IRCICA

Institut de Recherche sur les Composants logiciels et matériels pour l'Information et la Communication Avancée

Energy savings in wireless sensor networks

David SIMPLOT-RYL

Centre de recherche INRIA Lille – Nord Europe
IRCICA/LIFL, CNRS 8022, Université de Lille 1

http://www.lifl.fr/~simplot
simplot@lifl.fr

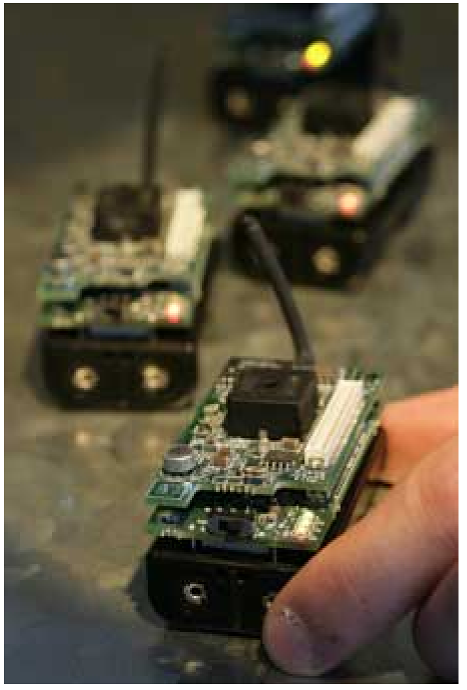
USTL CNRS INRIA FUTURS LIFL

Outlines

I. Wireless Sensor Networks

II. Activity scheduling and coverage problem

III. Routing with guaranteed delivery

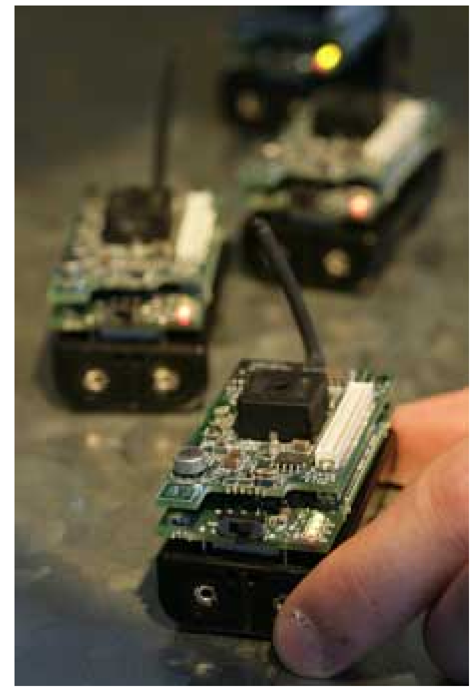


Outlines

I. Wireless Sensor Networks

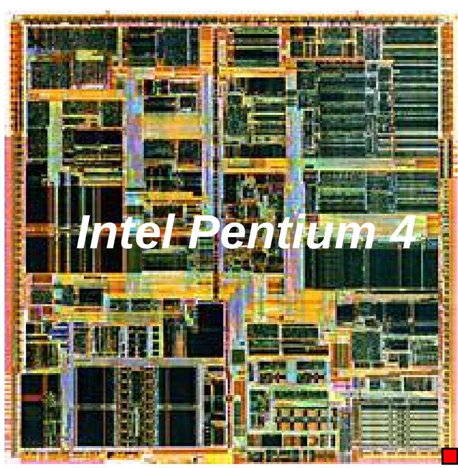
II. Activity scheduling and coverage problem

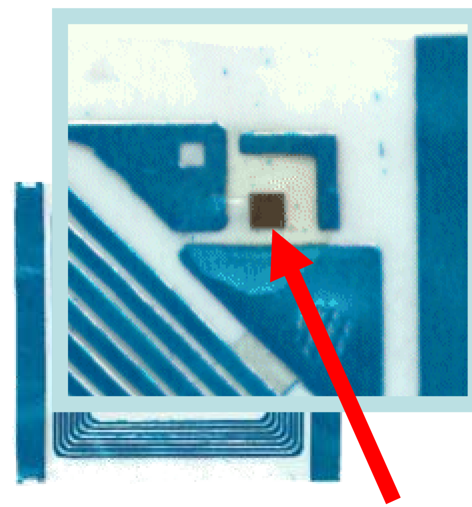
III. Routing with guaranteed delivery



RFID Tags

- **Smart labels**
  - Radio Frequency Identification Tag
  - By opposition to **bar code** which use **optical** principles
- **A stronly limited component:**
  - 500 times smaller than a classical microprocessor





EAS Application

- **Electronic Article Surveillance**
  - Once powered, the tag emits
  - The reader listen channel and activate alarm as early as transmission is detected
  - During checkout, the tag is burned out
  - Problem: power and hear the tag whatever the tag orientation

Applications

- **Batch identification**
  - It is the capability to collect information from a set of tags
  - In opposition to optical identification

Marathon
Automatic clocking in

Automatic luggage sorting

Automatic inventory
50 items in less than one second

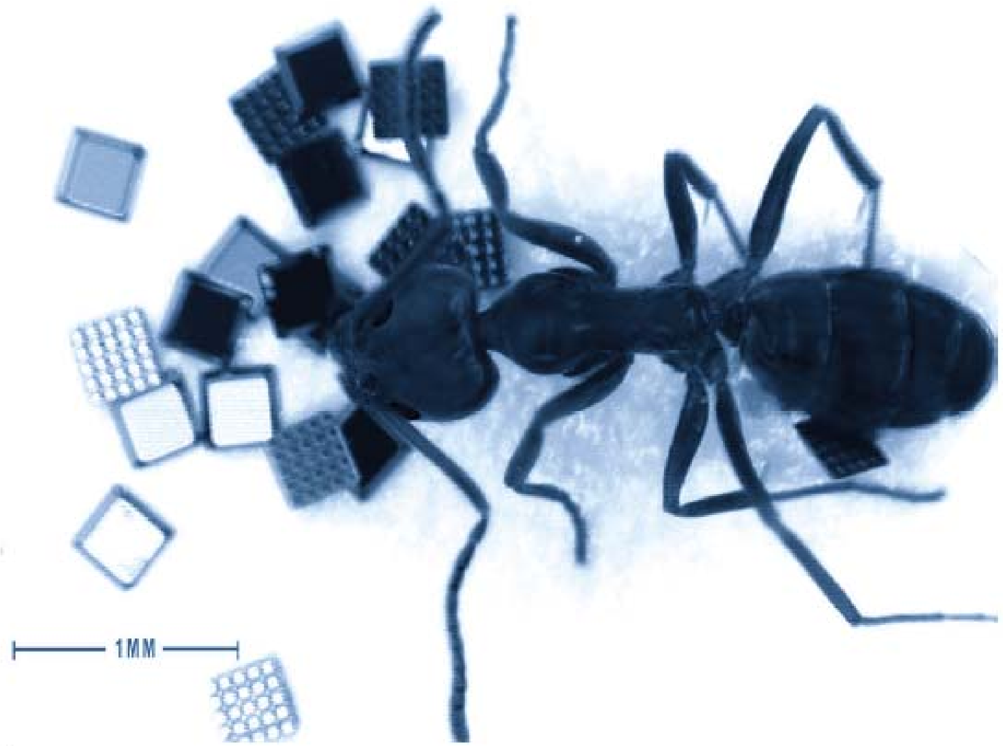
More POPS, smaller POPS...

Courtesy, Alien Technology

POPS = Portable Objects Proved to be Safe

POPS = Petits Objets Portables et Sécurisés

USTL CNRS INRIA LIFL

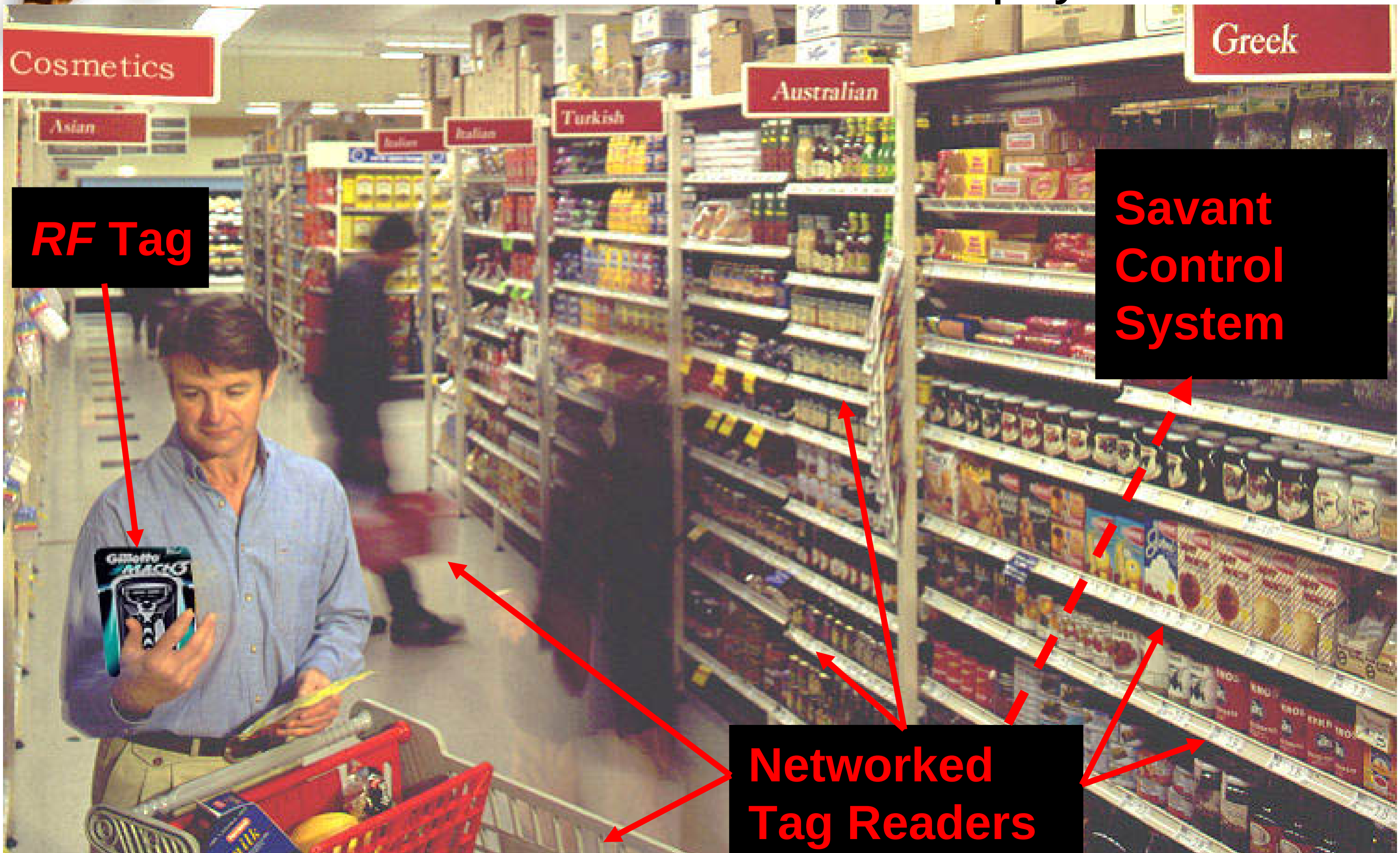


Courtesy, Auto-ID Center

The MIT Auto-ID Center Vision of "the internet of things"

IRCICA Institut de Recherche sur les Composants logiciels et matériels pour l'Information et la Communication Avancée

Networking the physical world



Auto-ID Center classification

Class V tags
Readers. Can power other Class I, II and III tags;
Communicate with Classes IV and V.

Class IV tags:
Active tags with
broad-band peer-to-peer communication

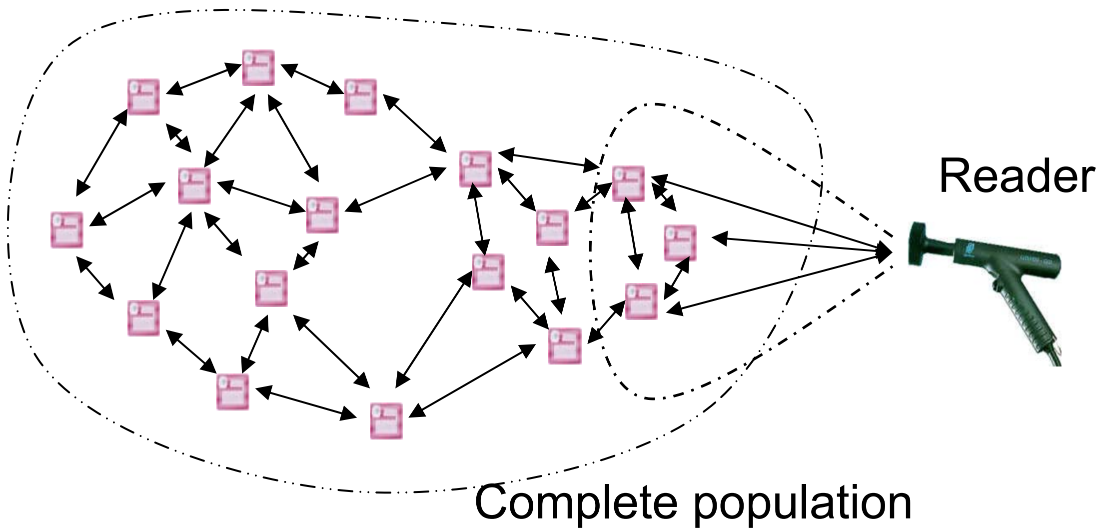
Class III tags:
semi-passive RFID tags

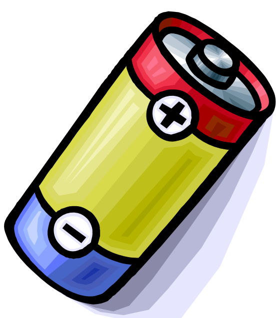
Class II tags:
passive tags with additional
functionality

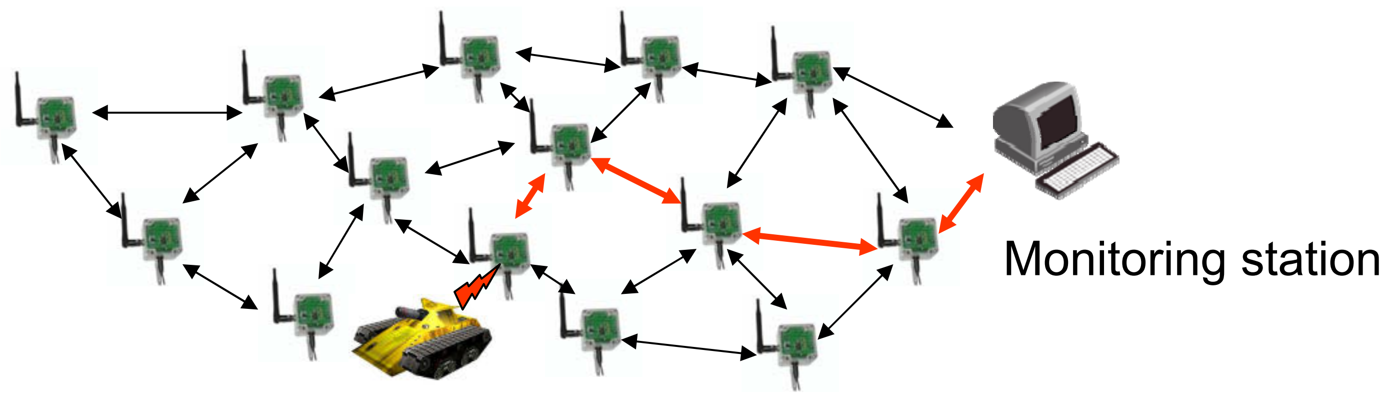
Class 0/Class I:
read-only passive tags

Benefits of class IV tags

- **Decentralized behavior**
  - The request is broadcasted in the whole network by using multi-hop method
- **Similar to sensor networks**





