Crankshaft: An Energy-Efficient MAC-Protocol for Dense Wireless Sensor Networks

Final Presentation

by Jonathan Hontschik

August 4th, 2008

References

[1] G. P. Halkes and K. G. Langendoen:
Crankshaft: An Energy-Efficient MAC-Protocol for Dense Wireless Sensor Networks, EWSN 2007

Other sources are labeled.

Agenda

1. Introduction
2. Principles of Crankshaft
3. Simulations
4. Discussion

Introduction – MAC-Protocols

- Protocols for Medium Access Control

- Different approaches
  - Time Division
  - Carrier Sensing
  - Collision Detection

Introduction – Wireless Sensor Networks

- Applications

- Special characteristics
  - Many nodes / one or few sinks
  - Nodes have limited energy source
    - Need for energy efficiency

- Density

Introduction – Dense Networks

Density many neighbours

Problems intensify in dense networks:

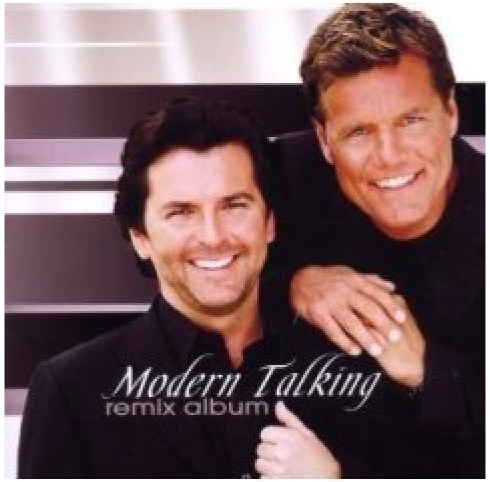
- Overhearing
- Communication grouping
- Over-provisioning
- Neighbour state

Introduction – Problems

Overhearing

– Listening to messages you do not need to listen to
  waste of energy

– The same in everyday's life:



Communication grouping

– Some protocols split the time into an active and a non-active part (sleep).

– This saves energy but increases the probability of contention and collisions.
  more traffic
  more energy spent

Introduction – Problems

Over-provisioning

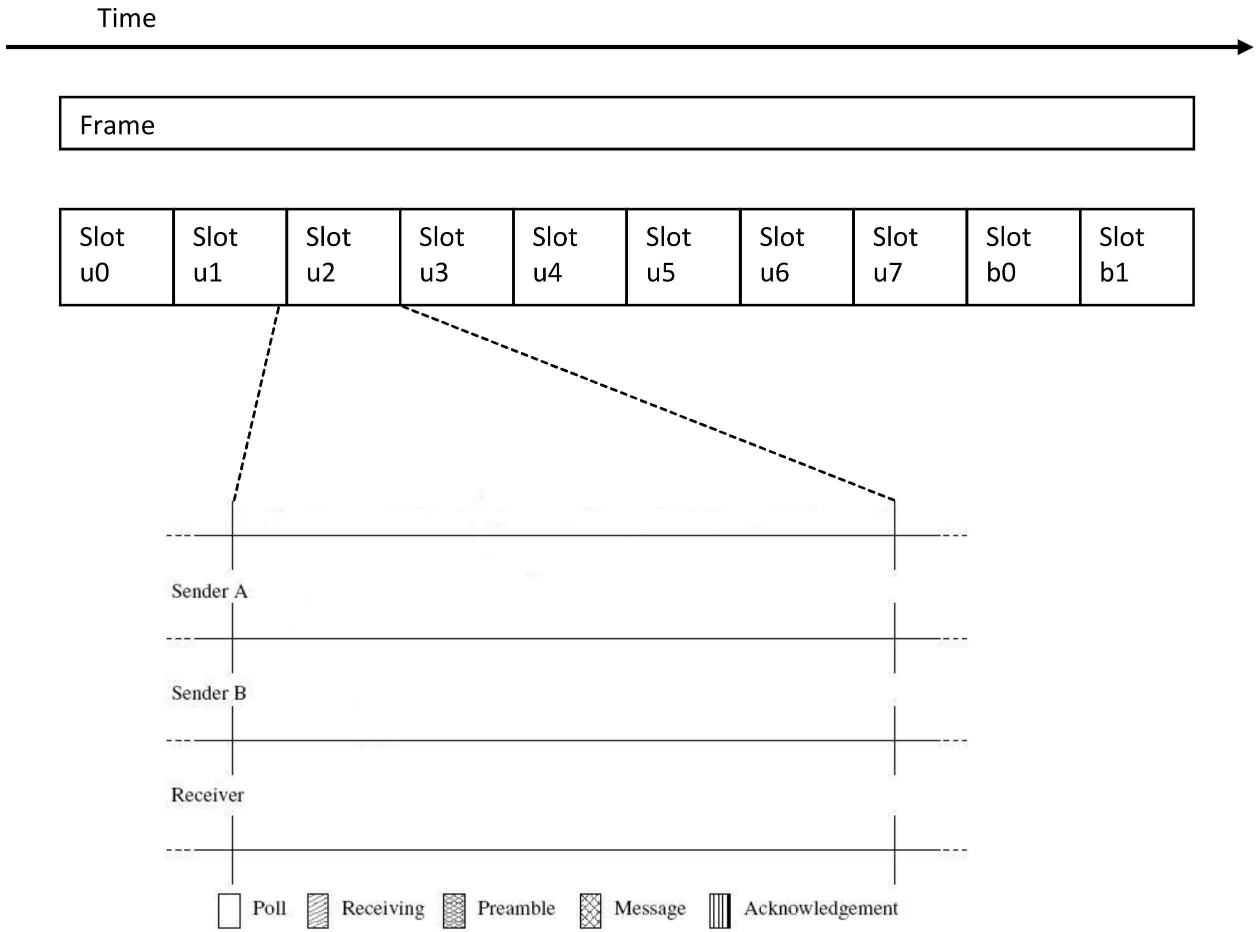
- When providing slots to send, many of these slots are wasted as the node may have nothing to send.
- Other nodes at least have to check if the node is willing to send energy spent for checking that .

