



Rheinische Friedrich-Wilhelms-Universität Bonn

Institute for Computer Science IV

Sensor Networks and
Pervasive Computing Group

Römerstraße 164
D-53117 Bonn

Time-bounded and Space-bounded Sensing in Wireless Sensor Networks

Olga Saukh, Robert Sauter and Pedro José Marrón

Universität Bonn, Bonn, Germany and Fraunhofer IAIS, St. Augustin, Germany

{saukh, sauter, pjmarron}@cs.uni-bonn.de

DCOSS 2008

Santorini Island, Greece

Composite event detection

- Today's sensor network applications require **complex sensors**
 - Complex data needs to be **preprocessed** on sensor nodes to locally detect a composite event
 - Detection of composite events requires
 - Sensor readings from **several complex sensors**
 - However, it is not feasible for a sensor node to be equipped with all complex sensors
 - Sensor readings taken **simultaneously**
 - However, it is difficult to implement parallel data access to different sensors on the same node
- Goal:** Distributed **space-bounded** and **time-bounded** sensing to enable **composite event detection**



Space-bounded Sensing

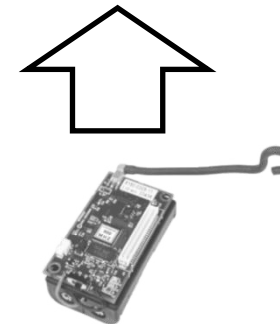
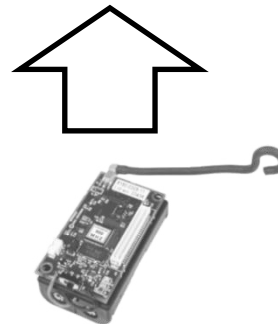
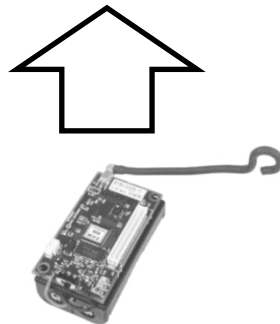
Complex Event = $((S_1 > A) \text{ AND } (S_2 > B)) \text{ OR } (S_3 < C)$

Problem: Space resource conflict
Solution: Distributed space coordination -
Distribution of sensing tasks among several nodes

Available sensors

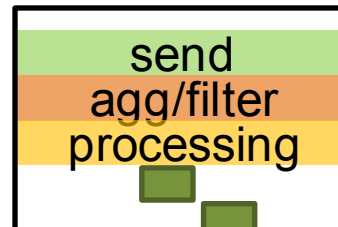
Available sensors

Available sensors



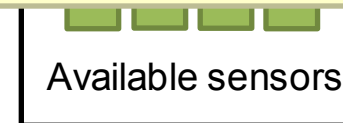
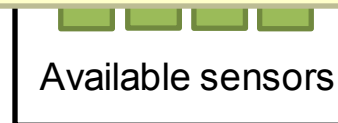
Time-bounded Sensing

Scenario = Monitoring of Bridge State; **Event** = Appearance of a crack in the construction

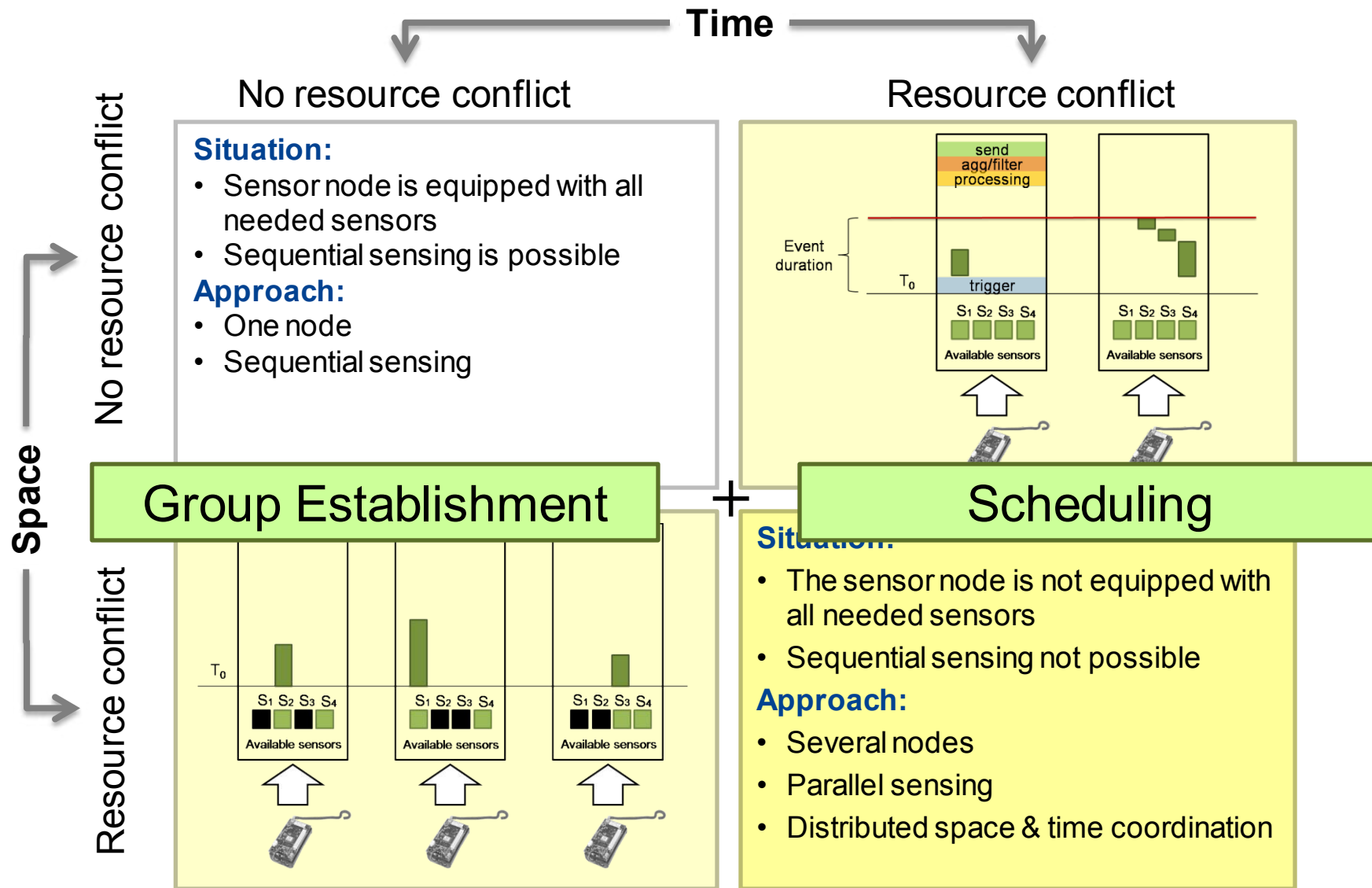


Event
duration

Problem: Time resource conflict
Solution: Distributed time coordination -
Distribution of sensing tasks among several nodes



