Sleep-Route: Assured Sensing with Aggressively Sleeping Nodes

Chayan Sarkar, Vijay S. Rao, R. Venkatesha Prasad, Koen Langendoen
Delft University of Technology
{c.sarkar, v.rao, r.r.venkateshaprasad, k.g.langendoen}@tudelft.nl

***Abstract*—In data gathering wireless sensor network applications, data correlation among the sensor nodes have been utilized to extend network lifetimes. It has been shown that the data correlation also exists between nodes that are far away, contrary to the assumption that correlation decreases as a function of distance. Therefore, it is possible to group the nodes based on the correlation among their data regardless of their location. Given that data from one *active* node per group is sufficient to reconstruct the sensed data for the remaining sensor nodes, most of the nodes can be kept in low-power sleep mode. However, only few active nodes will usually create a disconnected network, and hence failing the purpose of the deployment. In this paper we formalize this problem, referred to as *Sleep-route*, of selecting the minimum number of connected active nodes that are sufficient to predict the sensed data for remaining sleeping nodes with high accuracy. We prove that the problem is NP-hard. Thus, we develop a greedy algorithm, *Sleep-route heuristic* that provides near-optimal solutions. Using Contiki-based simulations, we show that our scheme can extend network lifetime up to 42% as compared to the state-of-the-art solutions.**

I. Introduction

Wireless sensor networks (WSNs) have enabled monitoring applications such as structural health monitoring, habitat monitoring, precision agriculture, etc. In these WSN applications, all the sensor nodes periodically report their data to the sink node. Of the many energy efficient data collection techniques proposed in the literature, a popular technique has been to utilize correlation among the sensor nodes. It is a fact that the data correlation is not necessarily a function of distance. To demonstrate our claim, we consider the temperature data collected from 54 sensor nodes in Intel Berkeley Lab deployment [12]. Fig. 1 shows the absolute correlation coefficient between Node 1 and the other nodes. According to general presumption, the correlation among nodes should decline with increasing distance. However, from Fig. 1, it is clear that some closer-by nodes have poorer correlation whereas, some far away nodes have higher correlation.

This property of correlation can be exploited to achieve higher energy efficiency and hence, higher network lifetime. All nodes having high data correlation in the network can be grouped to form a *constellation*. Only one representative, called *active* node from a constellation is required to report data since other nodes' values can be estimated statistically. Thus, all non-active nodes are switched to low-power sleep mode (*dormant*). A large WSN deployment may have many such constellations. To this end, we make use of the *Virtual Sensing Framework* (VSF), as described in [14] for (a) computing correlations among nodes based on their sensed data, (b) creating pairs of highly correlated nodes, and (c) accurate prediction mechanism for dormant nodes. Another noteworthy feature of VSF is that nodes need not even sense the same physical parameter.

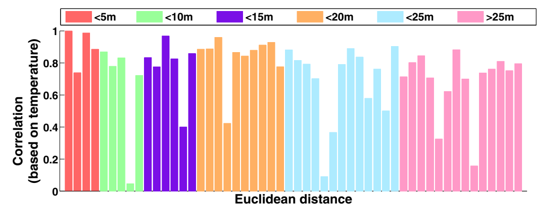


Fig. 1: Data correlation among sensor nodes is independent of Euclidean distances; data from deployment in [12].

In a large WSN with VSF in place, many nodes will be dormant with a few active nodes. Though more dormant nodes decreases energy consumption of the whole network, it creates a situation where isolated active nodes are surrounded by many dormant nodes, leading to a disconnected network. Thus, it is required to solve the problem of finding an optimal (minimum) subset of active nodes such that (i) data from active nodes can reach the sink; (ii) data of dormant nodes can be predicted within an error bound; while (iii) energy saving is maximum. To further extend the lifetime, a careful choice of active nodes needs to be made, i.e., alternating the roles of active and dormant nodes over time based on their remaining energy. We refer to this problem as *Sleep-route*. In this article we consider network lifetime as the timespan between deployment and the instant when all nodes in a constellation die due to energy drain.

Contributions: The key distinction of our study with respect to the vast and prolific work on data collection in WSNs is that we group the nodes solely based on their sensed data. This characteristic presents a general case where nearby sensor nodes, far away sensor nodes and even sensor nodes monitoring different physical parameters (e.g., temperature and light) can be highly correlated. Our main contributions are:

Theoretical foundation: First, we provide a formal definition of the Sleep-route problem, where it is defined as an optimization problem with the goal of increasing the lifetime of a WSN deployment. Further, we prove that the problem is NP-hard

TABLE I: Comparing Sleep-route with energy-efficient data collection techniques.

