UNIVERSITY OF CALIFORNIA,
IRVINE

Performance Optimization of Wireless Sensor Networks

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Electrical Engineering

by

Jun Guo

Dissertation Committee:
Professor Hamid Jafarkhani, Chair
Professor A. Lee Swindlehurst
Professor Ahmed Eltawil

2019

TABLE OF CONTENTS

LIST OF FIGURES

Page

LIST OF TABLES

LIST OF ALGORITHMS

ACKNOWLEDGMENTS

First, I would like to thank my advisor Prof. Hamid Jafarkhani for the continuous support of my Ph.D. study and related research. His professional and patient guidance helped me in aspect of learning, research and writing, including this dissertation. In addition, Professor Hamid Jafarkhani inspired my interests and willpower by posting weekly motto attachment on his door.

Besides my advisor, I am also grateful to my parents, who have provided me with moral and emotional support throughout my years of study and as well as process of researching and writing this thesis. Moreover, I acknowledge my other family members and friends who have supported me along the way. This accomplishment would not have been possible without them.

Another very special gratitude goes out to DARPA's GRAPHS Program, National Science Foundation, and also Holmes Foundation for providing funding for my research. I must express my very profound gratitude to Institute of Electrical and Electronics Engineers (IEEE) for offering me the permissions to incorporate my previous works into this dissertation.

And finally, graditude goes to everyone in the Center for Pervasive Communications and Computing (CPCC); it was great sharing laboratory with all of you during the last four and a half years. With a special mention to Erdem Koyuncu, Philipp Walk, and Saeed Karimi Bidhendi, it was extremely impressive to have the opportunity to collaborate with you for the majority of my research in the past several years.

Thanks for all your encouragement!

CURRICULUM VITAE

Jun Guo

EDUCATION

Doctor of Philosophy in Electrical Engineering **2014 – 2019**
University of California, Irvine *Irvine, USA*

Master in Electrical Engineering **2011 – 2014**
Beijing University of Posts and Telecommunications *Beijing, China*

Bachelor of Science in Electrical Engineering **2007 – 2011**
Beijing University of Posts and Telecommunications *Beijing, China*

RESEARCH EXPERIENCE

Graduate Research Assistant **2014 – 2019**
University of California, Irvine *Irvine, USA*

TEACHING EXPERIENCE

Teaching Assistant **2011 – 2014**
Beijing University of Posts and Telecommunications *Beijing, China*

WORKING EXPERIENCE

Data Engineer Intern **Jun. 2018 – Sept. 2018**
Amazon Corporation *San Francisco, USA*

Software Developer Engineer Intern **Jun. 2017 – Sept. 2017**
Amazon Corporation *San Francisco, USA*

Research Intern **Dec. 2012 – Dec. 2013**
Intel Corporation *Beijing, China*

REFEREED JOURNAL PUBLICATIONS

Sensor Deployment with Limited Communication Range in Homogeneous and Heterogeneous Wireless Sensor Networks — **Oct 2016**
IEEE Transactions on Wireless Communications
vol. 15, no. 10, pp. 6771-6784

A source coding perspective on node deployment in two-tier networks — **Jul 2018**
IEEE Transactions on Communications
vol. 66, no. 7, pp. 3035-3049

Movement-efficient Sensor Deployment in Wireless Sensor Networks with Limited Communication Range — **Jan 2019**
IEEE Transactions on Wireless Communications
revision is under review

Optimal deployment of UAVs equipped with directional antennas — **Feb 2019**
in preparation

REFEREED CONFERENCE PUBLICATIONS

Sensor Deployment in Heterogeneous Wireless Sensor Networks — **Dec 2016**
IEEE Global Communications Conference (Globecom)

Energy efficiency in two-tiered wireless sensor networks — **May 2017**
IEEE International Conference on Communications (ICC)

Movement-efficient Sensor Deployment in Wireless Sensor Networks — **May 2018**
IEEE International Conference on Communications (ICC)

Quantizers with Parameterized Distortion Measures — **Mar 2019**
IEEE Data Compression Conference (DCC)

Using Quantization to Deploy Heterogeneous Nodes in Two-Tier Wireless Sensor Networks — **Jul 2019**
IEEE International Symposium on Information Theory (ISIT)
submitted

Energy Efficient Node Deployment in Wireless Ad-hoc Sensor Networks — **Dec 2019**
IEEE Global Communications Conference (Globecom)
in preparation

ABSTRACT OF THE DISSERTATION

Performance Optimization of Wireless Sensor Networks

By

Jun Guo

Doctor of Philosophy in Electrical Engineering

University of California, Irvine, 2019

Professor Hamid Jafarkhani, Chair

In this dissertation, I study three factors, sensing quality, connectivity, and energy consumption in static/dynamic wireless sensor networks (WSNs). First, taking sensing quality and connectivity into account, I formulate the node deployment problem in both WSNs from a source coding perspective. According to our analysis, the techniques in regular quantizer can be applied to both homogeneous and heterogeneous WSNs. Second, a one-tier quantizer with parameterized distortion measures is proposed for 3-dimension node deployment problems. Similarly, a novel two-tier quantizer, which can be applied to energy conservation in two-tier WSNs consisting of N access points and M fusion centers, is appropriately defined and studied. In addition, to make a trade-off between sensing quality and communication energy consumption within static WSNs, routing algorithms are appropriately taken into the system model. Moreover, a comprehensive optimization problem is provided to process all three factors in a dynamic WSN where movement energy dominates total energy consumption. The necessary conditions for the optimal solutions in the above performance optimization problems are proposed in this dissertation. Based on these necessary conditions, a series of Lloyd-like algorithms are designed and implemented to optimize the performance in different WSNs. My experiment results show that the proposed algorithms outperform the existing algorithms in the corresponding WSNs.

Chapter 1

Introduction

1.1 Background

As a bridge between the physical world and the virtual information word, wireless sensor networks (WSNs) collect data from the physical world and communicate it with the virtual information world, such as computers. On the one hand, from the perspective of mobility, sensors can be classified as static sensors and mobile sensors [1]. Static sensors, such as temperature sensors, stay still after their one-time deployment while mobile sensors, such as robots and unmanned aerial vehicles (UAVs), are equipped with motion components and able to move to other locations. On the other hand, according to sensors' functionality/capacity, WSNs can be classified into homogeneous WSNs and heterogeneous WSNs. In homogeneous WSNs, extensively studied in the literature [2]–[31], sensors share the same capacity, e.g., storage, computation power, sensitivity, communication radius, coverage radius, moving efficiency, and battery. The heterogeneous WSNs consist of sensors with different capacities [32]–[38]. Proper sensor deployment improves monitoring and controlling of the physical environment, such as temperature, humidity, voice and so on. To accomplish their tasks, WSNs

should address three needs: (i) Sensing in the target region and (ii) network connectivity, and (iii) energy conservation.

One major challenge in both homogeneous and heterogeneous WSNs is the deployment of nodes to improve the sensing quality. To evaluate the sensing quality, the binary coverage model [2]–[9], [11], [12], [32], [33], [36], [39]–[45] and the probabilistic model [46]–[51] are widely used in WSNs. In the binary coverage model, sensors can only detect the points within a range of R_s. The range R_s is called the sensing range. In the probabilistic coverage model, the probability that a sensor detects an event depends on the distance between them. When the number of sensors is large enough to cover the whole sensing region, the coverage degree in [11], [52] is used to evaluate sensing performance. On the contrary, when the sensing region is too large to be covered by the given sensors, the coverage area is used as the performance measurement [36]. There are many coverage models and deployment algorithms, for different sensing tasks, in the literature; look at [2]–[4] and the references therein. According to the sensing tasks, coverage models can be classified into four popular categories: (i) area coverage, (ii) target coverage, (iii) barrier coverage, and (iv) sensing uncertainty. A natural sensing task is to maximize the area coverage, which is formulated by the total area covered by sensors. In another popular coverage model, target coverage, the specific target locations are detected and reported by relocated sensors. In this case, sensors or robots are required to collect detailed information from discrete targets. A full-target-coverage is achieved if and only if every discrete target in the 2-dimensional region is covered by at least one sensor. Sensors in barrier coverage model move along the boundary to detect intruders as they cross the border of a region or domain. To obtain full-barrier-coverage, one should place sensors to cover the whole barrier or boundary. Finally, the minimization of sensing uncertainty requires sensors to form a Centroidal Voronoi Tessellation (CVT). It is mainly used when there is no specific target. The widely used sensing uncertainty model can be presented as a quantizer with the sensing uncertainty as its distortion [2]–[10], [13], [15]–[18], [32]–[34], [53]–[55]. In fact, WSNs should be reconfigurable/flexible to support

different sensing tasks. However, to the best of our knowledge, there is no unified framework that models a variety of coverage tasks.

Connectivity is another important requirement in WSNs. In WSNs, mobile sensor nodes are relocated to collect physical information, such as magnetism, temperature, and voice. The collected data is forwarded to the outside world through access points (APs). Therefore, the collected data is useless if it cannot be forwarded to the AP via single-hop or multiple-hop communications. When sensors are connected by wirelines, the connectivity is guaranteed automatically. The connectivity is still a challenge in WSNs where sensors are communicating with each other through wireless channels. A common communication model [3], [6], [10]–[12], [14], [52] assumes that each sensor node is able to communicate with sensors in a limited communication range R_c. The Critical Sensor Density (CSD) [3] is the number of nodes per unit area required to provide full coverage when the communication range is limited.

Energy efficiency is another key issue in WSNs, as most sensors have limited battery energy, and it is inconvenient or even infeasible to replenish batteries of numerous densely deployed sensors [18]. In general, the energy consumption of a device includes communication energy, data processing energy [56], sensing energy, and movement energy. In fact, sensor movement has a much higher level of energy consumption compared to other types of energy [19], [20]. Except movement, wireless communication is the primary source of energy consumption. Therefore, movement energy and communication energy will dominate the energy consumption in static WSNs and mobile WSNs, respectively. Literature [57] studies the optimal angular velocity and the optimal acceleration to minimize the energy consumption for motion. Simulation results in [57] show that the energy consumption for one-step motion with the optimal angular velocity setting is approximately linear to the movement distance. In fact, the linear movement energy consumption is a popular assumption and widely adopted in the literature [21]–[23], [35], [58]. Particularly, the movement energy consumption in some specific sensors is 5.976J/m [58]. Total energy consumption and network lifetime are two

common energy-related measures. When battery charging is available, total energy minimization is an important issue in WSNs. Otherwise, network lifetime maximization, which balances the energy consumption among sensors, is a more common and challenging problem because it takes into account the available battery power of sensors.

1.2 Node deployment from a source coding perspective

1.2.1 One-Tier Quantization for Node Deployment

In many applications, one needs to provide service to and collect data from a geographical area of interest via multiple nodes, such as sensors or providers. Usually, the nodes are distributed such that each point or client in the area is only served by only one node, resulting in a partition of the target area to disjoint regions. The service cost for each node may depend on its characteristics and local service region. The fundamental goal in such a formulation is to jointly optimize the node locations and the corresponding service regions to optimize the overall performance, which is typically defined as an aggregate of node service costs.

