Robust Communications for Sensor Networks in Hostile Environments

Ossama Younis and Sonia Fahmy
Department of Computer Sciences, Purdue University

Paolo Santi
Instituto di Informatica e Telematica, CNR, Italy

In this work

- We propose a distributed clustering algorithm (REED) that builds a k-connected sensor network when the network is dense
- REED uses different transmission ranges for intra-cluster and inter-cluster communications
- Rationale:
  - Clustering → scalability and longevity
  - K-connectivity → robustness in case of unexpected failures, and multi-path communication

Outline

- Motivation
- System model
- Requirements
- Approach
- Analysis
- Simulation results
- Conclusions

Motivation

- In hostile environments, such as military fields or volcanic areas, sensor nodes may fail unexpectedly. Thus, higher degree of connectivity is desirable
- Sensor nodes have limited batteries and are left unattended. Thus, energy efficiency is essential for longer network lifetime

Goals

- Clustering the sensor network to provide communication scalability and longer network lifetime
- Building k-connected (k-fault tolerant) networks for robust communications

⟹ our goal is to construct a **clustered, fault-tolerant** sensor network

System model

- A set of *n* sensor nodes are dispersed uniformly and independently in a hostile field

Network

- ad-hoc, left unattended, and does not necessarily have any infrastructure

Nodes

- quasi-stationary, equally significant, location un-aware, and each possesses a fixed number of transmission power levels

Application

- Nodes may fail unexpectedly

Requirements

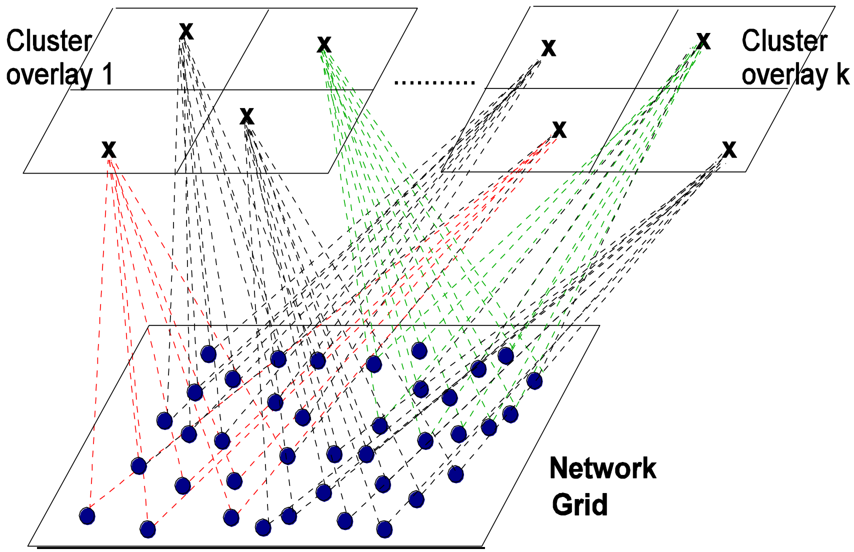
- Design a clustering protocol that has the following properties:
  - Constructs a k-connected network at both the intra-cluster and inter-cluster levels
  - Uses two different transmission ranges for cluster formation and inter-cluster communications
  - Converges in O(1) time with low message and processing overhead
  - Considers network longevity and load-balancing

Approach

- We propose the **REED** (Robust, Energy-Efficient, Distributed) clustering protocol
- REED is distributed:
  - Every node only uses information from its 1-hop neighbors (within cluster range)
- REED provides robustness:
  - A *k*-connected network is constructed (asymptotically almost surely) for dense networks
- REED is energy-efficient:
  - Elects cluster heads that are rich in residual energy
  - Re-clustering results in distributing energy consumption

Basic ideas

- Every node should have *k* distinct cluster heads
- Every cluster head should be able to communicate with at least *k* other cluster heads



REED approach:

Construct k distinct (node-disjoint) cluster head overlays

REED – Clustering parameters

- Consider energy-efficiency as well as robustness
- Parameters for electing cluster heads
  - Primary parameter: residual energy (E_r)
  - Secondary parameter: communication cost (used to break ties)

i.e., maximize energy and minimize cost

- Nodes autonomously elect to become cluster heads with probability $CH_{prob} = f(E_r/E_{max})$

