

# Strategies and Techniques for Node Placement in Wireless Sensor Networks: A Survey

Mohamed Younis

Dept. of Computer Sc. & Elec. Eng.  
Univ. of Maryland Baltimore County  
Baltimore, MD 21250  
[younis@cs.umbc.edu](mailto:younis@cs.umbc.edu)

Kemal Akkaya

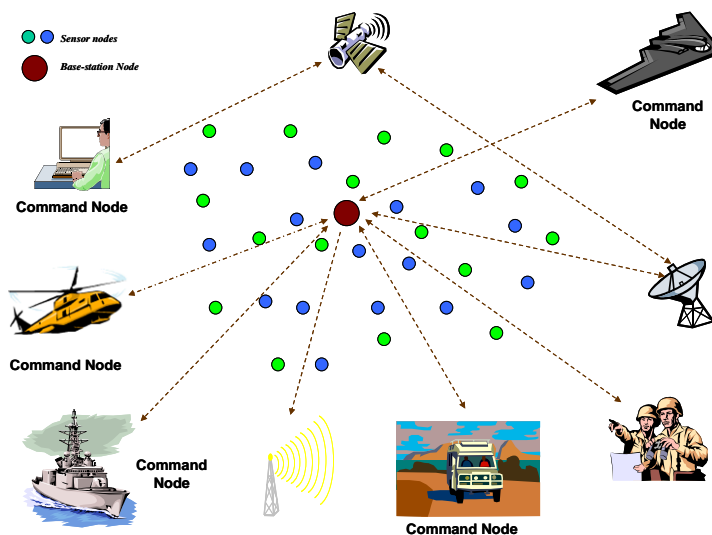
Department of Computer Science  
Southern Illinois University  
Carbondale, IL 62901  
[kemal@cs.siu.edu](mailto:kemal@cs.siu.edu)

## Abstract

The major challenge in designing wireless sensor networks (WSNs) is the support of the functional, such as data latency, and the non-functional, such as data integrity, requirements while coping with the computation, energy and communication constraints. Careful node placement can be a very effective optimization means for achieving the desired design goals. In this paper, we report on the current state of the research on optimized node placement in WSNs. We highlight the issues, identify the various objectives and enumerate the different models and formulations. We categorize the placement strategies into static and dynamic depending on whether the optimization is performed at the time of deployment or while the network is operational, respectively. We further classify the published techniques based on the role that the node plays in the network and the primary performance objective considered. The paper also highlights open problems in this area of research.

## 1. Introduction

Recent years have witnessed an increased interest in the use of wireless sensor networks (WSNs) in numerous applications such as forest monitoring, disaster management, space exploration, factory automation, secure installation, border protection, and battlefield surveillance [1][2]. In these applications, miniaturized sensor nodes are deployed to operate autonomously in unattended environments. In addition to the ability to probe its surroundings, each sensor has an onboard radio to be used for sending the collected data to a base-station either directly or over a multi-hop path. Figure 1 depicts a typical sensor network architecture. For many setups, it is envisioned that WSNs will consist of hundreds of nodes that operate on small batteries. A sensor stops working when it runs out of energy and thus a WSN may be structurally damaged if many sensors exhaust their onboard energy supply. Therefore, WSNs should be carefully managed in order to meet applications' requirements while conserving energy.



**Figure 1:** A sensor network for a combat fielded surveillance application. The base-station is deployed in the vicinity of the sensors to interface the network to remote command centers.

The bulk of the research on WSNs has focused on the effective support of the functional, such as data latency, and the non-functional, such as data integrity, requirements while coping with the resource constraints and on the conservation of available energy in order to prolong the life of the network. Contemporary design schemes for WSNs pursue optimization at the various layers of the communication protocol stack. Popular optimization techniques at the network layer include multi-hop route setup, in-network data aggregation and hierarchical network topology [3]. In the medium access control layer, collision avoidance, output power control, and minimizing idle listening time of radio receivers are a sample of the proposed schemes [1][4]. At the application layer, examples include adaptive activation of nodes, lightweight data authentication and encryption, load balancing and query optimization [5][6].

One of the design optimization strategies is to deterministically place the sensor nodes in order to meet the desired performance goals. In such case, the coverage of the monitored region can be ensured through careful planning of node densities and fields of view and thus the network topology can be established at setup time. However, in many WSNs applications sensors deployment is random and little control can be exerted in order to ensure coverage and yield uniform node density while achieving strongly connected network topology. Therefore, controlled placement is often pursued for only a selected subset of the employed nodes with the goal of structuring the network topology in a way that achieves the desired application requirements. In addition to coverage, the nodes' positions affect numerous network performance metrics such as energy consumption, delay and throughput. For example, large distances between nodes weaken the communication links, lower the throughput and increase energy consumption.

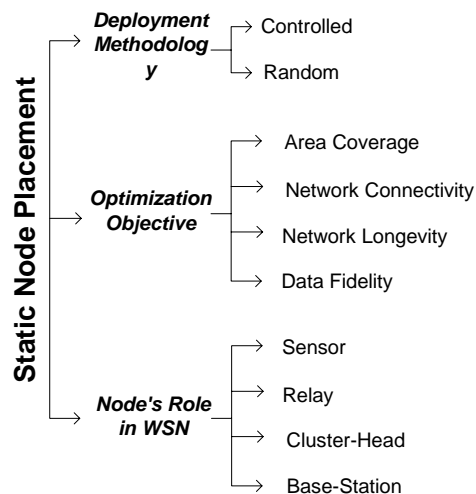
Optimal node placement is a very challenging problem that has been proven to be NP-Hard for most of the formulations of sensor deployment [7][8][9]. To tackle such complexity, several heuristics have been proposed to find sub-optimal solutions [7][10][11][12]. However, the context of these optimization strategies is mainly static in the sense that assessing the quality of candidate positions is based on a structural quality metric such as distance, network connectivity and/or basing the analysis on a fixed topology. Therefore, we classify them as static approaches. On the other hand, some schemes have advocated dynamic adjustment of nodes' location since the optimality of the initial positions may become void during the operation of the network depending on the network state and various external factors [13][14][15]. For example, traffic patterns can change based on the monitored events, or the load may not be balanced among the nodes, causing bottlenecks. Also, application-level interest can vary over time and the available network resources may change as new nodes join the network, or as existing nodes run out of energy.

In this paper we opt to categorize the various strategies for positioning nodes in WSNs. We contrast a number of published approaches highlighting their strengths and limitations. We analyze the issues, identify the various objectives and enumerate the different models and formulations. We categorize the placement strategies into static and dynamic depending on whether the optimization is performed at the time of deployment or while the network is operational, respectively. We further classify the published techniques based on the role that the node plays in the network and the primary performance objective considered. Our aim is to help application designers identify alternative solutions and select appropriate strategies. The paper also outlines open research problems.

The paper is organized as follows. The next section is dedicated to static strategies for node positioning. The different techniques are classified according to the deployment scheme, the primary optimization metrics and the role that the nodes play. In Section 3 we turn our attention to dynamic positioning schemes. We highlight the technical issues and describe published techniques which exploit node repositioning to enhance network performance and operation. Section 4 discusses open research problems; highlighting the challenges of coordinated repositioning of multiple nodes and node placement in three-dimensional application setups and describes a few attempts to tackle these challenges. Finally, Section 5 concludes the paper.

## 2. Static Positioning of Nodes

As mentioned before, the position of nodes have a dramatic impact on the effectiveness of the WSN and the efficiency of its operation. Node placement schemes prior to network startup usually base their choice of the particular nodes' positions on metrics that are independent of the network state or assume a fixed network operation pattern that stays unchanged throughout the lifetime of the network. Examples of such static metrics are area coverage and inter-node distance, among others. Static network operation models often assume periodic data collection over preset routes. In this section we discuss contemporary node placement strategies and techniques in the literature. We classify them according to the deployment methodology, the optimization objective of the placement and roles of the nodes. Figure 2 summarizes the different categories of node placement strategies.



**Figure 2:** Different Classifications of static strategies for node placement in WSN.

By deployment methodologies, as we elaborate in the next subsection, we mean how nodes are distributed and handled in the placement process and the implication of the sensor field of view, i.e., two or three dimensions. Section 2.2 discusses the contemporary objectives that node placement is usually geared to optimize and compares the different approaches in the literature. Section 2.3 looks at how the placement strategies vary based on the role that a node plays in the network. We note four types of nodes: sensors (which monitor their surroundings and are considered data sources), relays, cluster-heads and base-stations. Since the placement of sensors is discussed in detail in Section 2.2, the focus of Section 2.3 is on the other three types. Given the similarity between placement techniques for cluster-heads and base-stations, we cover them under the category of what we call data collectors. A comparative summary of all techniques discussed will be provided in a table at the end of the section.

### 2.1. Deployment Methodology

Sensors can generally be placed in an area of interest either deterministically or randomly. The choice of the deployment scheme depends highly on the type of sensors, application and the environment that the sensors will operate in. Controlled node deployment is viable and often necessary when sensors are expensive or when their operation is significantly affected by their position. Such scenarios include populating an area with highly precise seismic nodes, underwater WSN applications, and placing imaging and video sensors. On the other hand, in some applications random distribution of nodes is the only feasible option. This is particularly

true for harsh environments such as a battle field or a disaster region. Depending on the node distribution and the level of redundancy, random node deployment can achieve the required performance goals.

### 2.1.1 Controlled Node Deployment

Controlled deployment is usually pursued for indoor applications of WSNs. Examples of indoor networks include the Active Sensor Network (ASN) project at the University of Sydney in Australia [16], the Multiple Sensor Indoor Surveillance (MSIS) project at Accenture Technology Labs, in Chicago [17] and the Sensor Network Technology projects at Intel [18]. The ASN and MSIS projects are geared toward serving surveillance applications such as secure installations and enterprise asset management. At Intel, the main focus is on applications in manufacturing plants and engineering facilities, e.g., preventative equipment maintenance (Figure 3). Hand-placed sensors are also used to monitor the health of large buildings in order to detect corrosions and overstressed beams that can endanger the structure's integrity [19][20]. Another notable effort is the Sandia Water Initiative at Sandia National Lab which addresses the problem of placing sensors in order to detect and identify the source of contamination in air or water supplies [21][22].

Deterministic placement is also very popular in applications of range-finders, underwater acoustics, imaging and video sensors. In general, these sensors are involved in three-dimensional (3-D) application scenarios, which is much more difficult to analyze compared to two-dimensional deployment regions. Poduri et al. have investigated the applicability of contemporary coverage analysis and placement strategies pursued for 2-D space to 3-D setups [9]. They have concluded that many of the popular formulations, such as art-gallery and sphere-packing problems, which are optimally solvable in 2-D, become NP-Hard in 3-D. Most placement approaches for these types of sensors strive to enhance the quality of visual images and/or accuracy of the assessment of the detected objects. For underwater applications, acoustic signals are often used as data carriers, and the placement of nodes must also ensure that communicating nodes are in each other's the line-of-sight [23].

As we elaborate in Section 4, optimized deployment of large scale WSNs in 3-D applications is an emerging area of research that has just started to receive attention. Most of the publications on 3-D WSNs have considered very small networks [17], restricted the scope of the placement to a 2-D plane [24], and/or pursued the fulfillment of simple design goals [25]. For example, Gonzalez-Banos and Latombe have studied the problem of finding the minimum number of range-finders needed to estimate the proximity of a target, and their location in order to cover an area [24]. Unlike other sensors, such as acoustic or temperature, etc., the authors have to account for the restricted capabilities of the range-finders, which provide lower and upper bounds only, and for the difficulty of detecting objects at grazing angles. The problem is formulated as an Art-



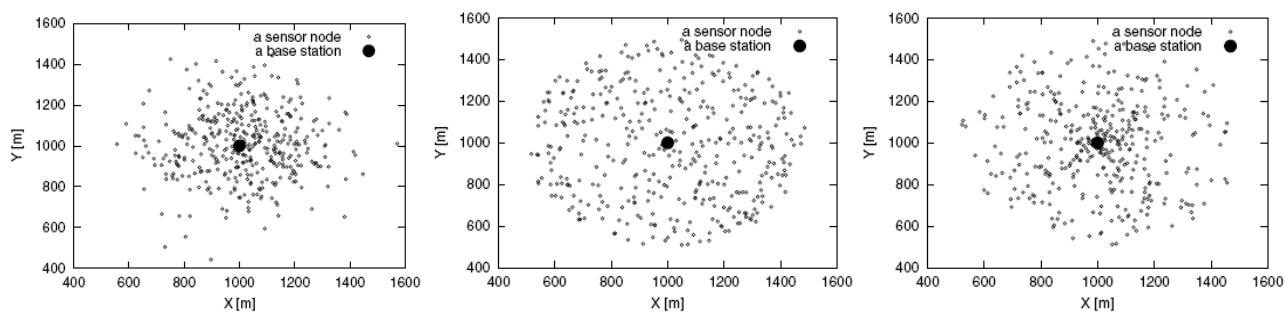
**Figure 3:** Sensors are mounted to analyze the vibration and assess the health of equipment at a semiconductor fabrication plant (picture is from [18])

Gallery model for which the fewest guards are to be placed to monitor a gallery. Similarly the work of Navarro et al. [25] addresses the orientation of video cameras for indoor surveillance so that high quality images of target objects are captured.

### 2.1.2 Random Node Distribution

Randomized sensor placement often becomes the only option. For example, in applications of WSNs in reconnaissance missions during combat, disaster recovery and forest fire detection, deterministic deployment of sensors is very risky and/or infeasible. It is widely expected that sensors will be dropped by helicopter, grenade launchers or clustered bombs. Such means of deployment lead to random spreading of sensors; although the node density can be controlled to some extent. Although it is somewhat unrealistic, many research projects, such as [26], have assumed uniform node distribution when evaluating the network performance. The rationale is that with the continual decrease in cost and size of micro-sensors, a large population of nodes is expected and thus a uniform distribution becomes a reasonable approximation.

Ishizuka and Aida [27] have investigated random node distribution functions, trying to capture the fault-tolerant properties of stochastic placement. They have compared three deployment patterns (Figure 4, from [27]): simple diffusion (2-dimensional normal distribution), uniform, and R-random, where the nodes are uniformly scattered with respect to the radial and angular directions from the base-station. The R-random node distribution pattern resembles the



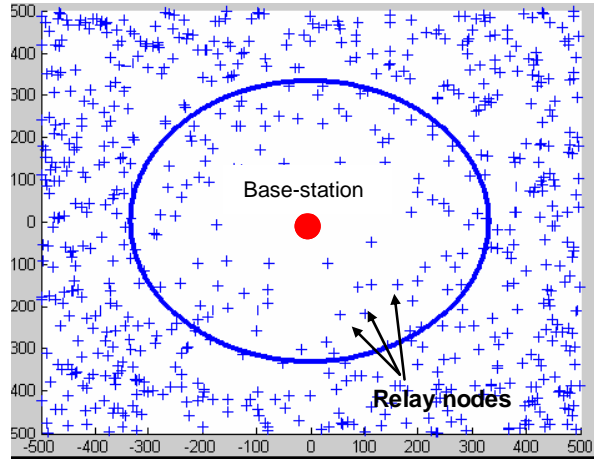
**Figure 4:** (a) Simple diffusion (b) Constant (uniform) placement (c) R-random placement

effect of an exploded shell and follows the following probability density function for sensor positions in polar coordinates within a distance  $R$  from the base-station:

$$f(r, \theta) = \frac{1}{2\pi R}, 0 \leq r \leq R, 0 \leq \theta < 2\pi \quad (1)$$

The experiments have tracked coverage and node reachability as well as data loss in a target tracking application. The simulation results indicate that the initial placement of sensors has a significant effect on network dependability measured in terms of tolerance of a node failure that may be caused by damage and battery exhaustion. The results also show that the R-random deployment is a better placement strategy in terms of fault-tolerance. The superiority of the R-random deployment is due to the fact that it employs more nodes close the base-station. In a multi-hop network topology, sensors near the base-station tend to relay a lot of traffic and thus exhaust their battery rather quickly. Therefore, the increased sensor population close to the base-station ensures the availability of spares for replacing faulty relay nodes and thus sustains the network connectivity.

While a flat architecture is assumed in [27], Xu et al. consider a two-tier network architecture in which sensors are grouped around relaying nodes that directly communicate with the base-station [28]. The goal of the investigation is to identify the most appropriate node distribution in order to maximize the network lifetime. They first show that uniform node distribution often does not extend the network lifetime since relay nodes will consume energy at different rates depending on their proximity to the base-station. Basically the further away the relays are from the base-station, the more the energy they deplete in transmission. To counter this shortcoming, a weighted random node distribution is then proposed to account for the variation in energy consumption rate in the different regions. The weighted random distribution, as depicted in Figure 5, increases the density of relays away from the base-station to split the load among more relays and thus extends their average lifetime. Although it has a positive impact on the network lifetime, the weighted random distribution may leave some relay nodes disjoint from the base-station since some relays may be placed so far that the base-station becomes out of their transmission range. Finally, a hybrid deployment strategy is introduced to balance the network lifetime and connectivity goals. The analysis is further extended in [29] for the case where relay nodes reach the base-station through a multi-hop communication path. The conclusion regarding the three strategies was found to hold in the multi-hop case as well.



**Figure 5:** An illustration of weighted random deployment (from [28]). The density of relay nodes inside the circle (close to the base-station) is lower and the connectivity is weaker than outside.