The cost of providing service to a client is usually related to its distance to the local service node. Therefore, minimizing the total cost by optimally deploying the nodes and the service regions is identical to a spatial tessellation problem [33]. Such problems (which have also been referred to as facility location or node deployment problems by different research communities) are equivalent to the quantization problem of data compression and source coding. In fact, in the language of quantization theory, the service nodes and service regions correspond to the reproduction points and quantization regions, respectively. Minimizing the total service cost becomes equivalent to minimizing the corresponding average distortion. We now present several specific applications to highlight the above equivalency.

In sensor networks, sensors are deployed in a two-dimensional planar to collect data from the environment. The goal of sensor/node deployment is to maximize the sensing performance and quality. An example of the sensor deployment in a two-dimensional planar is illustrated

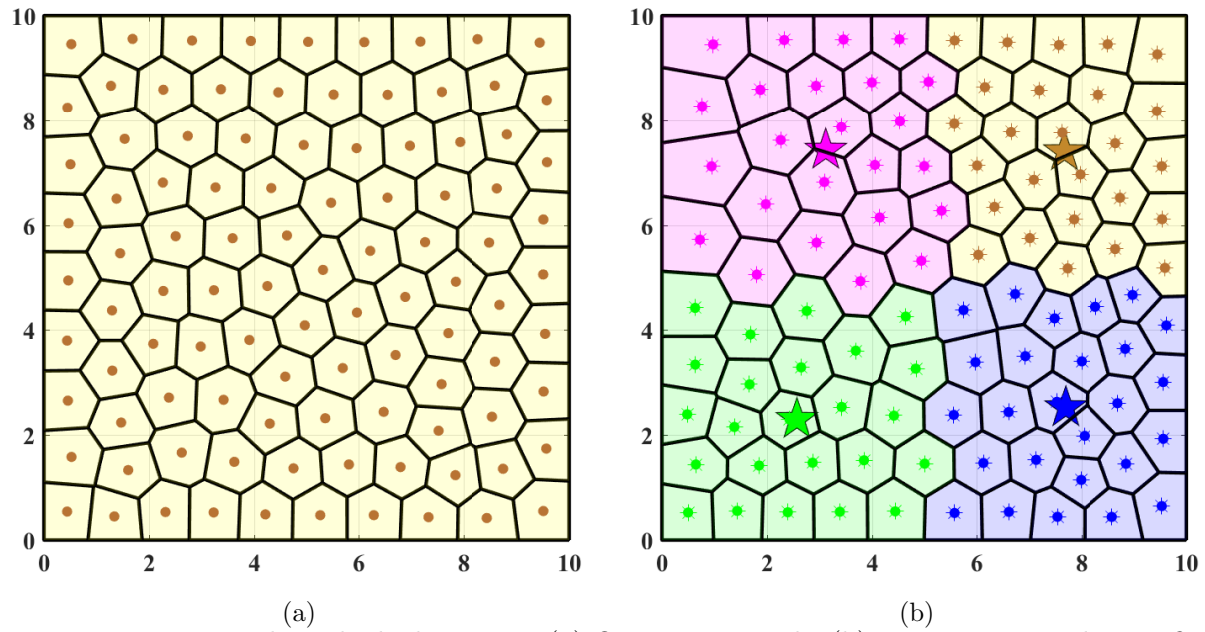


Figure 1.1: Two example node deployments. (a) One-tier network. (b) Two-tier network. 100 first-tier nodes and 4 second-tier nodes are denoted by dots and stars, respectively. The cell partitions are denoted by polygons. The symbols associated with the same second-tier node are filled with the same color.

in Fig. 1.1a, where each sensor is denoted by a dot that monitors its own region. In this scenario, the reproduction points correspond to the sensor nodes, and the quantization Voronoi regions correspond to the sensor partitions. The cost function (distortion measure) is usually a monotonically increasing function of the distance between the sensor and the event that is being sensed, and quantifies the inaccuracy of sensing. The most common cost function used in the quantization literature is the squared Euclidean distance [59]. It can be used directly in the sensor network example to measure the overall sensing inaccuracy in homogeneous wireless sensor networks (WSNs) [6]. For heterogeneous WSNs, a weighted Euclidean distance square measure can be used where the weights reflect the nodes' different sensing capabilities [32]. Other cost functions have also been utilized to formulate the sensing coverage or the sensing probability [32], [47], [60], [61].

Another example is the heterogeneous base station (BS) deployment [17], [62]–[64] in cellular

networks where user equipments (UEs) are considered as the source, heterogeneous BSs are recognized by the reproduction points, and the BS cells are represented by the quantization Voronoi regions. Because of the path loss, the signal strength at the receiver is a non-increasing function of the communication distance [65]–[67]. Using such a non-increasing function as the cost function, the distortion measure can be defined as the expected signal to noise ratio (SNR) at the UEs, where the expectation is calculated for a given channel probability distribution.

1.2.2 Two-Tier Quantization for Node Deployment

The conventional spatial tessellation and node deployment problems in Section 1.2.1 ignore the hierarchical architecture that is inherent in many networks. In fact, to reduce the network burden, many practical networks have a two-tier structure, as an extension to the one-tier examples considered in Section 1.2.1. For example, to provide an effective delivery service for the residents, the postal system uses a two-tier network including the local post offices and the postal hubs. Then, there will be two costs associated with each delivery. A *first-tier-cost* that is the cost of delivering packages from clients to local post offices, determined by the population density and the distance from clients to their local post offices, and a *second-tier-cost* which is the total cost of delivering packages between local post offices and the corresponding postal hubs. The *second-tier-cost* is determined by the workload of the local post offices and the distance from the local post offices to their postal hubs. A similar two-tier network appears in the hospital system where local hospitals provide the basic medical treatment and the residents with severe disastrous issues are transferred to the hospital centers with more medical facilities.

Two-tier WSNs and two-tier cellular networks are also very common network architectures. A two-tier WSN [37], [38], [68]–[70] includes densely deployed sensors, multiple access points (APs), and fusion centers (FCs). In such a network, sensor nodes collect the data and send it to their APs for processing. Then, each AP transmits its aggregated data to its associated

FC [37], [38], [68]–[70]. As depicted in Fig. 1.1b, the sensors, APs, and FCs correspond, respectively, to clients, first-tier nodes, and second-tier nodes in our two-tier network. One reasonable cost function in the two-tier WSN is the total energy consumption at sensors and APs. The objective is to optimize the trade-off between the sensor and AP energy consumption.

The goal of this paper is to study node deployment problems in two-tier networks, where two-tier nodes are deployed to provide service for the clients in the target region. In such two-tier networks, as depicted in Fig. 1.1b, the second-tier nodes provide service for the first-tier nodes that serve the clients. Similar to the one-tier networks, the cost between two points is generally a non-decreasing function of the Euclidean distance. Let the first-tier-cost be the total cost between clients and first-tier nodes, and the second-tier-cost be the total cost between the first-tier nodes and the corresponding second-tier nodes. Moving the first-tier nodes towards the second-tier nodes, usually, will increase the average distance between the first-tier nodes and the local clients, resulting in the increase of the first-tier-cost. On the other hand, moving the first-tier nodes towards the local clients, usually, will increase their distance to the second-tier nodes and will result in an increase in the *second-tier-cost*. Therefore, there is a trade-off between the *first-tier-cost* and the *second-tier-cost*.

1.3 Related Work

1.3.1 Quantization

A significant body of literature exists on designing the one-tier quantizers. Gray et al. [59] summarize the theory and practice of quantization since its inception. The best possible quantization distortion in a high-resolution regime has been discussed in [71]–[73] and the application of the high-resolution theory to node deployment in heterogeneous sensor networks is provided in [74]. Clustering is a related method where the cluster heads and cluster regions are, respectively, the reproduction points and quantization regions. Many different

hierarchical clustering methods, such as Agglomerative Clustering (AC) and Divisive Clustering (DC), are discussed in [75]. Furthermore, some existing quantization schemes in the literature can be employed to the two-tiered quantizers. For example, successively refinable vector quantizers (SRVQs) [76]–[78] have multiple stages. As another example, hierarchical vector quantizers (HVQs) [79]–[81] are proposed to reduce the quantizer encoding complexity. HVQs employ quantizers of different dimensions in different hierarchical steps.

1.3.2 Node Deployment

A huge body of literature exists on energy-efficient sensor relocation. However, most of the papers in the literature consider one or two key metrics rather than all the three (sensing quality, connectivity, and energy efficiency) together. Moreover, there is no unified framework that can support different coverage models. References [21]–[23], [35], [36] study the energy saving with a full-area-coverage guarantee. Hungarian Algorithm is applied to minimize the total energy consumption after the full-area-coverage is achieved by Genetic Algorithm [21]. Similarly, the grid-based algorithms are proposed in [82] to reduce the total energy consumption while keeping the full-area-coverage and full-connectivity. Kuei-Ping et al. [83] propose a distributed partition avoidance lazy movement (PALM) protocol, which avoids unnecessary movement, to ensure both full-area-coverage and connectivity. Shuhui et al. [84] provide a scan-based relocation algorithm, SAMRT, which is supposed to be energy-effective with densely deployed sensors. Note that the above method put sensing quality as the first priority, and the energy efficiency is merely the secondary objective. To provide a flexible and fair trade-off between area coverage and energy consumption, virtual force based algorithms, HEAL [22], VFA [23], [36], and DSSA [35], are proposed. In [35], the authors take into account the local sensor density, and thus avoid unnecessary movements in the region with densely deployed sensors. In [22], HEAL is designed to mend area coverage holes while minimizing the moving distance. However, the main assumption that there are enough sensors to achieve full-area-coverage, limits its usage. A virtual force based algorithm, VFA

[23] is proposed to maximize the area coverage while saving energy. Instead of saving the total energy consumption, another virtual force based algorithm, VFA, is proposed in [23] to prolong the network lifetime during the area coverage maximization. Besides, a variant of VFA is designed in [36] to maximize the area coverage in a heterogeneous WSN. However, connectivity is not considered in [22], [23], [35], [36].

Sensor relocation for target coverage and barrier coverage is also well studied. Rout et al. [24] design a virtual-force based algorithm, OATIDA, to obtain both full-target-coverage and full-connectivity on a region with obstacles, while energy consumption is ignored. Chen et al. [25] propose a two-phase algorithm to achieve full-target-coverage with minimum total energy consumption, but connectivity is not taken in to account. Similarly, Njoya et al. [27] design an evolutionary-based framework to make the trade-off between target coverage and network lifetime while the connectivity issue is ignored. Liao et al. [26] investigate how to deploy mobile sensors with minimum total energy consumption to form a WSN that provides both full-target-coverage and full-connectivity. Although all three factors are considered in [26], full-target-coverage and full-connectivity are implemented sequentially, which requires redundant sensors. On the other hand, the existing literature on barrier coverage also seeks the perfect sensing quality, i.e., full-barrier-coverage. Chen et al. [28] focus on 1-dimensional barriers, and then provide an energy-efficient relocation plan to obtain full-barrier-coverage. In [29], a greedy algorithm with binary search is applied to achieve maximum network lifetime and 2-dimensional full-barrier-coverage simultaneously. A faster algorithm which achieves the same purpose as [29] is provided in [30]. Still, the above sensor relocation algorithms designed for barrier coverage ignore the connectivity requirement.