Problem Space



Group Establishment

Set of sensors
- ordered by
priorities

$$S = S_1 S_2 S_3$$

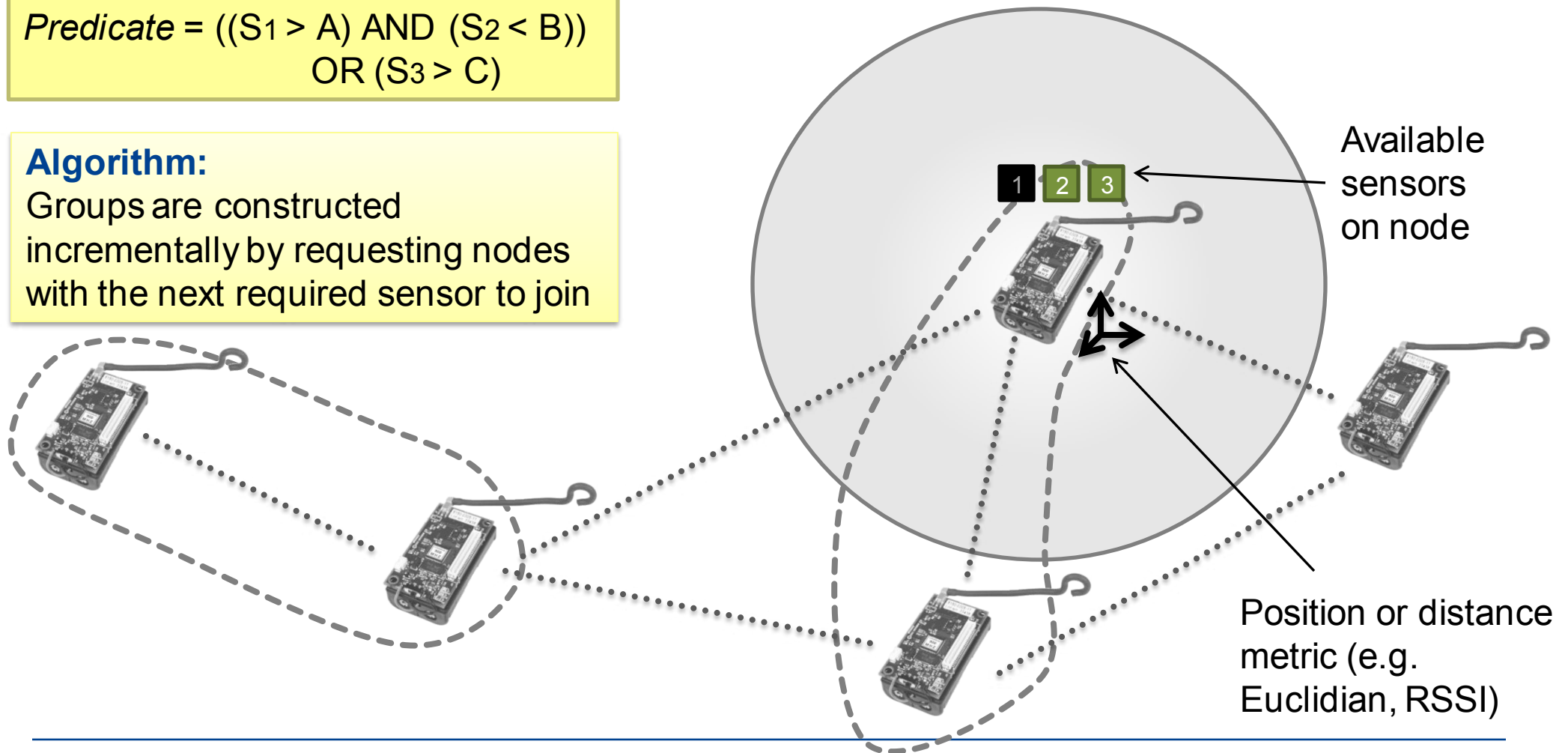
$$R = R_1 R_2 R_3$$

← Sensing ranges

Predicate = (($S_1 > A$) AND ($S_2 < B$))
OR ($S_3 > C$)

