Technische Universiteit
Eindhoven
University of Technology

QoS Trade-off Analysis for Wireless Sensor Networks

Rob Hoes, Twan Basten

Joint work with
Phillip Stanley-Marbell, Marc Geilen,
Chen Kong Tham, Henk Corporaal

Department of Electrical Engineering
Electronic Systems

'Knowing is not understanding.'
Charles Kettering

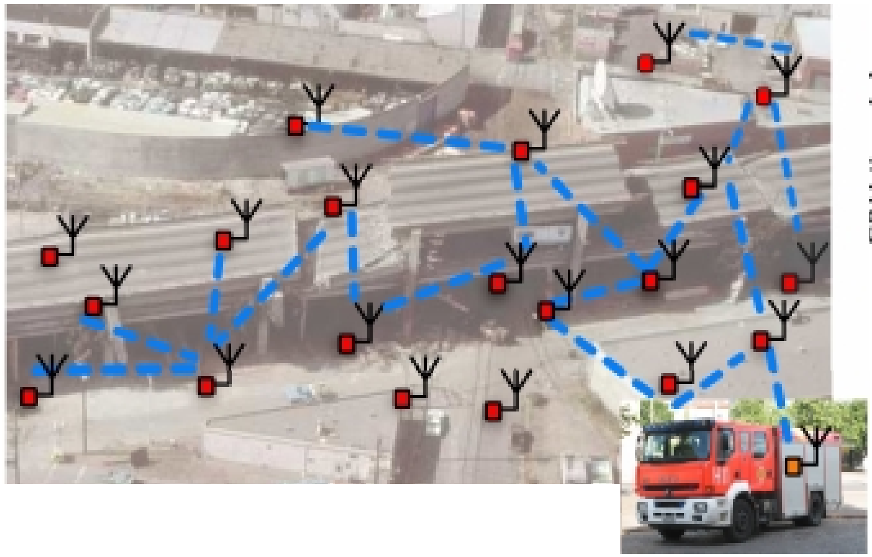
In short …

Wireless sensor network

- 900 nodes
- 27 configurations per node
- 4 quality metrics

27^{900} configurations in 4D space

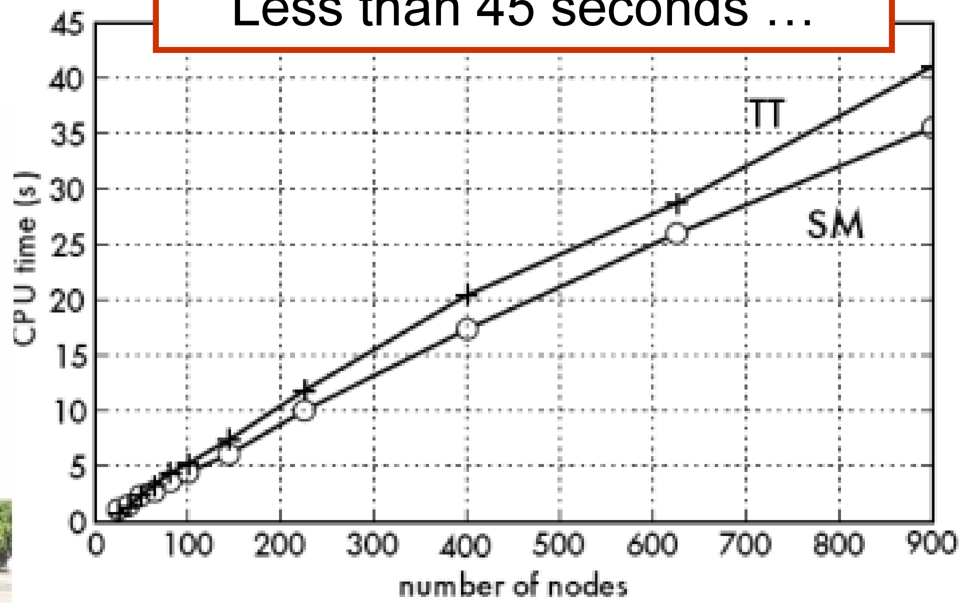
even with fixed routing and data aggregation strategies …



Problem
How to configure the network ?

Approach
Find all Pareto optimal configurations via an exact algebraic analysis

Less than 45 seconds …



MSWiM 2007 best paper

Outline

- Quality of Service (QoS)
- Models
- QoS analysis
- Experiments
- Looking forward

