An Experimental Study of Routing and Data Aggregation in Sensor networks

Ossama Younis
Department of Electrical & Computer Engineering
University of Arizona

Sonia Fahmy
Department of Computer Science
Purdue University

LOCAN 2005
In this paper

- We consider a hierarchically organized multi-hop network and do not assume the presence of any specially-equipped nodes

- We integrate node clustering with tree aggregation in the “iHEED” system

- We implement the iHEED system in TinyOS and evaluate it on Berkeley motes to show its effect on network lifetime
Outline

- Introduction
- System model
- The HEED protocol
- The iHEED system
- Evaluation of iHEED
- Concluding remarks
Introduction

➢ In sensor networks:
  ➢ Nodes may be deployed and left unattended
  ➢ Battery lifetime is limited
  ➢ Radio communication is a major source of energy consumption

➢ Objective:
  ➢ Reduce energy consumption to prolong the network lifetime
Energy-efficiency

Network lifetime:
- The time until the first (last) node in the network dies
- The time until the observer is disconnected

Alternatives:
- Node duty cycle
- Redundant node deployment
- Efficient topology management

Our approach
- Construct a hierarchical (clustered) network
Communication models

Flat topology

Hierarchical (clustered)
A set of $n$ sensor nodes are dispersed uniformly and independently in a field

- **Network**: ad-hoc, unattended, and no infrastructure support
- **Nodes**: quasi-stationary, equally significant, location un-aware, and their transmission power levels are tunable
- **Application**: Source-driven or data-driven with high load and opportunity for data aggregation
HEED clustering (Infocom’04, TMC’04)

- **Initialization**
  - Discover neighbors within cluster range
  - Compute the initial cluster head probability $\text{CH}_{\text{prob}} = f(E_r/E_{\text{max}})$

- **Main processing**
  - If received cluster head messages, choose one head with min. cost
  - If no cluster head message is heard, elect to become a cluster head with $\text{CH}_{\text{prob}}$.
    - $\text{CH}_{\text{prob}} = \min(\text{CH}_{\text{prob}} \times 2, 1)$
    - Repeat until $\text{CH}_{\text{prob}}$ reaches 1

- **Finalization**
  - If cluster head is found, join its cluster
  - Otherwise, elect to become cluster head
HEED example

1. Discover neighbors
2. Compute $\text{CH}_{\text{prob}}$ and cost
3. Elect to become cluster head
4. Resolve ties
5. Select your cluster head

Diagram showing a network of nodes labeled a1 to a14 with associated costs and probabilities.
HEED implementation (iHEED)

- Implement HEED in TinyOS
- Integrate clustering with multi-hop routing

Application:
- Data aggregation at a base station
- Consider simple data aggregators, such as AVG, MIN, MAX, SUM, and COUNT

Challenges:
- Tracking the battery level
- Handling node asynchrony
Hardware platform

- The Mica2 mote has a 7.38 MHz processor, while the Mica2Dot has a 4 MHz microprocessor.
- 128 KB program memory, 4 KB RAM, and 512 KB non-volatile storage.
- The radio is a Chipcon SmartRF CC1000, with 916 MHz frequency,
- FSK modulation with data rate 38.4 kBaud (19.2 Kbps) and Manchester encoding.
- Output power is digitally programmable by setting the PA_POW register.
iHEED: Computing residual energy

- Store a soft-state view of residual energy by
  - Using a Credit-Point (CREP) system
  - Decrementing the available points periodically

- Example: Two 1.5V AA batteries for Mica2
  - Given: packet size = 36B, bit time = 62.4 μsec
  - $E_{\text{max}} = 2.2 \text{ A-hr} \times 3\text{V} \times 3600 \text{ sec} = 23760 \text{ Joule}$
  - $E_{\text{tx}} (1 \text{ packet}) = 16.8 \text{ mA} \times 3\text{V} \times 62.4 \mu\text{sec} \times 288 \text{ bits} = 0.9 \text{ mJoule}$
  - $E_{\text{rx}} (1 \text{ packet}) = 8 \text{ mA} \times 3\text{V} \times 62.4 \mu\text{sec} \times 288 \text{ bits} = 0.43 \text{ mJoule}$
iHEED: Handling node asynchrony

- Nodes can start operation at different times
- Clustering is triggered every $T_{NO}$ seconds
- To handle asynchrony:
  - Every node announces its state (cluster head) with every routing update
  - A new node starts the clustering process when it starts its operation
  - When $T_{NO}$ expires at a cluster head, it sends a routing update to trigger its neighboring cluster heads and cluster members to start the clustering process
The iHEED system
Evaluation

- Experimental setup:
  - 6 Mica2 + 4 Mica2Dot sensors + base station
  - Distributed in a research lab
Experimental setup

- Packet size: 36 bytes
- Routing update: 1 packet every 10 seconds
- Data rate: 1 packet every 2 seconds
- $E_{\text{max}} = 150,000 - 350,000$ points
- Intra-cluster trans. cost = 230 points/packet
- Inter-cluster trans. cost = 291 points/packet
- Re-cluster the network every 6 (or 9) minutes
Network lifetime is prolonged by a factor of 2-4
Evaluation -- #successful transmissions

# successful transmissions is almost doubled
Overhead of iHEED is less than that of COLLECT
Conclusion

- Integrating clustering with data aggregation has important advantages:
  - Prolongs the network lifetime
  - Reduces channel contention
- The overhead incurred with clustering is small, compared to that of forwarding
- Integration with node duty cycle is essential for more energy conservation