Technique	Characteristics	Sleep-route
Clustering [2], [9], [17]	Groups or clusters of collocated nodes select one among them to act as cluster-heads (CHs); nodes send their data to the local CH and the CHs collaboratively send their data to the sink. Thus there is less traffic (and less energy expenditure for data transmission). However, all the nodes are active or at most duty-cycling to conserve energy.	Grouping is not based on collocation but on data. Only one node in a group (or constellation) needs to be active. Therefore, higher savings of energy can be achieved.
In-network data aggregation [4], [8]	The idea is to combine data from multiple sources and reduce data transmissions within the network. Nodes are required to sense and send data; compressed sensing solutions are applicable when node density is very high.	Sleep-route creates a tree-based topology and, hence, data of the children and parent nodes are piggy-backed to reduce data transmissions. While in-network data aggregation techniques do not deliver the data values measured by individual sensors at the sink, in Sleep-route we ensure that data (within an error bound) for all sensors is at the sink.
Reduced transmission rate [7], [13], [15]	A data estimation model is calculated; if the estimated data does not deviate much from the sensed data, transmission is discarded; transmission occurs only when large deviation is detected. Even though a large amount of energy is saved by avoiding data transmission whenever possible, all the nodes sense the field.	Sleep-route creates data estimation models to create constellations. Only one node requires being active in a constellation, and this node does not transmit data if the estimated data does not deviate much from the sensed data. Hence, we can expect significant savings in Sleep-route.
On-Off scheduling [1], [16], [18]	When multiple sensor nodes can monitor a common sub-area or a point, data from only one node is sufficient; rest of the nodes are kept in sleep mode. This is applicable only when sensor nodes within close proximity produce correlated data. Moreover, spatial correlation is assumed to be known (or follows a given pattern).	One node in a constellation is enough to monitor several points of interests over the whole network. Furthermore, we do not assume an *a priori* correlations.

(see Section III).

A practical algorithm and implementation: We implement a simple heuristic, *Sleep-route heuristic* (SRH), that solves Sleep-Route problem in polynomial time. The implementation is done in Contiki and it is independent of the underlying MAC and routing protocols. Periodically, the heuristic requires information about a node's residual energy and immediate parent nodes. We avoid additional control messages by piggybacking this information with the data packet and incur very low additional energy usage in the sensing nodes (see Section IV).

Evaluation: To evaluate performance of the heuristic, we have used the Contiki-based Cooja simulator and created WSN deployments of various sizes. Our results show that the energy saving and lifetime extension of the deployment depends on the correlations among the sensor nodes. Highly correlated deployment can reduce the number of constellations and increase the network lifetime up to 42% as compared to the state-of-the-art (Section V).

II. Related Work

A large body of energy-efficient data gathering techniques exists in the literature. As data transmission consumes a large amount of energy, a common approach is to reduce the amount of traffic in the network. A brief summary of the various classes of techniques is provided in Table I. Apart from these techniques, significant work has also been done on constructing sensed data using a small number of active sensors. Works such as [3], [5] assume *a priori* high correlations among collocated sensors and leverage this correlation to reconstruct data from a subset of active nodes.

In this paper, the saving in energy is due to a large subset of dormant nodes with a small subset of nodes actively sensing the field of interest. For the remaining non-active nodes, the sensed values are predicted by exploiting data correlation among the nodes. Our method differs from the existing methods [11], [16] of selecting a small subset of active sensor as we does not assume geographical collocation of the correlated sensors and correlation among the nodes is not known *a priori*.

III. Sleep-Route Problem Formulation

Since the Sleep-route problem has dependency on Virtual Sensing Framework (VSF), we provide a very brief overview of VSF. If two sensors a and d are reporting highly correlated data, the data of d can be predicted (with high accuracy) using the data from a while keeping node d in dormant state. To predict the sensor data, virtual sensor (VS) is created at the sink for each physical sensor (PS). VSF divides the data collection in three phases [14]. Initially, during the training period, all the sensor nodes report their data. Using these data, correlated node pairs and prediction models for potential dormant nodes are found. Then in operational period, a node from each correlated node pair remains dormant and its data is being predicted. Next, in the revalidation period, new set of active nodes are found. Operational and revalidation period continues in tandem.

