Senior Project Proposal - Distributed Hash Tables on Adhoc Networks

Prasanna Gautam

September 22, 2010

Introduction

In an age where decentralization and parallelization are the key, distributed hash tables (DHTs) have had revolutionary effects. In essence, a DHT is a system designed to harness the storage and network resources of large number of computers by providing a hash-table interface where the *(key,value)* pairs are stored across various nodes; where *keys* are hashes of the objects or *values* stored in the system [6]. Some of the most common uses, apart from academic research projects have been in peer-to-peer file transfer protocols like Bittorrent's distributed tracker where nodes can be distributed anywhere in the internet and can drop in and out of connection. More recently, projects like Cassandra, Bigtable have emerged to provide structured storage systems using DHTs [2][5]. A parallel in the wireless networks of this type of communication is mobile ad-hoc networking (MANET) where the communicating nodes do not rely on preexisting infrastructures such as routers or access points to communicate with each other. Instead, each node routes by forwarding data to other nodes. My project aims to explore the use of DHTs on adhoc networks to answer questions about the practicality, reliability, integrity and consistency in such networks. From experience I know that testing and measuring adhoc in the field is difficult at best, so I want to leverage the existing network simulation frameworks like NS-2/NS-3, OPNET, GloMoSim, NetSim to accurately model DHTs on ad-hoc networks and creating my own simulation environment that will augment the existing simulation infrastructure or if needed, be entirely new based on the requirements of the project.

Description

A DHT layer shields many difficult issues including fault-tolerance, locating object, scalability, availability, load balancing for distributed application designer[8]. Existing algorithms like "consistent hashing" implemented in DHTs have been proven to be fault tolerant and able to adapt to changing network topologies by efficiently par-

titioning a keyspace among distributed set of nodes and provide an additional overlay network which connects nodes such that the node responsible for any key can be efficiently located[4]. Various options have been explored to implement DHTs in MANETs using a proximity-aware DHT Pastry[7] and on-demand MANET routing protocol DSR[3]. Based on my experience working on the Random Walk Gossip algorithm implementation on POSIT[1], I have realized that testing mobile ad-hoc networks consistently in the field can be very challenging without a detailed test simulation to base it on. My goal is to design a simulator for experimenting with DHTs on Ad-hoc networks, exploring network topologies and power consumption in the devices communicating. That said, I will be exploring and understanding various existing simulation frameworks to come up with a workable model to run my tests. The most important questions I aim to explore and answer are:

- How power efficient can DHT on Adhoc network be?
- What kinds of network topologies are preferrable?
- What are the effects of adding or removing nodes on the network?
- How do you deal with network partition?

The simulator I want to work on would be written in clojure, a lisp dialect running on the java JVM. My main motivation is that given that I have a rather short time frame of a year to complete this project, I want to be able to test the various components of the applications as independent of each other as possible. Using a functional language makes it much easier to do as I can test individual functions separately. I can also use various existing Java and C libraries when needed and simulate code that's much closer to the actual system's code. The architectures I'll be looking primarily would be Android phones- the ADP1 and the Nexus one and ad-hoc network implementations in Linux in laptops. I'll run a battery drain test on my targets with various stress levels to get an idea of the network usage with respect to power for the phones and run the simulations around those parameters. The end result of this project would be a simulator that does the network simulation that lets us create DHTs on a random number of nodes, observe progress on each node and report and record the results. The variables that are relevant are - loss of data when a certain percentage of nodes drop, power consumption on average of each of the algorithms, throughput of the network, availability of a key/value pair on a network over time. Along with this, I want to explore these conditions in various network topologies and network conditions.

Timetable

- End of September - Finish research on frameworks, relevant papers, required devices, libraries
- 15th October - Get data about the phones being tested
- 22nd October - Finish preliminary architecture of the simulator
- 29th October - Report on existing frameworks
- 12th November - Technical Paper Presentation
- 7th December - Prototype of the simulator for implementing basic DHT network on simulated ad-hoc networks
- 15th December - Project report and findings from the semester

Budget

I'll need two or more laptops/netbooks that I can create ad-hoc networks on and get real data that I need for the project. Also since adhoc networks on laptops is much better understood, that'd let me benchmark and test against existing ad-hoc network implementations. Since books on this topic are rather sparse in the library, I'd like to order a few books too. Most of the network simulators I've mentioned are either open source or free under academic license but getting software might be necessary.

Conclusion

With this project I'm aiming to get a better understanding of the state of Distributed Hash Tables on Adhoc networks and test the practicality of those networks. I'll also learn about running network simulation and testing hypotheses about various DHTs on Adhoc network implementations with them. In the end, I want to have a working simulator to test and observe various network conditions by simulating real network hardware and conditions as well as possible.

References

[1] M. Asplund, T. deLanerolle, C. Fei, P Gautam, R. Morelli, S. Nadjm-Tehrani, and G. Nykvist. Wireless ad hoc dissemination for search and rescue. In *7th International Conference on Information Systems for Crisis Response and Management*. ISCRAM, May 2010.

[2] Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach, Mike Burrows, Tushar Chandra, Andrew Fikes, and Robert E. Gruber. Bigtable: A distributed storage system for structured data. In *IN PROCEEDINGS OF THE 7TH CONFERENCE ON USENIX SYMPOSIUM ON OPERATING SYSTEMS DESIGN AND IMPLEMENTATION - VOLUME 7*, pages 205–218, 2006.

[3] David B. Johnson and David A. Maltz. Dynamic source routing in ad hoc wireless networks. In *MOBILE COMPUTING*, pages 153–181. Kluwer Academic Publishers, 2002.

[4] David Karger, Eric Lehman, Tom Leighton, Rina Panigrahy, Matthew Levine, and Daniel Lewin. Consistent hashing and random trees: distributed caching protocols for relieving hot spots on the world wide web. In *STOC '97: Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 654–663, New York, NY, USA, 1997. ACM.

[5] Avinash Lakshman and Prashant Malik. Cassandra- a decentralized structured storage system.

[6] Moni Naor and Udi Wieder. Novel architectures for p2p applications: the continuous-discrete approach. *ACM TRANSACTIONS ON ALGORITHMS*, 3(3):50–59, 2007.

[7] Antony Rowstron and Peter Druschel. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems, 2001.

[8] Himabindu Pucha Saumitra, Saumitra M. Das, and Y. Charlie Hu. How to implement dhts in mobile ad hoc networks? In *Student poster, the 10th ACM International Conference on Mobile Computing and Network (MobiCom 2004), September-October*, 2004.