Node Deployment Optimization in Wireless Sensor Networks

Zhenhuan Wang
Gulliver Preparatory School

Abstract. Recently, wireless sensor networks (WSNs) which consist of a group of cheap but energy-limited nodes with functionalities of sensing, computation, and communication, are widely applied to detect or monitor the events in interest. One of the fundamental design issues in WSNs is how to deploy or relocate the nodes to optimize the desired sensing performance. This survey focuses on a variety of sensing performance models and movement-assisted node deployment approaches that have been studied and proposed in the literature and highlights their strengths and limitations. On one hand, this tutorial proposes an exhaustive classification of existing movement-assisted self-deployment algorithms. On the other hand, some physical constraints and their impacts, such as limited-battery and obstacles are introduced in this tutorial. Then, the detailed comparisons between the existing algorithms with respect to different scenarios are well performed in this tutorial. Therefore, this paper provides not only a view of the state-of-the-art but also the insights for selecting the appropriate self-deployment algorithms for different sensing scenarios. Furthermore, the open problems and challenges in the area of node deployment are summarized at the end of this tutorial.

Keywords: wireless sensor networks, energy consumption, movement-assisted algorithms, environment, materials

1. Introduction
Three important problems (sensing performance, energy conservation, and connectivity) are receiving increased consideration [1]-[3]. Sensing performance or quality of service (QoS) focuses on how well the sensor nodes collect information from the physical space. As the most common sensing performance measurement, coverage represents the percentage of points can be covered by at least one node [4]-[6]. When sensors are densely deployed into the target region, full-coverage, i.e., coverage = 1, is guaranteed. In this case, the coverage degree will replace coverage to evaluate the sensing performance. Coverage degree is k if and only if every target point is covered by at least k nodes. However, both coverage and coverage degree are based on an

2. Sensing performance and energy consumption models
Notations: Let \( N \) be the number of sensor nodes, \( \mathbb{R} \) be the set of real numbers. We denote the sensor node locations by \( P = \{ p_1, p_2, \cdots, p_N \} \), and target region by \( \Omega \subset \mathbb{R}^d \). Let \( \omega \in \Omega \) be a target point. \( \| \| \) denotes the Euclidean distance. \( C(\cdot) \) and \( D(\cdot) \) represent the sensing accuracy and sensing uncertainty (or distortion), respectively.
2.1. Sensing Performance

As sensor nodes have different physical characteristics, numerous sensing models have been proposed and applied to measure the sensing performance. Nonetheless, the existing models follow a common rule: sensing accuracy decays as the sensing distance increases [1]. Therefore, a natural choice is directly modeling the point-to-point sensing accuracy as a non-increasing function of the distance. Instead of increasing the sensing accuracy, other researchers make an effort to minimize a negative sensing performance, i.e., the sensing uncertainty, from a quantization prospective [2]. In what follows, we analyze the two types of sensing performance in detail.

A widely used sensing accuracy model is the binary disc/coverage model. Each sensor in this model can perfectly detect the points within its sensing radius while missing anything outside the range. The corresponding point-to-point sensing accuracy is thus a step function

\[ I_{[s]}(\|p_n - \omega\|) = \begin{cases} 1, & \|p_n - \omega\| < R_s \\ 0, & \|p_n - \omega\| \geq R_s \end{cases} \quad (1) \]

Where \( R_s \) is the sensing range. Taking the whole target region into consideration, the overall sensing accuracy is written as

\[ C(P) = \sum_{n=1}^{N} \int_{\Omega} I_{[s]} \left( \min_{\omega \in \Omega} (\|p_n - \omega\|) \right) d\omega \quad (2) \]

This model can be easily extended to heterogeneous WSNs with the sensing accuracy defined by

\[ C(P) = \sum_{n=1}^{N} \int_{\Omega} I_{[s,n]} \left( \min_{\omega \in \Omega} (\|p_n - \omega\|) \right) d\omega \quad (3) \]

Where \( R_{s,n} \) represents Node \( n \)'s sensing range. The literature [3] also considers the difference among the target points in \( \Omega \) by adding a probability density function, \( f(\cdot) : \mathbb{R}^2 \rightarrow \mathbb{R}^+ \),

\[ C(P) = \sum_{n=1}^{N} \int_{\Omega} I_{[s,n]} \left( \min_{\omega \in \Omega} (\|p_n - \omega\|) \right) f(\omega) d\omega \quad (4) \]

The value of Eq. (4) is, in general, interpreted as the ratio of points covered by at least one sensor, i.e., area coverage.

In the above formulations, the target region is a continuous two-dimensional region. However, in some particular scenarios, the target region is a set of discrete points or curves. The target coverage and barrier coverage-defined as the ratio of discrete points and curves covered by at least one sensor node-are used to evaluate the sensing performance when targets are discrete points or curves.