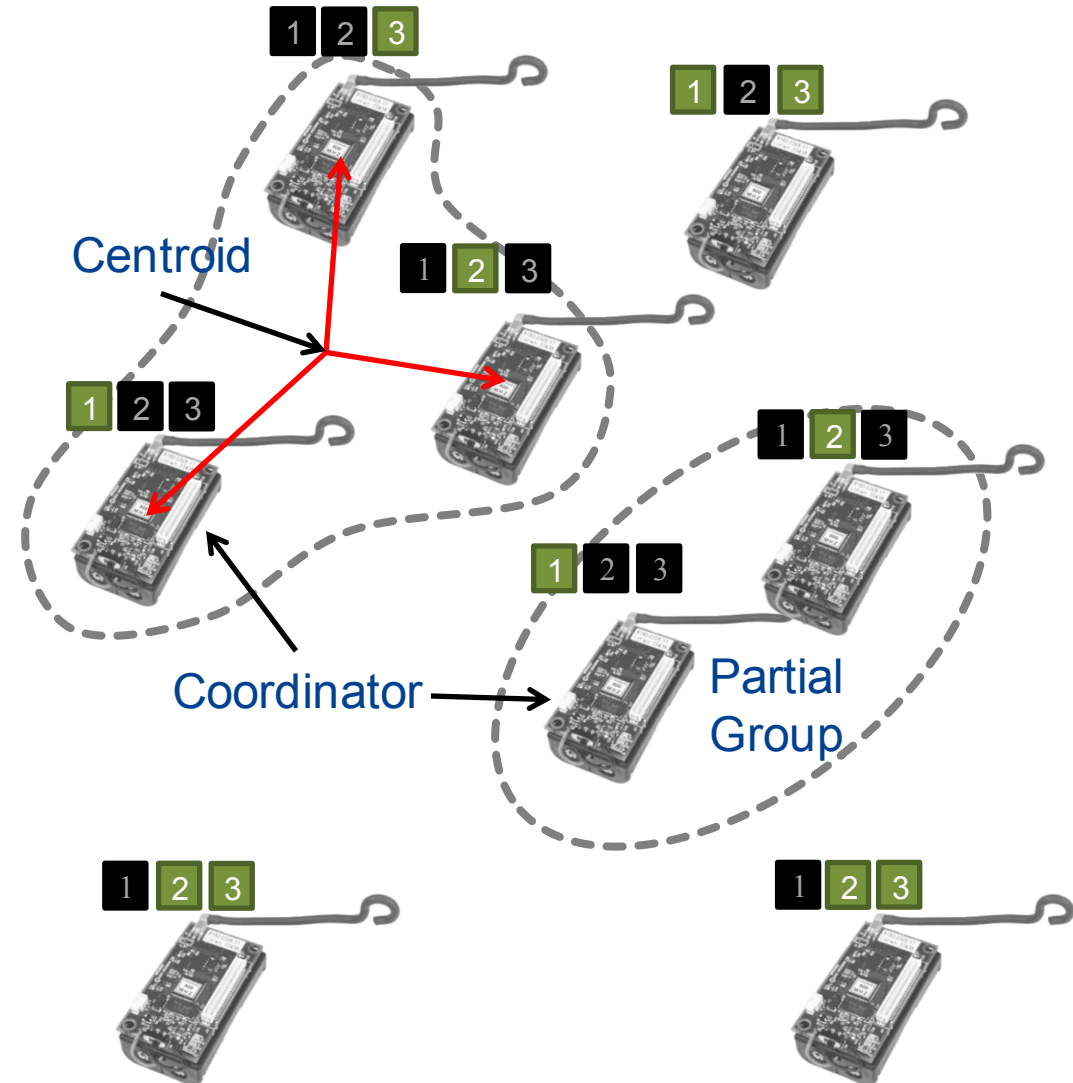
Algorithm:

Groups are constructed incrementally by requesting nodes with the next required sensor to join



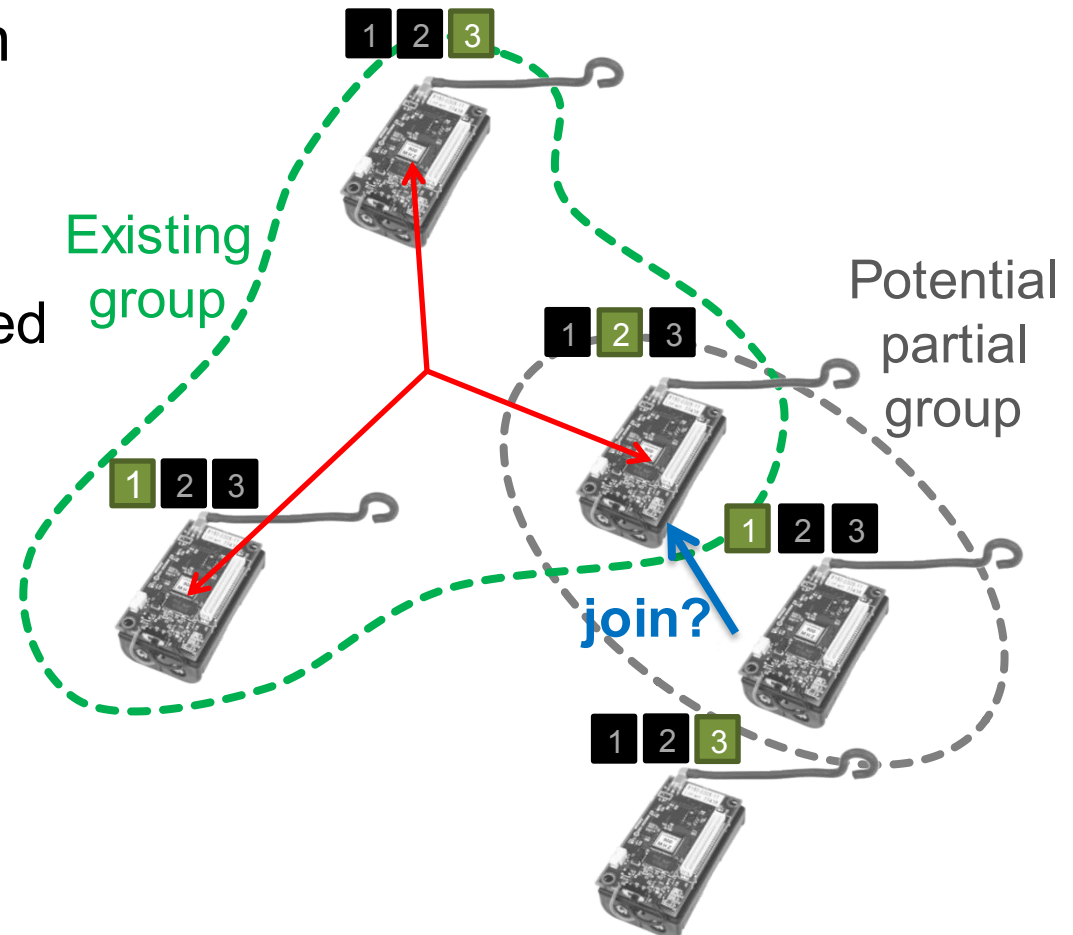
Group Establishment – Algorithm Components

- Dispersion Metric
 - Measures the closeness of the members of a group → **Quality**
 - E.g., maximum distance to centroid
- Selection Rule
 - Defines who selects the new member
 - Coordinator only
 - All Members (higher overhead)
- Grouping Algorithm
 - Controls overall process
 - **FirstChoice: Greedy**
 - **BestChoice: Backtracking**



