probabilidade de serem relevantes que os 250 documentos seguintes (já que são mais similares à consulta). Logo, valores de r cada vez maiores levam a incrementos cada vez menores na eficácia.

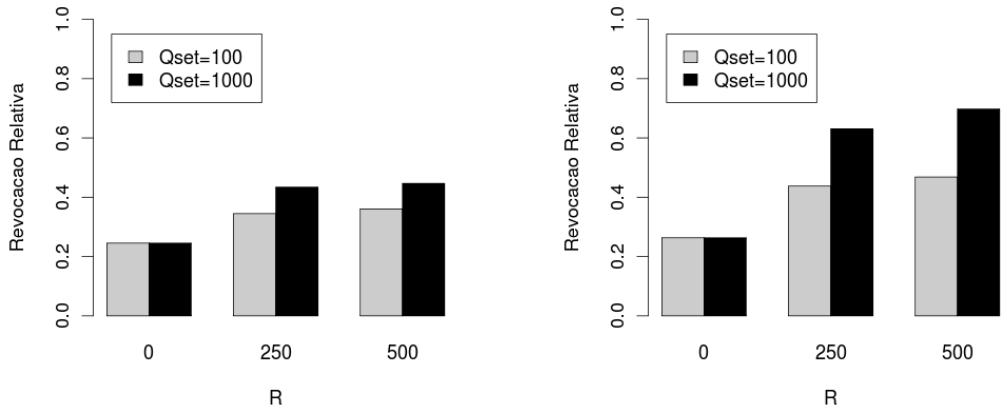
Outro ponto importante é que a eficácia da replicação por similaridade aqui apresentada está muito abaixo do potencial de eficácia apresentado na Seção 5.3 ($col = TREC$ e $A = 0,25$). No Capítulo anterior, mostramos que a replicação por similaridade, por si só, poderia elevar a eficácia da busca P2P por conteúdo para a coleção TREC a um valor potencial de 94% da eficácia da busca centralizada, no entanto, utilizando a replicação por similaridade em um cenário mais realista obtivemos a eficácia da busca P2P equivalente à apenas 54% do equivalente centralizado, já durante a seleção de pares.

Esta diferença entre a eficácia obtida e a eficácia potencial pode ser explicada analisando cenário utilizado (coleção de documentos TREC) e observando as diferenças entre o *modelo detalhado* de simulação e o *modelo simplificado*: no *modelo simplificado* (1) não há falhas de comunicação ou limitação de banda entres os pares, e (2) na etapa de seleção de pares, todos os pares que possuem pelo menos um termo da consulta são selecionados (mais detalhes no Capítulo 4). Como dito anteriormente, a distribuição de documentos para a coleção TREC é uniforme, a mais desafiadora para o seletor de pares, já que os documentos relevantes estão uniformemente distribuídos entre todos os pares da rede. Portanto, ao escolher um subconjunto de pares, o Seletor de Pares está dispensando vários pares que possuem documentos relevantes à consulta. Isso não ocorre na análise de potencial da máquina de busca P2P apresentada no Seção 5.3, dado que o *modelo simplificado* seleciona todos pares que possuem ao menos um termo da consulta. Portanto, para o Seletor de Pares atingir o potencial de eficácia encontrado no Capítulo anterior, é necessário uma redistribuição dos documentos entre os pares. Embora o mecanismo de replicação por similaridade replique os documentos mais similares às consultas entre os pares, em nossos experimentos essa redistribuição de documentos não foi suficiente para atingir todo o potencial de eficácia encontrado na Seção 5.3 (apesar de incrementar a eficácia da seleção de pares em até 222%).

6.3 Mensurando a Eficácia e a Eficiência do Processamento Federado de Consultas em Cenários de Baixa Topicidade (TREC)

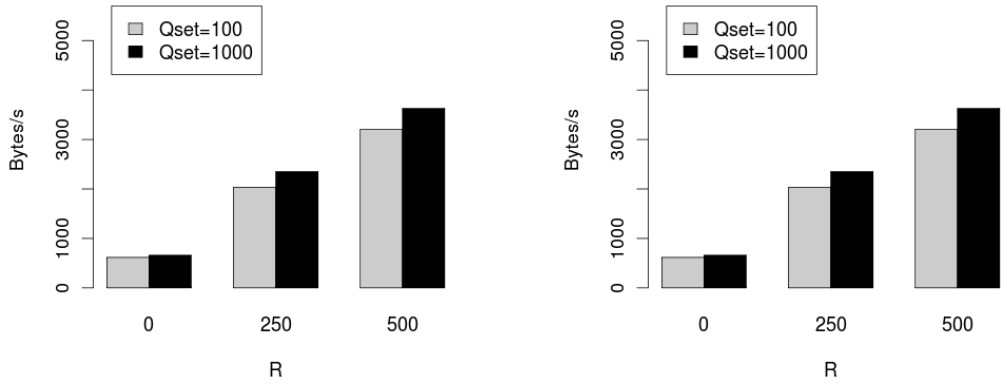
Avaliamos agora a eficiência e eficácia da etapa de processamento federado de consultas em cenários que há uma baixa topicidade (TREC). Para tanto, fixamos o número de pares selecionados ($top_p = 250$), e variamos os parâmetros que afetam diretamente o processamento federado de consultas, como o tamanho 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, inicialmente, que o espaço em disco de cada par é ilimitado ($C{=}\infty$).