Even if the entire region, targets, or barriers are covered, sensors have different sensing accuracy at different points in the covered region. In fact, sensing capability diminishes as the distance increases [6], [33], [85]–[92]. Therefore, a distance based model, sensing uncertainty (or CVT model), has also been investigated in the literature. Distortion, as an important parameter in source coding, models the sensing uncertainty via a function

of distance [2]–[10], [13], [15]–[18], [32]–[34], [53]–[55], [91]. One can minimize distortion in WSNs through vector quantization techniques in [59], [93]. Lloyd algorithm [94] is one of the tools to minimize distortion in WSNs. The convergence of the Lloyd algorithm has been studied in [95]–[97]. Taking both connectivity and sensing uncertainty into account, our previous work [32], [98] proposes the necessary conditions for the optimal sensor relocation in heterogeneous WSNs. Unfortunately, another key metrics, energy consumption, is not taken into consideration. Li et al. [9] explore directional sensors whose sensing uncertainty varies among different directions, and then design two iterative algorithms to optimize the sensor deployment. However, energy consumption is not taken into their objective function. The energy consumption of directional sensors or UAVs are analyzed in my previous work [55]. A natural approach to save energy is to add an energy-related penalty term into the objective function. In [10], the authors propose two Lloyd-like algorithms, Lloyd-α and DEED, to minimize sensing uncertainty with a movement related penalty function. For Lloyd-α, the movement in each iteration is scaled by a parameter $\alpha \in [0, 1]$. In DEED, the penalty function is properly selected with a positive definite matrix depending on a parameter δ, and then the movement is optimized with the help of the gradient and Hessian matrix of the distortion. Note that one has to manually adjust the parameter α (or δ) for Lloyd-α (or DEED) to satisfy a specific energy constraint. To overcome this weakness, two Lloyd-like algorithms without any intermediate parameter are proposed in my previous work [53]. These two algorithms can be employed to minimize sensing uncertainty with a total energy constraint or a network lifetime constraint.

1.4 Contributions and Organization

In this dissertation, I study the performance (sensing quality, connectivity, and energy consumption) optimization in WSNs and make the following contributions: (1) The sensing quality in heterogeneous WSNs with limited communication range is formulated as a one-tier

quantizer. (2) Both theoretical analysis and numerical algorithms are proposed to optimize the sensing quality with the connectivity guarantee. (3) The optimal UAV deployment is found to minimize the the uplink communication energy conservation in 3-Dimension WSN. (4) The energy consumption in a two-tier WSN is formulated as a two-tier quantizer. (5) Based on the necessary conditions, Lloyd-like algorithms are proposed to search the optimal node deployment. (6) A unified optimization framework for different coverage models is provided that takes three key metrics, sensing quality, connectivity, and energy consumption, into consideration. (7) By providing analytical necessary conditions, I design centralized and distributed Lloyd-like algorithms to optimize sensor relocation with (i) network lifetime constraints and (ii) limited communication ranges.

In the rest of this dissertation, I first study the sensing quality in both homogeneous and heterogeneous WSNs with limited communication range in Chapter 2. In Chapter 3, I explore the energy conservation of WSNs with directional UANS. In Chapter 4, I analyze the optimal node deployment and cell partition to minimize the average power in a two-tier WSN. After that, a unified optimization framework for different coverage models is provided in Chapter 5. Finally, I conclude my work in Chapter 6.

Chapter 2

Sensor Deployment with Limited Communication Range in Homogeneous and Heterogeneous Wireless Sensor Networks

In this chapter, I study the heterogeneous wireless sensor networks (WSNs) and propose the necessary condition of the optimal sensor deployment. Similar to that in homogeneous WSNs, the necessary condition implies that every sensor node location should coincide with the centroid of its own optimal sensing region. Moreover, I discuss the dynamic sensor deployment in both homogeneous and heterogeneous WSNs with limited communication range for the sensor nodes. The purpose of sensor deployment is to improve sensing performance, reflected by distortion and coverage. I model the sensor deployment problem as a source coding problem with distortion reflecting sensing accuracy. However, when the communication range is limited, a WSN is divided into several disconnected sub-graphs under certain conditions as I will discuss in this paper. In such a scenario, neither the conventional distortion nor the coverage represents the sensing performance as the collected data in disconnected

sub-graphs cannot be communicated with the access point. By defining an appropriate sensing performance measure, I propose a Restrained Lloyd (RL) algorithm and a Deterministic Annealing (DA) algorithm to optimize sensor deployment in both homogeneous and heterogeneous WSNs. Our simulation results show that both DA and RL algorithms outperform the existing algorithms when communication range is limited.

2.1 System Model and Problem Formulation

Let Ω be a simple convex polygon in $\mathbb{R}^2$ including its interior. Given N sensors in the target area Ω, sensor deployment is defined by $\mathbf{P} = (\mathbf{p}_1, \cdots, \mathbf{p}_N) \subset \Omega^N$, where $\mathbf{p}_n$ is Sensor n's location. For any point $\boldsymbol{\omega} \in \Omega$, $f(\boldsymbol{\omega})$ is the probability density function of an event at point $\boldsymbol{\omega}$. I denote the first N natural numbers by $[N] = \{1, \ldots, N\}$. A cell partition $\mathcal{R}$ of Ω is a collection of disjoint subsets of $\{\mathcal{R}_n(\mathbf{P})\}_{n\in[N]}$ whose union is Ω. Let $\mathbb{B}(c, r) = \{\boldsymbol{\omega} |\ \|\boldsymbol{\omega} - c\| \leq r\}$ be a disk centered at c with radius r in two-dimensional space. For two points a and b, let equation $Eq + F = 0$, where $E \in \mathbb{R}^{1\times 2}$ is a 1×2 matrix and $F \in \mathbb{R}$ is a constant, define the perpendicular bisector hyperplane between the two points. Then, the equations $Eq + F \geq 0$ and $Eq + F \leq 0$ define two half spaces. I denote the half space that contains point a by $HS(a, b)$.

As mentioned before, I define the AP as the sensor node that can communicate with the outside information world. Let $\mathcal{S}(\mathbf{P})$ be the set of sensor nodes that can communicate with the AP when the sensor deployment is $\mathbf{P}$. Note that in general not all nodes can communicate with the AP and $card(\mathcal{S}(\mathbf{P})) \leq N$, where $card(A)$ is the number of elements in set A. I define a new sensor deployment, which is a subset of the all sensor locations, $\mathcal{H}(\mathbf{P})$ as the vector of sensor locations for the $card(\mathcal{S}(\mathbf{P}))$ sensor nodes connected to the AP. When $\mathcal{S}(\mathbf{P})$ includes all sensor nodes, we have $\mathbf{P} = \mathcal{H}(\mathbf{P})$ and $card(\mathcal{S}(\mathbf{P})) = N$. Let $\mathcal{U}$ be the set of sensor deployments that provide full connectivity, i.e., $\mathcal{U} = \{\mathbf{P} | card(\mathcal{S}(\mathbf{P})) = N\}$. In my model, two sensor nodes can communicate with each other within one hop if and only if the distance

between the two is smaller than R_c, where R_c is referred to as the communication range. A sensor node can transfer data outside if and only if there exists a path from the sensor to the AP. The path consists of a sequence of sensor nodes where each hop distance is smaller than the communication range R_c. Sensor nodes that are connected to the AP construct the backbone network. Specifically, I can choose the AP as the root and run Breadth First Search (BFS) or Depth First Search (DFS) to obtain the spanning tree. Obviously, sensors in the spanning tree construct the backbone network. If all sensors are included in the backbone network, I call the network fully connected. Otherwise, the network is divided into several disconnected sub-graphs.

Another important factor in analyzing the performance of a WSN is its sensing distance. Sensing performance directly depends on distance [6], [33], [85]–[92]. Therefore, to represent the average sensing accuracy in the target area, I define the following general performance measure:

$$\mathcal{D}(\mathbf{P}) = \sum_{n=1}^{N} \int_{\mathcal{R}_n(\mathbf{P})} f^{(n)}(\|\boldsymbol{\omega} - \mathbf{p}_n\|) f(\boldsymbol{\omega}) d\boldsymbol{\omega}, \tag{2.1}$$

where performance function $f^{(n)} : \mathbb{R}_+ \to \mathbb{R}_+$ is a non-increasing function of the distance between $\mathbf{p}_n$ and $\boldsymbol{\omega}$. When sensors have an identical performance function, i.e., $f^{(n)}(\cdot) = f(\cdot)$, (2.1) becomes the performance measure in homogeneous WSNs and has been widely used in different applications, such as the precipitation estimation problem in [33]. The partition $\{\mathcal{R}_n(\mathbf{P})\}_{n\in[N]}$ in the above definition include all sensor nodes. However, as explained previously, when the communication range is limited, some sensor nodes cannot transfer their data back to the AP. As a result, only the sensor nodes in the backbone network can contribute to the sensing and therefore the performance should be revised as

$$\mathcal{D}(\mathbf{P}) = \sum_{n\in\mathcal{S}(\mathbf{P})} \int_{\mathcal{R}_n(\mathcal{H}(\mathbf{P}))} f^{(n)}(\|\boldsymbol{\omega} - \mathbf{p}_n\|) f(\boldsymbol{\omega}) d\boldsymbol{\omega}. \tag{2.2}$$

Note that to derive (2.2) from (2.1), one has to replace $\mathbf{P}$ with $\mathcal{H}(\mathbf{P})$, i.e., one has to consider

only the sensor nodes that are in the backbone network. I reiterate that in the case of a fully connected network, $\mathcal{H}(\mathbf{P}) = \mathbf{P}$, Eqs. (2.1) and (2.2) are identical. Given a fully connected network, the distortion $\mathcal{D}(\mathbf{P}) = \mathcal{D}(\mathbf{P}, \mathcal{R}(\mathbf{P}))$ is determined by the sensor deployment $\mathbf{P}$ and the cell partition $\mathcal{R}(\mathbf{P})$. In homogeneous WSNs, every point should be detected by the nearest sensor in order to make the largest contribution to the total performance. This is because (i) all sensors in homogeneous WSNs have the same performance function $f(x)$ which is only determined by the Euclidean distance $x = \|\boldsymbol{\omega} - \mathbf{p}_n\|$, and (ii) the sensing ability diminishes as the distance increases. Therefore, given the sensor deployment, Voronoi partitions [33], [99] provide the optimal performance. The Voronoi partition of Ω generated by $\mathbf{P}$ with respect to the Euclidean norm is the collection of sets $\{\boldsymbol{V}_n(\mathbf{P})\}_{n\in[N]}$ defined by

$$\boldsymbol{V}_n(\mathbf{P}) = \{\boldsymbol{\omega} \in \Omega |\ \|\boldsymbol{\omega} - \mathbf{p}_n\| \leq \|\boldsymbol{\omega} - \mathbf{p}_k\|, \forall k \in [N]\}, \tag{2.3}$$

where $\|\cdot\|$ is the Euclidean norm.

