

Localização de Conteúdo em Redes Peer-to-Peer

André Detsch

¹ PIPCA - Programa de Pós-Graduação em Computação Aplicada
Centro de Ciências Exatas e Tecnológicas - UNISINOS

Resumo. *Com a rápida popularização de aplicações como Napster e Gnutella, redes peer-to-peer têm sido o foco de atenção de muitos projetos de pesquisa. Um dos desafios mais relevantes nesta área é a forma com que é realizada a busca por conteúdo dentro de uma rede destas dimensões e com esta dinamicidade. Neste artigo, são apresentadas as principais técnicas criadas com esta finalidade, bem como uma visão geral do processo e das métricas de avaliação de sistemas peer-to-peer.*

1. Introdução

A Internet, como inicialmente concebida no final dos anos 60, era um sistema peer-to-peer (P2P), ou seja, os nós que faziam parte da rede tinham funções similares entre si. Nos anos 90, com o avanço da Internet, esta forma de comunicação passou a dar lugar a uma estrutura principalmente cliente-servidor, onde os usuários finais passaram a agir quase que exclusivamente como consumidores de informação ([1]). Esta realidade passou a mudar quando aplicações P2P como Napster ([2]) e Gnutella ([3]) passaram a possibilitar que usuários comuns, mesmo desprovidos de uma conexão estável ou de um endereço IP válido, contribuíssem para a disseminação de dados através da Internet. De certa forma, algumas idéias da Internet original passaram a ganhar força, mas em um ambiente já bastante adaptado ao modelo vigente. Dentro dessa nova gama de aplicações, surgem diversas questões de pesquisa que precisam serem investigadas.

Um dos principais desafios para a franca utilização de recursos P2P é a localização de conteúdo dentro da rede. Enquanto no modelo cliente-servidor a informação pode ser encontrada diretamente em um servidor projetado para ficar no ar 24 horas por dia, em uma rede P2P os dados estão distribuídos em milhares de nós que, muitas vezes, se conectam à rede por curtos períodos de tempo.

Este artigo faz uma análise do estado da arte na área de P2P, descrevendo brevemente as principais tecnologias e apontando desafios pendentes. Assim, a Seção 2. apresenta os principais sistemas desenvolvidos até o momento, enquanto a Seção 3. explica as técnicas utilizadas na avaliação de sistemas deste tipo. A Seção 4. finaliza o artigo, tecendo observações gerais e apontando alguns horizontes na área.

2. Mecanismos de Busca em Redes P2P

Conforme citado acima, a localização dos dados é um dos principais campos de pesquisa em redes P2P. O principal precursor dos sistemas puramente P2P estudados até hoje é o Napster ([2]). Apesar de ele ser considerado por muitos um sistema P2P, já que os dados ficam distribuídos nos diversos nós que compõem a rede, o seu mecanismo de busca não segue esta política, estando baseado em um servidor que centraliza os índices. Ou seja, apesar dos dados estarem distribuídos através dos integrantes da rede P2P, a localização pode ser realizada fazendo-se uma consulta direta ao servidor central.

Embora tenha como grande vantagem sua simplicidade, esta abordagem é inadequada pela sua falta de escalabilidade e pela presença de um ponto central de falha. Protocolos posteriores,

como o Gnutella ([3]), surgiram como uma alternativa a este modelo centralizado de busca. Nestes protocolos não existe uma entidade central responsável por indexar os dados da rede: todos os nós, a princípio, acumulam as mesmas funções de servir arquivos, efetuar buscas e auxiliar no roteamento das mensagens envolvidas no processo. As redes P2P descentralizados podem ser divididas em duas categorias principais: aquelas que não estabelecem uma estruturação forte dos dados contidos nos nós componentes, e aquelas que buscam realizar esta estruturação para aumentar a eficiência do processo de busca, conforme descrito a seguir.