As Figuras 6.2(a-b) mostram a revocação relativa média para valores de $r = [0, 250, 500]$, $|Q_{set}|{=}[100, 1000]$ e $A{=}[0,25, 0,75]$ para a coleção TREC. 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



$A = 25\%$
(a)

$A = 75\%$
(b)



$A = 25\%$
(c)

$A = 75\%$
(d)

Figura 6.2. Revocação relativa (a-b) e consumo de banda (c-d) do processamento de consultas para a coleção TREC, considerando diferentes níveis de disponibilidade (A), tamanho de respostas ($|Q_{set}|$) e número de documentos replicados (r).

cada consulta (r) tem impacto maior na eficácia que o tamanho da resposta de cada par ($|Q_{set}|$). Fixado o valor de A=0,25, realizamos uma análise de variância com níveis do fator $|Q_{set}|$=[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, mais em específico no cenário com $r = 0$, incrementar o tamanho das respostas dos pares não aumenta, ou aumenta desprezivelmente, 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. Como cada consulta possui 1000 documentos relevantes, é de se esperar que cada par possua em média apenas 1 documento relevante ($n = 1000$). 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 mais documentos relevantes. Nestes cenários, o tamanho da resposta passa a ter um efeito maior na qualidade da busca. De fato, realizamos uma análise de variância apresentada na Tabela 6.2, e observamos que 8% da variação da eficácia pode ser explicada pela interação entre a replicação (fator r) e o o tamanho das respostas (fator $|Q_{set}|$). Nesta mesma tabela, observamos que 83% da variação da eficácia é devido apenas à replicação por similaridade.

As Figuras 6.2(c-d) apresentam a banda média utilizada (em bytes/s) por cada par para os 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 de 0 para 250 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 6.2 apresenta a percentagem de variação do consumo de banda devido aos fatores $|Q_{set}|$ e r. Observe que a replicação por similaridade, por si só, é responsável por 97% da variação do consumo de banda. Novamente, apesar da replicação por similaridade apresentar uma grande melhoria na eficácia, seu o impacto na eficiência da máquina busca (banda dos pares) é ainda maior, e por este motivo, apresenta um compromisso pior que aumentar o valor do $|Q_{set}|$ (exceto no caso de $r = 0$).

Por fim, é interessante notar também que a eficácia do processamento federado de consultas observada quando utilizamos o *modelo detalhado* de simulação está muito abaixo do potencial de eficácia da máquina de busca P2P apresentados na Seção 5.3 ($col = TREC$ e $A = 0,25$). Lembrando que as principais diferenças entre o *modelo*

Tabela 6.2. Percentagem de variação da revocação relativa e do consumo de banda do processamento da consulta devido ao fator r (replicação por similaridade) e ao fator $|Q_{set}|$ (tamanho das respostas dos pares em número de documentos), interação entre os fatores e residual. ($A = 0,25$, $col = TREC$, $C{=}\infty$, $top_p = 250$ e níveis de $|Q_{set}|{=}[100, 1000]$ e $r{=}[0, 100]$)

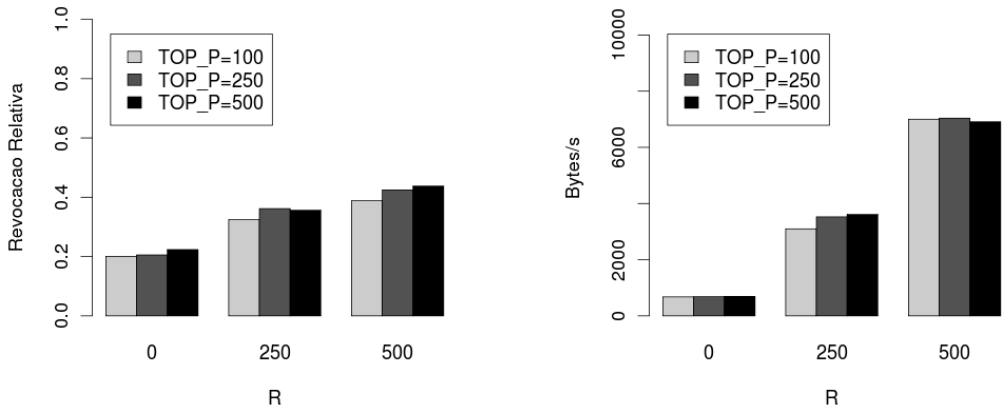
	Percentagem Da Variação	
Fator	**Revocação Relativa**	**Consumo de Banda**
r	83,4%	96,8%
$\|Q_{set}\|$	8%	1,3%
$r * \|Q_{set}\|$	8%	0,7%
$Residual$	0,5%	0,7%

detalhado de simulação e o *modelo simplificado* são que no *model simplificado* (1) não há falhas de comunicação ou limitação de banda entres os pares, e (2) durante a etapa de seleção de pares, todos os pares que possuem pelo menos um termo da consulta são selecionados; podemos concluir que a discrepância da eficácia apresentada entre os modelos pode ser explicada ou por falhas de comunicação ou pela diferença no processo de seleção de pares. Comparando a eficácia obtida pela seleção de pares com a eficácia obtida pelo processamento federado de consultas (em cenários equivalentes), vemos que a degradação da eficácia nesta última etapa é de no máximo 2%, o que indica que a eficácia da máquina de busca P2P é degradada já durante a seleção dos pares.

6.4 Mensurando a Eficácia e a Eficiência da Seleção de Pares em Cenários de Alta Topicidade (WBR-TOP1000)

Nesta seção, avaliamos a eficácia e a eficiência do Seletor de Pares CORI do nosso modelo de máquina de busca em um cenário que os pares possuem coleções de alta topicidade (col =WBR-TOP1000), variando apenas os parâmetros que afetam a eficácia e a eficiência da seleção de pares, como o número de pares selecionados a cada consulta (top_p) e o número de documentos replicados após o processamento de cada consulta (r). Apesar da presença da topicidade, não avaliaremos o impacto da mesclagem Kirsch na Seleção de Pares, dado que não há interação direta entre os dois mecanismo - a mesclagem Kirsch é executada somente após o processamento da consulta, o que ocorre muito após a seleção dos pares, e além disso, a sua função é apenas a reordenar das respostas fornecidas por todos os pares. Inicialmente, consideramos apenas um

cenário otimista em que não há limitação de espaço em disco de cada par (tamanho da cache $C{=}\infty$) e concentramos nossa avaliação em cenários com baixa disponibilidade (disponibilidade média (A) de 25%), já que estes apresentam uma maior degradação da eficácia da busca P2P.