Quality of Service

Provide services to clients with agreed quality guarantees

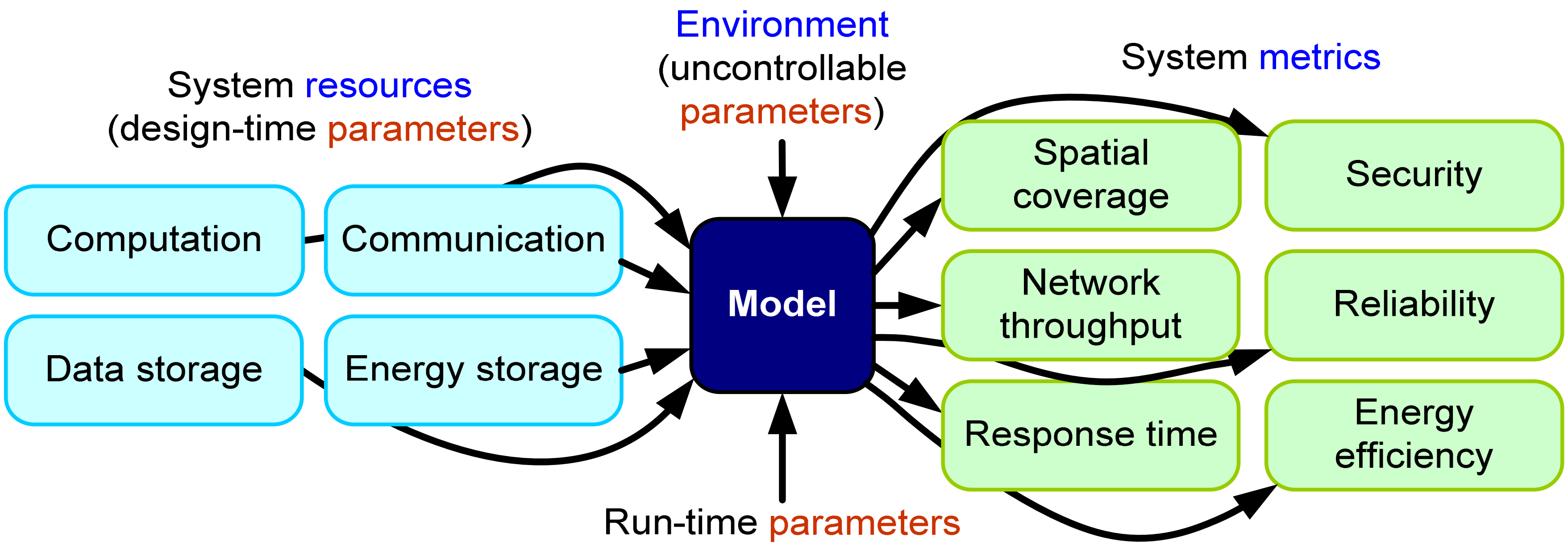
- Services
  - Target tracking
  - Spatial mapping
  - …
- Quality metrics
  - Spatial coverage
  - Reliability
  - Reporting frequency
  - Response time
  - Network lifetime
  - Security
  - …
- Guarantees
  - Soft: violations are acceptable to some extent
  - Firm: an incidental violation is not critical
  - Hard: violations cannot be tolerated

Outline

- Quality of Service (QoS)
- Models
- QoS analysis
- Experiments
- Looking forward

A model

… predicts system metrics given parameter values



… should be targeted to the problem at hand

- system level quality metrics
- trade offs

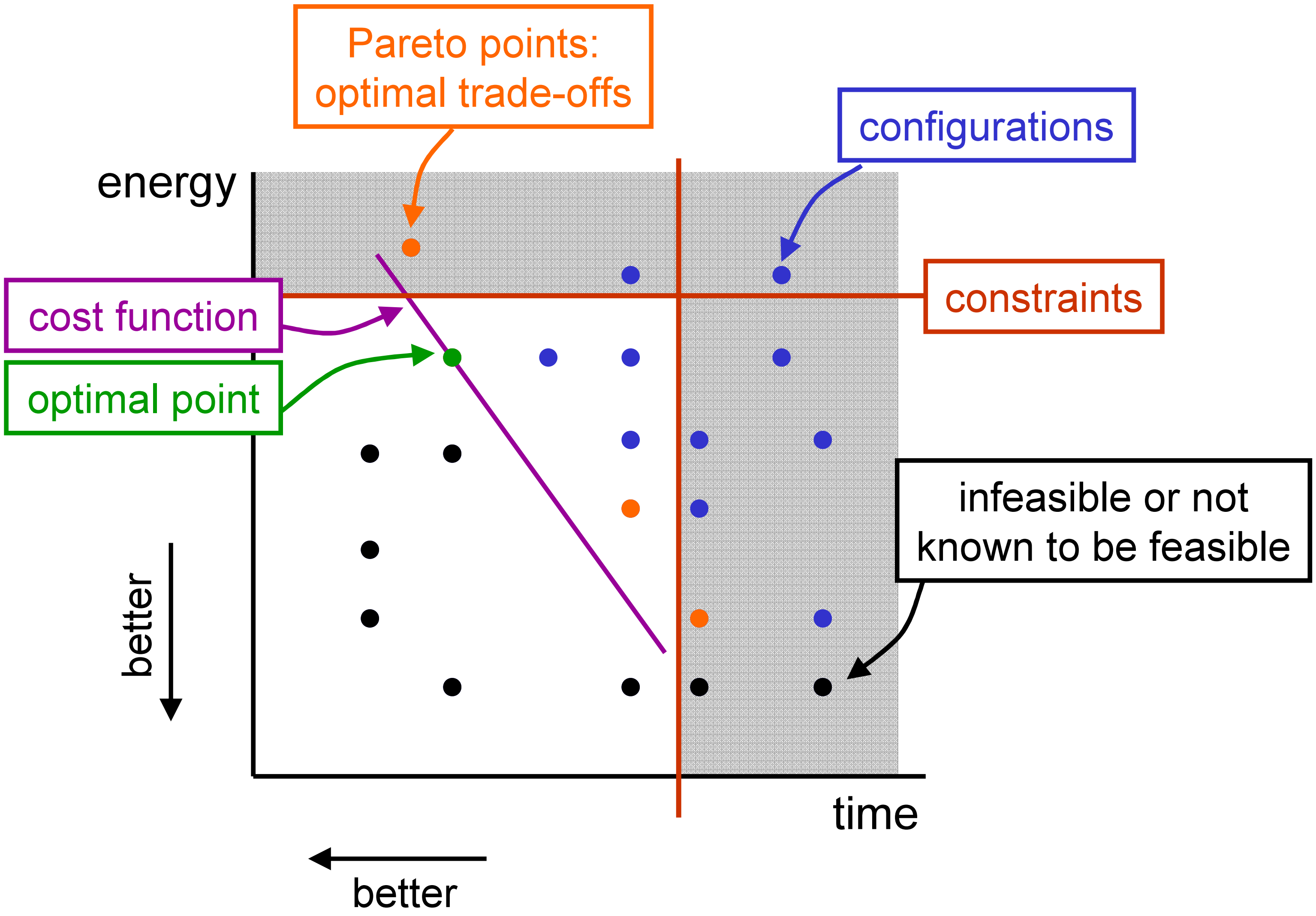
Vilfredo Pareto (1848–1923)

- Pareto improvement:

