Sensor networks and applications

Network topology, routing

Topology

WSN topology

- Tipically, sensor networks are self-organizing, ad-hoc networks, the network structure „can not be planned”.

- The physical connections (links) are randomly created, resulting in a random (physical) topology.

- The logical topology construction can be important!
  - Taking into account scalability.

WSN topology

Distributed (flat) structure

- There is no constructed logical structure, all nodes take part in network control.
- Since the nodes have information only about their neighbors, this **information must be distributed within the whole network** from time to time.
  - E.g., every node broadcasts its routing table periodically.
- <u>Drawback: not scalable</u>! (e.g., for thousands of nodes...)

WSN topology

Clustering

- The network is divided into so-called **clusters**, so that each node belong to at least one cluster.
- All clusters have a **cluster head**, which controls all the nodes within its cluster.
- **Gateway** nodes proved communication between neighboring clusters.
- Scales better.
- <u>Problem</u>: All clusters know the neighboring clusters, but how can it be informed by the far away clusters?
  - The same problem on a higher hierarchical level!
- <u>Solution</u>: Cluster heads are organized into a hierarchic tree.

WSN topology

Clustering (cont'd)

- Optimal clustering is not easy
  - „NP-hard", but there are $O(n^2)$ heuristics
- Another problem is that theload on the cluster heads can be much higher than on ordinary nodes.
  - Solution: The cluster head role is changed from time to time.

Next: Routing

- Routing planning issues
- Modeling the network
- Structure-based solutions
  - flat
  - hierarchic
  - location-based

Content

- Medium Access Control
  - S-MAC

- Networking layer
  - WSN topology
  - Routing

WSN features

- High number of network nodes:
  - **Global addressing is not possible**, because unique ID can not be provided to all the nodes.
  - Consequence: The „traditional” IP-based protocols can not be applied.

- Sensors are deployed in an ad-hoc way:
  - The network must be **self-organizing**.
  - **Random node distribution** must be dealt with.
  - **Self-managing** is required.

WSN features

- Typically, sensor nodes are static after deployment.
  - In contrast to mobile ad-hoc networks, where the nodes can move freely.
  - Some nodes can be mobile according to the given application (but with low mobility, typically).

- Typically, the information is flowing from many sources (sensors) to a single node (base station).
  - It can be multicast or peer-to-peer as well!

WSN features

- Sensors have limited energy, computing- and storage capacity.
  - Efficient **recource-management** is needed.

- Sensor networks are application specific.
  - The design requirements are different for different application areas. (E.g., high precision monitoring systems vs. periodic wheather monitoring.)

WSN features

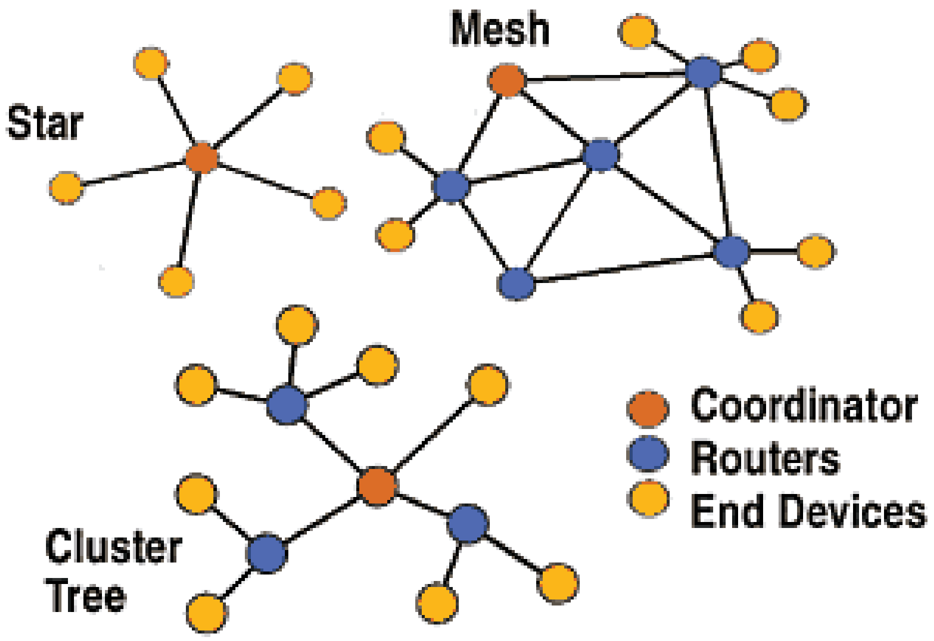
- The **location aware operation** is important, because data collection is location dependent.
  - Localization based on triangulation using reference points.
- The collected data are **reduntand**.
  - E.g., more sensors measure and collect data about the same phenomenon.
  - Redundance can be exploited during routing.
- Most WSNs are **data centric**.
  - We are looking for data that are attribute-based. (E.g., where is the temperature higher than 40 degrees?)

