Routing Indices For Peer-to-Peer Systems

A Critical Review

By Alireza Mortezaei
Outline

• Summary
• Technical Content
  • Strengths
  • Weaknesses
• Thoughts and Recommendations
• Final Evaluation
Background

• P2P systems facilitates content sharing

• No global knowledge over the network

• Routing is a major open problem in P2P networks
Taxonomy of Solutions
The Nature of Query

• Conjunction of keywords
• Forwarded until satisfied
• **Stop condition**: certain number of documents
• **Keywords**: subject topics e.g.

"Find me 100 documents on database and networks"
General Idea

• Direction versus location
• Use hints to guide the query through the network
• Basis of routing:

“Choose the neighbor that is more likely to yield better result”
General Idea Cont’
Basic Model

• Compound Routing Indices (CRI):
  – Local indices
  – Routing indices

• Route table functionality

• Summarize RIs by coarsely categorizing topics:
  – Compresses RI table
  – Information loss due to summarization
## Basic Model Cont'

![Diagram showing network connections between A and B with data tables]

<table>
<thead>
<tr>
<th>#</th>
<th>DB</th>
<th>N</th>
<th>T</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>130</td>
<td>23</td>
<td>42</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>412</td>
<td>123</td>
<td>300</td>
<td>65</td>
</tr>
</tbody>
</table>

- **Local Index**
- **Routing Index**

- **Total number of documents**
- **Number of documents on topic T**
300 = 150 \cdot 5 + 100 \cdot 5 + 100 \cdot 10
Estimation Model

- Estimate the number of documents along the path as a measure of “goodness”
- Prioritize neighbors based on their estimated goodness
- GLOSS estimator:

\[ \text{NumOfDocs} \times \prod_{i} \frac{\text{CRI}(s_i)}{\text{NumOfDocs}} \]
Estimation Model Cont’

<table>
<thead>
<tr>
<th>#</th>
<th>DB</th>
<th>N</th>
<th>T</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>...</td>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>500</td>
<td>500</td>
<td>04</td>
<td>20</td>
</tr>
<tr>
<td>B</td>
<td>200</td>
<td>56</td>
<td>00</td>
<td>08</td>
</tr>
</tbody>
</table>

\[
G(b) = 500 \times \frac{5100}{5000} \times \frac{30}{50} \times \frac{24}{300} = 8 \\
G(d) = 100 \times \frac{56}{200} \times \frac{28}{200} = 0
\]
Creation and Maintenance

A's table

<table>
<thead>
<tr>
<th>#</th>
<th>DB</th>
<th>N</th>
<th>T</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>400</td>
<td>240</td>
<td>120</td>
<td>5</td>
</tr>
<tr>
<td>E</td>
<td>200</td>
<td>50</td>
<td>0</td>
<td>28</td>
</tr>
<tr>
<td>B</td>
<td>300</td>
<td>100</td>
<td>44</td>
<td>42</td>
</tr>
</tbody>
</table>

B's table

<table>
<thead>
<tr>
<th>#</th>
<th>DB</th>
<th>N</th>
<th>T</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>150</td>
<td>60</td>
<td>34</td>
<td>12</td>
</tr>
<tr>
<td>C</td>
<td>50</td>
<td>5</td>
<td>0</td>
<td>30</td>
</tr>
<tr>
<td>D</td>
<td>100</td>
<td>35</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>A</td>
<td>600</td>
<td>290</td>
<td>120</td>
<td>33</td>
</tr>
</tbody>
</table>

Update for E: 500 340 1640 473 6065

Update for C: 860 386 464 43 150

Update for D: 800 355 154 75 175
Variations of the Design

- Compound Routing Indices
- Hop-count Routing Indices
- Exponentially Aggregated Routing Indices
Hop-count RI

- Store aggregated RIs for a maximum of $m$ hops. $m$ is also called \textit{horizon}.
- New goodness: The hop that yield the most result with the least number of hops.
- Assumptions:
  - Uniform distribution
  - Regular tree with constant fan-out
- Estimator:

$$goodness_{hc}(\text{Neighbor}_i, Q) = \sum_{j=0}^{m} \frac{goodness(N_i[j], Q)}{F_j}$$
Hop-count RI Cont’

<table>
<thead>
<tr>
<th></th>
<th>1 Hop</th>
<th></th>
<th>2 Hop</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#</td>
<td>DB</td>
<td>N</td>
<td>T</td>
</tr>
<tr>
<td>B</td>
<td>150</td>
<td>60</td>
<td>34</td>
<td>12</td>
</tr>
<tr>
<td>C</td>
<td>50</td>
<td>5</td>
<td>0</td>
<td>30</td>
</tr>
</tbody>
</table>

\[
G(b) = \frac{1}{150} \times \frac{34}{150} \times \frac{12}{150} + \frac{230}{230} \times \frac{140}{230} \times \frac{0}{230} = 2.72
\]

\[
G(c) = \frac{50}{1} \times \frac{0}{50} \times \frac{30}{50} + \frac{35}{35} \times \frac{8}{35} \times \frac{45}{35} = 140
\]
Exponentially Aggregated RI

- Compresses HRI tables into one table
- Simply apply the regular-tree cost to HRI model
- Is not bounded by the *horizon*
- Effects of updates dissipates through out the path
- Trades storage for accuracy
Cycles in the Network

Updates can loop indefinitely throughout the network
Dealing With Cycles

• No solution
• Cycle avoidance solution
• Cycle detection and recovery solution
No Solution?

Simply no changes are made to any of the algorithms

- **Pros:**
  Works for Hop-count and Exponential RI

- **Cons:**
  Can loop indefinitely for CRI
Cycle Avoidance

Do not allow update connections that results in cycles

- **Pros:**
  Consistency is maintained since there are no loops

- **Cons:**
  Sub-optimal update model
Cycle Detection & Recovery

Include a message identifier in the update in order to detect cycles

- **Pros:**
  Consistency of the system is maintained

- **Cons:**
  Race condition, ignores hop-count
Strengths

- **Major**: Great experiment setup and high gain
- **Minor**: High peer autonomy
- **Minor**: DV inspired update propagation
Weaknesses

- **Major**: Starvation
- **Major**: keywords, compression and narrowed application domain
- **Major**: Race condition regarding cycle recovery solution
Starvation

VS.

G(c) = 100

G(b) = 10
Race Condition
Future Extension

- Probabilistic Routing Model:
  Instead of blindly choosing the neighbor with the highest goodness, distribute query among paths based on the combination of goodness and query size.
Final Evaluation

• Main issues:
  – Radical assumptions regarding certain design decisions
  – Narrowed application space

• Final score: 2.75
Questions?
Thank You!