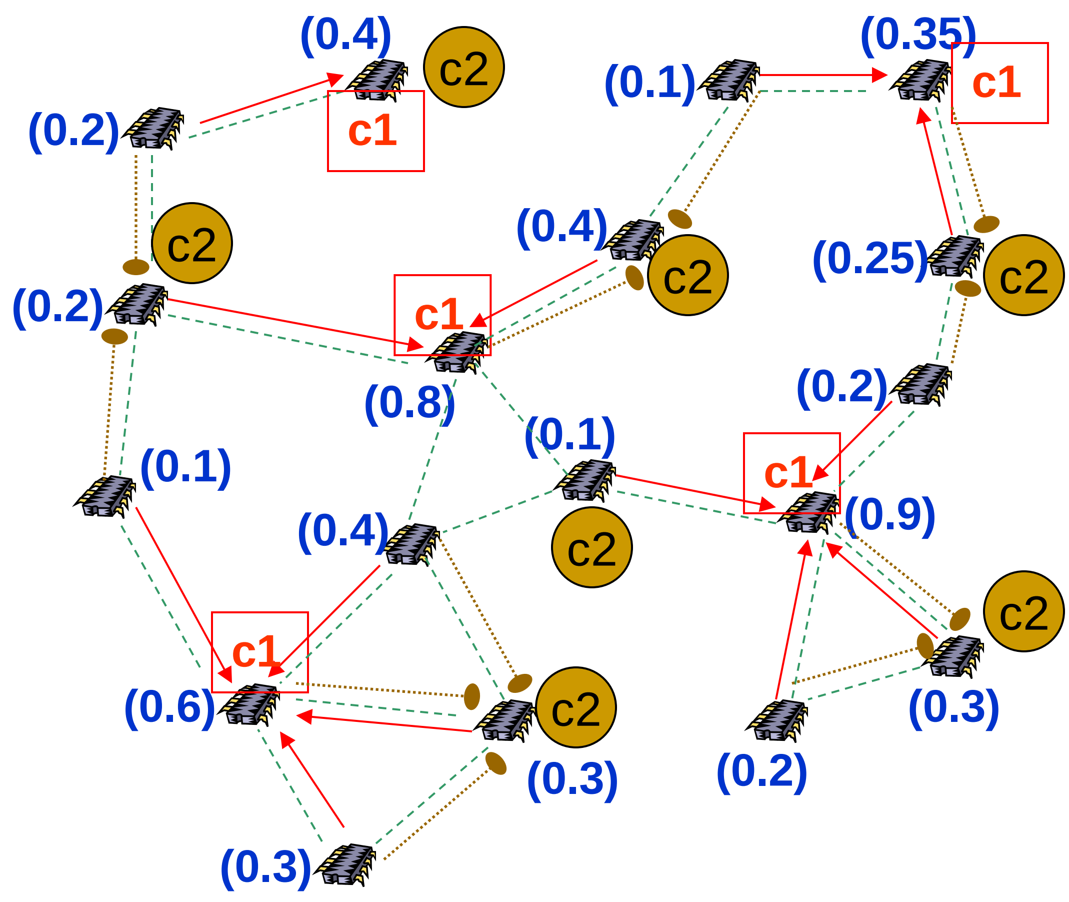
The REED protocol at node v

- Initialization
  - Discover neighbors within cluster range
  - Compute the initial $CH_{prob} = f(E_r/E_{max})$

- Main processing
  - Repeat while $CH_{prob} \leq 1$
    - Arbitrate among received cluster head messages to select min cost heads (one for each cluster head overlay CS_i, $1 \leq i \leq k$)
    - If v does not have a cluster head for CS_i, elect to become a cluster head for CS_i with CH_{prob} .
    - $CH_{prob} = \min(CH_{prob} * 2, 1)$
  - Continue listening to messages

- Finalization
  - For every CS_i, if a cluster head is found, then join its cluster. Otherwise, elect to become a cluster head for CS_i,

REED in action (k = 2)



(1) Discover neighbors within cluster range

(2) Compute CH_{prob} and cost

(3) Select your cluster head or elect to become cluster head for overlay c1 then c2

(4) Elect to become cluster head if one is not found

Analysis

- The interleaved clustering process is $\Omega(k)$, and k is a constant. Let p_{min} = constant (minimum E_r/E_{max})