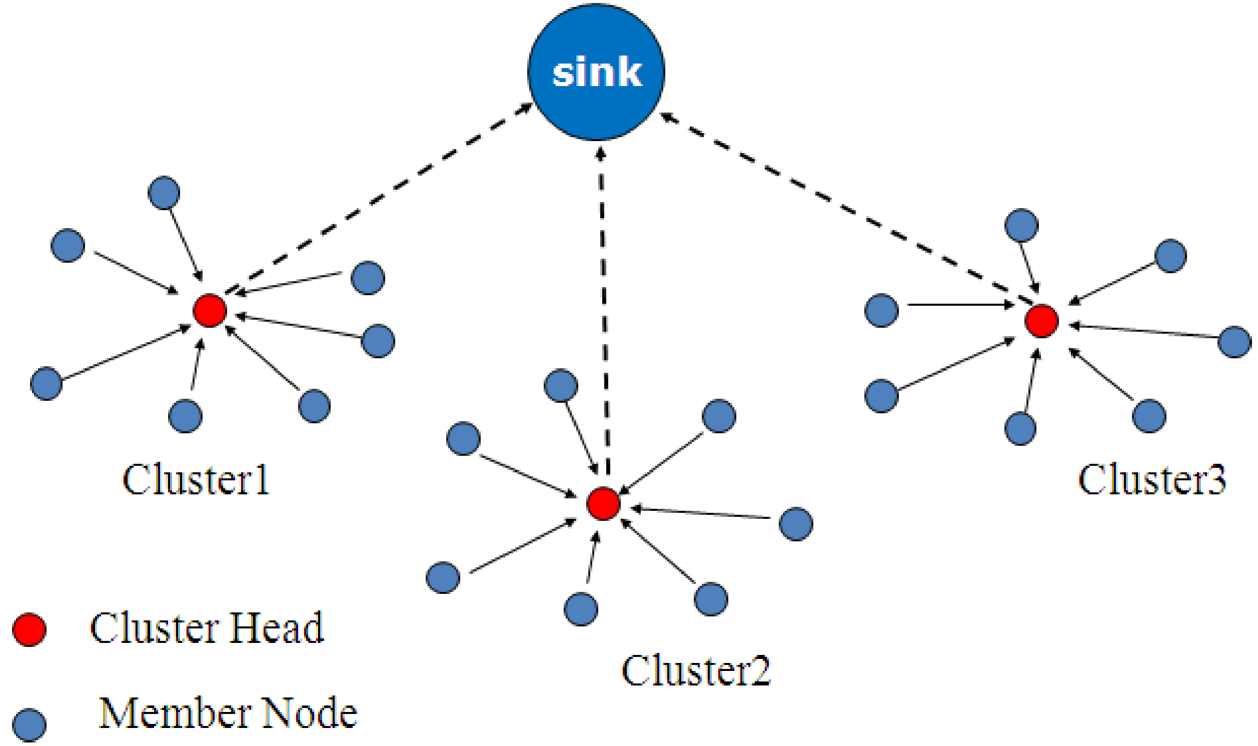
Next: Routing

- Routing design issues
- Network and routing modeling
- Network topology-based solutions
  - Flat
  - Hierarchic
  - Location based

Routing

WSN topology – clustering





Routing design issues in WSNs

- Csomópontok telepítése: **deterministic** or **random**
  - Deterministic: nodes are deployed manually, the data traffic is carried by routes that are set up in advance.
  - Random: the infrastructure is set up in an ad-hoc way.

- **Energy efficiency** without loosing reliability and accuracy.
  - Dual role of nodes: data source and routing.

