Nokia Siemens Networks

An Autonomic Framework for Self-Optimizing Next Generation Mobile Networks

David Soldani, Gordon Alford, Flavio Parodi and Mikko Kylväjä

Nokia Siemens Networks

Research, Technology and Platforms

IEEE IWAS07, June 18th 2007

Outline

Introduction

Autonomic framework

Case studies and simulation assumptions

Simulation results

Conclusions and further studies

Introduction

- Generic framework for self-optimizing mobile radio access networks
- Autonomic system assessment in terms of
  - Coping with unanticipated changes in the environment
  - Network performance (within limits)
- Level of autonomicity is evaluated through two case studies
  - Neighbor cell list (NCL) prune
  - Cell removal and redeployment

Autonomic framework

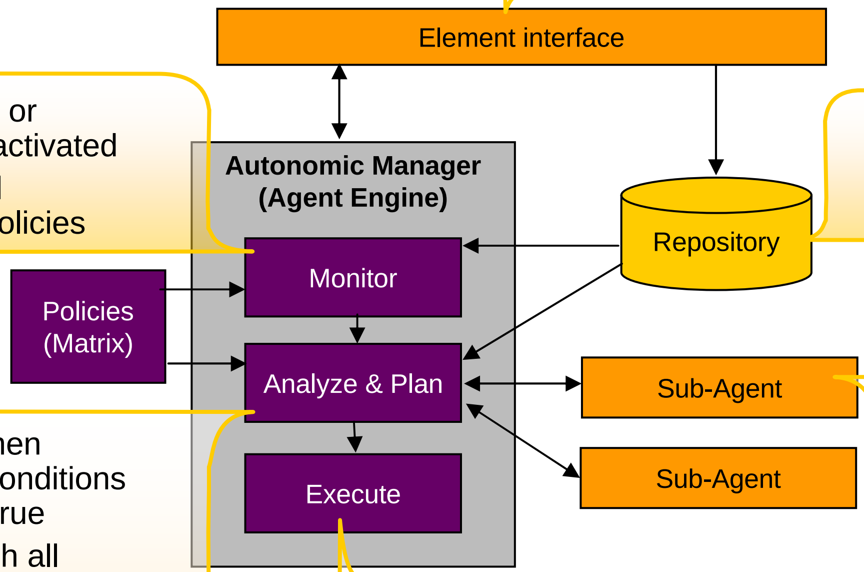
- Stable and flexible structure with various self-management functionalities
- The behavior is governed by a *policy matrix*
  - Monitoring policies (conditions) for **Monitoring** process
  - Analysis policies (conditions & functions) for **Analyze & Plan** process

Condition	**Function**
Monitoring Condition	-
Analysis Condition	*Analysis Function*

Autonomic Manager functional elements

- <u>Application</u> contained within a logical element of the autonomic system with <u>centralized or distributed architecture</u>

- To/from other elements
- To/from other agent engines



- Event based or periodically activated
- Evaluates all monitoring policies

- Measurements
- Parameters
- …

- Activated when monitoring conditions evaluate to true
- Goes through all analysis policies
- Initiates the functions (sub-agents) when condition evaluates to true
- Passes data to the Execute process

- Executes proposed actions
- Checks for and resolves conflicts
- Executes those actions that remain

- Scalability (content, multi-vendor and different elements)
- Handle different algorithmic analysis or functions using data from the repository provided by the AM
- Return a list of actions to the AM: E.g. **Add**, **Prune**, **Generate** and **Initialize**

Case studies and simulation assumptions

- Assess adaptability, time to react, stability and sensitivity of the *distributed* optimization function
- Case 1: **NCL prune**
  - Delete unnecessary adjacent cells based on predefined conditions