## 2.2. Primary Objectives for Deployment

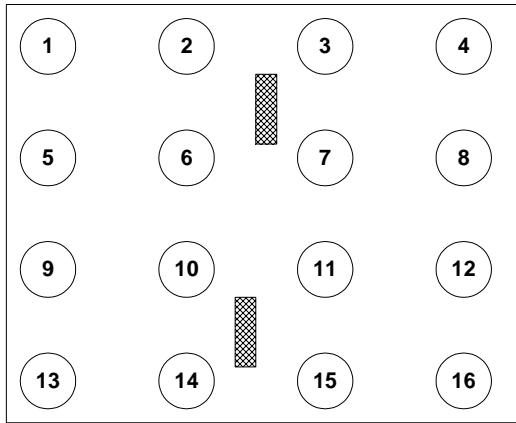
Application developers surely like the sensors to be deployed in a way that aligns with the overall design goals. Therefore, most of the proposed node placement schemes in the literature have focused on increasing the coverage, achieving strong network connectivity, extending the network lifetime and/or boosting the data fidelity. A number of secondary objectives such as tolerance of node failure and load balancing have also been considered. Most of the work strives to maximize the design objectives using the least amount of resources, e.g., number of nodes. Obviously, meeting the design objectives through random node distribution is an utmost challenge. Meanwhile, although intuitively deterministic placement can theoretically meet all primary and secondary objectives, the quest for minimizing the required network resources keeps the problem very hard. In this section, we categorize published work according to the optimization objective of the sensor placement. As we mentioned earlier the focus will be on sensor nodes that probe the environment and report their findings. Section 2.3 will cover placement strategies for nodes that play other roles, i.e., relaying and data collection.

### 2.2.1 Area Coverage

Maximal coverage of the monitored area is the objective that has received the most attention in the literature. Assessing the coverage varies based on the underlying model of each sensor's field of view and the metric used to measure the collective coverage of deployed sensors. The bulk of

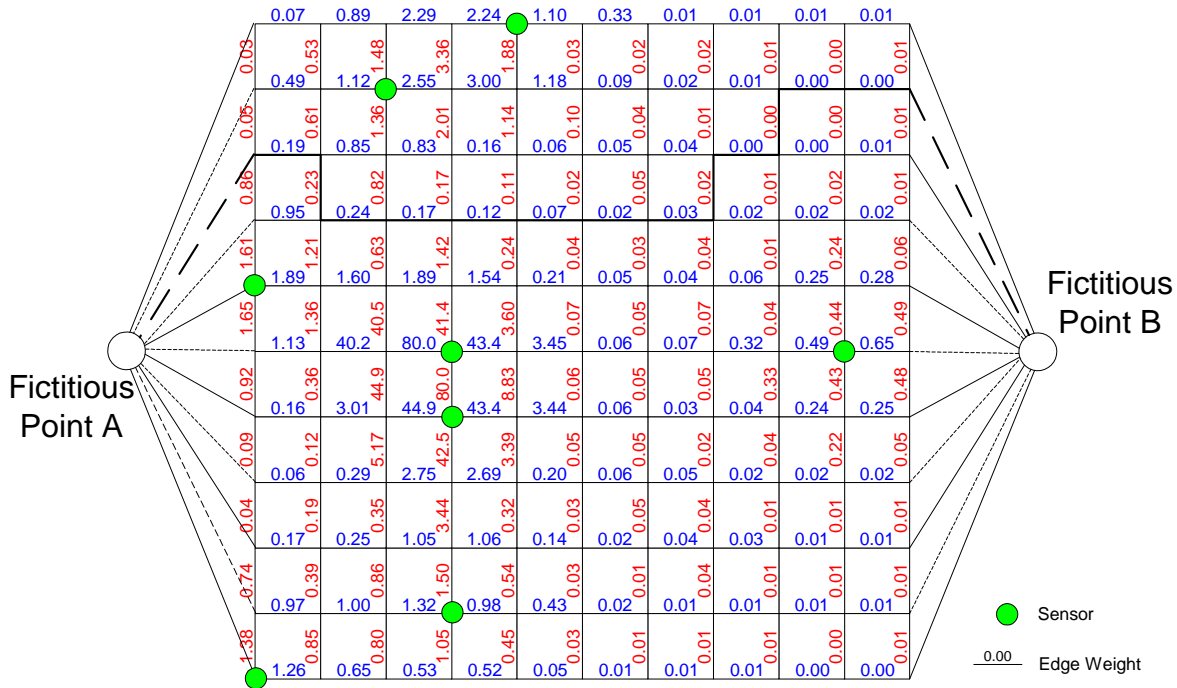
the published work, e.g., [30], assumes a disk coverage zone centered at the sensor with a radius that equals its sensing range. However, some recent work has started to employ more practical models of the sensor's field of view in the form of irregular polygons [31]. Some of the published papers, especially early ones, use the ratio of the covered area to the size of the overall deployment region as a metric for the quality of coverage [30]. Since 2001, however, most work has focused on the worst case coverage, usually referred to as least exposure, measuring the probability that a target would travel across an area or an event would happen without being detected [32]. The advantage of exposure-based coverage assessment is the inclusion of a practical object detection probability that is based on signal processing formulations, e.g., signal distortion, as applicable to specific sensor types.

As mentioned earlier, optimized sensor placement is not an easy problem, even for deterministic deployment scenarios. Complexity is often introduced by the quest to employ the least number of sensors in order to meet the application requirements and by the uncertainty in a sensor's ability to detect an object due to distortion that may be caused by terrain or the sensor's presence in a harsh environment. Dhillon and Chakrabarty have considered the placement of sensors on a grid approximation of the deployment region [11]. They formulate a sensing model that factors in the effect of terrain in the sensor's surroundings and inaccuracy in the sensed data (Figure 6). Basically the probability of detecting a target is assumed to diminish at an exponential rate with the increase in distance between a sensor and that target. A sensor can detect targets that lie in its line of sight. An obstacle may thus make a target undetectable. The sensing model is then used to identify the grid points on which sensors are to be placed, so that an application-specific minimum confidence level on object detection is met. They propose a greedy heuristic that strives to achieve the coverage goal through the least number of sensors. The algorithm is iterative. In each iteration, one sensor is placed at the grid point with the least coverage. The algorithm terminates when the coverage goal is met or a bound on the sensor count is reached.



**Figure 6:** Knowing the coordinate of obstacles, sensor's field of view is adjusted. In the shown example, redrawn from [11], the sensor on grid point 14 is not sufficient to cover the area in direction to point 11. The same applies for (2,7), (6,3) and (10,15).

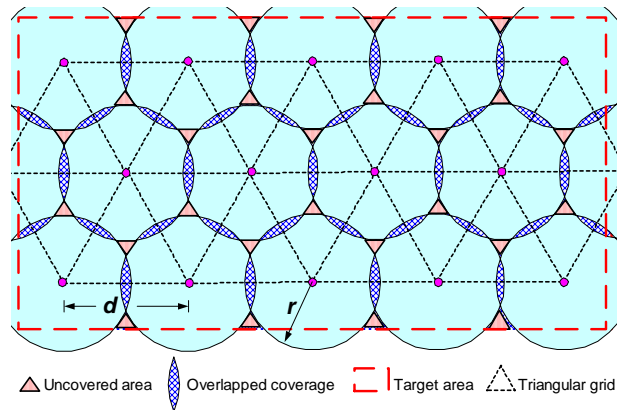
Clouqueur et al. also studied the problem of populating an area of interest with the least number of sensors so that targets can be detected with the highest probability [12]. Unlike [11], random deployment is assumed in this work. The authors propose a metric called path exposure to assess the quality of sensor coverage. The idea is to model the sensing range of deployed nodes and establish a collective coverage map of all sensors based on a preset probability of false-alarm (detection error). The map is then checked in order to identify the least exposure path, on which a target may slip by with the lowest probability of detection. Figure 7, which is redrawn from [12], illustrates the idea on a grid structure. Employing such a metric, the authors further introduced a heuristic for incremental node deployment so that every target can be detected with a desired confidence level using the lowest sensor count. The idea is to randomly deploy a subset of the available sensors. Assuming that the sensors can determine and report



**Figure 7:** The dark circle marks the location of each sensor. The weights on the individual line segments are based on the probability that a target is detected by all sensors (combined exposure). The least exposure path, marked with a dark line, is found by applying Dijkstra’s algorithm.

their positions, the least exposure path is identified and the probability of detection is calculated. If the probability is below a threshold, additional nodes are deployed in order to fill holes in the coverage. This procedure would be repeated until the required coverage is reached. The paper also tried to answer the question of how many additional nodes are deployed per iteration. On the one hand, it is desirable to use the least number of sensors. On the other hand, the means for sensor deployment may be expensive or risky, e.g., sending a helicopter. The authors derive a formulation that account for the cost of deploying nodes and the expected coverage as a function of sensor count. The formulation can be used to guide the designer for the most effective way to populate the area.

Pompili, et al. [33] have investigated the problem of achieving maximal coverage with the lowest sensor count in the context of underwater WSNs. Sensors are to be deployed on the bottom of the ocean along with a few gateway nodes. The sensors send their data to nearby gateways which forward it over vertical communication links to floating buoys at the surface. To achieve maximal coverage with the least number of sensors, a *triangular grid* has been proposed. The idea, as depicted in Figure 8, is to pursue a circle packing such that any three adjacent and non-collinear sensors



**Figure 8:** Sensor placement based on a triangular grid. Coverage can be controlled by adjusting the inter-node distance “ $d$ ”. The figure is redrawn from [33].



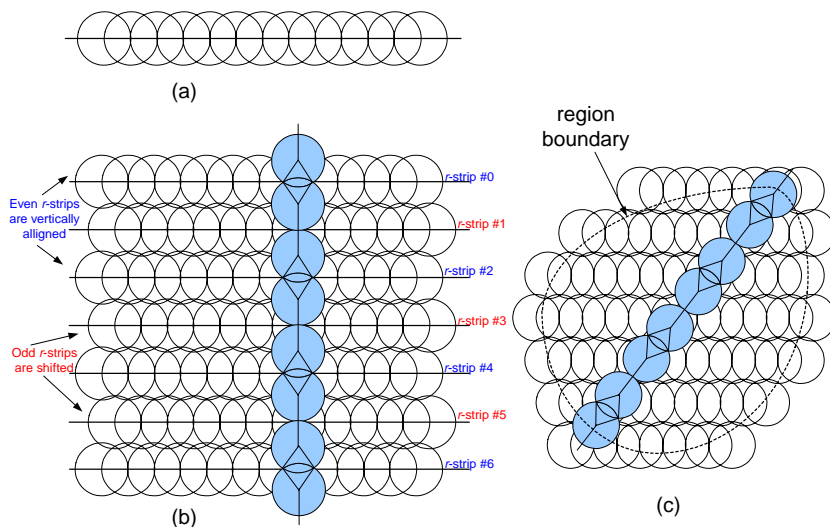
form an equilateral triangle. In this way, one can control coverage of the targeted region by adjusting the distance  $d$  between two adjacent sensors. The authors have proven that achieving 100% coverage is possible if  $d = \sqrt{3}r$  where  $r$  is the sensing range. The communication range of sensor nodes is assumed to be much larger than  $r$  and thus connectivity is not an issue. The authors further study how nodes can practically reach their assigned spots. Assuming nodes are to be dropped from the surface and sink until reaching the bottom of the ocean, the effect of water current is modeled and a technique is proposed to predict the trajectory of the sensor nodes and to make the necessary adjustment for the drop point at the surface.

The sensor placement problem considered by Biagioni and G. Sasaki [34] is more difficult. They opt to find a placement of nodes that achieves the coverage goals using the least number of sensors and also maintain a strongly connected network topology even if one node fails. The authors review a variety of regular deployment topologies, e.g., hexagonal, ring, star, etc. and study their coverage and connectivity properties under normal and partial failure conditions. They argue that regular node placement simplifies the analysis due to their symmetry despite the fact that they often do not lead to optimal configurations. The provided analytical formulation can be helpful in crafting a placement as a mix of these regular topologies and in estimating aggregate coverage of nodes.

### 2.2.2 Network Connectivity

Unlike coverage, which has constantly been an objective or constraint for node placement, network connectivity has been deemed a non-issue in some of the early work based on the assumption that the transmission range  $T_r$  of a node is much longer than its sensing range  $S_r$ . The premise is that good coverage will yield a connected network when  $T_r$  is a multiple of  $S_r$ . However, if the communication range is limited, e.g.,  $T_r = S_r$ , connectivity becomes an issue unless substantial redundancy in coverage is provisioned. It is worth noting that some work tackled the connectivity concern through deploying relay nodes that have long haul communication capabilities. Such approaches will be discussed in Section 2.3.

Kar and Banerjee have considered sensor placement for complete coverage and connectivity [35]. Assuming that the sensing and radio ranges are equal, the authors first define an  $r$ -strip as shown in Figure 9(a). In an  $r$ -strip, nodes are placed so that neighbors of a sensor along the x-axis are located on the circumference of the circle that defines the boundary of its sensing and communication range. Obviously, nodes on an  $r$ -strip are connected. The authors then tile the entire



**Figure 9:** Illustration of the placement algorithm in a plane and a finite size region [35].

plane with  $r$ -strips on lines  $y=k(0.5\sqrt{3}+1)r$  such that the  $r$ -strips are aligned for even values of the integer  $k$  and shifted horizontally  $r/2$  for odd values of  $k$ , as illustrated in Figure 9(b). The goal is to fill gaps in coverage with the least overlap among the  $r$ -disks that define the boundary of the sensing range. To establish connectivity among nodes in different  $r$ -strips, additional sensors are placed along the  $y$ -axis (the shaded disks in Figure 9(b)). For every odd value of the integer  $k$ , two sensors are placed at  $[0, k(0.5\sqrt{3}+1)r \pm 0.5\sqrt{3}r]$  to establish connectivity between every pair of  $r$ -strips. For a general convex-shaped finite-size region, connectivity among nodes in horizontal  $r$ -strips is established by another  $r$ -strip placed diagonally within the boundary of the region (Figure 9(c)). The authors generalize their scheme for the case where points of interest are to be covered rather than the whole area. However, unless the base-station is mobile and can interface with the WSN through any node, establishing a strongly connected network is not essential in WSNs since data are gathered at the base-station. Therefore, ensuring the presence of a data route from a node to the base-station would be sufficient and thus fewer nodes can be employed to achieve network connectivity than the presented approach would use. In addition, vertically placed nodes or diagonal  $r$ -strips can become a communication bottleneck since they act as gateways among horizontal  $r$ -strips, which may require the deployment of more sensors to split the traffic.

The focus of [36] is on forming  $K$ -connected WSNs.  $K$ -connectivity implies that there are  $K$  independent paths among every pair of nodes. For  $K>1$ , the network can tolerate some node and link failures and guarantee certain communication capacity among nodes. The authors study the problem of placing nodes to achieve  $K$ -connectivity at the network setup time or to repair a disconnected network. They formulate the problem as an optimization model that strives to minimize the number of additional nodes required to maintain  $K$ -connectivity. They show that the problem is NP-Hard and propose two approaches with varying degrees of complexity and closeness to optimality. Both approaches are based on graph-theory. The idea is to model the network as a graph whose vertices are the current or initial set of sensors and the edges are the existing links among these sensor nodes. A complete graph  $G$  for the same set of vertices (nodes) is then formed and each added edge is associated a weight. The weight of the edge between nodes  $u$  and  $v$  is set to  $\left(\frac{\overline{uv}}{r}-1\right)$ , where  $\overline{uv}$  is the Euclidean distance between  $u$  and  $v$ , and  $r$  is the radio range of a node. The weight basically indicates the number of nodes to be placed to establish connectivity between  $u$  and  $v$ . The problem is then mapped to finding a minimum-weight  $K$ -vertex-connected sub-graph “ $g$ ”. Finally, missing links (edges) in  $g$  are established by deploying the least number of nodes. The authors proposed employing one of the approximation algorithms in the literature for finding the minimum-weight  $K$ -vertex-connected sub-graph, which often involves significant computation. An alternative greedy heuristic has been also proposed for resource constrained setups. The heuristic simply constructs  $g$  by including links from  $G$  in a greedy fashion and then prunes  $g$  to remove the links that are unnecessary for  $K$ -connectivity. Again in most WSNs, it is not necessary to achieve  $K$ -connectivity among sensors unless the base-station changes its location frequently.

On the other hand, the authors of [37][38][39] promote the view that in massively dense sensor networks, it is unrealistic to model the network at the node level; something they call the microscopic level. Instead, they promote studying the node deployment problem in terms of macroscopic parameters such as node density. For example, in [38][39] they consider a network

of many sensors reporting their data to a spatially dispersed set of base-stations and opt to find a distribution function for sensor nodes so that data flows to base-stations over short routes and the traffic is spread. The goal is to minimize communication energy, limit interference among the nodes' transmission and avoid traffic bottlenecks. In [38], they assume that relaying nodes do not generate data, and that the amount of traffic going from one part of the network to another across a very small line segment is bounded. The latter assumption captures the effect of bandwidth limitation. They then formulate the node placement problem as an optimization function to find the best spatial density of nodes, i.e., probability density function. Analytical results indicate that the traffic flow between sensors and base-stations resembles the electrostatic field induced between positive and negative charges. The work is further extended in [39] by dropping the two assumptions about bandwidth limitation and the use of dedicated relay nodes and by employing a more elaborate physical layer model for the radio transmission and reception. The goal of the optimization in the new formulation is to minimize the number of sensors and find their optimal spatial density for delivering all data to the base-stations. Calculus of variations is pursued to derive an analytical solution. It is worth noting that coverage goals are not considered in the formulation and the authors seem to implicitly assume that massively dense networks ensure good area coverage.

### 2.2.3 *Network Longevity*

Extending network lifetime has been the optimization objective for most of the published communication protocols for WSNs. The positions of nodes significantly impact the network lifetime. For example, variations in node density throughout the area can eventually lead to unbalanced traffic load and cause bottlenecks [28]. In addition, a uniform node distribution may lead to depleting the energy of nodes that are close to the base-station at a higher rate than other nodes and thus shorten the network lifetime [40]. Some of the published work, such as [29] which we discussed earlier, has focused on prolonging the network lifetime rather than area coverage. The implicit assumption is that there is a sufficient number of nodes or the sensing range is large enough such that no holes in coverage may result.