Neighbour state

- Some protocols save the neighbour states.
- In a dense environment this costs a lot of energy (and memory) for maintaining this information.

Structure of Crankshaft

- Time
- Frame
- Slot
  - Unicast
  - Broadcast
- Further subdivision



Crankshaft „at work“

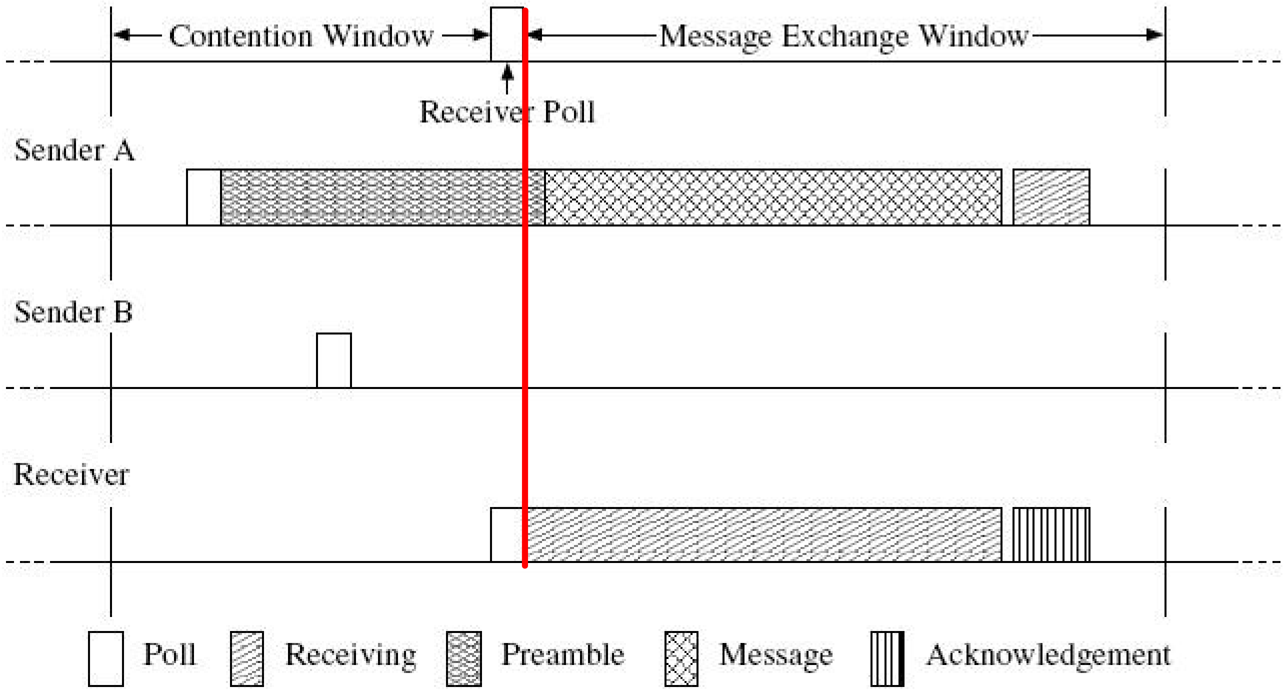
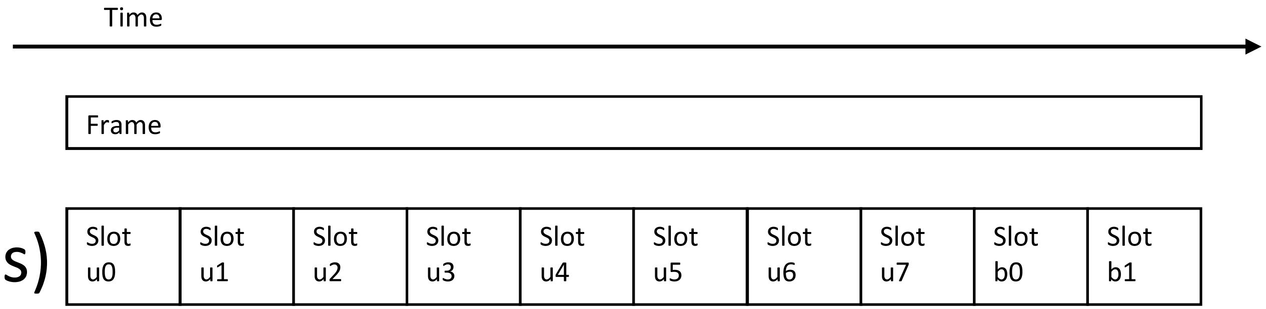


Figure 1: „Contention and message exchange in the Crankshaft protocol“ taken from [1]

Crankshaft: Additional notes

– Slot assignment (MAC address modulo #u-slots)