The binary disc model provides an ideal but not realistic formulation for sensing accuracy. A more realistic and perceptual model is a probabilistic sensing model in which the sharp step function is replaced by a smooth exponential function, e.g.,

\[ Y(\|p_n - \omega\|) = \begin{cases} 1, & \|p_n - \omega\| - R_s < -R_e \\ e^{\|p_n - \omega\| - R_s} - R_e < \|p_n - \omega\| - R_s < R_e \\ 0, & \|p_n - \omega\| - R_s \geq R_e \end{cases} \quad (5) \]
Similarly, one can extend the probabilistic sensing model to heterogeneous WSNs by using different sensing ranges.

\[
Y(p_n - \omega) = \begin{cases} 
1, & \|p_n - \omega\| - R_{e,n} < -R_{c,n} \\
 e^{\lambda R_{e,n}^2}, & -R_{e,n} \leq \|p_n - \omega\| - R_{e,n} < R_{c,n} \\
 0, & \|p_n - \omega\| - R_{e,n} \geq R_{c,n} 
\end{cases}
\]  

(6)

Sensing uncertainty, which estimates the sensing error rate, is another family of sensing performance. Based on the assumption that each target is served by at most one sensor node, the authors in [4] formulate the node deployment problem as a quantization problem with distortion as its sensing uncertainty.

\[
C(P) = \sum_{n=1}^{N} \int_{\nu_{n}(P)} \|p_n - \omega\|^2 f(\omega) d\omega,
\]  

(7)

Where \(\nu_{n}(P) \Delta \big\{ \omega \|p_n - \omega\| - \|p_m - \omega\|, \forall m \neq n \big\}\) is called Voronoi Diagram (VD) [3]. By adding an additional coefficient of \(\eta_n\) which denotes the sensing ability, the sensing uncertainty model is thus extended to heterogeneous WSNs [2].

\[
D(P) = \sum_{n=1}^{N} \int_{\nu_{n}(P)} \eta_n \|p_n - \omega\|^2 f(\omega) d\omega,
\]  

(8)

With Multiplicatively Weighted Voronoi Diagrams (MWVD) [4] \(\nu_{n}(P) \Delta \big\{ \omega \|p_n - \omega\| < \eta_n \|p_m - \omega\|, \forall m \neq n \big\}\). According to the simulations in [5], node deployments with low sensing uncertainty are more likely to get high sensing accuracy/coverage.

2.2. Energy Consumption

On the one hand, sensor nodes are equipped with more powerful but energy-consuming communication and computation components in recent decades. Some sensor nodes, like robots and unmanned aerial vehicles (UAVs), even have mobility, which will dominate the total energy consumption. On the other hand, sensors have limited battery energy, and it is inconvenient or even infeasible to recharge numerous densely deployed sensors. As a result, energy conservation becomes a crucial issue in WSNs. In fact, sensor movement and wireless communications have much higher energy consumption compared to other types of energy and then dominates energy consumption. For static WSNs, minimizing communication power is the primary task. However, saving energy consumed by movement is more critical in mobile WSNs. The minimization of communication energy and movement energy is discussed as follows.

Derived from propagation model [5,(2.51)] and Shannon theorem [6], the minimum power to provide reliable communications between nodes [4]-[6] is

\[
\mathcal{E}_n^c \left(\|p_n - \omega\| \right) = \gamma \|p_n - \omega\|^a,
\]  

(9)
For motion energy minimization, the optimal angular velocity and the optimal acceleration are well explored by [3]. In particular, the simulation results in [6] show that the minimum motion energy is approximately linear to the movement distance.

\[ \epsilon^m_n(P) = \kappa \left\| p_n - p_n^0 \right\| \] (10)

Two popular energy-related measures are total energy consumption and network lifetime defined by

\[ \epsilon = \sum_{n=1}^{N} \epsilon^c_n(P) + \sum_{n=1}^{N} \epsilon^m_n(P) \] (11)

and

\[ T = \min_n \frac{\epsilon^0_n - \epsilon^m_n(P)}{\epsilon^0_n(P)} \] (12)

Where \( p_n^0 \) and \( \epsilon^0_n \) are node \( n \)'s initial location and battery. Numerical sensor deployment algorithms have been designed to either minimize total energy consumption or maximize the network lifetime, which will be discussed in the next section.

3. Movement-assisted self-deployment algorithms

A vast body of literature exists on movement-assisted self-deployment algorithms. For static WSNs where sensors stay still once deployed, self-deployment algorithms do not physically move sensor nodes but determine the virtual movement path in each iteration. On the contrary, a random sensor placement is used as the initial node deployment in mobile WSNs. The self-deployment algorithms guide sensor nodes physically move to spots with better sensing performance or lower energy consumption. As a traditional approach, centralized algorithms utilize a fusion center to collect global information, such as real-time node deployments, and determine the relocation by complicated computations. Different from centralized algorithms, the distributed algorithms give freedom to each sensor node to determine their own locations individually.

