# Crankshaft: An Energy-Efficient MAC-Protocol For Dense WSNs

EWSN 2007

G.P. Halkes, K.G. Langendoen

30 January 2007





TU Delft
Delft University of Technology

# Introduction: Standard problems

- Overhearing
- **Idle listening**
- Protocol overhead
- Collisions

# Introduction: Standard problems

- Overhearing
- **Idle listening**
- Protocol overhead
- Collisions

- Solutions: duty-cycling, TDMA, polling
- T-MAC, LMAC, SCP-MAC, etc.

# Introduction: Dense networks

- High connectivity (>15) makes existing MAC protocols energy-inefficient
- Such (dense) networks are not uncommon!

# Introduction: New problems

High density creates new problems:

- Aggravated overhearing
- Communication grouping
- Over-provisioning
- Neighbour state

# Crankshaft: Basic ideas

- Schedule receive slots
- Don’t wake all nodes simultaneously
  - Reduces overhearing
  - No communication grouping
- Wake schedule derived from node ID
  - No per-neighbour state

# Crankshaft: Structure

- Synchronise nodes
- Divide time into frames, and frames into slots
- Nodes wake up in fixed slot within frame, based on node ID

- Broadcast in dedicated slots

# Crankshaft: Structure

- Synchronise nodes
- Divide time into frames, and frames into slots
- Nodes wake up in fixed slot within frame, based on node ID
  - (MAC address) *mod* (#unicast slots)
- Broadcast in dedicated slots

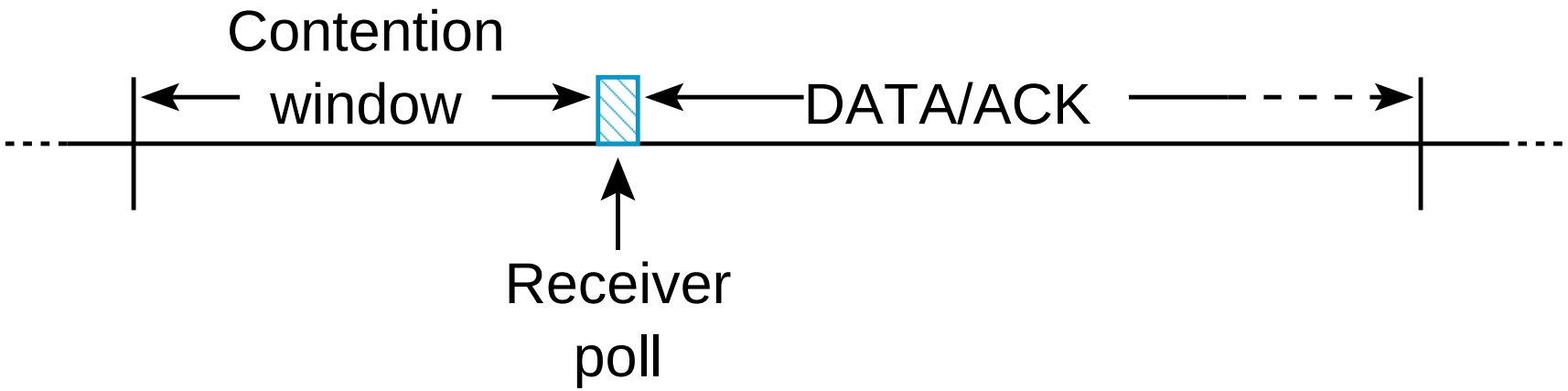
# Crankshaft: Structure

- Synchronise nodes
- Divide time into frames, and frames into slots
- Nodes wake up in fixed slot within frame, based on node ID
  - (MAC address) *mod* (#unicast slots)
- Broadcast in dedicated slots