Grouping Algorithms

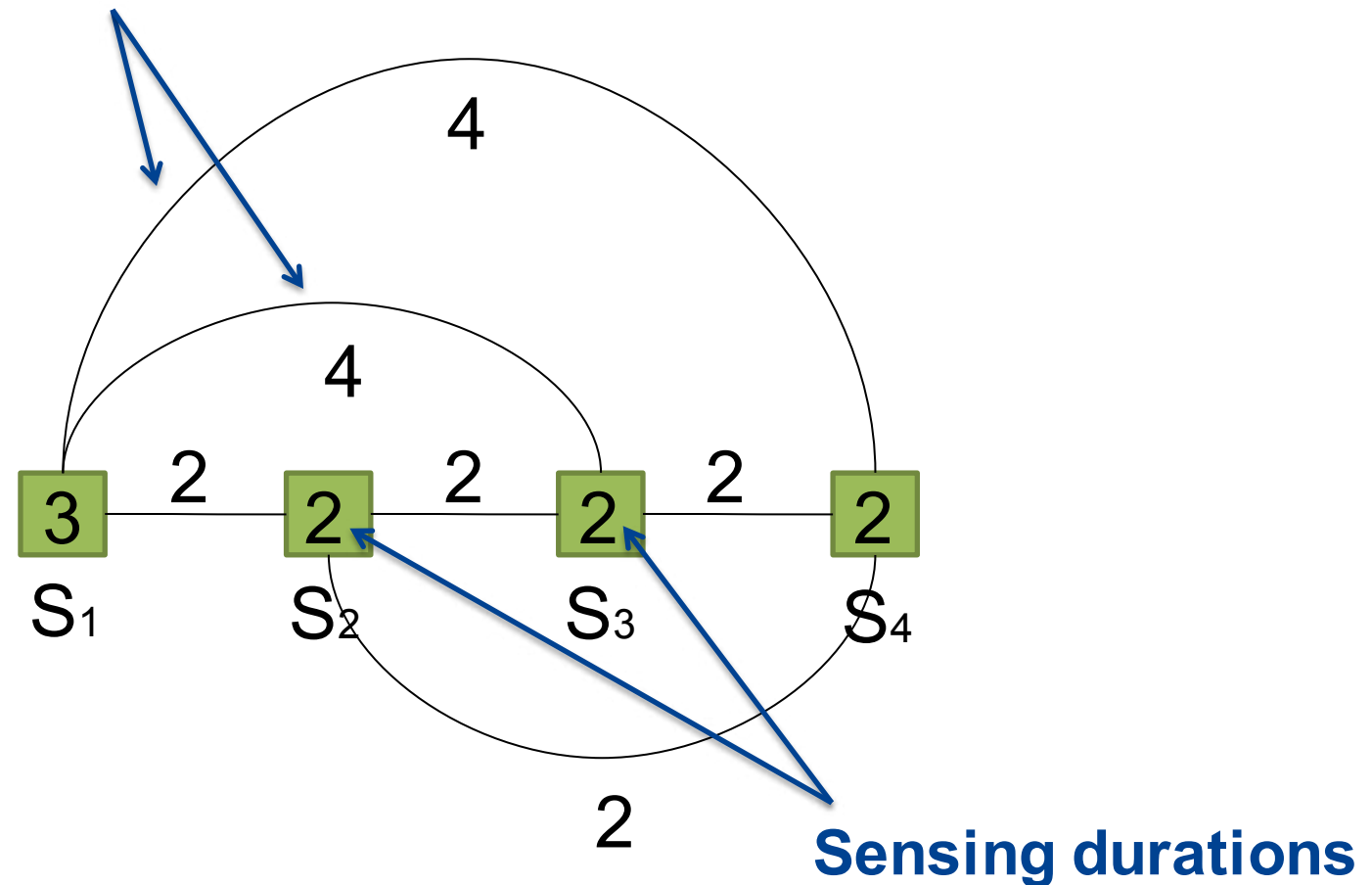
- Question: When does a node join a partial group?
 - **FirstChoice**: When the requested sensor is **not already used** in another group
 - **BestChoice**: When the partial group is “**better**” (based on dispersion metric) than the existing group
 - Existing group must **backtrack**



