Energy-Efficient Connectivity Re-establishment in WSN in the Presence of Dumb Nodes

Pushpendu Kar, Arijit Roy, Sudip Misra
School of Information Technology
Indian Institute of Technology Kharagpur
Kharagpur 721302, India
Email: {pkar, arijitr, smisra}@sit.iitkgp.ernet.in

Mohammad S. Obaidat
Fellow of IEEE, Fellow of SCS
Dept. of Computer Science and Software Engineering
Monmouth University, West Long Branch, NJ, USA
Email: obaidat@monmouth.edu

Abstract—In this work, we propose a scheme, named LECRAD, for the re-construction of temporarily lost connectivity in the presence of dumb nodes in stationary wireless sensor networks. A sensor node is termed as “dumb”, when it continues its physical sensing, but fails to communicate due to shrinkage in communication range, typically attributed to adverse environmental effects such as rainfall, fog and high temperature. On the resumption of favorable environmental conditions, the node starts to behave normally. So, dumb behavior is dynamic in nature. Such behavior of a node leads to network partitioning and node isolation, which results in disruption of connected topology. Therefore, the proposed scheme in this paper reconstructs the lost connectivities between nodes by activating intermediate sleep nodes or by adjusting the communication range of the sensor nodes, while there is no neighbor node within the reduced communication range. In the proposed scheme, a learning automata-based approach is used for activation of intermediate sleep nodes or adjustment of communication range of isolated or intermediate nodes to decrease the message overhead and energy consumption of the network. Simulation results show that the proposed scheme, LECRAD, exhibits better energy efficiency and message overhead than that of the recently proposed topology management protocols, LETC and AI, if they are applied in such scenario.

Index Terms—Dumb Node, Environmental effect, Connectivity re-establishment, Learning automata.

I. INTRODUCTION

A wireless sensor network (WSN) is a collection of wireless sensor nodes deployed in a planned manner or randomly, which work collaboratively to sense their physical surroundings, while transmitting the sensed information to a central station via single-hop or multi-hop connectivity [1]. Characteristics of WSN include limited energy and resources, low computational capability, and dynamic topology. There are several useful applications of WSN, such as security, surveillance, target tracking, disaster management, military, health care, wildlife monitoring, and environment monitoring [2], [3]. Cooperation and collaboration are major considerations in multi-hop wireless networks for their proper functioning. Misbehaviors, faults, and attacks are the challenges in cooperation and collaboration among sensor nodes in WSN [4] [5] [6]. In this work, we consider a specific type of misbehavior, termed as “dumb” behavior [7]. A sensor node behaves as dumb, when it continues its sensing operation, but fails to communicate due to the shrinkage in communication range attributed to the sudden occurrence of adverse environmental effects such as rainfall, fog, and high temperature. The dumb nodes behave normally on the resumption of favorable environmental condition. So, dumb behavior of a node is dynamic in nature.

In this work, we present a scheme for the reconstruction of lost connectivity, which has temporarily broken due to node isolation and network partitioning due to the occurrence of dumb behavior among sensor nodes. We activate intermediate sleep nodes or adjust the communication range of sensor nodes for the re-establishment of connectivity. To reconstruct the lost connectivity, a scheme Learning automata based Energy-efficient Connectivity Reconstruction using Adjustable sensors in the presence of Dumb node in WSN (LECRAD), has been proposed. We use the theory of learning automata [8] to reduce the number of message transmissions, and hence the energy consumption of the network.

A. Motivation

During the period of isolation, a dumb node unnecessarily consumes energy without providing any significant service to the network. Consequently, a temporary communication hole is created, which is dynamic in nature. To maintain topology of the network, we need to re-establish connectivity of the isolated nodes with the network, which becomes more challenging due to temporary node isolation attributed to the dumb behavior. We re-establish the connectivity between a pair of isolated nodes by activating the intermediate sleep nodes or by adjusting the communication range of nodes, when there is no neighbor node within the reduced communication. Exploration of all the possible links to re-establish connectivity between a pair of isolated nodes is not desirable in energy constrained WSNs. Exploration of all the possible links consumes more energy due to the transmission and reception of large number of control messages. Thus, the use of learning automata in the proposed scheme reduces the number of control messages for increasing the energy efficiency of the network.