- Like piston firing at fixed offset from start of crankshaft rotation

# Crankshaft: Slot structure

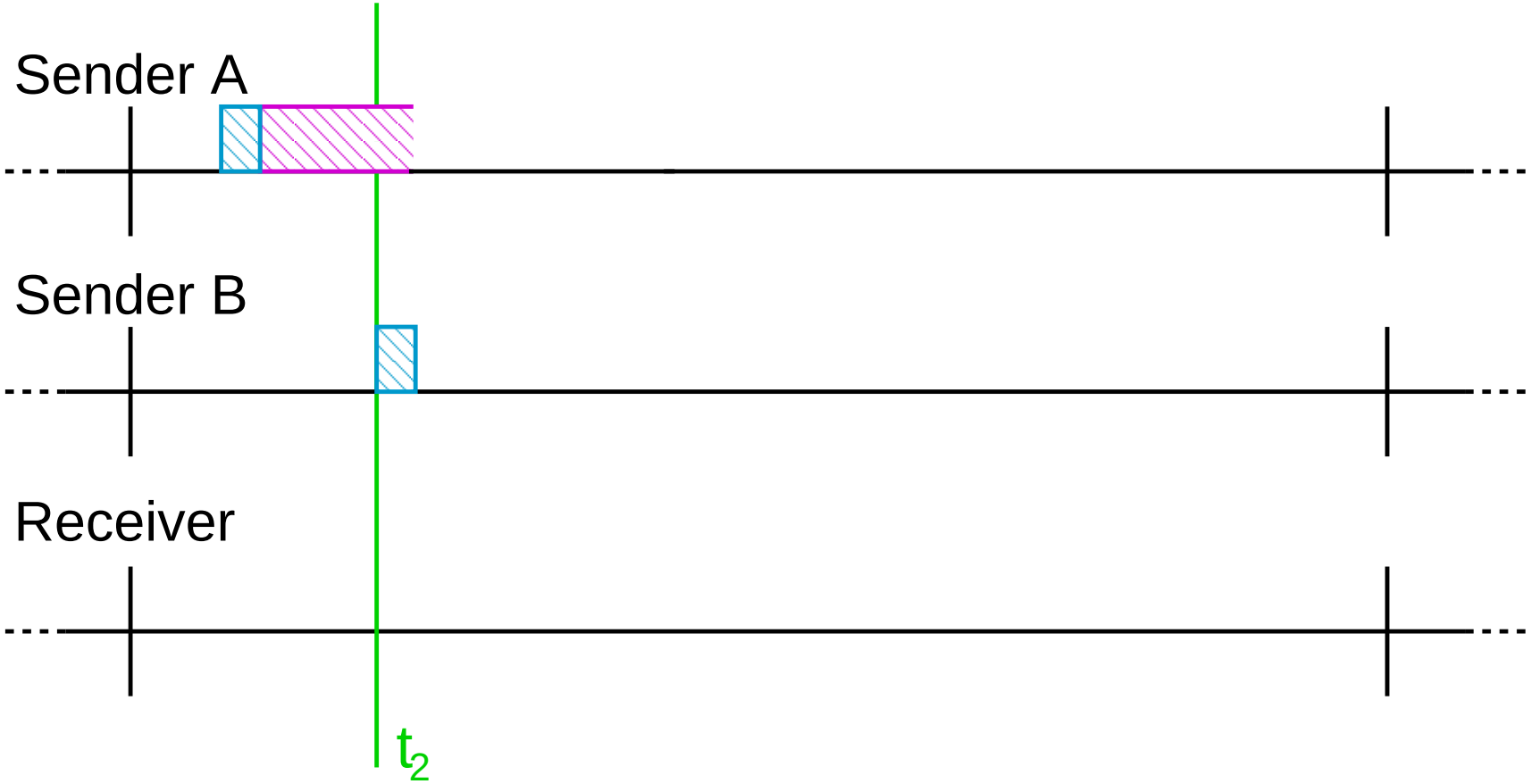
- Inspired by SCP-MAC
- Fixed length contention window
- Short channel poll