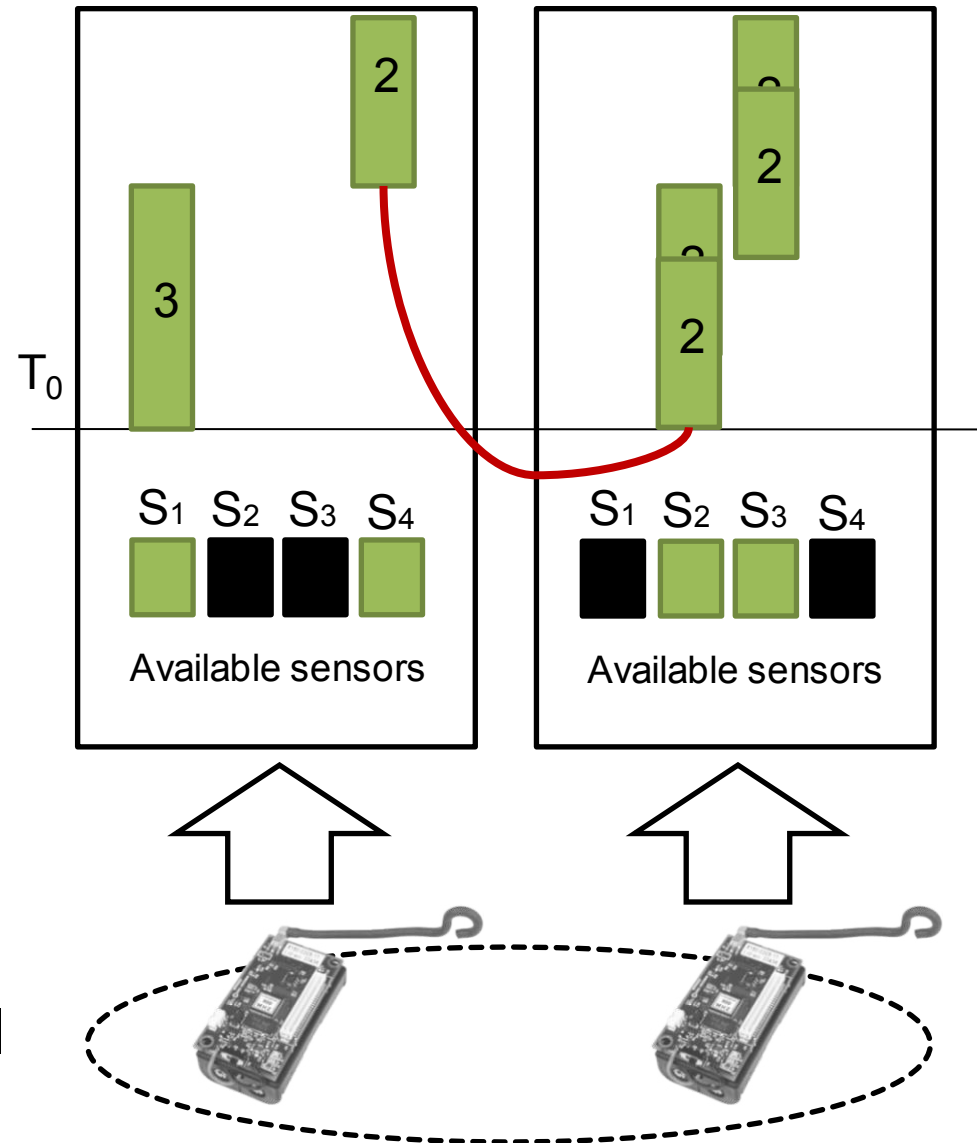
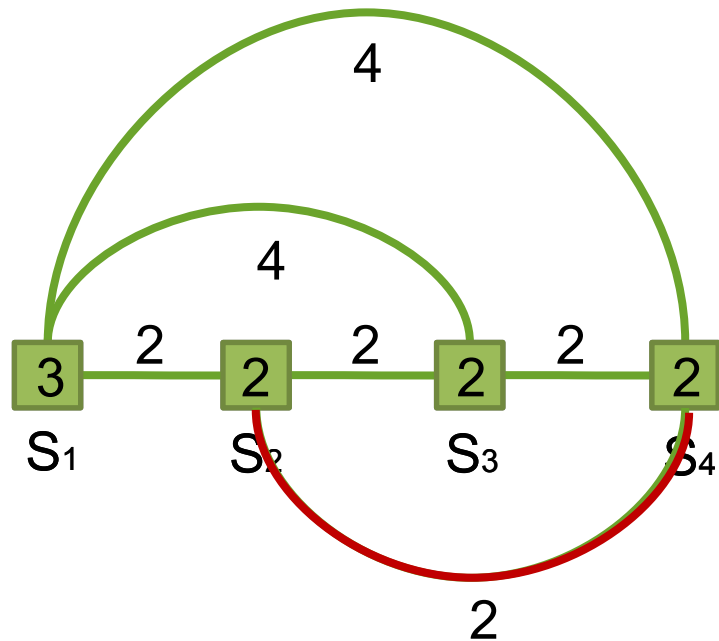
Scheduling

Concurrency constraints:

- Maximum time difference between executions of individual tasks
- Also defines the maximum overall schedule duration