Redes P2P não estruturadas

Neste tipo de rede, cujo principal representante é o Gnutella, quando um novo dado é inserido, o nó envolvido se resume a armazenar o arquivo e indexá-lo localmente. Para localizar um dado, utiliza-se “inundação”: em termos gerais, uma mensagem de busca é repassada por cada nó a todos os seus vizinhos. Conceitualmente, é realizado um *Breadth-First Search* (BFS) na rede em busca dos arquivos desejados ([4]). O número de níveis pela qual a mensagem passa nesta busca é limitada por um atributo TTL (*time to live*), com semântica análoga à utilizada no IP, mas em nível de aplicação.

Este tipo de rede é comumente utilizada na vida real em função da sua simplicidade e não necessidade de armazenamento de informações complexas nos nós ([5]). Quando um nó se desconecta, a rede deixa de contar com os dados que o nó compartilhava, mas nenhuma outra atualização é necessária para manter a consistência da rede. Desta forma, mudanças frequentes na topologia da rede têm pouco impacto no desempenho. Além disso, estes sistemas são flexíveis: podem operar sobre qualquer estrutura P2P construída, o que permite a utilização de redes ótimas segundo algum critério arbitrário, como a aproximação da configuração da rede em nível de aplicação (a rede P2P) com a rede real sobre a qual ela está baseada.

A principal desvantagem de redes P2P sem estruturação de dados é o volume de tráfego gerado durante o processo de busca. Como forma de aperfeiçoar o sistema, em especial quanto ao tráfego gerado, a literatura recente registra uma série de propostas baseadas nas características da Internet e de seu uso.

A exploração da heterogeneidade da rede é uma forma de melhorar o funcionamento do sistema como um todo ([6]). Nós com maior capacidade de processamento e banda disponível devem desempenhar papéis mais importantes dentro da rede, como p.ex. na rede FastTrack ([7]). Nesta abordagem, um conjunto de *SuperNodes* (com recursos de processamento e banda abundantes) forma o núcleo da rede, enquanto os demais nós se conectam a um destes *SuperNodes*. Cada *SuperNode* indexa, localmente, os arquivos compartilhados por todos os nós que se conectam a ele. Desta forma, o roteamento de buscas é realizado apenas dentro do núcleo da rede, reduzindo bastante o consumo da banda naqueles nós com menos recursos. Sua desvantagem é o tráfego gerado durante a transmissão dos índices para o *SuperNode*. Esta idéia foi adaptada para o Gnutella, com o nome de *Ultrapeers* ([8]).

Substituir a inundação por formas menos agressivas de distribuição de mensagens de busca (*queries*) também pode trazer bons resultados, como p.ex. *Iterative Deepening* e *Directed BFS* ([4]). Com *Iterative Deepening*, os nós realizam a inundação de maneira gradual, variando o valor de TTL. Em virtude da forma de distribuição dos dados da rede, itens populares tendem a ser encontrados sem a necessidade de um TTL alto. Caso o dado não seja encontrado, o valor de TTL é incrementado e a busca recomeça. *Directed BFS* utiliza heurísticas para repassar as buscas apenas para os vizinhos com maior chance (baseado no desempenho anterior) de levar à localização do dado. Outra técnica eficaz é o *Random Walk* ([6]), onde a consulta assume o papel de um “*walker*” que caminha de nó em nó (sem uso de *broadcast*) em busca do dado. Para aumentar a eficiência do mecanismo, são disparados vários “*walkers*” paralelos a cada busca realizada.

Conforme citado acima, redes não estruturadas podem trabalhar sobre qualquer estrutura P2P

construída. Vários projetos de pesquisa exploraram esta capacidade para melhorar o desempenho do sistema. Em [9], é explorado o interesse comum por conteúdo: os nós tendem a se aproximar e estabelecer conexões diretas com aqueles que mais freqüentemente possuem o conteúdo procurado. Em [10] é discutido um algoritmo para a minimização do diâmetro da rede. Basicamente, algoritmo trabalha na etapa de inserção de novos nós na rede: ao invés do nó se conectar a algum membro aleatório, é escolhido um membro que respeite certas propriedades, de forma que o diâmetro não seja aumentado desnecessariamente. Existem também técnicas que tentam aproximar a topologia em nível de aplicação (formada pelas conexões entre vizinhos) à topologia real (formada pelos enlaces entre roteadores e hosts), como em [11].