(a) (b)

Figura 6.3. Revocação relativa (a) e consumo de banda (b) da seleção de pares para a coleção WBR-TOP1000 e $A = 0,25$, agrupados por número de pares selecionados (top_p) e número de documentos replicados (r).

A Figura 6.3 (a) mostra os valores da revocação relativa média da seleção de pares ao final da simulação, para os mesmos cenários avaliados na Seção 6.2 (com $A = 0,25$) mas utilizando desta vez a coleção de documentos WBR-TOP1000, o que nos leva a resultados substancialmente diferentes. Como podemos ver, nos cenários em que não há replicação ($r = 0$), aumentar o número de pares selecionados (maior valores de top_p) não fornece ganhos significativos na eficácia (no máximo 10%), principalmente se comparado a aumentar a replicação r de 0 para 250 (um ganho de 62% a 82%) - resultado oposto quando comparado ao mesmo cenário para a coleção TREC. Isso ocorre devido à alta topicidade da distribuição de documentos da coleção WBR-TOP1000, que faz com que os documentos relevantes a uma consulta estejam localizados em poucos pares especializados em um tópico, e por este motivo, muitas é vezes inútil selecionar um número maior de pares. De fato, realizando uma análise de variância para $A = 0,25$ com níveis de $r = [0, 250]$ e níveis de $top_p = [100, 250]$, verificamos que a replicação (fator r) é responsável por 95% da variação da eficácia, enquanto o número de pares selecionados é responsável por apenas 3% da variação.

A Figura 6.3(b) apresentam o valor médio do consumo de banda (em bytes/s) por par nos mesmos cenários. Note que apesar dos resultados de eficácia da máquina

de busca nos cenários para a coleção *WBR-TOP1000* divergirem quando comparado ao mesmo cenário para $TREC$, os resultados para eficiências são qualitativamente semelhantes. Nossos resultado novamente apontam que a replicação por similaridade tem um impacto muito maior na banda consumida que o número de pares selecionados. Assim como no caso da $TREC$, a análise de variância para $A = 0,25$ e de $r = [0, 250]$ e níveis de $top_p = [100, 250]$ aponta que o fator r por si só é responsável por 98% variação do consumo de banda.

Tabela 6.3. Percentagem de variação da revocação relativa e do consumo de banda do processamento da consulta devido ao fator r (replicação por similaridade), ao fator top_p (número de pares selecionados) e residual. ($A = 0,25$, *col* =WBR-TOP1000, C=∞ e níveis de top_p=[100, 250] e r=[0, 100])

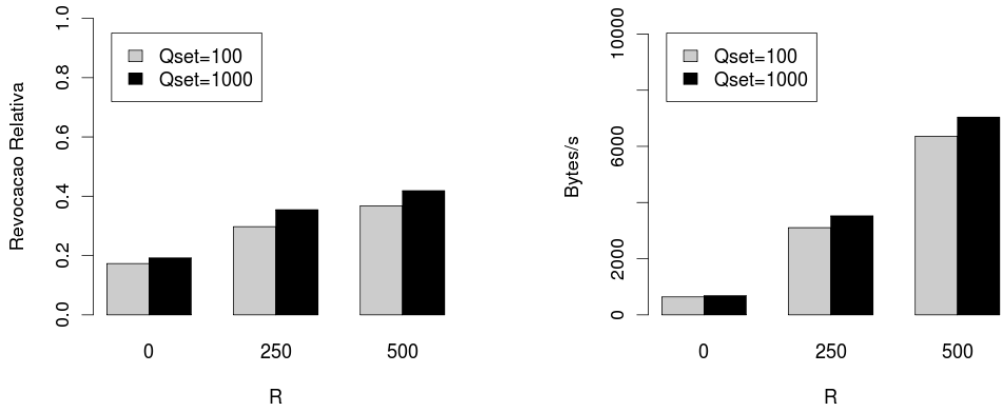
	Percentagem Da Variação	
Fator	**Revocação Relativa**	**Consumo de Banda**
r	95,6%	98,6%
top_p	2,4%	0,8%
Residual	2,1%	0,6%

As Tabela 6.3 apresenta as percentagens da variação da revocação e consumo de banda devido a replicação por similaridade (fator r) e número de pares selecionados (fator top_p). Novamente, a replicação por similaridade apresenta um baixo compromisso entre eficácia e eficiência, representando uma variação no consumo da banda proporcionalmente superior a sua melhoria na eficácia. No entanto, observamos que neste cenário a replicação é a única alternativa para melhorar a eficácia na busca, visto que o fator fator top_p tem baixíssimo impacto na eficácia (2,5%), em outras palavras, selecionar mais pares não melhora significativamente a eficácia da busca no cenário da coleção WBR-TOP1000, onde os pares possuem coleções com alta topicidade e distribuição não uniforme de documentos.

6.5 Mensurando a Eficácia e a Eficiência do Processamento Federado de Consultas em Cenários de Alta Topicidade (WBR-TOP1000)

Agora avaliamos a eficiência e eficácia do processamento federado de consultas em cenários que há uma alta topicidade dos pares (*col* =WBR-TOP1000). Para tanto, fixamos o número de pares selecionados ($top_p = 250$), e variamos apenas os parâmetros que afetam diretamente o processamento federado de consultas, como o tamanho da

resposta de cada par ($|Q_{set}|$) e o número de documentos replicados (r) . Como há topicidade, apresentaremos também os resultados da utilização da mesclagem Kirsch. Novamente, consideramos inicialmente um cenário otimista em que o espaço em disco de cada par é ilimitado (C=∞) e restringimos a análise ao cenário em que os pares possuem baixa disponibilidade ($A = 0,25$), no qual a degradação da eficácia da busca P2P é maior.



