IEEE-IECON 2009
THE 35th ANNUAL CONFERENCE OF THE IEEE INDUSTRIAL ELECTRONICS SOCIETY
November 3-5, 2009, Porto, Portugal

Convex Groups for Self-organizing Multi-sink Wireless Sensor Networks

Olga Saukh
Structural Engineering Research Laboratory, EMPA, Switzerland
olga.saukh@empa.ch
Robert Sauter
Pedro José Marrón
University of Bonn and Fraunhofer IAIS, Germany

MOTIVATION

- Efficient access of MANET devices to context information provided by a WSN with the help of gateway nodes
- Goal:
  - Reduction of the message overhead and the overall energy consumption of the WSN
- Challenges:
  - Scoping of query dissemination to areas of interest
  - Efficient network reconfiguration due to topology changes
  - Efficient handling of node and especially sink mobility
- Approach **“Convex groups”** - encapsulates coverage information of the WSN
  - Convex group for each subtree of the routing tree to the nearest sink
  - Changes in the routing tree, e.g. due to node mobility, require recalculation only of a limited number of convex groups

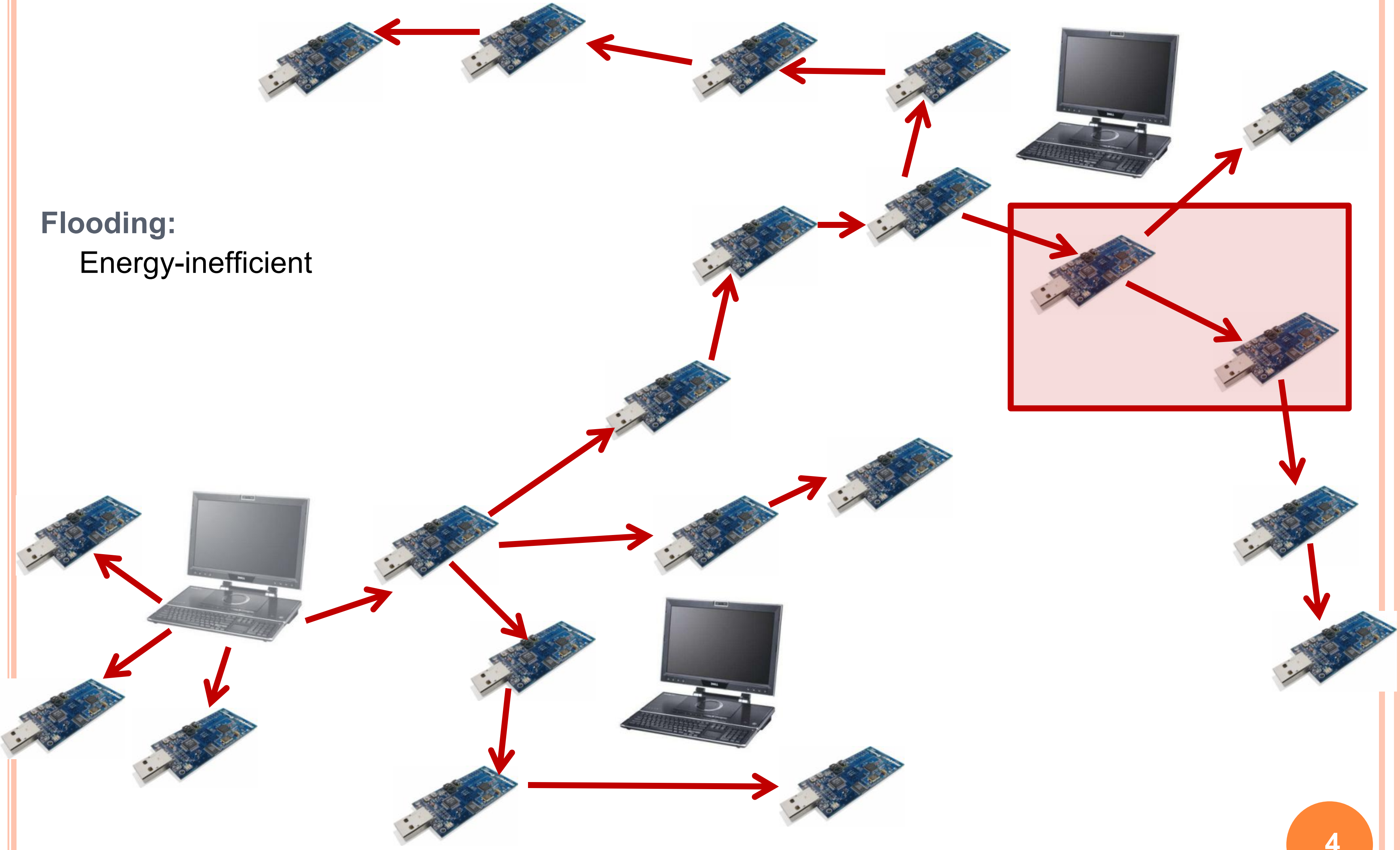
Application Example: "AWARE"

- **Goal:** Development of a platform for the cooperation of autonomous unmanned aerial vehicles (UAVs), sensor networks and video cameras

Data Dissemination in Multi-sink Scenarios

Flooding:

Energy-inefficient



Gateway Cooperation

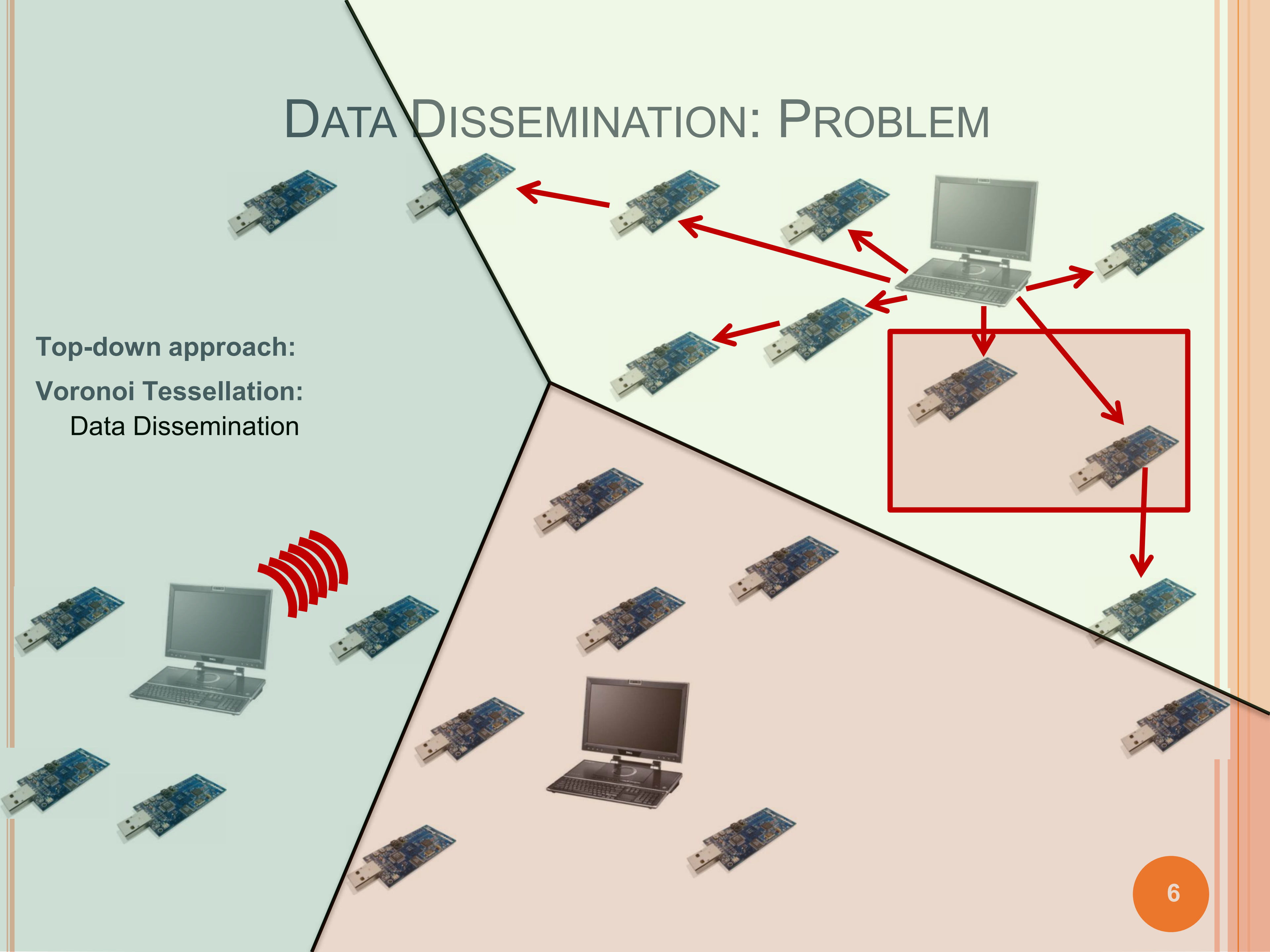
- **Top-down** approaches
  - Gateway disseminates information about its area of responsibility
  - \+ Simple for sensor nodes
  - – Reachability problems
  - – No support for gateway mobility
  - E.g. Voronoi tessellation
- **Bottom-up** approaches
  - The sensor nodes define area of responsibility for every gateway
  - \+ Solves reachability problems
  - \+ Support for node and gateway mobility
  - – More complex for sensor nodes
  - E.g. **Convex groups**