B. Contribution

In this work, we reconstruct the lost connectivities by activating the intermediate sleep nodes of isolated nodes or by adjusting the communication range of nodes with the cost
of additional energy consumption, when there is no neighbor node within the reduced communication range. We outline the overall contributions of this work as follows.

- Development of a learning automata-based scheme for the reconstruction of lost connectivity in the presence of dumb nodes.
- Reduction in the number of control message exchanges for the reconstruction of lost connectivities, by exploring a minimum subset of intermediate connectivities between isolated nodes, instead of exploring all possible connectivity options.
- Reduction of the energy consumption of the network by transmitting and receiving less number of control messages.

II. RELATED WORK

Connectivity among sensor nodes is a major consideration in multi-hop wireless sensor networks. Many existing works in the literature addressed the issues of connectivity and topology management. Misra and Jain [9] proposed a Policy Controlled Self-Configuration scheme (PCSSN) for topology management and maintenance in Unattended Wireless Sensor Networks. In this scheme, they consider a densely deployed sensor network and use the redundancy property of it. They activate a optimum subset of nodes based on the distance between neighboring nodes, the residual energy, and the neighbor count, the network lifetime, and state of connectivity of the network using Markov Decision Process (MDP). Dini et al. [10] proposed a method using mobile nodes for establishing connectivity among different partitions of a partitioned WSN. The mobile node finds its proper position based on the degree of connectivity with its neighbors. Senel et al. [11] considered a WSN partitioned due to structural damage and proposed a spider-web based approach to reconnect different partitions. They permanently deployed relay nodes to re-establish connectivity among the partitions. Ghosh and Das [12] investigated the sensing coverage and connectivity problem in WSN. They also studied the importance of coverage and connectivity with respect to different applications. However, the works of Misra et al. [9], Dini et al. [10], Senel et al. [11], Ghosh and Das [12] cited above only considered permanent node isolation and network partitioning. They proposed different schemes to reconstruct the topology in such scenario. They did not consider temporary node isolation and network partitioning due to external environmental factors.

Rajagopalan and Varshney [13] addressed the problem of quantifying the connectivity of WSNs in the presence of different channel fading models, and sensor failures. They presented an analytical framework for the computation of node isolation probability and network connectivity under different channel fading conditions and analyzed the connectivity of sensor networks in the presence of unreliable sensors. Ruiz et al. [14] proposed a failure detection scheme in WSN. They proposed a scheme, called MANNA, using a management architecture for self-configuration, self-diagnostic, and self-healing, and some of the self-managing capabilities in WSNs.

In all these works on misbehaviors and faults in WSN, the authors considered that the sensor nodes exhibit misbehavior or faulty behavior due to attacks by malicious external entities, node failure, or malfunctioning. However, these works did not consider temporary behavior, when a sensor node cannot transmit its sensed information to others due to the shrinkage in communication range when adverse environmental effects prevail.

A review of the existing literature reveals that the authors of the existing works considered different types of misbehavior, faults, and connectivity issues in WSN, which are mostly permanent in nature. However, in these works the authors did not consider the situation in which a sensor node can sense but cannot communicate with its neighbor due to the shrinkage in communication range on the onset of sudden adverse environmental effects. After the resumption of favorable environmental conditions, the sensor nodes behave normally. We termed this behavior of sensor nodes as dumb behavior [7], which is characteristically dynamic in nature. Similar to the other types of misbehavior, the dumb behavior also has detrimental effect on the network performance. So, this behavior of sensor nodes can be considered as a kind of misbehavior. Roy et al. [15]–[17] proposed schemes for the detection of nodes exhibiting dumb behavior in a WSN. However, no existing literature specifically address the issue of re-establishment of connectivity of temporarily isolated nodes, arising due to dumb behavior, with the network. Therefore, in this work we propose a scheme to re-establish connectivity of temporarily isolated nodes with the network.

