Adaptive Agent-based Self-organization for Robust Hierarchical Topologies

International Conference on Adaptive and Intelligent Systems

Evangelos Pournaras, Martijn Warnier, Frances M.T. Brazier

Evangelos Pournaras, September 2009
Motivation

Hierarchical topologies ➔ Tree structures

- Aggregation
- Decision-making
- Search
- Information dissemination

Simple in principle
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
Propose AETOS

The Adaptive Epidemic Tree Overlay Service

Evangelos Pournaras, September 2009
Approach

Agent-based

Evangelos Pournaras, September 2009
Application requirements abstraction

Optimization metrics

Node degree

Application-dependent

Application-independent

Robustness (rank)

Max # of children

Evangelos Pournaras, September 2009
Target topology

Optimization problem:

Sort nodes according to their robustness and max # of children
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)

Evangelos Pournaras, September 2009
Agent knowledge

3 type of **views**

Random View ➔ Myopic View ➔ Tree View
Information flow

Local Self-organization

Random View
Candidate Parents
Candidate Children
Tree View

Evangelos Pournaras, September 2009
Tree Management

Parent request
Acknowledgment
Child request
Rejection
Removal
Myopic View Reconfigurations

Downgrade reconfiguration (rejection, removal)

Agent picks candidates with lower robustness than the ones it tried before
Example
Message Overhead

ATOM (top layer)  ARMOS (middle layer)  PSS (bottom layer)

# of Messages

Round

Evangelos Pournaras, September 2009
Conclusions & Future Work

- **Building & maintaining** hierarchical structures in distributed environments *is challenging*
- Importance: **Robustness, self-organization, application-independence**
- 3-layer architecture:
  - **Bottom**: randomness -> **proactive robustness**
  - **Middle**: proximity -> **reconfigurable knowledge**
  - **Top**: connectivity -> **reactivity**

- Further large-scale experimentation in dynamic settings, e.g. changing rank values
- Test in different applications, e.g. energy management, application-level multicast
Questions?

Evangelos Pournaras, September 2009