  Making someone better off, without
  making anyone else worse off

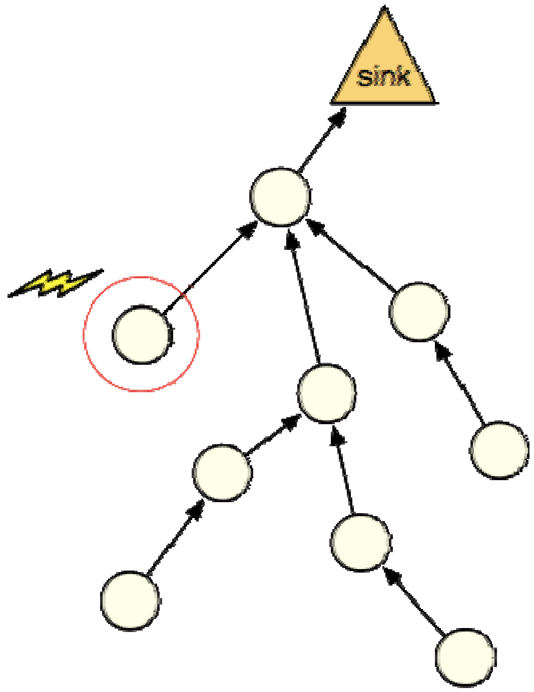
- Pareto optimal:
  - When no further Pareto improvements can be made
  - Result: Pareto points

A Pareto Space

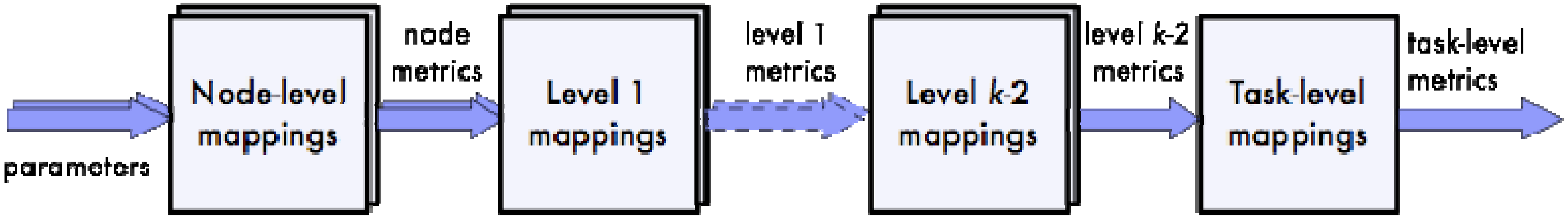


System Architecture

- Network of sensor nodes
- Node locations and routing tree are known
- Nodes can be configured:
  - Transmit power
  - Sample rate
  - …
- One sink node
- Typical client applications:
  - Target Tracking
  - Spatial Mapping



Model Hierarchy

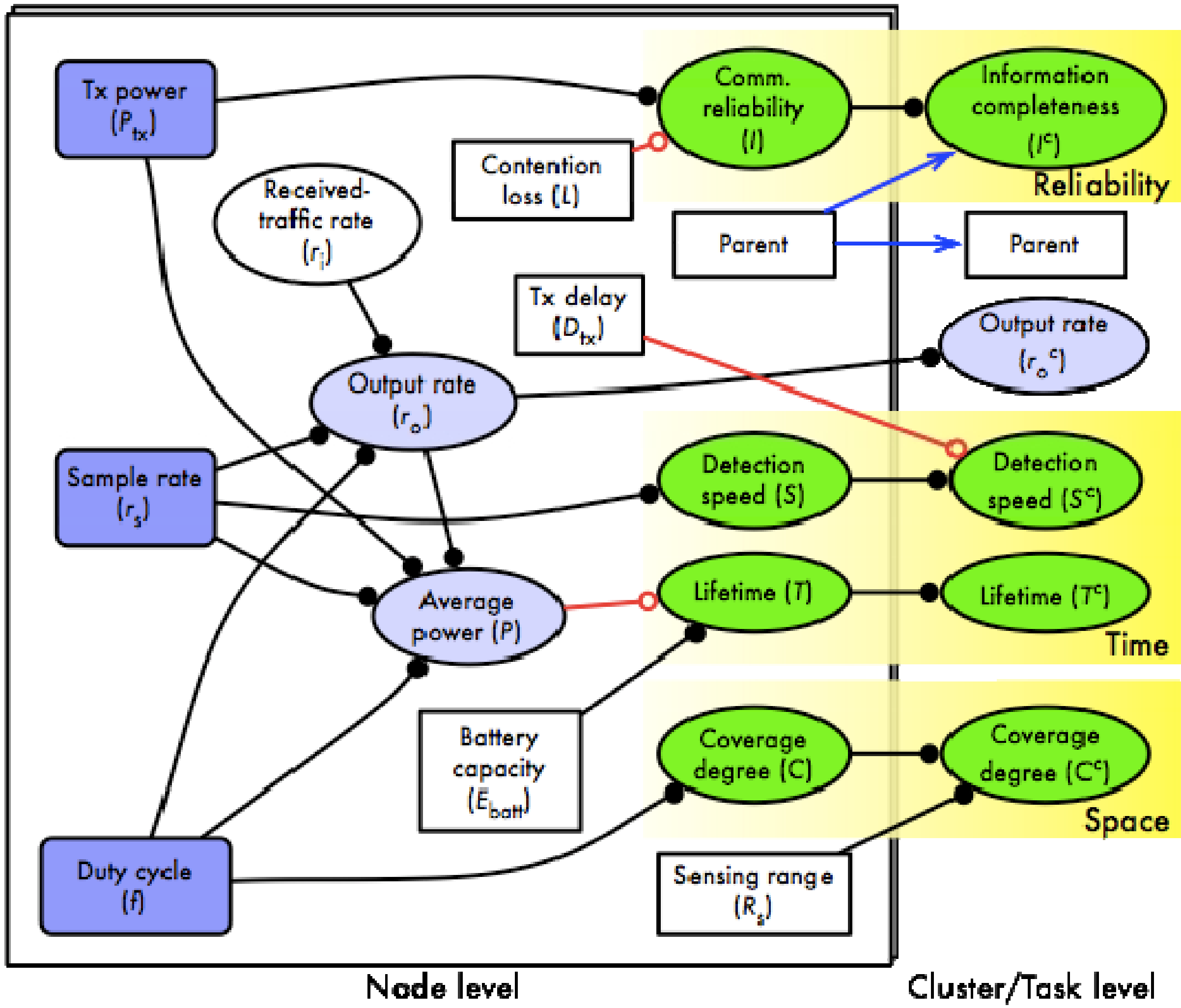


Levels:

- Node
- Clusters: groups of nodes
- Task: the whole network

QoS Trade-off Model

… for the target tracking task



Closed-form Expressions

TU/e

Node-level mappings (F_{n}) for a node n

Reliability

$$I(n) = \left(1 - Q\left(\sqrt{2P_{\mathrm{rx}}(P_{\mathrm{tx}}(n))/N}\right)\right)^{b} (1 - L) \qquad (1)$$

Time

$$T(n) = \frac{E_{\mathrm{batt}}}{P(n)} \qquad (2)$$

$$r(n) = r_{\mathrm{s}}(n) \quad [\mathbf{SM}] \qquad (3)$$

$$S(n) = \left(\frac{1}{r_{\mathrm{s}}(n)} + D_{\mathrm{s}}\right)^{-1} \quad [\mathbf{TT}] \qquad (4)$$

Space

$$C(n) = f(n) \qquad (5)$$

Additional metrics

$$\cdots \qquad (6)$$

Heterogeneity

- Networks with various node types. Various
  - parameters
  - mapping functions
  - metrics
- Various tasks
  - Target Tracking & Spatial Mapping combined

Outline

- Quality of Service (QoS)
- Models
- QoS analysis
- Experiments
- Looking forward

Problem Statement

TU/e

Given

- Task-level QoS trade-off model
- Parameter ranges

… compute all Pareto-optimal configurations

(satisfying given QoS constraints)

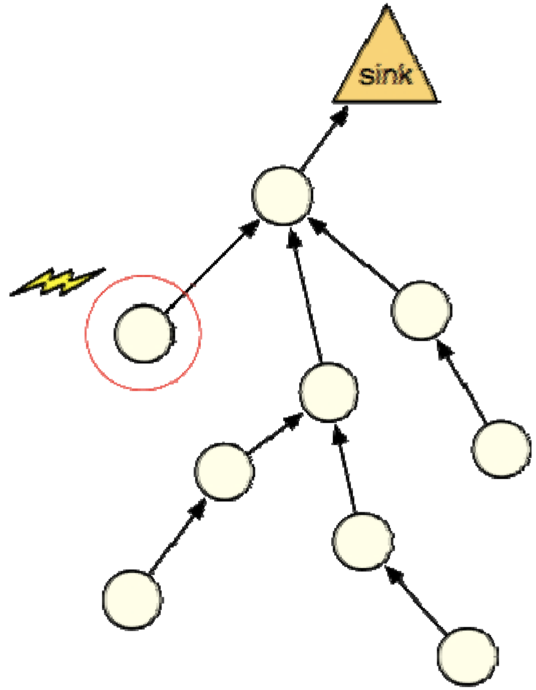
Complexity

- Suppose:
  - n nodes in the network
  - k parameter vectors per node
  - results in k^n configurations

n = 9 nodes

3 parameters with 3 settings each

$27^9 = 7.6*10^{12}$ configurations



- Scalability problem:

  #configurations grows exponentially with the number of nodes!

Idea: Clustering

- Compute Pareto points in an incremental way

n nodes / clusters

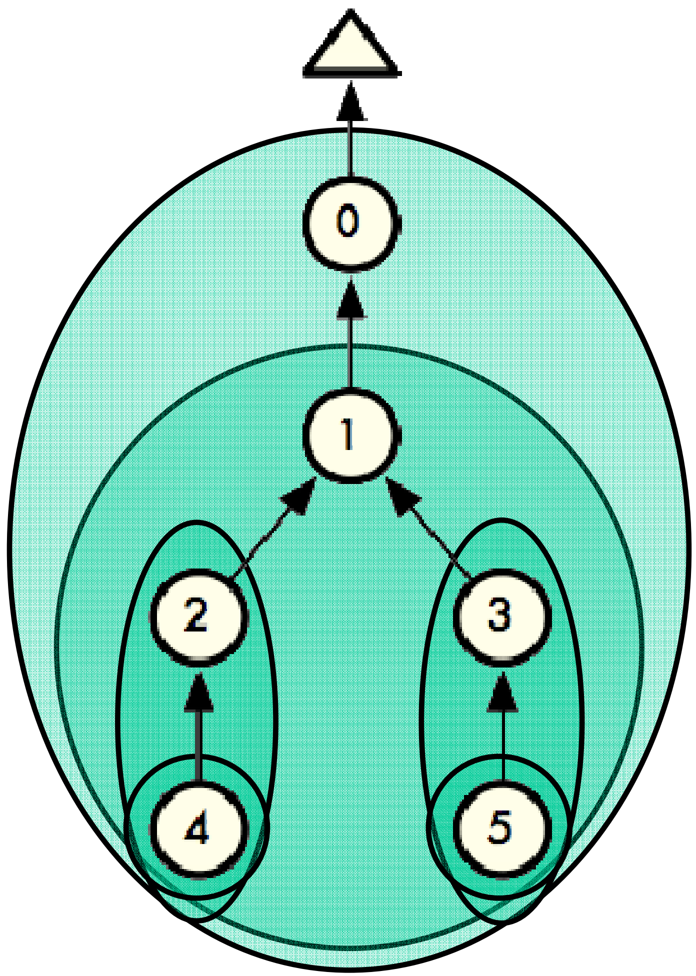
p Pareto-optimal configurations per cluster

