Ouroboros: Uma DHT Auto-organizável Tolerante a *Churn*

João Carvalho, Nuno Preguiça, João Leitão

NOVA LINCS & Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa

1 Motivação

As tabelas de dispersão distribuídas (DHT, do inglês *Distributed Hash Table*) foram propostas há mais de 10 anos como uma componente de suporte a aplicações distribuídas de grande escala. As DHT's foram utilizadas em vários contextos, como o suporte para aplicações entre-pares (*Peer-to-Peer*) [5] assim como na computação em nuvem, onde as DHT's são frequentemente utilizadas no desenho de bases de dados distribuídas, como por exemplo no Cassandra [3].

O sucesso das DHT's deve-se ao seu desenho descentralizado, o que lhes permite ser altamente escaláveis. Na sua essência a maioria das DHT's recorre a um anel [11, 9] que liga os nós do sistema de acordo com o seu identificador. O correcto funcionamento de uma DHT depende da manutenção de uma topologia em anel entre os nós. As DHT's usualmente recorrem ainda a um conjunto de ligações adicionais entre os nós (*ponteiros*) para acelerar a navegação no anel [11, 9].

Apesar da eficiência alcançada através do uso de topologias estruturadas baseadas em anéis, existem vários aspetos negativos que limitam o uso das DHT's na prática, sendo o mais relevante o facto de estas apresentarem uma baixa tolerância a *churn*[1], durante o qual a estrutura do anel pode ficar comprometida, levando à formação de múltiplos anéis independentes ou outras incorreções na topologia [10]. Este fenómeno foi observado na prática quando a DHT que suportava o índice distribuído do Skype falhou devido a um fenómeno de *churn*, o que levou á indisponibilidade do serviço por um período superior a 24 horas [1].

Observamos que a baixa tolerância a *churn* exibida pelas DHT's tem como base o facto dos algoritmos usados para estabelecer e gerir a topologia, dependerem da correção da topologia do anel, condição essa que tipicamente não é válida durante períodos de *churn*. Neste trabalho, pretendemos endereçar esta limitação das DHT's recorrendo ao uso de técnicas comumente usadas na gestão de topologias dinâmicas não estruturadas (*i.e,* aleatórias) seguindo metodologias encontradas em propostas anteriores como nos protocolos T-Man [2] e X-BOT [7].

Com o aparecimento de novas tecnologias, como o *Web Real Time Communication* (*WebRTC*), que permite ligações directas e transparentes entre *browsers*, existe um novo interesse no desenho de DHT's altamente eficientes e robustas, para suportarem novas aplicações web enriquecidas com modelos de comunicação entre-pares [13].

2 Proposta

Para ultrapassar as limitações nos desenhos das DHT's, propomos o *Ouroboros*[2], um novo algoritmo auto-organizável para a gestão de topologias de DHT's baseadas em anel. No *Ouroboros* recorremos ao uso de duas vistas parciais, uma vista ativa e outra passiva. A vista ativa de um nó n guarda a informação sobre os nós que são vizinhos diretos de n no anel e os nós que são usados por n como ponteiros para acelerar a navegação no anel. Em contraste, a vista passiva de n mantém uma amostra dinâmica e aleatória (de tamanho limitado) dos nós no sistema, que é mantida através de um processo de rumor (do inglês *gossip*) executado entre todos os nós do sistema. O uso de duas vistas parciais para fins diferentes foi originalmente proposto no HyParView [6].

Quando um novo nó n' entra no sistema, este determina a sua posição no anel recorrendo ao mecanismo de encaminhamento disponibilizado pela DHT. Neste processo, n' recolhe também alguma informação sobre a filiação do sistema que este usa para inicializar a sua vista passiva. Uma vez que a posição inicial atribuída a n' neste processo pode estar errada, devido a erros na topologia, recorremos a um protocolo inspirado na abordagem proposta pelo X-Bot [7]. Esta abordagem permite a cada nó p utilizar continuamente os conteúdos da sua vista passiva, para ajustar os conteúdos da sua vista activa. Note-se que os conteúdos da vista passiva de p não dependem

[1] Churn é um fenómeno caracterizado pela entrada e saída simultânea de vários nós no sistema [12].

[2] Ouroboros é um símbolo representado por uma serpente a comer a sua própria cauda, simbolizando o contínuo renascimento e transformação (`https://en.wikipedia.org/wiki/Ouroboros`).

da topologia definida pela sua vista activa, visto que esta é gerida por um processo de rumor totalmente aleatório. Consequentemente, ainda que o anel se encontre numa configuração errada, a vista passiva de p pode conter os identificadores do sucessor e antecessor desse nó, permitindo a p corrigir a topologia localmente. Assim garante-se que o anel e os ponteiros adicionais adaptam-se continuamente e que a DHT se auto-organiza e converge continuamente para a configuração correta. Contrariamente ao proposto no X-Bot [7], no Ouroboros este processo adapta a vista ativa de cada nó usando dois critérios complementares: *i*) cada nó conhece e liga-se ao seu sucessor e antecessor adequados; *ii*) enviesa os ponteiros extra mantidos na vista activa de forma a promover ligações de baixa latência.