However, different sensors with different complexity, power, and sensing ability are used in heterogeneous WSNs. As I will show later, the optimal partitioning in this case is MWVD [44]. The MWVD of Ω generated by $\mathbf{P}$ is the collection of sets $\boldsymbol{V}_n^H(\mathbf{P})_{n\in[N]}$ defined by

$$\boldsymbol{V}_n^H(\mathbf{P}) = \{\boldsymbol{\omega} \in \Omega |\ \eta_n\|\boldsymbol{\omega} - \mathbf{p}_n\|^2 \leq \eta_k\|\boldsymbol{\omega} - \mathbf{p}_k\|^2, \forall k \in [N]\}, \tag{2.4}$$

where the cost parameters $\{\eta_n \in \mathbb{R}_+\}_{n\in[N]}$ are constants that depend on the sensor characteristics, indicating the quality of sensor nodes. For example, the transmission power of Radar sensors, like the distance between the sensor and the events, has a direct influence on the sensing performance [87]. Therefore, radar sensors with different transmitting power will have different sensing abilities. In this case, η_n depends on the transmitting power. The smaller the cost parameter, the stronger the sensing ability. Both Voronoi regions $V_n(\mathbf{P})_{n\in[N]}$ and MWVDs $\boldsymbol{V}_n^H(\mathbf{P})_{n\in[N]}$ are functions of $\mathbf{P}$. Since the Voronoi partitioning can be considered as a special case of the weighted Voronoi partitioning, in which $\eta_n = 1, n \in [N]$, I

simply use $\boldsymbol{V}_n^H(\mathbf{P})_{n\in[N]}$ to represent both. Placing (2.4) back to (2.2), I will have

$$\mathcal{D}(\mathbf{P}) = \sum_{n\in\mathcal{S}(\mathbf{P})} \int_{\boldsymbol{V}_n^H(\mathcal{H}(\mathbf{P}))} f^{(n)}(\|\boldsymbol{\omega} - \mathbf{p}_n\|) f(\boldsymbol{\omega}) d\boldsymbol{\omega}. \tag{2.5}$$

Obviously, choosing different functions $f^{(n)}(\cdot)$ in (2.5) results in different problem formulations. One natural choice for the function $f^{(n)}(\cdot)$ is a continuous function defined by

$$f^{(n)}(x) = -\eta_n x^2. \tag{2.6}$$

(2.6) represents the negative of the mean squared error (MSE) in source coding and I refer to it as the **negative weighted MSE model**. By this definition, each sensor can detect all points in its sensing region $\mathcal{R}_n(\mathbf{P})$. Using MWVDs, every event is sensed by the node with the smallest cost. Therefore, given the sensor deployment, MWVDs provide the minimum distortion. In this model, maximizing $\mathcal{D}(\mathbf{P})$ is equivalent to minimizing the distortion.

Another choice for the function $f^{(n)}(\cdot)$ is the one that results in the **binary coverage model**. In this model, Sensor n can detect events within a circle with a fixed radius $\frac{R_s}{\sqrt{\eta_n}}$. The covered area with respect to density function is $\int_{\bigcup_{n\in\mathcal{S}(\mathbf{P})} \mathbb{B}(\mathbf{p}_n, R_s)} f(\boldsymbol{\omega}) d\boldsymbol{\omega}$. This model is equivalent to (2.5) if the performance function is defined by

$$f^{(n)}(x) = \begin{cases} 1, & \text{for } \eta_n x^2 < R_s^2 \\ 0, & \text{for } \eta_n x^2 \geq R_s^2 \end{cases}. \tag{2.7}$$

This is easily shown by replacing $f^{(n)}(\cdot)$ from (2.5) by (2.7) and using the fact that

$$\bigcup_{n\in\mathcal{S}(\mathbf{P})} \mathbb{B}(\mathbf{p}_n, R_s) = \bigcup_{n\in\mathcal{S}(\mathbf{P})} \left[\boldsymbol{V}_n^H(\mathcal{H}(\mathbf{P})) \bigcap \mathbb{B}(\mathbf{p}_n, \frac{R_s}{\sqrt{\eta_n}}) \right]. \tag{2.8}$$

The proof of (2.8) is trivial and is omitted here. The resulting coverage area for the homogeneous case, $\eta_n = 1$ for all i, is the same as the model used in [36], [40]–[44].

A third choice for the function $f^{(n)}(\cdot)$ is the one that results in the **exponential coverage model**. Our exponential coverage model is a modified version of the probabilistic coverage model for homogeneous WSNs in the literature [46]–[51], where I assume each event is only sensed by the sensor with highest probability of sensing. In the probabilistic coverage model, events within the confident range of $\frac{Rs}{\sqrt{\eta_n}}$ can be sensed without loss. For the events outside of the confident range, the probability of correctly sensing event $\boldsymbol{\omega}$ by Sensor n is a non-increasing function of the distance $x = \|\boldsymbol{\omega} - \mathbf{p}_n\|$. I model this sensing probability via

$$P_n^s(x) = \begin{cases} 1, & \text{for } \eta_n x^2 < R_s^2 \\ e^{-\varepsilon(\eta_n x^2 + R_s^2)}, & \text{for } \eta_n x^2 \geq R_s^2, \end{cases} \tag{2.9}$$

where $\varepsilon \in \mathbb{R}_+$ is a positive constant. The exponential coverage model approximates the binary coverage model when $\varepsilon \to +\infty$. The covered area with respect to density function is $\int_\Omega \max_n P_n^s(\|\boldsymbol{\omega} - \mathbf{p}_n\|) f(\boldsymbol{\omega}) d\boldsymbol{\omega}$. Our exponential coverage model, that allows sensing each event by only one sensor, is equivalent to (2.5) with performance function $f^{(n)}(x) = P_n^s(x)$. Without loss of generality, I represent every performance function $f^{(n)}$ as a piece-wise continuous and differentiable function with l discontinuities, i.e., jumps, at $r_1^{(i)}, \ldots, r_l^{(i)} \in \mathbb{R}_+$, with $r_1^{(i)} < \cdots < r_l^{(i)}$. For convenience, I set $r_0^{(i)} = 0$ and $r_{l+1}^{(i)} = +\infty$ and rewrite $f^{(n)}(x)$ as

$$f^{(n)}(x) = \sum_{\alpha=1}^{l+1} f_\alpha^{(n)}(x) 1_{[r_{\alpha-1}, r_\alpha)}(x), \tag{2.10}$$

where $f_\alpha^{(i)}(\cdot)$ is a continuous and differentiable function and $1_s(x)$ is the indicator function defined by $1_s(x) = 1$ if $x \in s$, and $1_s(x) = 0$ if $x \notin s$. Applying this representation to the specific models presented before, we have no discontinuity, i.e., $l = 0$ for negative weight MSE and $l = 1$ for binary coverage and exponential coverage. Our main goal is to find a sensor deployment that maximizes $\mathcal{D}(\mathbf{P})$ defined in (2.2). It is easy to show that a necessary condition for such an optimal sensor deployment is to have a fully connected

network. Moreover, MWVDs are the optimal partitions for the three models in this section. Using MWVDs, the performance is determined by the sensor deployment only. This is the topic of the discussion in the next section.

2.2 Optimal Deployment in Heterogeneous WSNs

In this section, I study the optimal deployment in heterogeneous WSNs with infinite or limited communication range. Finding the global optimal deployment is difficult because one needs to compare all the local maxima. Let $\mathbf{P}^*(R_c)$ be a deployment set including all the critical deployments that obtain the performance maxima when communication range is R_c. Instead of finding the global optimal deployment, one should at least find a deployment $\mathbf{P} \in \mathbf{P}^*(R_c)$. The connectivity is guaranteed when the communication range R_c is infinite. Under such circumstance, the objective function is continuous and differentiable. Therefore, the critical deployment set $\mathbf{P}^*(\infty)$ is just the set of deployment with zero-gradient. However, the wireless communication range is limited due to the finite transmitting power, and the network will be divided into several subgraphs when some sensors are out of the AP's spanning tree. It is self-evident that the objective function is a discontinuous and indifferentiable function when communication range is limited. The jumps result from the changes of nodes in the backbone network. In this case, the critical deployment set becomes $\mathbf{P}^*(R_c) = \{\mathbf{P} = (\mathbf{p}_1, \ldots, \mathbf{p}_N) | \frac{\partial \mathcal{D}(\mathbf{P})}{\partial \mathbf{p}_n}$is 0 or it does not exist, $\forall n \in [N]\}$. Next, I provide the necessary condition for the optimal deployment in heterogeneous WSNs.

Lemma 1. *The necessary condition for the optimal deployment in heterogeneous WSNs with communication range R_c is $\mathbf{P} \in \mathbf{P}^*(R_c)$.*

Proof. Let $\Delta = \sum_{n=1}^{2N} \Delta_n e_n$ be a $2n$-dimensional vector, where $e_n, n \in [N]$, are standard basic vectors. For any local maximum point $\mathbf{P}^*$, there exists some $\epsilon > 0$ such that $\mathcal{D}(\mathbf{P}) \geq \mathcal{D}(\mathbf{P}^* + \Delta)$ for all Δ with $\|\Delta\| \leq \epsilon$. Specially, $\mathcal{D}(\mathbf{P}^*) \geq \mathcal{D}(\mathbf{P}^* + \epsilon e_n)$ for all $n \in [N]$. As a

result, the partial derivative $\frac{\partial \mathcal{D}(\mathbf{P})}{\partial \mathbf{p}_n}|_{\mathbf{P}=\mathbf{P}^*}$ is 0 or does not exist for $n \in [N]$. Specially, when $R_c = +\infty$, the partial derivative $\frac{\partial \mathcal{D}(\mathbf{P})}{\partial \mathbf{p}_n}|_{\mathbf{P}=\mathbf{P}^*} = 0$ for all $n \in [N]$. □

Now, I can easily obtain the the critical deployment set by calculating the gradient of the objective function $\mathcal{D}(\mathbf{P})$. In homogeneous WSNs, every sensor uses the same performance function and (2.5) can be rewritten as

$$\mathcal{D}(\mathbf{P}) = \sum_{n \in \mathcal{S}(\mathbf{P})} \int_{V_n(\mathcal{H}(\mathbf{P}))} f(\|\boldsymbol{\omega} - \mathbf{p}_n\|) f(\boldsymbol{\omega}) d\boldsymbol{\omega} \tag{2.11}$$

The partial derivatives of (2.11) are calculated in [6] as

$$\begin{aligned} \frac{\partial \mathcal{D}(\mathbf{P})}{\partial \mathbf{p}_n} = & \int_{V_n(\mathbf{P})} \frac{\partial}{\partial \mathbf{p}_n} f(\|\boldsymbol{\omega} - \mathbf{p}_n\|) f(\boldsymbol{\omega}) d\boldsymbol{\omega} \\ & + \sum_{\alpha=1}^{l} \left[(f_\alpha(r_\alpha) - f_{\alpha+1}(r_\alpha)) \int_{arc_n(P, r_\alpha)} \mathbf{n}^t(\boldsymbol{\omega}) f(\boldsymbol{\omega}) d\boldsymbol{\omega} \right], \end{aligned} \tag{2.12}$$

where $arc_n(\mathbf{P}, r_\alpha)$ consists of arcs in the boundary of $V_n(\mathbf{P}) \bigcap \mathbb{B}(\mathbf{p}_n, r_\alpha)$, $\mathbf{n}^t(\boldsymbol{\omega})$ is the unit outward normal to $\bigcup_{\alpha=1}^{l} arc_n(\mathbf{P}, r_\alpha)$ at point $\boldsymbol{\omega}$. Before I discuss the partial derivatives in heterogeneous WSNs, I need to present the following definitions and Lemmas.

