Hybrid Periodical Flooding in Unstructured Peer-to-Peer Networks

ICPP’03
Outline

• Introduction
• Hybrid Periodical Flooding
• Performance Evaluation
• Conclusion
Introduction

• Search Mechanisms on Unstructured P2P Networks
  – Blind flooding
  – Statistics-based
Introduction

• Blind flooding
  – Blind flooding mechanism relays the query message to all its logical neighbors, except the incoming peer
  – Large volume of unnecessary traffic
Introduction

• Statistics-based
  – a peer selects a subset of its neighbors to relay the query based on some statistics information

• Partial Coverage Problem
  – Large percentage of the peers may be unreachable no matter how large the TTL value is set
Introduction
Introduction

• Uniformed selection of relay neighbors
  – Breadth-first search (BFS)
  – Depth-first search (DFS)

• Weighted selection of relay neighbors
  – Directed BFS (DBFS)
Hybrid Periodical Flooding

- **Goal**
  - Reducing the unnecessary traffic
  - Solving partial coverage problem

- **Statistics-based + Periodical flooding**
Hybrid Periodical Flooding

• Periodical flooding (PF)
  – Given a peer with $n$ logical neighbors and the current value of TTL, the number of relay neighbors, $h$, is defined by the following function $h = f(n, \text{TTL})$

  $Ex. \ h = f_{BFS}(n, \text{TTL}) = n \ , \ h = f_{DFS}(n, \text{TTL}) = 1.$
Hybrid Periodical Flooding

\[ f(n, \text{TTL}) = \begin{cases} \frac{1}{2}n, & \text{if TTL is odd} \\ \frac{1}{3}n, & \text{if TTL is even} \end{cases} \quad \text{TTL} = 7 \]

(a) BFS

(b) PF
Hybrid Periodical Flooding

Table 1. PF and Blind Flooding

<table>
<thead>
<tr>
<th></th>
<th>TTL</th>
<th>Query Msg</th>
<th>New Peers</th>
<th>Msg Per Peer</th>
</tr>
</thead>
<tbody>
<tr>
<td>BFS</td>
<td>7</td>
<td>4</td>
<td>4</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>17</td>
<td>8</td>
<td>2.12</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>15</td>
<td>2</td>
<td>7.50</td>
</tr>
<tr>
<td></td>
<td></td>
<td>36</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF</td>
<td>7</td>
<td>2</td>
<td>2</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>4</td>
<td>4</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>9</td>
<td>8</td>
<td>1.12</td>
</tr>
</tbody>
</table>
Hybrid Periodical Flooding

- Hybrid Periodical Flooding (HPF)
  - The number of relay neighbors can be changed periodically based on a periodical function and the relay neighbors are selected based on multiple metrics in a hybrid way

\[ h = h_1 + h_2 + \ldots + h_t \quad t : \# \text{ of metrics} \]

\[ h_i = \left\lfloor h \times w_i \right\rfloor, \quad 1 \leq i \leq t \]
Performance Evaluation

Simulation setup
Physical topology: 10000 nodes
Logical topology: 1000 to 5000 nodes
Metric: communication cost(0.6), shared # of files(0.4)

Period function:

\[ f(n, TTL) = \begin{cases} \left\lfloor \frac{1}{2} n \right\rfloor, & \text{if } TTL \text{ is odd} \\ \left\lceil \frac{1}{4} n \right\rceil, & \text{if } TTL \text{ is even} \end{cases} \]
Performance Evaluation

Traffic comparison

![Graph showing traffic comparison for different methods (BFS, RIPF, DBFS, HPF) with request vs. normalized query cost on the y-axis and requested number of response results on the x-axis.](image-url)
Performance Evaluation

Response time comparison

![Graph showing response time comparison for BFS, RPF, DBFS, and HPF algorithms.](image)
Performance Evaluation

Coverage percentage comparison
Performance Evaluation

Coverage percentage comparison

[Graph showing nodes distribution against coverage size with two lines for DBFS and HPF]
Conclusion

- HPF improves the efficiency of blind flooding by retaining the advantages of statistics-based search mechanisms and by alleviating the partial coverage problem.
- Still have some work to do on the performance of response time.