Scheduling Algorithm



Properties:

- Finds the best solution if exists
- Constructs schedule of minimal length

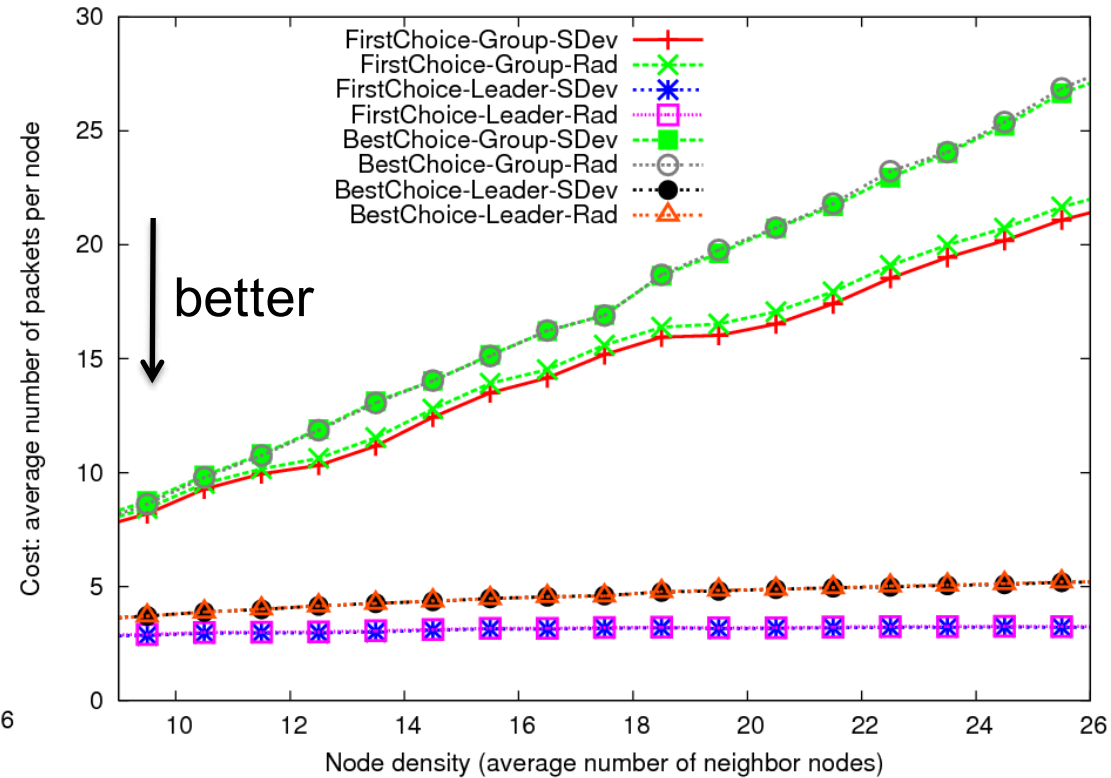
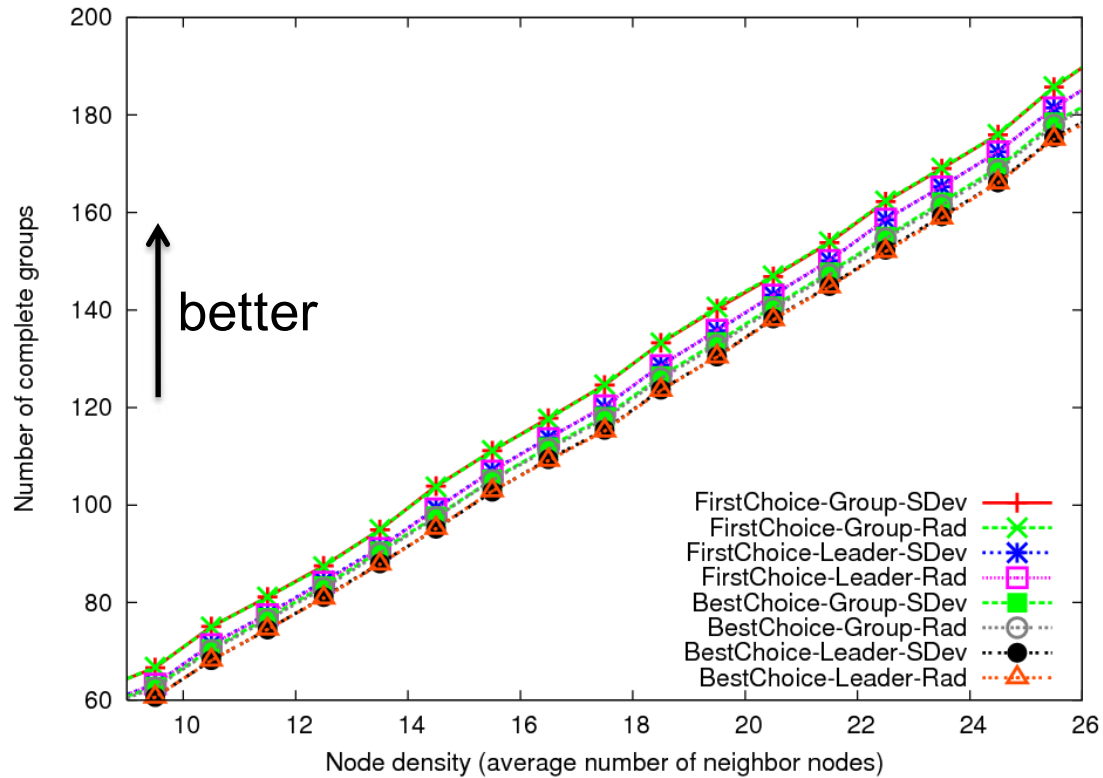
Simulation Setup