The maximum lifetime sensor deployment problem with coverage constraints has been investigated in [41]. The authors assume a network operation model in which every sensor periodically sends its data report to the base-station. The network is required to cover a number of points of interest for the longest time. The average energy consumption per data collection round is used as a metric for measuring the sensor's lifetime. The problem is then transformed to minimizing the average energy consumption by a sensor per round by balancing the load among sensors. The idea is to spread the responsibility of probing the points of interest among the largest number of sensors and carefully assign relays so that the data is disseminated using the least amount of energy. A heuristic is proposed that tries to relocate sensors in order to form the most efficient topology. First, sensors are sorted in descending order according to their proximity to the point of interest that they cover. Starting from the top of the sorted list, the algorithm iterates on all sensors. In each iteration, the sensor is checked for whether it can move to another location to serve as a relay. The new location is picked based on the traffic flow and the data path that this node is part of or will be joining. Basically, the relocating node should reduce its energy consumption by getting close to its downstream neighbor. A sensor repositioning is allowed only if it does not risk a loss in coverage.

Chen et al. have studied the effect of node density on network lifetime [42]. Considering the one-dimensional placement scenario, the authors derive an analytical formulation for the network

lifetime per unit cost (deployed sensor). They also argue that network lifetime does not grow proportionally to the increased node population and thus a careful selection of the number of sensors is necessary to balance the cost and lifetime goals. Considering the network to be functional until the first node dies, an optimization problem is defined with the objective of identifying the least number of sensors and their positions so that the network stays operational for the longest time. An approximate two-step solution is proposed. In the first step, the number of sensors is fixed and their placement is optimized for maximum network lifetime. They formulate this optimization as a multi-variant non-linear problem and solve it numerically. In the second step, the number of sensors is minimized in order to achieve the highest network lifetime per unit cost. A closed form solution is analytically derived for the second step. A similar problem is also studied by Cheng et al. [43]. However, the number of sensors is fixed and the sensor positions are determined in order to form a linear network topology with maximal lifetime.

#### *2.2.4 Data Fidelity*

Ensuring the credibility of the gathered data is obviously an important design goal of WSNs. A sensor network basically provides a collective assessment of the detected phenomena by fusing the readings of multiple independent (and sometimes heterogeneous) sensors. Data fusion boosts the fidelity of the reported incidents by lowering the probability of false alarms and of missing a detectable object. From a signal processing point of view, data fusion tries to minimize the effect of the distortion by considering reports from multiple sensors so that an accurate assessment can be made regarding the detected phenomena. Increasing the number of sensors reporting in a particular region will surely boost the accuracy of the fused data. However, redundancy in coverage would require an increased node density, which can be undesirable due to cost and other constraints such as the potential of detecting the sensors in a combat field.

Zhang and Wicker have looked at the sensor placement problem from a data fusion point of view [44]. They note that there is always an estimation distortion associated with a sensor reading which is usually countered by getting many samples. Thus, they map the problem of finding the appropriate sampling points in an area to that of determining the optimal sampling rate for achieving a minimal distortion, which is extensively studied in the signal processing literature. In other words, the problem is transformed from the space to the time domain. The set of optimal sensor locations corresponds directly to the optimal signal sampling rate. The approach is to partition the deployment area into small cells, then determine the optimal sampling rate per cell for minimal distortion. Assuming that all sensors have the same sampling rate, the number of sensors per cell is determined.

Similar to [44], Ganesan et al. have studied sensor placement in order to meet some application quality goals [45]. The problem considered is to find node positions so that the fused data at the base-station meets some desired level of fidelity. Unlike [44], the objective of the optimization formulation is to minimize the energy consumption during communication; a tolerable distortion rate is imposed as a constraint to this optimization. Also, in this work the number of sensors is fixed and their position is to be determined. Given the consideration of energy consumption, data paths are modeled in the formulation, making the problem significantly harder. The authors first provide a closed form solution for the 1-dimensional node placement case and then use it to propose an approximation algorithm for node placement in a circular region. Extending the approach to handle other regular and irregular structures is noted as future work.

Wang et al. have also exploited similar ideas for a WSN that monitors a number of points of interest [46]. Practical sensing models indicate that the ability to detect targets or events diminishes with increased distance. One way to increase the credibility of the fused data is to place sensors so that a point of interest would be in the high-fidelity sensing range of multiple nodes. Given a fixed number of sensors, there is a trade-off between deploying a sensor in the vicinity of one point of interest to enhance the probability of event detection and the need to cover other points of interest. The probability of event detection by a sensor is called the *utility*. The utility per point of interest is thus the collective utility of all sensors that cover that point. The authors formulate a nonlinear optimization model to identify the sensor locations that maximize the average utility per point of interest. To limit the search space, the area is represented as a grid with only intersection points considered as candidate positions.

Finally we would like to note that the work of Clouqueur et al. [12], which we discussed earlier, can also be classified under data fidelity based sensor placement. They estimate the credibility of fused data from multiple sensors and use it to identify the position of sensors for maximizing the probability of target detection.

### **2.3. Role-Based Placement Strategies**

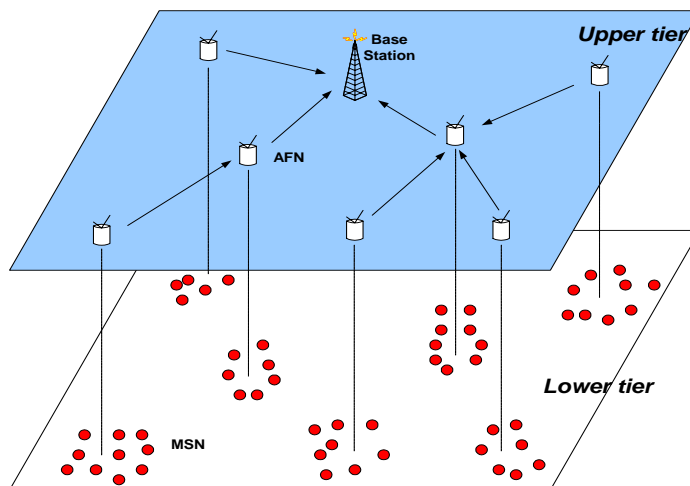
The positions of nodes not only affect coverage but also significantly impact the properties of the network topology. Some of the published work has focused on architecting the network in order to optimize some performance metrics, for example, to prolong the network lifetime or minimize packet delay. These architectures often define roles for the employed nodes and pursue a node-specific positioning strategy that is dependent on the role that the node plays. In this section, we opt to categorize role-based node placement strategies. Generally, a node can be a regular sensor, relay, cluster-head or base-station. Since the previous section covered the published work on sensor placement, we limit the scope in this section to surveying strategies for relay, cluster-head and base-station positioning. Since cluster-heads and base-stations often act as data collection agents for sensors within their reach, we collectively refer to them as data collectors.

#### **2.3.1 Relay Node Placement**

Positioning of relay nodes has also been considered as a means for establishing an efficient network topology. Contemporary topology management schemes, such as [47][48][49], assume redundant deployment of nodes and selectively engage sensors in order to prolong the network lifetime. Unlike these schemes, on-demand and careful node placement is exploited in order to shape the network topology to meet the desired performance goals. Many variants of the relay node placement problem have been pursued; each takes a different view depending on the relationship between the communication ranges of sensor and relay nodes, allowing a sensor to act as a hop on a data path, considering a flat or tiered network architecture and the objective of the optimization formulation. The assumed capabilities of the relay nodes vary widely. Some work considers the relay node (RN) to be just a sensor; especially for flat network architectures. In two-tier networks, RNs usually play the role of a gateway for one or multiple sensors to the other nodes in the network. The transmission range of RNs is often assumed larger than sensors. When RNs do not directly transmit to the base-station, the placement problem becomes harder since it involves inter-RN networking issues.

Hou et al. [50] have considered a two-tier sensor network architecture where sensors are split into groups; each is led by an aggregation-and-forwarding (AFN) node (Fig 10). A sensor sends its report directly to the assigned AFN, which aggregates the data from all sensors in its group.

The AFNs and the base-station form a second tier network in which an AFN sends the aggregated data report to the base-station over a multi-hop path. The authors argue that AFNs can be very critical to the network operation and their lifetime should be maximized. Two approaches have been suggested to prolong the AFNs' lifetime. The first is to provision more energy to AFNs. The second is to deploy relay nodes (RNs) in order to reduce the communication energy consumed by an AFN in sending the data to the base-station. The RN placement and energy provisioning problem is formulated as a mixed-integer



**Fig 10:** Logical view of the assumed two-tier network architecture (redrawn from [50]). MSN stands for micro-sensor nodes.

nonlinear programming optimization. For a pool with an  $E$  energy budget and  $M$  relay nodes, the objective of the optimization is to find the best allocation of the additional energy to existing AFNs and the best positions for placing the  $M$  relays. To efficiently solve the optimization problem, the formulation is further simplified through a two-phase procedure. In the first phase a heuristic is proposed for optimized placement of the  $M$  relay nodes. Given the known positions of the RNs, in the second phase the energy budget is allocated to the combined AFN and RN population, which is a linear programming optimization.

Unlike the work described earlier on randomized deployment of relay nodes (RNs) [28][29], the focus in [51][52][53][54] is on deterministic placement. The considered system model fits indoor applications or friendly outdoor setups. In [51], the authors consider the placement of relay nodes that can directly reach the base-station. Given a deployment of sensor nodes (SNs), the problem is to find the minimum number of RNs and where they can be placed in order to meet the constraints of network lifetime and connectivity. The network's lifetime is measured in terms of the time for the first node to die. Connectivity is defined as the ability of every SN to reach the base-station, implying that every SN has to have a designated RN. The problem is shown to be equivalent to finding the minimum set covering, which is an NP-Hard problem. Therefore, a recursive algorithm is proposed to provide a sub-optimal solution. The algorithm pursues a divide-and-conquer strategy and is applied locally. Sensors collaboratively find the intersections of their transmission ranges. Relay nodes are placed in the intersections of the largest number of sensors so that all sensors are served by a relay node.

This work has been further extended in [52][53][54] to address the problem of deploying a second tier of relay nodes so that the traffic is balanced and the least number of additional relay nodes are used. In [52], a lower bound on the number of required nodes is derived. The authors have also proposed two heuristics. The first is very simple and places a second level relay (SLR) at a distance from the first level relay (FLR) so that FLRs stay operational for the longest time. In the second heuristic, SLRs are allocated only to those FLRs that cannot reach the base-station. The number of SLRs is then reduced by removing redundancy. Basically, if  $SLR_i$  can serve both  $FLR_i$  and  $FLR_j$ ,  $SLR_j$  is eliminated since  $FLR_j$  is already covered. The validation results show that the second heuristic provides near optimal solutions in a number of scenarios. More

sophisticated approaches have been proposed in [53][54]. The main idea is that an existing FLR that is close to the base-station is considered as a candidate SLR before deploying new RNs to serve as SLRs.

The objective of the relay placement in [55] is to form a fault-tolerant network topology in which there exist at least 2 distinct paths between every pair of sensor nodes. All relay nodes are assumed to have the same communication range  $R$  that is at least twice the range of a sensor node. A sensor is said to be covered by a relay if it can reach that relay. The authors formulate the placement problem as an optimization model called “2-Connected Relay Node Double Cover (2CRNDC)”. The problem is shown to be NP-Hard and a polynomial time approximation algorithm is proposed. The algorithm simply identifies positions that cover the maximum number of sensors. Such positions can be found at the intersections of the communication ranges of neighboring sensors. Relay nodes are virtually placed at these positions. The analysis then shifts to the inter-relay network. Relays with the most coverage are picked and the algorithm checks whether the relays form a 2-connected graph and every sensor can reach at least 2 relays. If not, more relays are switched from virtual to real and the connectivity and coverage are rechecked. The latter step is repeated till the objective is achieved.

Tang et al. have also studied the RN placement problem in a two-tiered network model [26]. The objective is again to deploy the fewest RNs such that every SN can reach at least one RN node and the RNs form a connected network. In addition, the authors consider a variant of the problem in which each SN should reach at least 2 RNs and the inter-RN network is 2-connected in order for the network to be resilient to RN failures. Two polynomial time approximation algorithms are proposed for each problem and their complexity is analyzed based on a uniform SN deployment and on the assumption that the communication range  $R$  of a relay node is at least 4 times the communication range  $r$  of a SN. The idea is to divide the area into cells. For each cell an exhaustive search is performed to find positions that are reachable to all SNs in the cell. These positions can be determined using analytical geometry on the circumference of the circles that define the communication range of SNs. The authors argue that for small cells this is feasible since few sensors are expected to be in a cell. These positions are the candidates for placing RNs. The final RN positions are selected so that the RN of a cell is reachable to those RNs in neighboring cells. If this is not possible, additional RNs are placed at grid points to form a path between disjoint neighboring RNs. To support tolerance to RN failures, two RNs positions are picked per cell such that there are at least two non-overlapping paths between every pair of relay nodes. More RNs may be required to enable forming such independent routes.

On the other hand, Cheng et al. have considered a class of WSNs, such as biomedical sensor networks, in which the sensors positions are determined prior to deployment [8]. To boost the network lifetime and limit interference, it is desired to maintain network connectivity with minimum transmission power per node. The authors have tried to achieve this design goal by construction. They consider a homogeneous network where sensors can act as a data source and a relay. Given a set of deployed sensors in an area, they opt to place the minimum number of relay nodes for maintaining network connectivity such that the transmission range of each sensor does not exceed a constant  $r$ . They formulate the optimization problem as a Steiner Minimum Tree with Minimum number of Steiner Points (SMT-MSP), which is NP-Hard. To tackle the high complexity, they propose a polynomial time approximation algorithm. This algorithm runs in two phases. In the first, a minimum-cost spanning tree  $T$  is formed using the edge length as the cost. If an edge length exceeds the transmission range, RNs are placed on that edge to maintain

connectivity. In the second phase, the transmission power of each node is reduced to the minimum level needed to maintain the link to the next node on the tree  $T$ .

Lloyd and Xue have considered two RN placement problems [56]. The first is a generalization of the model of [8], discussed above, allowing RNs to have longer transmission range than SNs and allowing both SNs and RNs to act as data forwarders on a particular data path. This network architecture is still flat. In the second placement problem, a two-tier architecture is considered. The placement objective in that case is to deploy the least number of RNs in order to ensure inter-sensor connectivity. In other words, a sensor must be able to reach at least one RN, and the inter-RN network must be strongly connected. The second problem can be viewed as a generalization of [26] with  $R > r$ . The authors have crafted approximation algorithms for both problems based on finding the minimum spanning tree for the first problem and a combined SMT-MSP and Geometric Disk Cover algorithm for the second problem. The approach is further extended in [57] to form a  $k$ -connected network topology in order to tolerate occasional failure of relay nodes.

Table 1 categorizes the relay node placement problems and mechanisms discussed above.

**Table 1:** A comparison between the various approaches for relay node placement

Paper	Tiers	Objective	Radio ranges	Nodes on path	Connectivity goals/constraints
[8]	Single	- Minimal # of relays - Minimal per node transmission power	$R = r$	- Sensors - Relays	- Path between every pair of sensors
[26]	Two	- Minimal # of relays	$R \geq 4r$	- Relays	- Every sensor can reach at least 2 relays - 2 vertex-disjoint paths between every pair of relays
[50]	Two	- Max. AFN lifetime	N/A	- Relays	- Path from each AFN to base-station
[51]	Two	- Minimal # of relays	$R > r$	- Relays	- Relays directly reach base-station - Path from each sensor to base-station - Time to first node to die must exceed a threshold value
[52] [53] [54]	Two	- Minimal # of relays	$R > r$	- Relays	- Balance traffic loads among relays - Relay to base-station connectivity - Time to first relay node to die must exceed a threshold value
[55]	Two	- Minimal # of relays	$R \geq 2r$	- Relays	- 2 vertex-disjoint paths between every pair of sensors
[56]-1	Single	- Minimal # of relays	$R \geq r$	- Sensors - Relays	- Path between every pair of sensors
[56]-2	Two	- Minimal # of relays	$R \geq r$	- Relays	- Path between every pair of sensors
[57]	Two	- Minimal # of relays	$R \neq r$	- Relays	- $K$ vertex-disjoint paths between every pair of nodes (sensors and relays)

### 2.3.2 Placement of Data Collectors

Clustering is a popular methodology for enabling a scalable WSN design [58]. Every cluster usually has a designated cluster-head (CH). Unlike RNs, which forward data from some sensors as we discussed earlier, CHs usually collect and aggregate the data in their individual clusters



and often conduct cluster management duties consistent with their capabilities. When empowered with sufficient computational resources, CHs can create and maintain a multi-hop intra-cluster routing tree, arbitrate medium access among the sensors in the individual clusters, assign sensors to tasks, etc. When inter-CHs coordination is necessary, or CHs have too limited communication range to directly reach the base-station, CHs may need to form a multi-hop inter-CH network, i.e., a second-tier network. CHs can be also picked among the deployed sensor population. In that case, little control can be exerted unless the nodes are moveable.

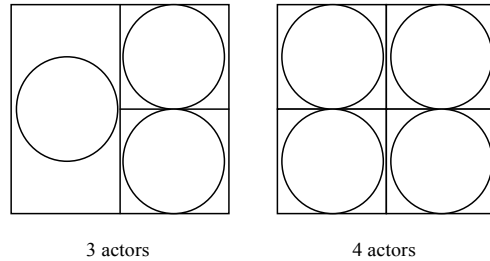
Careful positioning of cluster-heads in a hierarchical network has been deemed an effective strategy for establishing an efficient network topology [59]-[66]. The same applies to base-stations. In fact, most published approaches for CH placement apply to base-stations for the same network model, and vice versa. The similarity is mostly due to the fact that both CHs and base-stations collect the data from sensor nodes and take some leadership role in setting up and managing the network. In order to simplify the discussion in this section and better define the scope and limitations of the presented placement approaches, we will refer to both cluster-heads and base-stations as data collectors (DCs).

