Cross-layer interaction in wireless sensor networks

Paul Havinga
University of Twente

Current work

- The EYES project
  - 3 years EU FP5 project, runs till March 2005
  - Focus on research
    - Explore various options
    - Partly multi-disciplinary
    - Quite wide
    - Not a real focus on applications
      - For the final demonstrator we selected cow networking and applications
  - Exploration and technology driven
- The CONSENSUS project
  - 500 kEuro national project
  - On collaborative distributed algorithms for WSNs

Current and future work (1)

- The Smart Surroundings project
  - A 14 Meuro project on ambient intelligence
    - 15 partners, industry and research institutes
    - April 2004 – October 2008
    - Ambition to move from prototypes towards architecture
    - Explore various settings, such as
      - Fitness and healthcare
      - Security
      - Home and office automation
  - Platforms and tools for distributed algorithms for ambient intelligence applications
  - Embedded networking, localization, resource management, context and interaction, …
  - A mix of bottom up and top down approach

Current and future work (2)

- The Featherlight project
  - 500 kEuro national project
  - To provide a reconfigurable SW platform for distributed algorithms on small resource poor embedded devices
  - Industrial collaboration with Philips research, Thales, Nedap, and TNO

Application areas

- Application areas
  - Environmental monitoring
  - Smart farming
  - Space exploration
  - Business objects (e.g. logistics, tracking dangerous goods in a chemical plant)
  - Security
  - Home and office automation

Common challenges and drivers

- Energy efficiency
- Heterogeneity and diversity
- Dynamics
- Competition
- Abstraction

Cross layer interaction

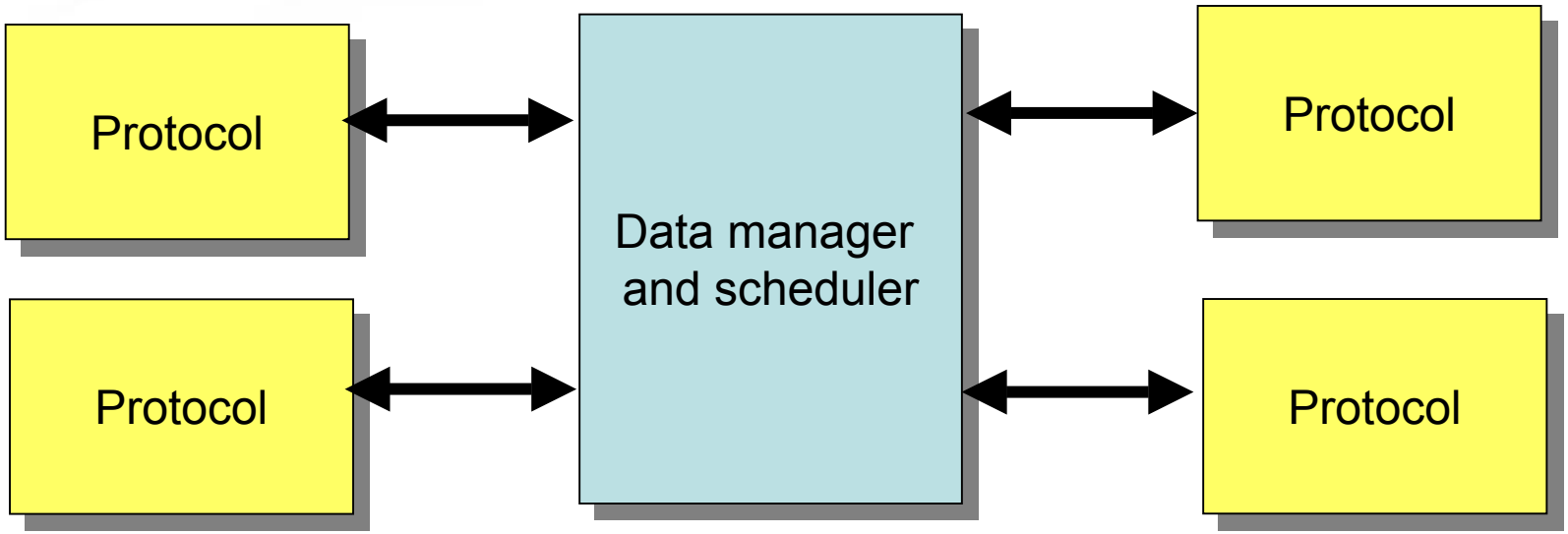
- Traditional a layered structure
  - Pass a limited set (scope) of information over well-defined interfaces between separate layers of the protocol
  - **Good** for abstraction and development
  - **Bad** for efficiency in case
    - High level info is useful in lower layers
    - Low level info is useful in lower layers
      - examples are power control, error control, aggregation, fusion, localization, service discovery, semantic addressing, etc.
  - Layers tend to merge