Condition	Function
HOshare < 1%	*Prune*

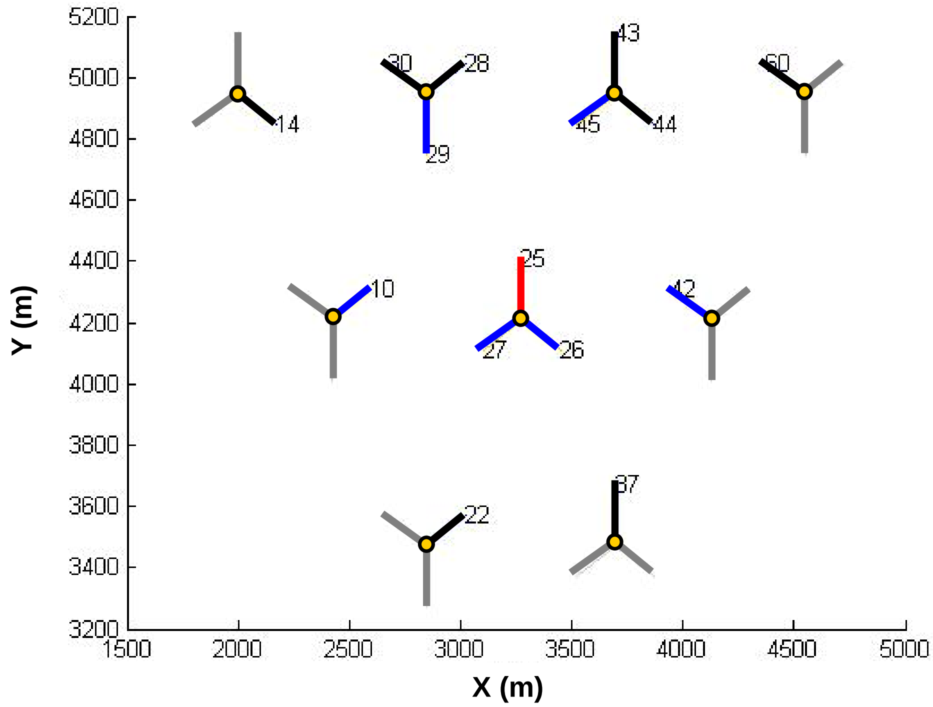
- Case 2: **cell removal and redeployment**
  - React when a cell is removed/re-deployed in the network
  - Adapt the NCLs of the affected cells to the new system configurations

Condition	Function
Cell state = off	*Prune*
Previous Cell state = off & *Cell state* = on	*Generate*
HOshare < 1%	*Prune*

Nokia Siemens Networks

Simulation environment

- Simple mobility with reflection on the borders of the map
- 300 terminals uniformly randomly distributed
- 25 base stations with 3 cells each (75 cells in total)
- Two scenarios: dense and sparse



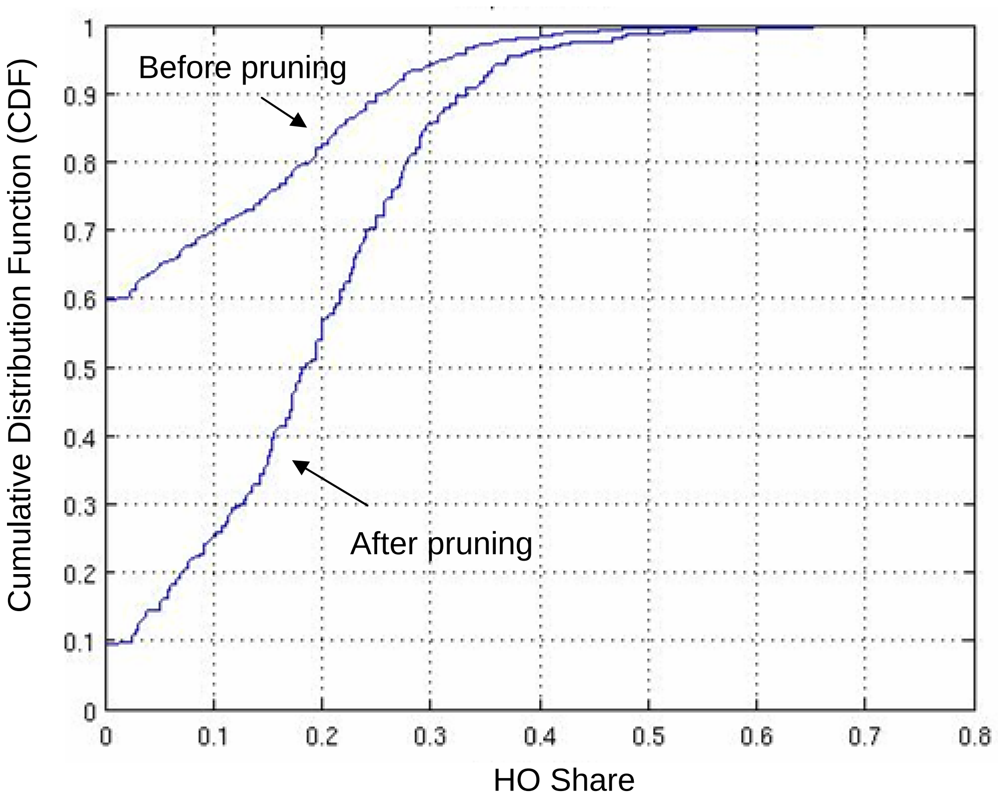
Case 2:

- **Generated NCL of Cell 25**: Cell **10, 14, 22, 26, 27, 28, 29, 30, 37, 42, 43, 44, 45**, and **60** (*Cluster Under Test*)
- **Pruned NCL of Cell 25:** Cell **10, 26, 27, 29, 42,** and **45**

Nokia Siemens Networks

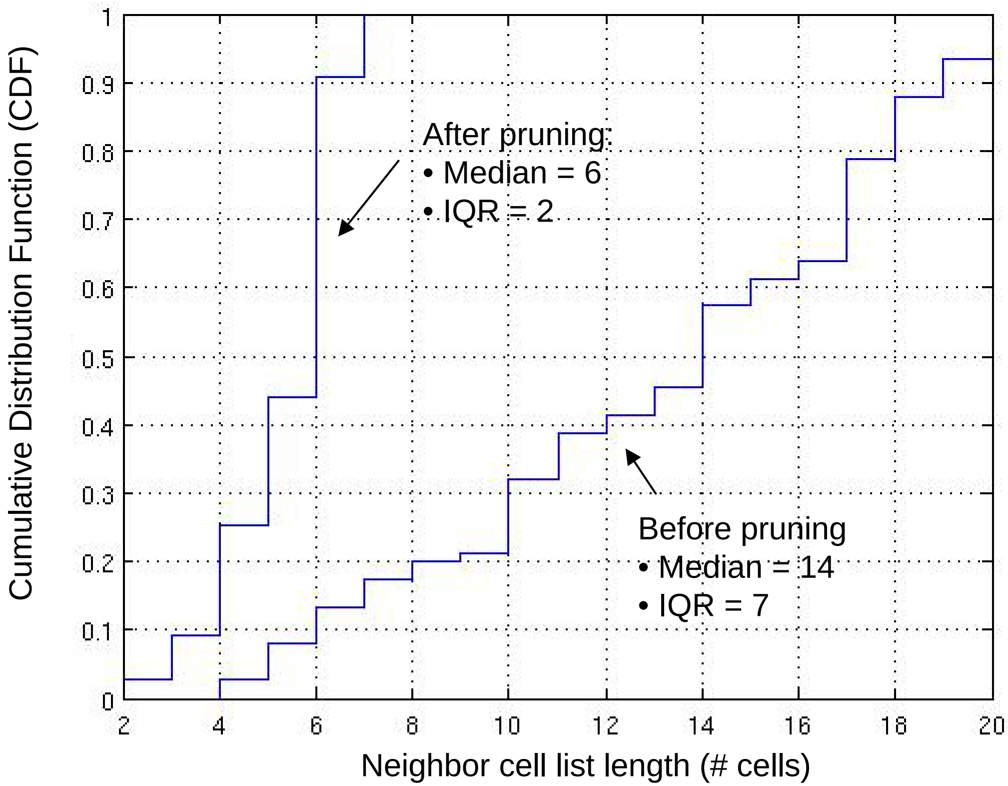
Case 1 - results: NCL Prune (1/3)

- This proves the ability of the AM to delete unnecessary adjacencies based on the input policy