In general, the complexity of the DC placement problem varies based on the planned network architecture. If the sensors are assigned to distinct clusters prior to placing or finalizing the positions of DC nodes, the scope of the problem becomes local to the individual cluster and only concerns each DC independently from the others [60]. However, if the DC placement precedes the clustering process, the complexity is NP-Hard, proven in [59] through a reduction to the dominating set problem on unit disk graphs. We first discuss those approaches where clusters are formed before the DCs are positioned. In this case, the theme is typically to group sensors based on a static metric like physical proximity, then place a head node for every group of sensors in order to optimize some design objective. There is always an implicit assumption on how the network will operate. Usually sensor nodes are expected to transmit data continuously at a constant rate.

Oyman and Ersoy [60] employ the popular *k-means* clustering algorithm to form disjoint groups of sensors so that the average Euclidian distance between sensors in a cluster and its assigned DC is minimized. The algorithm is repeated for different numbers of clusters until a lower bound on network lifetime is reached with the least DC count. The network lifetime is estimated based on the energy consumed by sensors to reach their respective DC. A DC is placed at the centroid of the sensors of the individual clusters. The Genetic Algorithm for Hop count Optimization (GAHO) approach proposed in [61] follows a similar process. However, GAHO employs artificial neural networks for optimal sensor grouping and strives to minimize the data collection latency by reducing the number of hops that sensor readings have to pass through before reaching a DC. Instead of placing a DC at the centroid of its cluster, a search is conducted around the centroid to find a nearby position such that the DC is in the communication range of the largest number of sensors in its cluster. A lightweight version of GAHO, called Genetic Algorithm for Distance Optimization (GADO), is also proposed to achieve the same objective using the Euclidian distance between sensors and their assigned DC. GADO essentially trades the optimality of the DC positions for the complexity of the algorithm.

Data latency is also the optimization objective of COLA, an actor (DC) placement algorithm presented in [62]. In some application scenarios like disaster management and combat field reconnaissance, DCs not only collect and process the data but also can do some reactive actions, such as extinguishing a fire or de-mining a travel path. That is why DCs are referred to as actors.

In this case, the design objective is to minimize the delivery latency of sensor data and the time for an actor to reach the spot that needs attention. Initially the actors are positioned uniformly in order to maximize the coverage (i.e., minimize the overlap among the action ranges) of the area as shown in Figure 11. Sensors are then grouped into clusters; each is led by an actor. After clustering, each actor considers the positions of its assigned sensors as vertices and computes the vertex 1-center [7]. Relocating the actor at the vertex 1-center location ensures minimum delay from the farthest sensor node, i.e., minimize the maximum latency for data delivery.



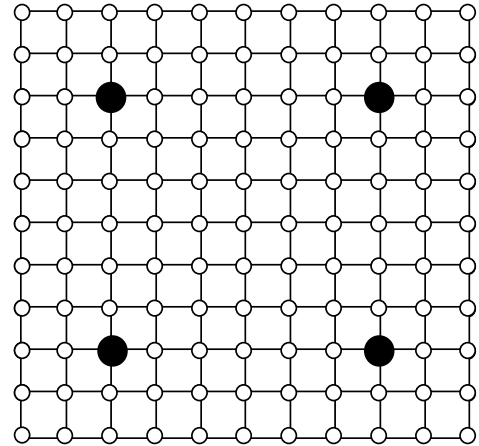
**Figure 11:** Initially, data-collectors (actors) are uniformly placed in the area of interest. Circles define the acting range of an actor.

However, when relocating an actor to its 1-center location, it may lose its connection with the other actors in the network. In order to also ensure inter-actor connectivity, the approach in [62] is further extended in [63]. Connectivity is maintained by moving an actor close to the vertex 1-center of its cluster as much as possible without breaking the links with its neighbors. The relocation of the individual actors is organized following a global order based on the IDs of actors so as not to disconnect the network with simultaneous relocations of neighboring actors.

There are also some approaches which deploy the DCs before grouping the sensor nodes. The clustering process in this case strives to set up an optimal network topology; picking the right DC for sensors to send their data to. Given the scope of the paper, we focus on those approaches that pre-select the positions of DC nodes; that is, they pursue a controlled placement of DCs. The placement problem in this case is more challenging and is shown to be NP-Hard [59]. Published solutions usually tackle the complexity of the optimization by restricting the search space. For instance, in [7] the search space is restricted to the sensor locations and the best position “ $s$ ” among them is picked in terms of network lifetime. This solution is shown to be a constant approximation of the optimal solution, e.g., achieves a fraction of the optimal network lifespan. The approximation ratio is further improved to  $(1-\epsilon)$ , where  $\epsilon > 0$  is any desired error bound, by factoring in the routes and transmission schedule. However, the improvement comes at the cost of increased computation for solving multiple linear programs. To limit such high computation, a technique is proposed which explores the potential overlap among the elements of the search space [64]. The idea is to replace an infinite search space for each variable by a finite-element search space with a guaranteed bound on the possible loss in performance. Specifically, the search space grows exponentially with the increase in the number of variables and such growth can be reduced by exploring the potential overlap among the elements of the search space. In order to determine the potential overlap, the variables are expressed in the form of a geometric progression and a common factor among these geometric progressions is identified.

Bogdanov et al. have studied the problem of determining the optimal network topology for multiple DCs with the aim of maximizing the data rate while minimizing the energy consumption at the individual sensors [59]. Such optimization arises in setups where the sensors’ batteries can be recharged at a fixed rate and most data is to be collected between successive energy scavenging cycles. For networks with a fixed data rate, the problem becomes equivalent to minimizing the power used to collect the data. The data rate mainly varies at relaying nodes,

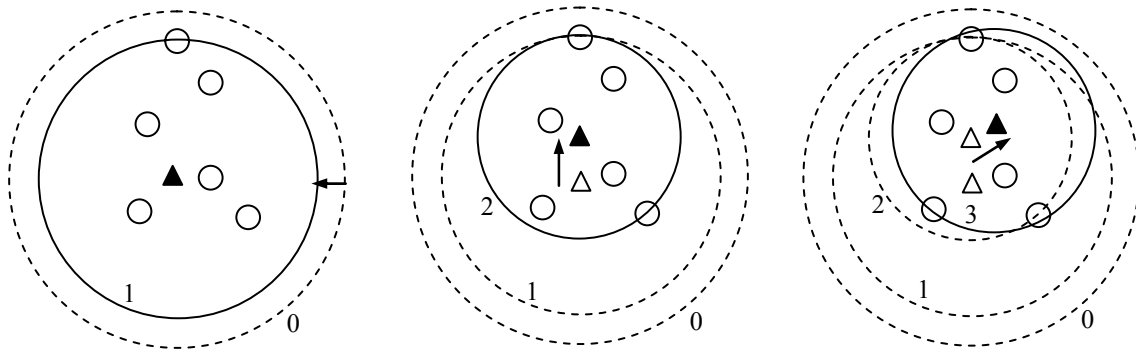
which forward other sensor's data in addition to sending their own. The authors argue that appropriate placement of data collectors can shorten the data routes and prevent the overloading of relaying nodes beyond their maximum achievable data rate, which is determined based on the capacity of their onboard battery. Given the complexity of the placement problem, the solution space is also limited by allowing DCs to be located only at sensor positions. Once the DCs are placed, each sensor designates the DC that can reach it over the shortest path. There is no explicit clustering performed. Two different heuristics, namely greedy and local search, were presented. The greedy scheme deploys DCs incrementally. Basically, the first DC is placed so that the data rate is maximized and then the second is placed assuming the position of the first DC is fixed and so on. On the other hand, the local search starts with a random placement of DCs. Each DC then tries to relocate to the position of a neighboring sensor in order to maximize the data rate. If no improvement is possible, the algorithm records the best achievable data rate and stops. This process is repeated several times for different random configurations and the one with the highest data rate is finally picked. Figure 12 shows the optimal layout for 4 DCs for a 10×10 grid of sensors whose transmission ranges are 2.2 units. In the figure, the dark circles designate the DCs while the smaller circles designate the sensor nodes.



**Figure 12:** Optimal DC locations (black circles) achieved by the local search.

In [65], optimizing the placement of the DC nodes is formulated as an integer linear programming (ILP) model. The objective function of the ILP formulation is to minimize the maximum energy consumption at the individual sensors while minimizing the total communication energy. The constraints of the ILP formulation include a bound on the total energy consumed by a node in a data collection round and a restriction on the candidate DC location to be picked from a set of pre-determined positions. Other constraints are also specified to ensure a balanced flow through the individual nodes and to allow transmission of messages to a feasible site only if a DC is to be placed at that site. The optimization is conducted by one of the DCs, i.e., the approach is centralized. After the DC locations are determined, a flow-based routing algorithm is used in order to pick the right DC for each sensor and determine the data paths. Unlike [59], DC positions are re-computed periodically at the beginning of each data collection round in order to cope with changes in the network state and distribute the routing load among the sensors evenly.

The approach of [66] does not consider data relaying and thus the problem becomes solvable in polynomial time. In order to minimize the total communication power, the DC node is to be located such that the maximum distance to a sensor node is minimized [66]. A computational geometry based algorithm, whose complexity is linear in the number of nodes  $n$ , is proposed. This algorithm tries to determine the circle with the smallest diameter that encloses the nodes. Such a circle can be formed with at most three points picked among the locations of the sensors. The DC will then be positioned at the center of the circle [66]. The same algorithm is further extended for a two-tiered WSN where special *application nodes* are designated as first tier DCs [10]. An application node interfaces its cluster with the base-station. The minimum enclosing circle is thus found for the application nodes as shown in Figure 13, redrawn from [10].



**Figure 13:** Finding the minimum enclosing circle for 6 application nodes [10]. The data-collector (black triangle) is placed at the center of the smallest disk that contains all application nodes (small circles).

Table 2 provides a comparative summary of the characteristics of the static node placement mechanisms discussed in this section.

**Table 2:** A comparison between the various approaches for nodes placement

Paper	Application	Space	Deployment	Node Type	Primary Objective	Secondary Objective	Constraint
[7]	Generic	2-D	Deterministic	Data Collector	Network Lifetime	--	--
[8]	Biomedical sensor networks	2-D	Deterministic	Relay	Network Lifetime	Min. relay count	Connectivity
[10]	Generic	2-D	Deterministic	Data Collector	Network Lifetime	--	--
[11]	Surveillance	2-D	Deterministic	Sensor	Coverage	Min. Sensor count	--
[12]	Outdoor	2-D	Random	Sensor	Data fidelity & Coverage	Min. Sensor count	--
[16]	Surveillance	3-D	Deterministic	Sensor	Coverage	Connectivity	--
[17]	Surveillance	3-D	Deterministic	Sensor	Coverage	Data fidelity	--
[18]	Manufacturing	3-D	Deterministic	Sensor	Data fidelity	Connectivity	
[19]	Structural health monitoring	3-D	Deterministic	Sensor	Data fidelity	Connectivity	--
[20]	Structural health monitoring	3-D	Deterministic	Sensor	Data fidelity	Connectivity & Fault-tolerance	--
[21]	Contamination detection	2-D	Deterministic	Sensor	Coverage	--	Fixed sensors count
[22]	Contamination detection	2-D	Deterministic	Sensor	Coverage	Delay	Fixed sensors count
[24]	Generic	3-D	Deterministic	Sensor	Data fidelity	Min. Sensor count	--
[25]	Generic	3-D	Deterministic	Sensor	Data fidelity	--	--
[26]	Outdoor	2-D	Deterministic	Sensor	Min. relay count	Fault-tolerance	Connectivity
[27]	Outdoor	2-D	Random	Sensor	Coverage & Connectivity	Fault-tolerance	--
[28]	Outdoor	2-D	Random	Relay	Network lifetime	--	--
[29]	Outdoor	2-D	Random	Relay	Network lifetime	--	--
[33]	Underwater	2-D	Deterministic	Sensor	Coverage	Min. Sensor count	--
[34]	Outdoor	2-D	Deterministic	Sensor	Coverage & Connectivity	& Fault-tolerance	--
[35]	Outdoor	2-D	Deterministic	Sensor	Coverage & Connectivity	Min. Sensor count	--
[36]	Outdoor	2-D	Deterministic	Sensor	Connectivity	Fault-tolerance	--
[38]	Massively dense networks	2-D	Random	Relay	Min. sensor count	Delay and energy	Bandwidth
[39]	Massively dense networks	2-D	Random	Sensor	Min. sensor count	Delay and energy	--
[41]	Surveillance	2-D	Controlled	Sensor	Network Lifetime	--	Coverage

			(nodes move)				
[42]	Generic	1-D	Deterministic	Sensor	Min. relay count Network lifetime	--	Coverage
[43]	Generic	1-D	Deterministic	Sensor	Network lifetime	--	Connectivity
[44]	Outdoor	2-D	Random	Sensor	Data fidelity	--	--
[45]	Generic	1-D 2-D	Deterministic	Sensor	Data fidelity	Minimal energy consumption in communication	- Lower Bound on tolerable distortion - Fixed sensors count
[46]	Surveillance	2-D	Deterministic	Sensor	Data fidelity	--	--
[50]	Generic	2-D	Controlled (nodes move)	Relay	Network Lifetime	--	Fixed relays count
[51]	Indoor or non-harsh outdoor	2-D	Deterministic	Relay	Min. relay count	--	- Network lifetime - Connectivity
[52]	Indoor or non-harsh outdoor	2-D	Deterministic	Relay	Network Lifetime	Min. relay count	--
[53]	Indoor or non-harsh outdoor	2-D	Deterministic	Relay	Network Lifetime	Min. relay count	--
[54]	Indoor or non-harsh outdoor	2-D	Deterministic	Relay	Network Lifetime	Min. relay count	--
[55]	Generic	2-D	Deterministic	Relay	Connectivity & Fault-tolerance	Min. relay count	--
[56]	Generic	2-D	Deterministic	Relay	Min. relay count	--	Connectivity
[57]	Generic	2-D	Deterministic	Relay	Fault-tolerance	Min. relay count	Connectivity
[59]	Generic	2-D	Deterministic	Data Collector	Max. Data Flow	Min. energy	--
[60]	Generic	2-D	Deterministic	Data Collector	Network Lifetime	Min CH count	--
[61]	Generic	2-D	Deterministic	Data Collector	Delay	Energy	--
[62]	Generic	2-D	Random	Data Collector	Coverage	Delay	--
[63]	Generic	2-D	Random	Data Collector	Coverage	Delay	Connectivity
[64]	Generic	2-D	Deterministic	Data Collector	Network Lifetime	--	--
[65]	Generic	2-D	Deterministic	Data Collector	Network Lifetime	Load Balancing	--
[66]	Surveillance	2-D	Deterministic	Data Collector	Network Lifetime	--	--

### 3. Dynamic Repositioning of Nodes

Most of the protocols described above initially compute the optimal location for the nodes and do not consider moving them once they have been positioned. Moreover, the context of the pursued optimization strategies is mainly static in the sense that assessing the quality of candidate positions are based on performance metrics like the data rate, sensing range, path length in terms of the number of hops from a sensor node to the base-station, etc. In addition, the placement decision is made at the time of network setup and does not consider dynamic changes during the network operation. For example, traffic patterns can change based on the monitored events, or the load may not be balanced among the nodes, causing bottlenecks. Also, application-level interest can vary over time, and the available network resources may change as new nodes join the network, or as older nodes run out of energy.

Therefore, dynamically repositioning the nodes while the network is operational is necessary to further improve the performance of the network. For instance, when many of the sensors in the vicinity of the base-station stop functional due to the exhaustion of their batteries, some redundant sensors from other parts of the monitored region can be identified and relocated to replace the dead sensors in order to improve the network lifetime. Such dynamic relocation can also be very beneficial in a target tracking application where the target is mobile. For instance, some of the sensors can be relocated close to the target to increase the fidelity of the sensor's data. Moreover, in some applications it may be wise to keep the base-station a safe distance from

harmful targets, e.g., an enemy tank, by relocating it to safer areas in order to ensure its availability.

Relocating the nodes during regular network operation is very challenging. Unlike initial placement, such relocation is pursued in response to a network- or environment-based stimulus. It thus requires continual monitoring of the network state and performance as well as analysis of events happening in the vicinity of the node. In addition, the relocation process needs careful handling since it can potentially cause disruption in data delivery. The basic issues can be enumerated as follows: when does it make sense for a node to relocate, where should it go and how will the data be routed while the node is moving? In this section we discuss these issues in detail and survey published approaches on dynamic node repositioning. We group published work according to whether the node being repositioned is a sensor or a data collector. In all the techniques covered in this section, no coordination among relocated nodes is provisioned. Collaborative multi-node relocation is an emerging area of research and is exclusively covered in Section 4.

### **3.1. Relocation Issues**

When to Consider Relocation: The decision for a node movement has to be motivated by either an unacceptable performance measure (despite setting up the most efficient network topology) or a desire to boost such measures beyond what is achievable at the present node position. Motives vary based on the targeted design attributes. Examples include the observation of bottlenecks in data relaying, decreases in node coverage in an area, increases in packet latency or excessive energy consumption per delivered packet. A weighted average may also be pursued to combine multiple metrics based on the application at hand.

Once a node has its motive, it will consider moving to a new position. Such consideration does not necessarily lead to an actual relocation. The node first needs to qualify the impact of repositioning at the new location on the network performance and operation. Therefore the “When” and “Where” issues of node movement are very closely inter-related. In addition, the node must assess the relocation overhead. Such overhead can be incurred by the node and the network. For example, if the node is a robot, the energy consumed by the mechanical parts during the movement is a significant overhead to the lifetime of the robot’s battery and thus should be minimized. Moreover, when energy and timeliness metrics are of utmost concern, the impact on the lifetime of individual sensors and on route maintenance has to be considered respectively.