Cross layered integration

- Application driven
  - Protocols should fit to these demands
  - Which implies that protocols, traffic characteristics, and requirements, can be different per node!
- Useful for dynamics caused by
  - Mobility
  - Failures
  - Dynamic power modes
- Scalability and resources
  - Adapt to node density, available resources, and size

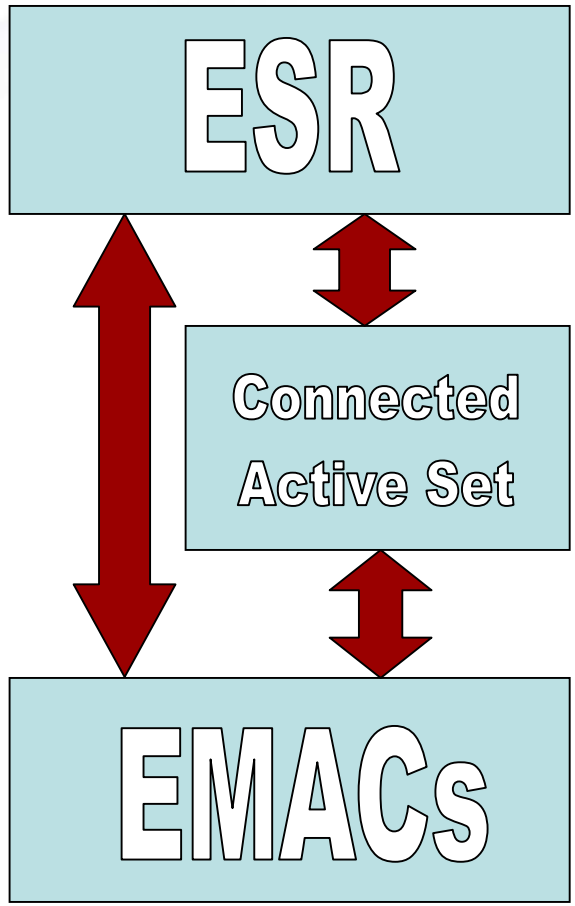
Information centric

- It is the available information that **dynamically** determines what protocols to use
- Based on a match between available resources, capabilities and information



Example of a cross-layered approach

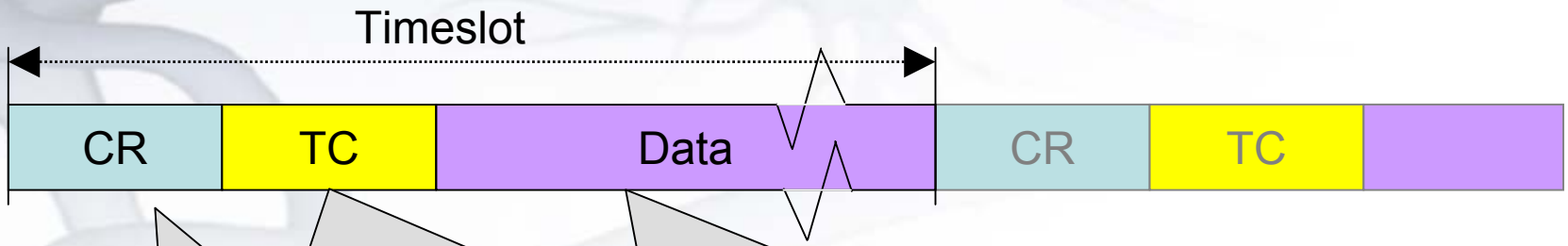
- EMACs
  - self-organizing, TDMA-based, MAC-scheme
- Connected Active Set
  - Identify nodes that are needed for connectivity (“ACTIVE” nodes)
  - Other nodes can follow sleeping pattern (“PASSIVE” nodes)
- ESR (Eyes Source Routing)
  - On-demand, dynamic routing protocol
  - Limited flooding to reduce routing overhead during dynamic changes in topology