(a) (b)

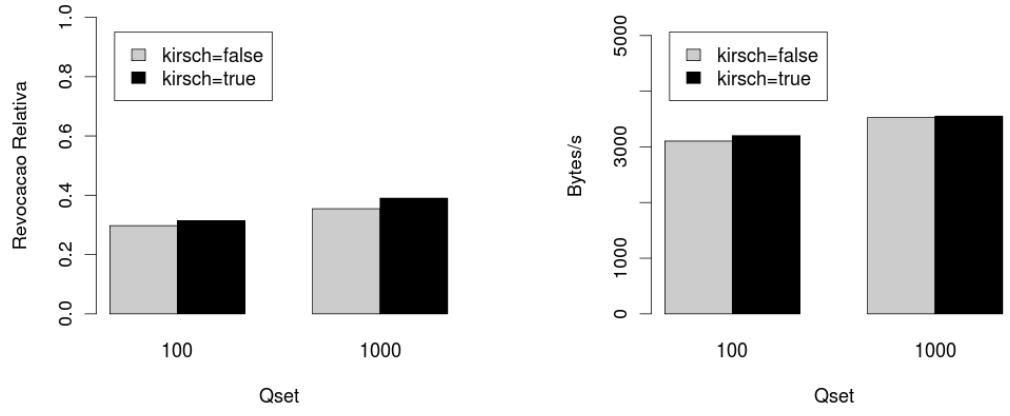
Figura 6.4. Revocação relativa (a) e consumo de banda (b) do processamento de consultas para a coleção WBR-TOP1000 e $A = 0,25$, agrupados por tamanho de respostas ($|Q_{set}|$) e número de documentos replicados (r).

As Figura 6.4 (a) mostra a revocação relativa média em cenários com valores de $r = [0, 250, 500]$ (número de documentos replicados por consulta), $|Q_{set}|$=$[100, 1000]$ (número de documentos por resposta de cada par), $A = 25\%$ e coleção de documentos WBR-TOP1000. Como nos cenários anteriores, podemos observar que a replicação por similaridade melhora em muito a eficácia da busca P2P por conteúdo. Por exemplo, aumentar o número de documentos replicados a cada consulta r de 0 para 250 pode melhorar a eficácia da máquina de busca em até 88%. Diferentemente dos resultados para a TREC, aumentar o tamanho das respostas dos pares pode beneficiar a eficácia da busca, mesmo quando não há replicação. Isto ocorre devido à topicidade da coleção e à distribuição não uniforme de documentos entre os pares, que faz com que alguns pares possuam muitos documentos relevantes para uma mesma consulta. Note que aumentar o tamanho da resposta dos pares ($|Q_{set}|$) de 100 para 1000 documentos incrementa a eficácia da máquina de busca em pelo menos 10% em qualquer cenário.

A Figura 6.4 (b) apresenta o consumo de banda média (em bytes/s) para os mesmos cenários acima apresentados. Novamente, a replicação por similaridade (r)

possui um impacto muito maior no consumo de banda de cada par que o tamanho das respostas dos pares ($|Q_{set}|$). A variação do $|Q_{set}|$ de 100 para 1000 gera um aumento no consumo de banda de apenas 6%, enquanto o aumento do número de documentos replicados r de 0 para 250 gera um aumento do consumo de banda de até 420%. Assim como nos resultados anteriores, a replicação por similaridade apresenta um impacto muito maior na eficiência da busca P2P que na eficácia, e portanto apresenta um pior compromisso entre eficácia e eficiência. Note também que o impacto da replicação por similaridade na eficiência é superior para a coleção WBR-TOP1000 quando comparado ao mesmo cenário da TREC. Isso se deve ao fato do tamanho médio dos documentos da WBR-TOP1000 (10,988KB) ser superior ao tamanho médio dos documentos da TREC (2,902KB), o que implica em um maior custo na replicação.

Vimos no Capítulo 5, mais em específico nos resultados apresentados nesta Seção 5.3, que a replicação por similaridade pode reduzir a degradação da eficácia da máquina de busca P2P devido à dinamicidade dos pares, no entanto, não é capaz de superar o problema da topicidade. Como a coleção de documentos WBR-TOP1000 possui uma alta topicidade, é mister avaliarmos o impacto da mesclagem Kirsch na eficácia e eficiência do processamento federado de consultas para esta coleção. Com este objetivo em mente, mensuramos a eficácia e a eficiência da busca P2P em cenários com o $r = 250$ fixo (já avaliamos o impacto da eficácia e eficiência deste mecanismo), variando entre cenários que utilizam ou não o mecanismo de mesclagem ($kirsch = [true, false]$) e com diferentes tamanho de resposta dos pares ($|Q_{set}|$).



(a) (b)

Figura 6.5. Revocação relativa (a) e consumo de banda (b) do processamento de consultas para a coleção WBR-TOP1000 e $A = 0,25$, agrupados por número de documentos replicados (r) e mecanismo de mesclagem ($kirsch$).