c children per node

$O(p^c)$ work per clustering step

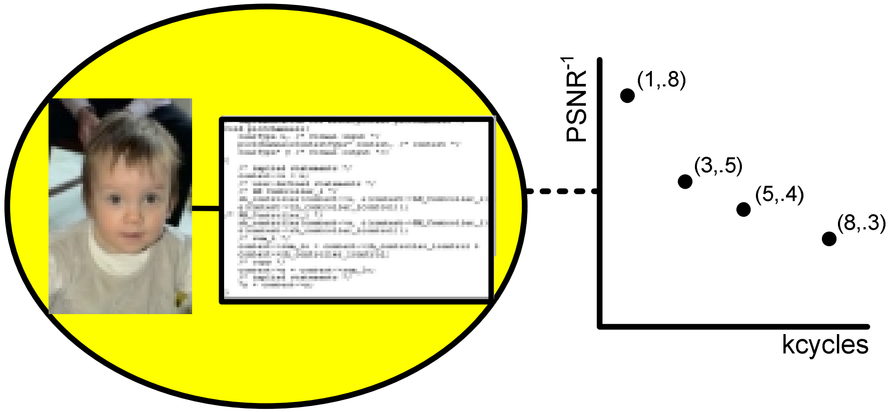
$O(n)$ work for the network

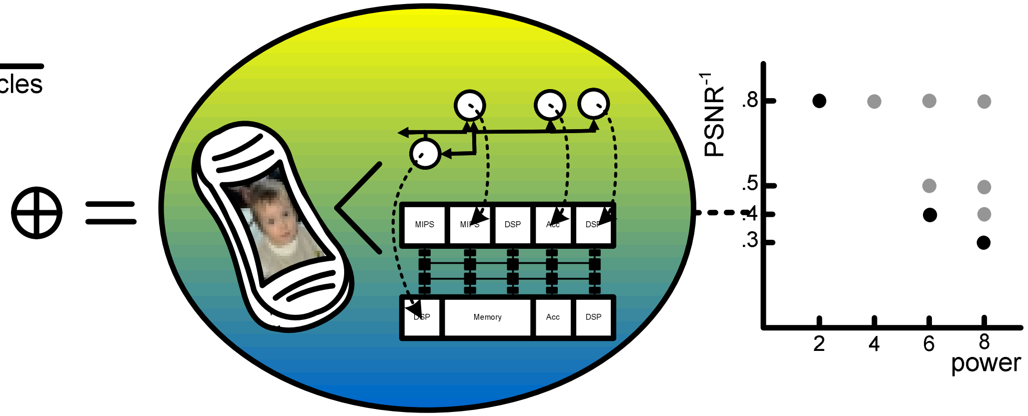
Linear in the network size !

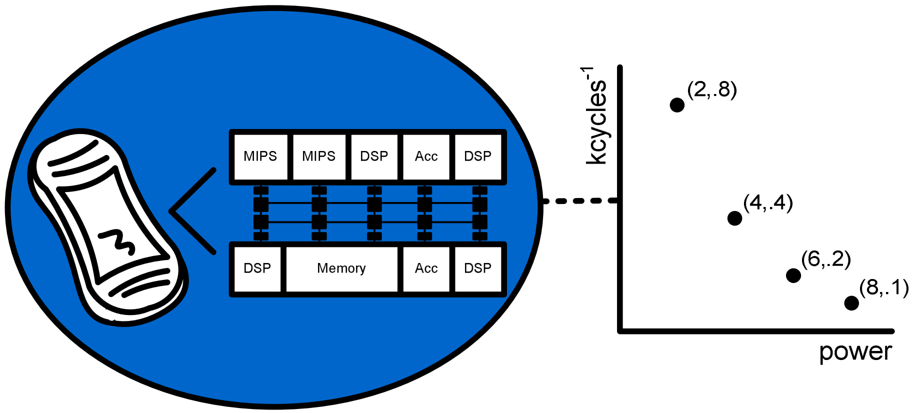


Intermezzo: Pareto Algebra

essential for efficiency



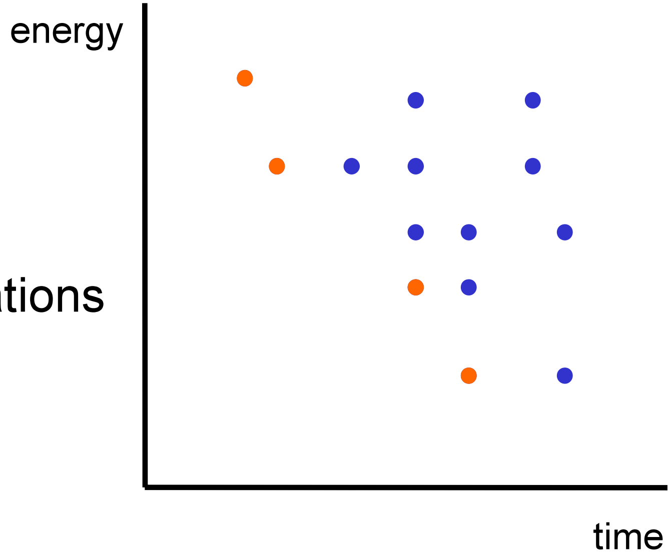




$$S \oplus T = ((S \bullet T) \cap \{c \mid kcyc_S(c) \leq 1/kcyc_T(c)\}) \downarrow \{PSNR^{-1}, power\}$$

