

# Software Updates for Sensor Networks in Changing Process Environments

Jan Blumenthal, Steve Dübel, and Dirk Timmermann  
University of Rostock, Germany

## Situation

- Huge distributed sensor network
- No base infrastructure
- Maintenance free

## Challenge

- How to organize
- Process changes
  - Change of evaluation algorithms
  - Software updates without recollection of sensor nodes?

## OTA Flashing Protocol

### 1. Update Phase

- Segmenting of new software in pages
- Initiate update by a single node (base station)
- Transmitting and flashing of pages at receiver
- Forwarding of packets after waiting time depending on
  - Distance to sender
  - Randomly generated time
- Forwarding stopped if
  - Pages received plurally
  - Distance to sender smaller than  $d_{threshold}$

Update

Timeout

### 3. Data Collection Phase

- A node collects, processes, and forwards data
- Neighbors
  - Detect obsolete applications
  - Transmit own applications to force update of node
- Node updates software

Timeout

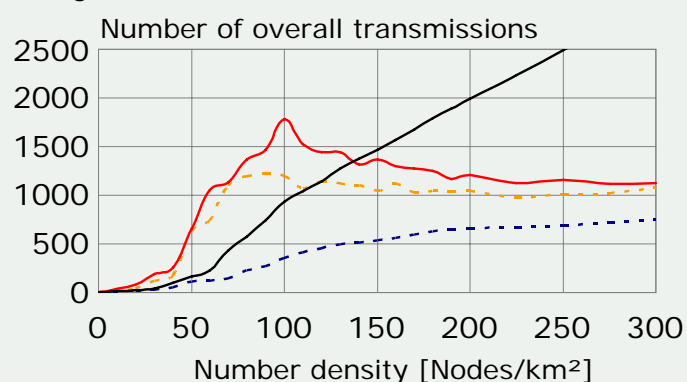
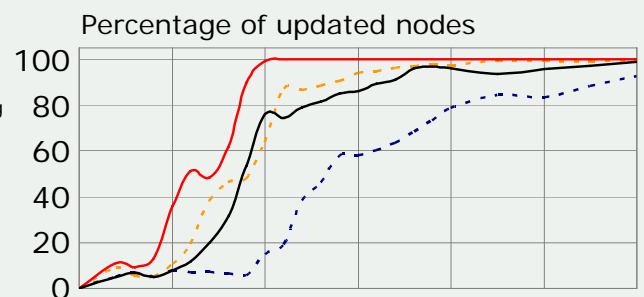
### 2. Correction Phase

- Isolated nodes may contain incomplete application
- Requesting missing pages by neighbors
- Neighbors transmit requested pages to nodes

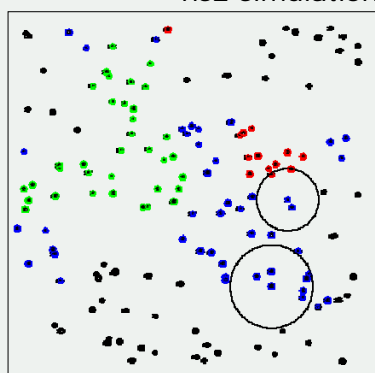
## Features

- Incoming nodes updated automatically
- Handles obstacles
- Memory efficient
- Huge coverage
- Small overlaps
- Decentral
- Scalable
- Robust

- OTA flashing protocol
- Without correction phase
- Without enforcing updates
- Updating via simple flooding



ns2 simulation



- Node in "Data Collection Phase"
- Node in "Update Phase"
- Node in "Correction Phase"
- Node updated

Status

Solution

Results