Definition 2.1. *A set $S \subseteq \mathbb{R}^N$ is called star-shaped if and only if there exists a point $p \in int(S)$ such that for all $s \in \partial S$ and all $\lambda \in (0, 1]$, one has $\lambda p + (1-\lambda)s \in int(S)$, where $int(S)$ is the interior of S and ∂S is the boundary of S. The point p is the reference point.*

Definition 2.2. *A set $S \subseteq \mathbb{R}^N$ is called a convex region if and only if for every pair of points $x, y \in S$ and all $\lambda \in (0, 1)$, one has $\lambda x + (1-\lambda)y \in int(S)$.*

Lemma 2. *If a set $S \subseteq \mathbb{R}^N$ is convex, then S is star-shaped.*

Proof. For any convex region $S \subseteq \mathbb{R}^N$, pick a point $p \in int(S) \subset S$. For any point $s \in \partial S \subset S$ and all $\lambda \in (0, 1)$, one has $\lambda p + (1-\lambda)s \in S$. When $\lambda = 1$, $\lambda p + (1-\lambda)s = p \in int(S)$. Therefore, the set S is star-shaped. □

I will use the fact that the intersection of any collection of convex sets is convex [100], [101] and as a result star-shaped according to Lemma 2.

Lemma 3. *The union of star-shaped sets that are associated with the same reference point p is star-shaped.*

Proof. Given K star-shaped sets $S_k, k \in [K]$ with the same reference point p, the corresponding union is $S = \bigcup_{k=1}^{K} S_k$. Since point p is the reference point of star-shaped sets S_k, where $k \in [K]$, we have $p \in \bigcap_{k=1}^{K} S_k$ and therefore $\bigcap_{k=1}^{K} S_k \neq \emptyset$. The boundary of S comes from the boundaries of K star-shaped sets S_k, where $k \in [K]$. Thus, for any point $s \in \partial S$, we have $s \in \bigcup_{k=1}^{K} \partial S_k$. Because of K star-shaped sets, for all $s \in \partial S_k$ and all $\lambda \in (0, 1]$, I will have $\lambda p + (1-\lambda)s \in int(S_k)$. Thus, for all $s \in \partial S$ and for all $\lambda \in (0, 1]$, one can find a subset S_k such that $s \in \partial S_k$ and so $\lambda p + (1-\lambda)s \in int(S_k) \subset int(S)$. □

Lemma 4. *Let $S = \bigcup_{k=1}^{K} S_k$ be a star-shaped set that consists of K disjoint sub-sets S_k, where $k \in [K]$. I then have*

$$\int_{\partial S} \varphi(\gamma)\mathbf{n}^t(\gamma)d\gamma = \sum_{k=1}^{K} \int_{\partial S_k} \varphi(\gamma)\mathbf{n}^t(\gamma)d\gamma, \tag{2.13}$$

where φ is a continuous function of γ, and $\mathbf{n}^t(\boldsymbol{\omega})$ is the unit outward normal to $\bigcup_{k=1}^{K} \partial S_k$ at $\boldsymbol{\omega}$.

Proof.

$$\begin{aligned} &\sum_{k=1}^{K} \int_{\partial S_k} \varphi(\gamma)\mathbf{n}^t(\gamma)d\gamma \\ =&\sum_{k=1}^{K} \left[\sum_{i \neq k} \int_{\partial(S_k \bigcap S_i)} \varphi(\gamma)\mathbf{n}^t(\gamma)d\gamma + \int_{\partial(S_k \bigcap S)} \varphi(\gamma)\mathbf{n}^t(\gamma)d\gamma \right] \end{aligned} \tag{2.14}$$

For any k and i such that $S_k \bigcap S_i = \varnothing$, the corresponding curve integral $\int_{\partial(S_k \bigcap S_i)} \varphi(\gamma)\mathbf{n}^t(\gamma)d\gamma$ is 0. On the other hand, for any k and i such that $S_k \bigcap S_i \neq \varnothing$, the corresponding curve integral $\int_{\partial(S_k \bigcap S_i)} \varphi(\gamma)\mathbf{n}^t(\gamma)d\gamma = -\int_{\partial(S_i \bigcap S_k)} \varphi(\gamma)\mathbf{n}^t(\gamma)d\gamma$ because of opposite unit outward

normal. Therefore, we have

$$\sum_{k=1}^{K}\int_{\partial S_k}\varphi(\gamma)\mathbf{n}^t(\gamma)d\gamma=\sum_{k=1}^{K}\left[\int_{\partial(S_k\bigcap S)}\varphi(\gamma)\mathbf{n}^t(\gamma)d\gamma\right]=\int_{\partial S}\varphi(\gamma)\mathbf{n}^t(\gamma)d\gamma \tag{2.15}$$

□

Now, we have enough tools to derive the main results in this section. The calculation of the gradient in homogeneous WSNs, (2.12), relies on Proposition 1.6 in [6], which requires star-shaped integral regions. In homogenous WSNs, the integral regions are star-shaped Voronoi diagrams. However, the integral regions in MWVDs can be non-star-shaped [102], and therefore I need to show how to calculate the partial derivatives in heterogeneous WSNs. The following proposition is needed to calculate these partial derivatives in $\mathbb{R}^2$.

Proposition 1. *In a heterogeneous sensor network including different types of sensors, let* $\mathbf{P}=[\mathbf{p}_1,\mathbf{p}_2,\cdots,\mathbf{p}_N]$ *be the sensor deployment, and* $W\in\mathbb{R}^2$ *be an arbitrary convex set. Let a series of functions* $\varphi_n:\mathbb{R}^2\times(a,b)\to\mathbb{R}$*, where* $n\in[N]$*, be continuous on* $\mathbb{R}^2\times(a,b)$*, continuously differentiable with respect to its second argument for all* $\mathbf{p}_n\in(a,b)^2$*, where* $n\in[N]$*, and almost all* $\boldsymbol{\omega}\in\mathbb{R}^2$*, and such that for each* $\mathbf{p}_n\in(a,b)^2$*, the maps* $q\mapsto\varphi_n(\boldsymbol{\omega},\mathbf{p}_n)$ *and* $q\mapsto\frac{\partial\varphi_n}{\partial x}(\boldsymbol{\omega},x)$ *are measurable, and integrable on* $\mathbb{R}^2$*. Then the function*

$$\int_{\boldsymbol{V}_n^H(\mathbf{P})\bigcap W}\varphi_n(q,\mathbf{p}_n)d\boldsymbol{\omega} \tag{2.16}$$

is continuously differentiable and

$$\frac{\partial\int_{\boldsymbol{V}_n^H(\mathbf{P})\bigcap W}\varphi_n(\boldsymbol{\omega},\mathbf{p}_n)d\boldsymbol{\omega}}{\partial\mathbf{p}_m}=\int_{\boldsymbol{V}_n^H(\mathbf{P})\bigcap W}\frac{\partial\varphi_n(\boldsymbol{\omega},\mathbf{p}_n)}{\partial\mathbf{p}_m}d\boldsymbol{\omega}+\int_{\partial[\boldsymbol{V}_n^H(\mathbf{P})\bigcap W]}\varphi_n(\gamma,\mathbf{p}_n)\mathbf{n}^t(\gamma)\frac{\partial\gamma}{\partial\mathbf{p}_m}d\boldsymbol{\omega}. \tag{2.17}$$

Proof. See Appendix A.1. □

Theorem 2.1. *Given a density function $f(\cdot)$ and N piece-wise continuous and differentiable functions $f^{(n)}, n \in [N]$, the multiple-center function $\mathcal{D}(\mathbf{P})$ is continuously differentiable on Ω^n, and for each $n \in [N]$,*

$$\begin{aligned}\frac{\partial \mathcal{D}(\mathbf{P})}{\partial \mathbf{p}_n} = &\int_{\boldsymbol{V}_n^H(\mathbf{P})} \frac{\partial}{\partial \mathbf{p}_n} f^{(n)}(\|\boldsymbol{\omega} - \mathbf{p}_n\|) f(\boldsymbol{\omega}) d\boldsymbol{\omega} \\ &+ \sum_{\alpha=1}^{l} \left[\left(f_\alpha^{(n)}(r_\alpha) - f_{\alpha+1}^{(n)}(r_\alpha) \right) \int_{arc_n(P, r_\alpha)} \mathbf{n}^t(\boldsymbol{\omega}) f(\boldsymbol{\omega}) d\boldsymbol{\omega} \right],\end{aligned} \tag{2.18}$$

where $arc_n(P, r_\alpha)$ consists of arcs in the boundary of $\boldsymbol{V}_n^H(\mathbf{P}) \bigcap \mathbb{B}(\mathbf{p}_n, r_\alpha)$, $\mathbf{n}^t(\boldsymbol{\omega})$ is the unit outward normal to $\bigcup_{\alpha=1}^{l} arc_n(P, r_\alpha)$ at point $\boldsymbol{\omega}$.

The proof is similar to that of Theorem 2 in [6] and then is omitted here. Particularly, when we fix the partitions as $\boldsymbol{V}_n^H(\mathbf{P}'), n \in [N]$, for a given deployment P', the constrained performance becomes

$$\mathcal{D}(\mathbf{P}, \boldsymbol{V}^H(\mathbf{P}')) = \sum_{n=1}^{N} \int_{\boldsymbol{V}_n^H(\mathbf{P}')} f^{(n)}(\|\boldsymbol{\omega} - \mathbf{p}_n\|) f(\boldsymbol{\omega}) d\boldsymbol{\omega} \tag{2.19}$$

By Proposition 1, for each $n \in [N]$, the partial derivative of $\mathcal{D}(P, \boldsymbol{V}^H(P'))$ becomes

$$\begin{aligned}\frac{\partial \mathcal{D}(\mathbf{P}, \boldsymbol{V}^H(\mathbf{P}'))}{\partial \mathbf{p}_n} = &\int_{\boldsymbol{V}_n^H(\mathbf{P}')} \frac{\partial}{\partial \mathbf{p}_n} f^{(n)}(\|\boldsymbol{\omega} - \mathbf{p}_n\|) f(\boldsymbol{\omega}) d\boldsymbol{\omega} \\ &+ \sum_{\alpha=1}^{l} \left[\left(f_\alpha^{(n)}(r_\alpha) - f_{\alpha+1}^{(n)}(r_\alpha) \right) \int_{ARC_n(\mathbf{P}, \mathbf{P}', r_\alpha)} \mathbf{n}^t(\boldsymbol{\omega}) f(\boldsymbol{\omega}) d\boldsymbol{\omega} \right]\end{aligned} \tag{2.20}$$

where $ARC_n(\mathbf{P}, \mathbf{P}', r_\alpha)$ consists of arcs in the boundary of $\boldsymbol{V}_n^H(\mathbf{P}') \bigcap \mathbb{B}(\mathbf{p}_n, r_\alpha)$. Obviously, $\frac{\partial \mathcal{D}(\mathbf{P})}{\partial \mathbf{p}_n}$ and $\frac{\mathcal{D}(\mathbf{P}, \boldsymbol{V}^H(\mathbf{P}'))}{\partial \mathbf{p}_n}$ will have the identical value at the deployment $\mathbf{P}'$. Next, I provide the corresponding gradients for specific performance functions defined in Section 2.1.

