Characterizing Selfishly Constructed Overlay Routing Networks

March 11, 2004
Byung-Gon Chun, Rodrigo Fonseca, Ion Stoica, and John Kubiatowicz
University of California, Berkeley
Content

• Motivation
• Network creation model
• Cases: simple and realistic
• Simulation results: topology, performance, and resilience
• Conclusion and future work
Motivation
Motivation
Network Creation Model

• Extension of the model of Fabrikant et al.
• Non-cooperative game of $n$ overlay nodes
  – Each node choose neighbors to set up links to minimize its cost
    \[
    C_i(s) = \alpha \sum_{j \in B(i)} t_j + \sum_{j=0}^{n-1} d_{G[s]}(i, j)
    \]
  – Each node reaches the stable state where no node can benefit by changing its links, while the other nodes keep their links unchanged.
Case Study (Simple)

- Physical topology: fully-connected topology with unit distance between nodes
- $d_{G}(i,j)$: number of hops

<table>
<thead>
<tr>
<th>Cost Model</th>
<th>Linking Cost ($t_j$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit</td>
<td>1</td>
</tr>
<tr>
<td>Exponential</td>
<td>$c_j \ (E(c_j)=1)$</td>
</tr>
<tr>
<td>Node-degree</td>
<td>degree(j)</td>
</tr>
</tbody>
</table>

- 20 overlay nodes
- Exhaustive search

[Fabrikant et al. 03]
Case Study (Realistic)

- Physical topology: transit-stub topology
- \( d_G(i,j) \): latency from physical topology
- Linking cost \((t_j)\): Unit with the degree bound (MaxDegree)
- 100 overlay nodes
- Randomized local search: similar to Narada ([Chu et al. 03])
Case Study (Realistic)

- Each node runs link addition and dropping.
- Link addition
  - Randomly select a node not in the neighbor set, compute latency, and get the linking cost
  - If its cost decreases with the selected node, add the link
- Link Dropping
  - For each node in the neighbor set, compute the cost without the node.
  - Pick up the node whose removal gives minimum cost.
  - If its cost decreases by dropping the neighbor, drop the link.
Simulation Results

• Widely different networks produced by selfish nodes
  – Simple scenario: complete graphs, k-regular graphs, k-core stars, trees
  – Realistic scenario: networks with exponential degree distribution or Pareto degree distribution

• Tradeoff between performance and resilience in the selfishly constructed networks
Topology (Simple)

<table>
<thead>
<tr>
<th>Linking Cost</th>
<th>1</th>
<th>5</th>
<th>10</th>
<th>60</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exponential</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nodedegree</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Topology (Realistic)

<table>
<thead>
<tr>
<th>Max Degree</th>
<th>$\alpha$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1000</td>
</tr>
<tr>
<td></td>
<td>2000</td>
</tr>
<tr>
<td></td>
<td>3000</td>
</tr>
<tr>
<td></td>
<td>5000</td>
</tr>
<tr>
<td></td>
<td>10000</td>
</tr>
</tbody>
</table>

![Graphs showing different network topologies for varying max degrees and alpha values.](image-url)
Performance / Resilience
(Realistic)

(a) Stretch

(b) Attack tolerance
(10% subject to attacks)
Degree Distribution (Realistic)
Conclusion

• Examine networks created by selfish nodes using a non-cooperative game model

• Show diverse networks produced by games when we vary $\alpha$, linking cost functions, and underlying physical topology.
  – Complete graphs to trees with different properties
  – Networks with node degree distribution whose tails vary from exponential to pareto.

• Show the tradeoff between performance and resilience when we vary parameters.
Future Work

• Variations of linking cost functions
• Relationship between the underlying topology and the produced overlay topologies
• Dynamic network
• Traffic into consideration
• Different cost metrics
• Algorithmic mechanism design
Questions?