III. PROBLEM DESCRIPTION

Due to the shrinkage in communication range attributed to the onset of adverse environmental effects, the links between the nodes get effected. The shrinkage in communication range occurs due to the reduction of signal strength caused by attenuation and fading for adverse environmental conditions. If signal strength decreases below a threshold, the link between the nodes breaks. If shrinkage in communication range of a sensor node occurs below its nearest active neighbor node, the node gets isolated from the network. In some situations, for non-uniform distribution of adverse environmental conditions, the network may be partitioned into different partitions. Node isolation and network partition is dynamic in nature as adverse environmental effects are temporal in nature. Mathematically, A WSN can be modeled as a graph $G(V,E)$, where $V$ is the set of nodes and $E$ is the set of links between nodes, and $s$ is the sink node, where $s \in V$. For all positive integers $n_1, n_2, \cdots, n_k$, such that $\sum_{i=1}^{k} n_i = n$, there exists a partition of $V(G)$ into $k$ parts $V_1, V_2, \cdots, V_k$ such that $|V_i| = n_i$, and $V_i$ induces a connected sub-graph of $G$ for $1 \leq i \leq k$ and $\exists v \in V_i$ there is no path $sv$. Node isolation and network partition due to shrinkage in communication range are shown in Fig. 1. In this figure, the solid line represents normal connectivity and the dotted line represents broken connectivity.
In such a situation, to maintain the topology of the network, we need to reconstruct the topology by re-establishing connectivity of an isolated node with the sink node. Topology reconstruction becomes more challenging due to the dynamic behavior of node isolation or network partition.

IV. SYSTEM MODEL

To solve the temporary node isolation and network partition problem using learning automata, we model a WSN as \(<\mathcal{N}, \mathcal{L}, \mathcal{D}>\), where \(\mathcal{N} = \{n_1, n_2, n_3, \ldots\}\) is the set of nodes, \(\mathcal{L} = \{l_{ij}\} \subseteq \mathcal{N} \times \mathcal{N}\) is the set of communication links, and \(\mathcal{D} = \{D_i\forall n_i \in \mathcal{N}\}\) is the set of effective distances associated with the set of nodes. Due to the shrinkage in communication range attributed to environmental conditions and node failure, \(\mathcal{N}, \mathcal{L}, \mathcal{D}\) are time variant in nature. At stage \(t\) these parameters are \(\mathcal{N}(t)\), \(\mathcal{L}(t)\), and \(\mathcal{D}(t)\). An isolated node \(U_i\) establishes connectivity with the connecting node \((V, )\), through the intermediate forwarder nodes. The connecting node is the nearest activated neighbor node, with which the isolated node was connected, but it has ceased to be connected at present. The effective distance, \(D_i\), of node, \(n_i\), is calculated as:

\[
D_i = \nu_i + \varepsilon_i
\]

where \(\nu_i\) is the vertical distance of node \(n_i\) to the line joining isolated node and connecting node and \(\varepsilon_i\) is the Euclidean distance of node \(n_i\) to the connecting node. If the position of an isolated node, connecting node, and forwarder node \(n_i\) are \((x_u, y_u)\), \((x_v, y_v)\), and \((x_i, y_i)\), respectively, the effective distance \(D_i\) is:

\[
D_i = \sqrt{(y_v - y_u)^2 + (x_v - x_u)^2} + \sqrt{(y_i - y_u)^2 + (x_i - x_u)^2} - \sqrt{(y_v - y_i)^2 + (x_v - x_i)^2}
\]

A collection of sensor nodes is deployed over a terrain. To reduce energy consumption of the energy constrained WSN, an optimal subset of deployed sensor nodes remain activated at a time and cover the entire region optimally. Rest of the sensor nodes remain in the sleep state. All the sensor nodes are GPS-enabled and have the capability of adjusting their communication range by the cost of additional energy expenditure, when there is no neighbor node within the reduced communication range. There are different energy levels of a sensor node. A sensor node switches from the lower energy level, \(\tau_i\), of node \(n_i\), to the higher one to increase its communication range. Initially, a sensor node remains in its lowest energy level. A sensor node participate in connectivity reconstruction process at stage \(t\) if it satisfies following constraint.

\[
(re_{th} \leq re(t)) \land (rss_{th} \leq rss(t)) \land (\varepsilon_i \leq \varepsilon_i^f) \tag{3}
\]

where \(re(t)\) and \(rss(t)\) are residual energy and received signal strength at stage \(t\) respectively. \(re_{th}\) and \(rss_{th}\) are communication threshold of residual energy and received signal strength. \(\varepsilon_i\) and \(\varepsilon_i^f\) are the euclidean distance of node \(n_i\) and its forwarder node to the connecting node respectively. In the proposed scheme, each sensor node \(n_i\) prepares a variable action-set learning automaton \(A_i\) \[18]\.