Where to Relocate: When having a motive to relocate, the node needs to identify a new position that would satisfy the motive, e.g., boost overall network performance. Again, the qualification of the new position and possibly the search criteria may vary based on the design attributes. Finding an optimal location for the node in a multi-hop network is a very complex problem. The complexity is mainly resulting from two factors. The first is the potentially infinite number of possible positions that a node can be moved to. The second factor is the overhead of keeping track of the network and the node state information for determining the new location. In addition, for every interim solution considered during the search for an optimal position, a new multi-hop network topology may need to be established in order to compare that interim solution to the current or previously picked positions.

A mathematical formulation of the node relocation problem may involve huge numbers of parameters including the positions of all deployed nodes, their state information (including energy reserve, transmission range, etc.) and the data sources in the network. In addition, a node may need to know the boundaries of the monitored region, the current coverage ratio of the network, the location of dead sensor nodes or other information in order to determine its new location. Given the large number of nodes typically involved in applications of WSNs, the pursuance of exhaustive search will be impractical. In addition, the dynamic nature of the network makes the node state and sources of data may change rapidly; thus the optimization process may have to be repeated frequently. Moreover, it may be undesirable to involve the nodes in complex computation since this diverts both the computation capacity needed for application level processing, e.g., data fusion, and the energy needed for movement of the node. Therefore, approximate and local solutions, or search heuristics, are more popular in the context of WSNs [7][13][14].

Managing and Justifying the Move: Once the new location of the node has been picked and confirmed to enhance some desired design attributes, the node should identify a travel path to the new location. The main contributing factors to the path selection are the total distance to be traveled, the suitability of the terrain, the path safety and the risk of disrupting the network operation. Minimizing the travel distance for the nodes is crucial since the energy consumed by the mechanical parts in such a movement is much more than the communication and computation energy. Therefore, the shortest possible path should be identified to reach the new location. However, the node also has to pick a path that is physically feasible to travel over. The node may need to consult a terrain map or rely on special onboard equipments, e.g., cameras, to avoid obstacles and dead ends. The other concern is protecting the node during the move. Since a WSN is usually deployed in harsh environments to detect and track dangerous targets or events, the node should avoid exposure to harm or getting trapped. For example, the node should not go through a fire to reach the new location.

The node should also minimize any negative impact on network operation. While the node is on the move, it must ensure that data continues to flow. For example, a sensor may have to increase its transmission power to cover the planned travel path in order to make sure that packets will continue to reach it. Continual data delivery prevents the node from missing important reports during the relocation, which may cause an application level failure. Such application level robustness is a design attribute in itself. Therefore, it is desirable to restrict changes to the network topology. Avoiding radical changes to the data routes limits the disruption of ongoing data traffic, and also curtails the overhead that the relocation introduces. Again, the node performs a trade-off analysis between the gain achieved by going to a new location and the overhead in terms of additional energy consumption that the motion imposes on the other nodes. If the motion is justified, the node can physically relocate.

The last issue is whether there are constraints on the time duration that the node budgets for the move. These constraints may arise in very dynamic environments in which the traffic pattern changes frequently. In these cases, the gains achieved by going to a location may be lost or degraded very quickly and the node would find out that it has to move yet again to a third location or even return to the old position. In the worst case, the node continues to move back and forth among these locations. Therefore, a gradual approach to the new location may be advisable in order to prevent this scenario.

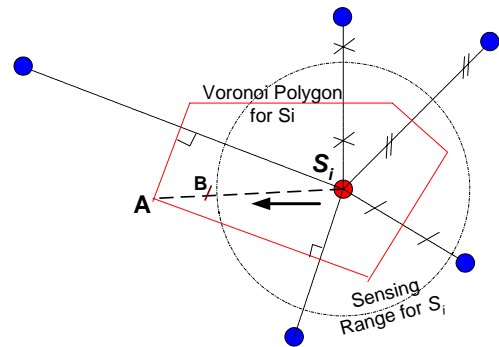
### 3.2. Sensor Repositioning Schemes

While the bulk of published work envisions sensors to be stationary, some investigate the possibility of attaching sensors to moveable entities such as robots [68][69]. Sensor mobility has been exploited to boost the performance of WSNs. For example, mobile sensors can re-spread in an area to ensure uniform coverage, move closer to loaded nodes in order to prevent bottlenecks or increase bandwidth by carrying data to the base-station [15][70][71][72]. Proposed schemes for dynamic sensor positioning in the literature can be categorized into two groups, based on when relocation is exploited, into post-deployment or on-demand relocation. We discuss these two categories of relocation in detail in the following subsections.

#### 3.2.1 Post-Deployment Sensor Relocation

This type of relocation is pursued at the conclusion of the sensor deployment phase when the sensor nodes are being positioned in the area. As we discussed earlier, in most WSN applications, sensor deployment is performed randomly due to the inaccessibility of the monitored areas. However, this random configuration usually does not provide adequate coverage of the area unless an excessive number of nodes are deployed. Alternatively, the coverage quality can be improved by moving the sensor nodes if they are able to do so. In that case, sensor nodes can be relocated to the regions with inadequate coverage, or no coverage at all. Given the energy cost of mechanical movement and the communication messages involved in directing the motion, the relocation process should be lightweight and should conclude in a reasonable time.

Wang et al. utilizes each sensor's ability to move in order to distribute them as evenly as possible in the region [70]. The goal is to maximize the area covered within the shortest time duration and with minimal overhead in terms of travel distances and inter-sensor message traffic. The main idea is that each sensor assesses the coverage in its vicinity after deployment and decides whether it should move to boost the coverage. To assess the coverage, a sensor node creates a Voronoi polygon with respect to neighboring sensors, as illustrated in Figure 14. Every point inside a Voronoi polygon is closer to the sensor of that polygon, i.e.,  $S_i$  in Figure 14, than any other sensor. The intersection of the disk that defines the sensing range and the Voronoi polygon represents the area the sensor can cover. If there are uncovered areas within the polygon, the sensor should move to cover them.



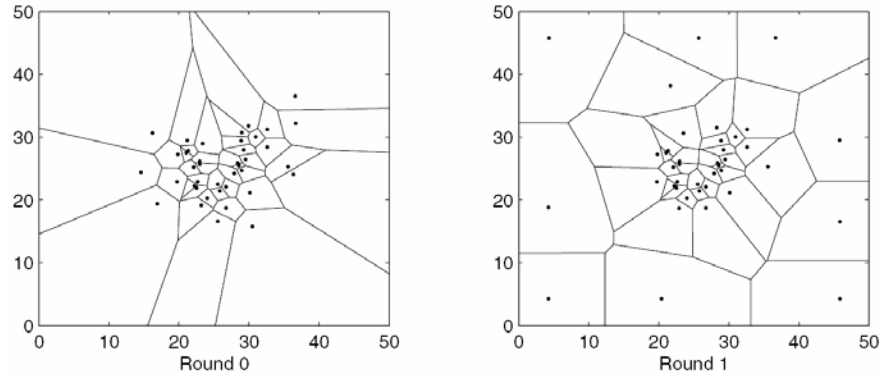
**Figure 14:** Every sensor  $S_i$  forms a Voronoi polygon with respect to the position of its neighboring sensors. The part of the polygon that lies outside the sensing range is not covered by  $S_i$ .

In order to decide where to reposition a sensor, three methods have been proposed: Vector-based (VEC), Voronoi-based (VOR) and Minimax. The main idea of the VEC method is borrowed from electromagnetic theory where nearby particles are subject to an expelling force that keeps them apart. In the context of WSNs, virtual forces are applied to a sensor node by its neighbors and by the boundaries of its Voronoi polygon in order to change its location. While in VEC the nodes are pushed away from the densely populated areas, VOR pulls the sensors to the sparsely populated areas. In VOR, the sensor node is pulled towards the farthest Voronoi vertex



to fix the coverage hole in the polygon, which is point A for sensor  $S_i$  in Figure 14. However, the sensor will be allowed to travel only a distance that equals half of its communication range, point “B” in Figure 14. This prevents the node from stepping into the area handled by another sensor that was out of reach prior to the move (in other words, not a current neighbor of  $S_i$ ), which can lead to an unnecessary move backward later on. In the Minimax method, a sensor also gets closer to its farthest Voronoi vertex. However, unlike VOR, the Minimax approach strives to keep most of the other vertices of the Voronoi polygon within the sensing range. It thus relocates the sensor to a point inside the Voronoi polygon whose distance to the farthest Voronoi vertex is minimized. The Minimax scheme is more conservative in the sense that it avoids creating coverage holes by going far from the closest vertices, leading to a more regularly shaped Voronoi polygon.

The conserved departure from the current sensor position leads to a gradual relocation, round by round, as shown in Figure 15. This usually causes the sensor to zigzag rather than move directly to the final destination. In order to shorten the total travel distance, a proxy-based approach is proposed in [73]. In this approach,



**Figure 15:** Sensors pursue relocation iteratively until movement no longer improves coverage (taken from [70]).

the sensor nodes do not move physically unless their final destination is computed. The authors consider a network with stationary and mobile sensors. Mobile sensors are used to fill coverage holes identified in a distributed way by stationary nodes. Thus, mobile sensors only move logically and designate the stationary sensor nodes as their proxies. This approach significantly reduces the total and average distance traveled by mobile nodes while maintaining the same level of coverage as [70]. The approach only increases the message complexity. However, given that movement is more costly in terms of energy, this increase can be justified. Nonetheless, this process still can be very slow and hence prolong the deployment time.

With the objective of reducing the overall deployment time, Wu et al. have proposed another solution to the same problem based on two dimensional scanning of clustered networks, called SMART [15]. The approach adopts a popular scheme for balancing load among nodes in parallel processing architectures by assigning equal number of tasks to each processor. This idea is applied to a multi-cluster WSN where each cluster is represented by a square cell, forming a 2D mesh. An example is shown in Figure 16, which is redrawn from [15]. The number of sensors annotated on every cell represents the load of that cluster. Each cluster-head knows only its location within the mesh, i.e., row and column indices, and the number of sensors in its cluster. It is assumed that a cluster-head can only communicate with its counterparts in neighboring cells. Achieving uniform coverage is then mapped to the load-balancing problem with a goal of evening the distribution of sensors among the clusters. To achieve this goal, each cluster-head performs both row-based and column-based scans to exchange load information. In a row-based scan, the leftmost cluster-head forwards its load, i.e., number of sensors, to its right neighbor. Each neighbor in the row adds the received load to its own load and forwards it until the

1	2	3	2	1	3
2	22	3	7	56	18
4	12	2	30	1	5
3	7	2	1	2	3
1	5	19	3	21	77
6	7	10	10	4	5

2	2	2	2	2	2
18	18	18	18	18	18
9	9	9	9	9	9
3	3	3	3	3	3
21	21	21	21	21	21
7	7	7	7	7	7

10	10	10	10	10	10
10	10	10	10	10	10
10	10	10	10	10	10
10	10	10	10	10	10
10	10	10	10	10	10
10	10	10	10	10	10

(a)
(b)
(c)

**Figure 16:** Steps for SMART: a) Initial 2D mesh b) Row scan c) Column scan

rightmost cluster-head is reached. This cluster-head computes the average load for its row and sends a message back until the left most cluster-head receives it (Figure 16b). After the scan process, the sensors are relocated to match the desired node count per cluster. That is, the overloaded clusters give sensors and the underloaded clusters take sensors. The same procedure is also applied within each column (Figure 16c). The approach also handles possible holes in the network when there are clusters with no sensors. The simulation results in [15] compared SMART to VOR, discussed above, with respect to the number of moves made by the sensors and the number of rounds before termination. SMART was shown to use the minimum number of moves. Although it was also shown that SMART converges in a fewer number of rounds for densely populated WSNs, VOR was found to be superior for sparsely populated networks.

Another similar post-deployment relocation work for improving the initial coverage and providing uniform distribution of sensors is presented in [74]. Although the idea is similar to the VEC mechanism of [70], this time it is inspired by the equilibrium of particles in Physics. The particles follow Coulomb's law, pushing against each other to reach equilibrium in an environment. Therefore, the authors define forces for each sensor node in the network based on the inter-node distances and the local density of nodes. The partial force on a node  $i$  from node  $j$  at time  $t$  is expressed as follows:

$$f_t^{i,j} = \frac{D_t^i}{\mu^2} (R | p_t^i - p_t^j |) \frac{p_t^j - p_t^i}{|p_t^j - p_t^i|}, \quad (2)$$

where  $D_t^i$  is the node density in the vicinity of node  $i$  at time  $t$ ,  $\mu$  is the average node density in the network,  $R$  is the transmission range and  $p_t$  is the location at time  $t$ . In equation (2), the  $\frac{D_t^i}{\mu^2} (R | p_t^i - p_t^j |)$  term defines the force strength. The authors note that including the average

node density  $\mu$  is found to expedite the convergence. The term  $\frac{p_t^j - p_t^i}{|p_t^j - p_t^i|}$  indicates the direction

of the force. Each sensor's movement is decided by the combined force applied to that sensor by all neighboring nodes. A sensor node is only allowed to move a certain distance per time step. In some cases, the node can move back and forth between two locations leading to an oscillation. If an oscillation continues for longer than a preset limit, the node stays at the center of gravity between the oscillation points. To validate the performance, the approach is implemented and compared to a simulated annealing based solution which provides optimal coverage. The validation results indicate that the proposed self-spreading approach provides nearly optimal

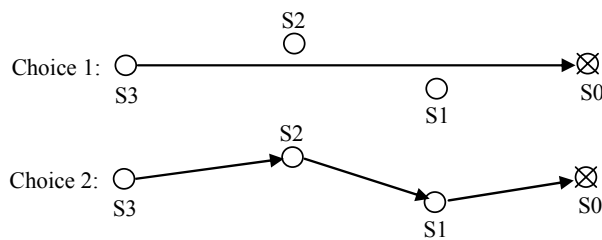
coverage and delivers superior performance in terms of the total distance traveled and the time to converge.

### 3.2.2 On-Demand Repositioning of Sensors

Instead of relocating the nodes at the deployment phase, sensors can be relocated on demand to improve certain performance metrics such as coverage or network lifetime. This can be decided during the network operation based on the changes in either application-level needs or the network state. For instance, the application can be tracking a fast moving target which may require repositioning of some sensor nodes based on the new location of the target. Furthermore, in some applications there can be an increasing number of non-functioning nodes in a particular part of the area, necessitating the redistribution of available sensors. In addition to improving coverage, the energy consumption can be reduced through on-demand relocation of sensors in order to reach the best efficient topology.

The approach presented in [13] performs sensor relocation to counter holes in coverage caused by sensors failure. The idea is simply to identify some spare sensors from different parts of the network that can be repositioned in the vicinity of the faulty nodes. The selection of the most appropriate choice among multiple candidate spare nodes is based on the recovery time and overhead imposed. Both criteria would favor close-by spares over distant ones. Minimizing the recovery time can be particularly crucial for delay sensitive applications. The overhead can be in the form of energy consumption due to the node's travel and due to the message exchange. The latter is especially significant if spares are picked in a distributed manner. In order to detect the closest redundant sensor with low message complexity, a grid-based approach is proposed. The region is divided into cells with a designated head for each cell. Each cell-head advertises/requests redundant nodes for its cell. A quorum-based solution is proposed to detect the intersection of advertisements and requests within the grid. Once the redundant sensor is located, it is moved to the desired cell without disrupting the data traffic or affecting the network topology.

Since moving a node over a relatively long distance can drain a significant amount of energy, a cascaded movement is proposed. The idea is to determine intermediate sensor nodes on the path and replace those nodes gradually. That is, the redundant sensor will replace the first sensor node on the path. That node also is now redundant and can move to replace the second sensor node, and so on. For the example shown in Figure 17, which is redrawn from [13], rather than directly moving  $S_3$  to the location of  $S_0$ , in choice 2 all sensors  $S_3$ ,  $S_2$ , and  $S_1$  move at the same time and replace  $S_2$ ,  $S_1$  and  $S_0$  respectively in order to minimize the relocation time. The path is selected such that it will minimize the total mechanical movement energy and at the same time maximize the remaining energy of sensor nodes. In order to determine such a path, Dijkstra's least cost path algorithm is used. The overall solution is also revisited to provide a distributed approach for determining the best cascading schedule.

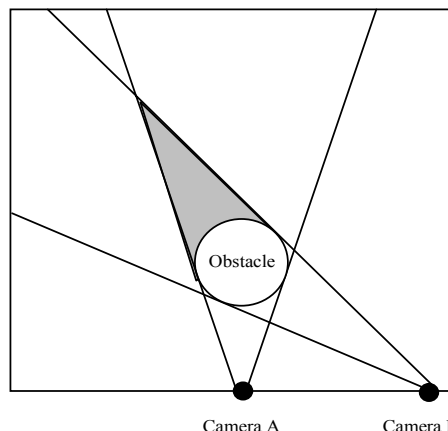


**Figure 17:** Cascaded Movement of Sensors;  $S_3$  replaces  $S_2$ ,  $S_2$  settles in  $S_1$ 's position and  $S_1$  move to where  $S_0$  is.

When validated, the approach has outperformed VOR of [70] with respect to the number of sensors involved in the relocation, the total consumed energy and the total remaining energy. In

addition, cascaded movement has delivered much better performance in terms of relocation time, energy cost and remaining energy than direct movement. However, obviously the cost of maintaining a grid, selection of cell-heads and identifying redundant nodes will grow dramatically as the number of nodes increases. For scalability, a hierarchical solution might be needed to restrict the size of the region and the cost of movement.

Coverage improvement is also the objective of relocating imaging sensors in [71]. Stationary cameras may not provide the desired coverage when there are environmental peculiarities, e.g., moving obstacles, in the event area as seen in Figure 18. Thus, moving the cameras in order to avoid obstacles that block their vision would increase the coverage. The mobility for the camera nodes is made possible through providing a traction mechanism under each camera that will enable motion in one dimension, as is also implemented in Robomote [68]. This mobility feature on cameras is actuated when the coverage of the monitored areas falls below a certain ratio. The experiments performed with real-life target tracking applications have verified that the mobile cameras can increase the coverage of the monitored area and thus decrease the target miss-ratio significantly when compared to stationary-node based setups.