We can easily extend VSF's functionality of finding correlated node pairs to find correlated groups of nodes or *constellation*. A constellation of nodes formed by VSF has a fundamental difference with the traditional clustering techniques. Whilst the nodes in a constellation need not be geographically collocated, they should be mutually and highly correlated. One representative node generating data from a constellation is enough to predict the sensed data of other nodes in that constellation. Therefore, in order to reduce energy consumption of the WSN, VSF selects m nodes to be active, one from each of the m constellations, while keeping the other nodes in a dormant state. VSF assumes that all the nodes are directly reachable from the sink, which is a limiting factor in a real-world deployment. Thus the nodes selected as active may actually be isolated i.e., they are not in the communication range of the sink and a multi-hop path to the sink cannot be built.

The goal is to reduce the overall energy consumption of WSNs in order to increase the network lifetime while ensuring data from every sensor reaches the sink. Therefore, it is required to solve the problem of Sleep-route – to find a set of active nodes that requires minimum energy to report data to the sink satisfying the following conditions: (a) they form a connected graph, specifically, all of them can reach the sink node; (b) every constellation has at least one representative active node so that the data for all the dormant nodes within that constellation can be predicted with a tolerable error bound. In order to create a connected graph, more than m active nodes may be required. Moreover, there can be a set of active nodes containing more than m active nodes but consumes less energy than a set with fewer number of active nodes. Hence the problem is a combinatorial optimization problem.

We represent the network as a node- and link-weighted, undirected graph $G = (S, L)$, where S is the set of n+1 nodes in the network and L is the set of links in the network. A link between a node i and a node j exists only if they are within the transmission range of each other. The cost of this link, denoted by $e_l(i, j)$ or $e_l(j, i)$, is equal to the energy consumption for transmitting one packet from one node to the other. Each node is also associated with a non-zero cost (node-weight), where $e_a(i)$ and $e_d(i)$ are the amount of energy consumed in active state and dormant state respectively. We now formally define the problem as follows.

Definition 1. *Sleep-route: Given a node- and link-weighted undirected graph $G = (S, L)$ and a collection of subsets $\{S_1, ..., S_m\}$ such that $\bigcup_{k=1:m} S_k = S$, find the minimum cost subgraph $G' = (A, L')$, where $A \subseteq S$ and $L' \subseteq L$. Here cost of G' is defined as, $\sum_{i \in A} e_a(i) + \sum_{(i,j) \in L'} e_l(i, j)$. The minimum cost subgraph has to satisfy the following conditions – (i) A includes at least one node from each of the given subset, i.e., $S_k \bigcap A \neq \phi, \quad \forall k$, and (ii) G' is a connected graph.*

In the above definition, a subset $(S_i : i \in \{1, \dots, m\})$ represents a constellation, A is the subset of active nodes, and L' is the set of links connecting the nodes in A. Before proposing a solution, we discuss the time complexity of the problem.

Theorem 1. *Sleep-route (decision) problem is NP-complete. The optimization version of Sleep-route is NP-hard.*

Proof. It is easy to show that Sleep-route $\in$ NP, since a non-deterministic algorithm needs only to find a subgraph A and then verify in polynomial time that (i) it includes at least one node from each constellation (complexity is O(n)), and (ii) it is connected (complexity of DFS is O(n)). Therefore, Sleep-route (decision) problem is NP. The problem can be reduced to the Group Steiner Tree (GST) problem in polynomial time, which is proved to be NP-hard [6].

Definition 2. *Group Steiner Tree (GST) problem: Given a graph $G = (S, L)$ with non-zero link cost, and a collection $S_1, S_2, ..., S_m$ of vertex sets called groups. Find a minimum-cost connected subgraph of G that contains at least one vertex from each group.*

We map the WSN in Sleep-route into a graph in GST. The constellations in Sleep-route are the groups of vertices in GST. The edge weights are calculated based on the energy consumed to deliver one packet through that link (constant time), thus, are the same for both Sleep-route and GST. Similarly, the node weights are calculated based on the energy consumed by a node for being active and sensing (constant time). The residual energy can be considered as same across the nodes. Thus, the transformed graph becomes a graph in Sleep-route problem. As a result, a "yes-instance" of Sleep-route, will always reply "yes" for GST. Similarly, Sleep-route will never reply a "yes" to a "no-instance" in GST. Therefore, the decision version of Sleep-route is NP-complete as this is the case for GST. By default, the optimization problem then becomes a NP-hard problem [10]. We have omitted the rigorous proof of the mapping due to paucity of space. □