$$N_{iter} \leq \left\lceil \log_2 \frac{1}{p_{min}} \right\rceil + 1$$

- Message overhead: $O(k)$ per node
- Processing overhead: $O(kn)$ per node

Connectivity

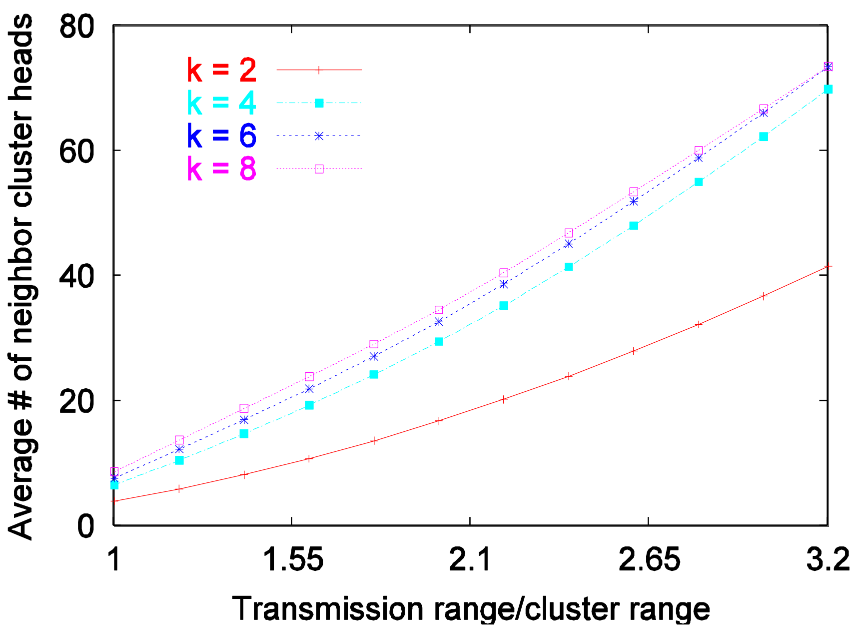
- Assumptions:
  - *n* nodes are dispersed in an area $A=[0,L]^2$.
  - A is divided into N square cells, each of side $\frac{R_c}{\sqrt{2}}$
  - R_t denotes the inter-cluster communication range
  - A cluster head overlay is connected if $R_t \geq 6R_c$ (Younis and Fahmy, Infocom'04)
  - Let η(n,N) be a random variable that denote the minimum number of nodes in a cell
- Requirement:
  - What is the necessary condition(s) to (asymptotically) have $\lim_{n,N \to \infty} \eta(n,N) \to k$?

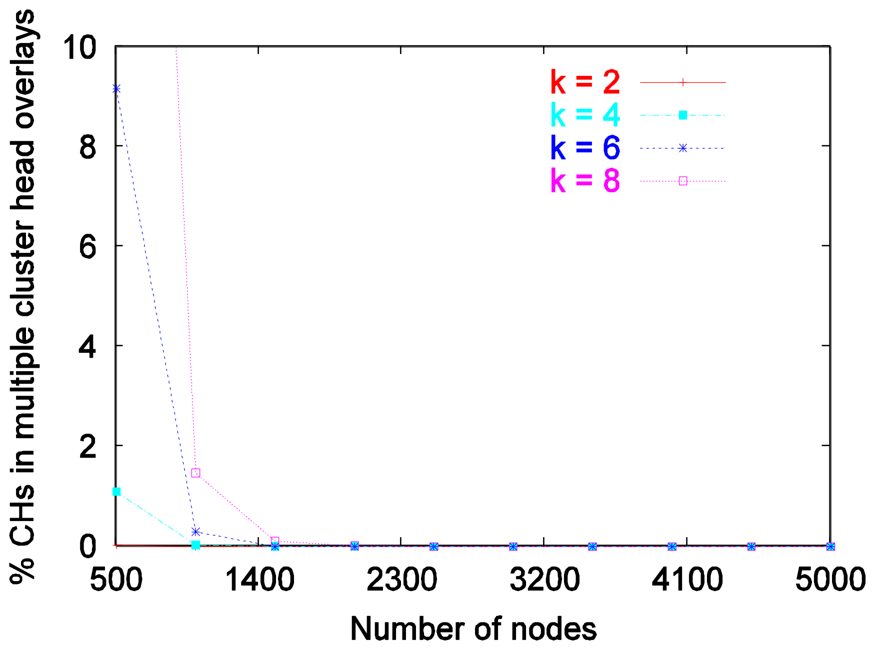
Connectivity (cont^{d})

- Theorem:
  - For any fixed arbitrary $k > 0$, assume that *n* nodes are dispersed uniformly and independently in an area $A=[0,L]^2$. Assume A is divided into N square cells (as defined above). If $R_c^2 n = aL^2 \ln N$, for some $a \geq 2$, $R_c \ll L$, and $n \gg 1$, then $\lim_{n,N\to\infty} E(\eta(n,N)) \to k$, iff $k \approx \ln N$
- Proof sketch:
  - (Kolchin'78): A minimally occupied cell has h or h+1 nodes a.a.s. if $\lim n/(N \ln N) \to 1$, as $n,N \to \infty$, and h is chosen such that $h < \alpha$ ($\alpha = n/\ln N$), and $Np_h(\alpha) \to \lambda$, where $p_h(\alpha) = (\alpha^h/h!) \times e^{\alpha}$.