– Sinks

– Collisions (probabilistic retry)

– Sender and receiver sharing the same slot

The Crankshaft Protocol

- Overhearing
  - Time division
- Communication grouping
  - Time division
- Over-provisioning
  - Division (number of slots) is bounded
- Neighbour state
  - Neighbour state is computable
    - no maintaining necessary

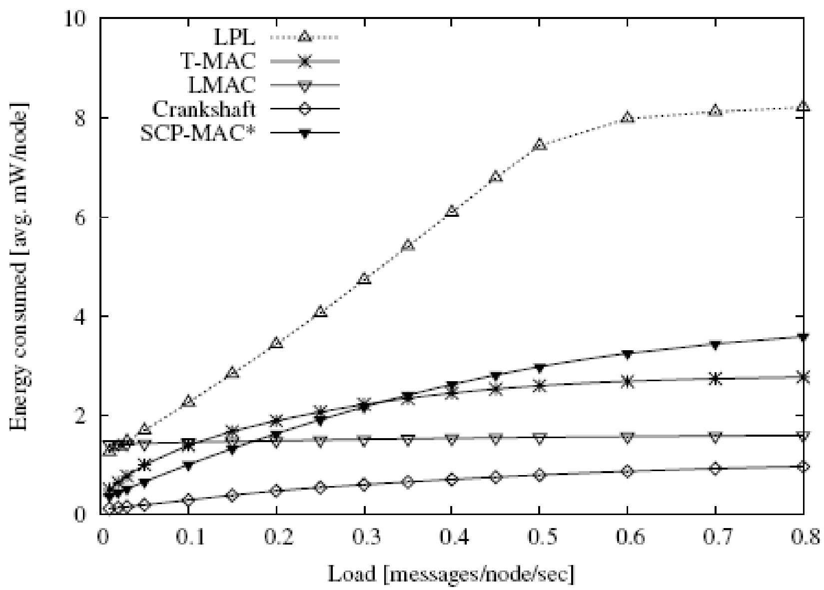
Simulation

- Potato-field experiment
- Connectivity is approximately 17,3 (high density)!
- Two important traffic patterns
  - Convergecast
  - Broadcast

