CSC2231 – Internet Systems and Services

Paper Review – Chord
Name: Alex Wun
Date: Nov. 6th, 05

The authors of this paper propose a P2P lookup protocol based on consistent hashing. Briefly, their algorithm is as follows: Each node is hashed to an id of *m* bits, where *m* must be chosen large enough to prevent collisions. An *m*-bit key (*k*) hashes to the first node with identifier *M* in $[0, 2^m – 1]$ such that *k* is less than or equal to *M*. The routing table for a node with id *M* lists the succeeding nodes that hash to keys $k = M + 2^{n-1}$ for *n* in [1, *m*]. Searching for a node occurs by hashing into this routing table and forwarding the search as necessary.

This hashing mechanism is deterministic and can therefore be implemented efficiently. The deterministic nature of the protocol also means that a P2P network built on top of Chord will exhibit predictable behaviour for searches. For researchers, deterministic P2P networks are much easier to analyze as well.

However, deterministic protocols depend heavily on preconditions that are required for correct operation. A deterministic algorithm can easily be broken by maliciously misrepresenting state. Dealing with these problems seems to result in “patch” solutions that detract from the elegance of the fundamental protocol. For instance, Chord resorts to a successor list to deal with failures. Which leads to another point: that there are periods of inconsistency whenever nodes enter/leave the network as Chord attempts to stabilize the state of routing tables. However, as the authors admit, nodes are constantly entering and leaving in practical P2P networks. This means that a Chord P2P will never be stable – although they claim that searches aren’t adversely affected. It would be interesting to obverse Chord’s behaviour in the face of network changes and failures occurring simultaneously rather than separately as the authors have done.

Regardless, Chord presents a very interesting and powerful distributed hashing mechanism. It seems to be suited for tightly controlled, high stability networks rather than Internet scale applications though. In which case, you could argue that non-overlay approaches may be more appropriate.