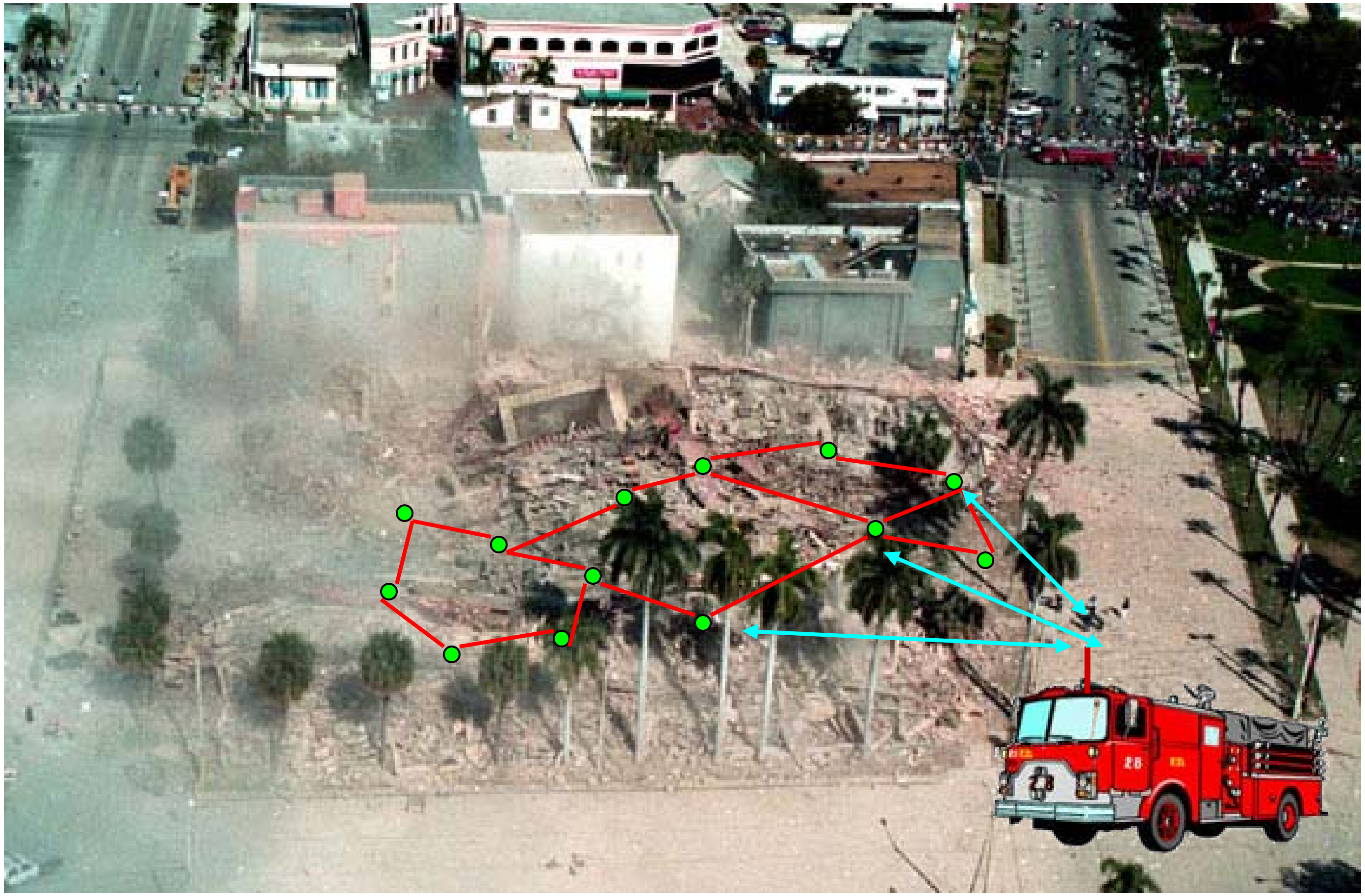


Sensor Nets for Search and Rescue

● **Inactive Sensor**

Sensor Nets for Search and Rescue

Sensor Nets for Search and Rescue

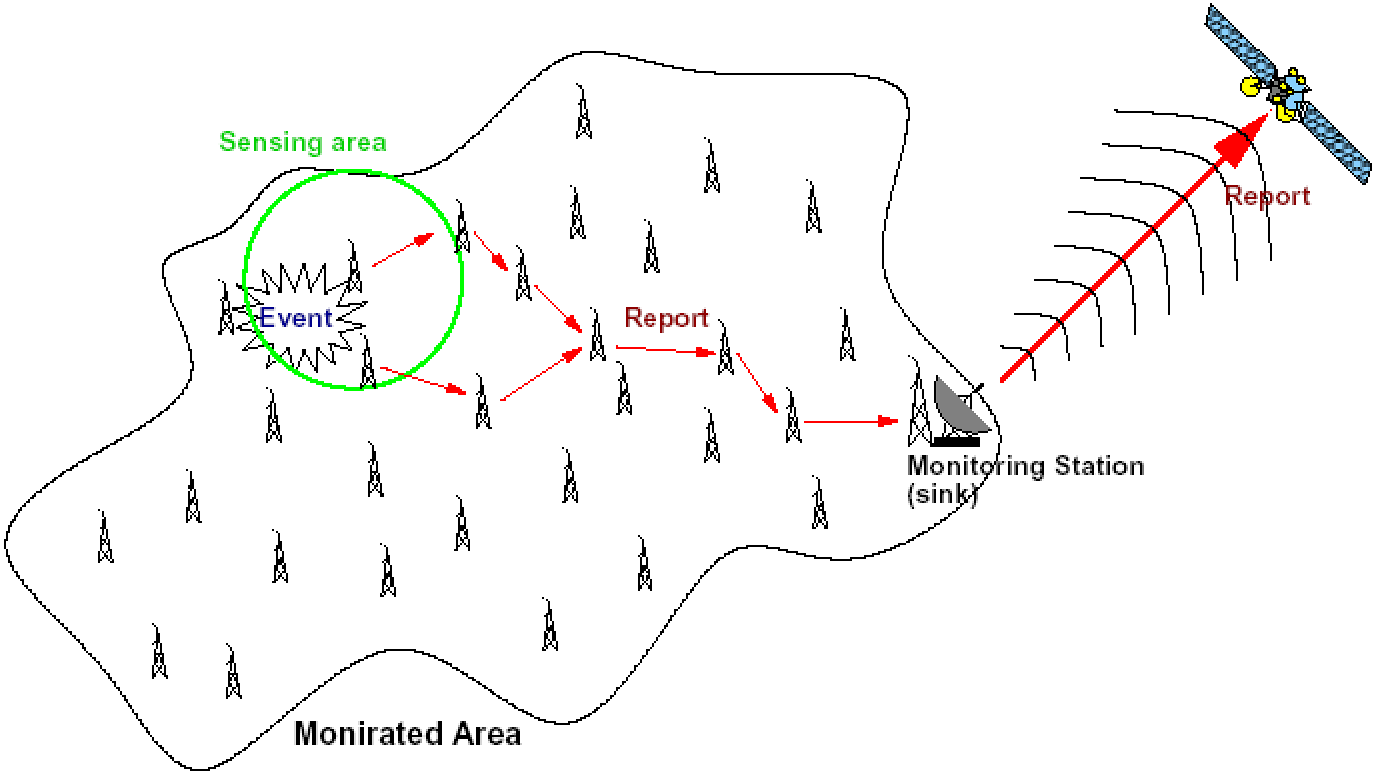


● **Active Sensor**

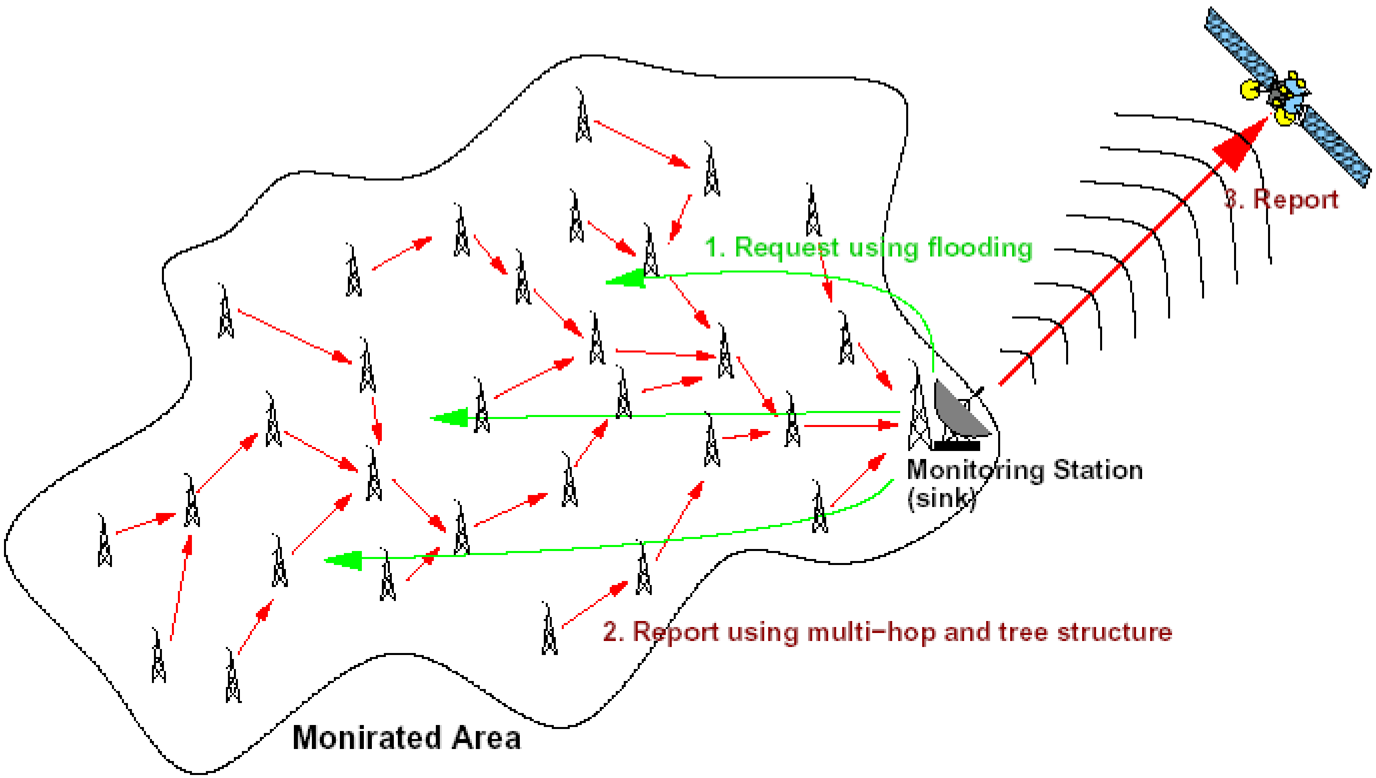
Application

- **In the UC Berkeley Botanical Garden, 50 "micromotes" sensors are dangled like earrings from the branches of 3 redwood trees to monitor their growth.**

Event-driven model

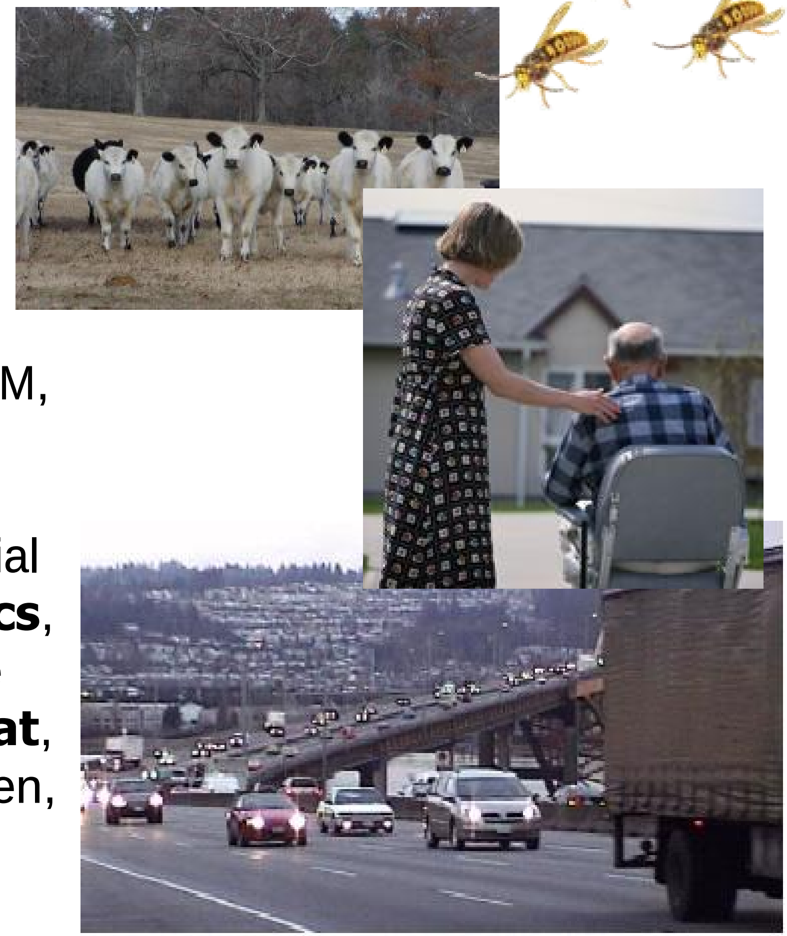


On-demand model



WASP Project

Wireless Accessible Sensor Populations

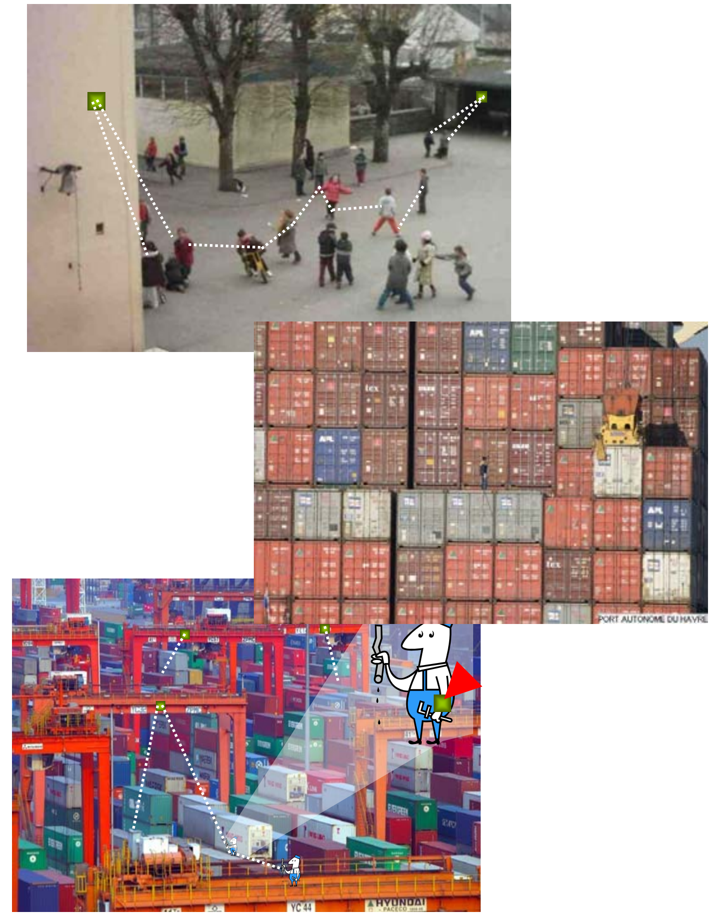


Philips Research Eindhoven, Philips Forschung Laboratorium, IMEC, CSEM, TU/e, **Microsoft** Aachen, Health Telematic Network, Fraunhofer IIS, Fokus, IGD, Wageningen UR, Imperial College London, **STMicroelectronics**, INRIA, Ecole Polytechnique Federale Lausanne, Cefriel, Centro Ricerce **Fiat**, Malaerdalen University, RWTH Aachen, **SAP**, Univ of Paderborn

SVP Project

Surveiller et Prévenir

CEA, INRIA, Institut Maupertuis, Aphycare, LIP6, M2S, Thales, ANACT



Dispositif expérimental

Un Anthony motivé par la recherche !