Negative weighted MSE model:

The performance function in this model is continuous without jump discontinuities. As a result, all the terms in the second summand of (2.18) vanish and we have

$$\frac{\partial \mathcal{D}(\mathbf{P})}{\partial \mathbf{p}_n} = 2\eta_n(\mathbf{p}_n - \mathbf{c}_n(\mathbf{P}))\mathbf{v}_n(\mathbf{P}), \tag{2.21}$$

where $\mathbf{v}_n(\mathbf{P}) = \int_{\boldsymbol{V}_n^H(\mathbf{P})} f(\boldsymbol{\omega})d\boldsymbol{\omega}$ and $\mathbf{c}_n(\mathbf{P}) = \frac{\int_{\boldsymbol{V}_n^H(\mathbf{P})} \boldsymbol{\omega} f(\boldsymbol{\omega})d\boldsymbol{\omega}}{\mathbf{v}_n(\mathbf{P})}$ are, respectively, the mass and the center of MWVD $\boldsymbol{V}_n^H(\mathbf{P})$ with respect to the density function $f(\cdot)$. The critical deployment set $\mathbf{P}^*(\infty) = \{\mathbf{P} = (\mathbf{p}_1, \ldots, \mathbf{p}_N) | \mathbf{p}_n = \mathbf{c}_n(\mathbf{P}), \forall n \in [N]$ is similar to the centroid Voronoi configurations [6] although the critical deployments in heterogeneous WSNs are based on MWVDs instead of Voronoi diagrams.

Binary coverage model:

The performance function in this model is a step function with one jump discontinuity. In this case, the first term in (2.18) vanishes and we have

$$\frac{\partial \mathcal{D}(\mathbf{P})}{\partial \mathbf{p}_n} = \int_{arc_n(\mathbf{P}, r_\alpha)} \mathbf{n}^t(\boldsymbol{\omega}) f(\boldsymbol{\omega})d\boldsymbol{\omega}. \tag{2.22}$$

Exponential coverage model:

The performance function in this model is continuous with one indifferentiable point. All the terms in the second summand of (2.18) vanish and we have

$$\frac{\mathcal{D}(\mathbf{P})}{\partial \mathbf{p}_n} = \int_{\boldsymbol{V}_n^H(\mathbf{P}) \setminus \mathbb{B}(\mathbf{p}_n, \frac{R_s}{\sqrt{\eta_n}})} 2(\mathbf{p}_n - \boldsymbol{\omega}) e^{-\eta_n \|\boldsymbol{\omega} - \mathbf{p}_n\|^2 + R_s^2} f(\boldsymbol{\omega})d\boldsymbol{\omega}. \tag{2.23}$$

Using the above gradients, we can design algorithms, such as gradient descent and Lloyd Algorithm, to optimize the performance. Moreover, zero-gradient can help us to check if a given deployment provides a local optimal performance (minimum distortion or maximum coverage area). Again, given the gradient of the objective function, we can get the critical deployment set as we discussed the beginning of this section. Furthermore, in order to

get a locally optimal solution, the designed algorithm should converge to a deployment $\mathbf{P} \in \mathbf{P}^*(R_c)$. In the next section, we will discuss how to design algorithms to achieve this convergence requirement.

2.3 Restraint Lloyd Algorithm and Deterministic Annealing Algorithm

In this section, we focus on the negative weighted MSE model because of its trackable critical deployment set. Two algorithms are designed to optimize the performance of this model in heterogeneous WSNs. Although the algorithms are designed for the negative weight MSE model, their main ideas can be applied to the other two models too. First, we quickly review the conventional Lloyd algorithm. Lloyd Algorithm has two basic steps in each iteration: (i) Sensor nodes move to their centroid; (ii) Partitioning is done by assigning the optimal partitions, MWVDs, to each sensor node. Lloyd Algorithm provides good performance and is simple enough to be implemented distributively. It converges to a critical deployment when the communication range is infinite [95], [96]. Unfortunately, Lloyd Algorithm also has three shortcomings. First, since maximizing multi-center performance is a non-convex optimization problem, Lloyd Algorithm ends at a local maximum rather than the global maximum. Second, Lloyd Algorithm results in a disconnected network when the spanning tree rooted at the AP does not include all sensors due to the limited communication range. Third, when WSNs are divided into several disconnected sub-graphs, some sensors cannot collect the complete local information and as a result Lloyd Algorithm is not feasible. In other words, since there is no global information available about the sensor locations, each sub-graph will run the algorithm separately. To deploy a network with full connectivity and smaller distortion (or larger coverage area), I add some restraints on sensors' movements. I design a class of algorithms based on the Lloyd algorithm, referred to as RL Algorithm.

2.3.1 Restrained Lloyd Algorithm

Before introducing the details of RL Algorithm, I introduce two concepts: (1) local performance; (2) desired region. First, the global performance is the sum of local performances defined by

$$\mathcal{D}_n(\mathbf{P}) = \int_{\boldsymbol{V}_n^H(\mathcal{H}(\mathbf{P}))} f^{(n)}(\|\boldsymbol{\omega} - \mathbf{p}_n\|) f(\boldsymbol{\omega}) d\boldsymbol{\omega}, n \in \mathcal{S}(\mathbf{P}). \tag{2.24}$$

Second, to define the desired region, without loss of generality, let us assume we are trying to move Sensor n at a given step. Our goal is to keep the connectivity of the backbone network after moving Sensor n. Therefore, all Sensor i's locations that result in connecting Sensor n to the backbone network is defined as Sensor n's desired region, denoted by $\mathbb{D}_n(\mathbf{P})$. In RL Algorithm, if Sensor n is in the backbone network, we will restrain its movement within its desired region. To achieve this goal, we need to find the desired region $\mathbb{D}_n(\mathbf{P})$. Given a deployment $\mathbf{P}$, if Sensor n from the backbone network is removed, the rest of the sensor nodes in the backbone network will be divided into K_n components: $U_{i1}(\mathbf{P}), U_{i2}(\mathbf{P}), \cdots, U_{iK_n}(\mathbf{P})$, where $U_{ij}(\mathbf{P})$ is a set of sensors included in the jth component. Then, we can calculate the desired region as

$$\mathbb{D}_n(\mathbf{P}) = \bigcap_{k=1}^{K_n} \left[\bigcup_{j \in U_{ik}} \mathbb{B}(\mathbf{p}_j, R_c) \right]. \tag{2.25}$$

I provide an example of the desired region that is not star shaped in Fig. 2.1. In this simple example, $n = 1$, i.e., I am trying to move Sensor 1, $R_c = 0.5$, $K_1 = 2$, $U_{11}(\mathbf{P}) = \{2, 3, 4, 5, 6\}$, and $U_{12}(\mathbf{P}) = \{7, 8, 9, 10, 11\}$. According to the definition of the desired region, the green overlaps between blue region ($\bigcup_{j=2}^{6} \mathbb{B}(\mathbf{p}_j, R_c)$) and yellow region ($\bigcup_{j=7}^{11} \mathbb{B}(\mathbf{p}_j, R_c)$) construct Sensor 1's desired region. As shown in Fig. 2.1, the desired region $\mathbb{D}_1(\mathbf{P})$ has two disconnected parts, indicating a non-star-shaped region.

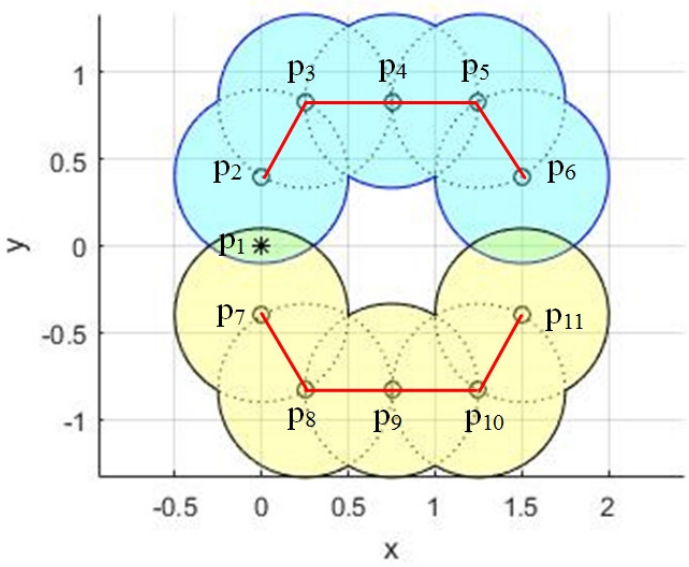


Figure 2.1: A non-starshaped desired region.

Since the desired region is primarily influenced by the neighboring sensor nodes, we can approximate it by

$$\widetilde{\mathbb{D}}_n(\mathbf{P}) = \bigcap_{k=1}^{K_n} \left[\bigcup_{j \in U_{nk} \bigcap \mathcal{N}_n(\mathbf{P})} \mathbb{B}(\mathbf{p}_j, R_c) \right], \tag{2.26}$$

where $\mathcal{N}_n(\mathbf{P})$ consists of Sensor n's neighbors when the deployment is $\mathbf{P}$. Note that the approximation in (2.26) can be calculated locally, but to calculate the exact desired region, one needs global information. Also, according to Lemma 3, the approximate desired region $\widetilde{\mathbb{D}}_n(\mathbf{P})$ is a star-shaped set.

Now, I provide the workflow of RL Algorithm as follows:

(i) Sensors in the backbone network move one by one. Every sensor in the backbone network calculates its own approximate desired region $\widetilde{\mathbb{D}}_n(\mathbf{P})$ and moves to a critical location with maximum local performance. Sensors outside the backbone network move randomly and check if there is a path to the AP. Unlike the conventional Lloyd algorithm, these new

locations may not be the centroid of the partition regions; (ii) The target area, Ω, is partitioned to MWVDs for sensors in the backbone network, $\mathcal{S}(\mathbf{P})$. More details are provided in Algorithm 1.