V. PROPOSED SOLUTION

In this section, a distributed learning automata-based connectivity re-establishment scheme, LECRAD, is proposed for topology reconstruction in the presence of dumb nodes. An isolated node \(n_i\) initiates the proposed scheme and broadcasts an activation message, ACK, to activate the neighbor sleep nodes. Among the activated neighbor nodes, those which satisfy the constraint given in Equation (3), calculate their effective distance and send as acknowledgment, ACK, to the node \(n_i\). If isolated node \(n_i\) does not get any ACK, it increases the communication range by switching from lower to higher energy level and continue the same process. At stage \(t\) if a node \(n_i\) does not get any ACK in its highest energy level, it is not possible to re-establish the connectivity. After receiving an ACK, node \(n_i\) forms the action-set of learning automaton \(A_i\). Let \(\sigma\) denote the set of actions that can be taken by learning automaton \(A_i\). Each learning automaton \(A_i\) forms the action-set \(\sigma_i\) by reserving an action for every node \(n_j\) replying to the ACK message to node \(n_i\). For each link \(l_{ij} \in L(t)\) incident at node \(n_i\) reserves an action \(\sigma_i^j\). The set of learning automata associated with the set of nodes is defined as \((A(t), \sigma(t))\), where \(A(t) = \{A_i\forall n_i \in \mathcal{N}(t)\}\) is the set of learning automata for all the sensor nodes, and \(\sigma(t) = \{\sigma_i\}^{|A_i|}\) is all the action-sets corresponding to all the automata at stage \(t\). Let choose probability for action \(\sigma_i^j\) at stage \(t\) be \(p_i^j(t)\). Each automaton initializes the action probability vector proportional to the effective distance of its neighboring sensor nodes. The initial choice probability of action \(\sigma_i^j\) at stage \(t\) is:

\[
p_i^j(t) = 1 - \frac{D_j(t)}{\Gamma_i(t)} \tag{4}
\]

where \(D_j(t)\) is the effective distance of neighbor node \(n_j\) at stage \(t\), and \(\Gamma_i(t) = \sum_{\forall l_{ij} \in L(t)} D_j(t)\) is the total effective distance of neighbors of node \(n_i\), which satisfy the constraint at stage \(t\). Due to the dynamic shrinkage in communication range of sensor nodes, the action set (and also the action probability vector) may change with time. Due to the change in communication range at stage \((t + 1)\), node \(n_i\) updates the action set \(\sigma_i(t + 1)\) and the action probability vector \(p_i(t + 1)\) in the similar manner as discussed by Torkestani et al. \[19\] as follows. The action set \(\sigma_i(t + 1)\) is calculated as action set \(\sigma_i(t + 1)\) added with new action \(\sigma_i^j\). The choice probability
of the new action, \( p_i^j (t + 1) \), is:

\[
p_i^j (t + 1) = \frac{1}{\Delta_i (t + 1)}
\]

where \( \Delta_i (t + 1) \) is the number of sensor nodes within the reduced communication range of node \( n_i \) at stage \( (t+1) \) which satisfy the constraint. The choice probability of other actions \( \sigma_i^j (\neq \sigma_i^j) \) is updated as:

\[
p_i^j (t + 1) = \frac{\Delta_i (t + 1) - 1}{\Delta_i (t + 1)} \cdot p_i^j (t)
\]

If at stage \( (t + 1) \), the communication link \( l_{ij} \in \mathcal{L}(t) \) breaks, the action \( \sigma_i^j \) is removed from the action set \( \sigma_i (t+1) \) and the choice probability of the corresponding action is set to zero. Then, the choice probability of the other actions \( \sigma_i^j \) is updated as:

\[
p_i^j (t + 1) = p_i^j + p_i^{j'} \cdot \frac{p_i^{j'}}{1 - p_i^{j'}}
\]

We use the \( L_{R\rightarrow I} \) reinforcement scheme to update the action set of each learning automaton \( A_i \). Therefore, the action probability vector remains unchanged, if the newly added node does not satisfy the constraint defined in Equation (3).