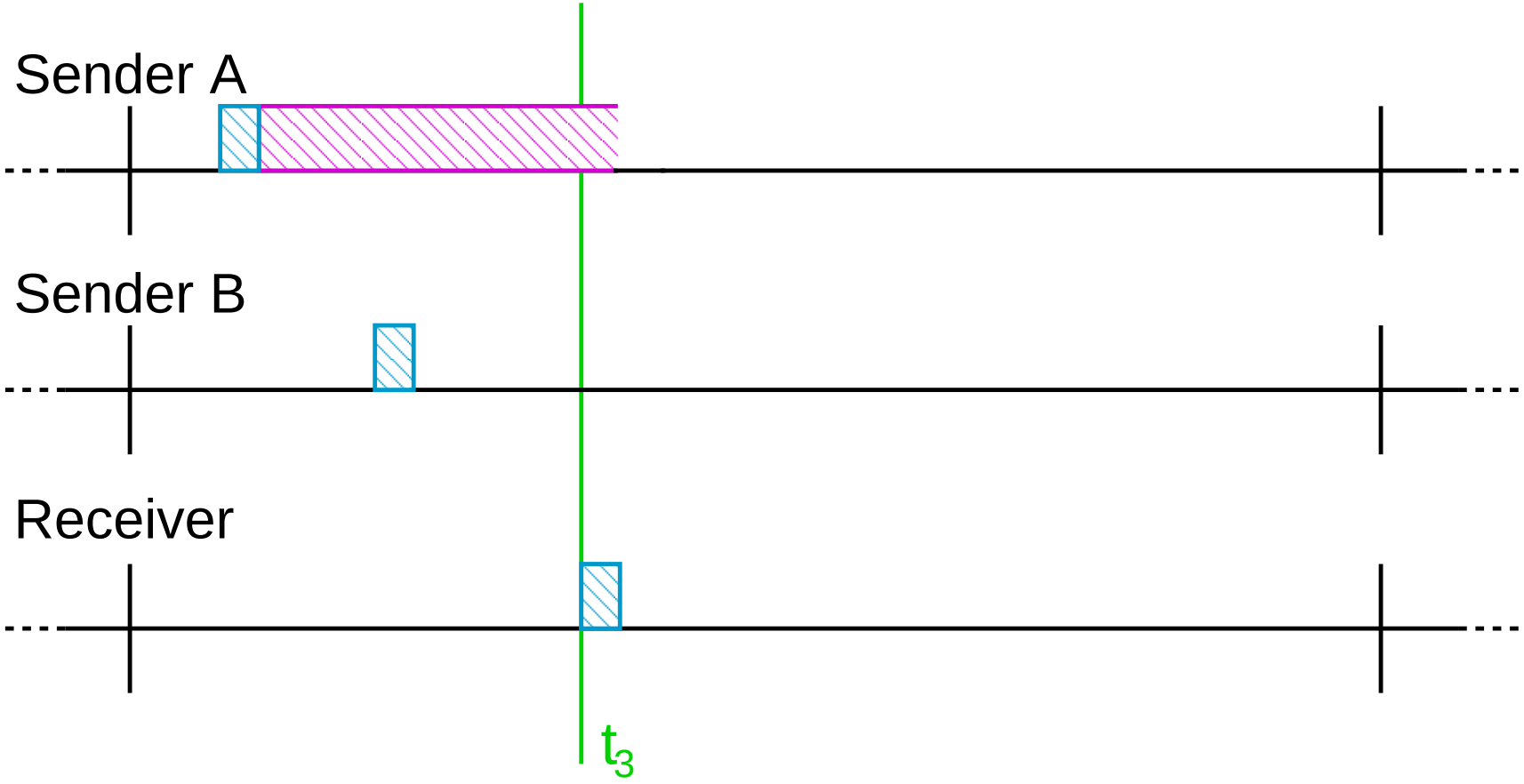
# Crankshaft: Slot structure

# Crankshaft: Slot structure

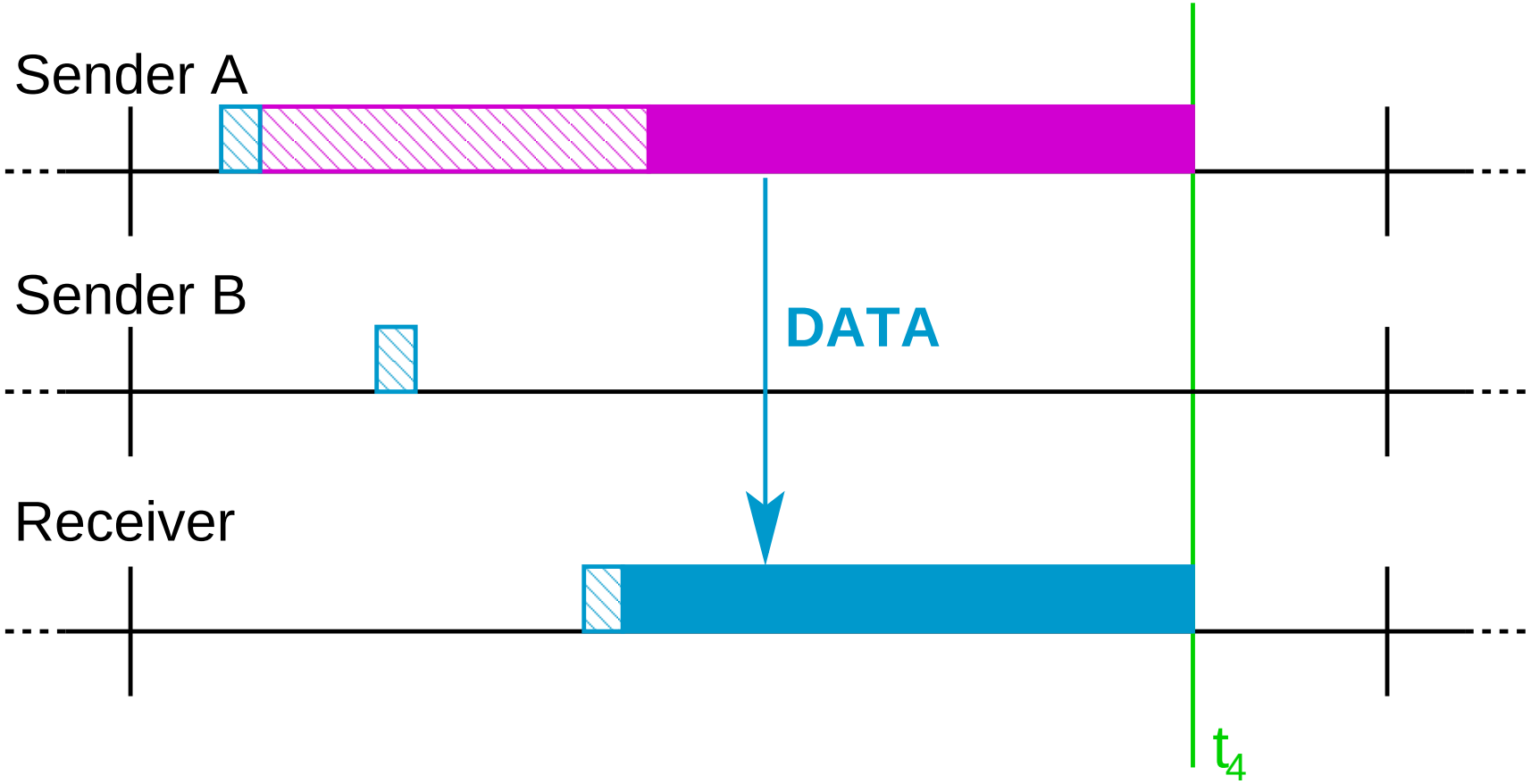
# Crankshaft: Slot structure



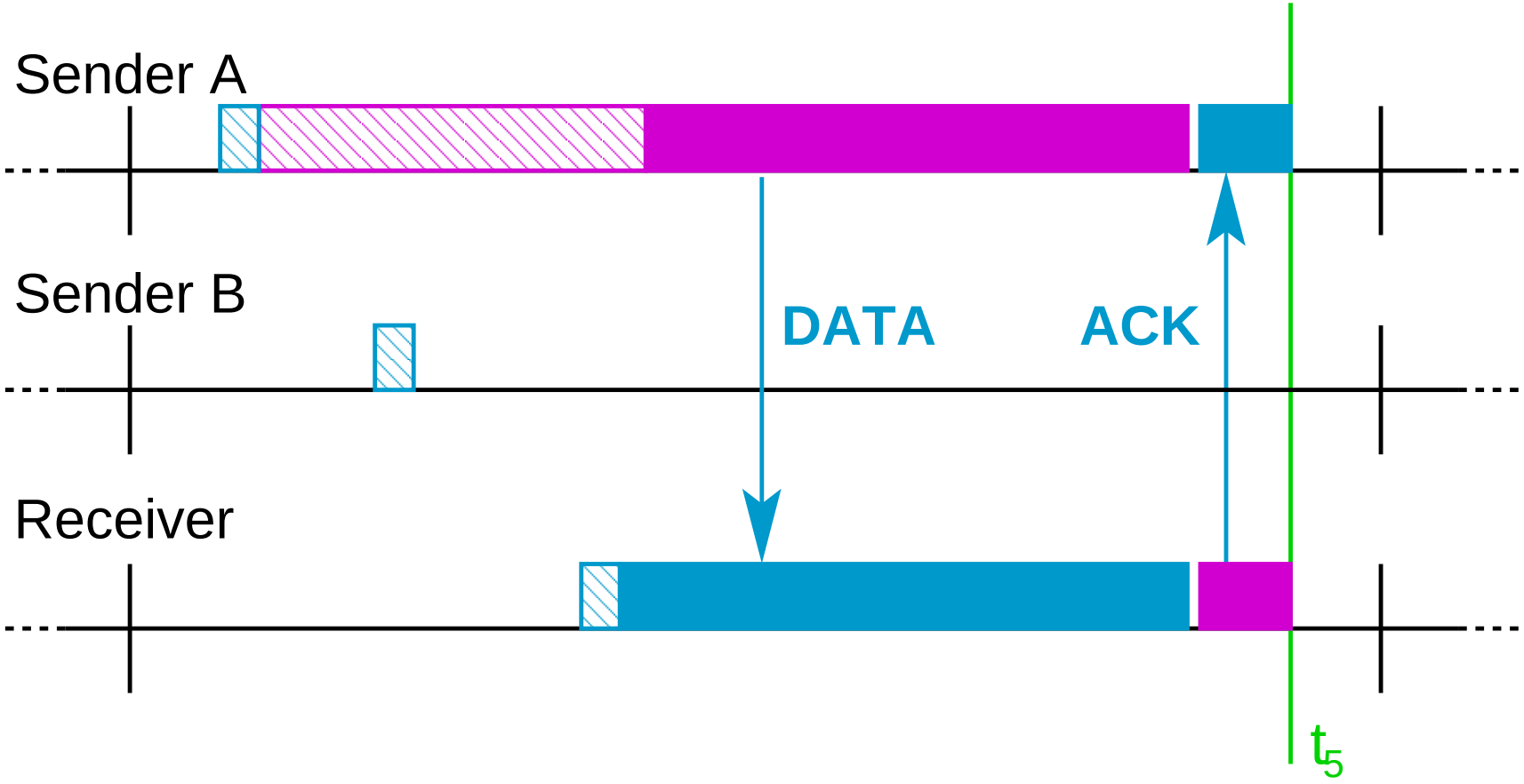
# Crankshaft: Slot structure



# Crankshaft: Slot structure



# Crankshaft: Slot structure



# Crankshaft: More details

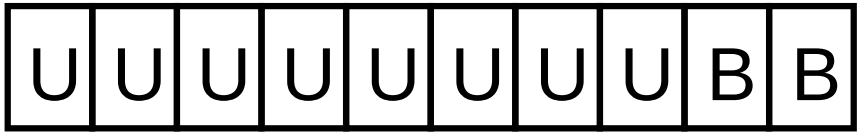
- Collisions will trigger retry
- Retry with probability (70%), to counter hidden terminal problem
- Base station listens in **all** slots
- Receivers may contend in their own slot

# Evaluation: Simulator setup

- Based on real deployment
- 96 nodes
- Field 90×50 meters
- Reception range 25 meters
- Average connectivity 17.3
- Network diameter 4 hops
- Base station in corner

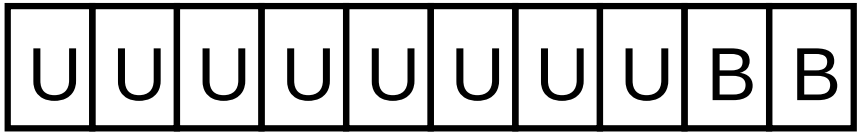
# Evaluation: Protocols

- T-MAC, LMAC, BMAC, SCP-MAC*
- Crankshaft parameters
  - 8 unicast slots
  - 2 broadcast slots