IV. SRH: A HEURISTIC ALGORITHM FOR SLEEP-ROUTE

To evolve a polynomial time solution to the Sleep-route problem we propose a heuristic, called *Sleep-route heuristic* (SRH). The heuristic must pick one node to be active from each constellation, so that the data of the other nodes in the constellation can be predicted. However, as explained earlier, this selection does not guarantee the existence of a path from each active node to the sink. To ensure connectivity, additional sensor nodes may need to be selected such that the active nodes can reach the sink. Even though this ensures a connected sub-graph, an impetuous active node selection may cause redundant active nodes. As a result, overall energy consumption is much higher than the optimal solution. Moreover, an incautious active node selection may prefer some of the nodes to be active more often than the rest. These preferred nodes will run out of energy faster than the rest. This might cause a disconnected network and reduces the lifetime of the network. Therefore, a careful active node selection should not only consider minimal energy consumption for the active set of nodes, but also balance the energy expenditure across the nodes.

We separate the active node selection from the route finding process. To find the shortest path from each sensor node to the sink node, we depend on the underlying routing protocol. The shortest path tree helps to derive the parent-child relationships for the nodes in the network. Now, if we ensure that the parent node of every active node is also selected as active, the connectivity can be ensured. Note that the sink node is always selected to be active.

The active node selection process, described in Algorithm 1, partitions the set of sensor nodes (S) into two subsets - active (A) and dormant (D). First, the algorithm selects a node that has the maximum residual energy among the nodes in S and add it to the active set. To ensure connectivity, its parent node (also the parent's parent) needs to be assigned as active, if it is not already assigned to the active set. Next, all the sensor nodes that belong to the same constellation with the recently selected active nodes are removed from S and added to the dormant set except the ones who are already assigned to the active set. The node selection process continues, until S becomes empty, i.e., all the nodes are assigned to either of the sets.

Algorithm 1 *SRH:* Active node selection process.

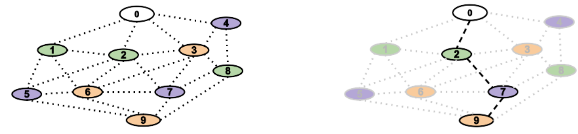
Input: Graph $G = (S, L)$, and constellation matrix (C).
Output: Two sets of nodes - Active (A) and Dormant (D).

$A \leftarrow \phi$; $\quad D \leftarrow \phi$;
while notEmpty(S) **do**
    $s \leftarrow \arg\max_{s \in S}\{e_{res}(s)\}$;
    $markAsActive(s)$;
end while
procedure $markAsActive(s)$
    $S \leftarrow S - s$; $\quad A \leftarrow A + s$;
    $s' \leftarrow parent(s)$;
    if s' is in S **then**
        $markAsActive(s')$;
    end if
    for all $\hat{s} = 1 : n$ **do**
        if $c(s, \hat{s}) = 1$ AND $\hat{s} \in S$ **then**
            $S \leftarrow S - \hat{s}$; $\quad D \leftarrow D + \hat{s}$;
        end if
    end for
end procedure

Maximum residual energy based criteria also balances energy expenditure amongst the nodes. If a node is assigned active, its residual energy will be lesser in the next round of selection as compared to the dormant nodes. As a result, a new set of nodes will be assigned as active. To check whether the maximum residual energy based active node selection is a fair choice, we use two other selection criteria – (i) minimal residual energy based active node selection, and (ii) random selection of active node. We refer these two algorithms as SRH-2 and SRH-3 respectively.

We have implemented SRH in Contiki OS. In general, the implementation is independent of underlying MAC and routing protocols. We use the ContikiMAC for radio duty cycling, and ContikiRPL as a routing protocol. As SRH runs along with VSF, sink gathers data during the training and revalidation periods. Along with the sensed data, they piggyback the residual energy and parent information in order to avoid the additional burden on the network in terms of control messages. Before the start of every operational period, the sink node assigns a new set of nodes as active using Algorithm 1.

The heuristic SRH is implemented in the sink node, which has approximately 150 lines of C code. As the sink node assumed to have large memory and energy, such an implementation does not pose any challenge. On the other hand, approximately 20 lines of code are added to the sensing node on top of the routing setup. A relatively smaller code footprint for the sensing nodes makes it easy to deploy on any type of sensor hardware.