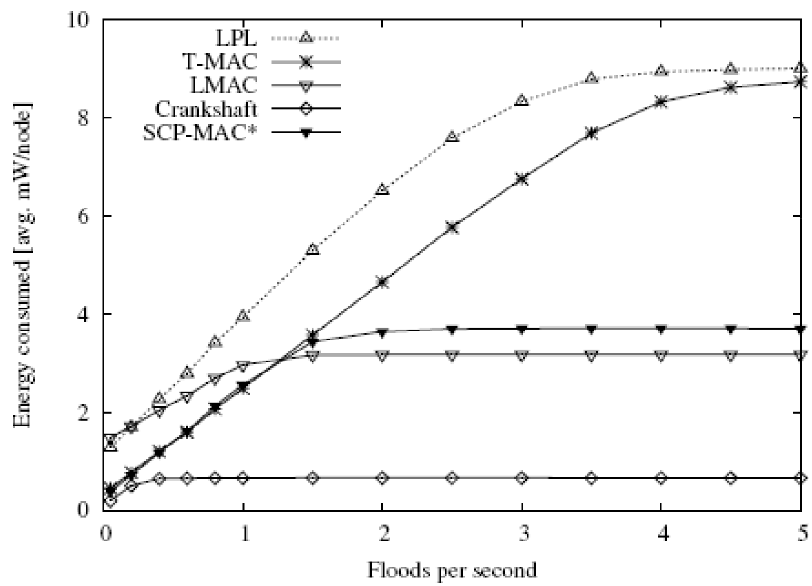
- Parameters
  - 10 slots (8 unicast, 2 broadcast)
  - All protocols‘ specific values