3 Sumário & Discussão

O objectivo primário deste trabalho é assegurar que a topologia da DHT consegue recuperar de fenómenos de *churn*. Para isso, e contrariamente a outras propostas na literatura (*e.g,* [4, 11, 9]), recorremos a um conjunto de técnicas usualmente utilizadas para gerir redes sobrepostas não estruturadas [6, 7], para permitir que cada nó tem a capacidade de corrigir erros na topologia. Um aspecto essencial neste processo é o uso de uma segunda vista cujos conteúdos são independentes da topologia da DHT, o que torna o processo de indentificação dos vizinhos correctos independente do estado da topologia. O trabalho mais próximo encontrado na literatura é o T-Chord [8] que recorre aos mecanismos do algoritmo distribuído T-Man [2]. No entanto, e como demonstrado anteriormente, o T-Man não protege a conectividade global da rede, podendo por isso levar a topologia da DHT a quebrar [7]. Para além disso, os autores deste trabalho não abordam cenários com *churn*.

Adicionalmente, tiramos proveito da natureza dinâmica da vista passiva mantida por cada nó, de forma a enviesar os ponteiros mantidos por cada nó para promover ligações de baixa latência. Otimizando assim os tempos de comunicação e encaminhamento de mensagens sobre a DHT.

Referências

[1] Villu Arak. Skype Blog: what happened on august 16, August 2007. URL `http://heartbeat.skype.com/2007/08/what_happened_on_august_16.html`.

[2] Márk Jelasity, Alberto Montresor, and Ozalp Babaoglu. T-Man: Gossip-based fast overlay topology construction. *Journal Computer Networks: The International Journal of Computer and Telecommunications Networking*, 53(13):2321 – 2339, August 2009.

[3] Avinash Lakshman and Prashant Malik. Cassandra: A decentralized structured storage system. *ACM SIGOPS Operating Systems Review*, 44(2):35 – 40, April 2010.

[4] Sergey Legtchenko, Sébastien Monnet, Pierre Sens, and Gilles Muller. Relaxdht: A churn-resilient replication strategy for peer-to-peer distributed hash-tables. *ACM Trans. Auton. Adapt. Syst.*, 7(2): 28:1–28:18, July 2012. ISSN 1556-4665.

[5] J. Leitão. *Topology Management for Unstructured Overlay Networks*. PhD thesis, Technical University of Lisbon, September 2012.

[6] J. Leitão, J. Pereira, and L. Rodrigues. Hyparview: A membership protocol for reliable gossip-based broadcast. In *Dependable Systems and Networks, 2007. DSN '07. 37th Annual IEEE/IFIP International Conference on*, pages 419–429, June 2007.

[7] J. Leitão, J.P. Marques, J. Pereira, and L. Rodrigues. X-bot: A protocol for resilient optimization of unstructured overlay networks. *Parallel and Distributed Systems, IEEE Transactions on*, 23(11): 2175–2188, Nov 2012.

[8] A. Montresor, M. Jelasity, and O. Babaoglu. Chord on demand. In *Peer-to-Peer Computing, 2005. P2P 2005. Fifth IEEE International Conference on*, pages 87–94, Aug 2005.

[9] Antony Rowstron and Peter Druschel. Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems. In *Proceedings of the IFIP/ACM International Conference on Distributed Systems Platforms Heidelberg (Middleware'01)*, pages 329 – 350, Heidelberg, Germany, November 2001.

[10] Tallat Shafaat, Ali Ghodsi, and Seif Haridi. Handling network partitions and mergers in structured overlay networks. In *Proceedings of the 7th IEEE International Conference on Peer-to-Peer Computing (P2P'07)*, pages 132 – 139, Galway, Ireland, September 2007.

[11] Ion Stoica, Robert Morris, David Liben-Nowell, David R. Karger, M. Frans Kaashoek, Frank Dabek, and Hari Balakrishnan. Chord: A scalable peer-to-peer lookup protocol for internet applications. *IEEE/ACM Trans. Netw.*, 11(1):17–32, February 2003. ISSN 1063-6692.

[12] Daniel Stutzbach and Reza Rejaie. Understanding churn in peer-to-peer networks. In *Proceedings of the 6th ACM SIGCOMM Conference on Internet Measurement (IMC'06)*, pages 189 – 202, Rio de Janeriro, Brazil, October 2006.

[13] Liang Zhang and Alan Mislove. Building confederated Web-based services with Priv.io. In *Proceedings of the 1st ACM Conference on Online Social Networks (COSN'13)*, Boston, MA, October 2013.