- Area: 900 x 530 m²
- Node distribution:
 - perturbed grid
 - 150 – 450 nodes
 - 9 – 26 average node degree
- Random distribution of sensors
- Transmission range: 100 m
- Average over 50 simulation outputs
- **Two kinds of tests**
 - Group establishment only
 - Group establishment + Scheduling

Sensor types
Sensor distribution
Sensing range

Group Establishment			
1	2	3	4
50%	50%	50%	50%
50m	50m	50m	50m

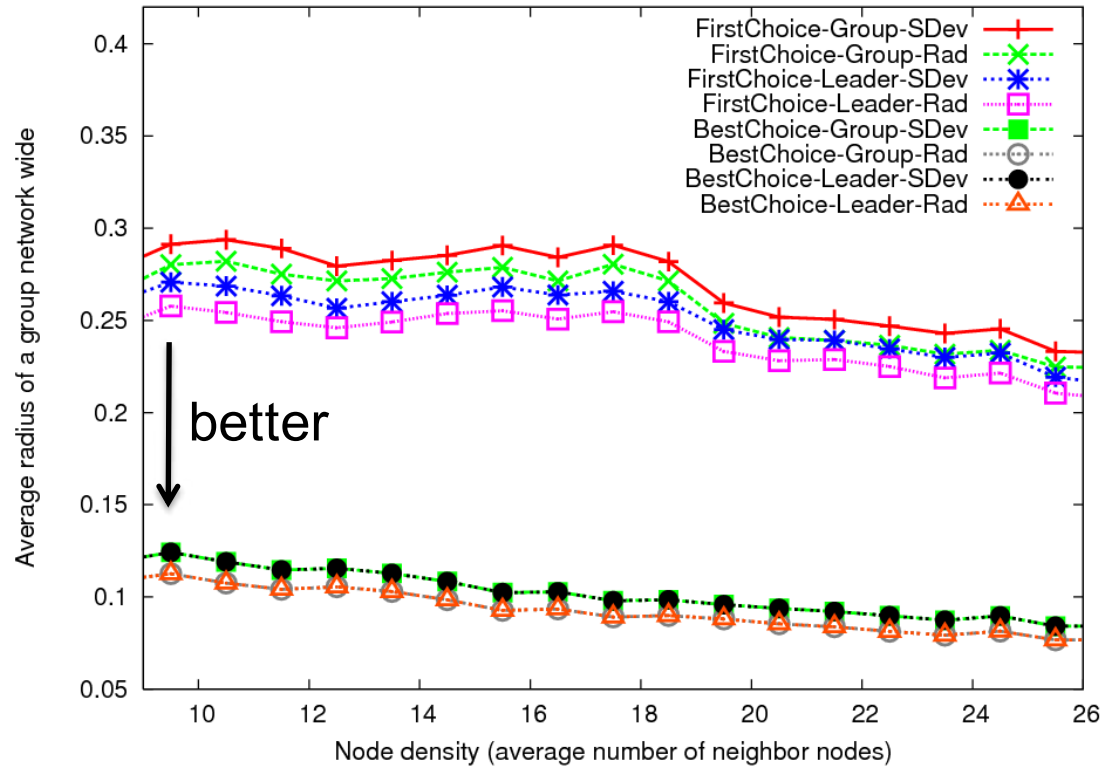
Evaluation: Group Establishment



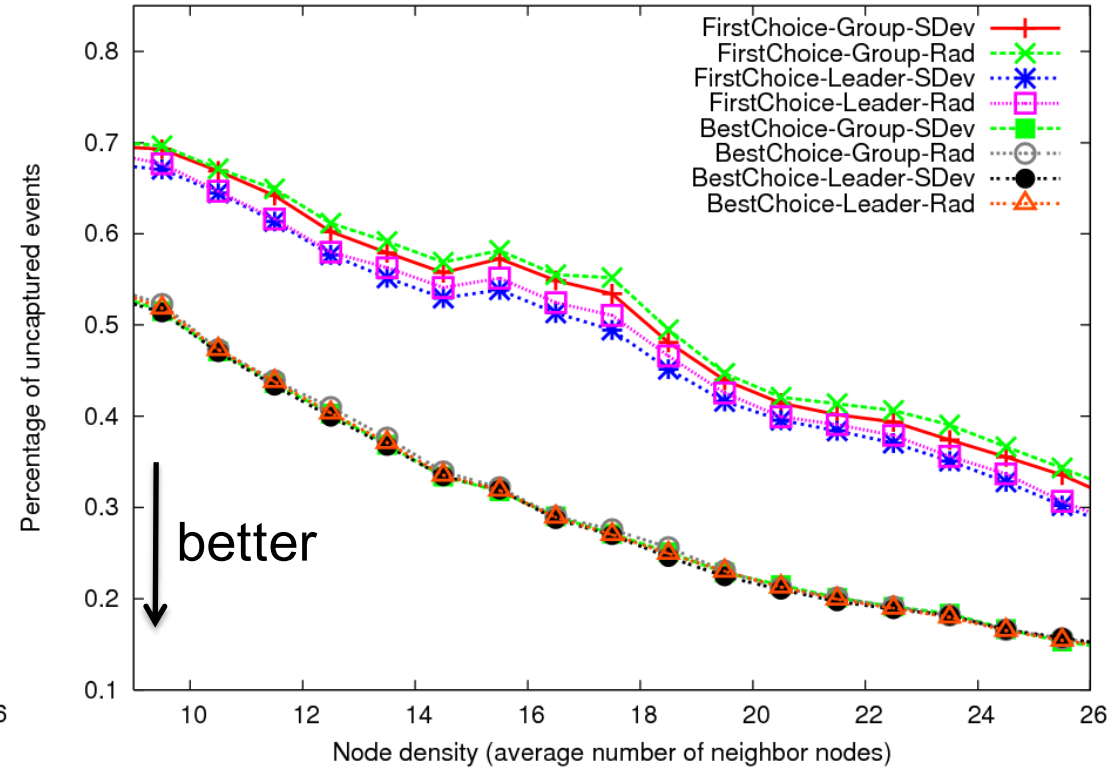
Number of complete groups
in the network

Cost analysis over complete
groups

Evaluation: Group Establishment (2)



Average radius of a group in the network



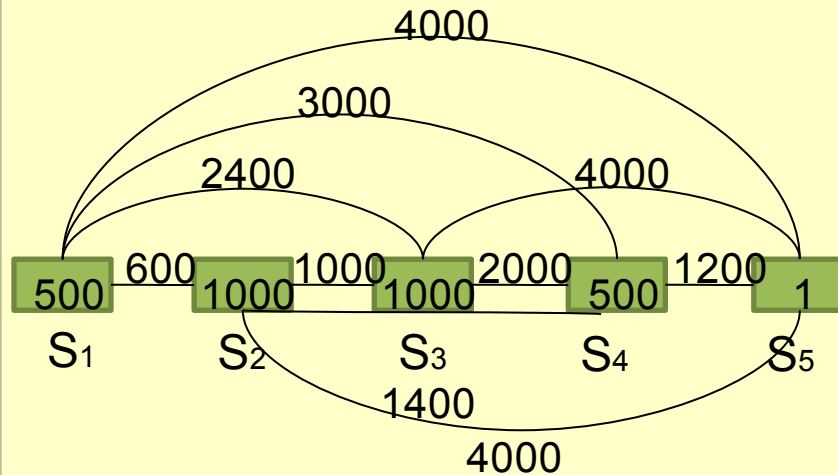
Percentage of un-captured events (uncovered region)

Test – Group Establishment & Scheduling

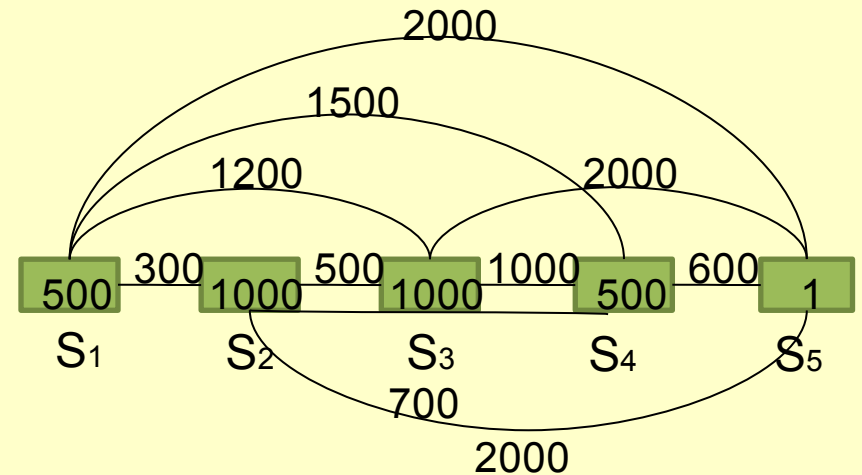
Sensor types
Sensor distribution
Sensing range