Intermezzo: Pareto Algebra

TU/e

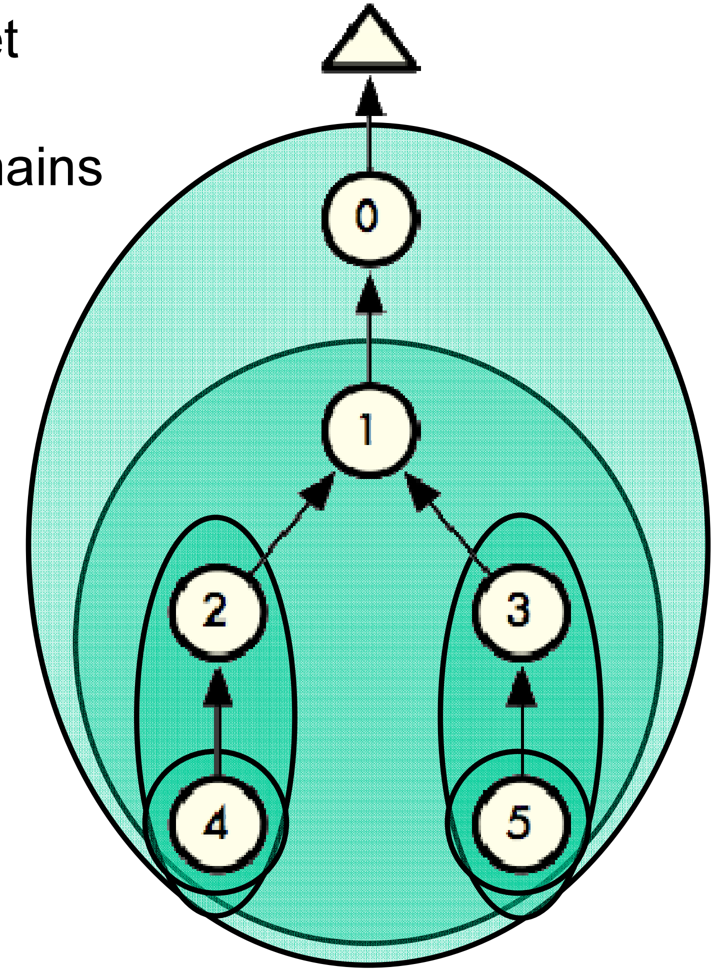


- The elements: sets of configurations
- A set of operators:
  - Minimization
    Gives the Pareto points of a set of configurations
  - Product
    Cartesian product of configurations
    e.g. application and platform configurations
  - Alternative
    Set union of configurations
    e.g. two sets of platform configurations
  - Constraints
    Selects solutions according to constraints
    e.g. all application configurations with some minimal quality
  - Abstraction
    Discards information about solutions
    e.g. bandwidth usage in bandwidth × energy × quality configurations

Fundamenta Informaticae, 2007

Clustered Approach

1. Initialise nodes as one-node clusters
2. Combine two or more clusters (product)
3. Derive quality metrics of new cluster
4. Minimise new cluster's configuration set
5. Optionally, check QoS constraints
6. Back to step 2 until a single cluster remains



Monotonicity

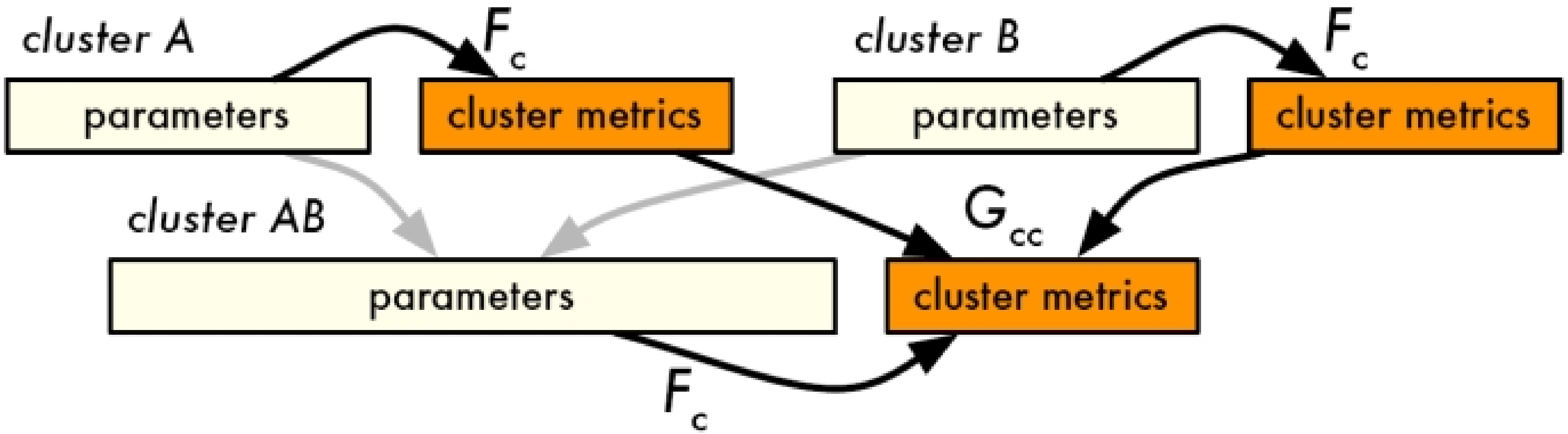
TU/e

Non-optimal cluster configurations
should remain
non-optimal at task level

Deriving Cluster Metrics

Two ways to compute cluster quality metrics:

- From node parameters (always possible)
- From lower-level cluster metrics (not always possible)