Algorithm 1 Restrained Lloyd Algorithm

Input: Target area Ω; Probability density function $f(\cdot)$; the initial sensor deployment $\mathbf{P}$; the stop threshold ϵ; the communication range R_c.
Output: Sensors deployment $\mathbf{P}$; Distortion $\mathcal{D}(\mathbf{P})$.

```
Calculate MWVDs, {V_n^H(P)}_{n∈[N]}
do
    Calculate the old distortion D_old = D(P)
    for n = 1 to N do
        Calculate Sensor n's ADR, {D̃_n(P)}
        Calculate the critical point q, closest point to c_n(P) within D̃_n(P)
        Update sensor deployment p_n = q
    end for
    Update MWVDs {V_n^H(P)}_{n∈[N]}
    Calculate the new distortion D_new = D(P)
while (D_old − D_new)/D_old > ε
```

The main difference between RL Algorithm and the Lloyd algorithm is in the first step. Given the derivatives of each local performance, the existing gradient descent can be used to find the critical locations in Step (i). For negative weighted MSE model, we can simplify the algorithm by moving Sensor n to the closest point to its centroid $\mathbf{c}_n(\mathbf{P})$ within $\widetilde{\mathbb{D}}_n(\mathbf{P})$. In what follows, we show that Step (i) in RL Algorithm provides the smallest local distortion. According to the parallel axis theorem, the local performance for negative weighted MSE model can be rewritten as

$$\mathcal{D}_n(\mathbf{P}) = -\int_{\boldsymbol{V}_n^H(\mathcal{H}(\mathbf{P}))} \|\boldsymbol{\omega} - \mathbf{c}_n(\mathbf{P})\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} - \int_{\boldsymbol{V}_n^H(\mathcal{H}(\mathbf{P}))} f(\boldsymbol{\omega}) d\boldsymbol{\omega} \cdot \|\mathbf{p}_n - \mathbf{c}_n(\mathbf{P})\|^2, n \in \mathcal{S}(\mathbf{P}) \tag{2.27}$$

where $\mathbf{c}_n(\mathbf{P}) = \frac{\int_{\boldsymbol{V}_n^H(\mathcal{H}(\mathbf{P}))} \boldsymbol{\omega} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}{\int_{\boldsymbol{V}_n^H(\mathcal{H}(\mathbf{P}))} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}$ is the centroid of the partition region $\boldsymbol{V}_n^H(\mathcal{H}(\mathbf{P}))$ with respect to the probability density function. Both $\int_{\boldsymbol{V}_n^H(\mathcal{H}(\mathbf{P}))} \|\boldsymbol{\omega} - \mathbf{c}_n(\mathbf{P})\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega}$ and $\int_{\boldsymbol{V}_n^H(\mathcal{H}(\mathbf{P}))} f(\boldsymbol{\omega}) d\boldsymbol{\omega}$ are constants when the integral area $\boldsymbol{V}_n^H(\mathcal{H}(\mathbf{P}))$ is fixed. In other words, the local distor-

tion is a monotonously increasing function of $\|\mathbf{p}_n - \mathbf{c}_n(\mathbf{P})\|$, i.e., the sensor's distance to its centroid. Therefore, the movements in Step (i) minimize the local distortion. As the sum of local distortions, the global distortion will not increase. Since the sequence of the global distortion values is a non-increasing sequence with a lower bound of zero, it will converge. It is self-evident that the sensor deployment in RL Algorithm converges to an element in $\mathbf{P}^*(R_c)$.

I also show that RL Algorithm guarantees the connectivity of the network with high probability after enough number of iterations. Note that once a sensor node finds a path to the AP, RL Algorithm will keep it in the backbone network. Intuitively, as we have more iterations, the sensors outside the backbone network will move randomly and eventually connect to the AP as well. Quantitatively, for the deployment after k iterations, the area in which a sensor can communicate with the backbone network can be calculated by $A_k = \text{AREA}\left(\Omega \bigcap \left[\bigcup_{n \in backbone} \mathbb{B}(\mathbf{p}_n, R_c)\right]\right)$. Then, the probability that a sensor outside the backbone network is not connected to the AP in its next move is $\frac{\text{AREA}_{(\Omega)-A_k}}{\text{AREA}_{(\Omega)}}$. After K iterations, the probability that a sensor is still out of the backbone network can be calculated by $P_{out}(K) = \prod_{k=1}^{K} \left[\frac{\text{AREA}_{(\Omega)-A_k}}{\text{AREA}_{(\Omega)}}\right] \leq \Omega \left[\frac{\text{AREA}_{(\Omega)-\min A_k}}{\text{AREA}_{(\Omega)}}\right]^K$ and then $\lim_{K\to\infty} P_{out}(K) = 0$ because of $\min A_k > 0$. In other words, as long as the number of iterations is large enough, almost all sensor nodes will be included in the backbone network, indicating full connectivity, with high probability.

2.3.2 Deterministic Annealing Algorithm

Like any other steepest-descent algorithm, RL Algorithm converges to a local optimum. One approach to improve the sub-optimal solution or find the global optimal solution, is to use annealing methods. Simulated Annealing (SA) [103], [104] is a method in which a candidate sensor movement is generated randomly. However, SA ignores the characteristics of the objective function and requires burdensome computations. In this paper, I design a DA algorithm which combines RA with annealing to minimize the distortion. Unlike SA, the

proposed DA generates two new sensor positions deterministically at each iteration; however, choose one of the two options randomly.

Like RL Algorithm, DA Algorithm iterates between two steps. The second step is identical to that of RL Algorithm. In the first step, the algorithm creates two candidate locations for each node in the backbone network. One candidate is the RL Algorithm's candidate that minimizes the local distortion. On the other hand, the second candidate increases the local distortion. It is easy to show that to maximize the local distortion of Sensor n in the backbone network for negative weighted MSE model, one should move it to the point q on the boundary of the desired region $\widetilde{\mathbb{D}}_n(\mathbf{P})$ that has the largest distance to the centroid $\mathbf{c}_n(\mathbf{P})$. But the goal of the second candidate is to increase the distortion and not necessarily maximize it. Moreover, the performance is more sensitive to the sensors with smaller cost parameters. In order to avoid increasing distortion too fast, Sensor n is moved to the point $\mathbf{p}_n + \frac{\eta_n}{\min_j(\eta_j)}(q - \mathbf{p}_n)$. The algorithm will choose the first candidate with probability p and increases p from 0 to 1. Otherwise, the algorithm will choose the second candidate. In this algorithm, the probability p is increased in proportion to $\log k$, where k is the iteration number and for the last M iterations we force the probability $p = 1$. More details of DA are provided in Algorithm 2.

Like RL Algorithm, DA Algorithm guarantees connectivity and convergence. The proof is similar to that of RL Algorithm and is omitted. Furthermore, both RL and DA can be easily extended to non-convex environments [90]–[92]. we only need to revise the desired region to

$$\mathbb{D}'_n(\mathbf{P}) = \bigcap_{k=1}^{K_n} \left[\bigcup_{j \in U_{ik}} \mathbb{B}(\mathbf{p}_j, R_c) \right] \bigcap \Omega_o^c, \tag{2.28}$$

where Ω_o is the obstacle region. To avoid the geometric calculation of $\mathbb{D}'_n(\mathbf{P})$, one can calculate $\widetilde{\mathbb{D}}'_n(\mathbf{P}) = \bigcap_{k=1}^{K_n} \left[\bigcup_{j \in U_{ik} \bigcap \mathcal{N}_n(\mathbf{P})} \mathbb{B}(\mathbf{p}_j, R_c) \right] \bigcap \Omega_o^c$ and then move Sensor n towards the critical point by TangentBug [91].

Algorithm 2 Deterministic Annealing Algorithm

Input: Target area Ω; Probability density function $f(\cdot)$; the initial sensor deployment $\mathbf{P}$; the number of iterations K; the communication range R_c.

Output: Sensors deployment $\mathbf{P}$; Distortion $\mathcal{D}(\mathbf{P})$.

Calculate MWVDs, $\{\boldsymbol{V}_n^H(\mathbf{P})\}_{n\in[N]}$
for $k = 1$ to K **do**
    Calculate the probability of doing RL , p.
    Generate a random value $v \in [0, 1]$
    for $n = 1$ to N **do**
        Calculate Sensor n's ADR, $\{\widetilde{\mathbb{D}}_n(\mathbf{P})\}$
        if $v \leq p$ **then**
            Calculate the critical point q, closest point to $\mathbf{c}_n(\mathbf{P})$ within $\widetilde{\mathbb{D}}_n(\mathbf{P})$
        else
            Calculate the critical point q, farthest point to $\mathbf{c}_n(\mathbf{P})$ within $\widetilde{\mathbb{D}}_n(\mathbf{P})$
        end if
        Update sensor deployment $\mathbf{p}_n = q$
    end for
    Update MWVDs $\{\boldsymbol{V}_n^H(\mathbf{P})\}_{n\in[N]}$
    Update Distortion $\mathcal{D}(\mathbf{P})$.
end for

2.4 Performance Evaluation

I compare the performance of DA Algorithm, RL Algorithm, Lloyd Algorithm [6], OPTGA [36], and VFA [36] using the negative weighted MSE model. I provide simulation in three sensor networks: (1) WSN1: A homogeneous WSN in which all sensors have the same cost parameter $\eta_n = 1, \forall n \in [16]$; (2) WSN2: A heterogeneous WSN including 2 kinds of sensors: four strong sensors with $\eta_n = 1$ and twelve weak sensors with $\eta_n = 16$; (3) WSN3: A heterogeneous WSN including three kinds of sensors: two strong sensors with $\eta_n = 1$, four medium sensors with $\eta_n = 4$ and ten weak sensors with $\eta_n = 16$. Sixteen sensors are provided in each sensor network. The parameter values are summarized in Table 2.1. The AP is chosen from the sixteen sensors randomly. However, when I report the distortion or coverage area for the Lloyd algorithm, I calculate the distortion and the coverage area of each connected subgraph and report the best. Obviously, this will be advantageous for the Lloyd algorithm, but my proposed algorithms still outperform the Lloyd algorithm. I use

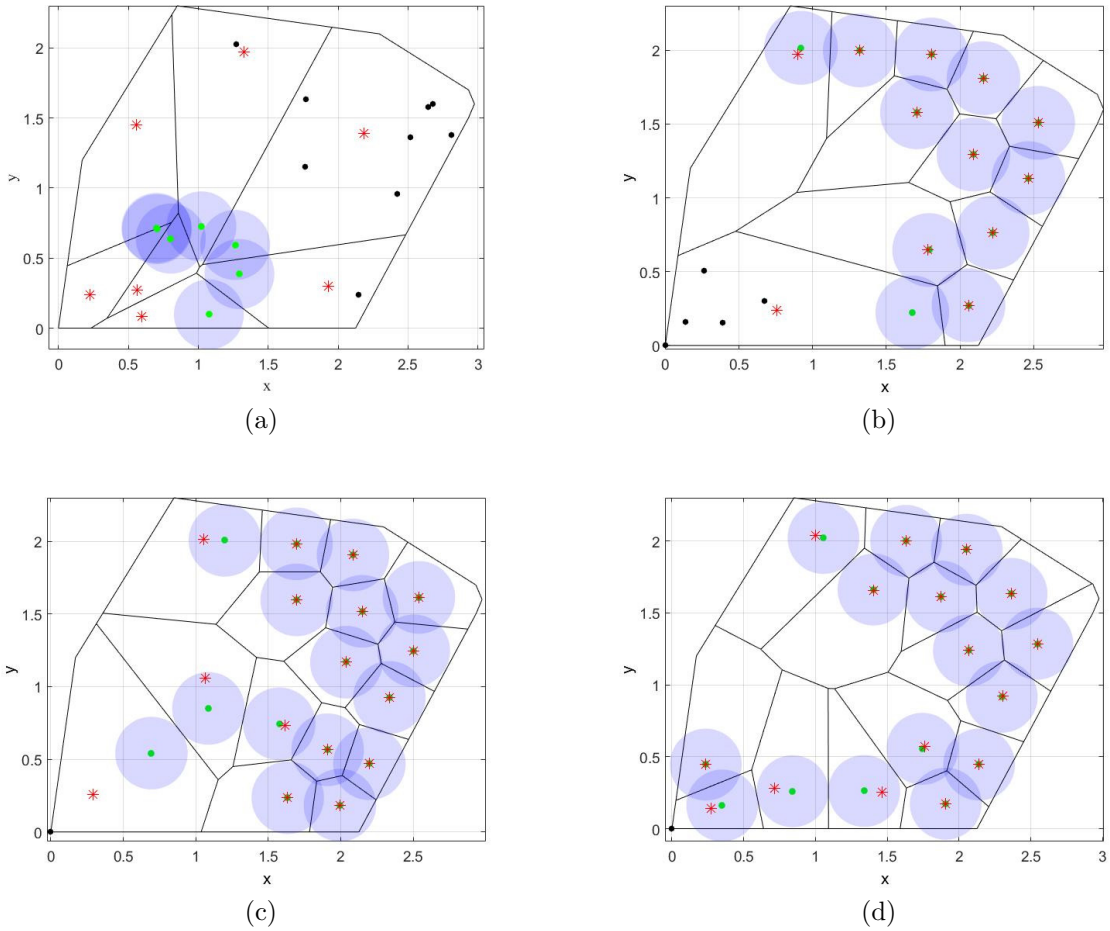


Figure 2.2: Sensor deployments in WSN1. (a) The initial sensor deployment and the corresponding Voronoi regions. (b) The final deployment of Lloyd Algorithm. (c) The final deployment of RL Algorithm. (d) The final deployment of DA Algorithm.