A Figura 6.5 (a) apresentam a revocação relativa ao final da simulação nos cenários supracitados. Como podemos ver, a mesclagem kirsch incrementa a eficácia da busca P2P por conteúdo de 5% a 10%, quando comparado a um cenário equivalente utilizando a mesclagem simples, enquanto aumentar o tamanho da resposta ($|Q_{set}|$) dos pares incrementa a eficácia de 20% a 25%. É importante notar que a mesclagem Kirsch funciona bem melhor no cenário com uma maior resposta dos pares, e que incrementar o tamanho das respostas dos pares ($|Q_{set}| = 1000$) e aplicar a mesclagem Kirsch em conjunto fornece um incremento na eficácia superior a soma dos incrementos individuais destas modificações. Isto se deve ao fato que a principal função da mesclage Kirsch é recomputar a similaridade dos documentos e reordena-los. Como a métrica revocação relativa mede a percentagem de documentos relevantes que são apresentados na resposta da máquina de busca (e não a ordem que são apresentados), a reordenação de documentos da mesclagem Kirsch tem impacto apenas quando o número total de documentos respondido pelos pares é muito maior que o número de documentos apresentados na resposta final da máquina de busca, já que neste caso a reordenação dos documentos define efetivamente quais os documentos que serão apresentados e quais serão ignorados. Por este motivo, a mesclagem Kisrch tem maior impacto em cenários com um maior valor de $|Q_{set}|$.

A Figura 6.5 (b) apresenta o consumo em banda médio de cada par (em bytes/s) para o mesmo cenário. Os resultados apontam que a mesclagem Kirsch apresenta um aumento no consumo banda médio desprezível (de 1% a 3%). Proporcionalmente, o impacto da mesclagem Kirsch na variação da eficácia da busca é superior ao seu impacto no consumo de banda. No Capítulo 3, demonstramos que a mesclagem Kirsch poderia duplicar o tamanho das respostas dos pares à uma consulta, no entanto, nossos resultados indicaram um impacto desprezível no consumo de banda médio dos pares, o que a princípio parece contraditório. De fato, o tamanho das respostas dos pares é aumentado devido a mesclagem Kirsch, no entanto as estatísticas das repostas dos pares às consultas constituem um porção menor dos dados que são trafegados entre os pares durante o funcionamento da máquina busca P2P, sendo a replicação dos documentos e a manutenção das estatísticas no diretório distribuído os principais responsáveis pelo consumo de banda total.

A Tabela 6.4 apresenta a percentagem da variação da revocação e consumo de banda médio, respectivamente, calculadas através da análise de variância dos resultados com níveis de $|Q_{set}|$=$[100, 1000]$ e $kirsch$=$[true, false]$, e fixados parâmetros C=∞, $top_p = 250$ e $A = 0,25$ para a coleção de documentos WBRTOP1000. Como podemos ver, a mesclagem Kirsch tem um impacto na variação da banda consumida menor que o fator $|Q_{set}|$ (número de documentos na resposta de cada par) e um impacto maior

Tabela 6.4. Percentagem de variação da revocação relativa e do consumo de banda do processamento da consulta devido ao fator $kirsch$ (mecanismo de mesclagem) e ao fator $|Q_{set}|$ (tamanho das respostas dos pares em número de documentos) e residual. ($A = 0,25$, col =WBRTOP1000, C=∞, $top_p = 250$ e níveis de $|Q_{set}|$=$[100, 1000]$ e $kirsch$=$[true, false]$)

	Percentagem Da Variação	
Fator	**Revocação Relativa**	**Consumo de Banda**
$\|Q_{set}\|$	97,1%	99,9%
$kirsch$	2,1%	0,1%
$Residual$	0,8%	0%

na variação da eficácia, apresentando portanto um melhor compromisso com ganhos modestos na eficácia da máquina de busca.

No entanto, é interessante notar que, assim como em todos nossos resultados anteriores, a eficácia obtida da máquina de busca no *modelo detalhado* de simulação é consideravelmente inferior à obtida utilizando o *modelo simplificado* (Capítulo 5). Novamente, o que percebemos nestes cenários é que a degradação da eficácia ocorre já durante a fase de seleção dos pares (vide resultados da Seção 6.4), o que limita severamente os benefícios da mesclagem Kirsch.

6.6 Mensurando a Eficiência em disco da Máquina de Busca P2P

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=$[0,25, 0,75]$).

Tabela 6.5. Eficácia da máquina de busca para diferentes tamanhos de cache.

	TREC		**WBR-TOP1000**	
C \ A	$0,25$	$0,75$	$0,25$	$0,75$
500	0,430	0,593	0,269	0,427
1000	0,436	0,629	0,309	0,452
∞	0,438	0,633	0,323	0,464

A Tabela 6.5 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 96% do mesmo cenário com C=∞. Considerando que o tamanho documento da médio do documento para a TREC (WBR-TOP1000) é de 2,902KB (10,988KB), estes resultados implicam que uma cache pequena, de apenas 1,38MB (5,24MB), é suficiente para atender o mecanismo de replicação por similaridade.

6.7 Considerações Finais

Neste Capítulo avaliamos a eficácia e a eficiência da máquina de busca P2P por conteúdo, considerando características práticas de sistemas P2P ignoradas em trabalhos anteriores [32, 38, 43, 42, 18, 16, 34, 6], tais como a a heterogeneidade e limitação dos pares quanto banda e espaço em disco, falhas de roteamento e troca de mensagens na rede sobreposta, comportamento dinâmico e topicidade da coleção de documentos. A avaliação deste Capítulo complementa de forma prática os resultados do Capitulo 5, no qual apresentamos o potencial de eficácia da máquina de busca P2P frente ao comportamento dinâmico dos pares e topicidade.

Mensuramos a eficácia e a eficiência da busca P2P por conteúdo em duas etapas do processamento da consulta na máquina de busca P2P: a seleção de pares e o processamento federado de consultas. Nesta avaliação variamos diversos fatores, como a disponibilidade média dos pares (A), número de documentos replicados (r), número de pares selecionados (top_p), tamanho das respostas de cada par ($|Q_{set}|$), o mecanismo de mesclagem utilizada ($kirsch = [true, false]$), a quantidade de espaço disponível em cada par para replicação (C) e a coleção de documentos utilizada (TREC ou WBR-TOP1000).