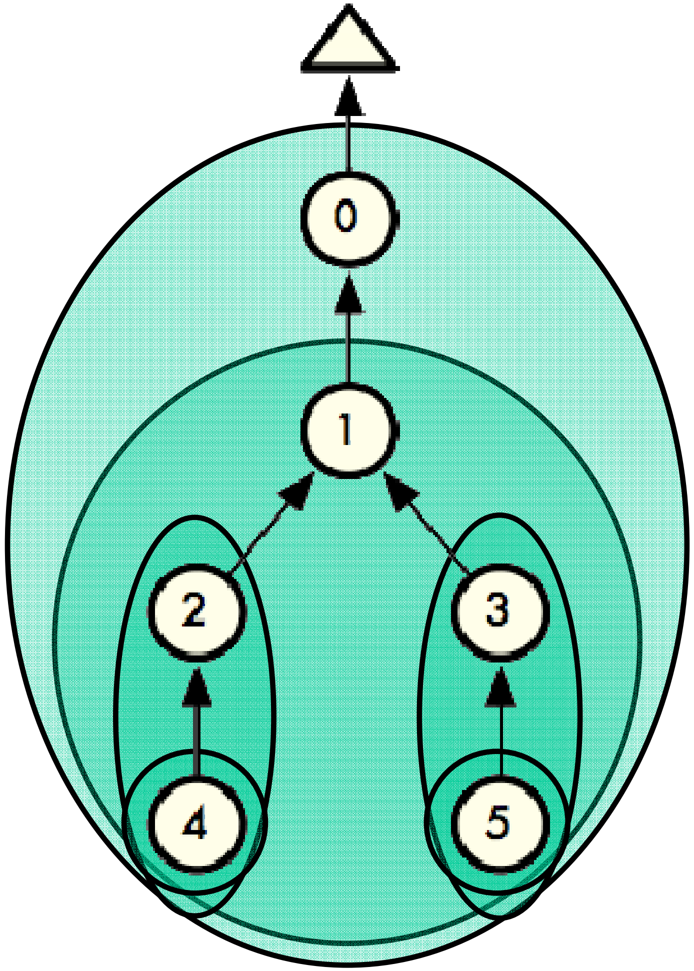
Monotone Clustering

- A clustering step is monotone, if
  - the metric-to-metric mapping exists
  - and is monotone
- This depends on…
  - which clusters are being combined
  - individual mapping functions (per quality metric)

Monotone Clustering

- Some properties depend on routing tree (e.g. delay, reliability)

- Following the tree achieves monotonicity



Centralised vs. Distributed Execution

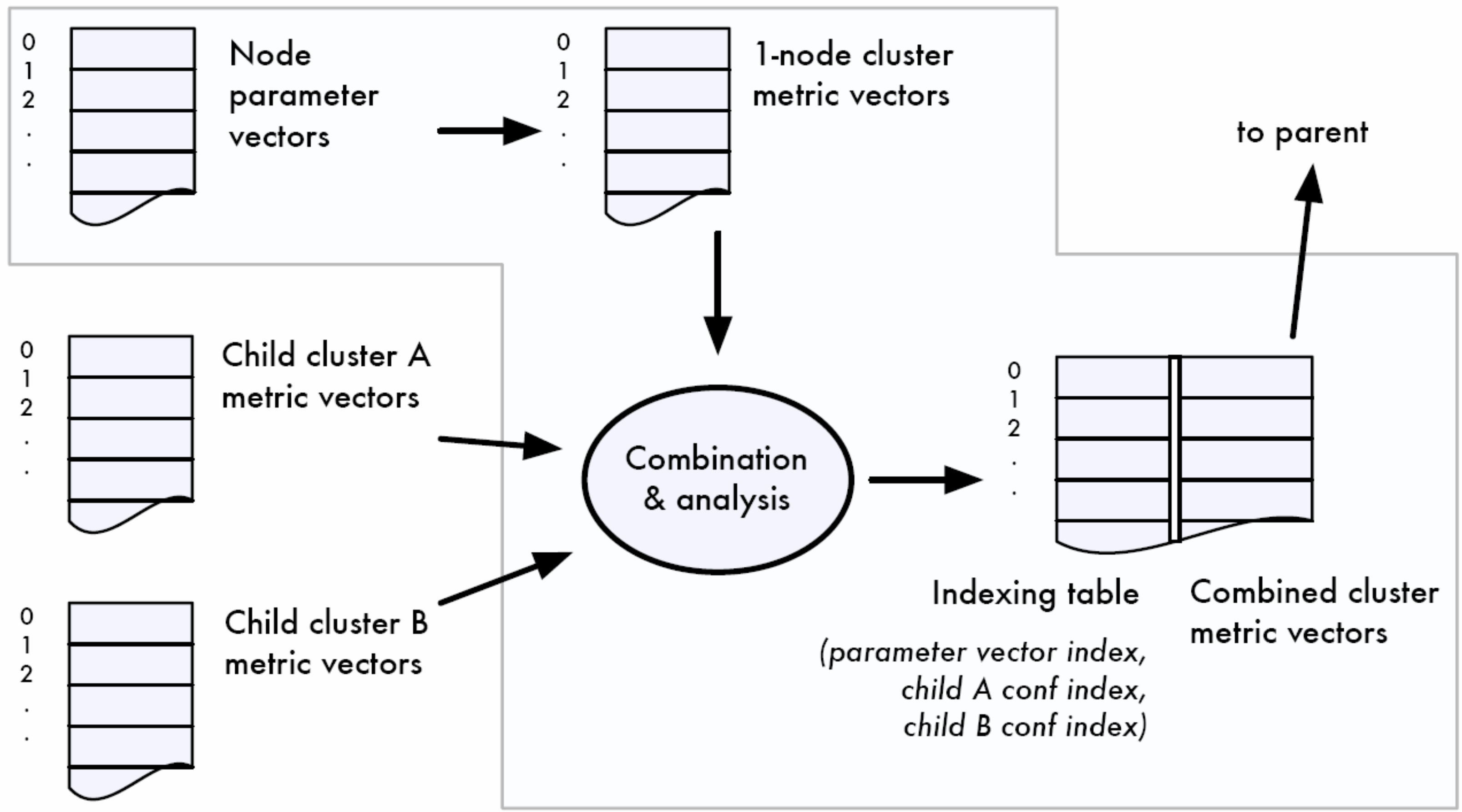
- Centralised
  - Running on powerful server
  - Needs information from whole network

- Distributed: running on the WSN itself
  - Parallel computation
  - Resource-constrained sensor nodes

Implementation

C++/Python and TinyOS

- Low memory demand
- Low communication overhead



Reducing the Run Time

- Processing time depends on
  - # Pareto points per cluster
  - # clusters combined