(a) (b)

Fig. 2: (a) A small WSN with 3 constellations; (b) a possible active set of nodes selected by our heuristic.

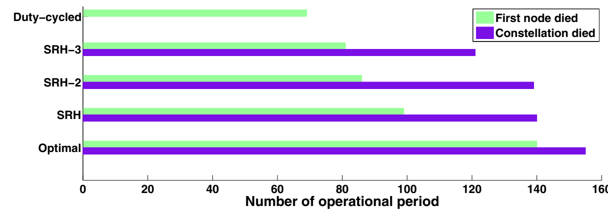


Fig. 3: Lifetime of the WSN in Fig. 2a for various active node selection schemes.

V. Evaluation

We have evaluated SRH using Cooja, a Contiki based simulator. We have considered Tmote sky sensor node. Contiki provides the '*energest*' module that can help estimate energy consumption. In our evaluation, we measured the energy consumption by all sensor nodes during the operational and revalidation periods, where these periods are fixed to 20 and 5 sensing intervals respectively. The sampling interval is set to 30 s. We compared our heuristic algorithm with the optimal solution (using exhaustive search) and other algorithms on a small network. This is to find the difference between the optimal solution and our heuristic. We then create a larger WSN deployment in Cooja to compare with different state-of-the-art solutions.

In the small WSN, there are three constellations with three sensor nodes in each of them as shown in Fig. 2. First, the optimal set of active nodes is decided using an exhaustive search method before every operational period. Similarly, the heuristic algorithms – SRH, SRH-2 and SRH-3 are used to select the active set of nodes based on - maximum residual energy, minimum residual energy, and random selection respectively. Fig. 3 shows the lifetime of the network for different algorithms. Out of the three heuristics, maximum residual energy based solution is preferable as it provides a better lifetime. It also provides a near optimal solution with only 9.65% lesser lifetime compared to the optimal solution.

We can also see the effect of alternating the active and dormant state in the nodes in Fig. 3 by looking at the difference between the death of the first node and the death of the first constellation, particularly in this scenario since there are equal number of nodes in all constellations. Lower the time difference more balanced is the residual energy. The selection criterion based on maximal residual energy (SRH) tries to

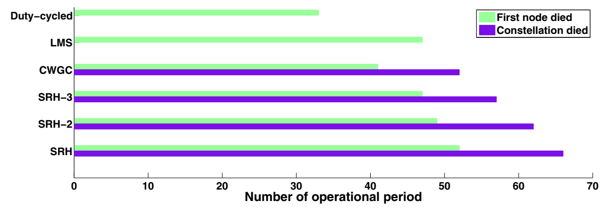


Fig. 4: Lifetime of the WSN in [12] for various active node selection schemes.

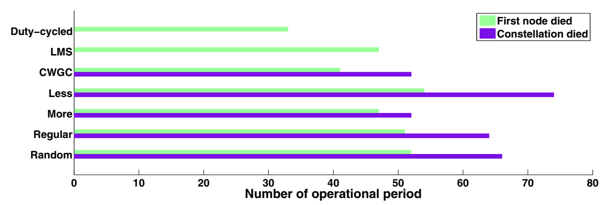


Fig. 5: Lifetime of a WSN with varying number of constellation based on correlation varying over time.

impose this behavior on the selection of nodes, and hence is better than other heuristics, i.e., SRH-2 and SRH-3.

Further, we recreated a simulated WSN of 54 nodes as described in [12]. We found 10 constellations and the geographical location of members of a constellation are random. We compare our solution with two energy-efficient data gathering techniques in the literature that closely match with our scheme, though the underlying principles are different (see Fig. 4). The lifetime improvement based on SRH as compared to LMS-based solution [13] can vary between 11%–57%. On the other hand, our method can improve lifetime up to 42% as compared to CWGC method [18].

As the number of active nodes selected by SRH depends on the number of constellations, we considered a varying number of constellations – 5, 10 and 27 (Less, More and Random respectively in Fig. 5). Additionally, we consider a case where members of a constellation reside within a close proximity (Regular in Fig. 5). From the figure, it is clear that when there is less number of constellations, the number of active nodes decreases. As a result, the overall lifetime of the network increases. Further, note that the lifetime of the network does not vary too much due to correlation pattern among the nodes (Random and Regular), but depends on the highly correlated nodes.