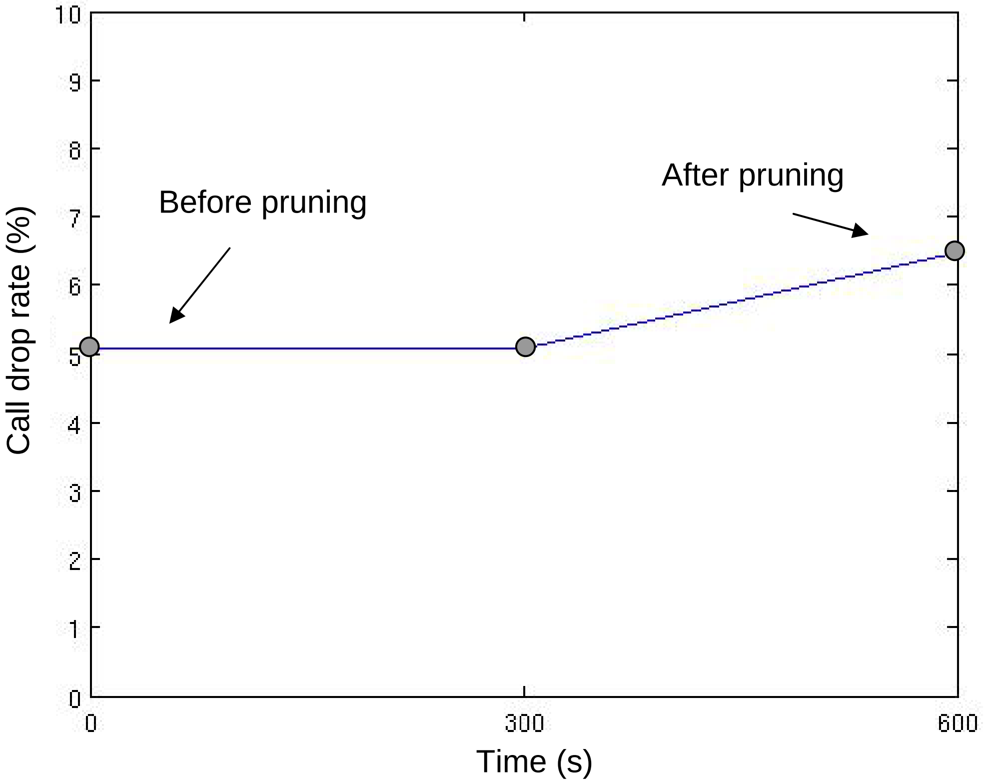
Case 1 - results: NCL Prune (2/3)

- Wasted resources are detected by the AM and every cell on the NCL with less than 1% of HO share is pruned from the list



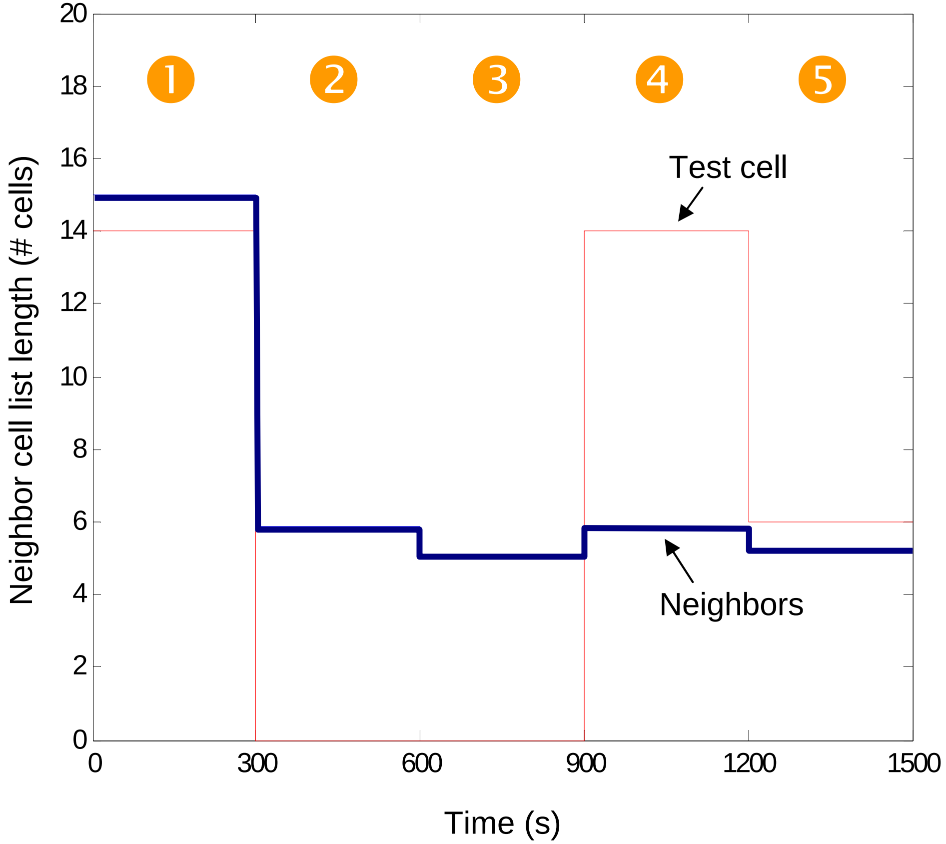
Case 1 - results: NCL Prune (3/3)