Ceinture POLAR pour la mesure ambulatoire de la FC

VO2000 : Mesure ambulatoire des échanges gazeux

5 capteurs Séréo'Z sans fil

Tapis déroulant
Vélo à effort paramétrable
Sac à dos chargé

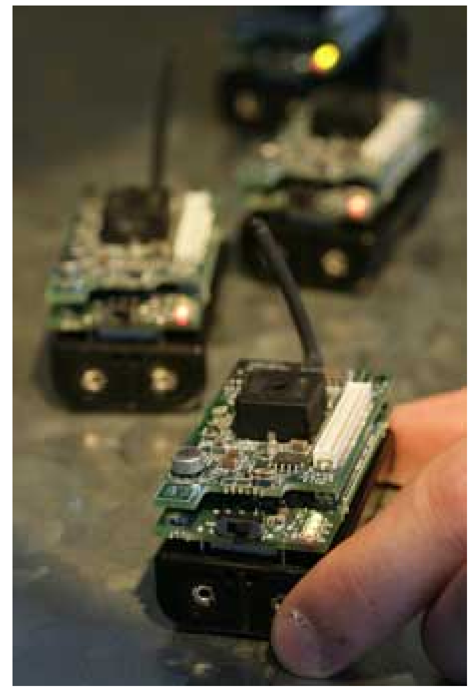
Outlines

I. Wireless Sensor Networks

II. Activity scheduling and coverage problem

III. Key distribution in wireless sensor networks

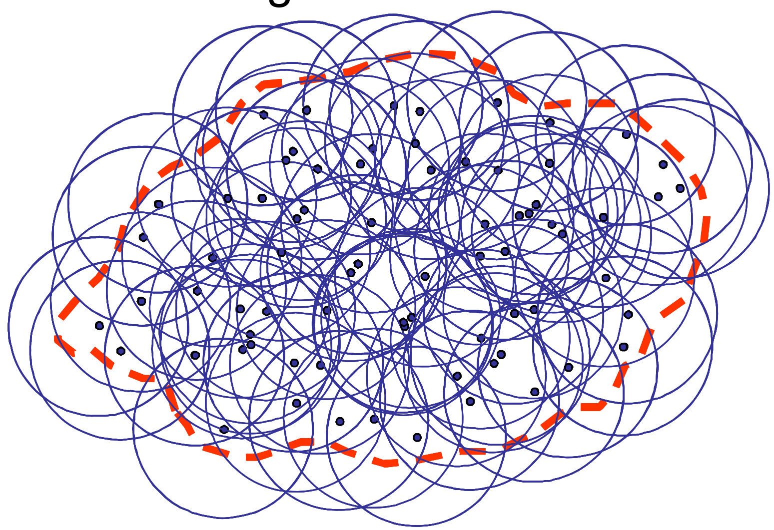
IV. Routing with guaranteed delivery



Sensor Networks

Heavy energetical constraints

- Unable to change or reload batteries



Idea: "Turn redundant sensor nodes to sleep"

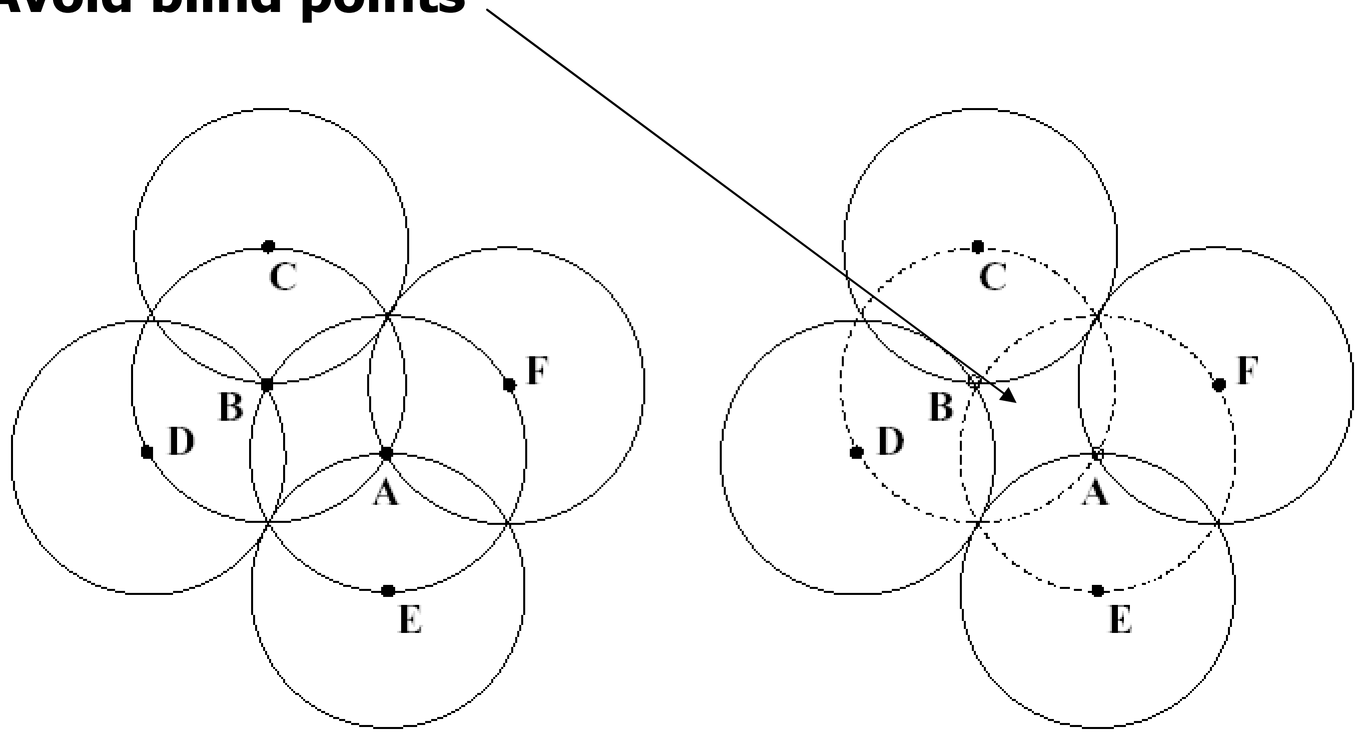
=> Nodes periodically turn into sleep or active mode
=> Need for **activity scheduling** and **self-organization**
=> ***Connectivity*** and ***area coverage*** must be preserved

Sensor Coverage

- **How well do the sensors observe the physical space**
  - Sensor deployment: random vs. deterministic
  - Sensor coverage: point vs. area
  - Coverage algorithms: centralized, distributed, or localized
  - Sensing & communication range
  - Additional requirements: energy-efficiency and connectivity
  - Objective: maximum network lifetime or minimum number of sensors
- **Area (point)-dominating set**
  - A small subset of sensor nodes that covers the monitored area (targets)
  - Nodes not belonging to this set do not participate in the monitoring – they sleep
- **Localized solutions**
  - With and without neighborhood information

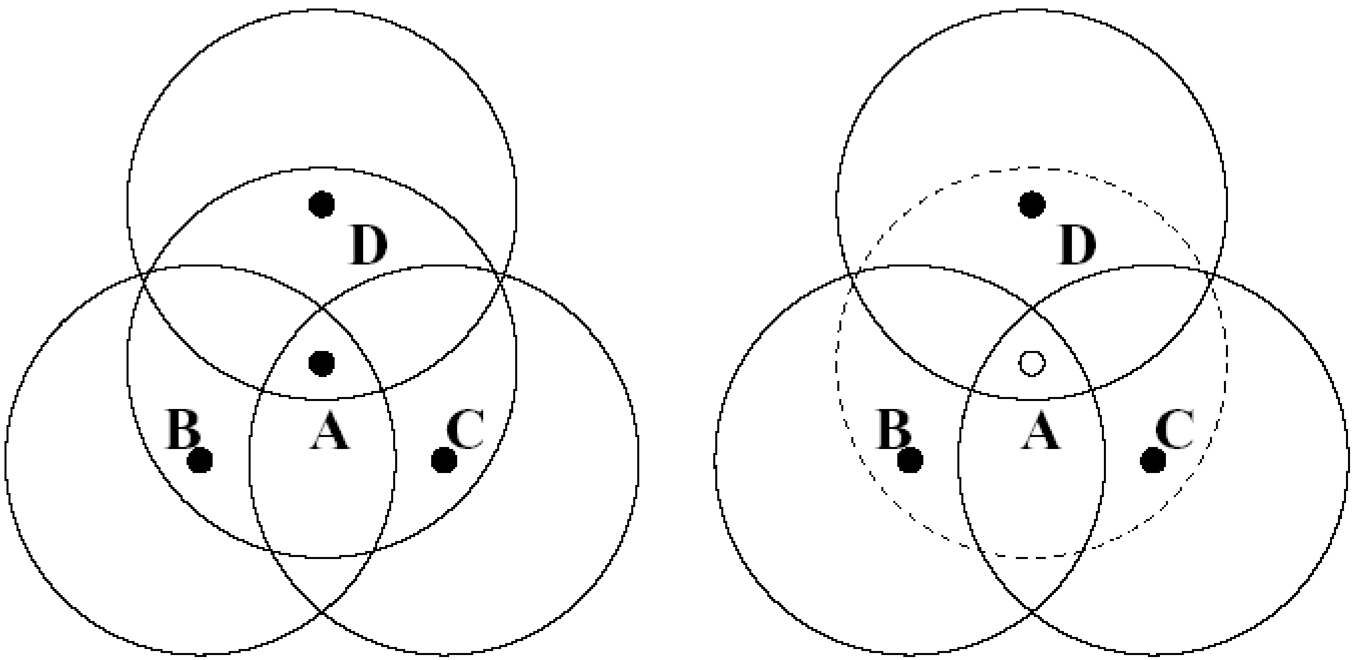
Sensor coverage (2)

- **Avoid blind points**



Sensor coverage (3)

- **Avoid lost of connectivity**



Area-dominating set

- **With neighborhood info [Tian and Geoganas, 2002]**
  - Each node knows all its neighbors' positions.
  - Each node selects a random timeout interval.
  - At timeout, if a node sees that neighbors who have not yet sent any messages together cover its area, it transmits a "withdrawal" and goes to sleep
  - Otherwise, the node remains active but does not transmit any message

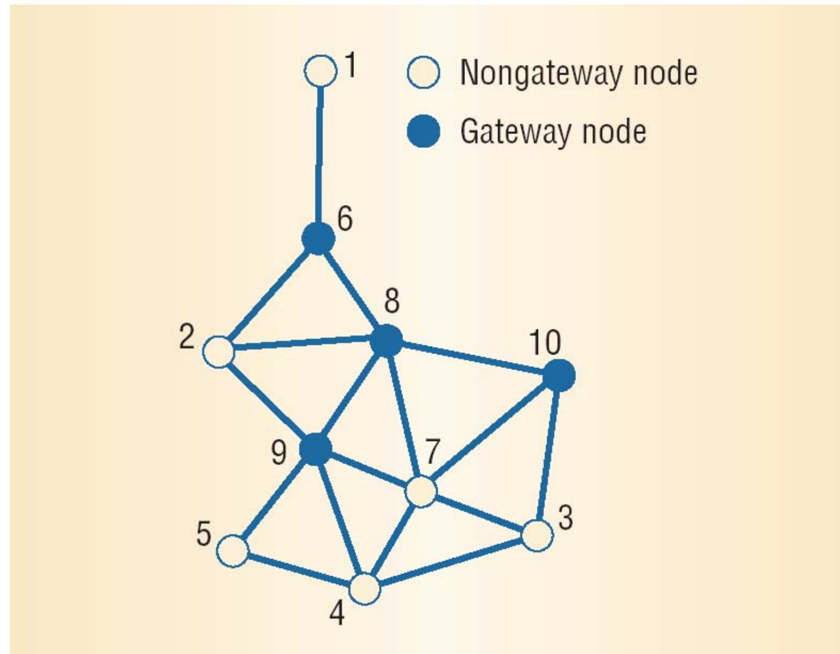
- **With neighborhood info based on Dai and Wu's Rule k [Carle and Simplot-Ryl, 2004]**
  - Each node knows either 2- or 3-hop neighborhood topology information
  - A node u is fully covered by a subset S of its neighbors iff three conditions hold
    - ✓ The subset S is connected.
    - ✓ Any neighbor of u is a neighbor of S.
    - ✓ All nodes in S have higher priority than u.