Através desta avaliação, pudemos observar que a replicação por similaridade possui grande impacto na eficácia da máquina de busca P2P, em alguns casos incrementando a eficácia da busca em até 166%. Entretanto, apesar de eficaz, o mecanismo de replicação por similaridade não é muito eficiente, já que seu impacto na eficiência da máquina de busca (consumo de banda) é proporcionalmente maior que seu benefício na eficácia. Apesar disso, o mecanismo de replicação por similaridade permanece uma ferramenta importante para máquinas de busca P2P em cenários que a eficácia for mais importantes que a eficiência. Em casos que a eficiência for mais importante que a eficácia, aumentar valores dos parâmetros $|Q_{set}|$ e top_p da máquina de busca fornece um melhor compromisso que o mecanismo de replicação.

Em cenários que há alta topicidade na coleção de documentos dos pares (*col*=WBR-TOP1000), o mecanismo de mesclagem Kirsch apresentou modestos ganhos de eficácia (de 5% a 10%) a um custo em banda proporcionalmente menor, o que indica que a mesclagem Kirsch é um mecanismo eficiente na máquina de busca P2P. Os ganhos obtidos na eficácia foram inferiores ao esperado, mas isso ocorreu por que a maior parte da degradação da eficácia, nesse cenário, decorreu da seleção de pares e não da topicidade.

No geral, apesar das melhorias que os mecanismo de replicação por similaridade e mesclagem Kirsch proporcionaram em nossos experimentos (uma melhoria de até 166%), podemos observar que, em todos os cenários em que os pares possuem uma baixa disponibilidade média ($A = 0,25$), a eficácia da busca P2P ficou consideravelmente abaixo do potencial de eficácia da máquina de busca P2P identificado no Capítulo 5. Identificamos que maior parte desta degradação ocorre já durante a seleção dos pares, o que aponta claramente que a melhor alternativa para incrementar a eficácia da máquina de busca P2P neste ponto é melhorar o funcionamento do Seletor de Pares ou a forma que os documentos estão distribuídos na rede.

Capítulo 7

Conclusões e Trabalhos Futuros

Neste trabalho, avaliamos quantitativamente a eficácia (qualidade) e a eficiência (utilização de recursos) de uma máquina de busca P2P, em cenários que os pares possuem comportamento altamente dinâmico e recursos limitados. Utilizamos duas coleções de testes reais (TREC-8 e WBR) e consideramos um modelo de rede sobreposta e física detalhado. Avaliamos uma estratégia de replicação para o contexto de busca P2P, uma estratégia de mesclagem de respostas dos pares, e o compromisso entre qualidade e utilização de recursos na escolha de parâmetros da máquina de busca P2P.

A avaliação é realizada em duas etapas: primeiramente quantificamos o potencial de eficácia da máquina de busca P2P utilizando um *modelo simplificado*, que nos permitiu avaliar a extensão do potencial dos mecanismos de *replicação por similaridade* e *mesclagem Kirsch* na melhoria da eficácia da busca P2P por conteúdo; em um segundo momento utilizamos um *modelo detalhado* de simulação, capaz de analisar a eficácia e a eficiência da máquina de busca P2P em cenários mais práticos, considerando características tais como a heterogeneidade e a limitação dos pares quanto banda e espaço em disco, falhas de roteamento e troca de mensagens na rede sobreposta, bem como o comportamento dinâmico dos pares e a topicidade da coleção de documentos.

A avaliação do potencial da máquina de busca P2P indicou que a replicação por similaridade pode elevar a eficácia da busca P2P a uma eficácia equivalente a 99% da eficácia da busca centralizada, em cenários em que não há topicidade. Em cenários em que a coleção de documentos possui uma alta topicidade, a replicação por similaridade por si só não é suficiente para manter um nível de eficácia semelhante ao obtido com a busca centralizada. Entretanto, quando emprega-se a replicação por similaridade e mesclagem Kirsch em conjunto, observamos que o potencial de eficácia da busca P2P pode chegar à 95% da eficácia da busca centralizada. Apesar de encorajadores, estes resultados indicam apenas o potencial da busca P2P por conteúdo, em um cenário em

que não há falhas de comunicação, limitação de banda ou erros durante a seleção de pares.

Em seguida, avaliamos a máquina de busca P2P em um contexto mais prático, utilizando o *modelo detalhado* de simulação. Nestes cenários consideramos o processo de seleção de pares, manutenção de estatísticas da máquina de busca no diretório distribuído, roteamento e troca de mensagens, atrasos de transmissões, falhas de comunicação, limitação de banda, entre outras características encontradas em sistemas P2P reais. Neste contexto mais prático, nossos resultados apontam que, em cenários em que não há topicidade, 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 pior compromisso entre eficácia e eficiência quando comparada com ajustes mais simples, como aumentar o número de pares contatados a cada consulta ou o tamanho da resposta de cada par. Ainda assim, em todos os cenários, a replicação por similaridade forneceu uma melhoria na eficácia superior a 40%, o que a indica como uma alternativa para cenários em que a eficácia da busca for mais importante que eficiência.

Em cenários em que a coleção de documentos possui uma alta topicidade, a replicação por similaridade foi utilizada em conjunto com a mesclagem Kirsch, o que forneceu uma incremento total na eficácia de 85% a 118%. A mesclagem Kirsch, por si só, forneceu um modesto ganho de 5% a 10%, mas demonstrou ser um mecanismo eficiente, provendo uma melhoria na eficácia da busca P2P proporcionalmente maior que seu impacto no consumo de banda (que foi de 3% no pior caso).

Por fim, verificamos que o mecanismo de replicação por similaridade mantém sua eficácia mesmo em cenários em que os pares possuem relativamente pouco espaço em disco dedicado para replicação (o equivalente a 500 documentos), para ambas as coleções de documentos avaliadas.