OKOS VÁROS @ BME TMIT

Routing design issues in WSNs

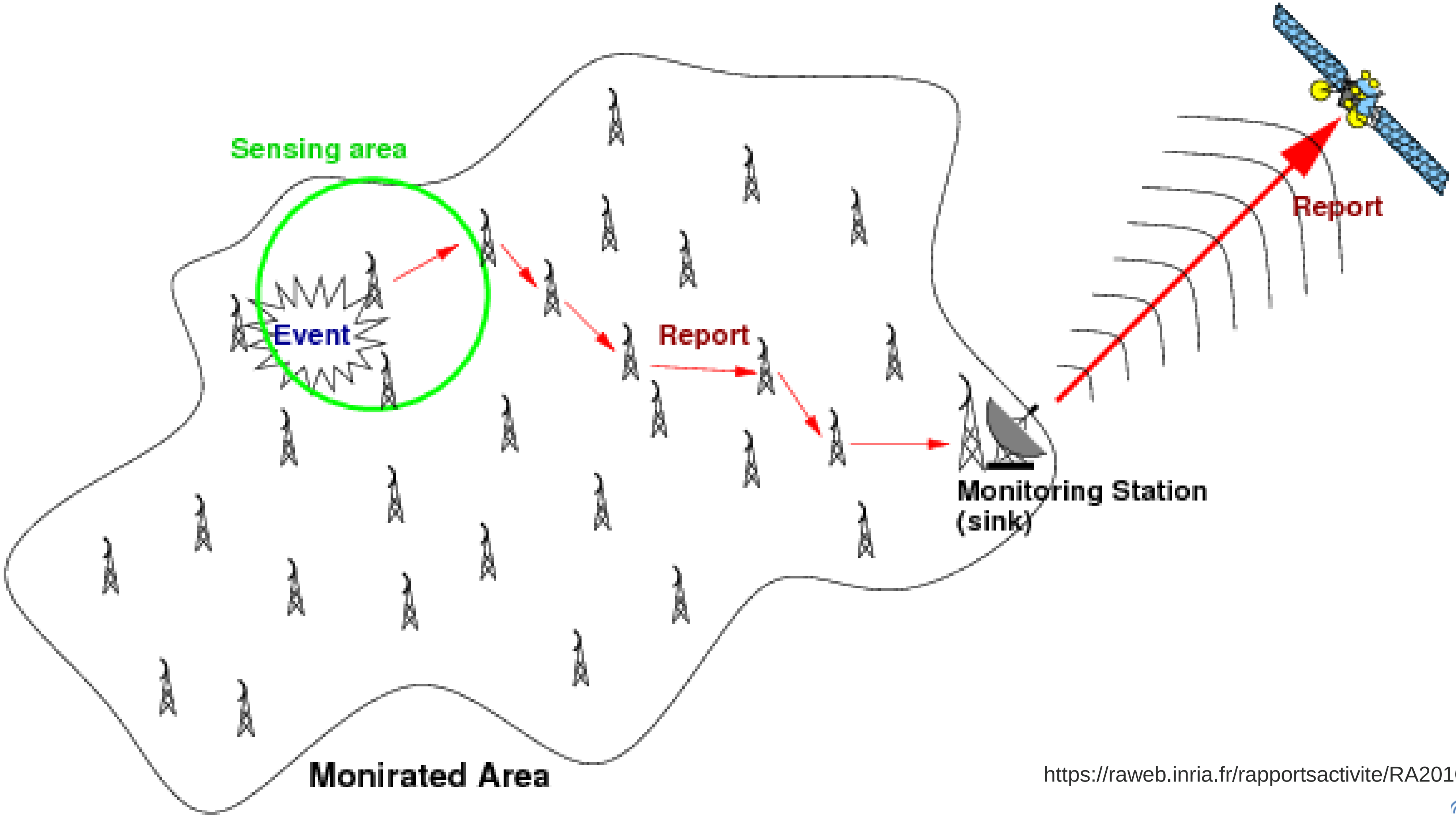
- **Data model**
  - Information sensing and data transmission is application specific.