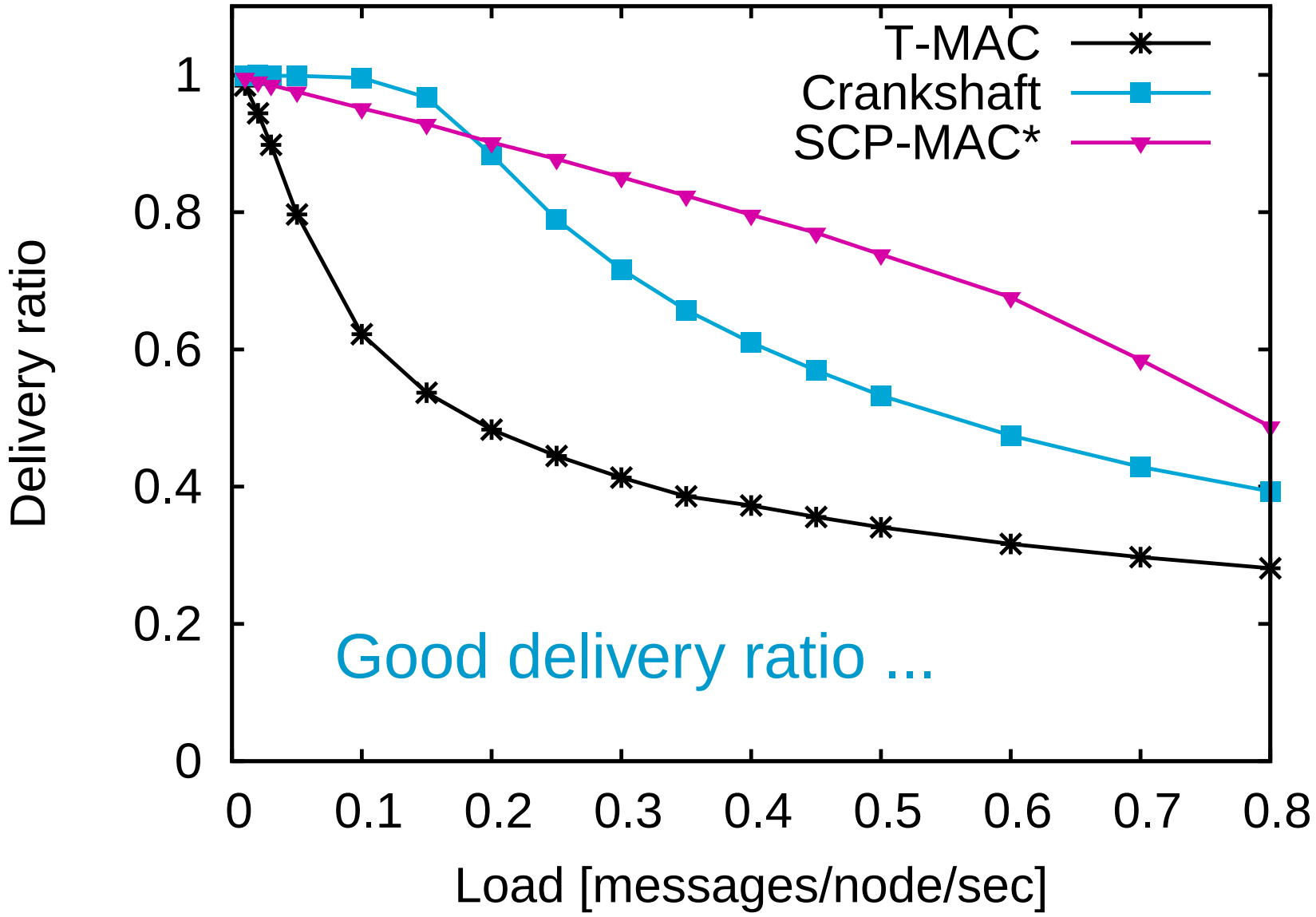


# Evaluation: Protocols

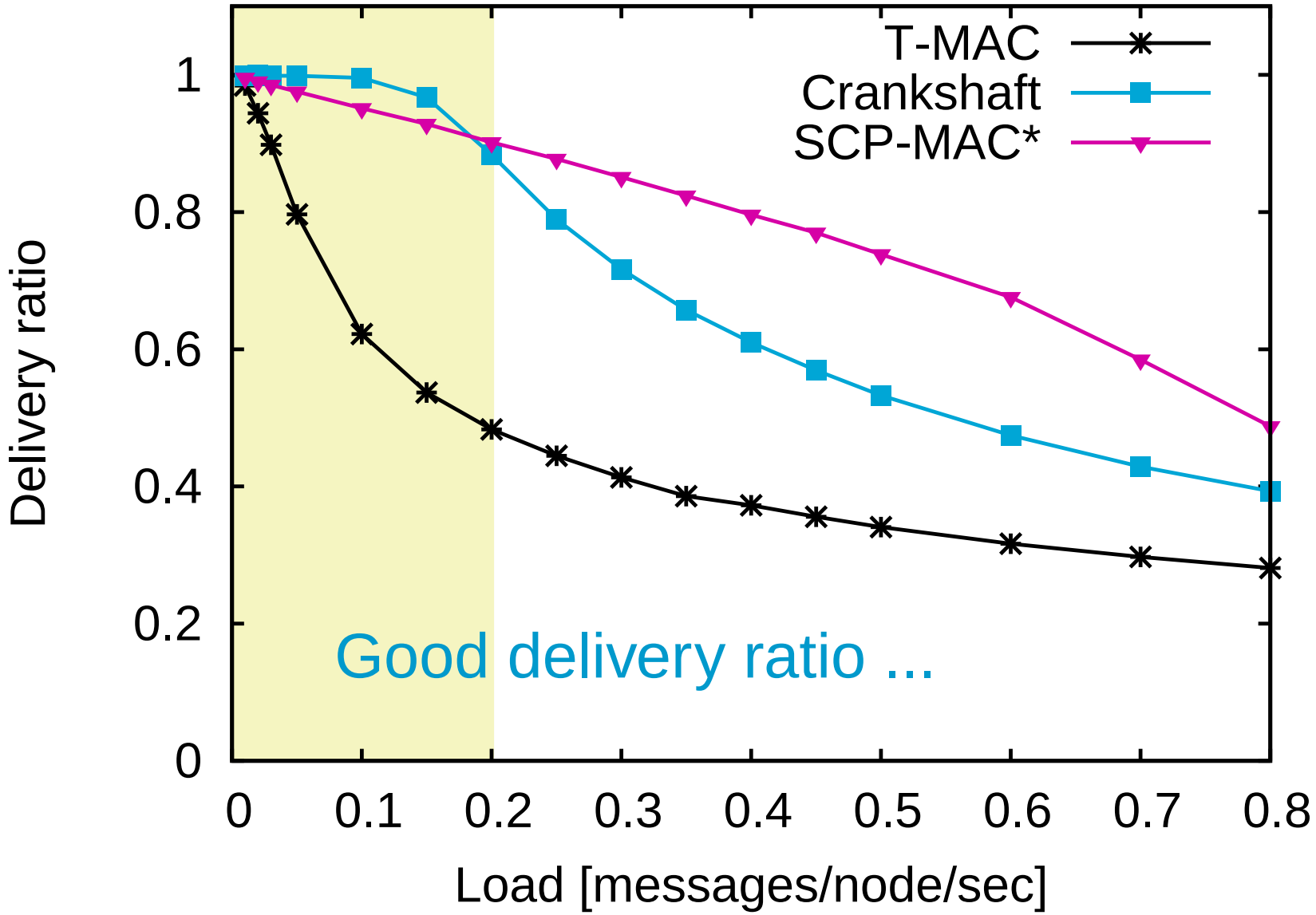
- **T-MAC**, LMAC, BMAC, **SCP-MAC***
- Crankshaft parameters
  - 8 unicast slots
  - 2 broadcast slots