Evaluation

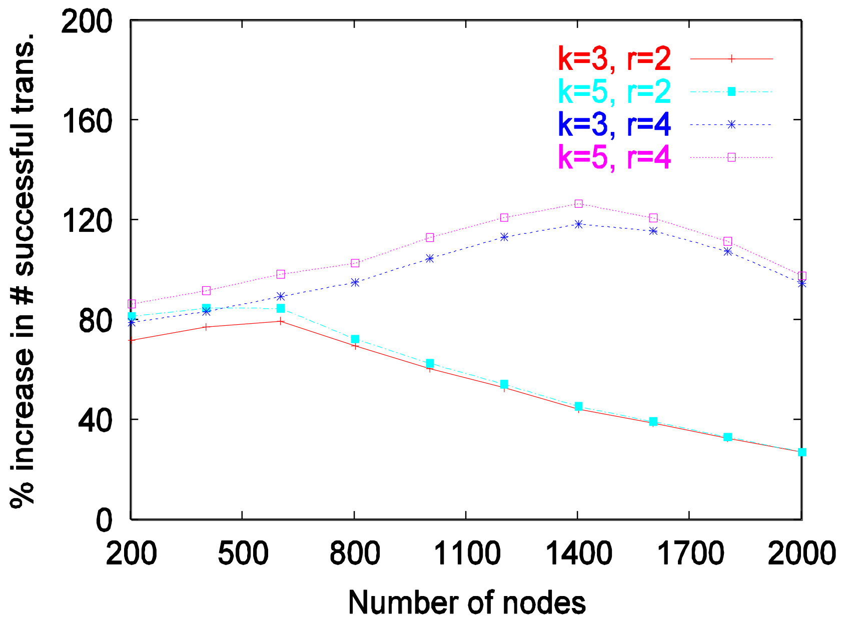
- Connectivity and cluster head uniqueness
  - Network area = 100 m x 100 m
  - Cluster range = 10 m

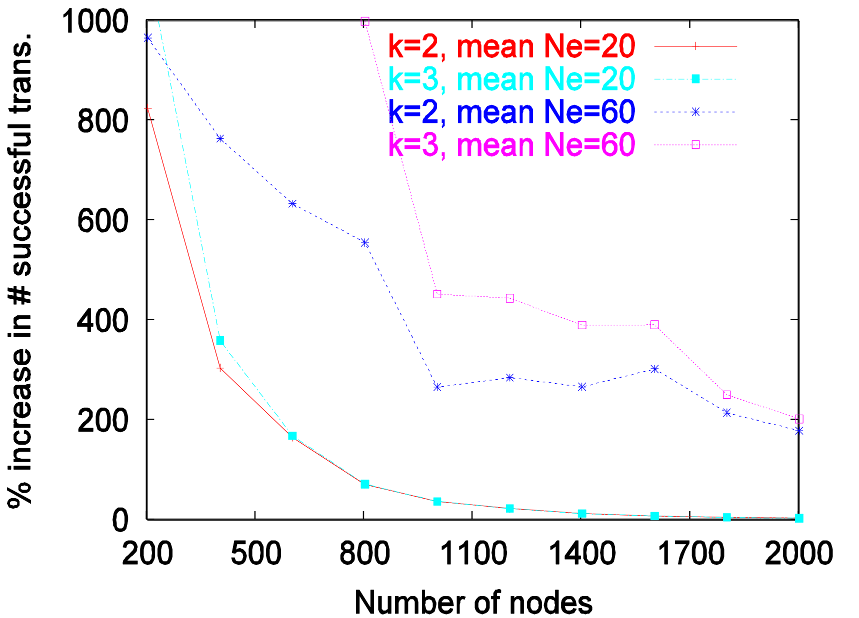




Evaluation (cont[d])

- Fault tolerance properties





- Single-hop communication with a base station
- Multi-hop communication between border nodes

Additional advantage

- For sufficiently large density:
  - Each cluster head overlay is a connected component that can route among any pair of nodes in the network
  - Each set of cluster heads in cluster head overlays has distinct nodes
- Therefore, there is an opportunity for multi-path communication
- This is useful, e.g., for security protocols.

Related work

- Previous research on fault-tolerance in ad-hoc networks suffers from limitations when applied to sensor networks, e.g.,
  - Centralized approaches, such as Hajiaghayi et al., 03, are not suitable for large-scale networks
  - Complexity (e.g., solving LP problem, Cheriyan et al., 02), is critical for sensors with limited processors
  - Requirement of special equipment (e.g., CBTC, Wattenhofer et al., 01) is not always feasible for sensor nodes

Conclusions

- We have presented, REED, a distributed clustering protocol for constructing k-fault tolerant sensor networks
- REED can be constructed on top of other clustering protocols, such as LEACH
- For future work, we will investigate:
  - How to adapt REED to changing node density
  - How REED performs under different node distributions
  - How to exploit the independent cluster head overlays to secure sensor communications, e.g., for symmetric key distribution