  Sensing and data transmission can be

  - **time-driven** (continuous)
    - e.g., periodic monitoring
    - Sensrs become active from time to time, sense and transmit data.
  - **event-driven**
    - Sensors react to sudden changes in the environment.
    - can be important in time criticl applications.
  - **query-driven**
    - Nodes answer to a query of a „controller”

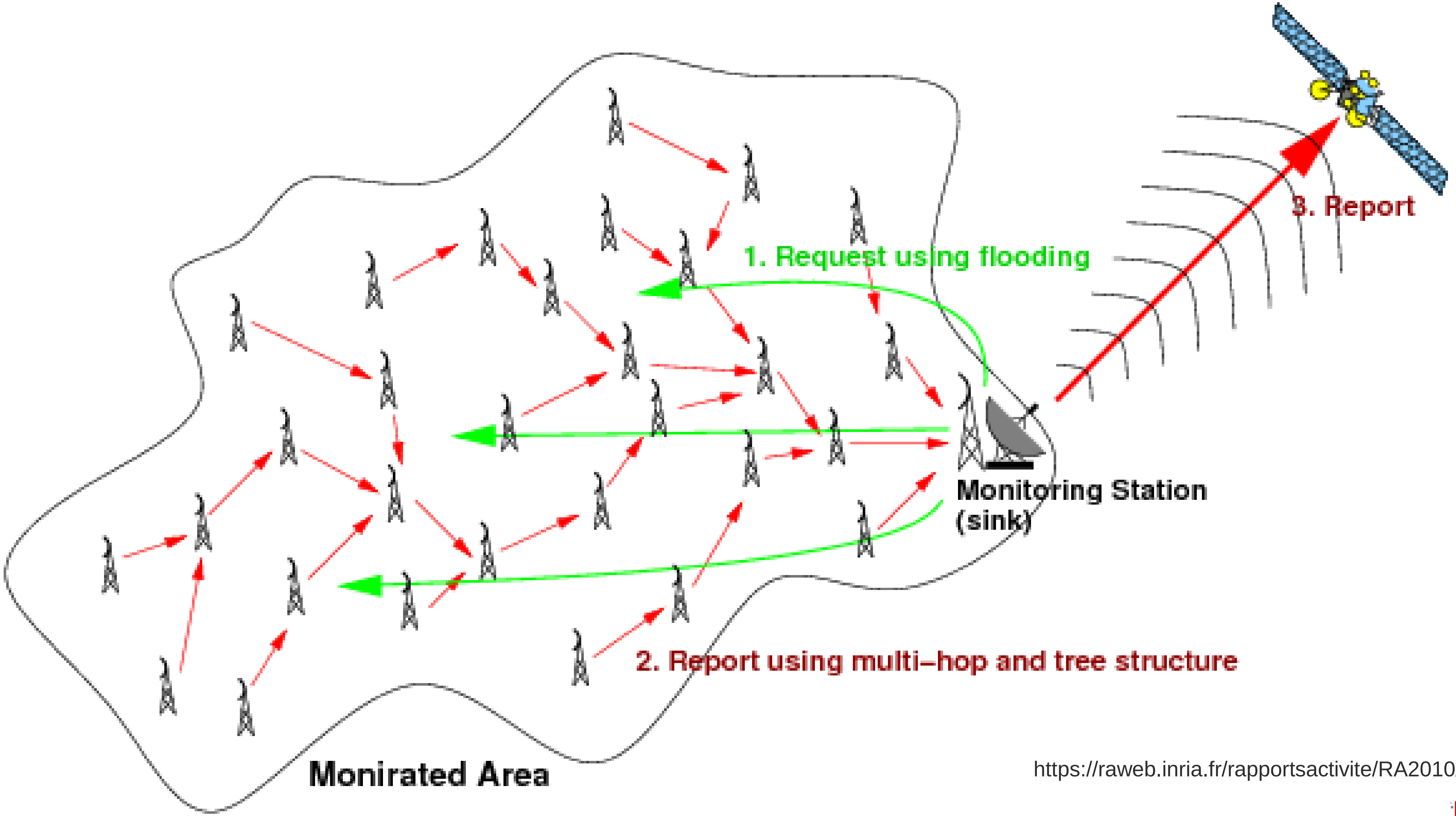
  - ***Routing heavily depends on the data model used!***

event-driven network (example)



https://raweb.inria.fr/rapportsactivite/RA2010/pops/IMG/fig1.png

Query-driven network (example)



https://raweb.inria.fr/rapportsactivite/RA2010/pops/IMG/fig1b.png

Routing design issues in WSNs

- Node/link **heterogeneity**
  - In most cases the sensor nodes are identical (**homogeneous**)
  - Afew nodes can be different according to the application.
    - E.g., they sense different physical things (video, noise, temperature).
  - The **special sensors** can be deployed differetly.
  - Data generation and transmission speeds can be different as well (e.g., temperature vs video).
  - Different data can require **different QoS** treatment.
  - E.g., **cluster heads** can be well positioned, stronger nodes with higher energy.