Simulation: Energy

Convergecast



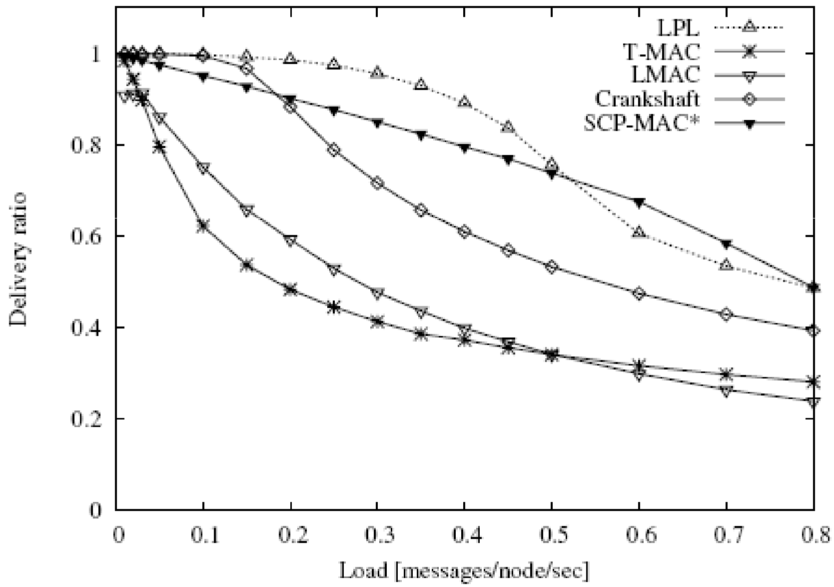
Broadcast flood



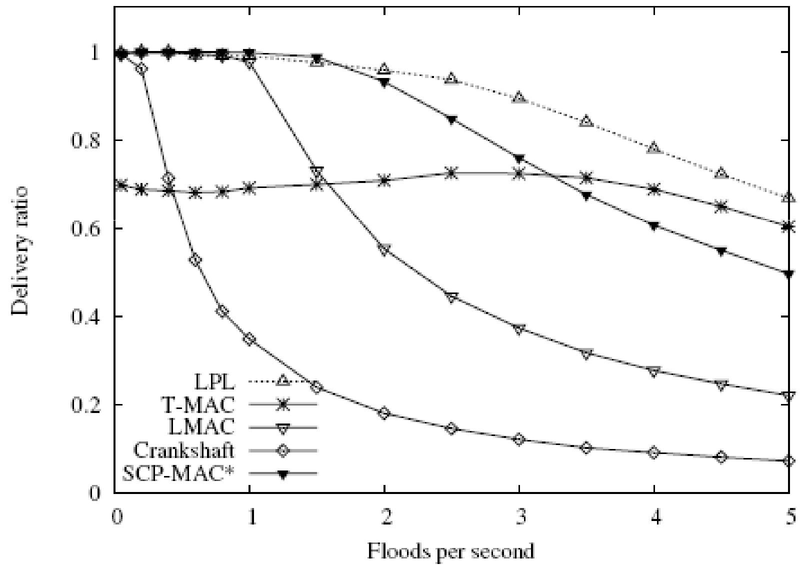
Lowest energy consumption in both scenarios.

Simulation: Delivery ratio

Convergecast



Broadcast flood



Delivery ratio good for convergecast, but poor for broadcast flood. Reasons?

Simulation shows trade-offs

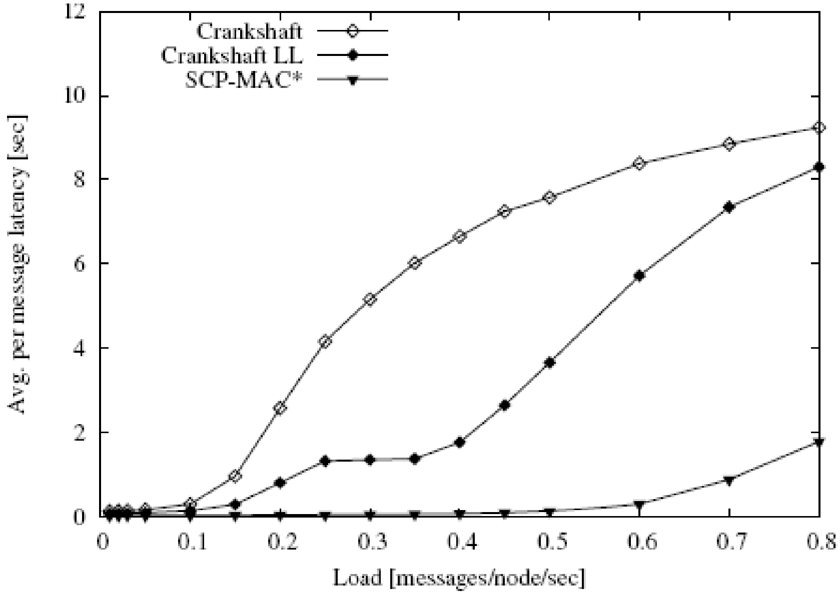
- Poor delivery ratio in broadcast scenario
  - Many messages many collisions a lot of additional traffic more collisions
  - Solution in Crankshaft: retry with lower probability but: higher latency

- Trade-offs
  - Energy efficiency
  - Delivery ratio
  - Latency
  - Broadcast vs. unicast