**Figure 18:** An obstacle reduces the coverage of two cameras directed at different areas.

Finally, we would like to note that the work of Dasgupta et al. [41], which we discussed in Section 2.2, can potentially fit under the category of dynamic positioning if sensors are relocated after deployment. However, the repositioning is done at the network planning stage to form the most efficient topology and thus we categorized the scheme as static.

### 3.3. Repositioning Data Collectors

As discussed earlier, sensor data is gathered at either the base-station or CHs for aggregation and possibly additional processing consistent with the computational capabilities of such data collectors (DCs). Dynamic repositioning of DCs has also been pursued as a means for boosting network performance, dealing with traffic bottlenecks and preventing interruptions in network operation. Unlike sensor repositioning, the goal for relocating DCs is usually not local to the individual node and involves numerous network state parameters. In this section, we focus on approaches that consider a single DC or uncoordinated repositioning of multiple DCs if more than one exist. Section 4 discusses collaborative relocation of nodes, which has not received much attention in the literature and has numerous open research problems.

#### 3.3.1 *Repositioning for Increased Network Longevity*

Although energy-aware multi-hop routing does dynamically adapt to changes in sensor energy and traffic pattern, sensors near a data collector (DC) die quickly as they are the most utilized nodes in the network [40][75]. Consequently, nodes that are further away from the DC are picked as substitute relays, as depicted in Figure 19. However, the amount of energy these nodes spend to communicate with the DC is considerably higher. This effect can spread in a spiral manner, draining the sensors energy and hence shortening the lifetime of the network. To stop such a pattern of energy depletion, the DC is repositioned [76].

The main idea is to move the DC towards the sources of highest traffic. The traffic density ( $P$ ) times the transmission power ( $E_{TR}$ ) is used as a metric for monitoring the network operation and searching for the best DC location. The idea is to track changes in the nodes that act as the closest hop to the DC and the traffic density going through these hops. If the distance between the DC and some of the nodes that are in direct communication is smaller than a threshold value  $\delta$ , the DC will qualify the impact of these nodes on the overall network lifetime by considering the number of packets routed through them. If the total  $P \times E_{TR}$  decreases by more than a certain threshold  $\Delta$ , the DC will consider relocating to a new position. Mathematically, the relocation takes place if:

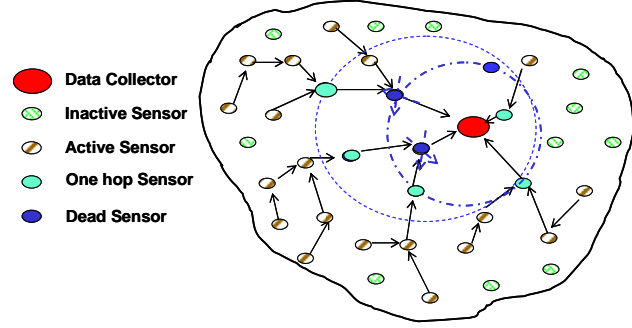
$$\sum_{\forall i \in S_R} E_{(TR_i)} \times P_i - \sum_{\forall j \in S_R^{new}} E_{(TR_j)} \times P_j > \Delta \quad (3)$$

Where:

- $S_R$  is the set of sensors that are one hop away from the data collector on active routes,
- $S_R^{new}$  is the set of sensors that are one hop away from the data collector at a new location,
- $P_i$  is the packet traffic going through node  $i$ , measured as the packet count per frame,
- $E_{(TR_i)}$  is the energy consumed by node  $i$  for transmitting a packet to the next hop, and
- the summations measure the total energy consumed for packet transmission through all the nodes at the old location and new location of the data collector, respectively.

While such positioning will be ideal for high traffic paths, it can worsen the performance on paths with lower traffic density or which are topologically opposite to the direction of the DC's motion. Therefore, before confirming the move, it has been recommended that the DC validates the overall impact on transmission energy by factoring in the possible extension of some data paths and the cost of signaling overhead to those sensor nodes affected by the move. In addition, when the DC starts to move, the data gathering process still continues and thus the routes should be adjusted before the DC gets out of range of some sensors (if any). Unlike the static approaches discussed in Section 2, route adjustment is an issue in dynamic DC positioning. This issue is handled in [76] by either increasing the transmission power or designating additional forwarder sensors. The change in DC position may also introduce shadowing or multi-path fading to some links. Slow or gradual advance towards the new position is shown to be effective in avoiding unexpected link failures that may cause negative performance impacts, since it allows the DC to rethink the suitability of the newly-selected position and/or the decision to move further.

Through simulation, this approach for relocating the DC is shown to not only increase the network longevity but also to enhance other performance metrics like latency and throughput. First, communication-related energy consumption and the average energy per packet are reduced as the DC moves closer to the area where more nodes are collecting data. In addition, nodes



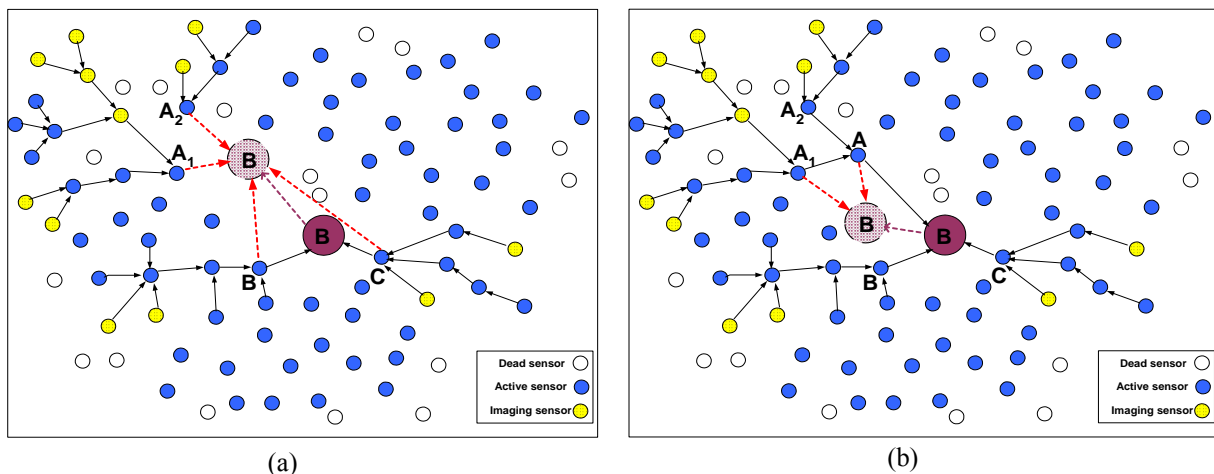
**Figure 19:** Nodes close to the data collector die rather quickly due to traffic overload; forcing the more distant nodes to relay the data to the data collector

collecting the most data are closer to the DC and fewer hops are involved, lowering the overall latency time for data collection. Moreover, the packet throughput grows since most messages pass through fewer hops and travel shorter distances, making them less likely to be dropped.

### 3.3.2 Enhancing Timeliness of Delay-Constrained Traffic

In addition to boosting network longevity, repositioning the DC is useful when real-time traffic with certain end-to-end delay requirements is involved [77]. For instance, when routes to the DC get congested, most requests for establishing paths for real-time data may be denied or the deadline miss rate of real-time packets may increase significantly. Traffic congestion can be caused by an increase in the number of real-time data packets coming from nodes close to a recent event. In such circumstances, it may be infeasible to meet the requirements for real-time data delivery. Therefore, repositioning the DC has been recommended in order to spread the traffic on additional hops and increase the feasibility of meeting the timeliness requirements.

A trigger for such relocation can be an unacceptable increase in the miss rate of real-time packets, or just a desire to increase timeliness even if the miss rate is at a level that is tolerable by the application. To boost timeliness, the DC is moved to the location of, or close to, the most heavily loaded node [77]. The rationale is to split the incoming traffic passing through that node without extending the delay experienced by real-time packets over other routes, as shown in Figure 20. Such loaded nodes are picked based on the real-time traffic service rate, often determined during route setup in order to allocate bandwidth to both real-time and non-real-time traffic. Again, the advantages of the relocation have to be qualified to make sure that the overhead is justified. It is also worth noting that in this approach the impact on link quality is not a major concern when the DC moves close to a heavily loaded node. This is because it is unlikely that the node is experiencing a disruptive level of interference while being able to relay a high volume of real-time packets.



**Figure 20:** (a) The DC (denoted as B) is relocated to the location of A if the delay is not extended on the path from C to B (b) If real-time traffic through C is affected, B is relocated close to C while still splitting traffic at A.

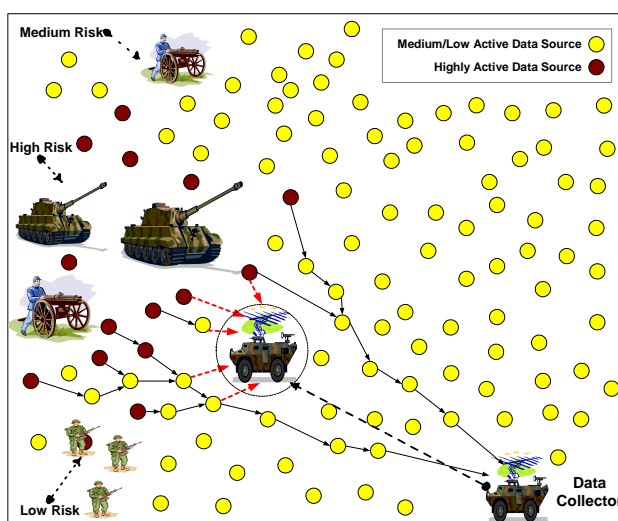
Handling the DC motion is similar to the approach described in Section 3.3.1. As long as the DC remains within the transmission range of all the last-hop nodes, the current routes can be maintained by only adjusting the transmission power. If the new location places the DC out of

the transmission range of some of the last hop nodes in the current routes, new forwarder nodes that are not involved in any routing activity are selected. It was argued that those unused nodes introduce very little queuing delay, which is desirable for on-time delivery of all real-time packets that use these nodes as relays. We note, however, that designating new forwarder nodes is still more costly than just adjusting the transmission power. Therefore, if the new location imposes excessive topology changes, alternative positions which will cause no or minimal topology changes should be considered.

### 3.3.3 Maintaining Uninterrupted Operation

Relocating data collectors has also been pursued to keep WSNs operational without interruption. Noting that some nodes may be isolated (the network may even become partitioned) if one of the DC nodes is damaged or becomes unreachable to sensors, some of the published schemes repositioned DC nodes in order to protect them and sustain network connectivity.

Keeping the DC away from harm is the goal of the GRENN approach proposed in [78]. Two main scenarios motivate GRENN to reposition a DC. First, the DC may be in the way of an approaching serious event, such as a fast spreading fire, or a harmful target like an enemy tank. The second scenario may be caused by a desire to boost network performance by moving the DC node towards the data sources, similar to the schemes discussed in the previous two subsections. The latter scenario, as depicted in Figure 21, may actually expose the DC to hazards and put it at high risk. Thus, handling such scenarios would be subject to performance and safety trade-offs.

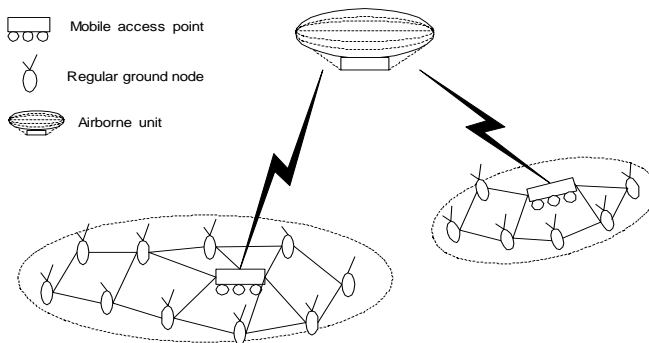


**Figure 21:** Performance-centric relocation may move a data collector dangerously close to serious events in the environment and thus risk the loss of such critical asset.

To assess the safety implication of repositioning the DC, an evolutionary neural networks based formulation is pursued. The idea is to track the DC safety levels at different locations and use this information to define the parameters of the DC safety model. Then the threat implication is estimated as a function of the proximity to reported events and the severity of those events. If the location of the event is not accurately known, the data volume related to that event and the location of the reporting sensors are factored in. An objective function is then formed to balance safety and performance goals and used to guide the search for the new location of the DC. GRENN has been further extended in [79] to identify a safe route for the DC to its new position. The same safety assessment model is employed to guide the selection of the travel path. The effect of the DC's motion on the network performance is also factored in by measuring the throughput at various spots the DC will visit along the way. The objective function is extended to not only trade off the safety of the DC and the network performance but also to minimize the travel distance.

Shen et al. [80] have investigated the placement of mobile access points in order to connect nodes in isolated networks through airborne units or satellites. The deployed nodes usually do

not have expensive radios for long haul communication and usually serve limited geographical areas. The limited communication range may result in partitioning the network, leaving some nodes unreachable to some others. To overcome such structural weaknesses in the network, mobile base-stations are employed to interconnect isolated sub-networks through an airborne relay, such as an unattended air vehicle (UAV) or satellite. As depicted in Figure 22, which is redrawn from [80], a mobile base-station acts as an access point for the nodes in its neighborhood. The approach is to place the mobile access point (MAP) at the centroid of the sub-network. Nodes will send messages which get flooded through the sub-network until they reach the MAP. As the MAP receives the packets, it records the paths each took to reach it. From this information the MAP builds a tree modeling the network, with itself at the root. The MAP then moves to the centroid of this tree. The approach shares some similarities to that of [81].



**Figure 22:** Mobile access points are placed to connect isolated sub-networks through an airborne unit so that the end-to-end delay among nodes in different sub-networks is minimized.

Table 3 summarizes the dynamic positioning schemes covered in this section.

**Table 3:** A comparison of dynamic node repositioning schemes

Paper	Node Type	Relocation Trigger	Primary Objective	Secondary Objective	Constraint
[13]	Sensor	Sensor Failures	Coverage of Holes	Travel distance	Convergence Time
[15]	Sensor	Poor Coverage	Coverage	Travel distance	Convergence Time
[41]	Sensor	High Traffic	Network Lifetime	--	Coverage
[70]	Sensor	Poor Coverage	Coverage	Travel distance	Convergence Time
[71]	Sensor	Poor Image Coverage	Image Coverage	--	--
[73]	Sensor	Poor Coverage	Coverage	Travel distance	Convergence Time
[74]	Sensor	Poor Coverage	Coverage	Travel distance	Convergence Time
[76]	Data Collector	High Traffic	Total Power	Throughput	--
[77]	Data Collector	Poor Timeliness	Timeliness	Avg. Delay	--
[78]	Data Collector	Application dependent	Physical Security	Network Lifetime	--
[79]	Data Collector	Application dependent	Physical Security and throughput	Travel distance	--
[80]	Data Collector	Network partitioning	Connectivity	--	--
[81]	Data Collector	Node mobility	Connectivity	--	--

## 4. Open Research Problems

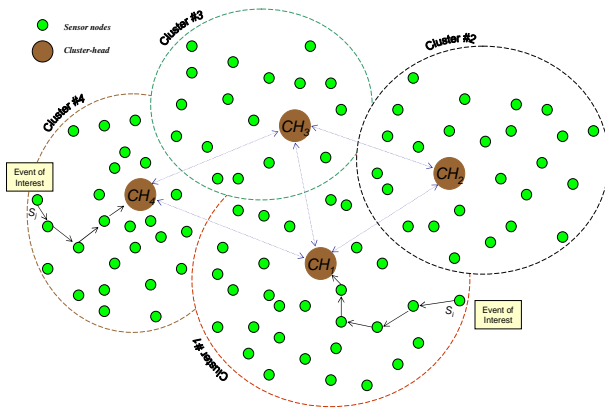
While significant progress has been made in researching the optimization of node positioning in WSNs, many challenging problems remain. In this section, we highlight open research problems, identify the issues involved and report on ongoing work and preliminary results. We categorize open problems into coordinated dynamic placement of multiple nodes and the positioning of sensors in three-dimensional application setups.



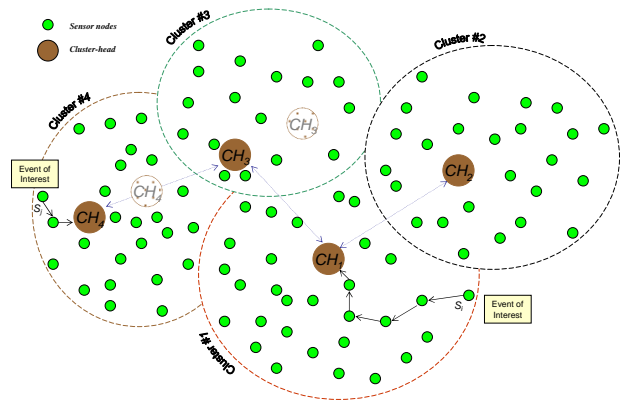
#### 4.1. Coordinated Multi-Node Relocation

In many application setups, nodes coordinate among themselves in order to efficiently and effectively handle application-level requirements. Examples of such applications include robotic-based land-mine detection and deactivation, multi-rover exploration of distance planets, employing a sensor-fleet for oceanic studies, etc. However unlike the case discussed in the previous section, such application-level coordination requires the relocation process to care for inter-node networking issues. For instance, consider the scenario depicted in Figure 23(a). A set of sensors are deployed in an area of interest. The sensors have been partitioned into non-overlapping clusters; each is managed by a distinct cluster-head (CH). To simplify the discussion, let us assume that CHs are the only nodes allowed or capable to relocate.