Routing design issues in WSNs

- **Robustness**
  - Any sensor node can drop out from the network for different reasons (e.g., interference, physical damage, energy depletion).
  - Loosing a single node must not violate the operation of the whole application.
  - The MAC and routing protocols must be able to deal with the loss of <u>more</u> nodes.
  - **Multi-level redundancy** is necessary for building a sensor network.

- **Scalability**
  - The number of sensors in a WSN can be hundreds or thousands(!).
  - E.g., as a consequence, the implemented routing table within a node can not contain all routing information to any other node.

Routing design issues in WSNs

- Network **dynamics**
  - In most of the cases the nodes are **static**, they don't move.
  - For **mobile nodes** (e.g., mobile BS) the mobility handling can be a great challenge in routing.
    - E.g., the network topology changes in time, ...
  - The sensed event itself can be static or dynamic as well. (e.g., temperature vs. target tracking)
- **Connectivity**
  - Typically, the **degree of a network node is high** due to the high node density.
  - Connectivity can change in time (e.g., nodes deplete their energy sources).

Routing design issues in WSNs

- **Coverage**
  - The sensors can monitor only its close environment with adequate accuracy.
  - The full area coverage (and keeping the coverage in time) is an important target in planning.

- **Data aggregation**
  - The sensors generate and send data with high redundancy.
    (E.g., temperature values at nearby locations)
  - The amount of data to send can be significantly reduced by extracting the valuable information from raw data.
    - E.g., minimum, maximum, average
  - By data aggregation higher energy efficiency can be reached.
  - Signal processing techniques can alo be applied for data compression.

Routing design issues in WSNs

- **Quality of Service (QoS)**
  - E.g., the measured data should be sent within a time frame, otherwise it will be outdated. → For time critical applications the delays must be bounded.
  - For most of the applications, the energy efficiency is more immportant than small delay.
  - E.g., Idea: The data quality can be decreased linearly with energy depletion.

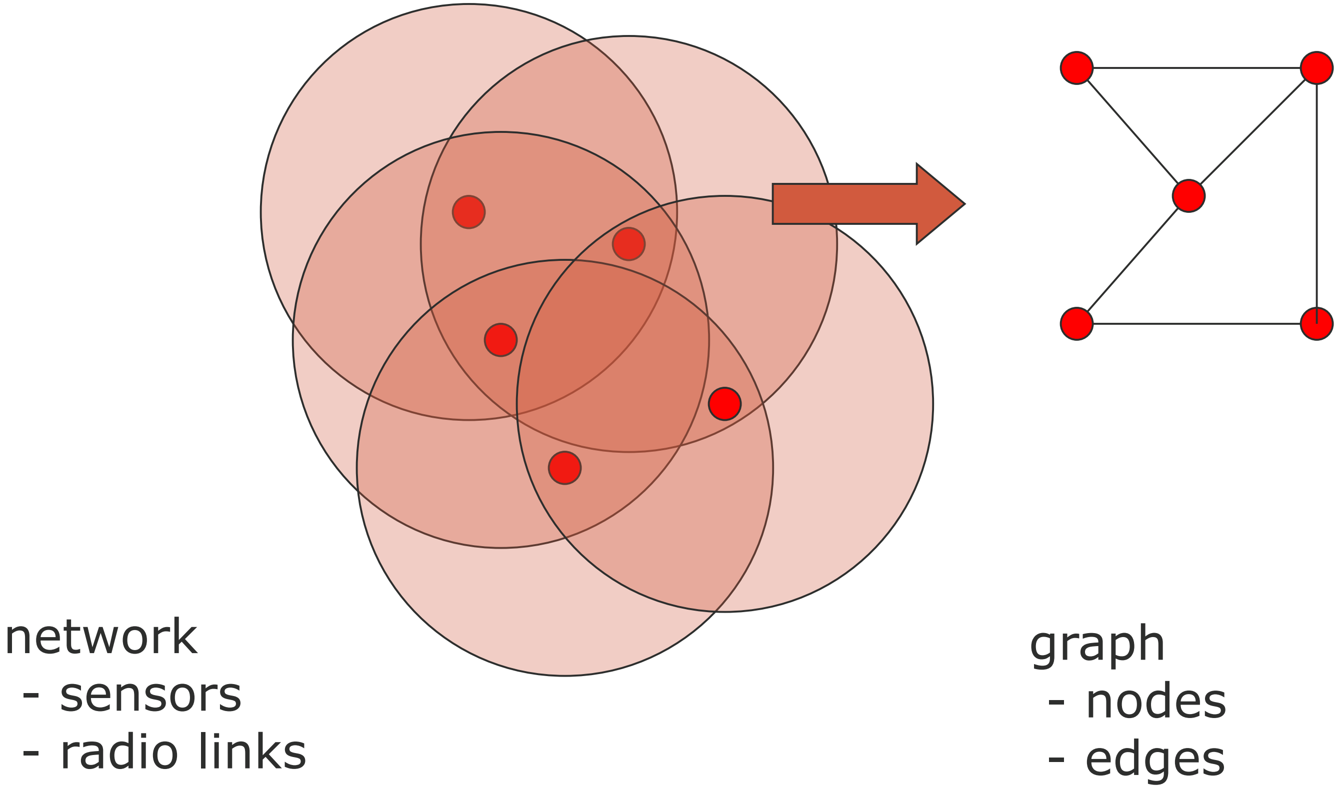
- Energy-aware routing is the key is WSNs!