Wu & Dai dominating sets

- (Simplified version [Carle and Simplot-Ryl 2004])

- **Each node has a priority (*e.g.* its ID)**
  - Variations:
    <degree, ID>
    <battery, degree, ID>
    <random, battery, degree, ID>
    …

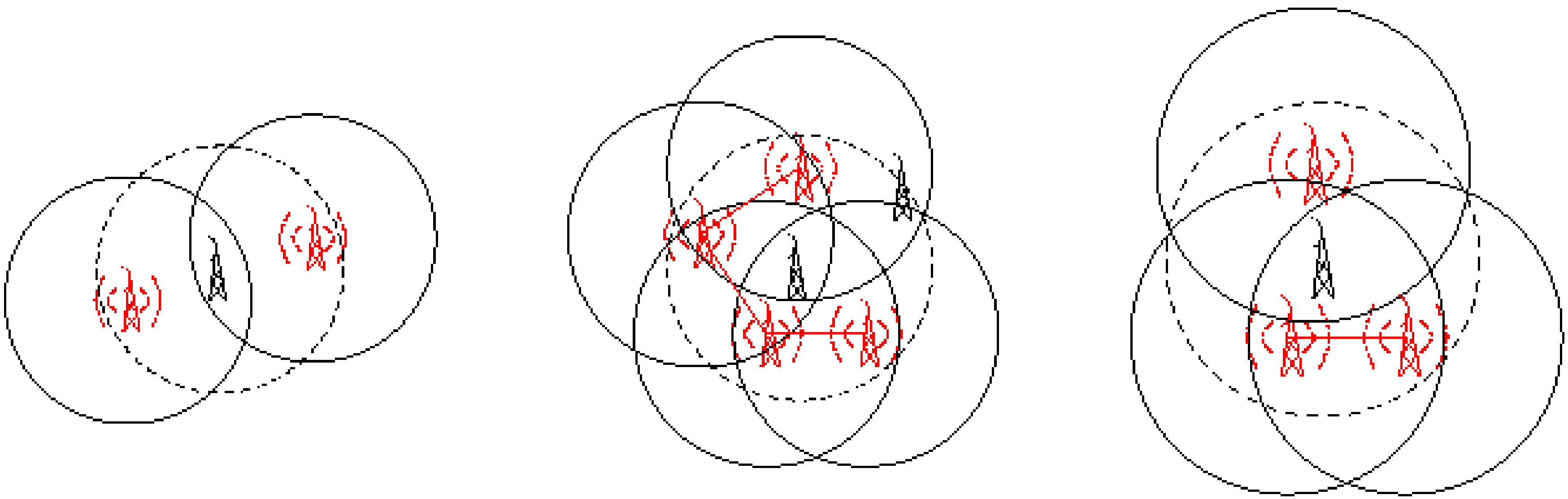
- **A node is not dominant iff**
  - The set of neighbors with higher priority is connected and covers the neighborhood



Connected area dominating sets

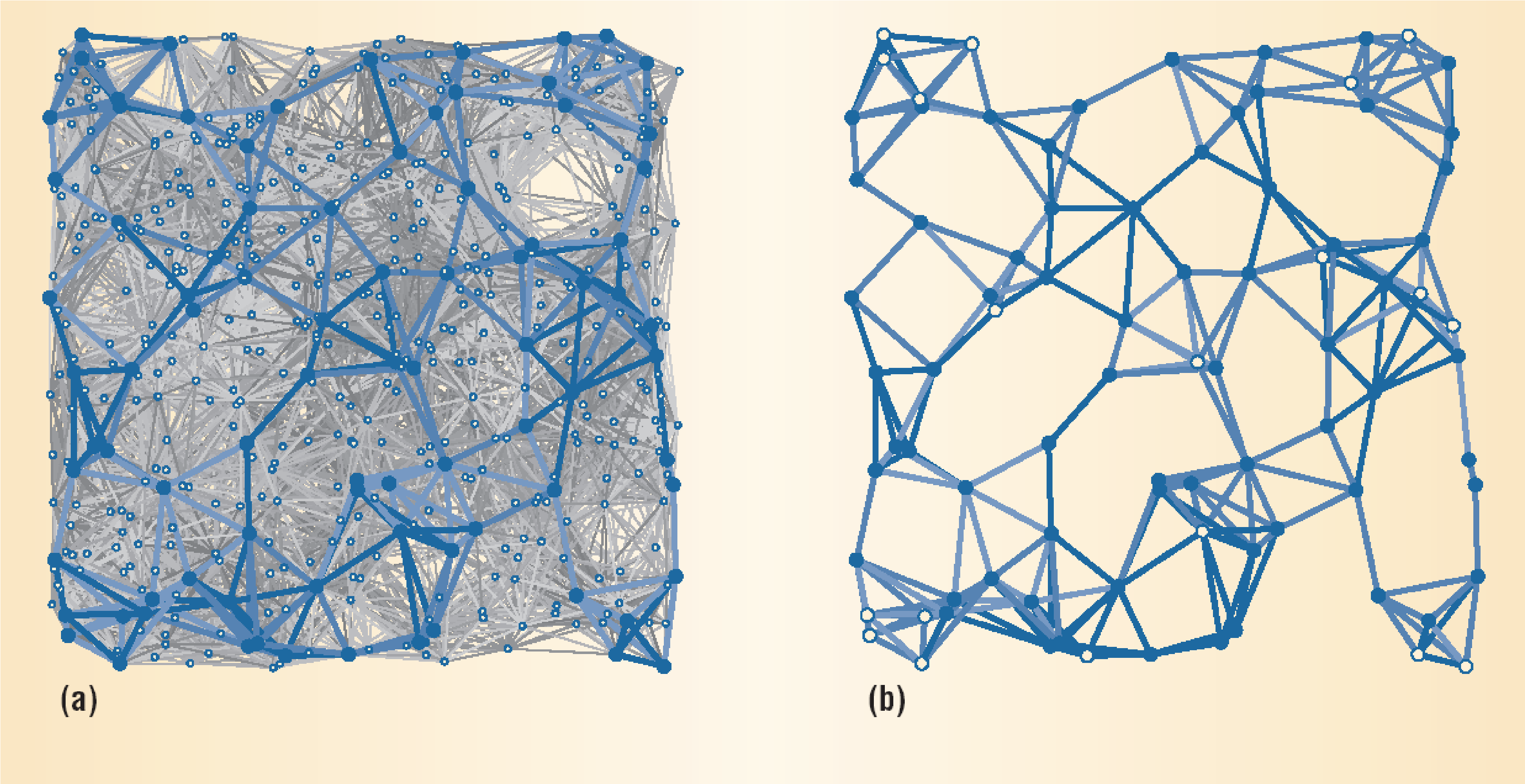
[Carle, Simplot-Ryl 2004][Gallais et al. 2006]

- **We change notion of coverage**
  - A node u is covered by a subset A of its neighborhood if the monitoring area of u is covered by nodes of A
- **Remaining battery is used as priority**
- **Timeout is used to inform transmit priority**
- **Examples:**



Connected area dominating sets (2)

- **We obtain a Connected** Area **Dominating Set**



Coverage without neighborhood info

- **PEAS: probabilistic approach [F. Ye et al, 2003]**
  - A node sleeps for a while (the period is adjustable) and decides to be active iff there are no active nodes closer than r'.
  - When a node is active, it remain active until it fails or runs out of battery.
  - The probability of full coverage is close to 1 if

$$r' < (1+\sqrt{5}) \times r$$

where r is the sensing (transmission) range

- **No guarantee of coverage or connectivity!**

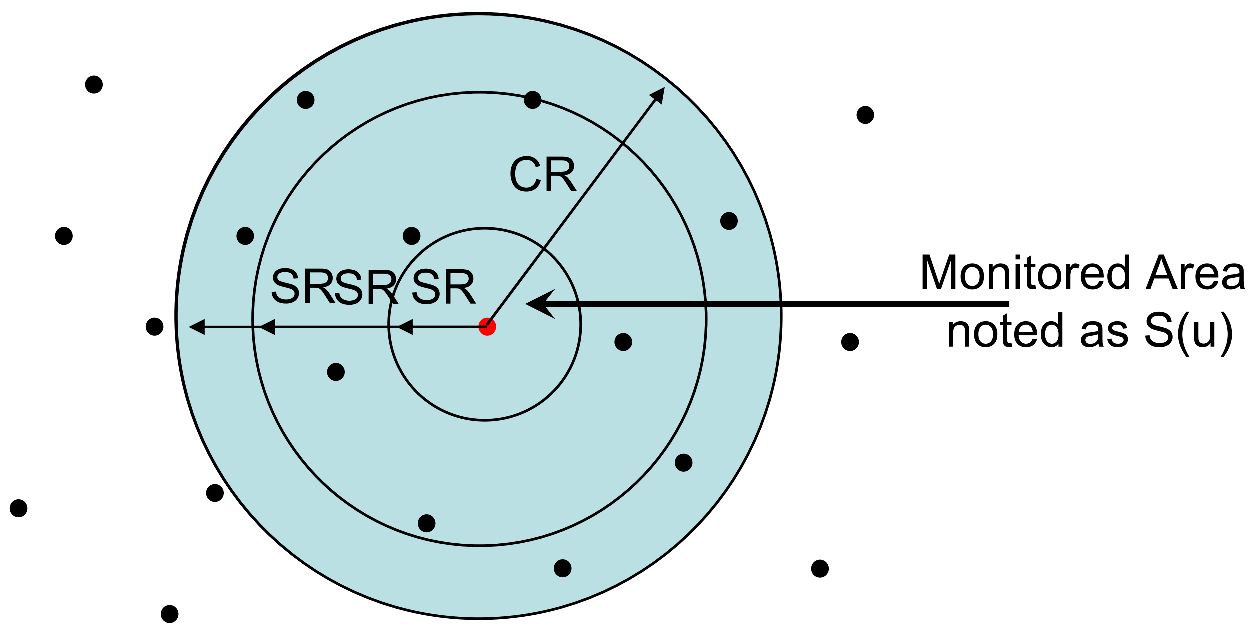
IRCICA — Institut de Recherche sur les Composants logiciels et matériels pour l'Information et la Communication Avancée

Ranges

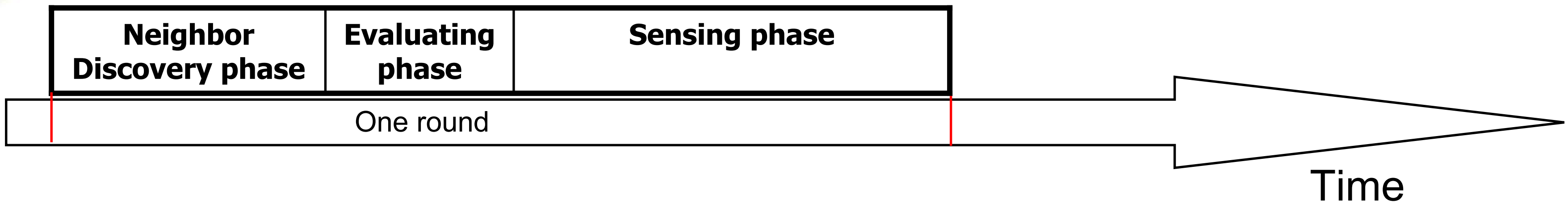
Communication Range, noted as CR

Sensing Range, noted as SR

SR2SR=CR2SR



Overview TGJD

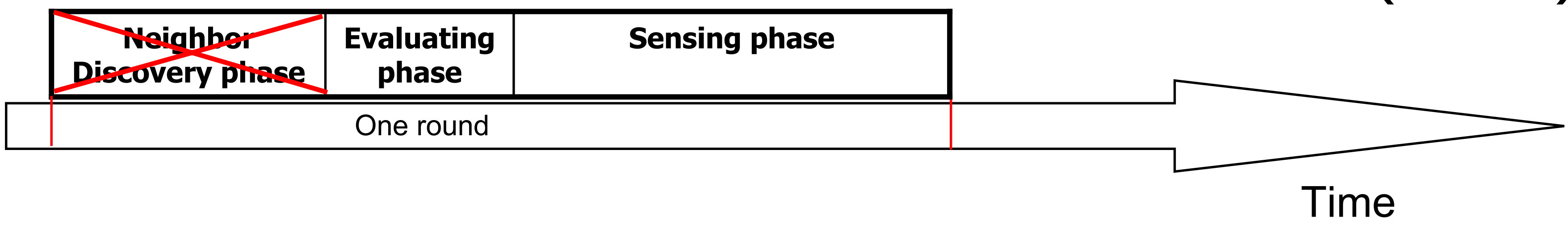


- **Neighbor discovery phase**

- **Evaluating phase**
  - ✓ Back-off scheme: avoid *blind point* (no neighboring nodes can simultaneously decide to withdraw)
  - ✓ A node decides to sleep: OFF message sent to one-hop neighbours
  - ✓ Receiving OFF messages during the timeout: corresponding neighbors are removed from the neighbor table
    - ⇒ At the end of the timeout, decision is made considering **only non-turned-off neighbors**

- **Connectivity is ensured as long as 2SR ≤ CR**

Carle, Gallais, Simplot-Ryl and Stojmenovic, 2005 (CGSS)



Evaluating phase

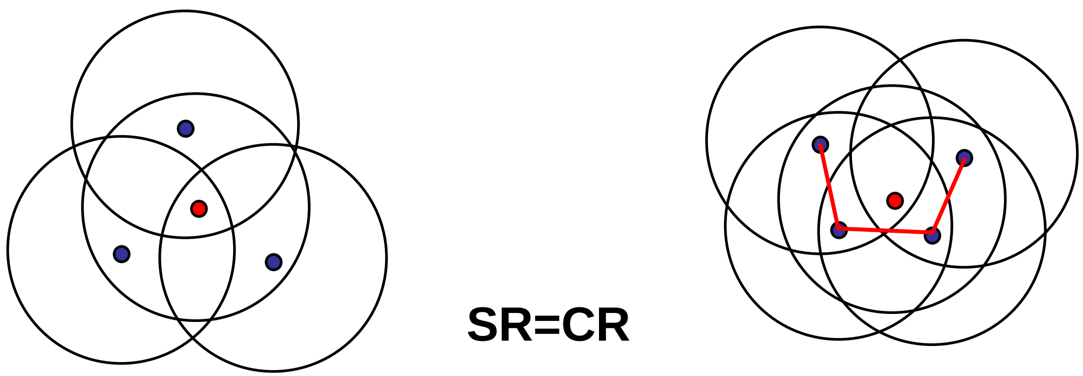
- Each node waits during a random timeout
- Once timeout ends, a node evaluates its **coverage** and decides
- **Advertisement** is then sent to one-hop neighbors
  - ✓ It only contains the position (x,y)

- Any message received during the timeout stands for a **new neighbor to be considered**

CGSS: Connectivity and Advertisements

- **Connectivity is ensured for any SR/CR ratio since full coverage must be provided by a connected set**

  (cf. Dai and Wu)



SR=CR

- **Which information must be broadcasted?**
  - ✓ Positive-only, **PO**: *u* broadcasts its position iff it remains active
  - ✓ Positive and **negative, PN**: *u* emits whatever the decision is.