O *Yappers* ([5]) utiliza alguns conceitos de Tabela Hash Distribuída (técnica apresentada a seguir) de forma a criar “categorias” de nós e dados. Nós de uma classe X são responsáveis por indexar dados da classe X, nós da classe Y são responsáveis por indexar dados da classe Y, e assim por diante, para um número determinado de classes. Com esta categorização, é possível realizar o roteamento das buscas por um dado de uma determinada classe apenas entre os nós daquela classe, direcionando o processo de pesquisa de forma a melhorar a eficiência.

Redes P2P estruturadas

Uma das principais fontes de ineficiência na busca em redes não estruturadas é a falta de informações sobre a distribuição dos dados na rede. Para que haja um controle maior neste sentido e, conseqüentemente, seja possível realizar buscas mais eficientes na rede, vários grupos de pesquisa elaboraram projetos de redes estruturadas. A estruturação dos dados propriamente dita se dá através da utilização de Tabela Hash Distribuída (*Distributed Hash Table - DHT*) ([12]).

DHT tem como base a utilização de uma chave que identifica o dado sendo compartilhado. No caso de compartilhamento de arquivos, esta chave pode corresponder à codificação do próprio nome do arquivo. Aos nós também é atribuída uma chave neste mesmo espaço de chaves (ou seja, com o mesmo número de dígitos). Cada nó da rede fica “responsável” por um intervalo de chaves de dados, ou seja, armazena a informação de qual máquina possui o dado para cada uma das chaves do seu intervalo. Em todos os algoritmos, existe uma função de proximidade entre a chave sendo procurada e a chave dos nós. A partir desta função, é possível encaminhar (rotear), de acordo com as informações armazenadas no nó, a busca para o vizinho “mais próximo” do dado sendo procurado. A função de proximidade, as informações armazenadas em cada nó e a forma de roteamento resultante são características de cada algoritmo.

De forma geral, a principal vantagem deste tipo de rede é a escalabilidade e o menor tráfego gerado a cada busca realizada. Outra é que mesmo itens raros podem ser facilmente encontrados. Em compensação, a complexidade do protocolo aumenta, e é preciso constantemente manter a consistência geral da estrutura mantida para roteamento das buscas: a cada inserção ou remoção de nós/dados, deve-se atualizar as informações mantidas por determinados nós.

Em [13], foi apresentado o primeiro algoritmo que poderia ser utilizado de forma escalável com DHT. Entretanto, um ambiente estático era assumido, fazendo com que este mecanismo não pudesse ser diretamente aplicado em redes P2P. Em 2001, apareceram trabalhos importantes sobre a utilização de DHT em ambientes dinâmicos, centrados em redes P2P ([12], [14], [15], [16]). A seguir, dois destes trabalhos são brevemente explicados.

O Chord ([15]) utiliza um espaço de chaves organizado em um anel unidimensional. A proximidade entre chaves é dada pela proximidade do identificador do nó em relação ao dado dentro deste anel. Em cada nó, são mantidos dois conjuntos de vizinhos. A *successor list* é mantida para garantir a eficácia do roteamento, e armazena a identificação de k hosts que seguem o nó no espaço de chaves. A eficiência do roteamento é alcançada através da *finger list*, de $O(\log n)$ nós espaçados exponencialmente no espaço de chaves. O número de *hops* por busca é da ordem de $O(\log n)$. O

CAN ([16]) distribui as chaves em um espaço d -dimensional, cabendo a cada nó uma subregião. O processo de busca se dá através do encaminhamento da consulta para o nó mais próximo do nó objetivo. O número de vizinhos mantidos pelos nós é $O(d)$, e o número de *hops* envolvidos na busca é $O(dn^{1/2})$.