Data Dissemination: Problem

Top-down approach:

Voronoi Tessellation:

Data Dissemination

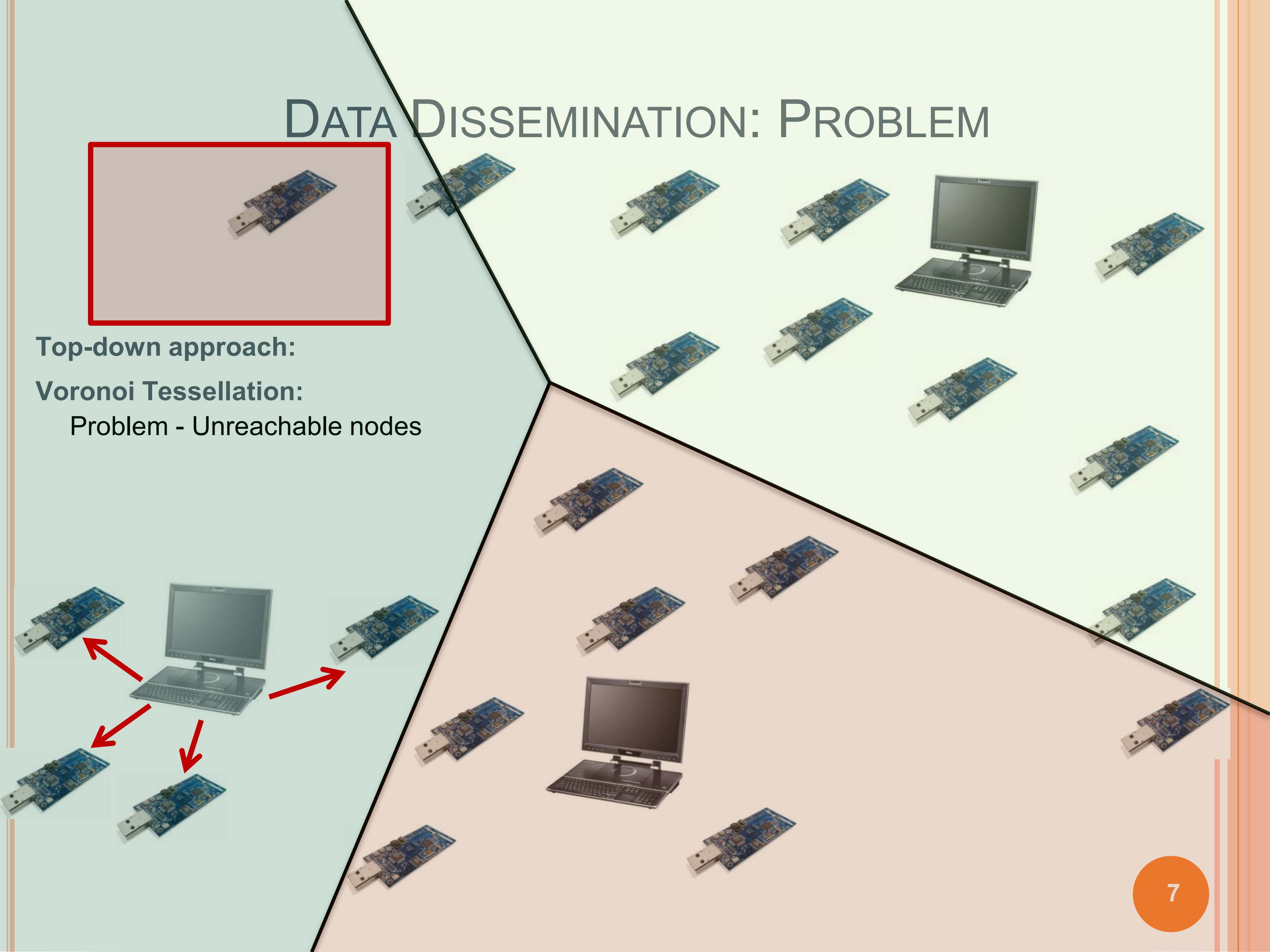


DATA DISSEMINATION: PROBLEM

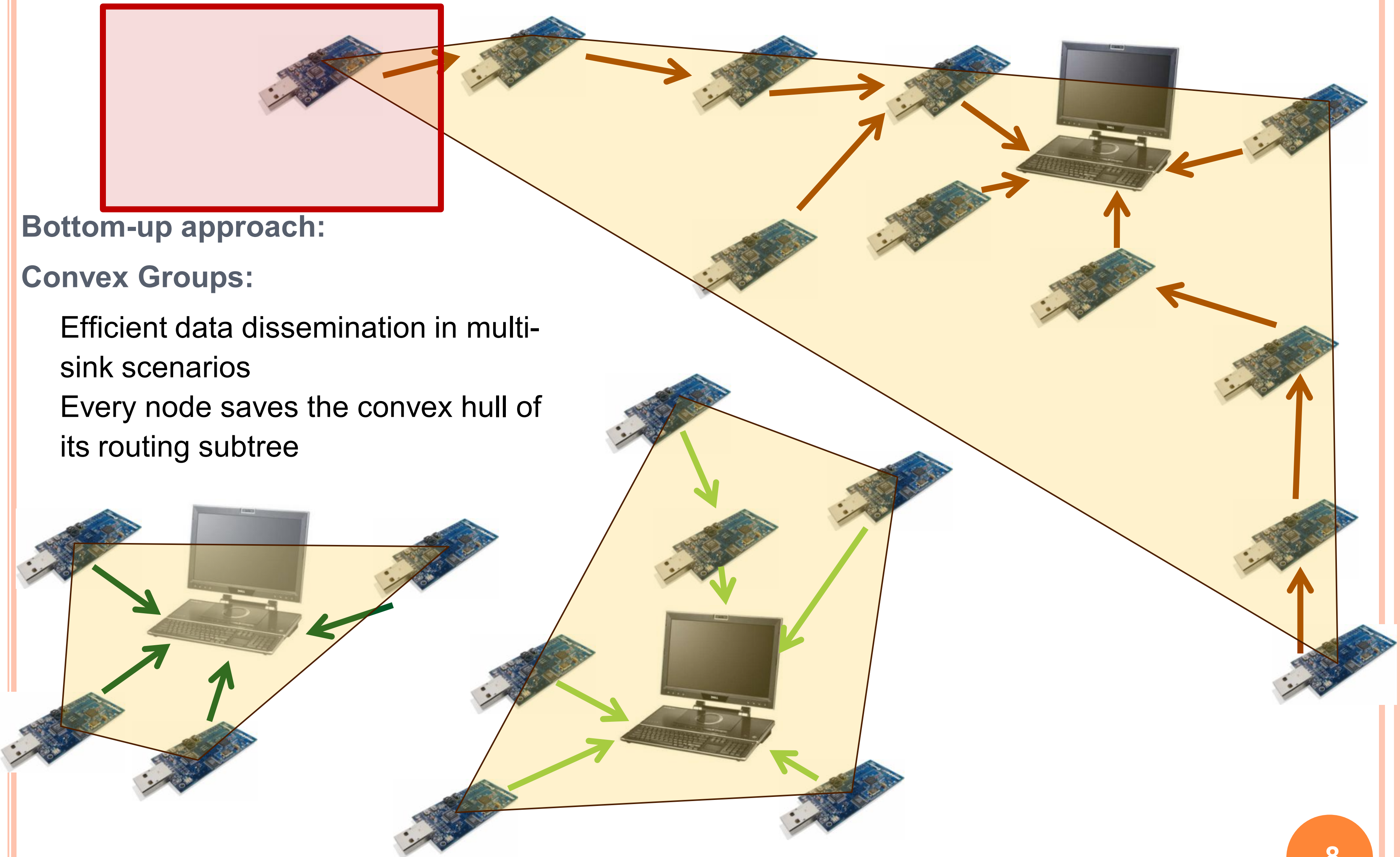
Top-down approach:

Voronoi Tessellation:

Problem - Unreachable nodes



DATA DISSEMINATION: APPROACH



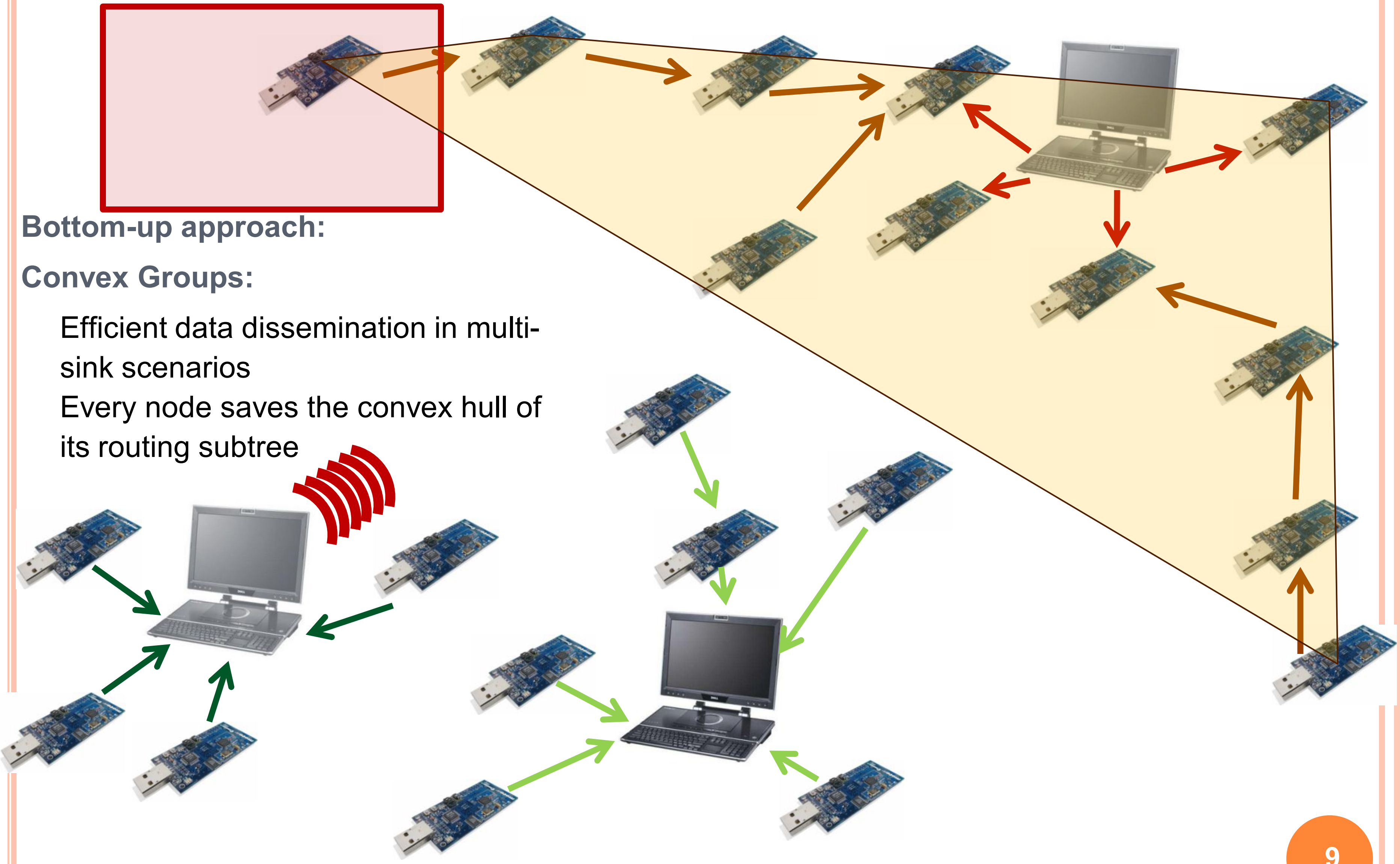
Bottom-up approach:

Convex Groups:

Efficient data dissemination in multi-sink scenarios

Every node saves the convex hull of its routing subtree

DATA DISSEMINATION: APPROACH

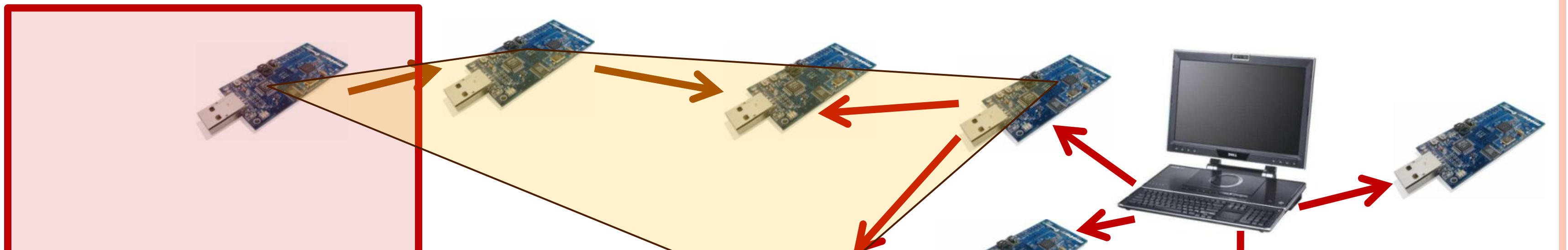


Bottom-up approach:

Convex Groups:

- Efficient data dissemination in multi-sink scenarios
- Every node saves the convex hull of its routing subtree

Data Dissemination: Approach

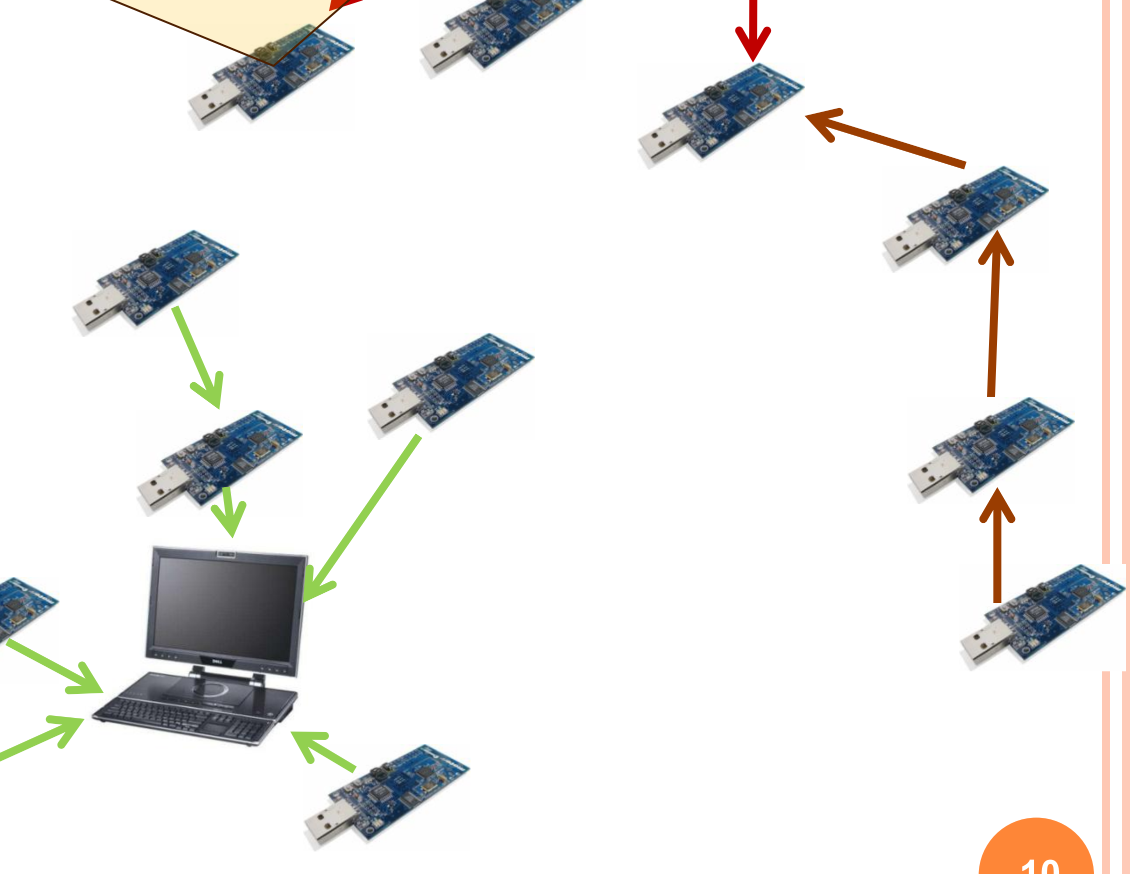


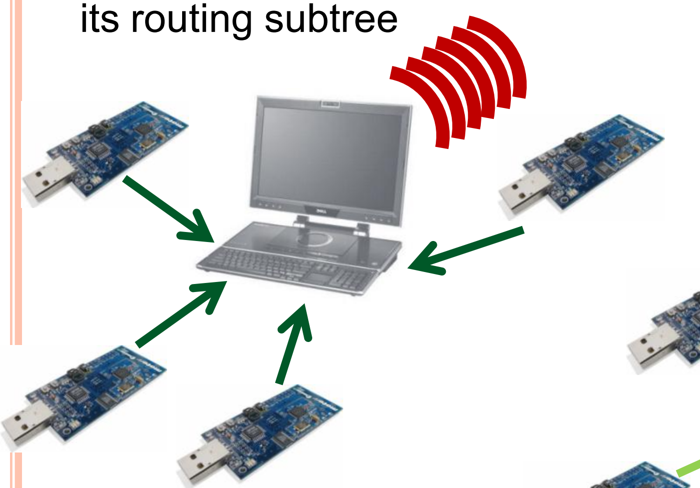
Bottom-up approach:

Convex Groups:

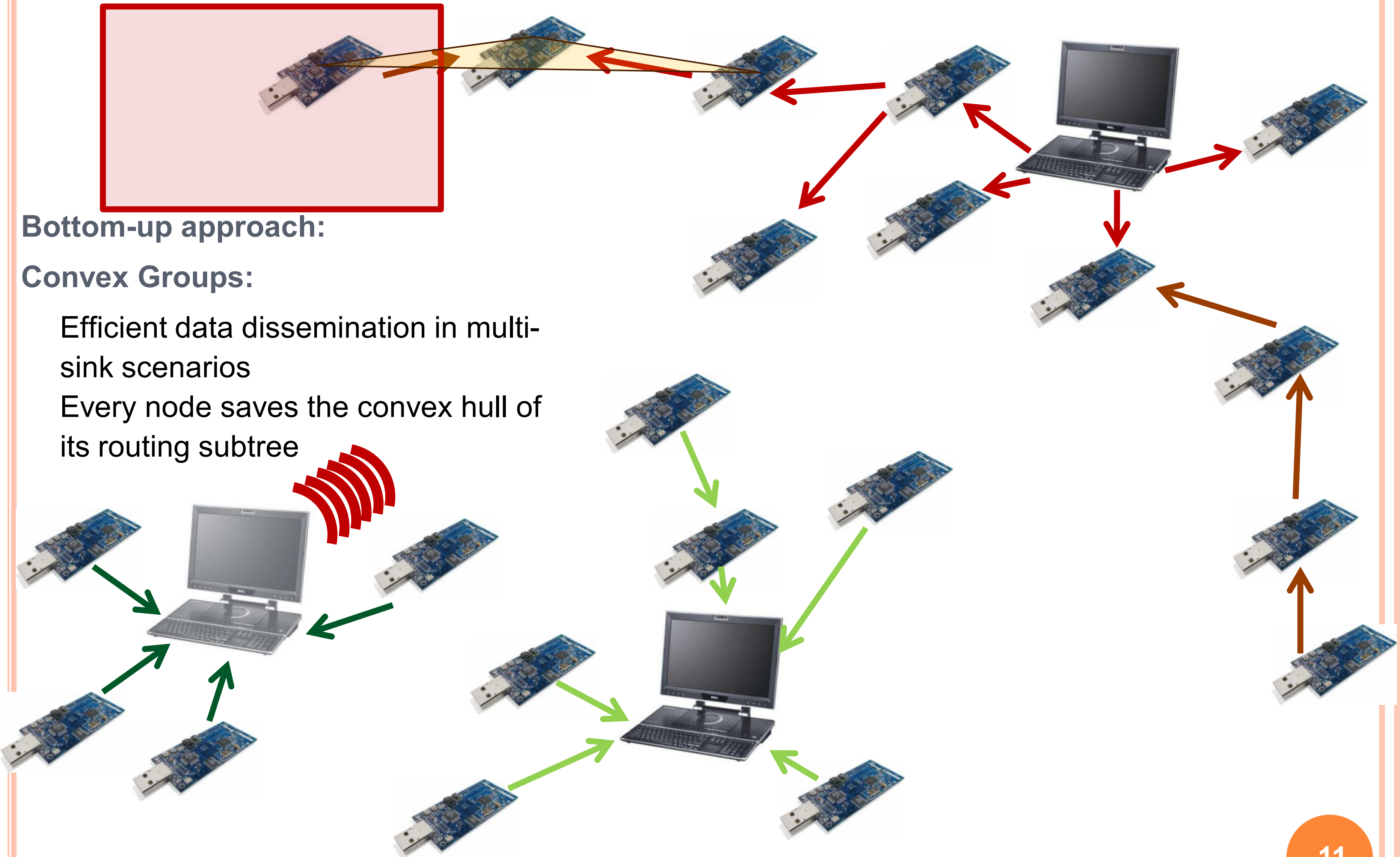
Efficient data dissemination in multi-sink scenarios

Every node saves the convex hull of its routing subtree





Data Dissemination: Approach



Bottom-up approach:

Convex Groups:

Efficient data dissemination in multi-sink scenarios

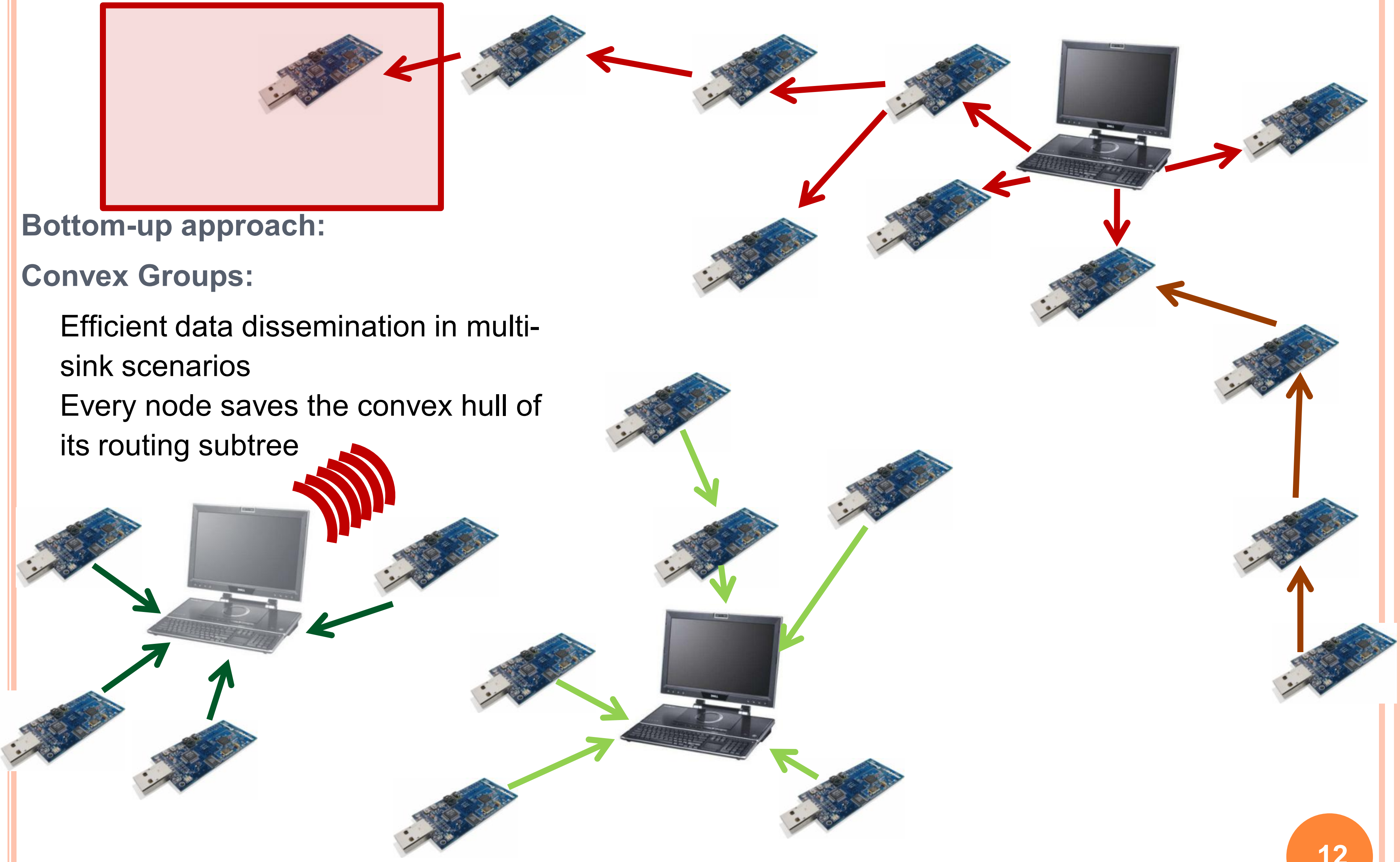
Every node saves the convex hull of its routing subtree

Data Dissemination: Approach

Bottom-up approach:

Convex Groups:

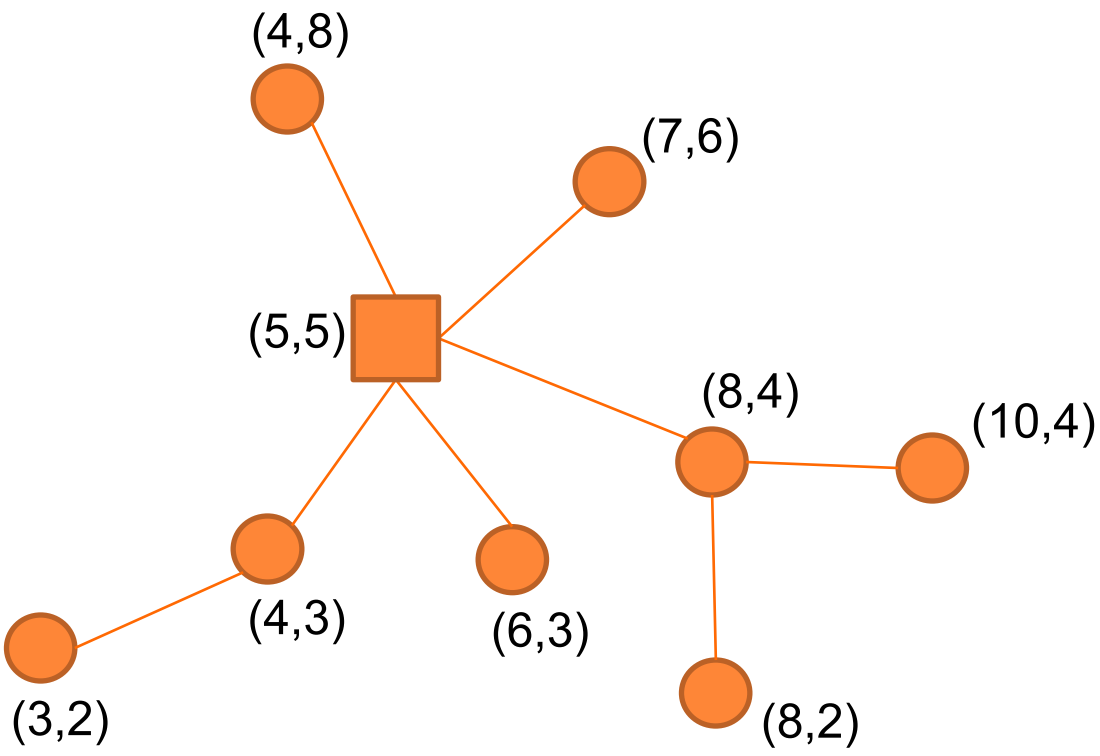
- Efficient data dissemination in multi-sink scenarios
- Every node saves the convex hull of its routing subtree



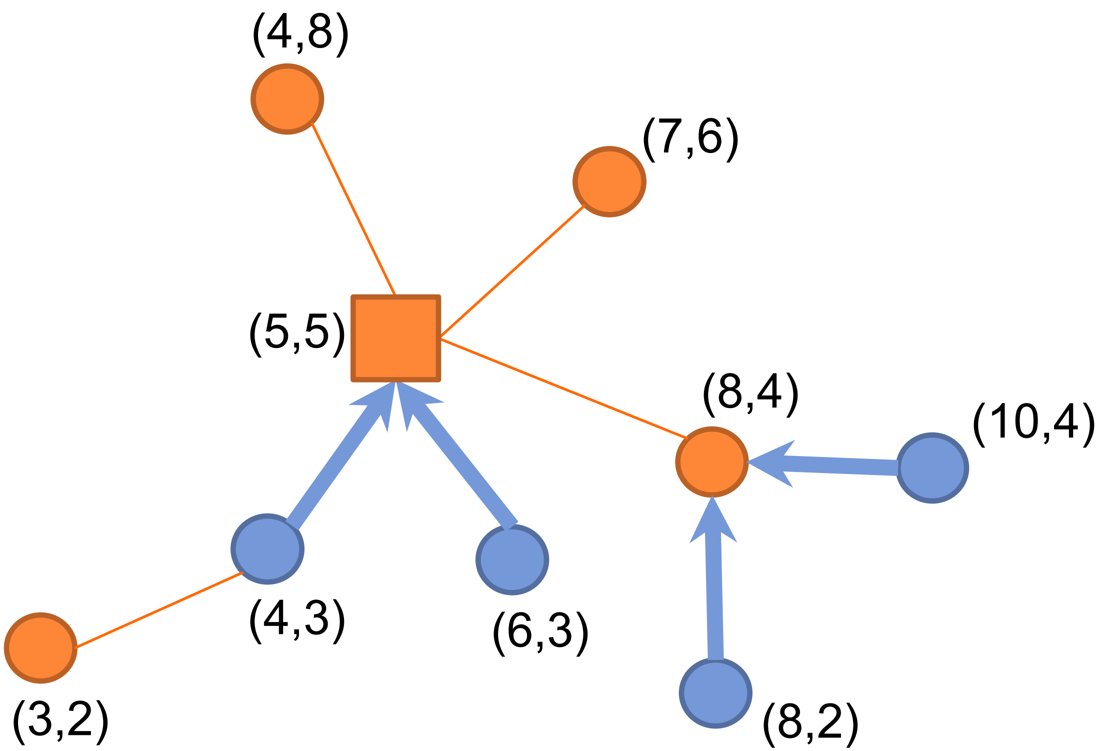
Establishment of Convex Groups

- Hierarchical convex groups along the routing tree to abstract the target area
- Consider a set of s nodes V[s] in V
- A polygon P[n] defines a **convex group** over V[s] if it is a convex polygon of n vertices that covers all s nodes in V[s].
  - If there is no limit on the number of vertices the polygon P[n] may comprise, then the minimum P[n] coincides with the convex hull C[m] built over the set of nodes V[s] (n = m).

