
Xenofon Fafoutis and Nicola Dragoni
{xefa, ndra}@imm.dtu.dk
Outline

• Motivation
• ODMAC: An On-Demand MAC protocol for Energy Harvesting WSNs
• Evaluation through Simulations
• Conclusive remarks
Energy Harvesting

• Battery-powered WSNs
  – Eventually will die and need battery replacement
    • Often not even possible (e.g. underground sensors)
  – Sacrifice performance for lower energy consumption

• Energy-Harvesting WSNs
  – Extracting energy from the environment
    • Solar, mechanical, thermal, etc.
    • Energy sources have spatiotemporal variations
  – Batteries / Super-capacitors operate as energy buffers
Energy Neutral Operation

- Energy Neutral Operation (ENO) provides continuous lifetime
- ENO-Max also maximizes the performance
  - Performance is strongly correlated with energy consumption
Designing EH-WSNs

Design Objective
• Operate at the maximum sustainable performance (ENO-Max)

Requirements for EH-WSNs
• *Adaptability*: Sensors should be able to adapt their energy consumption according to the energy harvesting rate
• *Performance*: Sensors should use their energy efficiently
• *Flexibility*: Capable sensor should be able to help the others

Requirements for MAC protocols
• Support for *individual duty cycles*
  – Sleeping / Activity periods cannot be synchronized!
• Efficient use of energy (e.g. mitigate idle listening)
Proposed approach: ODMAC

On Demand MAC (ODMAC)

- Sensors periodically broadcast beacons
  - According to their *individual duty cycle*
  - Stating their availability to *receive* frames
- Sensors with data to transmit are waiting for an appropriate beacon
  - Some energy wasted in idle listening (*challenge*)
- Sensors send a new beacon after a transmission to avoid congestion
- Typical back-off mechanism to avoid collisions

![Diagram of ODMAC](image)
ODMAC: Opportunistic Forwarding

Opportunistic Forwarding
- Forward the frame to the sensor that wakes up first
- Decreases the *sleeping delay* => Increases performance
- Decreases the energy wasted in idle listening
- For now, all the sensors closer to the sink are potential forwarders
  - Future Work: Routing algorithm extensions to account for other metrics

![Time diagram with sensors and states]

- Sensor_1
- Sensor_2
- Sensor_3

- Blue: Transmitting
- Green: Receiving
- Red: Idle Listening

*Time*
Duty Cycle Adaptation

- **Goal**: Adjust performance to the available environmental energy

- Two application-specific performance metrics
  - End-to-end delay (beaconing rate)
  - Amount of measurements (sensing rate)

- Dynamic Duty Cycle Adaptation
  - **SProb**: Probability that if there is a need for adjustment to the duty cycle, this will favor the sensing duty cycle
  - Simple algorithm (out of the scope)
    - Select an optimum battery level and periodically make adjustments to the duty cycle
Evaluation using Simulations

OPNET Simulator

- Energy Model
  - Accounts for the energy consumption when transmitting, receiving and listening
  - Periodic energy harvesting

- Topology
  - 9 sensors (3 groups of 3 sensors)
  - Each sensor can talk with the sensors of its own group and the neighboring groups

- Evaluation Metrics
  - Harvested to consumed energy ratio (sustainability)
  - End-to-end delay, average sensing rate (performance)

- Parameters
  - $P_{tx} = 10$ dBm, $Rate = 1$ Mbps, $CW = 8$
  - Dynamic Duty Cycle Adaptation is OFF (unless otherwise noted)
  - Energy Harvesting Rate is 400µW (unless otherwise noted)
Achieving ENO-Max State

Details
- Beacon Period is 0.2 sec
- Sensing Period is 0.6 sec
- Sprob is 0.5

Static Duty Cycles

Dynamic Duty Cycles
Energy Availability vs. Performance

Details
- Sensor 1-B has activated the Dynamic Duty Cycle Adaptation mechanism
- Beacon Period is 0.2 sec
- Sensing Period is 0.6 sec
- Different values for $SProb$
Load Balancing

Details

- All nodes have fixed duty cycles
- Apart from Sensor 1-B
- Sensing period is static (SProb=0)
Ongoing Work

• Apply boundaries to the sensing and beacon periods
  – Defined by the application
  – Incorporate a way to “slow down” the too capable nodes

• Introduce acknowledgements/retransmissions and
  – Evaluate them under channel errors

• Exploit beacons to propagate control messages (e.g. acks)
  – Energy-free flooding

• Study ODMAC using an analytical model
  – Arbitrary topologies
  – Effect of power adaptation
  – Incorporate routing decisions
Concluding Remarks

• The environmental energy sources have a dynamic nature

• EH-WSNs need to be able to adapt to the available energy
  – Use the surplus of harvested energy to increase performance
  – Decrease performance to maintain a sustainable operation

• MAC protocols need to efficiently support individual duty cycles

• ODMAC
  – Receivers decide on the period they offer forwarding services
  – Opportunistic forwarding reduces the energy wasted on idle listening
  – Distributed autonomous load balancing
  – Supports different application-based performance metrics
The End

Questions?