- The deterioration in performance can be eliminated by adjusting the policies (conditions and/or functions)



Case 2 - results: Cell removal & redeployment (1/2)

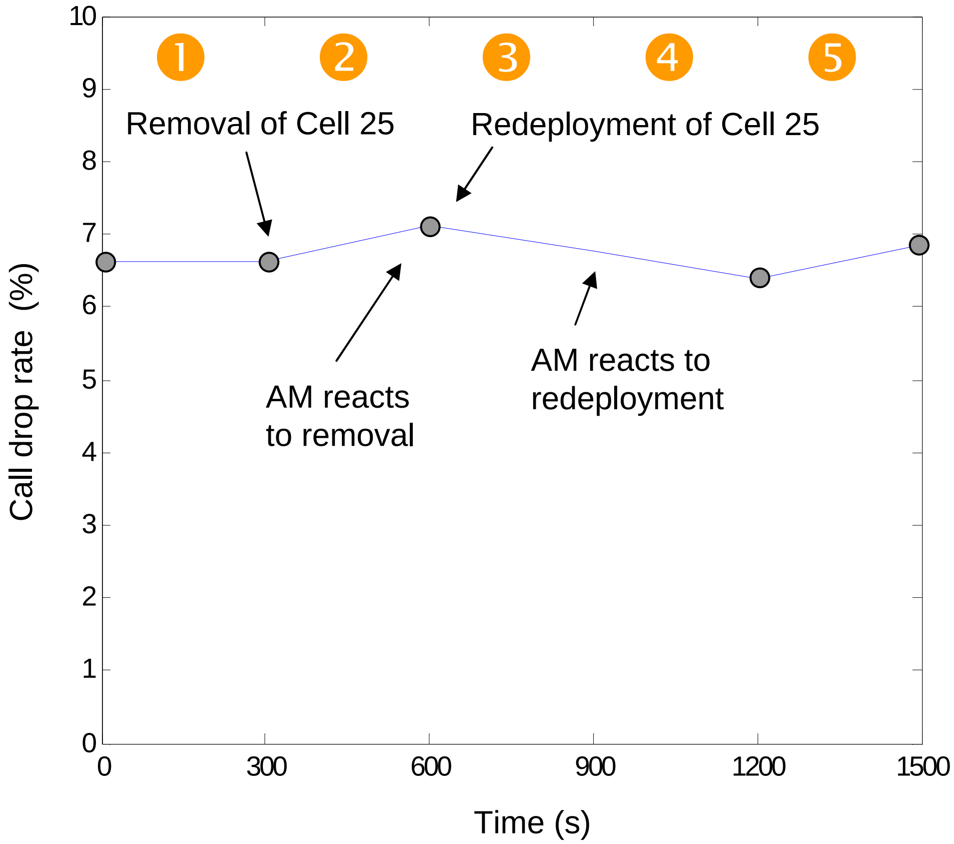
- Step by step representation of how the autonomic system copes with unexpected changes in the environment



1. Before pruning
2. When the neighborhood is pruned & Cell 25 is removed
3. When Cell removal is detected (⇨ Prune) & Cell 25 is redeployed
4. When Cell redeployment is detected (⇨ Generate & bidirectional enforcement)
5. After pruning

Case 2 - results: Cell removal & redeployment (2/2)

- Course of the call drop rate (measured in the cluster under test during the process)



1. Before pruning
2. When the Cell 25 is removed & neighborhood is pruned
3. When Cell removal is detected (⇨ Prune) & Cell 25 is redeployed
4. When Cell redeployment is detected (⇨ Generate & bidirectional enforcement)
5. After pruning

Conclusions and further studies

- Traditional O&M functionality may be seen as an integral part of this concept
- Solution scalable to autonomic functionalities in a real wireless networks and a multi-vendor environment
- System stability depends on the combination of policies and sub-agents
- It is for further study
  - "Plug and play" functionality for self-configuration
  - Performance in a real wireless network
  - Machine learning