EMACs: *Introduction*

- Self-organizing, TDMA-based MAC protocol
  - Nodes can autonomously chose time slot
  - Collision-free communication
- Supports efficient transmission of short multicast messages
  - Used in *clustering*, *routing, localization etc.*
- Scalable
  - adaptive for network topology caused by mobility or failures

Frames and Time Slots



Tra

- "Heart be
- Always tr

time slot

- Contains lis
- Used for efficie

(short) omnicast messages

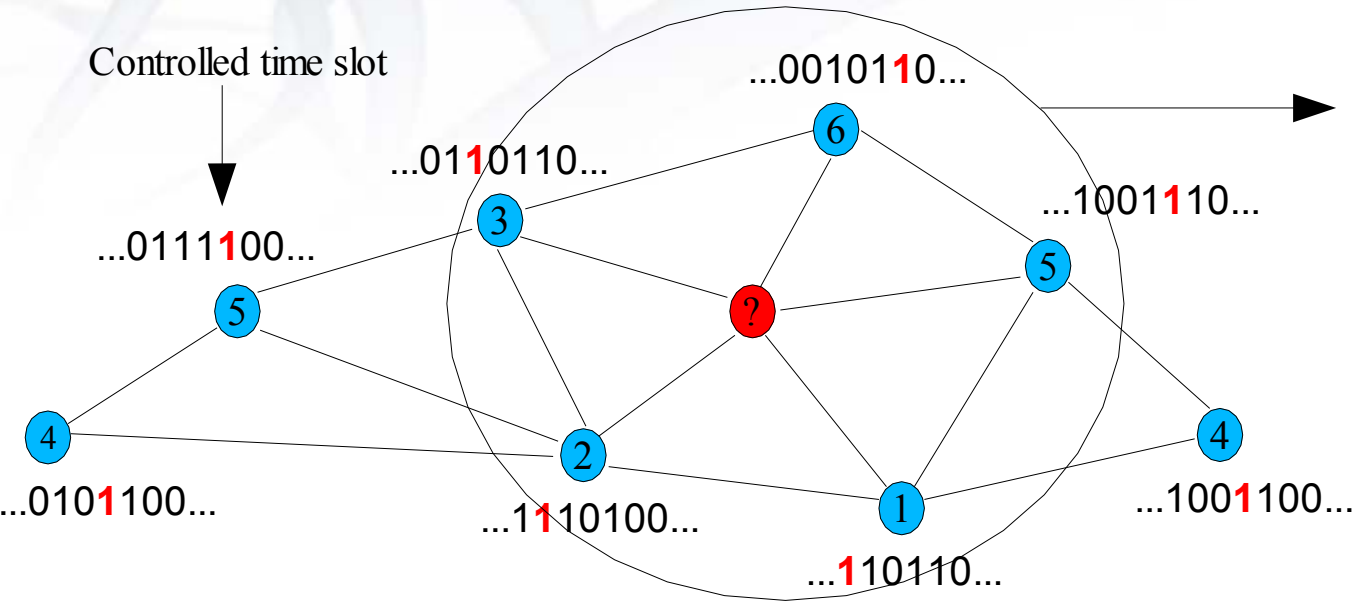
- Addresses other nodes

Data

- Contains data of higher layers
- Nodes sleep this interval if they are not needed
- Glued to TC if possible

