Distributed Energy-Efficient Scheduling Approach for k-Coverage in Wireless Sensor Networks

C. T. Vu, S. Gao, W.P. Deshmukh, and Y. Li

Georgia State University
Presentation outline

- Energy-Efficient Scheduling
- $k$-coverage problem
- Sensor Energy-Efficient Scheduling for $k$-coverage - SESK problem
- Distributed Energy-Efficient Scheduling for $k$-coverage – DESK algorithm.
- Simulation result
Energy-Efficient Scheduling

- **Objective:** Maximum network lifetime and balance energy consumption among sensors

- **Example:** Lifetime of each sensor is 2

Initial network

Schedule 1: $C_1 = \{(S_1, S_3), 2\}$
Lifetime = 2. Waste energy: $E_{S_2}$
Energy-Efficient Scheduling (cont.)

Schedule 2: Lifetime = 3.
Waste energy = 0

\[ C_1 = \{(S_1, S_2), 1\} \]
\[ C_2 = \{(S_1, S_3), 1\} \]
\[ C_3 = \{(S_2, S_3), 1\} \]
k-coverage problem

- *k*-coverage for an area
- *pcl* – Perimeter coverage level
- *k*-perimeter-coverage for a sensor

Outside the area: *

\[ pcl = \text{INF} \]

\[ k = 2 \]
Sensor Energy-Efficient Scheduling for k-coverage - SESK

- **Given:**
  - Monitored area $A$.
  - Set of sensors $S = \{s_1, s_2, .., s_N\}$.

- **Find:**
  - A schedule $(C_1, t_1), \ldots, (C_m, t_m)$
    - $C_j$: non-disjoint set cover, $t_j$: active time

- **Objective:**
  - Maximize $\sum t_j$.
  - $k$-cover whole area $A$.
  - Energy consumption is balanced.

- **is NP-complete**
  - SESK is general problem of SET K-COVER problem.
Distributed Energy-Efficient Scheduling for k-coverage - DESK

• Rule proposed by Huang & Tseng:
  – The whole area is $k$-covered iff each sensor is $k$-perimeter-covered.
Distributed Energy-Efficient Scheduling for k-coverage - DESK (cont.)

- DESK is completely distributed & localized.
- DESK works in rounds

![Network time line diagram]

- **Round 1**
- **Round 2**
- **...**
- **Round R**

- **Decision phase**
- **Sensing phase**

All sensors simultaneously run DESK in this phase
DESK (cont.)

- Useless/Useful neighbor
- A *counter* — number of neighbors to whom $s$ is useful.
- Each sensor $s$ maintain a waiting timer $w$ whose value depends on:
  - Residual energy, sensing range.
  - Current *counter* value.

![Diagram](k=1)
DESK run at a sensor s

- Receive message from neighbor v
- $t \geq w$
- end of decision phase?

- mACTIVATE
  - Update pcl
  - Send mASK2SLEEP to any useless neighbors.

- mASK2SLEEP
  - counter --
  - Update $w$.

- mGOSLEEP
  - Remove v out of neighbors' list

- counter = 0
  - YES
    - Send mGOSLEEP
    - Go to SLEEP
  - NO
    - Set to be ACTIVE
    - Send mACTIVATE

- STOP
Example: for $k=1$

- $S_1$ send $\text{mACTIVATE}$ to $S_2, 3, 4, 5$
- $S_2$ send $\text{mASK2SLEEP}$ to $S_3$
- $S_3$, $\text{counter} = 3$ and update $w$

- $S_2$ send $\text{mACTIVATE}$ to $S_1, 3, 4, 5$
- $S_1, 4, 5$ send $\text{mASK2SLEEP}$ to $S_3$
- $S_3$, $\text{counter} = 0$, update $w$

This part is outside

All are covered by $S_2$
Simulation setting

• Monitored area: $800^\text{m} \times 800^\text{m}$.
• Sensing range: $400^\text{m} \rightarrow 500^\text{m}$.
• Decision phase: 2 seconds
• Round length: 20 minutes
Simulation:
Network lifetime with different number of sensors

![Graph showing network lifetime with different number of sensor nodes. The x-axis represents the number of sensor nodes, ranging from 50 to 200. The y-axis represents network lifetime in hours, ranging from 2000 to 3000. The graph includes lines for different values of k: k=1 (black squares), k=2 (blue circles), k=4 (red stars), and k=8 (pink triangles).}]
Simulation: Network lifetime with different power ratio
QUESTION?