Assim como no caso de redes não estruturadas, a aproximação da topologia P2P (aplicação) com a topologia real pode trazer benefícios consideráveis. Entretanto, esta adaptação é mais complexa, posto que cada nó é responsável por uma região dos dados e deve, necessariamente, ter como vizinhos nós responsáveis pelas regiões “próximas” (segundo o critério específico do protocolo). O *Brocade* ([17]) atua com este objetivo, visando minimizar o tráfego de mensagens entre *Stub Domains*. Para tanto, é utilizada uma camada adicional de roteamento, estabelecida entre alguns “*supernodes*”. Sempre que um *supernode* intercepta uma mensagem de busca, ele identifica o domínio (de rede) onde o dado poderá ser localizado, e realiza o encaminhamento direto para um *supernode* daquele domínio. Com isso, pode-se reduzir consideravelmente o custo de rede e a latência envolvida no processo de busca.

Uma limitação na utilização direta de DHT é que cada dado é indexado por apenas uma chave; por exemplo, para que seja localizado um dado, deve-se conhecer o nome exato do arquivo buscado. Uma forma simples de se adaptar DHT para possibilitar a consulta por partes do nome do arquivo seria indexar cada “*n-grama*” (*substrings* distintas, de tamanho n , presentes em uma *string*) do nome do arquivo. Por exemplo, para a *string* “GNU Linux”, os n -gramas, para um n igual a três, são: “GNU”, “NU”, “U L”, “Li”, “Lin”, “inu” e “nux”. Ao se efetuar uma busca, a *string* seria dividida também nos seus n -gramas e seria realizada uma busca para cada um deles. A partir da consolidação dos resultados obtidos, seria possível identificar os arquivos de interesse. Ainda não existem estudos aprofundados para a avaliação do comportamento desta técnica ([18]).

Ainda sobre DHT, existe um grande foco de estudo na possibilidade de utilização de semântica de banco de dados relacional sobre a rede P2P. Isto permitiria buscas bem mais avançadas, abrindo um vasto leque de novas aplicações para estes sistemas. Tais estudos ([18]) ainda estão em fase inicial, e futuras descobertas poderão levar a grandes avanços na área de sistemas P2P.

A *Freenet* ([19]) pode ser definida como uma rede estruturada, embora fracamente. Na *Freenet*, também existe o uso de chaves *hash* para a localização de dados, porém os dados são replicados e posicionados de acordo com o volume de buscas, sem que algum nó seja responsável por algum intervalo de dados.

3. Avaliação de Sistemas P2P

São várias as métricas utilizadas para avaliar a qualidade dos sistemas P2P. A seguir, as principais métricas são brevemente descritas.

Volume de tráfego gerado pela indexação/busca de ítems. Em um sistema estável, a minimização do custo de se efetuar uma busca é muito mais importante do que a minimização do custo de indexação. Entretanto, em sistemas dinâmicos como redes P2P, o custo e a complexidade da indexação de novo ítems deve ser cuidadosamente considerada.

Latência de busca. O tempo entre início da pesquisa e recebimento dos resultados.

Sobrecarga de nós. Os mecanismos devem visar distribuir a carga entre os nós da rede, idealmente considerando as capacidades, mas sempre evitando a super-utilização de algum nó específico. Este problema pode ocorrer, por exemplo, em uma rede estruturada quando um nó fica responsável por uma região muito popular de dados.

Quantidade de informações armazenadas em cada nó. Muitas técnicas armazenam uma quantidade maior de informações (como índices dos nós vizinhos) para reduzir o tráfego gerado

pelas buscas, conforme visto na seção anterior. O número de vizinhos que o nó mantém também se encaixa nesta categoria. De forma geral, é desejável minimizar a quantidade de informações que cada nó deve armazenar.