Local Decisions on Time Slot

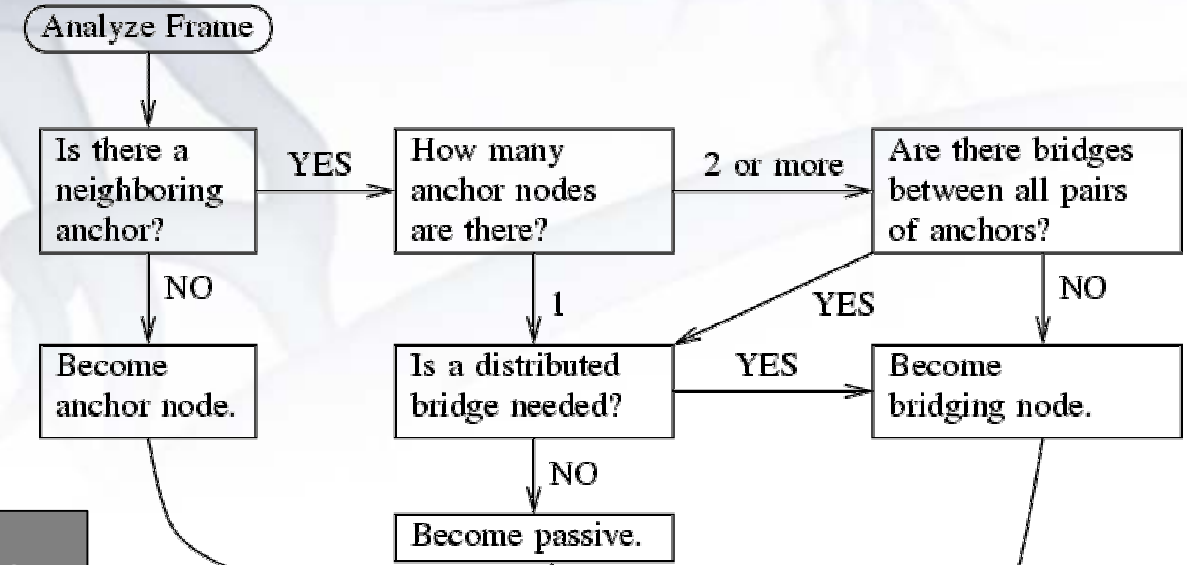
- Nodes choose time slots locally
  - TC-Section contains list of neighboring slots



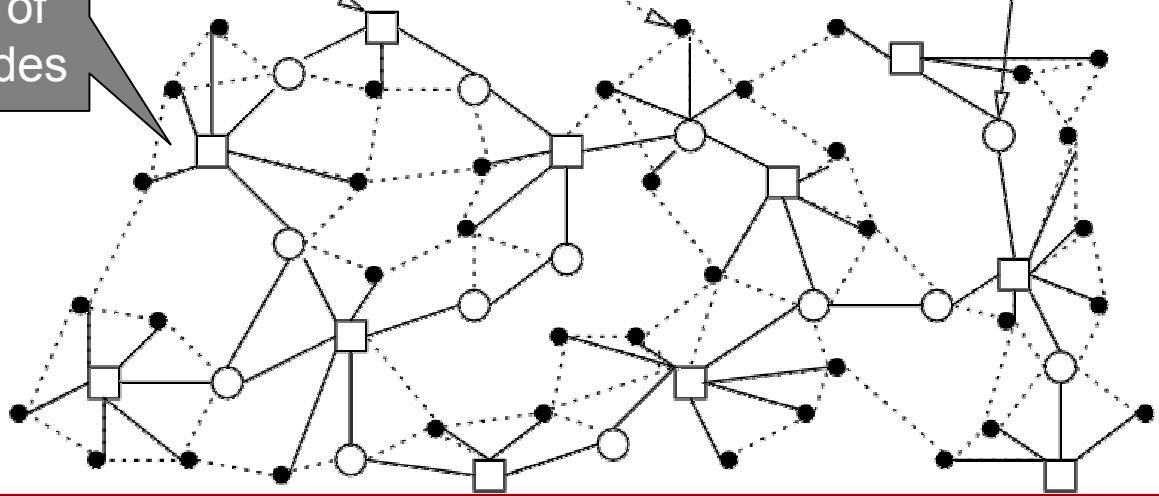
2 = Active node, that claimed time slot 2

