O artigo *Profiling a Million User DHT* relata medições de desempenho do protocolo DHT (tabelas de hash distribuídas) subjacente ao cliente BitTorrent Azureus, em condições de implantação em larga escala e sob carga de trabalho realista~~, simultaneamente~~. ~~Esse protocolo de rede é uma implementação baseada no algoritmo Kademlia.~~ Caracterizam-se a rotatividade, sobrecarga e desempenho nesse ambiente, propondo então um algoritmo de busca modificado, do lado cliente, que reduz o tempo de busca em uma ordem de grandeza.

~~Primeiramente, discutem-se características gerais das DHTs e do experimento. DHT é uma versátil API que exporta uma interface de tabela hash, provendo busca de objetos baseada em chaves, em sistemas distribuídos descentralizados. A responsabilidade pelo mapeamento das chaves para valores é distribuída entre os nós. Uma DHT pode escalar para um número enorme de participantes.~~

As medições foram realizadas entre fevereiro e maio de 2007 a partir de plataformas PlanetLab (uma rede de pesquisa ~~composta por 1080 nós distribuídos em 528 locais pelo mundo,~~ que disponibilizam serviços para o desenvolvimento de novas tecnologias de rede[[1]](#footnote-1)) e da Universidade de Washington. Os clientes foram utilizados para rastrear detalhadamente a atividade da rede DHT. Os dados foram disponibilizados para a comunidade e analisados pelos autores.

A análise obteve os seguintes resultados: na média os tempos de sessão são curtos, ~~mas com caudas longas~~; as buscas são robustas, ~~pois nós de vida longa provêm suporte para grande quantidade de tráfego de roteamento;~~ e o nó mais próximo para uma dada chave, do ponto de vista da rede, é parcialmente inconsistente ~~tanto no curto e como no longo prazo~~ ~~(motivando replicação e definição de uma taxa de atualização)~~. ~~Identificou-se que os parâmetros atuais são conservadores~~. Na prática, os fatores dominantes que influenciam o desempenho são as falhas e controle de recursos baseado em tempo-limite.

Em seguida, discute as principais ~~características e~~ desafios do algoritmo Kademlia, bem como as decisões adotadas pela implementação Azureus para tratá-los. O Azureus mantém uma DHT para tratar o gargalo de escalabilidade do protocolo BitTorrent, que prevê um coordenador para a lista de pares previamente conectados, permitindo inclusive distribuição sem coordenador central.

~~Os desafios do Kademlia são~~ a alta rotatividade, pois há pares chegando e partindo rapidamente, tornando a tabela de roteamento obsoleta (ao invés de consultar um par, o Azureus consulta até 10 pares paralelamente); manter a consistência, pois buscas concorrentes podem não ter a mesma visão, se os nós partirem prematuramente (Azureus usa replicação de cada valor nos 20 nós mais próximos da chave); e as falhas, pois como o transporte de mensagens é feito via UDP, a detecção de falhas deve ser implementada no nível de aplicação (Azureus aplica tempo de expiração de 20 segundos, considera que um nó cai se não responde a duas mensagens bate-e-volta e busca por chave aleatória a cada 5 minutos, para verificar se a rede está ativa).

Apresenta então as medições efetuadas nas características principais, que são: medir a distribuição dos tempos de sessão dos nós na DHT, identificar como se comporta a inicialização e sobrecarga para manter a DHT, avaliar a consistência e persistência e, por fim, identificar a probabilidade de resposta.

~~O duração das sessões determina a que taxa as entradas da tabela devem ser sondadas para atualização. O método utilizado na medição foram de amostras aleatórias, com 30000 nós testados, repetindo a sondagem a cada 2,5 minutos. O experimento durou 48 horas, identificando que apenas 5% dos nós permaneciam ativos. Sugere-se que os tempos de sessão são da ordem de horas. Não foi possível recuperar a duração precisa da duração antes da observação, nem foram capazes de contabilizar o viés dos nós de vida longa serem super-representados.~~

~~Com relação à inicialização e sobrecarga, há os desafios de não ser possível prever a longevidade dos nós e o risco de adicionar nós de vida curta (incorporar seus recursos) na rede. Azureus incorpora os nós instantaneamente na DHT, mas as tabelas de roteamento são atualizadas lentamente (aumento gradativo da responsabilidade dos pares). O experimento, com duração de dois dias, consistiu em conectar simultaneamente 125 clientes Planet-Lab, confirmando que o trafego de mensagens aumenta gradualmente. O rastreamento indica que a sobrecarga para manter a DHT domina o consumo de recursos (81% das mensagens trocadas eram ping ou popular tabelas de roteamento).~~

~~Sobre consistência e persistência, para avaliar consistência de curto prazo foram inseridas 931 chaves aleatórias na DHT, medindo-se a visibilidade de cada inserção a partir dos 250 clientes Planet-Lab. Cada cliente observou diferentes conjuntos de réplicas. Para avaliação de longo prazo, ou seja, mudanças no conjunto de réplicas foram feitas reinserções periódicas de chaves por 125 clientes em intervalo de 1, 8 ou 24 horas, obtendo como resultado que inserções periódicas causam pouco impacto na persistência dos dados.~~

~~A probabilidade de resposta de uma mensagem foi medida como uma função do tempo após a mesma ter sido enviada. Foram medidas mais de 45 milhões de mensagens e observou-se que a probabilidade de resposta manteve-se notavelmente estável tanto no curto como no longo prazo, em todos os clientes.~~

~~Os autores identificaram que~~ o desempenho é dominado por parâmetros que impactam o atraso de roteamento: paralelismo de busca, tempo limite e taxas de processamento limitadas das mensagens.

~~Com base nos parâmetros ressaltados,~~ Assim, propõem-se modificações no algoritmo utilizado pelo Azureus que melhoram substancialmente o tempo de resposta das buscas. O algoritmo revisado utiliza a probabilidade de resposta de mensagem como parâmetro base para encontrar o equilíbrio entre desempenho e sobrecarga. A solução proposta remove os controles por limite de tempo e define em 6 o número de buscas em paralelo. Com isso, obteve uma melhoria enorme no tempo médio de busca (de 127 para 13 segundos), o número de nós contatados triplicou (de 52 para 150) e, dada a eliminação dos limites de tempo, o número de nós contatados decresceu por uma ordem de grandeza.

~~Dado que as buscas no Kademlia disparam atualizações nas tabelas de roteamento, sua realização frequente diminui a obtenção de tabelas obsoletas (resultando em menos buscas). O artigo expôs a necessidade de atualizar agressivamente as tabelas, na ausência de demanda e sob forte rotatividade de pares (que é o caso do BitTorrent).~~

~~Por fim, discutem-se alguns trabalhos relacionados que obtém resultados distintos. Aumentar o paralelismo melhorou o desempenho da busca no Azureus, ao contrário dos resultados obtidos em outros trabalhos com redes que utilizam DHT, como em Stutzbach et al (buscas do eMule), nas quais essa modificação apenas induziu a aumento da sobrecarga sem ganho de desempenho.~~

1. <http://www.planet-lab.org/> [↑](#footnote-ref-1)