

**Centralized Algorithms for Lifetime of
Wireless Sensor Networks based on
Dependency Structure among Cover Sets**



Characteristics of Sensor Networks

- Low cost
 - Usually deployed in large numbers
 - Constraints – energy. Limited power supply in the form of a battery.
 - But the fact that sensors are deployed in large numbers means that there is significant overlap of their monitoring regions
 - *Idea:* Use a subset of these sensors
-
-

Problem

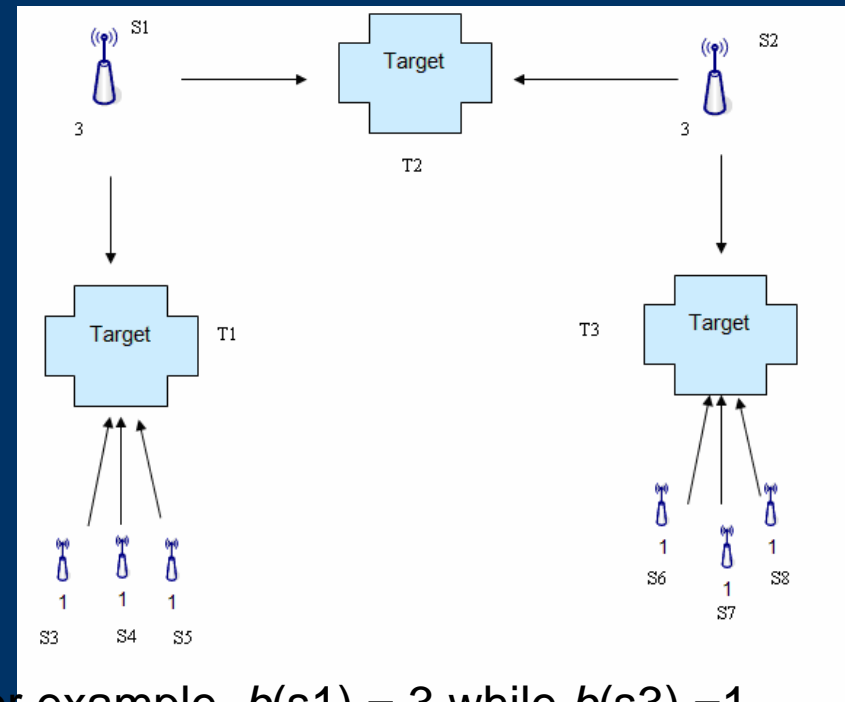
- Come up with a sleep-sense schedule for all sensors such that the lifetime of the network is maximized.
- Shown to be a NP-complete problem
- Centralized and Distributed heuristics proposed in the literature.



Existing work

- Centralized approaches are typically based on LP [Berman04][Cardei05].
 - Distributed algorithms based on rounds where at the beginning of a round a sensor negotiates with its neighbors usually on some simple greedy criteria.
-
-

Notation



$b(s)$: strength of a battery of sensor s ; for example, $b(s_1) = 3$ while $b(s_3) = 1$.

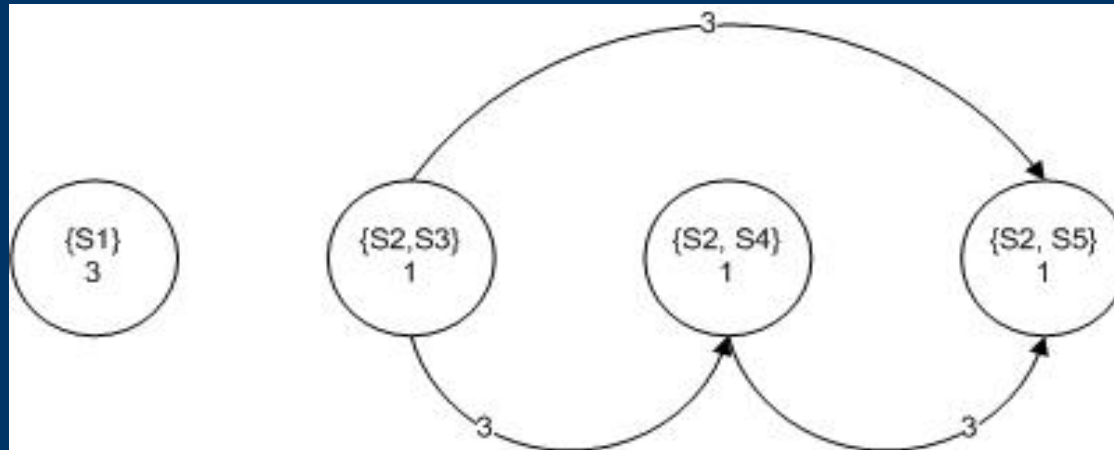
$T(s)$: set of targets that sensor s can sense; e.g., $T(s_1) = \{t_1, t_2\}$

$N(s, k)$: set of neighbors of sensor s at no more than k hops $N(s_1, 1) = \{s_1, s_2, s_3, s_4, s_5\}$.

Cover: C is a cover for targets in set T if (i) for each target $t \in T$ there is at least one sensor in C which can sense t and (ii) C is minimal. For example, the possible (minimal) covers for the two targets of s_1 are $\{s_1\}$, $\{s_2, s_3\}$, $\{s_2, s_4\}$ and $\{s_2, s_5\}$.

$lt(C) = \min_{s \in C} b(s)$, the maximum lifetime of a cover. The bottleneck sensor of the cover $\{s_2, s_3\}$ is s_3 with the weakest battery of 1. Therefore, $lt(\{s_2, s_3\}) = 1$.

Model: Lifetime Dependency Graph



$G = (V, E)$ where nodes in V are cover sets and edges in E are sets of common sensors between pairs of nodes, that is, an edge e between two covers C and C' is $C \cap C'$. Also,

$w(e) = \min_{s \in e} b(s)$, the weight of an edge e .

$d(C) = \sum_{e \in E \text{ and incident to } C} w(e)$, the degree of a cover C .

Shown for sensor s_1 considering its one-hop neighbors $N(s_1, 1)$ and its targets $T(s_1)$.

Aims

- a. Design centralized heuristics based on the *Lifetime dependency* graph model,
 - b. Compare the performance of these new heuristics to existing Linear Programming based solutions,
 - c. Possible obtain better upper bounds for the lifetime problem.
-
-

Node Authentication via Covert Channels in Ad-Hoc Wireless Networks

The Premise: In an ad-hoc wireless networks, neighboring nodes shall be able to authenticate themselves to one another via a covert channel.



The Covert Channel

- Create windows of β units of time
- Each window represents a symbol within a set S
- Packets are sent between neighbors to determine routing information, or to route data
- Hold packet for $\delta * \beta$ time (where δ is the index of a symbol in S)



Scenario

- Node 1 (node requiring authentication)
 - Node 2 (node receiving the authentication)

 - Node 2 sends a packet through Node 1 (broadcast or unicast packet to be routed)
 - Node 1 waits ($\beta * \delta$) and automatically transmits the packet.
 - This may result in a collision
 - This may result in a passive ACK
 - If either a passive ACK or a collision occurs during the time frame, Node 1 can authenticate Node 2
-
-

Pros & Cons

- Pros
 - Perfectly covert
 - Clock synchronization is not necessary (uses relative timing)
 - [should] provide a highly accurate method to authenticate ones neighbors
 - Cons
 - Reduces network throughput (retransmissions)
 - Reduces node throughput (waiting to send packet)
 - Packet being routed though the node is required
 - Assumes symmetry between nodes
-
-