Eficácia na localização de ítems raros e eficiência na busca de ítems populares. A distribuição dos dados em redes P2P tipicamente obedece à lei de Zipf ([20]). Esta característica pode ser utilizada para tornar mais eficiente a busca por ítems populares, bastante replicados. Por outro lado, é desejável que, uma vez presente na rede, qualquer dado possa ser localizado.

Complexidade de inserção/remoção de nós. Em especial no caso de redes estruturadas, o impacto resultante da entrada ou saída (abrupta ou não) de um nó de rede pode ser importante e até inviabilizar o uso em ambientes muito dinâmicos (como é tipicamente o caso de sistemas P2P).

Robustez. De forma análoga ao ítem anterior, o impacto de um mau funcionamento de um nó (seja acidental ou intencional) não deve inviabilizar o uso do sistema.

O processo de avaliação de sistemas P2P apresenta dificuldades, incluindo sua escala e padrões reais de uso. O problema de escalabilidade, bastante comum em simulações de rede, se agrava para o caso de redes P2P, onde o sistema em desenvolvimento tem o objetivo de atender a vários milhares de nós simultaneamente. Em virtude disto, avaliações analíticas são importantes, e estão presentes em vários estudos desta área. Exemplos de modelos analíticos que visam modelar os diferentes sistemas P2P são [21] e [22].

Apesar de simuladores de rede de uso geral serem utilizados para alguns experimentos, a simulação de P2P em larga escala ainda se restringe a simuladores específicos ([16], [15]), desenvolvidos em função do próprio projeto de pesquisa. Em [16], autores argumentam que a escala atingida ultrapassa 260.000 nós.

Quando se trata de simulação ou análise de redes P2P, um dos fatores vitais para a validade das conclusões é a utilização de padrões de uso reais, ou seja, gerar eventos (buscas, inserção e remoção de nós, etc.) e criar um ambiente (distribuição e tamanho dos arquivos) que sejam condizentes com a realidade. Neste sentido, existem vários estudos que objetivam realizar esta modelagem de maneira fiel (p.ex., [23] e [20]). Em [23], obteve-se um modelo que representa uma hora de funcionamento da rede Gnutella, no momento da pesquisa, com 400.000 nós. É bastante comum a utilização de alguns rastros de tráfego Web ([24]), que gerariam padrões de utilização compatíveis com redes P2P.

4. Conclusões

O estudo de redes P2P, como explorados atualmente, é uma área nova e em franca expansão. Diferente do que ocorre com a maioria das tecnologias, o estudo de redes P2P começou após a sua utilização em massa na Internet. Com isso, surgiu uma grande lacuna em termos de pesquisa, sendo a mesma preenchida gradativamente. Este volume de pesquisa ajuda a compreender as capacidades de sistemas P2P, mas ainda é difícil prever que tipo de aplicações resultarão deste esforço a médio prazo. De qualquer forma, a introdução deste tipo de tecnologia já caracteriza uma nova era na Internet, onde todos podem participar na construção do conteúdo disponível na rede.

De forma geral, podemos identificar vários pontos de estudo a serem tratados. A interação entre as otimizações propostas para os protocolos muitas vezes ainda é desconhecida. Também não se sabe até que ponto é possível aproveitar os benefícios das redes não estruturadas e os benefícios das redes estruturadas em um só sistema. A utilização de DHT promete ser cada vez mais difundida, mas a sua complexidade e relativa falta de maturidade nas implementações ainda limita o seu uso em grande escala. Face à revisão crítica e abrangente realizada sobre a literatura, cujos resultados são objeto deste artigo, percebe-se que muito será realizado nos próximos anos e que há grande potencial para desenvolvimento de novas idéias nesta área.

