Adaptation Strategies for Self-management of Tree Overlay Networks

Evangelos Pournaras, Martijn Warnier & Frances Brazier
Outline

• Context and Motivation

• AETOS: the Adaptive Epidemic Tree Overlay Service

• Self-management in AETOS

• Strategies for self-management

• Conclusions
Motivation

Hierarchical topologies ➔ Tree structures

- Aggregation
- Decision-making
- Search
- Information dissemination

Simple in principle
Dynamic Adaptive Systems Design

Adaptation Strategies for Self-management of Tree Overlay Networks

Motivation (cont.)

Distributed systems and tree overlays

- Node / link failures
- Congestions
- Attacks
- Heterogeneity

Sensitive in principle
Problem

Robustness
Minimization of the impact of failures in the topology

Self-organization
Nodes with local knowledge in dynamic environments

Application-dependence
Abstract application to self-organization requirements
AETOS

Adaptive Epidemic Tree Overlay Service

Agent-based

TU Delft

Adaptation Strategies for Self-management of Tree Overlay Networks
Application requirements abstraction

Optimization metrics

- Application-dependent
  - Robustness (rank)

- Application-independent

Node degree

- Max # of children

TU Delft Adaptation Strategies for Self-management of Tree Overlay Networks
Target topology

Optimization problem:
Sort nodes according to their robustness and max # of children
Dynamic Adaptive Systems Design

Adaptation Strategies for Self-management of Tree Overlay Networks

Architecture

Application

Robust Tree Overlay

AETOS Agent

Adaptive Tree Overlay Management (ATOM)

Connectivity Options

View Reconfiguration

Adaptive Rank-based Middleware Overlay Service (ARMOS)

Random Search Space

Peer Sampling Service (PSS)

TU Delft

Adaptation Strategies for Self-management of Tree Overlay Networks
Agent knowledge

3 type of views

Random View ➔ Potential Parent/Child View ➔ Tree View
Information flow

Local Self-organization

Random View
Candidate Parents
Candidate Children
Tree View

Local Robustness $r = 58$

Gossip
Tree Management

Adaptation Strategies for Self-management of Tree Overlay Networks
Local Agent Strategies

1. Collect Agent Information
2. Select Ranked Agents
3. Select Parent & Children
4. Request Connections
5a. Establish Connections
5b. Adapt

Outcome
Selected Parent & Children
Search Space
Views
## The eight Strategies

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Children</th>
<th>Parents</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>low</td>
<td>high</td>
</tr>
<tr>
<td>Presbyopic</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Myopic</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Humble</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Greedy</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Top-down</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Bottom-up</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Top</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Bottom</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

Adaptation Strategies for Self-management of Tree Overlay Networks
**Example Strategy: Greedy**

Try to get the best parent and children (Optimistic)

Exclude parent/children that are worse than already available

If the best parent/children are not available go for second best
Example Strategy: Greedy

Re(initialize) if so required
## The eight Strategies

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Children</th>
<th></th>
<th>Parents</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>low</td>
<td>high</td>
<td>low</td>
</tr>
<tr>
<td>Presbyopic</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Myopic</td>
<td></td>
<td>✓ ✓ ✓</td>
<td></td>
</tr>
<tr>
<td>Humble</td>
<td>✓</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Greedy</td>
<td></td>
<td>✓ ✓ ✓</td>
<td></td>
</tr>
<tr>
<td>Top-down</td>
<td></td>
<td>✓ ✓ ✓</td>
<td></td>
</tr>
<tr>
<td>Bottom-up</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Top</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bottom</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Adaptation Strategies for Self-management of Tree Overlay Networks
Experiments

- 1500 Agents
- Random rank for each agent
- Max 4 children per agent
- Bootstrap from (random) ring topology
## Results

<table>
<thead>
<tr>
<th></th>
<th>Epoch 50</th>
<th>Epoch 150</th>
<th>Epoch 250</th>
<th>Epoch 350</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dynamic</strong></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
</tr>
<tr>
<td><strong>Adaptive</strong></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
</tr>
<tr>
<td><strong>Systems Design</strong></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
</tr>
<tr>
<td><strong>Adaptation</strong></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
</tr>
<tr>
<td><strong>Strategies</strong></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
</tr>
<tr>
<td><strong>for Self-management</strong></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
</tr>
<tr>
<td><strong>Tree Overlay Networks</strong></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
<td><img src="image" alt="Graph" /></td>
</tr>
</tbody>
</table>

Adaptation Strategies for Self-management of Tree Overlay Networks

---

TU Delft
Results

<table>
<thead>
<tr>
<th>Top-down</th>
<th>Epoch 50</th>
<th>Epoch 150</th>
<th>Epoch 250</th>
<th>Epoch 350</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bottom-up</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Top</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bottom</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Adaptation Strategies for Self-management of Tree Overlay Networks
Conclusions & Future Work

- Building & maintaining hierarchical structures in distributed environments is challenging
- Strategies provide trade off, communication vs connectivity or connectedness vs robustness
- 3-layer architecture:
  - Bottom: randomness-> proactive robustness
  - Middle: proximity-> reconfigurable knowledge
  - Top: connectivity-> reactivity

- Future Work: Mixing strategies, Local vs Global Strategies
- Further large-scale experimentation in dynamic settings, e.g. changing rank values
Contact

Evangelos Pournaras (e.pournaras@tudelft.nl)
Martijn Warnier (m.e.warnier@tudelft.nl)

Dynamic Adaptive Systems Design
www.systemsdesign.tbm.tudelft.nl