É importante notar que, apesar de todas as melhorias que os mecanismo de replicação por similaridade e mesclagem Kirsch proporcionaram em nossos experimentos (uma melhoria de de até 166%), podemos observar que em todos os cenários em que os pares possuem uma baixa disponibilidade média ($A = 0,25$), a eficácia da busca P2P ficou consideravelmente abaixo do potencial de eficácia da máquina de busca P2P identificado no Capítulo 5, e identificamos que maior parte desta degradação ocorre já durante a seleção dos pares. Este resultado aponta duas novas oportunidades de pesquisa futura que podem melhorar significativamente a eficácia da máquina de busca P2P por conteúdo: (1) melhorar o desempenho do Seletor de Pares e (2) melhorar a distribuição dos documentos na rede. É interessante notar que a degradação da eficácia

do Seletor de Pares pode ser consequência da indisponibilidade dos pares e de erros nas estatísticas armazenados no diretório distribuído (ambos causados pelo *churn*), e não por ineficácia do algoritmo CORI. Por este motivo, faz-se necessário um estudo mais detalhado do impacto do *churn* na etapa de seleção de pares.

Além das oportunidades de pesquisa supracitadas, há a possibilidade de expandir nosso modelo de máquina de busca P2P para incluir novos modelos de processamento de consulta (utilizando textos âncora, por exemplo), novos mecanismos de mesclagem de respostas e de replicação de documentos. Especialmente quanto ao mecanismo de replicação, é importante lembrar que a *replicação por similaridade* (analisada neste trabalho) aumenta a *topicidade* da coleção de documentos dos pares, já que concentra localmente todos os documentos relacionados à consulta. A criação e análise de um mecanismo de replicação que, ao invés de concentrar, distribua os documentos uniformemente entre os pares, pode agregar resultados relevantes à avaliação aqui apresentada.

A avaliação de outras coleções de documentos e cargas de trabalho também podem adicionar valor aos resultados deste trabalho. Em especial, a distinção entre consultas transacionais, navegacionais e informacionais, bem como a utilização de diferentes cargas de trabalho com conjuntos de consultas populares (para as quais a topicidade pode ter maior efeito, por possuírem termos comuns) e consultas aleatórias (para as quais a topicidade pode possuir menor efeito, devido a maior raridade dos termos) podem refinar e ampliar nossos resultados.

Ainda outra possibilidade de expansão do tema de pesquisa seria avaliar o atual modelo em diferentes cenários. Uma avaliação com diferentes números de pares permitiria uma análise de escalabilidade da máquina de busca P2P, e a utilização de diferentes redes sobrepostas, tais como CAN e Pastry, permitiria um estudo comparativo sobre a eficiência das diferentes redes sobrepostas em um cenário prático.

Referências Bibliográficas

[1] (`http://www9.limewire.com/developer/gnutella_protocol_0.4.pdf` last acessed on November 2008). The Gnutella Protocol Specification v0.4.

[2] ACM (`http://portal.acm.org/dl.cfm` last acessed on January 2010). ACM Digital Library.

[3] Akamai (2009). The state of the internet. Technical report, Akamai.

[4] Alexander, T. & Kedem, G. (1996). Distributed predictive cache design for high performance memory system. In *Second International Symposium on High Performance Computer Architecture*, pp. 254--263.

[5] AOL (2006). 500k user session collection. Technical report, America Online.

[6] Atalla, F. (2008). Impacto docomportamento dinâmico dos pares na eficácia de máquinas de busca par-a-par. In *Master of Science Thesis*. UFMG, Minas Gerais, Brazil.

[7] Atalla, F.; Miranda, D.; Almeida, J.; Gonçalves, M. & Almeida, V. (2008). Analyzing the impact of churn and malicious behavior on the quality of peer-to-peer web search. In *SAC '08*.

[8] Baeza-Yates, R.; Ribeiro-Neto, B. et al. (1999). *Modern information retrieval*. Addison-Wesley Harlow, England.

[9] Baumgart, I.; Heep, B. & Krause, S. (2007). OverSim: A Flexible Overlay Network Simulation Framework. In *IEEE GIS '07*, pp. 79--84.

[10] Bender, M.; Michel, S.; Triantafillou, P.; Weikum, G. & Zimmer, C. (2005a). Minerva: collaborative p2p search. In *VLDB '05*, pp. 1263--1266.

[11] Bender, M.; Michel, S.; Weikum, G. & Zimmer, C. (2005b). The MINERVA project: Database selection in the context of P2P search. *Datenbanksysteme in Business, Technologie und Web*.

[BLOOM] BLOOM, B. Space/Time Trade-offs in Hash Coding with Allowable Errors.

[12] Calado, P. (1999). The WBR-99 Collection: Description of the WBR-99 Web collection data-structures and file formats. *LATIN, Universidade Federal de Minas Gerais, Brazil*.

[13] Callan, J. (2000). Distributed information retrieval. *Advances in information retrieval*.

[14] Callan, J.; Lu, Z. & Croft, W. (1995). Searching distributed collections with inference networks. In *ACM SIGIR '95*, pp. 21--28.

[15] Chawathe, Y.; Ratnasamy, S.; Breslau, L.; Lanham, N. & Shenker, S. (2003). Making gnutella-like P2P systems scalable. In *SIGCOMM '03*, pp. 407--418.

[16] Chen, H.; Jin, H.; Wang, J.; Chen, L.; Liu, Y. & Ni, L. (2008). Efficient multi-keyword search over p2p web.

[17] Compete.com (`http://siteanalytics.compete.com/geocities.com/` last acessed on July 2009). Site profile for geocities.com.

[18] Cuenca-Acuna, F.; Peery, C.; Martin, R. & Nguyen, T. (2003). PlanetP: Using Gossiping to Build Content Addressable Peer-to-Peer Information Sharing Communities. In *HPDC '03*.