3.1. Virtual Force Based Algorithms

Given the number of sensors, virtual-force based algorithms employ two kinds of virtual forces, i.e., attractive force and repulsive force, to each node. If a pair of sensor nodes are placed close to each other, the repulsive force dominated the virtual forces and prevented the two nodes from moving closer. On the contrary, if two sensor nodes are too far apart from each other, the two nodes will be attracted by each other via the attractive force. Besides, the boundary of the target region can also be regarded as a source of virtual force.

The genetic algorithm (GA) is a family of methods for solving both constrained and unconstrained optimization problems in terms of a natural selection process that drives biological evolution. Like natural selection, the genetic algorithm repeatedly modifies a population of individual solutions. At each step, GA randomly selects individuals from the existing population to compose parents and uses them to generate the children’s solutions with mutations. By selecting successive generations, the population evolves toward an optimal solution.

3.2. Particle Swarm Optimization Algorithm

Particle swarm optimization (PSO) is a computational approach that solves an optimization problem by iteratively moving particle swarms to search for better locations with a given sensing performance measurement. Each particle's movement is determined by the best-known position and their previous movement direction.
3.3. Lloyd Algorithms
From a perspective of source coding or quantization, node deployment optimization is a quantizer design problem where sensor nodes and cell partition act as reproduction points and quantizer partition, respectively. Therefore, some researchers apply Lloyd or Lloyd-like algorithms to minimize sensing uncertainty or energy consumption. The basic flow of the Lloyd-like algorithm iterates between two steps: (1) fix node locations and optimize cell partition; (2) fix cell partition and optimize node locations. In particular, if the sensing uncertainty from Sensor $n$ to a point $\omega$ is formulated as the distance square, $\|p_n - \omega\|^2$, the optimal deployment satisfied Centroidal Voronoi Tessellation (CVT), i.e., $p_n^* = c_n(P^*)$ [2]. This conclusion has been well studied and extended to heterogeneous WSNs in [2]. In addition, Lloyd-like algorithms, OTL and TTL [3], are proposed to minimize the communication energy in homogeneous two-tier static WSNs while another Lloyd-like algorithm, HTTL [3], have resolved the energy minimization in heterogeneous static WSNs. Taking routing protocol into consideration, a Routing-aware Lloyd Algorithm is able to reduce the communication energy in homogeneous WSNs further. To simplify the computation, two distributed Lloyd-like algorithms, Lloyd-and DEED, are studied in [5], [6] to replace the existing centralized algorithms at the cost of sensing performance. Different from static WSNs, energy conservation in mobile WSNs focuses on motion energy. The authors in [2] explore both centralized and distribution Lloyd-like algorithms to make a trade-off between sensing uncertainty and motion energy.

4. Conclusion
This tutorial introduces the background and main challenges, e.g., sensing performance, energy conservation, and connectivity, in wireless sensor networks (WSNs). The existing sensing performance models and energy models are summarized. The common movement-assisted node deployment strategies are divided into four classes: virtual-force based algorithms, genetic algorithms, particle swarm optimization algorithms, and Lloyd-like algorithms. We discuss each category and highlights their strengths and limitations.

Then, the detailed comparisons among the four classes of self-deployment algorithms concerning different scenarios are well explored in this tutorial.

References
[1] C. Zhu, C. Zheng, L. Shu, and G. Han, “A survey on coverage and connectivity issues in wireless sensor networks,” Journal of Network and Computer Applications, vol. 35, no. 2, pp. 619-632, Mar. 2012.
[2] Di Chang, Xia Zhang, Qiong Liu, Ge Gao, and Yue Wu. "Location based robust audio watermarking algorithm for social TV system." In Pacific-Rim Conference on Multimedia, pp. 726-738. Springer, Berlin, Heidelberg, 2012.
[3] Di Chang, Xia Zhang, and Yue Wu. "A Multi-Source Steganography for Stereo Audio." Journal of Wuhan University (Natural Science Edition), 2013(3): 277-284.
[4] Xia Zhang, Di Chang, Weimin Qi, and Zhiming Zhan. "Tree-like Dimensionality Reduction for Cancer-informatics." In IOP Conference Series: Materials Science and Engineering, vol. 490, no. 4, pp. 042028. IOP Publishing, 2019.
[5] Liang Ding, Di Chang, et al. "Efficient Learning of Optimal Markov Network Topology with k-Tree Modeling." arXiv preprint arXiv:1801.06900 (2018).
[6] Xia Zhang, Di Chang, et al. "An audio digital watermarking algorithm transmitted via air channel in double DCT domain." In 2011 International Conference on Multimedia Technology, pp. 2926-2930. IEEE, 2011.