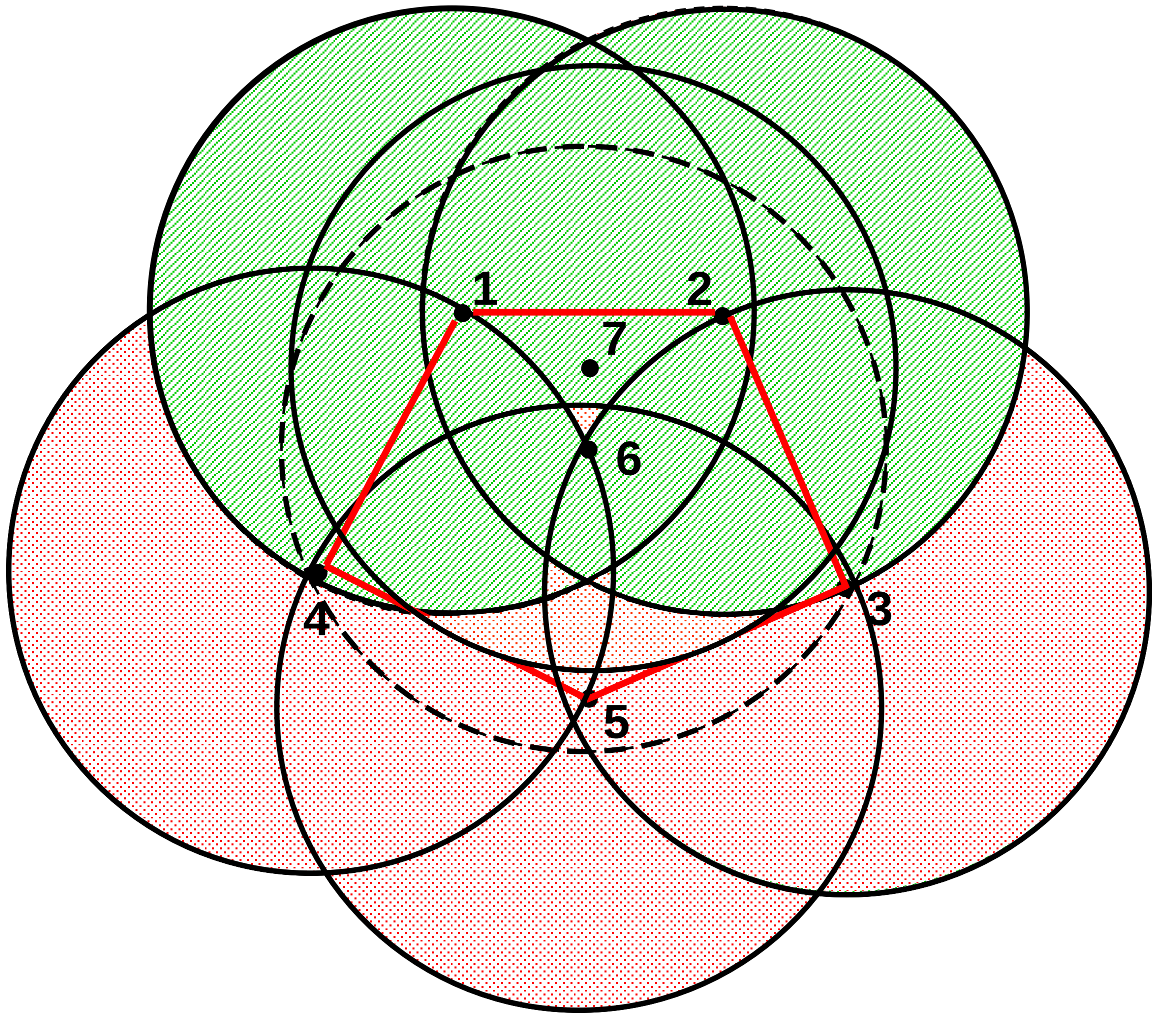
CGSS: Positive-only **(SR=CR)**

1 ?
empty
→ active

2 ?
1
→ active

3 ?
2
→ active

4 ?
1
→ active



5 ?
3 | 4
→ active

6 ?
1 | 2 | 3
4 | 5
→ sleep

7 ?
1 | 2
→ active

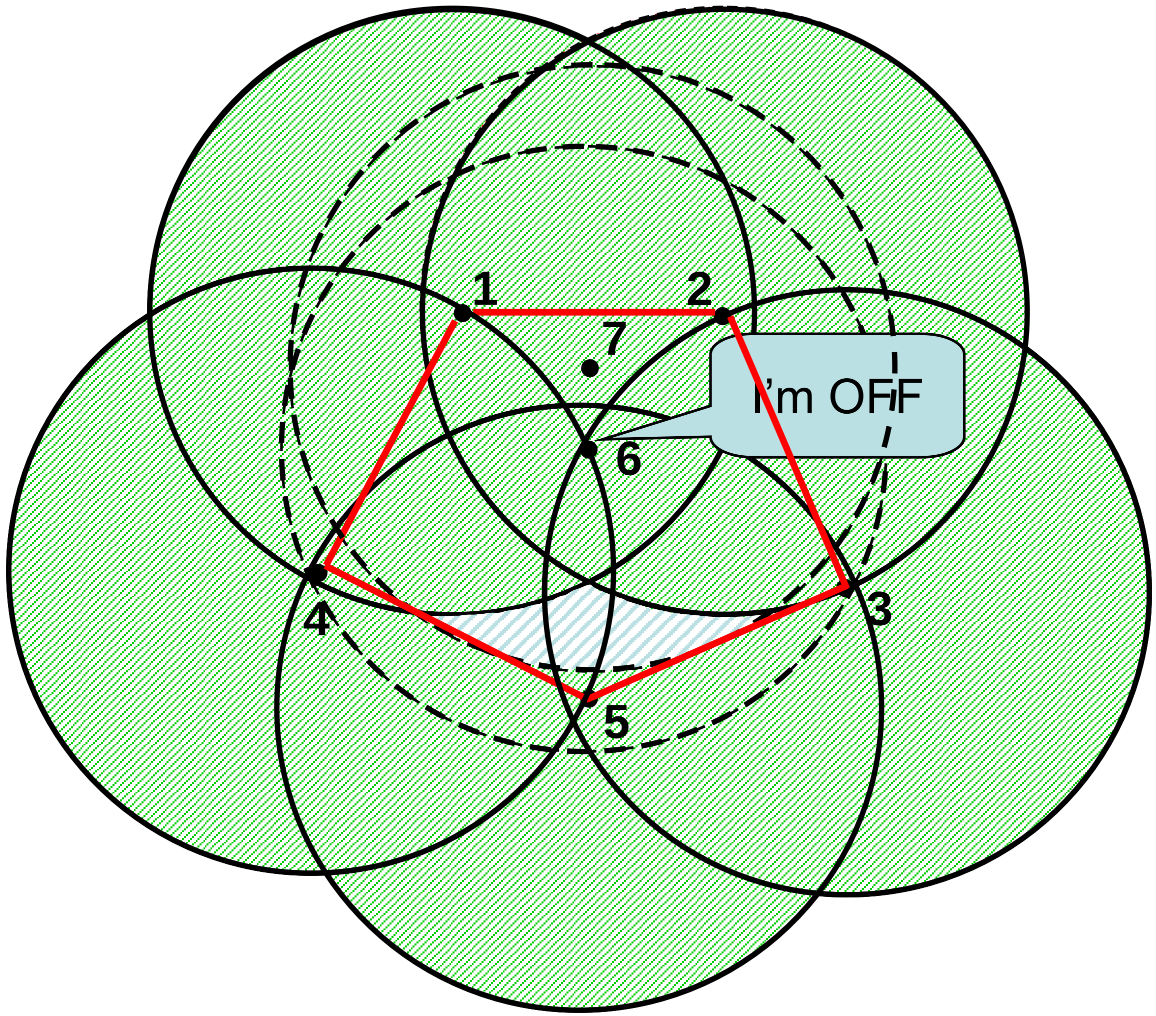
CGSS: Positive and Negative **(SR=CR)**

1
empty
→ **active**

2
1
→ **active**

3
2
→ **active**

4
1
→ **active**



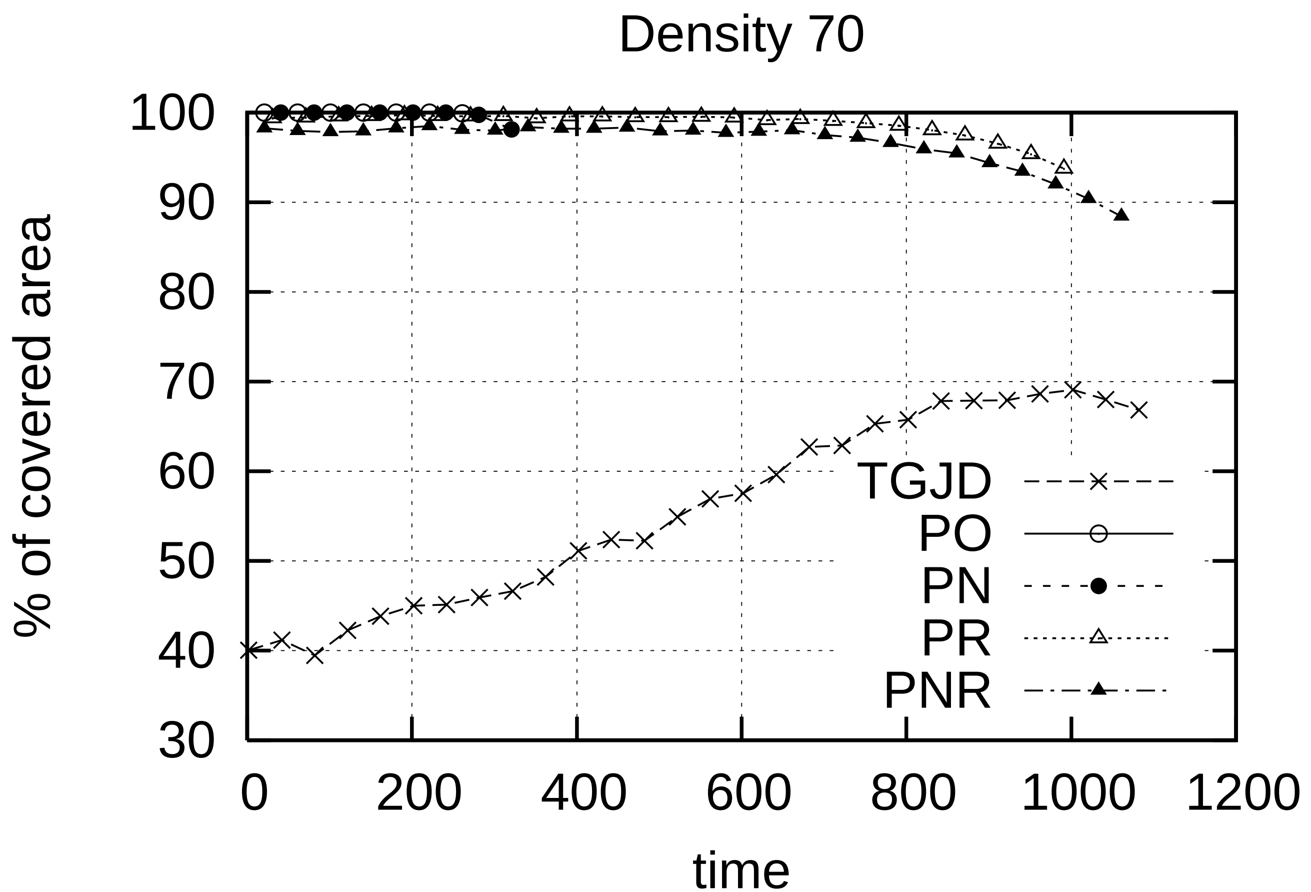
5
3 4
→ **active**

6 ?
1 2 3
4 5
→ **sleep**

7 ?
1 2 6
→ **sleep**

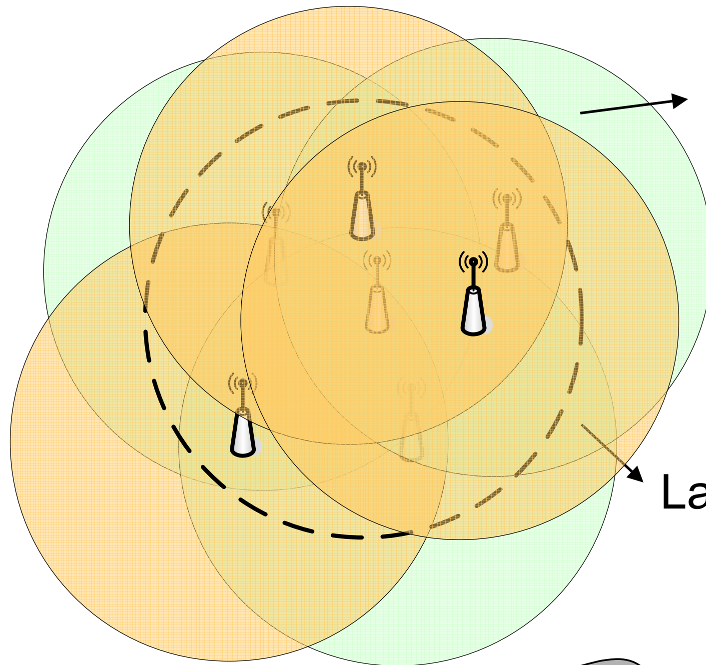
Experimental results

impact on protocol performances once messages get lost



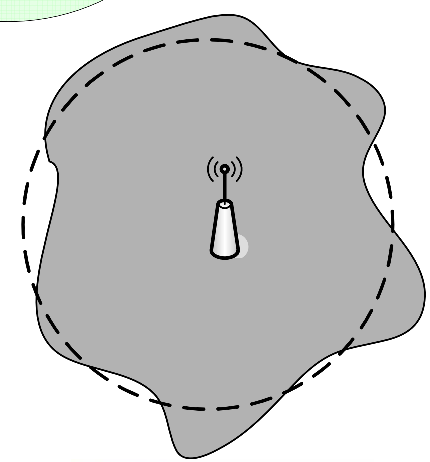
=> Loss of messages induce coverage loss for TGJD

Work in progress



- **Extend CGSS to k-coverage**
  - Insert a parameter (layer k) in activity messages
  - While i<k and covered at layer i, evaluate coverage at layer i+1
  - Each node looks at its k-coverage
  - Decides to be active or not

- **Non-ideal physical layer for sensing**
  - Unit disk may not be always valid ;-)
  - Would it really impact existing protocols?
  - Coverage evaluation scheme would be modified
  - Or shorter sensing radii could be announced

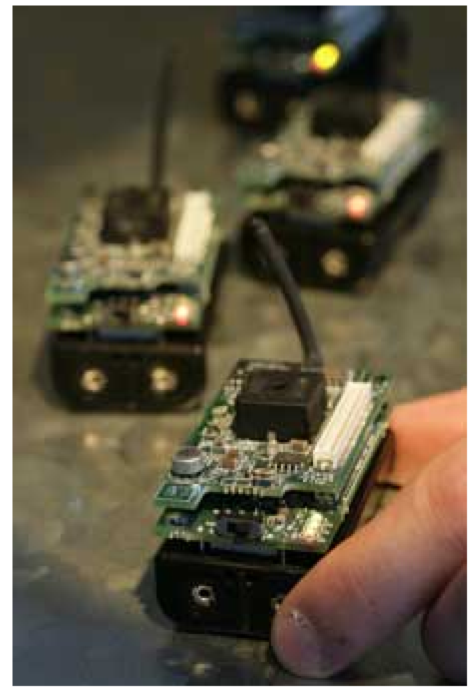


Outlines

I. Wireless Sensor Networks

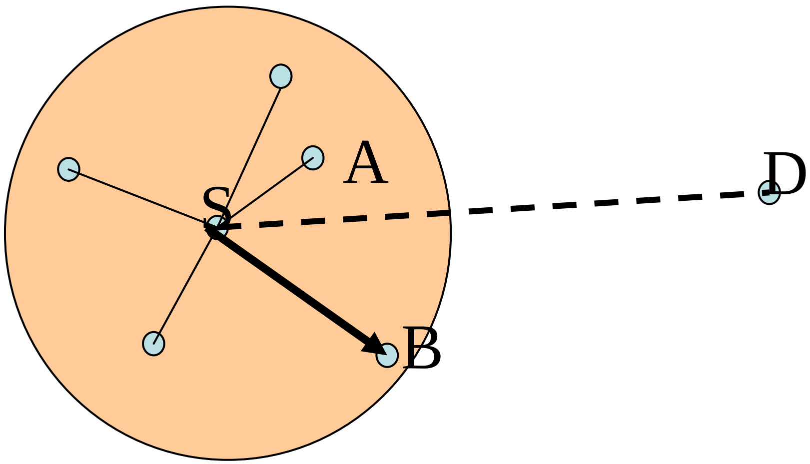
II. Activity scheduling and coverage problem