Given the commonly uneven distribution of sensor nodes or the battery exhaustion or failure of sensors in a certain region, some events would be hard to monitor or would overburden the scarce network resources in the proximity of the event. In Figure 23(a) two sample events articulate such an issue. The depicted events, may be targets or fires, are reported by very few sensors for which the events happen to be within their detection range. Clearly, the intra-cluster network topology for both clusters 1 and 4 are not efficient since the data is routed over many hops and may not be arriving at  $CH_1$  and  $CH_4$  reliably and/or on-time. In addition, some of the nodes that are involved in relaying data may not have abundant energy reserve to keep the path stable for an extended duration. Furthermore,  $CH_1$  and  $CH_4$  may have advanced sensing instruments that can be employed to boost the fidelity of the assessment. The following are some of the challenges that CHs may face when trying to boost the efficiency of the data collection and the operation of the intra-cluster network:



**Figure 23(a):** A sample multi-cluster sensor network architecture, where each cluster is handled/managed by a distinct cluster-head



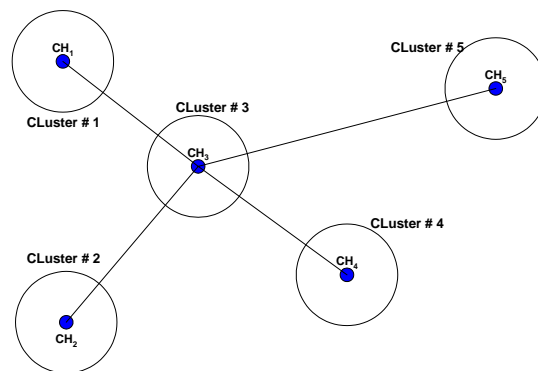
**Figure 23(b):**  $CH_3$  moves to maintain its communication link with  $CH_4$ , probably losing connectivity to  $CH_2$ . However, cluster-heads still form a connected graph.

1.  $CH_1$  may decide to relocate to better serve the event tracked by sensors in its cluster; that is, to monitor the event in a timely manner or to minimize the energy consumption at relaying sensor nodes. However, this repositioning may place  $CH_1$  out of the communication range of  $CH_4$ . Such a move can only be acceptable if  $CH_1$  does not need to interact with  $CH_4$ , which is unlikely, or when an alternative path can be established with an acceptable delay bound. Referring to Figure 23(a), relocating  $CH_1$  would be feasible if the communication path ( $CH_1$ ,  $CH_3$ ,  $CH_4$ ) meets the timeliness requirements.

2. The network must consider the possibility that a different cluster could handle the event. For example, while  $CH_1$  can relocate to better serve the event, the data can also be routed from  $S_j$  to  $CH_2$  over a shorter and more energy efficient path. That would require changing the association of  $S_j$  from cluster 1 to cluster 2, at least temporarily. This modification of cluster membership would surely impose an overhead. Such a trade-off is unavoidable.
3. When relocation of a CH is the only option for boosting the efficiency of the operation within a cluster, a ripple effect may be caused throughout the network. Consider, for example, the relocation of  $CH_4$  close to  $S_j$ . Although such a close proximity to the event would have a positive impact on the operation in cluster 4, moving  $CH_4$  that far makes it unreachable to other CHs. This scenario is not acceptable in a collaborative computing environment. Also,  $CH_4$  may become isolated from the base-station if it cannot directly reach it. A more complex alternative is to relocate multiple CHs in order to maintain connectivity. Figure 23(b) illustrates a possible multi-CH relocation that better serves the event reported by  $S_j$ . Basically,  $CH_3$  gets closer to the new location of  $CH_4$  in order to prevent their communication link from getting broken, possibly at the expense of losing connectivity to  $CH_2$ . This solution could have been more complex if  $CH_1$  did not have a link to  $CH_2$ , forcing it to move in order to be in the range of  $CH_3$  and  $CH_2$ .

Now imagine all nodes can move subject to inter-node topology constraints. Obviously, coordinated multi-node relocation will be even more complex and can introduce lots of trade-offs. In addition, simultaneous relocation of nodes raises the issue of convergence. Repositioning one node can make the move that was concurrently made by another node either redundant and/or incompatible. For example, an autonomous decision by multiple nodes for relocating toward an event may lower the coverage in another region while increasing the fidelity of monitoring the event more than necessary. Lightweight synchronization protocols and/or localized schemes may be required, especially for large networks, in order to ensure convergence of the relocation process to an optimized network state, consistent with the objectives and constraints of the move.

We envision coordinated multi-node relocation to be a promising research direction. Only a little attention has been paid to tackling the challenges of coordinated multi-node relocation. One of the few attempts is reported in [82], where a dynamic multi-node positioning is proposed in order to improve the network longevity. The motion of CHs is restricted to maintain the connectivity of the inter-CH network, as seen in Figure 24. When a CH is to be relocated, its links with its neighbors are checked first before the relocation is performed. If changing the position of a CH will partition the network, its neighbors are to move to restore broken links. For example in Figure 24, if  $CH_3$  is to move toward  $CH_2$ ,  $CH_5$  would follow along to avoid being disconnected from the network. In order to prevent simultaneous relocations, a mutual-exclusion based mechanism is used. The idea is to require exclusive access to a global token in order to



**Figure 24:** The CORE approach strives to maintain a connected inter-CH network by restricting the motion of the individual CH nodes.

perform the relocation. A CH will cease all motion until a token is granted. Each sensor's association to a cluster changes based on the proximity of CHs nodes to the individual sensor.

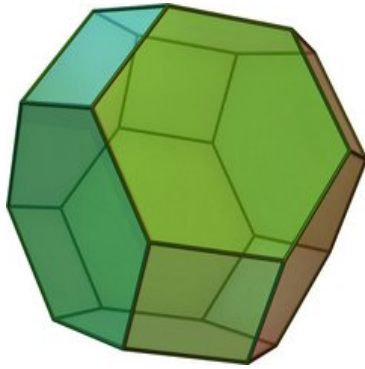
Another notable work is the COCOLA approach [63], which has been discussed in section 2.3. Unlike CORE [82], mutual exclusion is supported in a distributed fashion. The goal is again to maintain connectivity among CH nodes when some of them move. The synchronization protocol is simply to coordinate with immediate neighbors based on a predefined priority system. The priority can be static, e.g., ID-based like CORE, or a dynamic figure that relates to the application or the state of CHs. Assuming distinct node priorities, it was proven that multiple CHs can simultaneously relocate as long as they are not neighbors.

Although not particularly geared toward WSNs, the approach [83] also addresses inter-CH coordination. The authors consider a network of robots that collaborate on application tasks and move on demand. The network is assumed to be 2-connected at startup, i.e., every robot can reach every other robot over at least 2 vertex-disjoint paths. The goal is to sustain this level of connectivity even under link or node failure. The approach is based on relocating some of the robots in order to reestablish the 2-connectivity. The authors modeled the recovery as an optimization formulation in which nodes are to reposition using the minimal total travel distance. Two distributed algorithms were proposed. The first algorithm is called contraction and strives to move robots inward to the centroid of the deployed nodes (robots) in order to boost connectivity. To avoid excessive contraction, a gradual motion is proposed. The second approach employs graph theory to identify cut vertices and orchestrate block movements. The idea is to pick nodes that have a degree of 1 to move towards a cut vertex in order to establish new links that serve as alternatives to the critical links between cut vertices. Although this work does not account for coverage requirements and sensor-base-station connectivity, the approach is worth exploring in the context of WSNs.

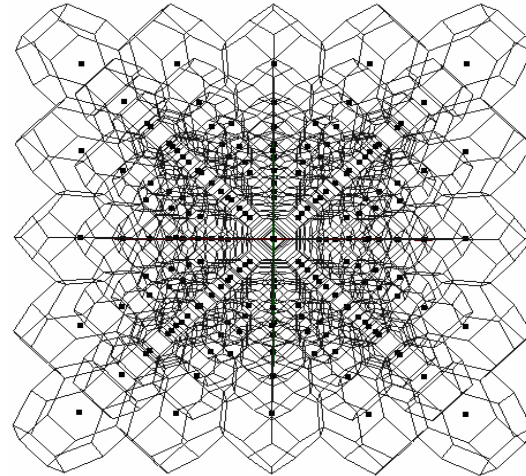
#### **4.2. Node Positioning in Three-Dimensional Application Setups**

The scope of the majority of published papers on static and dynamic node positioning strategies is limited to terrestrial networks, where covered space is two-dimensional (2-D), and to small indoor setups. Even for applications that employ range finders, imagers and video sensors the focus has been on the quality of the collected data and coverage of some 2-D space, basically by managing the angular orientation of the sensors in a plane. As we mentioned in Section 2.1, many of the popular coverage analysis and placement strategies pursued for 2-D space become NP-Hard in 3-D [9][84]. However, with the increased interest in applications of sensor networks in space exploration, airborne and underwater surveillance, oceanic studies, storm tracking, etc. tackling the contemporary design issues such as coverage and connectivity in 3-D has become a necessity. We expect optimization strategies for node positioning in WSN applications in large scale 3-D setups to be one of the hot topics of research in the next few years. In fact, some preliminary research results have started to emerge.

Alam and Haas [85] have investigated the problem of achieving maximal 3-D coverage with the least number of sensors. In the conventional 2-D scenario, sensor coverage is modeled as a circle and the maximal coverage problem is mapped to a circle packing formulation which has a polynomial time solution. Given the high complexity of the sphere packing problem [9], the authors argue that space filling polyhedrons would be more suitable for 3-D applications. The idea is to fill the 3-D application space with the least number of polyhedrons in order to provide maximal coverage, ideally 100%. The metric used for coverage was called a *volumetric quotient*,



**Figure 25:** A truncated octahedron has 14 faces; 8 are hexagonal and 6 are square. The length of the edges of hexagons and squares are the same. Figure is from [86].



**Figure 26:** Applying the truncated octahedron based node placement strategy in a  $20\text{m} \times 20\text{m} \times 20\text{m}$  3D space. Figure is from [85].

which is defined as the ratio of the volume of the shape to be used to that of its enclosing sphere. The paper compares the truncated octahedron, the rhombic dodecahedron, the hexagonal prism, and the cube in terms of volumetric quotient. The conclusion is that truncated octahedrons (Figure 25), created through the use of the Voronoi tessellation of 3-D space (Figure 26), yield the best results. In addition to coverage, the paper studies the relationship between the sensing and transmission ranges so that a connected topology is established. For truncated octahedron it has been concluded that connectivity is ensured if the transmission range of the employed nodes is at least 1.7889 times the sensing range.

Ravelomanana [87] has studied the properties of the network topologies that result from random deployment of nodes in a 3-D region of interest. The main goal is to analyze the implications of the sensing and communication ranges on coverage and connectivity, respectively. Considering a uniform distribution of nodes, the author derives conditions for the node transmission range  $r$  required for achieving a degree of connectivity  $d$ , where every node has at least  $d$  neighbors. In addition, the average path length between two nodes in the network is formulated as a function of  $d$  and  $r$ . The same analysis is performed for coverage, basically estimating the sensing range required to achieve a certain degree of coverage in a region using a predetermined number of nodes. The work strives to provide theoretical bounds that can help in preliminary design and feasibility studies of 3-D WSNs. Pompili, et al. [33], have used these bounds to validate the effectiveness of their random node deployment scheme. The idea is to randomly drop sensors from the ocean surface and control their depth below the surface by wires that attach them to some anchors at the bottom of the ocean. The validation experiments have been very consistent with the theoretical results of [87] by identifying the right combination of network size and sensing range for achieving 100% coverage.

## 4. Conclusion

Wireless sensor networks (WSNs) have attracted lots of attention in recent years due to their potential in many applications such as border protection and combat field surveillance. Given the criticality of such applications, maintaining efficient network operation is a fundamental objective. However, the resource-constrained nature of sensor nodes and the ad-hoc formation of

the network, often coupled with unattended deployment, pose non-conventional challenges and motivate the need for special techniques for designing and managing WSNs. In this paper, we have discussed the effect of node placement strategies on the operation and performance of WSNs. We categorized the various approaches in the literature on node positioning into static and dynamic. In static approaches, optimized node placement is pursued in order to achieve some desired properties for the network topology and coverage. On the other hand, dynamic repositioning of nodes after deployment or during the normal network operation can be a viable means for boosting the performance. Unlike the initial careful placement, node repositioning can assist in dealing with dynamic variations in network resources and the surrounding environment. We have identified the technical issues pertaining to relocating the nodes; namely when to reposition a node, where to move it and how to manage the network while the node is in motion. We have surveyed published techniques for node positioning and compared them according to their objectives, methodologies and applications.

In our opinion, static strategies are more practical when a deterministic node placement is feasible and when the cost of nodes is not an issue. For example, when populating a sensor network to monitor an oil pipeline or to conduct security surveillance, pursuing a static strategy would yield the best coverage and ensure connectivity using the fewest number of sensors. When random distribution of nodes is the only option, a static strategy is not a recommended choice unless the cost of the individual sensors is so insignificant that a very large population of sensors can be employed without concern. On other hand, the sole use of mobile nodes would not be practical due to the increased cost and management overhead. Instead, we envision that a mix of static and dynamic schemes would be the most effective and efficient placement strategy for large-scale and mission-critical WSN applications. Low to medium density of stationary nodes can be employed in the monitored region through random distribution. In addition, significantly fewer movable nodes should be added. Such movable nodes can be dynamically relocated to fill coverage gaps, deal with changes in user interests, or recover from connectivity problems.

We have identified the coordinated multi-node repositioning problem as an open research area. Little attention has been paid to such a challenging and interesting area. We also envision that the node placement problem in 3-D will need an increased attention from the research community in order to tackle practical deployment scenarios. As we indicated, most of the published work considered 2-D regions, and numerous issues are not tackled in the 3-D space. Applications like video and underwater surveillance raise a number of interesting node placement challenges. Finally, we expect domain and node specific positioning to gain increased attention with the growing list of WSN applications and the availability of sophisticated models of the capabilities of sensor nodes.

## References

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "Wireless sensor networks: a survey", *Computer Networks*, Vol. 38, pp. 393-422, 2002.
- [2] C-Y. Chong and S.P. Kumar, "Sensor networks: Evolution, opportunities, and challenges," *Proceedings of the IEEE*, Vol. 91, No. 8, pp. 1247- 1256, 2003.
- [3] K. Akkaya and M. Younis, "A Survey on Routing Protocols for Wireless Sensor Networks," *Elsevier Journal of Ad Hoc Networks*, Vol. 3, No. 3, pp. 325-349, May 2005.
- [4] P. Naik and K. M. Sivalingam, "A survey of MAC protocols for sensor networks," In *Wireless Sensor Networks*, C. S. Raghavendra, K. M. Sivalingam, and T. Znati, Eds. Kluwer Academic Publishers, Norwell, MA, pp. 93-107, 2005.

- [5] N. Sadagopan, B. Krishnamachari, and A. Helmy, "Active Query Forwarding in Sensor Networks (ACQUIRE)," *Journal of Ad Hoc Networks*, Vol. 3, No. 1, pp.91-113, January 2005.
- [6] A. Cerpa and D. Estrin, "ASCENT: Adaptive Self-Configuring Sensor Networks Topologies," in the *Proceedings of the 21<sup>st</sup> International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'02)*, New York, NY, June 2002.
- [7] A. Efrat, S. Har-Peled, J. S. B. Mitchell, "Approximation Algorithms for Two Optimal Location Problems in Sensor Networks" in the *Proceedings of the 3<sup>rd</sup> International Conference on Broadband Communications, Networks and Systems (Broadnets'05)*, Boston, Massachusetts, October 2005.
- [8] X. Cheng, DZ Du, L. Wang and B. Xu, "Relay sensor placement in wireless sensor networks," *ACM/Springer Journal of Wireless Networks* (to appear).
- [9] S. Poduri, S. Patten, B. Krishnamachari, and G. S. Sukhatme, "Sensor Network Configuration and the Curse of Dimensionality," in the *Proceedings of the 3<sup>rd</sup> IEEE Workshop on Embedded Networked Sensors*, Cambridge, MA, May 2006.
- [10] J. Pan, L. Cai, Y. T. Hou, Y. Shi, and S. X. Shen, "Optimal Base-Station Locations in Two-Tiered Wireless Sensor Networks" *IEEE Transactions on Mobile Computing*, Vol. 4, No. 5, pp. 458 – 473, September 2005.
- [11] S. S. Dhillon and K. Chakrabarty, "Sensor placement for effective coverage and surveillance in distributed sensor networks," in the *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC'03)*, New Orleans, LA, March 2003.
- [12] T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan and K. K. Saluja, "Sensor deployment strategy for target detection," In the *Proceedings of the 1<sup>st</sup> ACM international Workshop on Wireless Sensor Networks and Applications (WSNA '02)*, Atlanta, Georgia, September 2002.
- [13] K. Akkaya, M. Younis and M. Bangad "Sink Repositioning for Enhanced Performance in Wireless Sensor Networks," *Computer Networks*, Vol. 49, pp. 512-434, 2005.
- [14] G. Wang, G. Cao, T. La Porta, and W. Zhang, "Sensor Relocation in Mobile Sensor Networks," in the *Proceedings of the 24<sup>th</sup> International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM05)*, Miami, FL, March 2005.
- [15] J. Wu and S. Yang, "SMART: A Scan-based Movement Assisted Sensor Deployment Method in Wireless Sensor Networks," in the *Proceedings of the 24<sup>th</sup> International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'05)*, Miami, FL, March 2005.
- [16] A. Brooks, A. Makarenko, T. Kaupp, S. Williams, and H. Durrant-Whyte, "Implementation of an indoor active sensor network," in the *Proceedings of the 9<sup>th</sup> International Symposium on Experimental Robotics*, Singapore, June 2004.
- [17] V. A. Petrushin, G. Wei, O. Shakil, D. Roqueiro and V. Gershman, "Multiple-Sensor Indoor Surveillance System," in the *Proceedings of the 3<sup>rd</sup> Canadian Conference on Computer and Robot Vision (CRV'06)*, Québec city, June 2006.
- [18] L. Krishnamurthy et al., "Design and Deployment of Industrial Sensor Networks: Experiences from a Semiconductor Plant and the North Sea," in the *Proceedings of the 3<sup>rd</sup> ACM Conference on Embedded Networked Sensor Systems (SenSys'05)*, San Diego, CA, November 2005.
- [19] J. Paek, K. Chintalapudi, J. Cafferey, R. Govindan and Sami Masri, "A Wireless Sensor Network for Structural Health Monitoring: Performance and Experience," in the *Proceedings of the Second IEEE Workshop on Embedded Networked Sensors (EmNetS-II)*, Sydney, Australia, May 2005.
- [20] K. Mechtov, W. Kim, G. Agha, and T. Nagayama, "High-Frequency Distributed Sensing for Structure Monitoring," in the *Proceedings of the First International Conference on Networked Sensing Systems*, Tokyo, Japan, June 2004.
- [21] J. Berry, L. Fleischer, W.E. Hart, and C.A. Phillips, "Sensor placement in municipal water networks," in the *Proceedings of the World Water and Environmental Resources Conference*, Philadelphia, Pennsylvania, June 2003.
- [22] J.-P. Watson, H. Greenberg, W.E. Hart, "A multiple-objective analysis of sensor placement optimization in water networks," in the *Proceedings of the World Water and Environment Resources Congress*, Salt Lake City, June 2004.
- [23] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater Acoustic Sensor Networks: Research Challenges," *Journal of Ad Hoc Networks*, Vol. 3, No. 3, pp. 257-279, March 2005.
- [24] H.H. Gonzalez-Banos and J.C. Latombe "A Randomized Art-Gallery Algorithm for Sensor Placement," in the *Proceedings of the 17<sup>th</sup> ACM Symposium on Computational Geometry (SoCG'01)*, Boston, MA, June 2001.