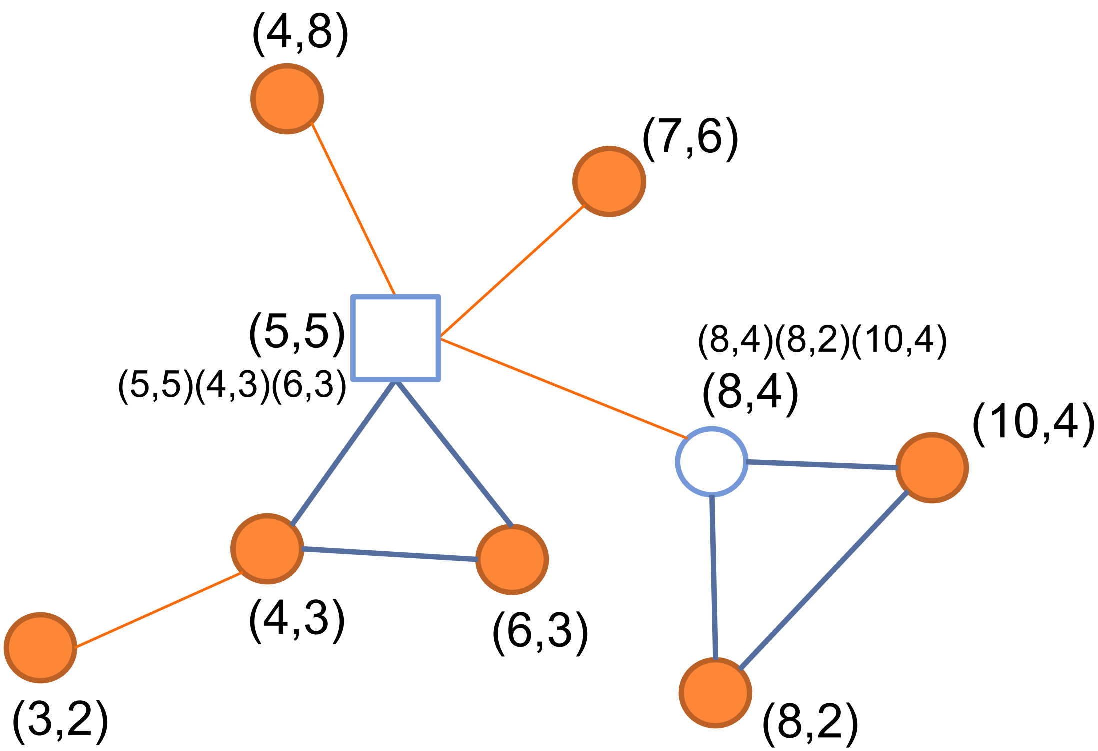
BUILDING CONVEX GROUPS



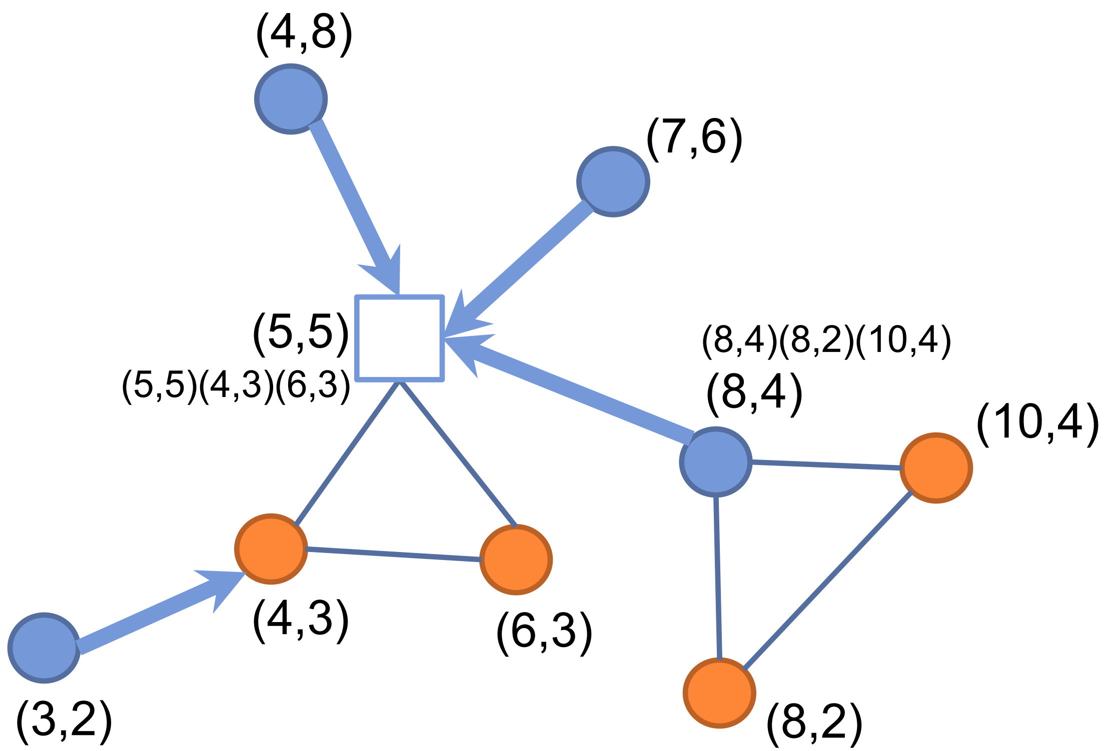
Building Convex Groups



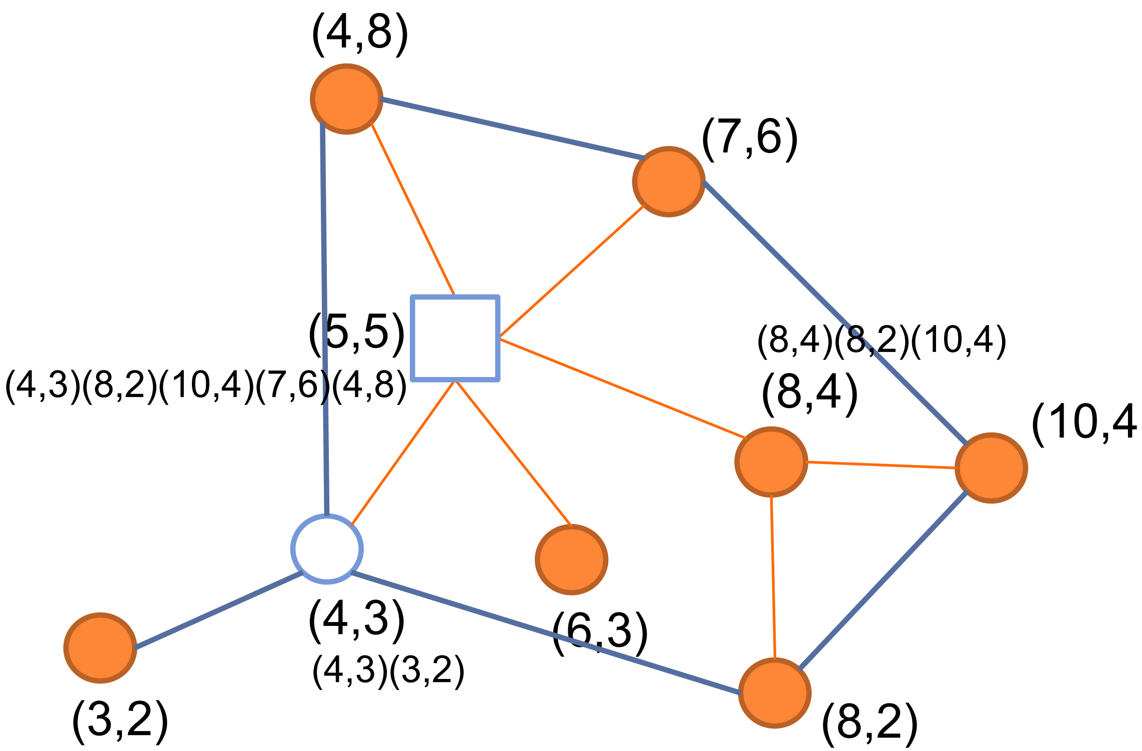
Building Convex Groups



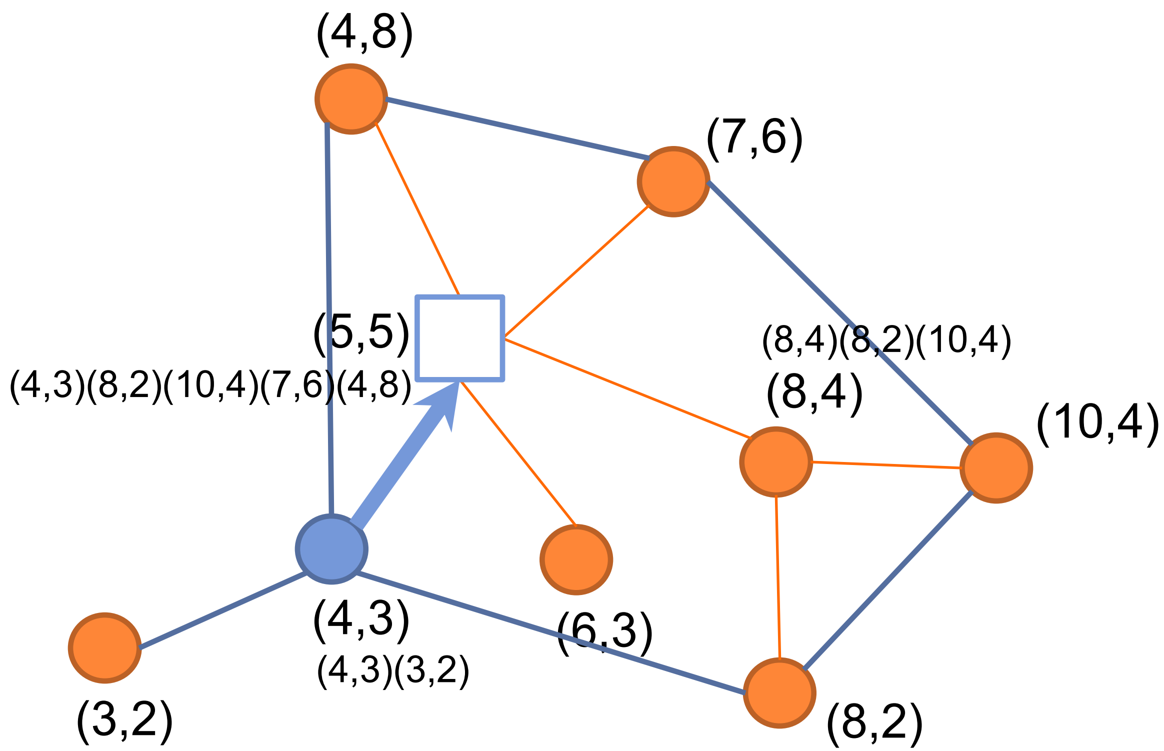
Building Convex Groups



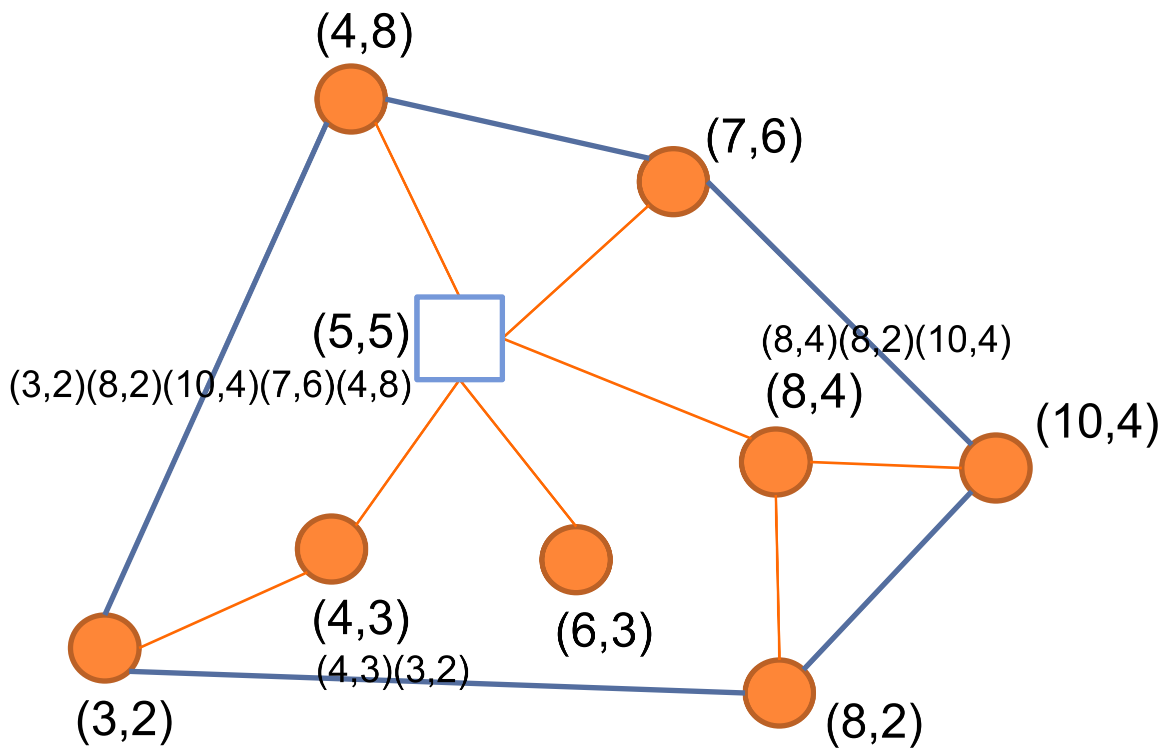
Building Convex Groups



BUILDING CONVEX GROUPS

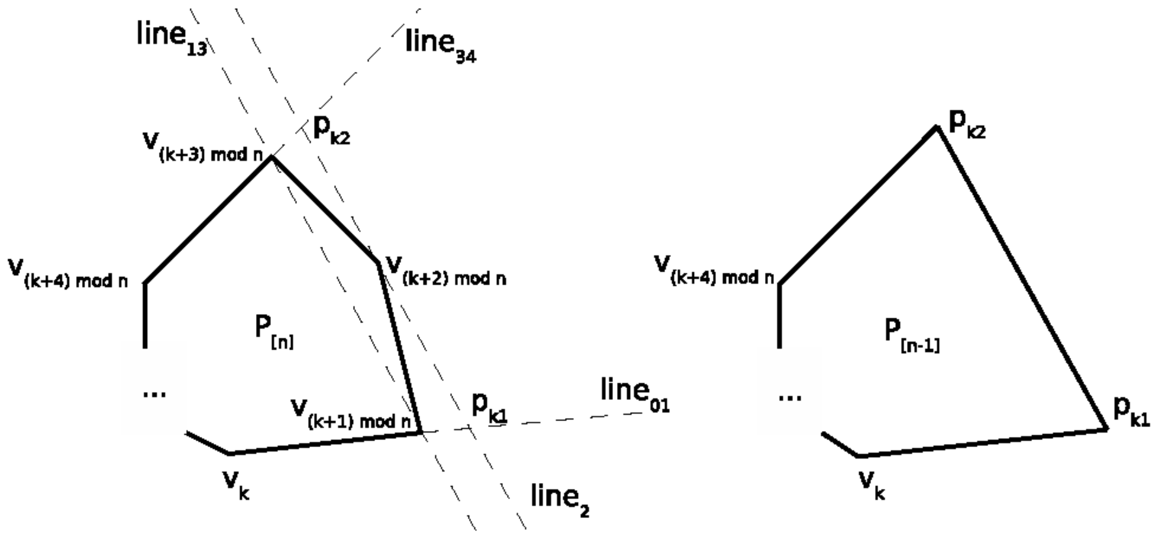


BUILDING CONVEX GROUPS



Compression of Convex Groups

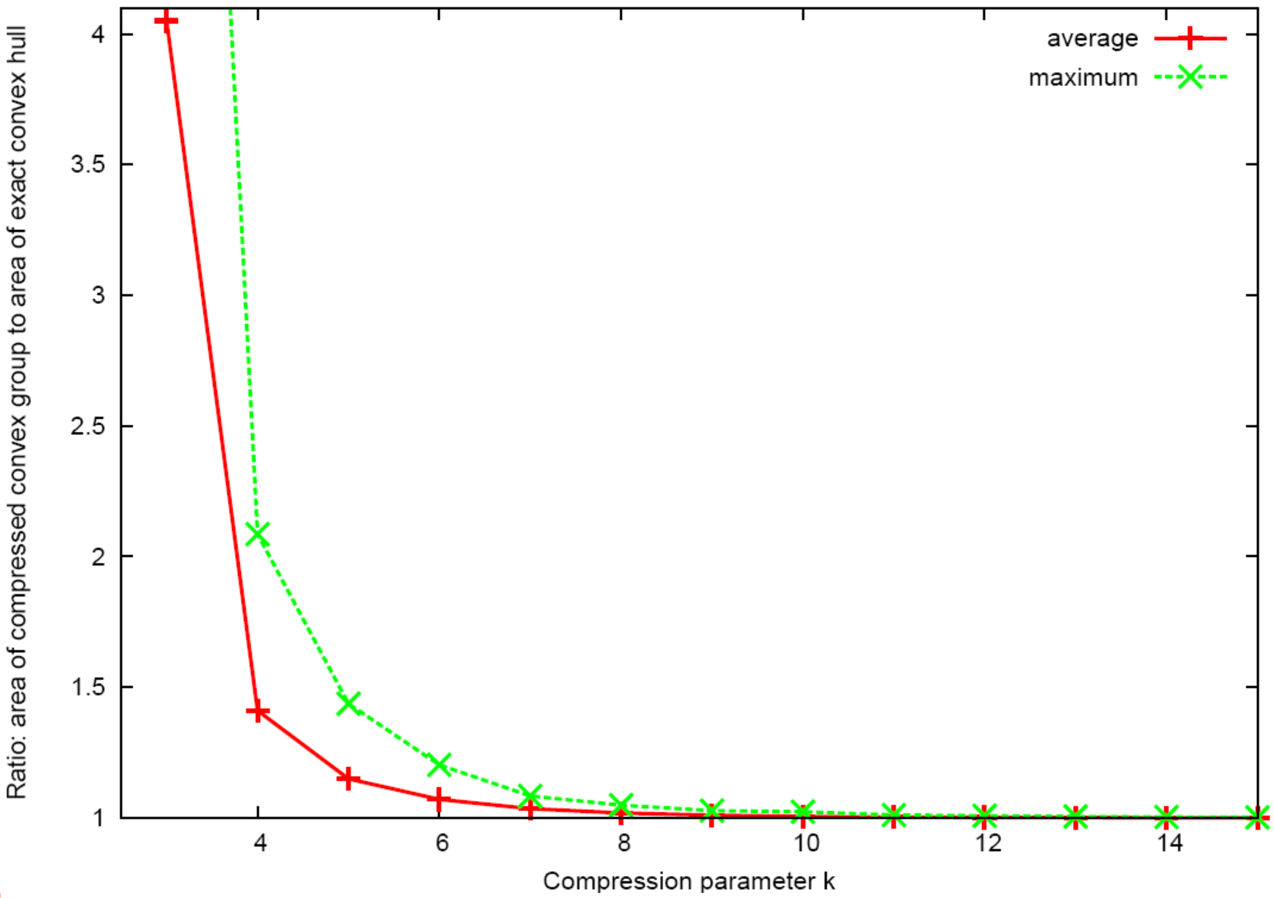
- It is possible to define a **compressed polygon** P[n′] which contains the convex hull but comprises less vertices (n′ < m) and which has, therefore, a larger area.
  - This polygon might consist of vertices with coordinates different from any of the actual sensor nodes.
  - Compression may be used to limit the amount of storage needed for polygon descriptions
- **Compression operator** c: P[n] →P[n−1] which converts a given convex polygon of n vertices into a convex polygon of n−1 vertices which contains P[n].