III. Routing with guaranteed delivery



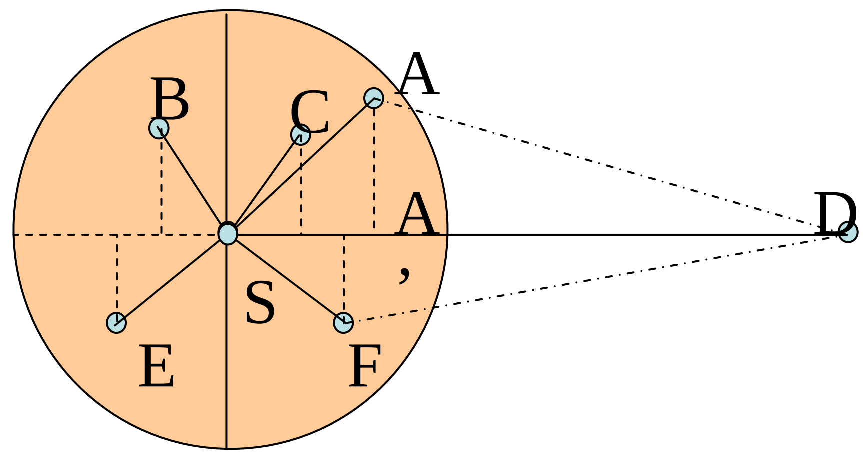
Basic principles

- **Nodes know their own geographical position and positions of their neighbors**
  - → localized
- **When a node has a packet for a given destination (knowing its position), it decides which neighbor is the next hop**
  - → memory-less
- **Example:**
  - S forwards to neighbor B closest to D
  - [Finn 1987]

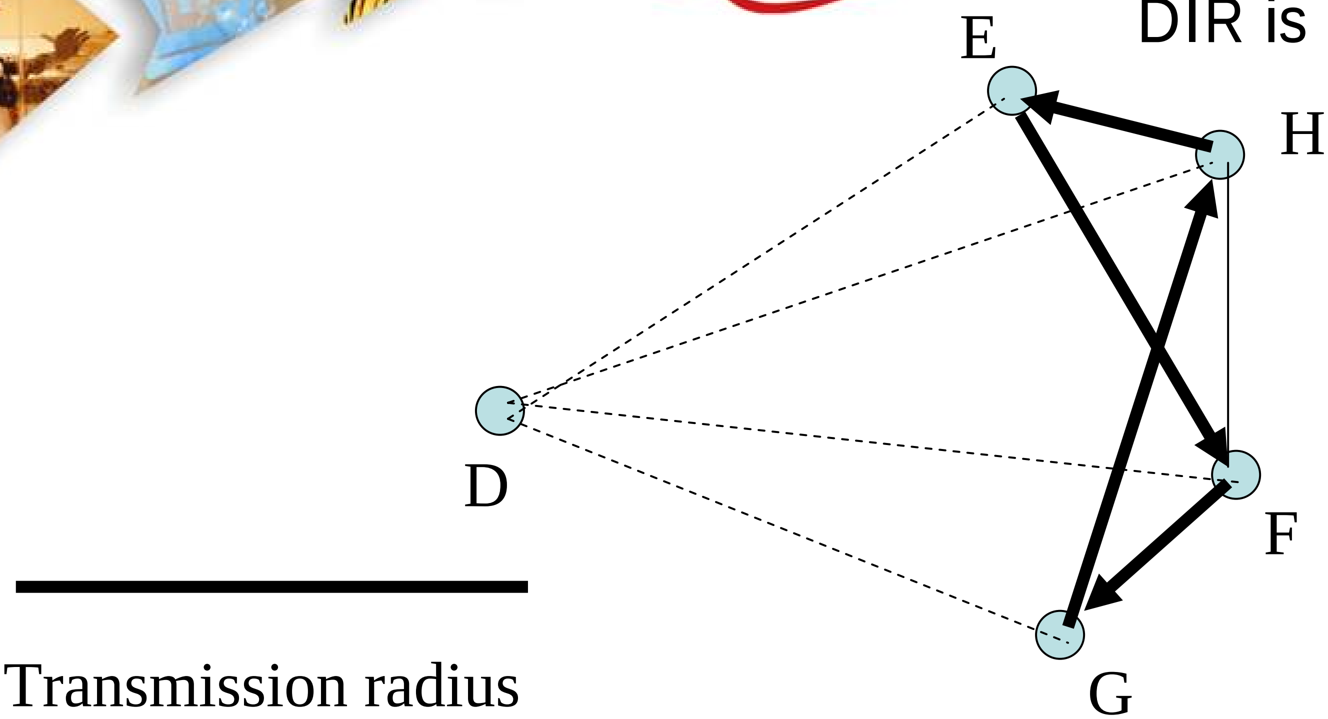


Some variations

- **Nearest forward progress [Hou]**
  - → small steps
- **More forward progress [Kleinrock]**
  - → big steps
- **Random progress [Nelson, Kleinrock]**
  - → ???
- **...**
- **Most close to SD with progress [Elhafsi, Simplot-Ryl]**
  - Variation of DIR [Basagni et al.] which is not loop free [Stojmenovic]

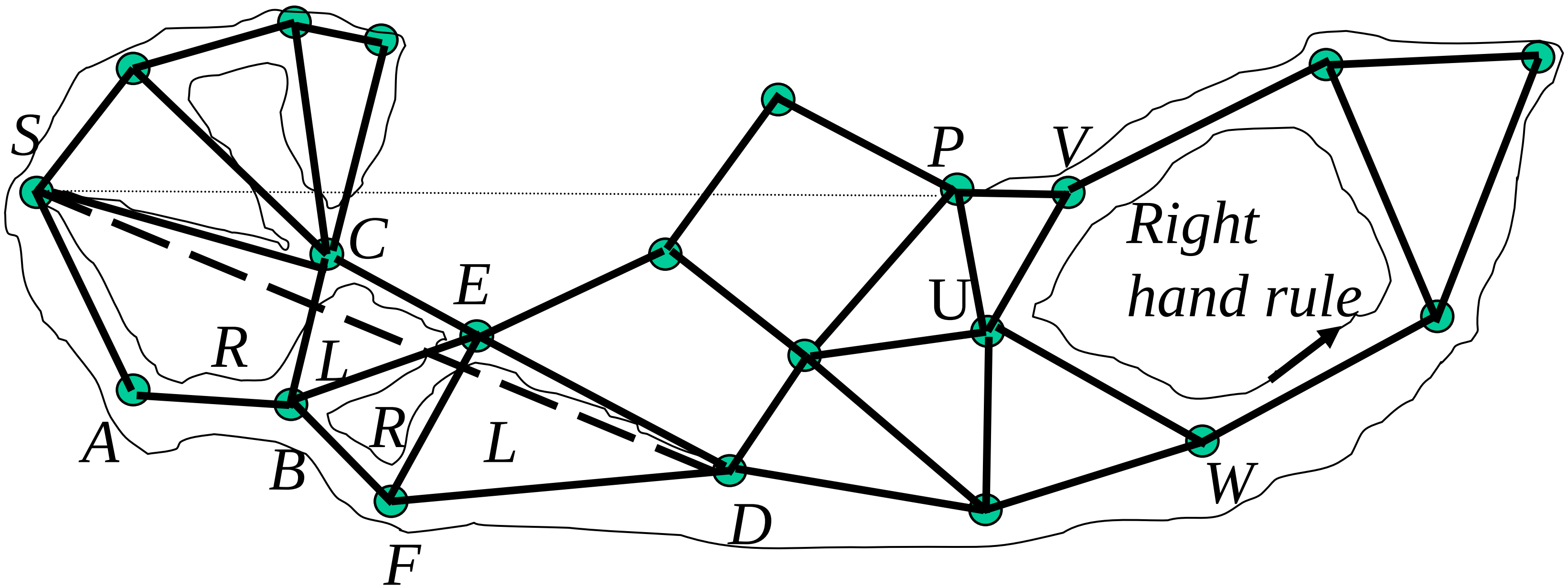


DIR is not loop-free !



- **Greedy and MFR are loop free**
- **If we include constraints on positive progress, the protocol is loop-free**
- **What if there is no neighbor with (strictly) positive progress?**
  - → Dead-end → Recovery process

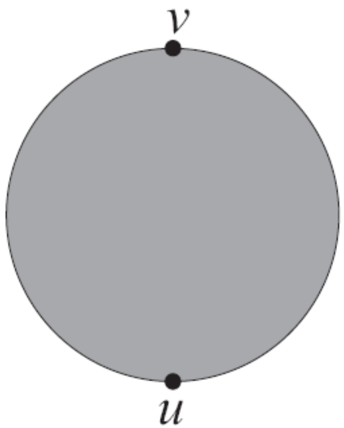
Face routing – guaranteed delivery

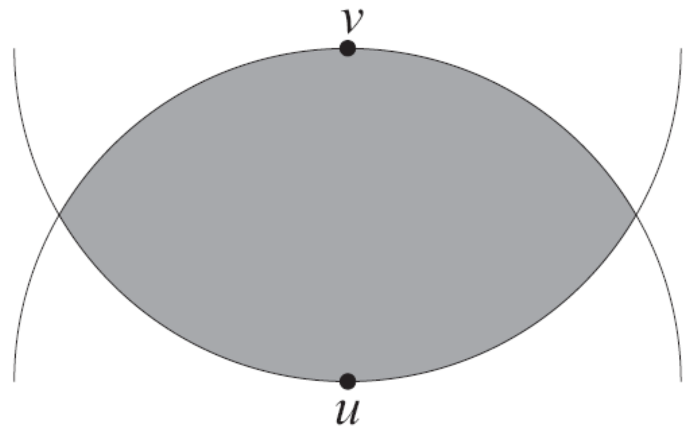


[Bose et al. 1999]

- **Construct planar subgraph**
- **Route in planar subgraph:**
  - SABCEBFED
  - SC…ABFD…W…VP

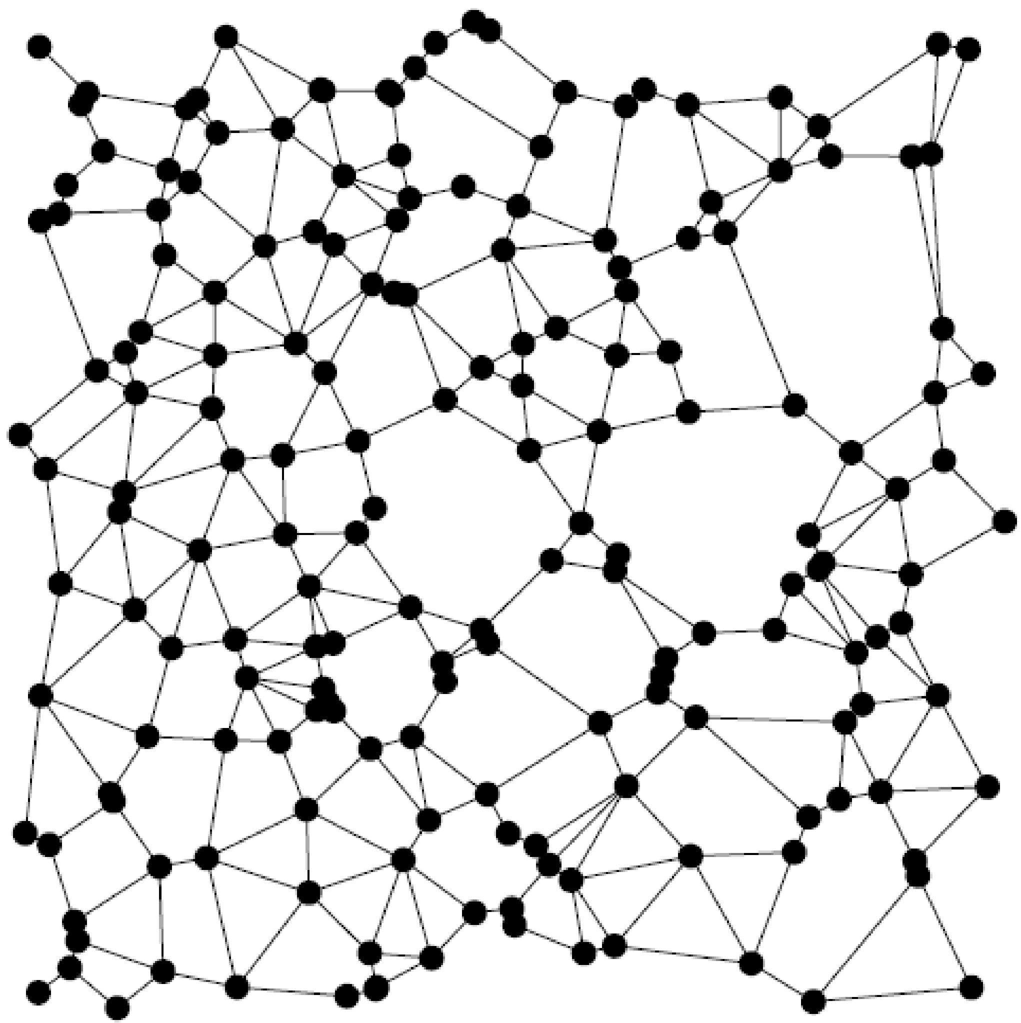
Planarization





- GG = Gabriel Graph
- RNG = Relative Neighborhood Graph
- GG contains RNG
- They are both planar (if the initial graph is the UDG)

Example of GG

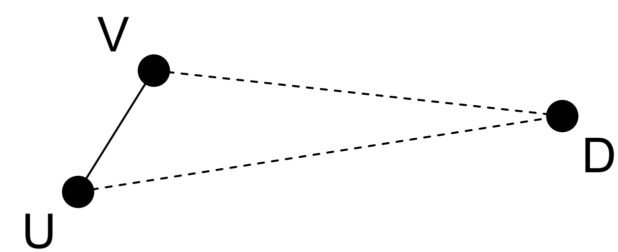


FACE is not energy efficient

- **For two reasons:**
  - The path itself can be long
  - Because of GG, edges are short → can be inefficient
- **Two counter-measures:**
  - Stop the recovery as soon as possible
    - ✓ When in dead-end: note the distance to the destination
    - ✓ Apply FACE until reaching a node which is closer than this distance
    - ✓ Frey and Stojmenovic have shown that only one face is used for each recovery
    - ✓ This scheme is called GFG Greedy-FACE-Greedy [Data et al.] where greedy part can be energy efficient (see later)
  - Apply Shortest path over FACE
    - ✓ → inefficient since edges maintained in GG are the shortest ones
    - ✓ Other solutions exist (see later)

Cost-over-progress scheme

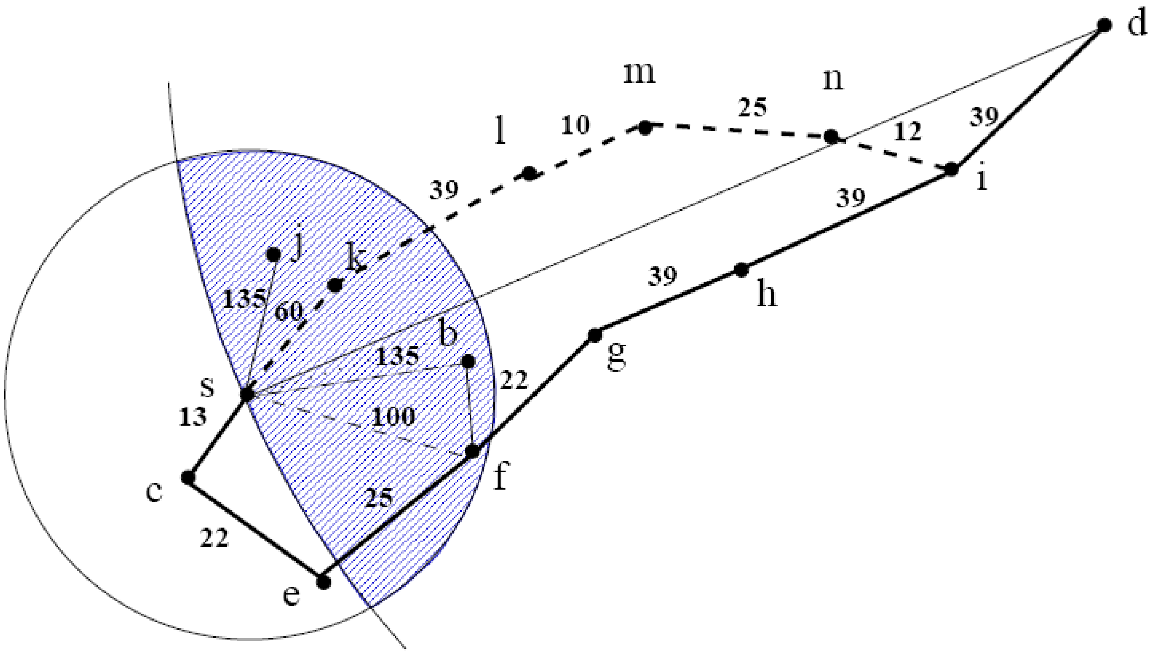
- **[Data, Stojmenovic]**
- **The principle is to consider each neighbor with positive progress and to evaluate the cost of the complete path**
  - Current node = U
  - Considered neighbor = V
  - Destination = D