Bibliografia

- [1] Andy Oram. *Peer-to-peer: Harnessing the Power of Disruptive Technologies*. O'Reilly, March 2001.
- [2] Napster, August 2003. <http://www.napster.com>.
- [3] Gnutella, August 2003. <http://gnutella.wego.com>.
- [4] Beverly Yang and Hector Garcia-Molina. Efficient Search in Peer-to-Peer Networks. Technical report, Computer Science Department, Stanford University, 2002.
- [5] Prasanna Ganesan, Qixiang Sun, and Hector Garcia-Molina. YAPPPERS: a peer-to-peer lookup service over arbitrary topology. In *INFOCOM 2003*, March 2003.
- [6] Qin Lv, Sylvia Ratnasamy, and Scott Shenker. Can Heterogeneity make Gnutella Scalable? In *IPTPS 2002*, March 2002.
- [7] Fasttrack, August 2003. <http://www.fasttrack.nu>.
- [8] Anurag Singla and Christopher Rohrs. Ultrapeters: Another Step Towards Gnutella Scalability, December 2001. <http://www.bearguru.com/kb/articles/Ultrapeters.html>.
- [9] Kunwadee Sripanidkulchai, Bruce Maggs, and Hui Zhang. Efficient content location using interest-based locality in peer-to-peer systems. In *INFOCOM 2003*, March 2003.
- [10] G. Pandurangan, P. Raghavan, and E. Upfal. Building Low-Diameter Peer-to-Peer Networks. *IEEE Journal on Selected Areas in Communications (JSAC)*, 21(6), August 2003.
- [11] T.S. Eugene Ng, Yang-hua Chu, Sanjay Rao, Kunwadee Sripanidkulchai, and HuiZhang. Measurement-based optimization techniques for bandwidth-demanding peer-to-peer systems. In *INFOCOM 2003*, March 2003.
- [12] Peter Druschel and Antony Rowstron. Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems. In *18th IFTP/ACM International Conference on Distributed Systems Platforms (Middleware 2001)W*, November 2001.
- [13] C. Plaxtron, R. Rajaraman, and A. Richa. Accessing nearby copies of replicated objects in a distributed environment. In *ACM SPAA 1997*, June 1997.
- [14] B. Y. Zhao, J. Kubiatowicz, and A. Joseph. Tapestry: An infrastructure for fault-tolerant wide-area location and routing. Technical report, University of California at Berkeley, 2001.
- [15] Ion Stoica, Robert Morris, David Karger, Frans Kaashoek, and Hari Balakrishnan. Chord: A scalable Peer-To-Peer lookup service for internet applications. In *SIGCOM 2001*, 2001.
- [16] Sylvia Ratnasamy, Paul Francis, Mark Handley, Richard Karp, and Scott Shenker. A Scalable Content-Addressable Network. In *SIGCOM 2001*, 2001.
- [17] Ben Y. Zhao, Yitao Duan, Ling Huang, Anthony D. Joseph, and John D. Kubiarowicz. Brocade: Landmark Routing on Overlay Networks. In *IPTPS 2002*, March 2002.
- [18] Matthew Harren, Joseph M. Hellerstein, Ryan Huebsch, B. T. Loo, S. Shenker, and I. Stoica. Complex Queries in DHT-based Peer-to-Peer Networks. In *IPTPS 2002*, March 2002.
- [19] Freenet, August 2003. <http://freenet.sourceforge.net>.
- [20] Juliano Santos, Leonardo Rocha, Diêgo Nogueira, Paulo Araújo, Virgílio Almeida, and Wagner Meira Júnior. Caracterização de carga de redes Peer-to-Peer. In *SBRC 2002*, May 2002.
- [21] Hung-Chang Hsiao and Ghung-Ta King. Modeling and Evaluating Peer-to-Peer Storage Architecture. In *IPDPS 2002*, 2002.
- [22] Zihui Ge, Daniel Figueiredo, Sharad Jaiswal, James F. Kurose, and Don Towsley. Modeling peer-peer file sharing systems. In *INFOCOM 2003*, March 2003.
- [23] M. Ripeanu, I. Foster, and A. Iamnitchi. Mapping the gnutella network: Properties of large-scale peer-to-peer systems and implications for system design. *IEEE ICJ*, 2002.
- [24] W3C Web Characterization Repository, August 2003. <http://repository.cs.vt.edu>.