Network and routing modeling

Graphs, algorithms

Modeling network and routing

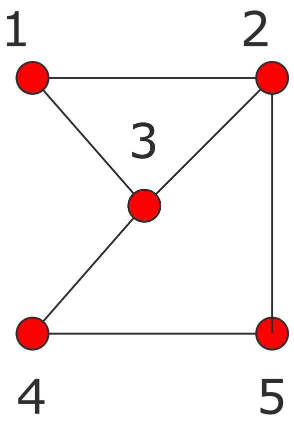
- Mathematical model:
  - network -> graph
  - routing algorithms -> graph theory algoritms



Modeling network and routing

- Network graph
  - Edges can be directed or undirected
    - E.g., a station does not take part in message relaying
  - The notion of distance can be used between any two nodes
    - E.g., the „cost” or „quality” of the link can be given

- Notations: *G*(*N*, *A*), where *G* is the graph and *N is the set of nodes*, *A is the set of edges*.



$N=\{1,2,3,4,5\}$

$A=\{(1,2), (1,3), (2,3), (3,4), (2,4), (2,5)\}$

Network graph, definitions

- path:

$$\left(n_1, n_2, \ldots, n_m\right), n_i \in N,\ (n_i, n_{i+1}) \in A$$

- route: Path where none of the edges are repeated.
- The graph is connected, if there exists a route from each node to any other node.
- loop: route, where $n_1 = n_m$
- tree: connected graph without loops.
- spanning tree: tree containing all the nodes.

Example: Finding the shortest path

- <u>Task</u>: Find the shortest path in graph *G*(*N*, *A*) from *s* (source) to *t* (target).
- Edges are labeled with weights:

$$d_{i,j} \in R \Leftrightarrow (i,j) \in A$$
$$d_{i,j} = \infty \Leftrightarrow (i,j) \notin A$$
$$d_{i,i} = 0$$

- Find the *L* path between *s* and *t* where the sum of weights is minimal (shortest path).
  - One possible solution: Dijkstra algorithm

Example: Dijkstra algorithm

Notation:

- D_i = iterated distance of node *i* from *s* (*s*=„1”)
- *P* = set of considered nodes

Algorithm:

1. <u>initialization</u>: $P = \{1\}$; $D_j = d_{1j}$ for all *j*
2. <u>iteration</u>: find *i* where $$D_i = \min_{\forall j \notin P} D_j$$
3. let: $$P = P \cup \{i\}$$ $$D_j = \min\{D_j, d_{ij} + D_i\}$$
4. stop condition: If $N = P$, STOP
   else GOTO 2.