Simulation Setup

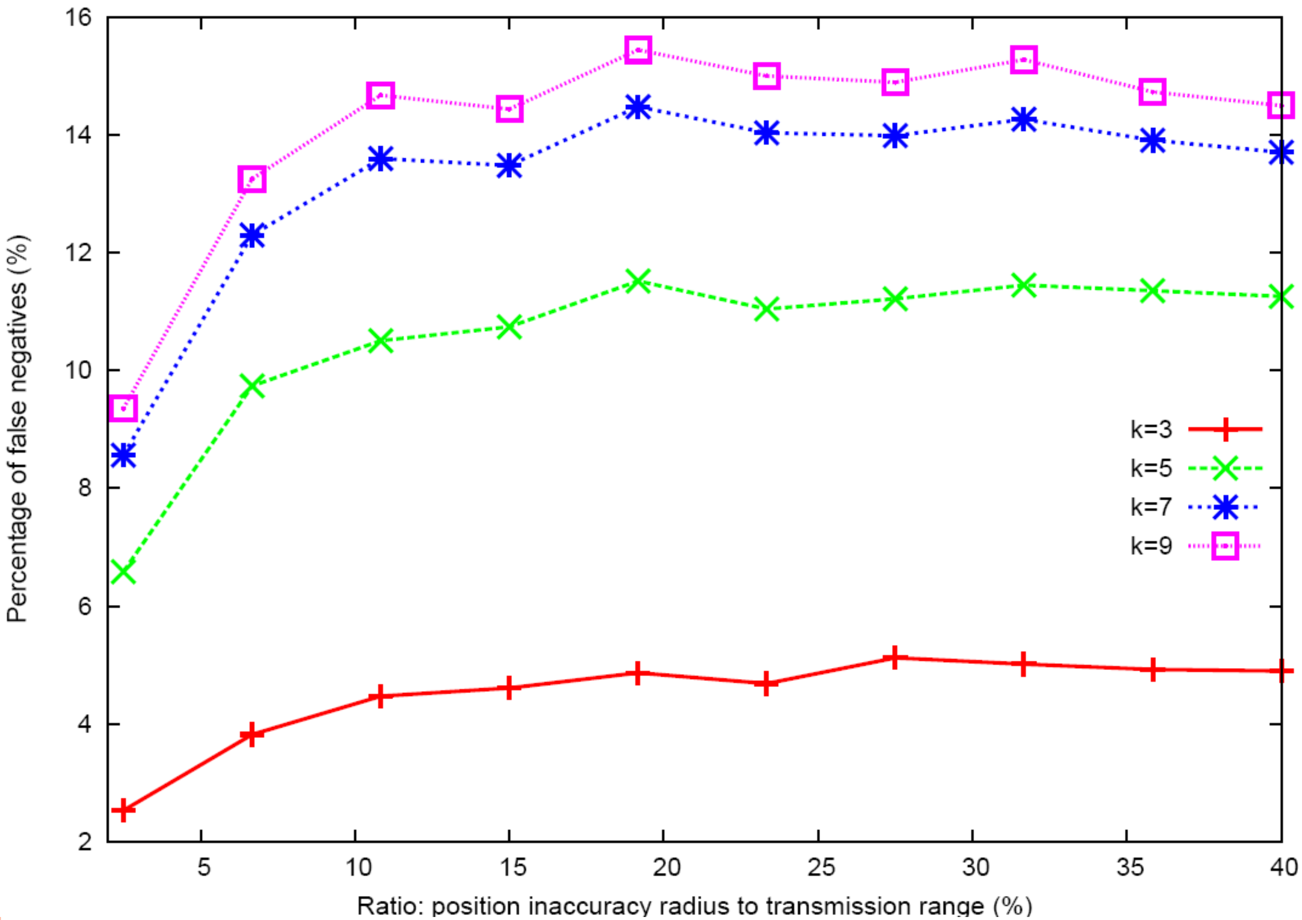
- Rectangular region of 1200 m × 800 m
- Every sensor node has a transmission range of 120 m
- Uniform distribution of the sensor nodes in the deployment area
- Random Waypoint movement model to simulate sensor node and sink mobility
  - Input parameter settings for user mobility in rescue missions
  - The typical speeds for UAVs (40-60 km/h) were used in the real-world AWARE experiments
- The mobility simulations lasted 30 simulation minutes each with update step of 10 seconds.
- Shortest Path First (SPF) is used as a routing metric to build a routing tree.

Evaluation (1)



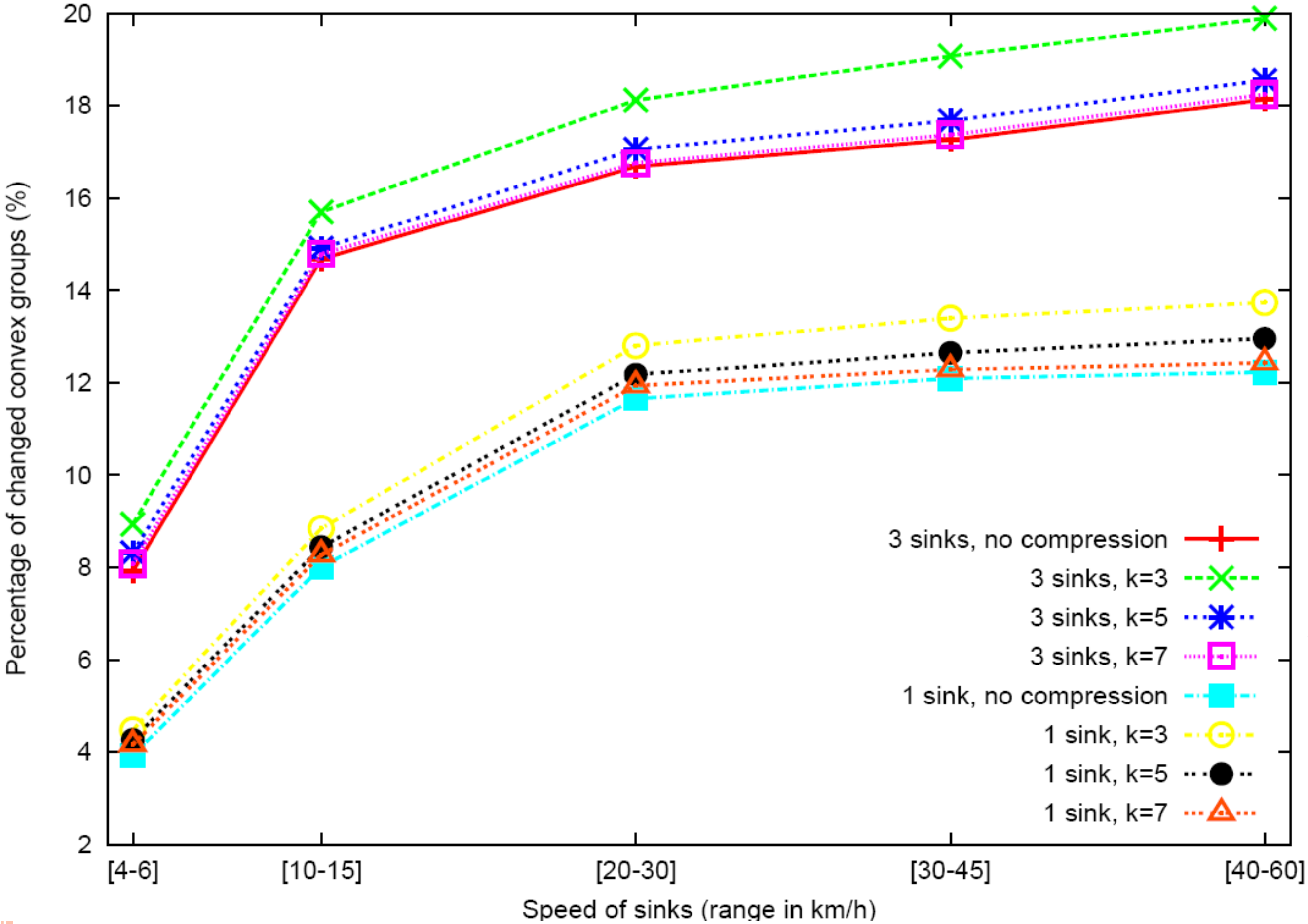
The average and the maximum accumulated compression error obtained for different values of the compression parameter k. Here we varied the number of nodes in the deployment from 150 to 300 and simulated a total of 200 topologies. The compression error decreases exponentially with increasing k.

EVALUATION (2)



The use of compression allows to hide inaccuracy of node positions to some extent. 200 deployments of 200 nodes are used for this evaluation. The positioning error is uniformly distributed within the given error radius. The positioning error of 40% (normalized by the transmission range) results in 16% of nodes being outside of their corresponding convex groups on average.

Evaluation (3)



On the plot 1 or 3 sinks move with different speeds according to the Random Waypoint model. We varied the speeds of mobile sinks from the range of a typical pedestrian (4-6 km/h) to a flying UAV (40-60 km/h) and calculated the average number of convex group changes per update interval (10 sec).

CONCLUSIONS

- **Scoping:** Every convex group describes a subregion on the monitoring area
- **Scalability:** Convex group scoping provides a practical method for specifying subregions of any size and due to compression provides a very scalable abstraction
- **Maintenance Overhead:** Computational, time and message overheads are low
- **Mobility:** Support for sink and sensor node mobility. If a sensor node changes its position, the changes are propagated along the tree only as far as the convex groups are affected. Mobile sinks have a greater influence on the sensor network topology due to more significant changes in the routing tree structure.
- **Reachability:** The algorithm guarantees that a query is disseminated to all nodes that are in its scope (in contrast to top-down approaches, e.g. a Voronoi tesselation)
- **Adaptation:** The compression parameter k can be adapted to:
  - Available memory and bandwidth, position accuracy, link quality, network density

Thank You for Your Attention!