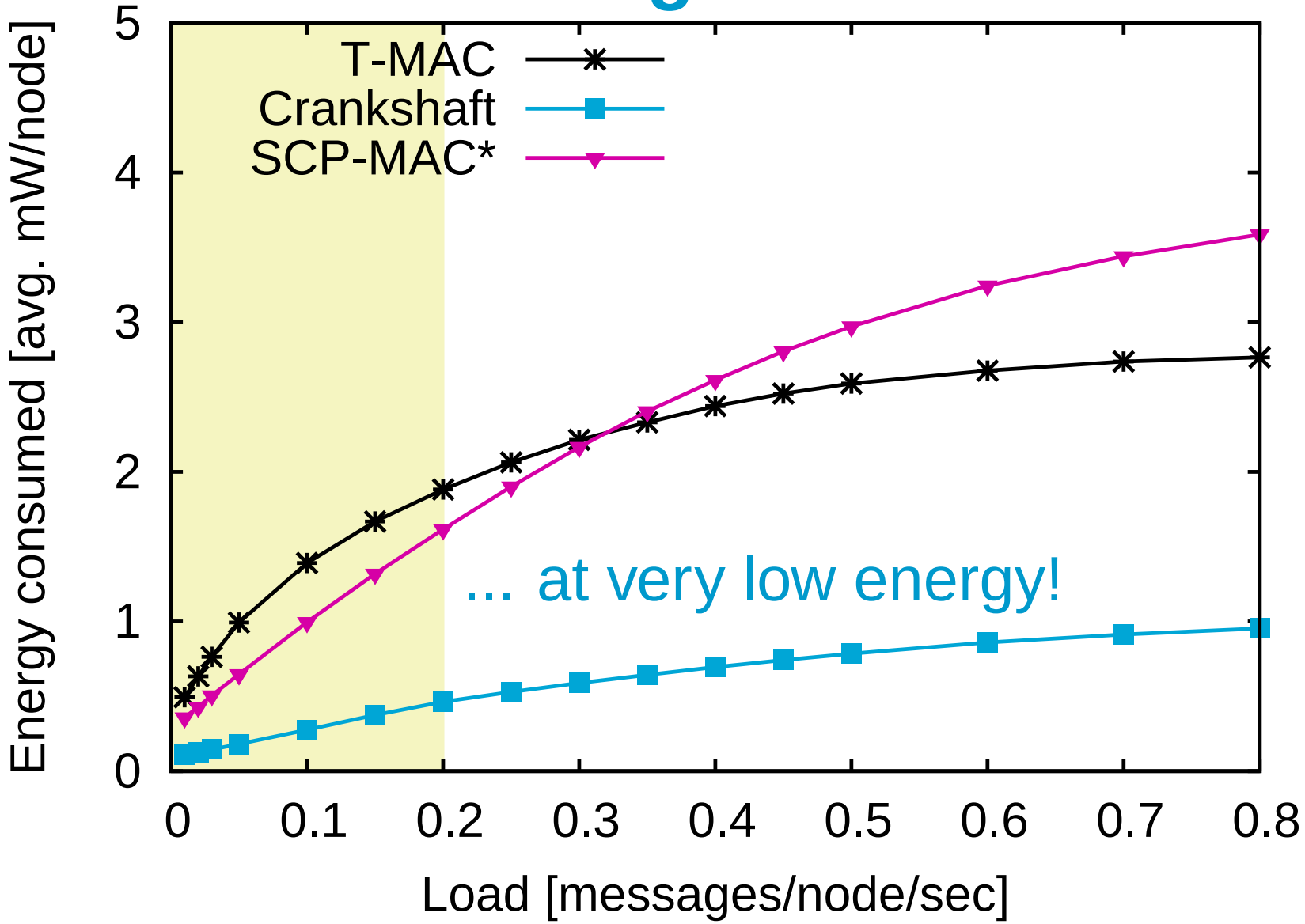
# Results: Convergecast



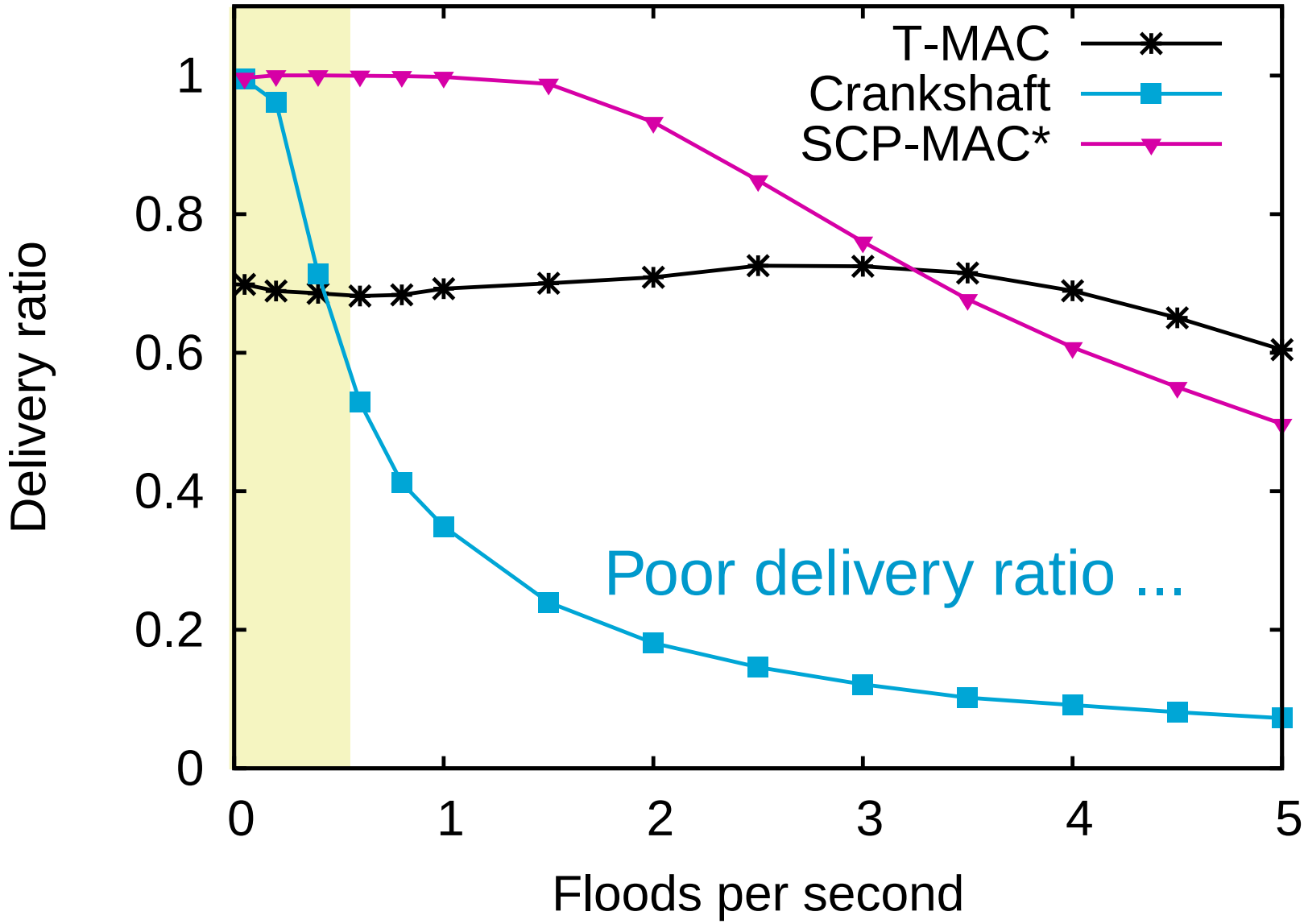
# Results: Convergecast



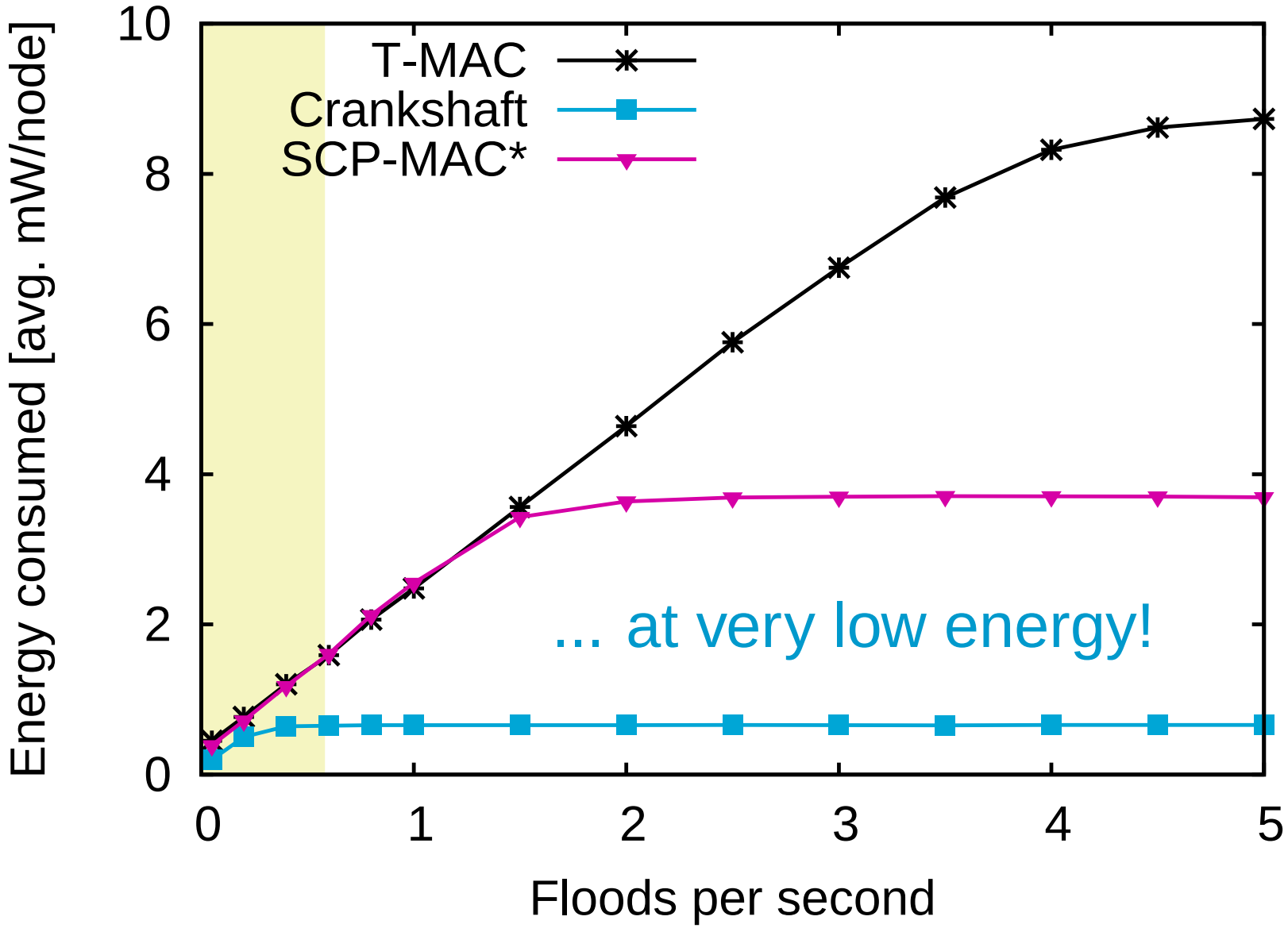
# Results: Convergecast



# Results: Broadcast flood