  - Cost of one step = cost(U,V)
  - Number of steps = |UD| / progress
    - ✓ Progress = |UD|-|VD|
  - Evaluation of the cost of the path under homogeneous hypothesis: cost(U,V)*|UD|/progress
- **When minimizing this total cost |UD| is constant, so it is equivalent to minimize cost/progress**

Introducing shortest path in greedy part

- **After choosing the next neighbor, SP can be apply**
  - Ruiz et al. proposed to apply this after MFR
  - Problem: SP can include nodes which are farther to the destination
  - → the path has to be embedded in the message
  - → or at least the "truncated SP"



Removing memorization of the SP

- **In order to prevent the insertion of SP (or truncated SP) in message, we can use positive path**
  - A path is said to be positive is the distance to the destination is strictly decreasing

- **Summary:**
  1. the current node choose the target neighbor (e.g. MFR)
     If there is no target, go to 5 (recovery)
  2. It computes the shortest positive path to this node
  3. It sends the packet to the next node in this path
  4. Go to 1

  (Steps 1–4: Greedy)

  5. Note the distance from the current node to the destination
  6. Apply FACE until reaching a node which is closer, then go to 1

  (Steps 5–6: Recovery)

Selecting the target neighbor

- **Ruiz et al. proposed to apply MFR**
- **Cost-over-progress can also be applied**
- **More accurate: we can replace the cost by the cost of the shortest positive path…**
- **Summary:**

1. the current node choose the target neighbor

   which is the neighbor with strictly positive progress which minimizes cost-over-progress where the cost is the cost of the shortest positive path

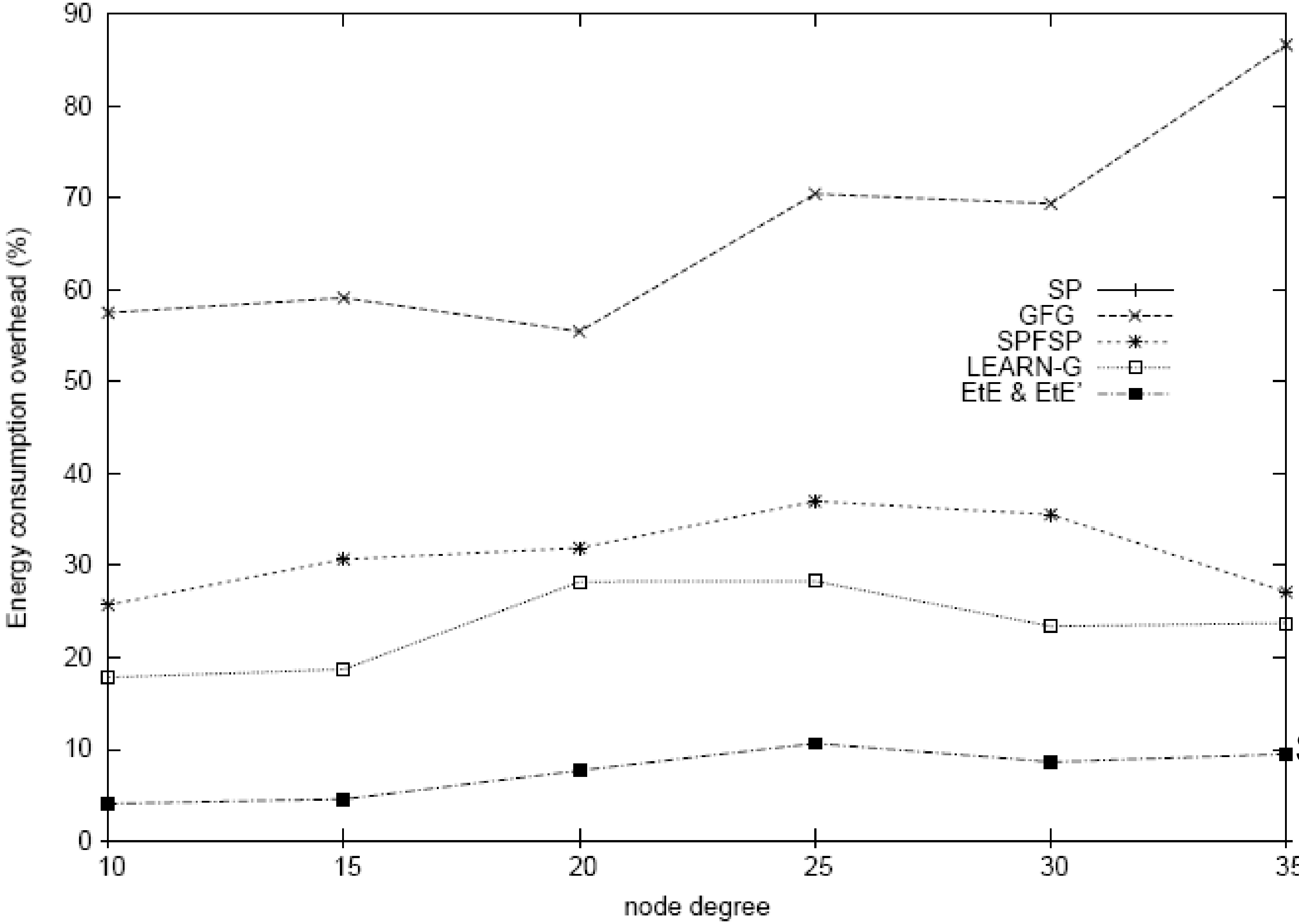
   If there is no target, go to 5 (recovery)
2. It computes the shortest positive path to this node
3. It sends the packet to the next node in this path
4. Go to 1

(Steps 1–4: Greedy)

5. Note the distance from the current node to the destination
6. Apply FACE until reaching a node which is closer, then go to 1

(Steps 5–6: Recovery)

Performances



Cost/progres

SP after MFR

SPP after C/P

Energy-efficient recovery

- **The main problem is length of the edges in GG**
- **→ idea:**
  - before applying GG:
  - apply a connected dominating set algorithm which minimizes the number of dominant nodes

- **A set of nodes is said to be dominant iff each node of the network is in this set of a neighbor of one node in this set**

Wu & Dai dominating sets

- (Simplified version [Carle and Simplot-Ryl 2004])

- **Each node has a priority (*e.g.* its ID)**
  - Variations:

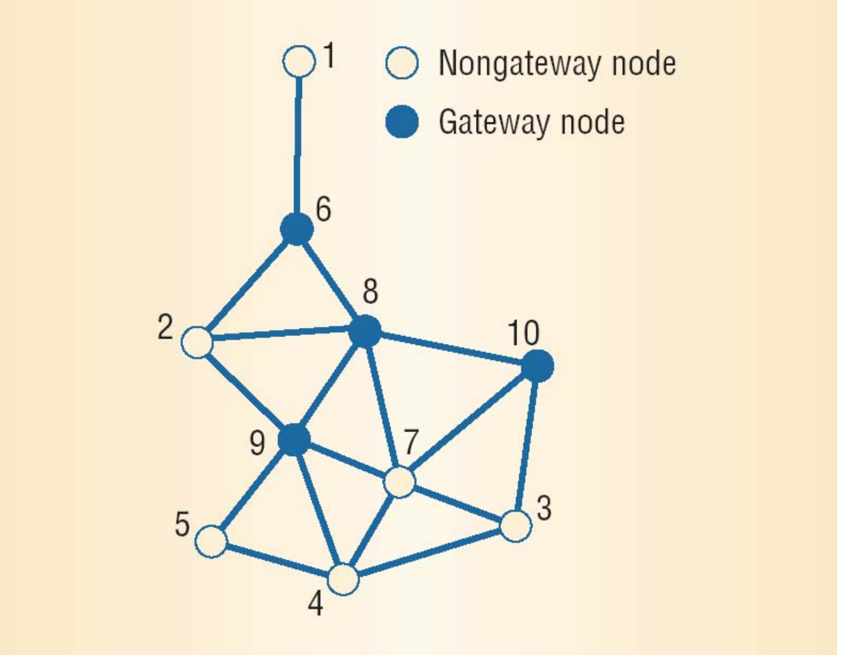
    <degree, ID>

    <battery, degree, ID>

    <random, battery, degree, ID>

    …

- **A node is not dominant iff**
  - The set of neighbors with higher priority is connected and covers the neighborhood



Energy-efficient recovery

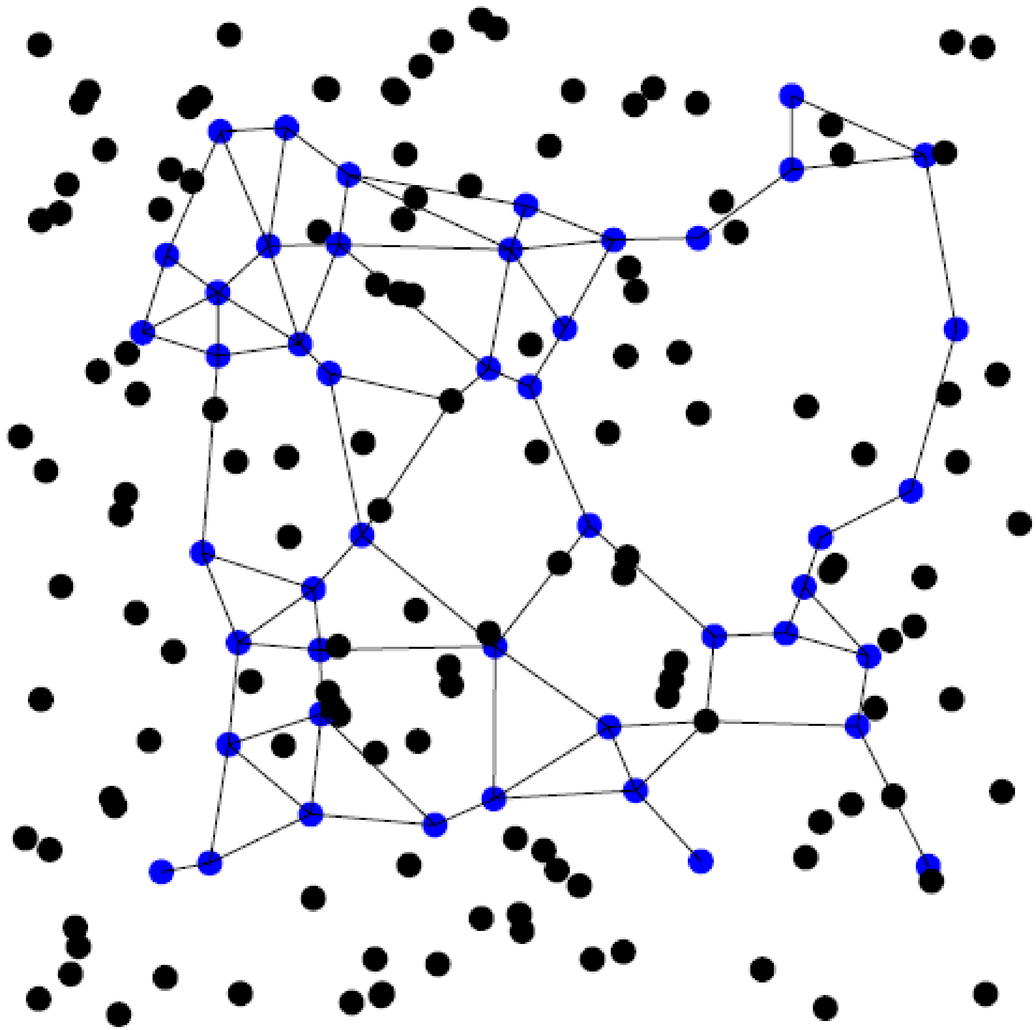
- **The main problem is length of the edges in GG**
- **→ idea:**
  - before applying GG:
  - apply a connected dominating set algorithm which minimises the number of dominant nodes

- **A set of nodes is said to be dominant iff each node of the network is in this set of a neighbor of one node in this set**

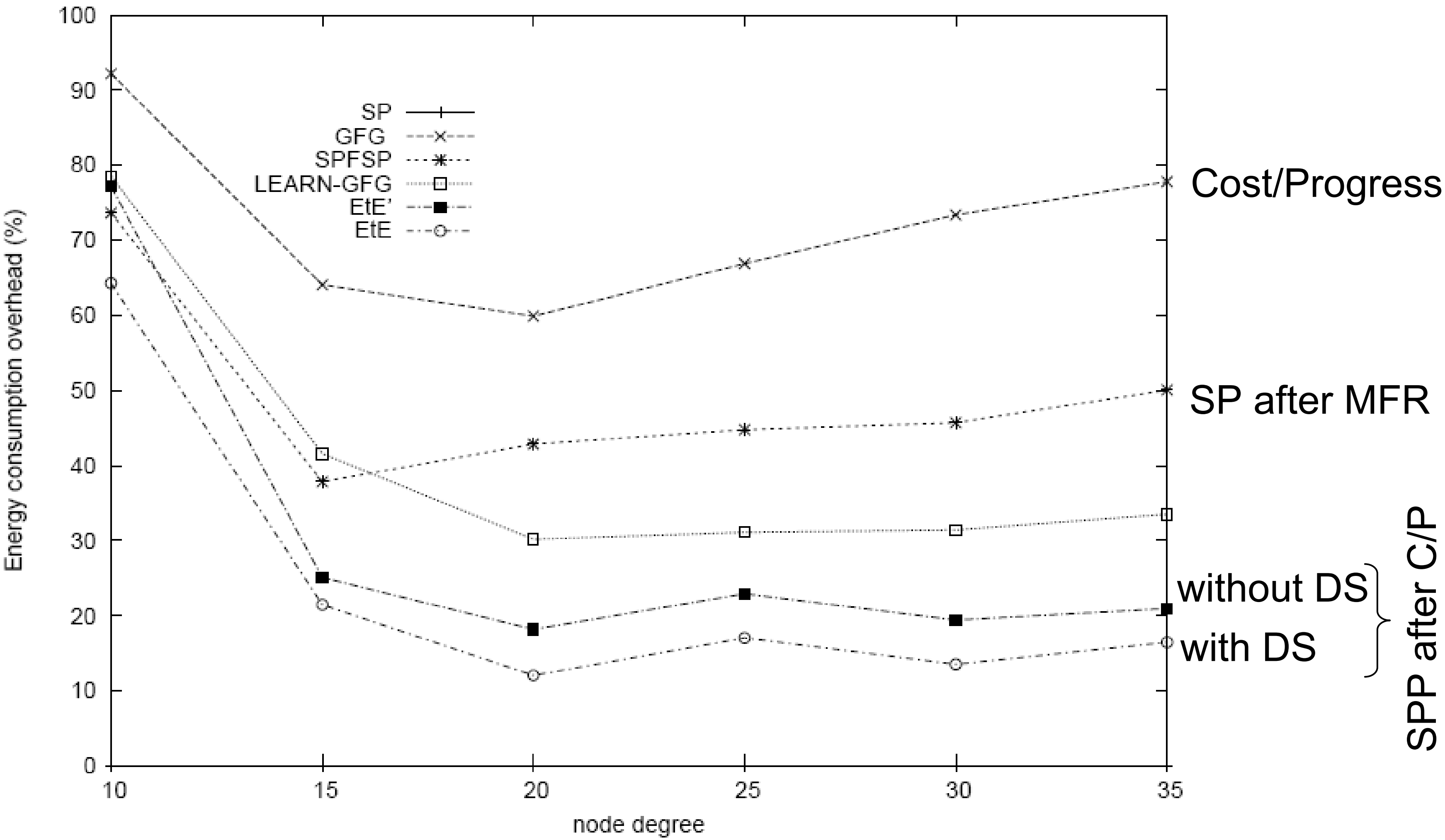
- **Then, apply FACE over the dominating set where routing along an edge uses shortest positive path**