VI. Conclusions

VSF provides correlated sensor node constellations where nodes from a particular constellation are highly correlated irrespective of their geographical location. In this work, we find minimal number of active sensor nodes while ensuring that: (i) there is at least one active node from each constellation; and (ii) all the selected active sensor nodes can reach the sink; and (iii) the network lifetime is maximized. As this problem is shown to be NP-hard, we proposed a heuristic algorithm, Sleep-route heuristic (SRH). Our solution is agnostic to the underlying MAC or routing protocol and it also works in heterogeneous sensor networks as long as there is data correlation among the sensor node. More number of correlated sensor nodes lead to a few number of constellations resulting in large amount of energy saving. However, if the sensor nodes do not produce correlated data, there is little scope for energy saving without compromising the data accuracy. Some important conclusions are: (i) SRH is only 10% higher than the theoretical lowest bound; (ii) SRH can improve the lifetime by 42%; and (iii) SRH indeed adapts to change in environment. We plan to extend this work by jointly considering VSF and Sleep-route by targeting accuracy of data and optimal active node selection when correlation changes.

Acknowledgment

This work is supported by an EU FP7 project, called iCore (http://www.iot-icore.eu/), contract number: 287708.

References

[1] M. Cardei, M. T. Thai, Y. Li, and W. Wu. Energy-efficient target coverage in wireless sensor networks. In *IEEE INFOCOM 2005*, volume 3, pages 1976–1984. IEEE, 2005.

[2] C.-T. Cheng, C. K. Tse, and F. C. Lau. A clustering algorithm for wireless sensor networks based on social insect colonies. *Sensors Journal, IEEE*, 11(3):711–721, 2011.

[3] R. Cristescu and M. Vetterli. On the optimal density for real-time data gathering of spatio-temporal processes in sensor networks. In *IPSN'05*, page 21. IEEE, 2005.

[4] S. Croce, F. Marcelloni, and M. Vecchio. Reducing power consumption in wireless sensor networks using a novel approach to data aggregation. *The Computer Journal*, 51(2):227–239, 2008.

[5] H. Gupta, V. Navda, S. Das, and V. Chowdhary. Efficient gathering of correlated data in sensor networks. *ACM Trans. Sen. Netw.*, 4(1):4:1–4:31, Feb. 2008.

[6] E. Ihler, G. Reich, and P. Widmayer. Class steiner trees and vlsi-design. *Discrete Applied Mathematics*, 90(1):173–194, 1999.

[7] H. Jiang, S. Jin, and C. Wang. Prediction or not? an energy-efficient framework for clustering-based data collection in wireless sensor networks. *Parallel and Distributed Systems, IEEE Trans. on*, 22(6):1064–1071, 2011.

[8] C. Joo and N. B. Shroff. On the delay performance of in-network aggregation in lossy wireless sensor networks. *IEEE/ACM Transactions on Networking*, 22(2):662–673, 2014.

[9] S.-M. Jung, Y.-J. Han, and T.-M. Chung. The concentric clustering scheme for efficient energy consumption in the pegasis. In *9th ICACT*, pages 260–265. IEEE, 2007.

[10] R. M. Karp. *Reducibility among combinatorial problems*. Springer, 1972.

[11] Y. Li and L. E. Parker. Nearest neighbor imputation using spatialtemporal correlations in wireless sensor networks. *Information Fusion*, 15:64–79, 2014.

[12] S. Madden. Intel berkeley research lab data. http://db.csail.mit.edu/labdata/labdata.html, 2004. [accessed 30-Nov-2013].

[13] S. Santini and K. Romer. An adaptive strategy for quality-based data reduction in wireless sensor networks. In *INSS 2006*, pages 29–36, 2006.

[14] C. Sarkar, V. S. Rao, and R. V. Prasad. No-sense: Sense with dormant sensors. In *2014 IEEE NCC*, Feb 2014.

[15] R. Tan, G. Xing, Z. Yuan, X. Liu, and J. Yao. System-level calibration for data fusion in wireless sensor networks. *ACM TOSN*, 9(3), Article 28, 2013.

[16] B. Wang. Coverage problems in sensor networks: A survey. *ACM Computing Surveys (CSUR)*, 43(4):32, 2011.

[17] O. Younis and S. Fahmy. Heed: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *Mob. Comp., IEEE Trans. on*, 3(4):366–379, 2004.

[18] Q. Zhao and M. Gurusamy. Lifetime maximization for connected target coverage in wireless sensor networks. *IEEE/ACM Transactions on Networking (TON)*, 16(6):1378–1391, 2008.