Node \( n_i \) chooses one of its neighbor nodes \( n_j \), based on the highest choice probability, which satisfies the constraint and sends a request message, \( \text{REQ} \). Node \( n_j \) performs the same process as node \( n_i \) and holds the address of node \( n_i \) as downstream node. If node \( n_j \) does not have any neighbor node within its reduced communication range which satisfies the constraint, sends a negative acknowledgment, \( \text{NACK} \), to node \( n_i \). Node \( n_i \) chooses another neighbor node with the next highest choice probability and continues this selection process until node \( n_i \) does not find any neighbor node with at least one neighbor satisfying the constraint. If, in this process, node \( n_j \) cannot choose any of its neighbor nodes, it increases the communication range of neighbor nodes remains in the same sequence to find their neighbor nodes satisfying the constraint. If node \( n_i \) cannot select any of its neighbor nodes after increasing its communication range, it sends a \( \text{NACK} \) to the downstream forwarder node and follows the same process. Node \( n_i \) cannot participate in the connectivity re-establishment process at stage \( t \). The selected neighbor node remains activated for next \( t_{rep} \) time, where \( t_{rep} \) is the estimated time for receiving reply message, \( \text{REP} \), from the connecting node. Gradually the neighbor selection process proceeds through the intermediate nodes. The process stops its execution when the connecting node is reached. After receiving the \( \text{REQ} \) message, the connecting node sends back the \( \text{REP} \) message to the isolated node through the selected forwarder nodes. The forwarder nodes receiving the \( \text{REP} \) message remain activated for next \( t_{rep} \) time, where \( t_{rep} \) is the estimated time for repeating the scheme.

The process of connectivity re-establishment using our proposed scheme is shown in Fig. 2. In this figure, the isolated node \( U \) re-establishes connectivity with the connecting node \( V \). Node \( U \) chooses Node 3 and Node 3 chooses Node 6 as their upstream node and send \( \text{REQ} \) message. Node 6 sends \( \text{REQ} \) message to Node 9 for its lowest effective distance among the neighbor nodes. However, Node 6 cannot choose Node 9 because it has no neighbor node within its reduced communication range as well as it cannot find any neighbor node after increasing its communication range. So, Node 6 chooses Node 8 as its upstream node and sends a \( \text{REQ} \) message. Node 8 finds Node 11 after increasing its communication range, represent by dotted line in the figure. In the similar manner, the process proceeds to the upstream nodes until it reaches the connecting node \( V \). When the connecting node \( V \) receives \( \text{REQ} \) message reply back, \( \text{REP} \), message to the isolated node \( U \) through Nodes 14, 11, 8, 6, 3, the intermediate nodes receiving the \( \text{REP} \) message remain activated for the next \( t_{rep} \) time. Algorithm 1 presents the proposed scheme \( \text{LECRAD} \).

**Algorithm 1 : LECRAD**

**Input:**
- \( ID_i \): id of isolated node
- \( ID_V \): id of connecting node
- \( (x_u, y_u) \): position of isolated node
- \( (x_v, y_v) \): position of connecting node
- \( t_{rep} \): expected reply time from connecting node
- \( t_{rpt} \): repeat time

**while** \( (ID_i \neq ID_V) \) **do**

- Broadcast \( \text{ACT} \) message
- if \( \text{ACK} \) received then
  - prepare \( \sigma_i(t) \) and \( p_i(t) \)
  - choose node \( n_j \) with highest \( p_i^j(t) \) &\& number of neighbor \( \neq 0 \)
  - Send \( \text{REQ} \) packet to \( n_j \)
  - Activate \( n_j \) for next \( t_{rep} \)
- else
  - while \( \text{ACK} \) not received do
    - Shift from low to high energy level
    - if \( \tau_{max} \) then
      - send \( \text{NACK} \) to downstream forwarder node
      - Break
    - end if
  - end while
  - if \( ID_i = ID_V \) then
    - send \( \text{REP} \) message to isolated node
    - if \( \text{REP} \) received then
      - activate for next \( t_{rpt} \)
    - end if
  - end if
- end if

**end while**

Fig. 2: Schematic diagram of connectivity re-establishment using the proposed scheme
VI. SIMULATION RESULTS

A. Simulation Design

In this section, we evaluate the performance by simulating our proposed scheme LECRAD. We deployed 150-350 sensor nodes over a terrain of 500m × 500m randomly. To reduce the energy consumption of the network, an optimal number of deployed sensor nodes remain activated at a time to cover the entire terrain optimally, while the rest of the sensor nodes remain in the sleep state. We consider that the sensor nodes can adjust their communication range at the cost of additional energy. The size of ACT, ACK, NACT, REQ, and REP are considered as 6, 6, 6, 20, and 12 bytes, respectively. The list of simulation parameters has presented in Table I. We evaluated the proposed algorithm LECRAD based on the following performance metrics.