Group Establishment				
1	2	3	4	5
50%	50%	50%	60%	60%
50m	50m	50m	50m	100m

Scheduling



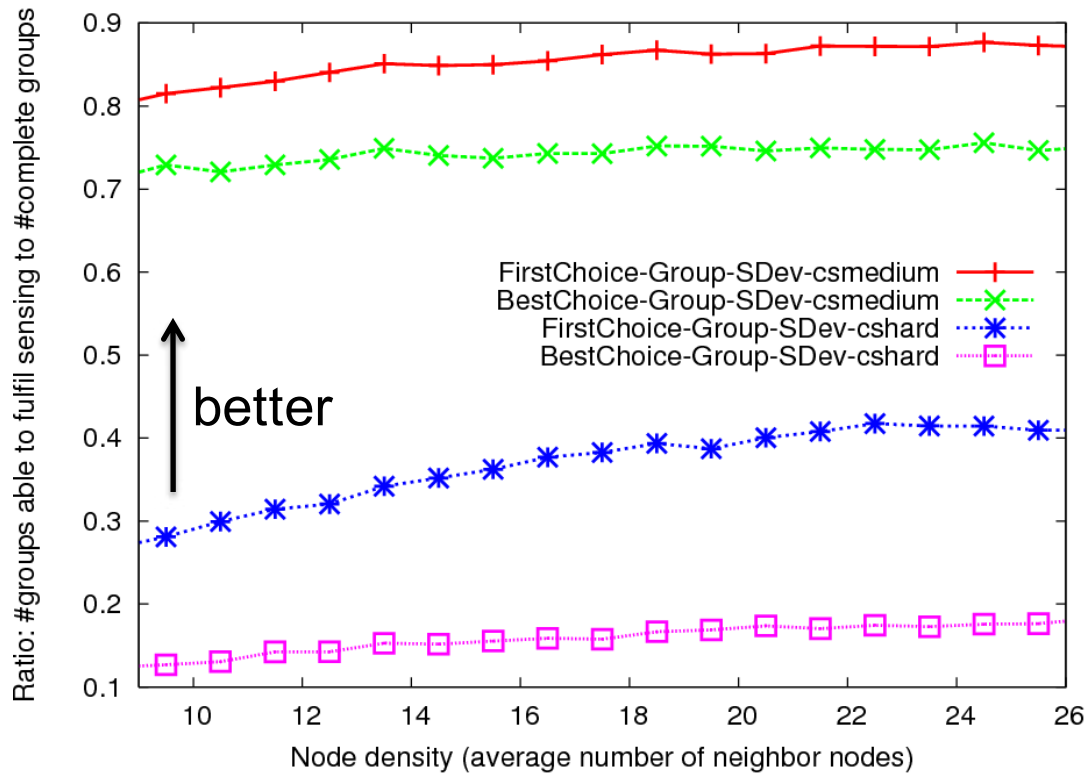
“medium” concurrency constraints



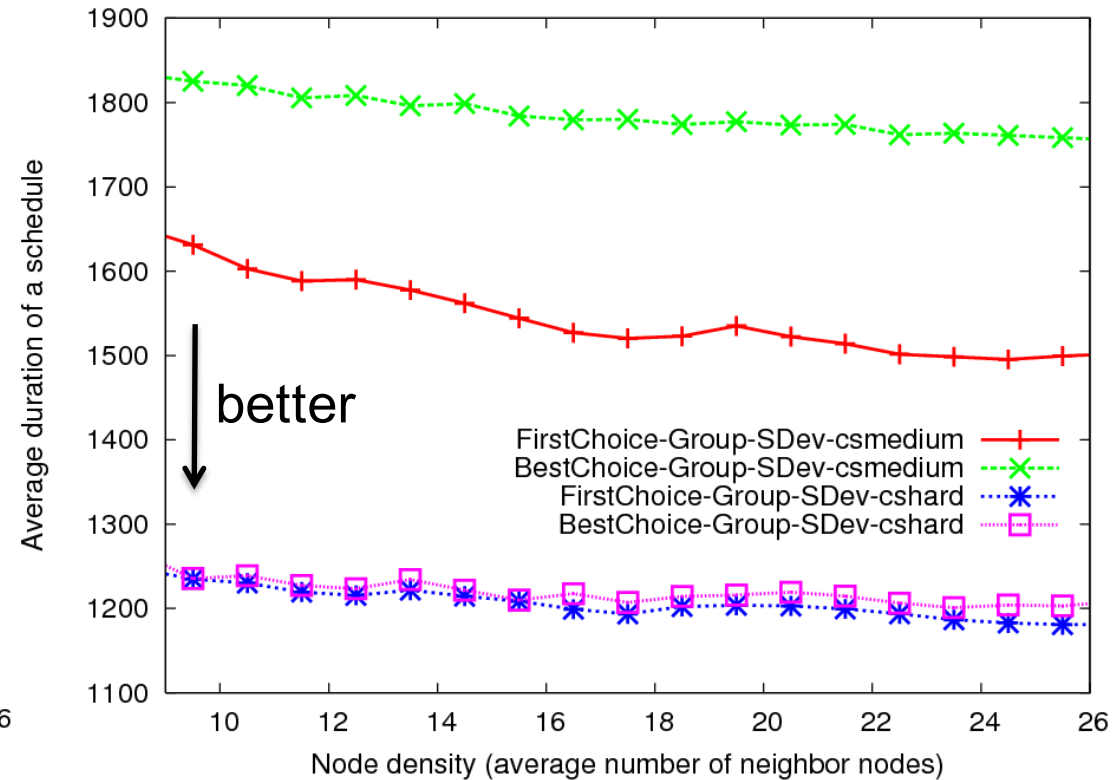
“hard” concurrency constraints

Neither schedule can be executed on one node: distribution is required for both

Evaluation: Scheduling



Percentage of groups with a valid schedule



Duration of a schedule

Related Work

- Query-based systems
 - TinyDB [S. R. Madden et al.], Cougar [W. F. Fung et al.]
→ Consider neither space nor time coordination
- Composite event detection
 - K-watched composite event detection [B. Krishnamachari et al.], [E. Ould-Ahmed-Vall et al.], [C. T. Vu et al.]
 - State transitions [K. Römer et al.]
→ Do not consider time coordination
- Related but different goals
 - Spatial node grouping
 - Clustering, etc. [M. Handy et al.], [S. Yoon and C. Shahabi]
 - Task scheduling [P. Brucker], ...

Conclusions

- Composite event detection with complex sensors requires
 - Group constructions with respect to sensor coverage
 - Distributed scheduling to fulfill concurrency constraints
- Space-bounded sensing
 - Two algorithms with different properties
- Time-bounded sensing
 - Priorities allow efficient and deterministic algorithm
 - Solution always found if one exists
- **Future work:**
 - Real-world tests of the developed concepts



Thank You for Your Attention!

Are there any questions?

Olga Saukh
Sensor Networks and Pervasive Computing Group
Institute for Computer Science IV
Universität Bonn
Römerstr. 164
53117 Bonn, Germany
saukh@cs.uni-bonn.de