? = New active node in the network

Connected Active Set: *Algorithm*



Mesh-like structure of active nodes

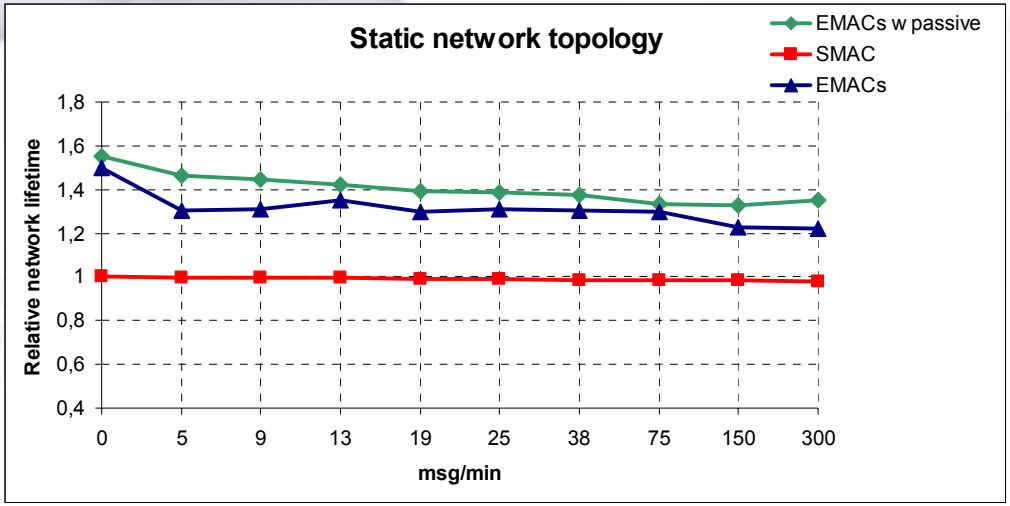


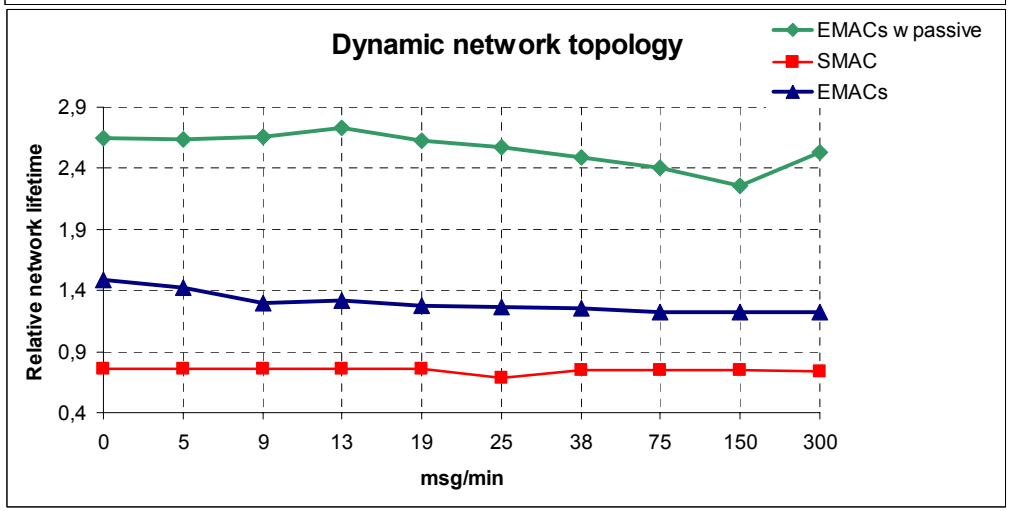
Anchor

Bridge

Passive

EMACs: *Sleep pattern saves energy*

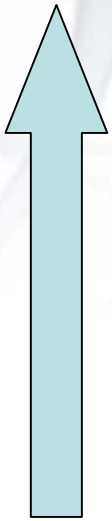




- Static scenario
  - Roles are not rotated
  - No dramatic increase of lifetime
- Mobile scenario
  - Energy consumption is more evenly distributed in the network
  - Roles are changed
- Significant increase of network lifetime!!

Conclusions

Routing



MAC
Active Set

Information provided by EMACs is used implicitly to create Active Set

Sleeping modes supported

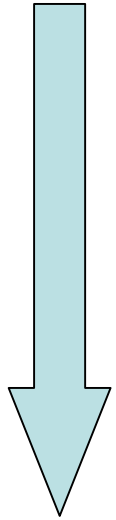
Maintenance routines are triggered by MAC-protocol

Routing relies on structure created by Active Set

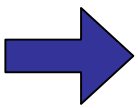
Routing needs not to account for any structural properties (e.g. connectivity)

Active Set helps to reduce interference, especially in dense networks

Routing



MAC
Active Set



Cross-layered approach outperforms similar, but layered approach (S-MAC + DSR)