[19] Dabek, F.; Zhao, B.; Druschel, P.; Kubiatowicz, J. & Stoica, I. (2003). Towards a common API for structured peer-to-peer overlays. *Lecture Notes in Computer Science*, pp. 33--44.

[20] Flajolet, P.; Martin, G. & national de recherche en informatique et en automatique (France, I. (1985). Probabilistic Counting Algorithms for Data Base Applications. *JCSS*, 31(2):182--209.

[21] Jain, R. (1991). *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. Wiley- Interscience, New York.

[22] Kirsch, S. (1997). Document retrieval over networks wherein ranking and relevance scores are computed at the client for multiple database documents. US Patent 5,659,732.

[23] Kishida, K. (2005). *Property of average precision and its generalization: An examination of evaluation indicator for information retrieval experiments*. National Institute of Informatics.

[Kurose] Kurose, J. *Computer Networking: A Top-Down Approach Featuring the Internet, 3/E*. Pearson Education India.

[24] Larkey, L.; Connell, M. & Callan, J. (2000). Collection selection and results merging with topically organized US patents and TREC data. In *CIKM '00*, pp. 282--289.

[25] Lee, D.; Chuang, H. & Seamons, K. (1997). Document ranking and the vector-space model. *Software, IEEE*, 14(2):67--75.

[26] Li, J.; Loo, B.; Hellerstein, J.; Kaashoek, M.; Karger, D. & Morris, R. (2003). On the Feasibility of Peer-to-Peer Web Indexing and Search. *LECTURE NOTES IN COMPUTER SCIENCE*.

[27] Liang, J.; Kumar, R. & Ross, K. (2005). The KaZaA Overlay: A Measurement Study. *Computer Networks Journal (Elsevier)*, 49(6).

[28] Luu, T.; Klemm, F.; Podnar, I.; Rajman, M. & Aberer, K. (2006). Alvis peers: a scalable full-text peer-to-peer retrieval engine. In *P2PIR '06*, pp. 41--48.

[29] Lv, Q.; Cao, P.; Cohen, E.; Li, K. & Shenker, S. (2002). Search and replication in unstructured peer-to-peer networks. In *ICS '02*, pp. 84--95. ACM New York, NY, USA.

[30] Maymounkov, P. & Mazieres, D. (2002). Kademlia: A peer-to-peer information system based on the XOR metric. In *IPTPS'02*, volume 258, p. 263.

[31] Menasce, D. & Almeida, V. (2001). *Capacity Planning for Web Services: metrics, models, and methods*. Prentice Hall PTR Upper Saddle River, NJ, USA.

[32] Michel, S.; Bender, M.; Ntarmos, N.; Triantafillou, P.; Weikum, G. & Zimmer, C. (2006). Discovering and exploiting keyword and attribute-value co-occurrences to improve P2P routing indices. In *CIKM '06*, pp. 172--181.

[33] Parreira, J.; Michel, S. & Bender, M. (2006). Size doesn't always matter: exploiting pageRank for query routing in distributed IR. In *Proceedings of the international workshop on Information retrieval in peer-to-peer networks*, pp. 25--32. ACM Press New York, NY, USA.

[34] Parreira, J. & Weikum, G. (2005). JXP: Global Authority Scores in a P2P Network. In *International Workshop on Web and Databases, Baltimore, USA*.

[35] Podnar, I.; Rajman, M.; Luu, T.; Klemm, F. & Aberer, K. (2007). Scalable Peer-to-Peer Web Retrieval with Highly Discriminative Keys. In *ICDE '07*.

[36] Ratnasamy, S.; Francis, P.; Handley, M.; Karp, R. & Schenker, S. (2001). A scalable content-addressable network. In *SIGCOMM '01*, volume 31, pp. 161--172.

[37] Rowstron, A. & Druschel, P. (2001). Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems. *LNCS 2001*, 2218:329--350.

[38] Skobeltsyn, G.; Luu, T.; Podnar Žarko, I.; Rajman, M. & Aberer, K. (2008). Query-driven indexing for scalable peer-to-peer text retrieval. *FGCS '08*.

[39] Skobeltsyn, G.; Luu, T.; Zarko, I.; Rajman, M. & Aberer, K. (2007). Web text retrieval with a P2P query-driven index. In *SIGIR '07*, pp. 679--686.

[40] Stoica, I.; Morris, R.; Karger, D.; Kaashoek, M. & Balakrishnan, H. (2001). Chord: A scalable peer-to-peer lookup service for internet applications. In *SIGCOMM '01*, pp. 149--160.

[41] Stutzbach, D. & Rejaie, R. (2006). Understanding churn in peer-to-peer networks. In *ACM SIGCOMM '06*, pp. 189--202. ACM New York, NY, USA.

[42] Suel, T.; Mathur, C.; Wu, J.; Zhang, J.; Delis, A.; Kharrazi, M.; Long, X. & Shanmugasundaram, K. (2003). Odissea: A peer-to-peer architecture for scalable web search and information retrieval. In *WebDB '03*, pp. 67--72.

[43] Tang, C.; Xu, Z. & Dwarkadas, S. (2003). Peer-to-peer information retrieval using self-organizing semantic overlay networks. In *SIGCOMM '03*, pp. 175--186.

[44] Voorhees, E. & Harman, D. (2000). Overview of the eighth text retrieval conference (TREC-8). *NIST SPECIAL PUBLICATION SP*, pp. 1--24.

[45] Wikipedia (`http://en.wikipedia.org/wiki/List_of_digital_library_projects` last acessed on May 2010). List of Digital Library Projects.

[46] Yahoo (`http://geocities.yahoo.com/` last acessed on November 2009). Sorry, GeoCities has closed. .