Simulation: Latency

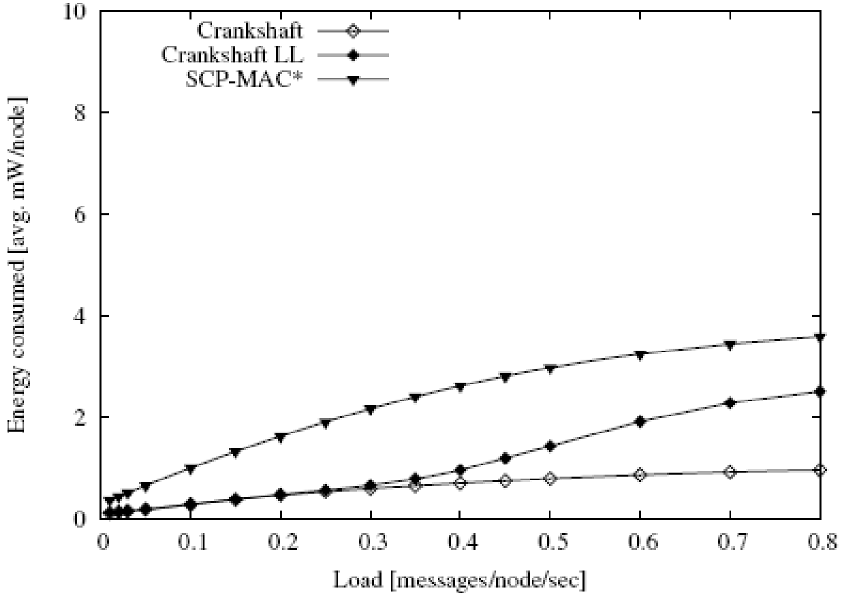
Low Latency Option

- Sending data to another node instead of waiting for one frame



Trade-off: Energy consumption

- Energy consumption increases (opposition to Crankshaft‘s main goal!)



Discussion: Crankshaft vs. IEEE 802.15.4

	IEEE 802.15.4	Crankshaft
Goal	Similar goal: low cost communication	
Time division	Dynamic	Static
Carrier sensing	Superframes	Contention window
Frequences	Three	One (possibly more)
Conditions	Two kinds of nodes : FFD and RFD(full/reduced function devices) FFDs needed as coordinators	

Discussion of Crankshaft

Pro

– Static slot assignment

– Main goal (energy efficiency) is achieved

– No differentiation of nodes (compared to IEEE 802.15.4)

Energy-efficient

Contra

– Static slot assignment

– Bandwidth

– Routing not considered in Simulation (broadcast used extensively)

„One-eyed“

Conclusion

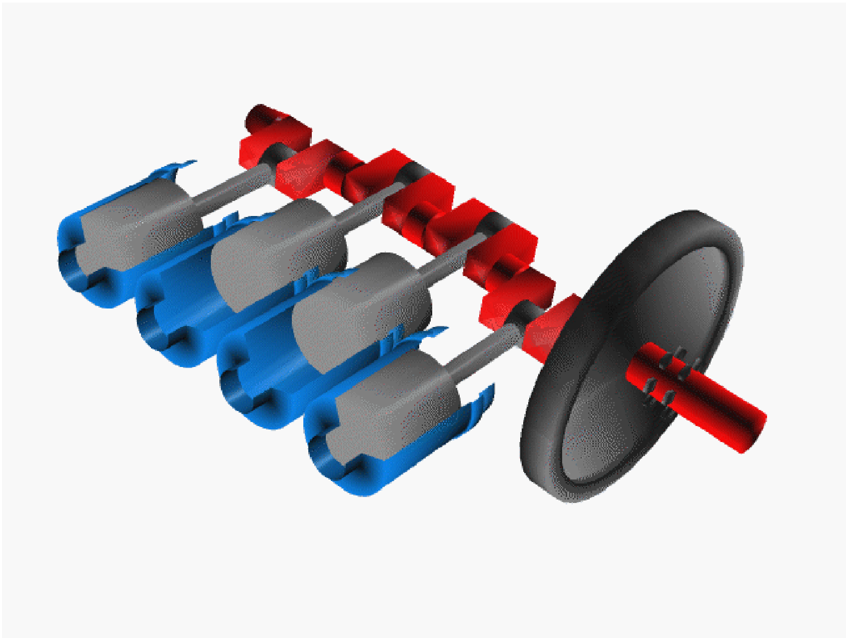
– Crankshaft‘s main goal (reduction of energy consumption) is achieved for the price of less bandwidth and quickly increasing latency.

– Crankshaft is no universal, but a customized energy-efficient protocol for specific (dense) network environments.

Finally

– Outlook of First Presentation:

  Being able to explain the choice of Crankshaft as name for the protocol.



Animation taken from Wikipedia, the free encyclopedia
http://en.wikipedia.org/wiki/Image:Cshaft.gif