- [25] L. Navarro, J. Dolan, and P. Khosla, "Optimal Sensor Placement for Cooperative Distributed Vision," in the *Proceedings of the International Conference on Robotics and Automation (ICRA'04)*, New Orleans, LA, April 2004.
- [26] J. Tang, B. Hao, and A. Sen, "Relay Node Placement in Large Scale Wireless Sensor Networks", *Computer Communications*, special issue on wireless sensor networks, Vol. 29. pp. 490–501, 2006.
- [27] M. Ishizuka and M. Aida, "Performance Study of Node Placement in Sensor Networks," In the *Proceedings of the 24<sup>th</sup> International Conference on Distributed Computing Systems Workshops - W7: EC (Icdcs'04) - Volume 7*, Hachioji, Tokyo, Japan, March 2004.
- [28] K. Xu, H. Hassanein, G. Takahara and W. Wang, "Relay node deployment strategies in heterogeneous wireless sensor networks: single-hop communication case," in the *Proceedings of the IEEE Global Telecommunication Conference (Globecom'05)*, St. Louis, MO, November 2005.
- [29] K. Xu, H. Hassanein, G. Takahara and Q. Wang, "Relay node deployment strategies in heterogeneous wireless sensor networks: multiple-hop communication case", the *Proceedings of 2<sup>nd</sup> IEEE Conference on Sensor and Ad Hoc Communications and Networks (SECON'05)*, Santa Clara, CA, September 2005.
- [30] C.-F. Huang and Y.-C. Tseng, "The Coverage Problem in a Wireless Sensor Network," in the *Proceedings of the ACM 9<sup>th</sup> Annual International Conference on Mobile Computing and Networking (MobiCom'03)*, San Diego, CA, September 2003.
- [31] A. Boukerche, X. Fei and R. B. Araujo, "A Coverage Preserving and Fault Tolerant Based Scheme for Irregular Sensing Range in Wireless Sensor Networks," in the *Proceedings of the 49<sup>th</sup> Annual IEEE Global Communication Conference (Globecom'06)*, San Francisco, CA, November 2006.
- [32] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava, "Coverage problems in wireless ad-hoc sensor networks," in the *Proceedings of the 20<sup>th</sup> International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'01)*, Anchorage, Alaska, April 2001.
- [33] D. Pompili, T. Melodia, and I. F. Akyildiz, "Deployment Analysis in Underwater Acoustic Wireless Sensor Networks," in the *Proceedings of the ACM International Workshop on UnderWater Networks (WUWNet)*, Los Angeles, CA, September 2006.
- [34] E. S. Biagioni and G. Sasaki, "Wireless Sensor Placement for Reliable and Efficient Data Collection," In the *Proceedings of the 36<sup>th</sup> Annual Hawaii international Conference on System Sciences (HICSS'03) - Track 5 - Volume 5*, Big Island, Hawaii, January 2003.
- [35] K. Kar and S. Banerjee, "Node placement for connected coverage in sensor networks," in the *Proceedings of the Workshop on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt'03)*, Sophia Antipolis, France, 2003.
- [36] J. Bredin, E. Demaine, M. Taghi Hajiaghayi, and D. Rus, "Deploying Sensor Networks with Guaranteed Capacity and Fault Tolerance," in the *Proceedings of the 6<sup>th</sup> ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC'05)*, Urbana-Champaign, Illinois, May 2005.
- [37] P. Jacquet, "Geometry of information propagation in massively dense ad hoc networks," in the *Proceedings of the 5<sup>th</sup> ACM Symposium on Mobile Ad-Hoc Networking and Computing (MobiHOC'04)*, Roppongi Hills, Tokyo, Japan, May 2004.
- [38] S. Toumpis and L. Tassiulas, "Packetostatics: Deployment of Massively Dense Sensor Networks as an Electrostatics Problem, in the in the *Proceedings of the 24<sup>th</sup> IEEE Conference on Computer Communications and Networking (INFOCOM'05)*, Miami, FL, March 2005.
- [39] S. Toumpis and G. A. Gupta, "Optimal Placement of Nodes in Large Sensor Networks under a General Physical Layer Model," in the *Proceedings of 2<sup>nd</sup> IEEE Conference on Sensor and Ad Hoc Communications and Networks (SECON'05)*, Santa Clara, CA, September 2005.
- [40] M. Younis, M. Youssef and K. Arisha, "Energy-Aware management in Cluster-Based Sensor Networks," *Computer Networks*, Vol. 43, No. 5, pp. 649-668, December 2003.
- [41] K. Dasgupta, M. Kukreja, K. Kalpakis, "Topology-aware placement and role assignment for energy-ecient information gathering in sensor networks," in the *Proceedings of the 8<sup>th</sup> IEEE Symposium on Computers and Communication (ISCC'03)*, Kemer-Antalya, Turkey, June 2003.
- [42] Y. Chen, C. Chuan, and Q. Zhao, "Sensor Placement for Maximizing Lifetime per Unit Cost in Wireless Sensor Networks," in the *Proceedings of the IEEE Military Communication Conference (MILCOM'05)*, Atlantic City, NJ, October, 2005.
- [43] P. Cheng, C.-N. Chuah and Xin Liu, "Energy-aware Node Placement in Wireless Sensor Networks," in the *Proceedings of the 47<sup>th</sup> IEEE Global Telecommunications Conference (Globecom'04)*, Dallas, TX, November 2004.

- [44] X. Zhang and S. B. Wicker, "How to distribute sensors in a random field?," in the *Proceedings of the 3<sup>rd</sup> International Symposium on Information Processing in Sensor Networks (IPSN '04)*, Berkeley, CA, April 2004.
- [45] D. Ganesan, R. Cristescu and B. Beferull-Lozano, "Power-efficient sensor placement and transmission structure for data gathering under distortion constraints," in the *Proceedings of the 3<sup>rd</sup> International Symposium on Information Processing in Sensor Networks (IPSN '04)*, Berkeley, CA, April 2004.
- [46] Q. Wang, K. Xu, G. Takahara and H. Hassanein, "Deployment for Information Oriented Sensing Coverage in Wireless Sensor Networks", in the *Proceedings of the 49<sup>th</sup> IEEE Global Telecommunication Conference (Globecom'06)*, San Francisco, CA, November 2006.
- [47] D. Cerpa and Estrin, "ASCENT: adaptive self-configuring sensor networks topologies," in the *Proceedings of the 21<sup>st</sup> International Annual Joint Conference of the IEEE Computer and Communications Societies ((INFOCOM'02)*, New York, NY, June 2002
- [48] B. Chen, et al., "Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," in the *Proceedings of the ACM 7<sup>th</sup> Annual International Conference on Mobile Computing and Networking (MobiCom'01)*, Rome, Italy, July 2001.
- [49] Y. Xu, J. Heidemann and D. Estrin, "Geography-informed energy conservation for ad hoc routing, in the *Proceedings of the ACM 7<sup>th</sup> Annual International Conference on Mobile Computing and Networking (MobiCom'01)*, Rome, Italy, July 2001.
- [50] Y. T. Hou, Y. Shi, and H. D. Sherali, "On Energy Provisioning and Relay Node Placement for Wireless Sensor Networks," *IEEE Transactions on Wireless Communications*, Vol. 4, Mo. 5, pp. 2579-2590, September 2005.
- [51] K. Xu, Q. Wang, H. S. Hassanein and G. Takahara, "Optimal Wireless Sensor Networks (WSNs) Deployment: Minimum Cost with Lifetime Constraint," in the *Proceedings of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob'05)*, Montreal, Canada, August 2005.
- [52] Q. Wang, G. Takahara, H. Hassanein and K. Xu, "On Relay Node Placement and Locally Optimal Traffic Allocation in Heterogeneous Wireless Sensor Networks", in the *Proceedings of the IEEE Conference on Local Computer Networks (LCN'05)*, Sydney, Australia, November 2005.
- [53] Q. Wang, K. Xu, H. Hassanein, G. Takahara, "Minimum Cost Guaranteed Lifetime Design for Heterogeneous Wireless Sensor Networks (WSNs)", in the *Proceedings of the 24<sup>th</sup> IEEE International Performance, Computing, and Communications Conference (IPCCC'05)*, Phoenix, Arizona, April 2005.
- [54] Q. Wang, K. Xu, G. Takahara, H. Hassanein, "Locally Optimal Relay Node Placement in Heterogeneous Wireless Sensor Networks", in the *Proceedings of the 48<sup>th</sup> Annual IEEE Global Telecommunications Conference (Globecom'05)*, St. Louis, Missouri, November 2005.
- [55] B. Hao, H. Tang and G. Xue, "Fault-tolerant relay node placement in wireless sensor networks: formulation and approximation," in the *Proceeding of the Workshop on High Performance Switching and Routing (HPSR'04)*, Phoenix, Arizona, April 2004.
- [56] E. L. Lloyd and G. Xue, "Relay Node Placement in Wireless Sensor Networks," *IEEE Transactions on Computers* (To appear).
- [57] X. Han, X. Cao, E. L. Lloyd and C.-C. Shen, "Fault-tolerant Relay Nodes Placement in Heterogeneous Wireless Sensor Networks," in the *Proceeding of the 26<sup>th</sup> IEEE/ACM Joint Conference on Computers and Communications (INFOCOM'07)*, Anchorage AK, May 2007 (To appear).
- [58] A. Abbasi and M. Younis, "A Survey on Clustering Algorithms for Wireless Sensor Networks," *Journal of Computer Communications*, Special Issue on Network Coverage and Routing Schemes for Wireless Sensor Networks (to appear).
- [59] A. Bogdanov E. Maneva, S. Riesenfeld, "Power-aware Base Station Positioning for Sensor Networks", in the *Proceedings of the 23<sup>rd</sup> International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'04)*, Hong Kong, March 2004.
- [60] E. I. Oyman and C. Ersoy, "Multiple Sink Network Design Problem in Large Scale Wireless Sensor Networks", in the *Proceedings of the IEEE International Conference on Communications (ICC'04)*, Paris, June 2004.
- [61] W. Youssef and M. Younis, "Intelligent Gateways Placement for Reduced Data Latency in Wireless Sensor Networks," in the *Proceedings of the IEEE International Conference on Communications (ICC'07)*, Glasgow, Scotland, June 2007 (to appear).
- [62] K. Akkaya and M. Younis, "COLA: A Coverage and Latency aware Actor Placement for Wireless Sensor and Actor Networks", in the *Proceedings of IEEE Vehicular Technology Conference (VTC-Fall'06)*, Montreal, Canada, September 2006.
- [63] K. Akkaya and M. Younis, "Coverage and Latency aware Actor Placement for Wireless Sensor and Actor Networks", *International Journal of Sensor Networks* (to appear).



- [64] Y. Shi, Y. T. Hou, and A. Efrat, "Algorithm Design for Base Station Placement Problems in Sensor Networks," in the *Proceedings of IEEE International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks*, Waterloo, Ontario, Canada, August 2006.
- [65] S. R. Gandham, M. Dawande, R. Prakash, and S. Venkatesan, "Energy efficient schemes for wireless sensor networks with multiple mobile base stations," in *Proceedings of the IEEE Globecom'03*, San Francisco, CA, December 2003.
- [66] J. Pan, et. al, "Locating Base-Stations for Video Sensor Networks" in the *Proceedings of the IEEE Vehicular Technology Conference (VTC-Fall'03)*, Orlando, FL, October, 2003.
- [67] T. W. Haynes, S. T. Hedetniemi, and P. J. Slater. *Domination in Graphs: Advanced Topics*. Marcel Dekker, Inc. New York, 1998.
- [68] <http://www-robotics.usc.edu/~robomote/>
- [69] <http://www.symbio.jst.go.jp/symbio/morph.html>
- [70] G. Wang, G. Cao, and T. La Porta, "Movement-Assisted Sensor Deployment," in the *Proceedings of the 23<sup>rd</sup> International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'04)*, Hong Kong, March 2004.
- [71] A. Kansal, E. Yuen, W.J. Kaiser, G.J. Pottie and M.B. Srivastava., "Sensing Uncertainty Reduction Using Low Complexity Actuation," in the *Proceedings of Information Processing in Sensor Networks (IPSN 2004)*, Berkeley, CA, April 2004.
- [72] A. Kansal et al., "Controlled Mobility for Sustainable Wireless Sensor Networks," in the *Proceedings of IEEE Sensor and Ad Hoc Communications and Networks (SECON'04)*, Santa Clara, CA, October 2004.
- [73] G. Wang, G. Cao and T. La Porta. "Proxy-Based Sensor Deployment for Mobile Sensor Networks," in the *Proceedings of the 1<sup>st</sup> IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS'04)*, Fort Lauderdale, Florida, October 2004.
- [74] N. Heo and P. K. Varshney, "Energy-Efficient Deployment of Intelligent Mobile Sensor Networks," *IEEE Transactions on Systems, Man, Cybernetics*, Part A, Vol. 35, No. 1, pp. 78-92, January 2005.
- [75] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless sensor networks," in the *Proceeding of the Hawaii International Conference System Sciences (HICSS'00)*, Hawaii, January 2000.
- [76] M. Younis, M. Bangad and K. Akkaya, "Base-Station Repositioning For Optimized Performance of Sensor Networks," in the *Proceedings of the IEEE Vehicular Technology Conference (VTC-Fall'03)*, Orlando, Florida, October 2003.
- [77] K. Akkaya and M. Younis, "Relocation of Gateway for Enhanced Timeliness in Wireless Sensor Networks," in the *Proceedings of the IEEE Workshop on Energy-Efficient Wireless Communications and Networks (EWCN 2004)*, in conjunction with the 23<sup>rd</sup> IEEE International Performance Computing and Communications Conference (IPCCC'04), Phoenix, Arizona, April 2004.
- [78] W. Youssef, M. Younis and K. Akkaya "An Intelligent Safety-Aware Gateway Relocation Scheme for Wireless Sensor Networks," in the *Proceedings of the IEEE International Conference on Communications (ICC'06)*, Istanbul, Turkey, June 2006.
- [79] W. Youssef, M. Younis and M. Eltoweissy "Intelligent Discovery of Safe Paths in Wireless Sensor Network," in the *Proceedings of the 23<sup>rd</sup> IEEE Biennial Symposium on Communication (QBSC'06)*, Queen's University, Kingston, Ontario, Canada, May 2006.
- [80] C.-C. Shen, O.Koc, C. Jaikao, and Z. Huang, "Trajectory Control of Mobile Access Points in MANET," in the *Proceedings of the 48<sup>th</sup> annual IEEE Global Telecommunications Conference (GLOBECOM '05)*, St. Louis, MO, November 2005.
- [81] M. Ahmed et al., "Positioning Range Extension Gateways in Mobile Ad Hoc Wireless Networks to Improve Connectivity and Throughput," in the *Proceedings of IEEE Military Communications Conference (MILCOM'01)*, Washington, D.C., October 2001.
- [82] J. English, M. Wiacek, M. Younis, "CORE: Coordinated Relocation of Sink Nodes in Wireless Sensor Networks," In the *Proceedings of the 23<sup>rd</sup> IEEE Biennial Symposium on Communications (QBSC'06)*, Kingston, Ontario, Canada, May 2006.
- [83] P. Basu and J. Redi, "Movement Control Algorithms for Realization of Fault-Tolerant Ad Hoc Robot Networks," *IEEE Networks*, pp. 2-10, July/August 2004.
- [84] I. F. Akyildiz, T. Melodia, K. R. Chowdhury, "A Survey on Wireless Multimedia Sensor Networks," *Computer Networks*, Vol. 51, No. 4, pp. 921-960, March 2007.

- [85] S. N. Alam and Z. Haas, "Coverage and Connectivity in Three-Dimensional Networks", in the *Proceedings of the 12<sup>th</sup> Annual ACM International Conference on Mobile Computing and Networking (MobiCom'06)*, Los Angeles, CA, September 2006.
- [86] [http://en.wikipedia.org/wiki/Truncated\\_octahedron](http://en.wikipedia.org/wiki/Truncated_octahedron)
- [87] V. Ravelomanana. "Extremal Properties of Three-dimensional Sensor Networks with Applications," *IEEE Transactions on Mobile Computing*, Vol. 3, No. 3, pp. 246–257, July/September 2004.