IRCICA Institut de Recherche sur les Composants logiciels et matériels pour l'Information et la Communication Avancée

GG over CDS



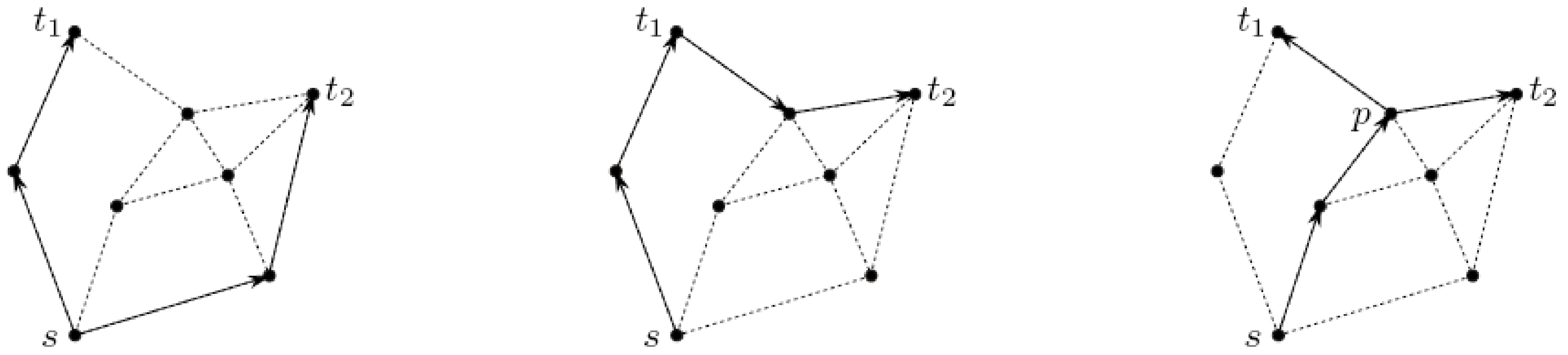
Performances



What else?

Multicasting

- **Same problem but the message as to be sent to several targets**



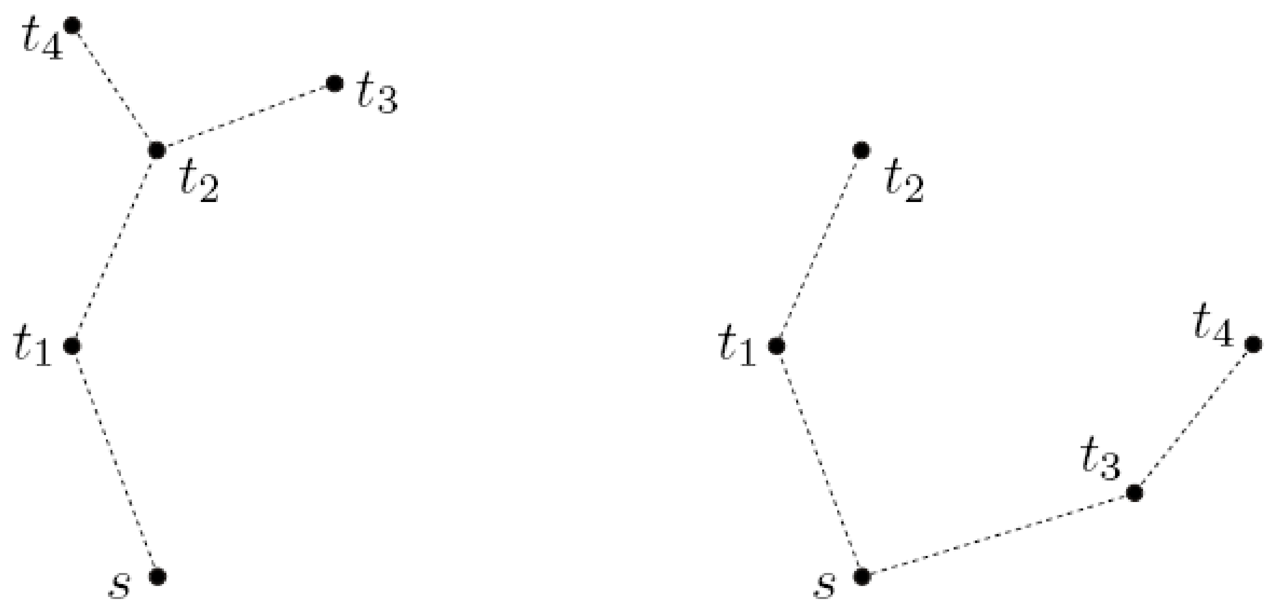
(a) The message is split at the source node.

(b) The message is never split.

(c) The message is split at the end of a common path.

MST-based splitting decision

- **Ingelrest et al. have shown that MST is a good approximation of the Steiner tree and propose to use it for splitting decision:**

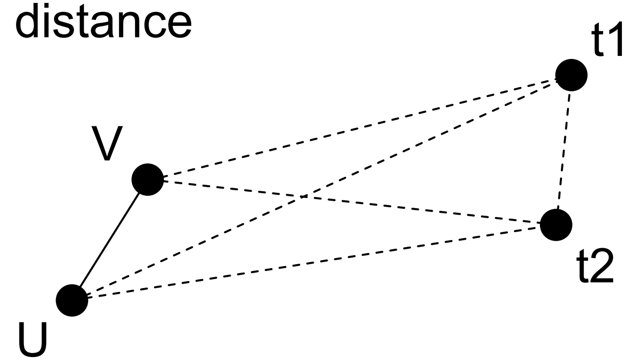


(a) All destinations are kept in the same set.

(b) Two subsets $\{t_1, t_2\}$ and $\{t_3, t_4\}$ are created.

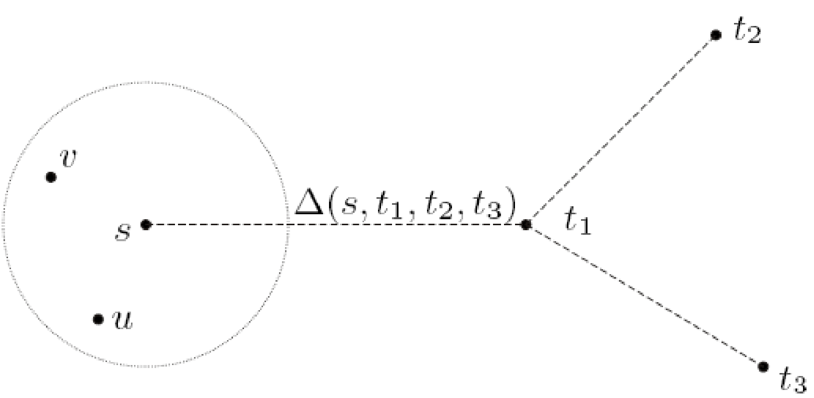
Greedy part

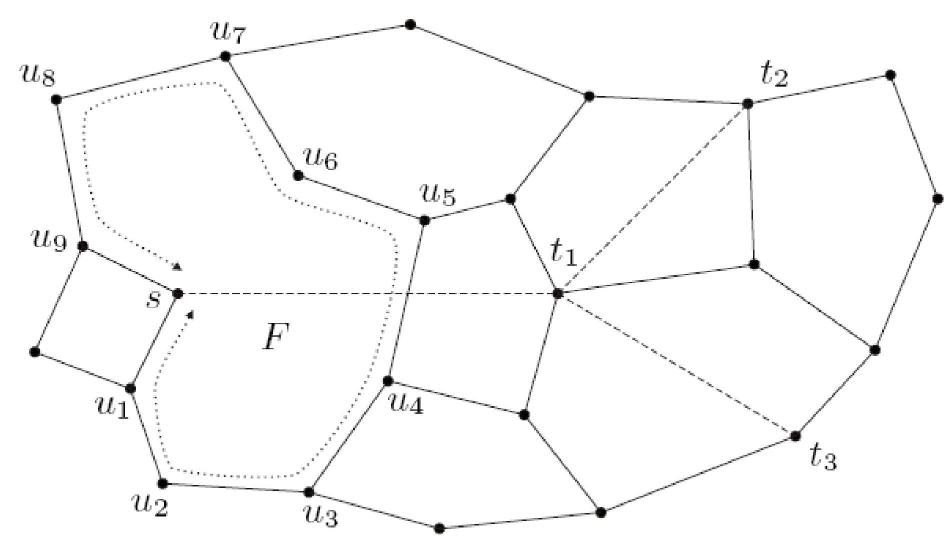
- **For each group, apply a variation of cost-over-progress**
  - The cost is as usual (techniques presented in the first part apply)
  - But what progress???
  - Again, we use MST in order to estimate to distance
  - If U is the current node,
    the progress when considering V is
    |MST(U,t1,…,tN)|-|MST(V,t1,…,tN)|



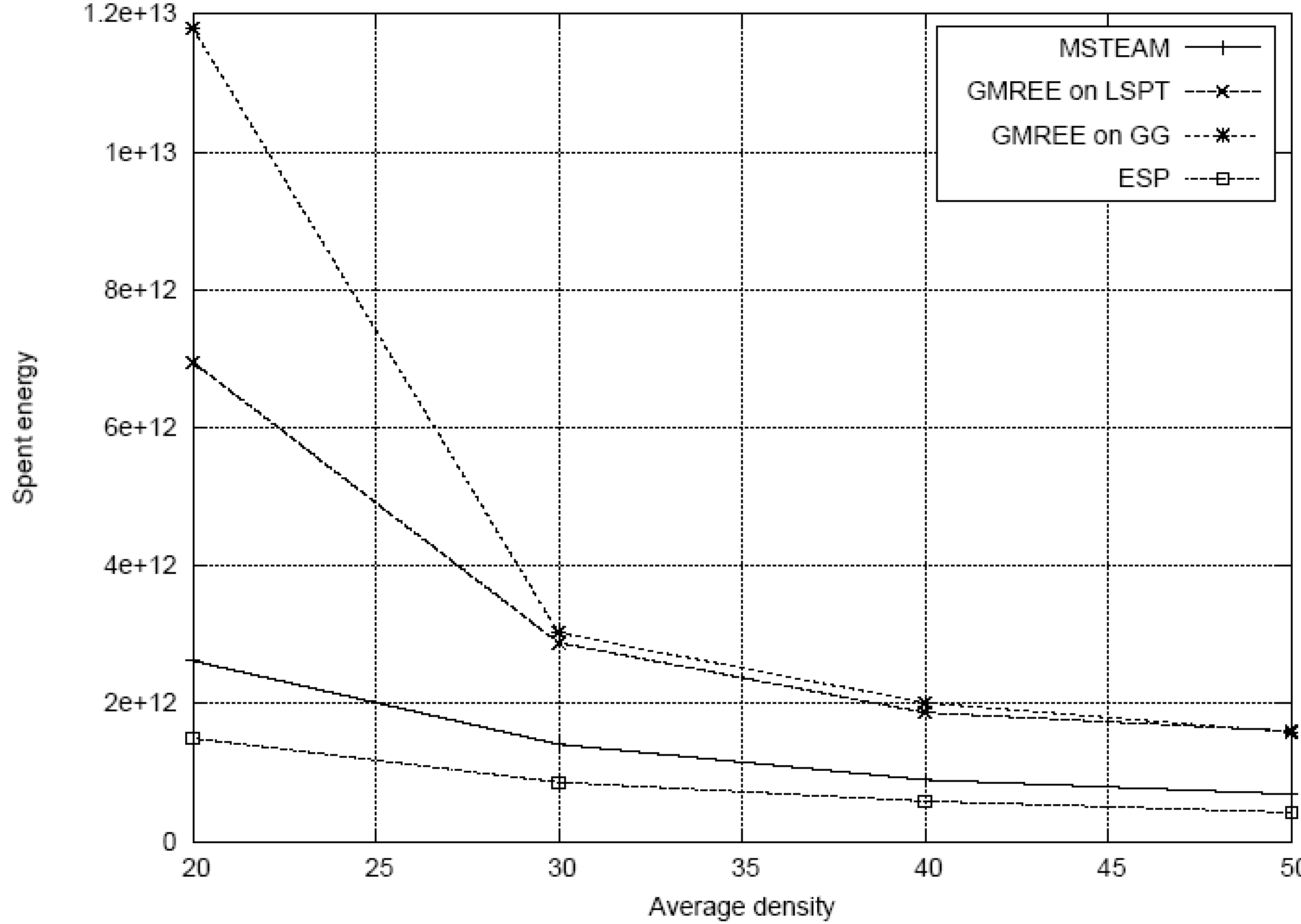
Recovery part

- **We can apply a variation of FACE:**
  - When a dead-end is detected we note the distance to the closest target/destination and we apply FACE to this destination until we reach a node for whom there exists a target closer than this distance.
  - Other possibility: replace the line to the destination by the MST edge





Performances



Ruiz et al. 2007

MSTEAM

Centralized Algorithm

Next?

Conclusion & perspectives

- **Best-known position-based routing (unicast and multicast) have been presented**

- **With geographical information**
  - Basic algorithm: YES (MFR)
  - Energy efficient (EE): YES (cost/progress, ETE…)
  - Guaranteed delivery (GD): YES (FACE, GFG)
  - EE+GD: **YES** **(ETE)**

- **Without geographical information**
  - Basic algorithm: YES (VCap)
  - Energy efficient (EE): **YES** **(VCost…)**
  - Guaranteed delivery (GD): **YES** **(LTP)**
  - EE+GD: **YES** **(HECTOR)**

Conclusion & perspectives

- **Are these protocols efficient in real world?**
- **Big problem: the guaranteed delivery part requires planarization of the connection graph…**
- **Planarization of an arbitrary graph cannot be done in a localized way**
- **Challenge: which properties are needed for the physical layer in order to find a localized algorithm for the planarization**

IRCICA

Institut de Recherche sur les Composants logiciels et matériels pour l'Information et la Communication Avancée

Energy savings in wireless sensor networks

David SIMPLOT-RYL

Centre de recherche INRIA Lille – Nord Europe
IRCICA/LIFL, CNRS 8022, Université de Lille 1

http://www.lifl.fr/~simplot
simplot@lifl.fr