- Approximate Pareto sets
- Balance or limit node degrees in routing tree

Outline

- Quality of Service (QoS)
- Models
- QoS analysis
- Experiments
- Looking forward

Complexity Results

- Worst-case complexity: exponential

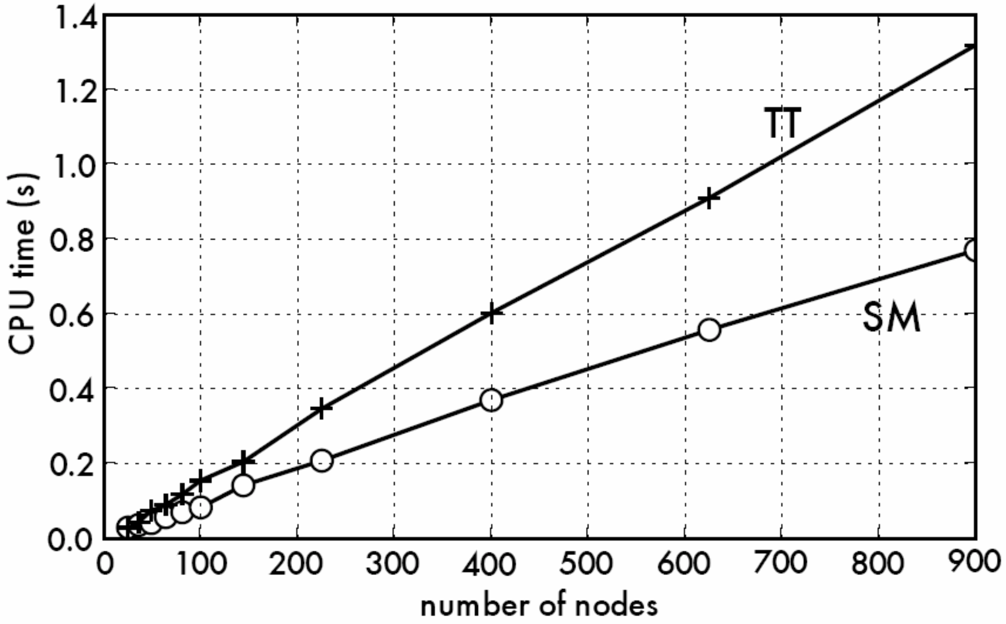
- Experimental results imply scalability in practice
  - Random networks
  - Varying number of nodes

- Example for 900 nodes (27^{900} configurations):
  - 12 Pareto configurations
  - Run time: 42 seconds
  - Configuration sets never larger than 1,111,320

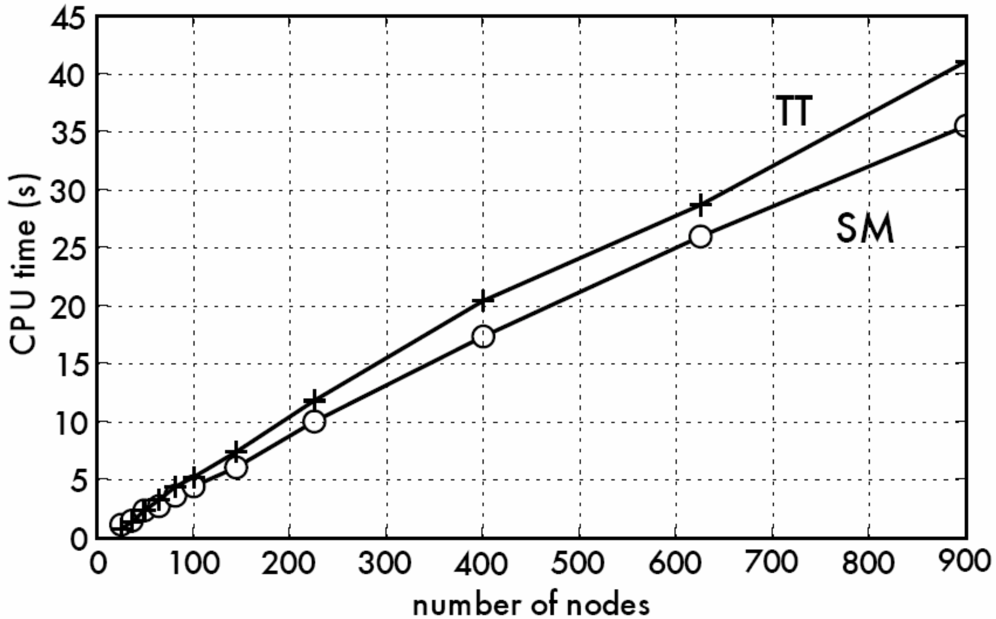
Run Time (centralised)

TU/e

8 configurations/node



27 configurations/node



Run Time (distributed, including communication)

- TelosB sensor nodes
- 900-node network, 8 confs/node
  - Without approximation: 25 min
  - With approximation: 5 min
- Quality loss due to approximation: <1%

Model Accuracy

- Compared model output with results of network simulations
- Deviations smaller than 6% on average

Genetic Algorithm

- The Genetic Algorithm ran for three days on a 900 node network
- Result: six “Pareto points”…

  … which were in fact not true Pareto points
- Our method finds all Pareto points within a minute

Outline

- Quality of Service (QoS)
- Models
- QoS analysis
- Experiments
- Looking forward

Summary

- WSN task models, which make trade-offs explicit

- QoS analysis for the WSN models
  - Analyses multiple QoS metrics at the same time
  - Scales linearly with network size in practice

What is Next …

- Complexity-control methods
- Routing-tree optimization
- Data-aggregation strategies
- Run-time adaptation
- Demonstrator

Thank you !

Questions ?

More info: www.es.ele.tue.nl/~tbasten/

> **‘An understanding of the natural world and what's in it is a source of not only a great curiosity but great fulfillment.’**
> **David Attenborough**