Utilizing Hierarchy in Node Namespace for Efficient Query Routing in Chord based P2P Telephony

Aug 28th, 2005
Vishal Kumar Singh
Problem with Chord based P2P Telephony

- Loss of locality information of the nodes
  - A physically (internet distance) closer node may be “far” in node id space and hence far on chord ring
    - potentially multiple hops to reach it
  - How much is the impact of randomization
    - Call set up time ?(Need to be determined)
Properties of Chord based DHT

- Each node knows more about nodes closely following it than about nodes farther away in node name space.
- The key is stored in a node whose id is closer to key id.
- A query to a key stored in a closer node in node name space will be faster.
  - Faster call set up time if close by nodes stored locally.
Possible Approaches

- **Chord of Chords based approach**: Break down node name space e.g. Domain name system kind of hierarchy
  - e.g. irt.cs.columbia.edu implies 4 level of chord rings
  - e.g. +001-212-939-7040 implies 4 level of chord rings
  - Local nodes are clustered in same chord ring and are not dispersed. Idea is to break name space to minimize message movements back and forth across boundaries

- **Node-Id generation algorithm based approach**
  - The node id generated takes into account the distance from a landmark node (designated node) (prevent loss of proximity in node id)
Chord of Chord based hierarchy

- A query can either traverse up to upper chord rings until it reaches a common ring on source and destination hierarchy or point of divergence in name hierarchy, else the query can directly start from upper-most point (this corresponds to a ring) and traverses there onward to lower rings.
- The overall algorithmic complexity remains same but the query and maintenance messages going back and forth become localized.
- Reduction in call set up time.
The innermost ring (top) which is the highest level looks up for country codes or domain identifiers - top level identifier.

The node then performs look up on area codes (sub domain) in the inner chord ring and so on – lower level identifiers.
Call flow examples

- irt.cs.columbia.edu to comet.ee.columbia.edu
  - Query “ee” at one level above
  - Query “comet” once “ee” is found
- irt.cs.columbia.edu to voiplab.iisc.ernet.in
  - Query “in” at top level
  - Query “ernet” -> “iisc” -> “voiplab”
- The same principle can be used with phone numbers too
Message Flow and Look up – Algorithmic analysis

- If $N$ is total number of nodes, look up complexity in chord is $\log(N)$, for $m$ queries number of messages in chord = $m(\log N)$, per node number of message = $m(\log N)/N$

- $N = n \times q \times r$, Three levels of hierarchy, Top or innermost chord has $n$ nodes, $r$ nodes in lowermost chord, With 3 levels of look up complexity = $\log(n) + \log(q) + \log(r)$, which is same as $\log(N)$.

- If all queries are routed from top most or innermost ring, the number of messages per innermost ring is $m*(\log(N))/n$, this is $N/n$ times more messages per node in top level (innermost) chord.

- But, Not all queries go through innermost or top most ring.
**Message Flow and Look up – Call Flow Distribution**

If the total number of calls (queries) is \( m \), \( m = A + B + C \). Topmost or inner most chord has \( n \) nodes, calls which need to go through innermost chord = \( C \), similarly number of calls in same chord ring = \( A \) and going to chord at intermediate level is \( B \).

- e.g. consider numbers 091-80-5103, 091-441-2398, 044-743-7654. Calls from 091-80-5103 to 044-743-7654 has to go through top level whereas call to 091-441-2398 goes only to next higher level chord ring.

So, only \( \frac{C}{A+B+C} \) fraction of total calls go through the inner most layer and hence extra message per node in the innermost layer becomes \( \left( \frac{N}{n} \right) \left( \frac{C}{A+B+C} \right) \). So, we can see that there is an increase in number of messages handled nodes in top/innermost ring by \( \left( \frac{N}{n} \right) \) ratio of calls which have to go through top/innermost ring.

The number of calls will depend on call distribution. If the inter-domain or inter country (ISD) calls are lower in proportion to total number of calls the load in terms of message exchange will not be high.
Interconnecting Chords to do look up

Chord Rings
- [clic, irt, ncl, nsl, vision] -> form a chord ring identified by LABS
- [cs, ee, ieor, physics] -> form higher level chord ring identified by DEPTS

A call coming from a node in irt.cs e.g. vs2140@irt.cs trying to call kns10@comet.ee needs to go to the chord ring DEPTS and then go down to ee -> comet. This needs that chords identified by LABS to be linked to DEPTS. There are two approaches to linking two different chord rings.
  - **Bootstrap way** - Every node has a list of nodes which act as super nodes for each of the levels. These super nodes are somewhat fixed in nature – means any change has to be propagated to lower level nodes which cache them
  - **Chord way of interconnecting**
Interconnecting Chords

- Consider example chords
  - [clic, irt, ncl, nsl, vision, comet] -> LABS
  - [cs, ee, ieor, physics] -> DEPTS
  - [stanford, gatech] -> UNIVS

- All nodes know their full name e.g. vs2140.irt.cs. -> so it knows its own chord ring i.e. LABS and that query has to go to chord ring in which “cs” is present which is DEPTS

- There are nodes which participate in multiple chords. The node in LABS whose node-id is closest to id “cs” also participates in chord at higher layer i.e. in DEPTS. In this case clic~cs (clic and cs are closer in node id space), clic participates in LABS and DEPTS. Similarly, a node in DEPT must be a member of “gatech” in UNIVS which will be “ieor” because ieor~gatech, so node “clic” participates in LABS and DEPTS, node “ieor” participates in DEPTS and UNIVS and hence query is routed from lower to higher chords (outer to inner ring).

- The query moves from higher to lower layers in same way. When “cs” in DEPTS gets the query to kns10.comet.ee it first does a look up for ee and finds the node “ee” and hand-offs the query. The ee which is also a member of LABS group searches for “comet” and hand offs the query for “kns10” and so on.