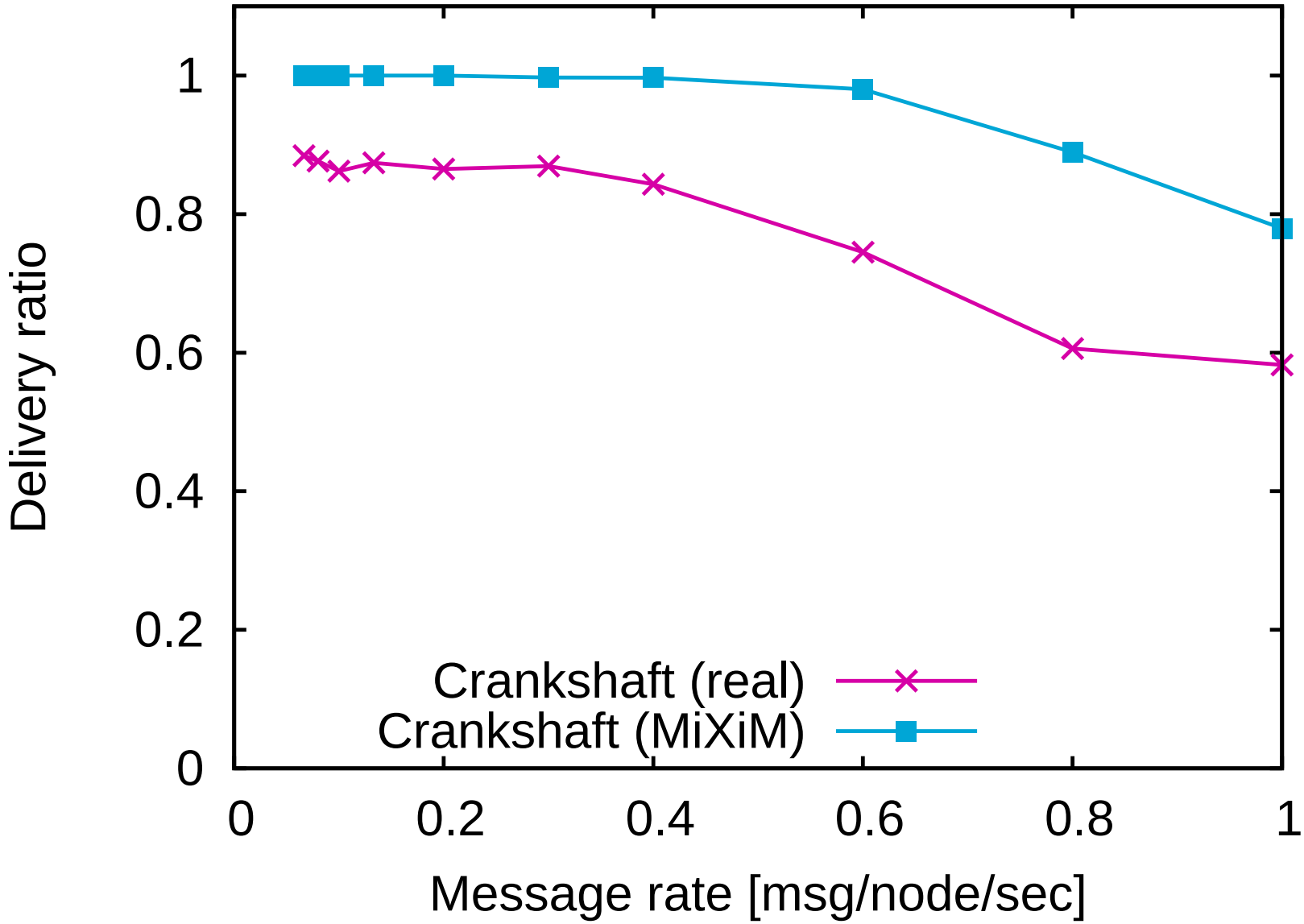
# Results: Broadcast flood



# Results: Real world

- 26 nodes
- Office building
- Network diameter 2 hops
- All nodes interfere with eachother

# Results: Real world



# Conclusions

- Overhearing
- Communication grouping

} → Nodes receive in different slots

- Over-provisioning → Slots not unique in neighbourhood
- Neighbour state → Derive wake-up slot from node ID

- Very energy efficient
- Broadcast performance suffers

# Questions