- Percentage of activated nodes: The number of nodes that are required to be activated to maintain topology per 100 nodes in the network.
- Success ratio: The ratio between the number of nodes with successfully re-established connectivity and the number of isolated nodes in the network.
- Message overhead: The total amount of control message exchanged to re-establish connectivity of all the isolated nodes in the network.
- Energy consumption: Amount of energy consumed during the reconstruction of network topology.

We compared the proposed algorithm LECRAD with two recently proposed existing topology management protocols, Learning automata-based Energy-efficient Topology Control (LECT) [19] and Distributed Topology Control Algorithm (A1) [20], with respect to the number of nodes activated and message overhead.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of nodes</td>
<td>150-350</td>
</tr>
<tr>
<td>Simulation area</td>
<td>500 × 500m</td>
</tr>
<tr>
<td>Sensing range</td>
<td>50m</td>
</tr>
<tr>
<td>Initial residual energy (J)</td>
<td>2.03201</td>
</tr>
<tr>
<td>Communication range in normal situation (m)</td>
<td>110m</td>
</tr>
<tr>
<td>Change in communication range</td>
<td>30 - 85m</td>
</tr>
<tr>
<td>Increase of communication range per energy level shifting (m)</td>
<td>20m</td>
</tr>
</tbody>
</table>

B. Discussion of Results

We evaluate and compare the performance of the proposed algorithm LECRAD. In Fig. 3, the percentage of activated nodes versus varying communication range for different numbers of nodes is shown. The plot presents the general trend of gradual decrease in the percentage of activated nodes with the increase in communication range and the decrease in the number of nodes in the network. The possible reason behind this is attributed to the decrease in the number of isolated nodes with the increase in the number of neighbor nodes due to the increase in communication range and decrease in number of nodes in the network.

Success ratio versus varying communication range for different number of nodes is shown in Fig. 4. The plot shows an increasing trend in success ratio with the increase in communication range and number of nodes in the network. The reason behind this trend is that the possibility of re-establishment of connectivity increases as the number of neighbor nodes increases due to the increase in the communication range and number of nodes in the network.

Energy consumption versus varying communication range for different number of nodes is shown in Fig. 5. The plot depicts that the energy consumption decreases with the increase in the communication range and decrease in the number of nodes in the network. Increase in communication range and decrease in the number of nodes decreases the number transmitted and received control messages, which, in turn, decreases the energy consumption.

In Figs. 6, 7, and 8, comparison of percentage of activated nodes, message overhead, and energy consumption are shown, respectively. In all the plots it is observed that in case of LECRAD the percentage of activated nodes, message overhead, and energy consumption decreases, however, in the cases of LETC and A1, this figure increases. The possible reason behind this is that the proposed scheme LECRAD re-establishes connectivity only between the isolated nodes, when the network topology is broken. However, the existing topology management schemes, LETC and A1, reconstruct the topology from scratch, when the network topology is broken. Additionally, with the decrease in the communication range the existing topology management schemes cannot reconstruct the topology of the entire network. However, the proposed scheme LECRAD can reconstruct the topology by increasing the communication range when there is no neighbor node within the reduced communication range.

VII. CONCLUSION

In this work, we consider the dynamic node isolation due to the presence of dumb nodes in stationary WSN. In such a situation, the sensor nodes get isolated and the network become partitioned temporarily. To maintain the topology in such a dynamic situation, we propose a connectivity reconstruction scheme, named LECRAD. The proposed scheme uses the concept of learning automata to re-establish connectivity of an isolated node with the network by activating the intermediate sleep nodes or by adjusting the communication range, when there is no neighbor node within the reduced communication range. Here, instead of exploring all the intermediate paths, we explore some of the selected paths. This reduces the number of transmitted and received control messages, which, in turn, reduces the message overhead in the network. Simulation results show that the proposed scheme exhibits better performance in terms of the number of nodes activated, message overhead, and energy consumption than that of the recently proposed existing topology management schemes.
REFERENCES