ten random initial deployments for each algorithm. To have a fair comparison, I consider the same target domain Ω as in [6]. Ω is determined by the polygon vertices (0,0), (2.125,0), (2.9325,1.5), (2.975,1.6), (2.9325,1.7), (2.295,2.1), (0.85,2.3), (0.17,1.2).

Table 2.1: Simulation Parameters of Static Wireless Sensor Networks

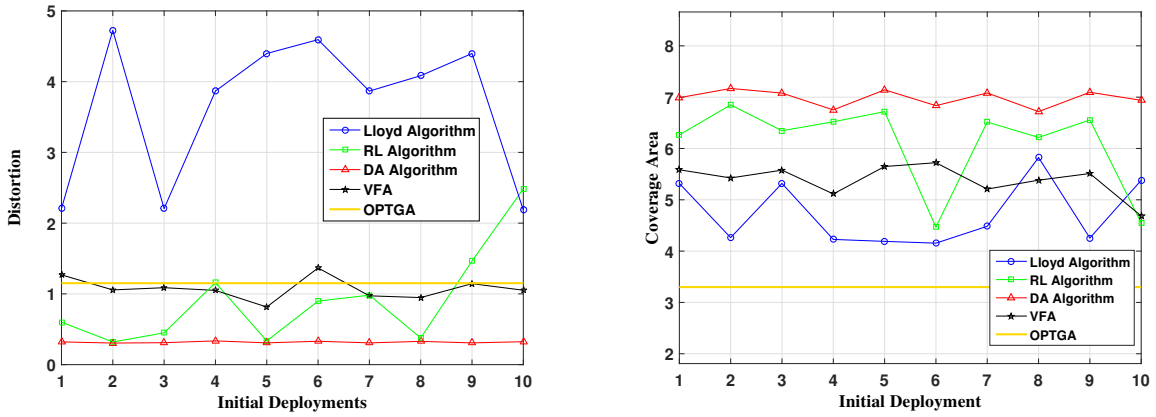
Parameters	WSN1	WSN2	WSN3
n	16	16	16
$\eta_1 - \eta_2$	1	1	1
$\eta_3 - \eta_4$	1	1	4
$\eta_5 - \eta_6$	1	16	4
$\eta_7 - \eta_{16}$	1	16	16
R_c	0.5	0.5	0.5
$R_s = R_c/2$	0.25	0.25	0.25

The distribution of the events is also the same as [6]. The probability density function is the sum of five Gaussian functions of the form $5exp(6(-(x-x_{center})^2-(y-y_{center})^2))$. The centers (x_{center}, y_{center}) are (2,0.25), (1,2.25), (1.9,1.9), (2.35,1.25) and (0.1,0.1). I use 0.5 as the communication range R_c. Also, when reporting the coverage area using (2.7), I use 0.25 as the sensing range R_s. In DA Algorithm, the first candidate is accepted at the kth iteration by a probability of $p(k) = log(k+1)/log(K+1)$, where K is the number of regular iterations. Additional $M = 25$ iterations are used in DA Algorithm to avoid ending with a process that increases the local distortions. Given the final deployment generated by different iterative algorithms, we can calculate the coverage area in the binary coverage model. Specially, Sensor n has a sensing range $\frac{R_s}{\sqrt{\eta}_n}$ in the heterogeneous WSNs. In following figures, strong sensors, medium and weak sensors in the backbone network are, respectively, denoted by green circles, yellow circles and magenta circles. Sensors out of the backbone network are denoted by black dots. The corresponding centroid for sensors in the backbone network are denoted by red stars. The radius of each blue circle is $R_c/2 = 0.25$. The maximum iterations is set to $K = 500$ and the stop threshold is set to $\epsilon = 10^{-6}$.

Figs. 2.2a and 2.2b show one example of the initial and the finial deployments of Lloyd Algorithm in WSN1. Lloyd Algorithm assumes an infinite communication range and requires the global knowledge of the sensor locations. Otherwise, disconnected sub-graphs run Lloyd Algorithm independently and there is no guarantee for convergence. Nonetheless, the calculation of the final distortion only considers sensors in the backbone network. In the final deployment of the example in Fig. 2.2b, there are four sensors disconnected from the backbone network, resulting in a large distortion $\mathcal{D}(\mathbf{P}) = 2.21$. Fig. 2.2c shows the outcome of RL Algorithm in WSN1. After 500 iterations, the distortion is decreased from 11.30 to 0.60. Simultaneously, the coverage area is increased from 0.15 to 6.26 and the final deployment is connected. Fig. 2.2d shows the final deployment of DA Algorithm in WSN1. After 500 iterations, the distortion is decreased from 11.30 to 0.32, which is better than that of RL Algorithm. Simultaneously, the coverage area is increased from 0.15 to 6.99 and full

connectivity is provided. Unlike Lloyd Algorithm, both RL Algorithm and DA Algorithm guarantee connectivity.

Fig. 2.3a illustrates the performance of VFA, Lloyd Algorithm, RL Algorithm, and DA Algorithm for 10 random initial deployments. I also add a line for the performance of OPTGA [6] which is independent of the initial deployment. While [6] reports the average of the 100 surviving deployments, the line in Fig. 3 reports the best performance. As can be seen from the figure, the performance of DA Algorithm is not sensitive to the initial deployment. In other words, DA Algorithm avoids most poor local minimum solutions. Fig. 2.3a shows that DA Algorithm has the best performance among the five algorithms.

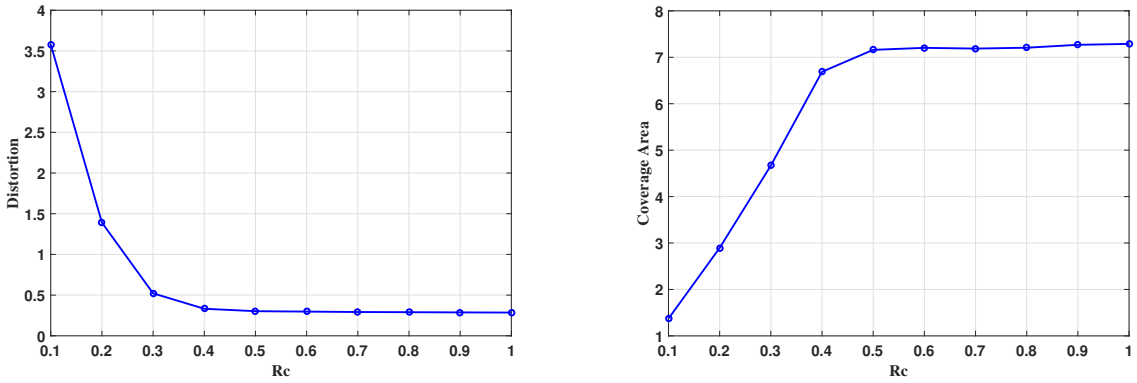


(a) (b)

Figure 2.3: Comparison of performance in WSN1. (a) Comparison of distortion for different algorithms in WSN1. (b) Comparison of coverage area for different algorithms in WSN1.

Fig. 2.3b compares the final coverage area of RL Algorithm and DA Algorithm with that of Lloyd Algorithm, OPTGA and VFA. In most cases, decreasing the distortion results in increasing the coverage area as well. Intuitively, this behavior can be explained by considering coverage area as a hard-decision version of distortion. Next, the relationship between performance (distortion and coverage area) and communication range R_c in homogeneous WSN1 using DA Algorithm is depicted in Fig. 2.4.

Figs. 2.5a and 2.5b show one example of the initial and the finial deployments of Lloyd Algorithm in WSN2. As usual, the final distortion only considers sensors in the backbone



(a) (b)

Figure 2.4: Relationship between performance and R_c in WSN1. (a) Relationship between distortion and R_c in WSN1. (b) Relationship between coverage and R_c in WSN1.

network. Initially, two strong sensors and four weak sensors are consisted in the backbone network shown in Fig. 2.5a. In Fig. 2.5b, only one strong sensor and four weak sensors are included in the backbone network, resulting in a large distortion $\mathcal{D}(\mathbf{P}) = 12.67$ which is only 0.48 smaller than the initial distortion. Fig. 2.5c shows the outcome of RL Algorithm in WSN2. After 500 iterations, the distortion is decreased from 13.15 to 5.52. Simultaneously, the coverage area is increased from 0.08 to 1.46 and the final deployment is connected. Fig. 2.5d shows the final deployment of DA Algorithm in WSN2. After 500 iterations, the distortion is decreased from 13.15 to 1.09, which is better than that of RL Algorithm. The coverage area is increased from 0.08 to 2.48 and full connectivity is provided.

Figs. 2.6a and 2.6b show one example of the initial and the finial deployments of Lloyd Algorithm in WSN3. Initially, one strong sensor, two medium sensors and four weak sensors are consisted in the backbone network shown in Fig. 2.6a. In Fig. 2.6b, two strong sensors, one medium sensors and one weak sensors are disconnected from the backbone network, resulting in a large distortion $\mathcal{D}(\mathbf{P}) = 19.83$. Fig. 2.6c shows the outcome of RL Algorithm in WSN3. After 500 iterations, the distortion is decreased from 10.96 to 3.22. Simultaneously, the coverage area is increased from 0.08 to 1.51 and the final deployment is connected. Fig. 2.6d shows the final deployment of DA Algorithm in WSN3. After 500 iterations, the distortion is decreased from 10.96 to 1.35, which is better than that of RL Algorithm.

Figs. 2.7a and 2.7b illustrate the distortion and coverage area of five different algorithms (OPTGA, VFA, Lloyd Algorithm, RL Algorithm, and DA Algorithm) in WSN2. The DA Algorithm provides both lower distortion and higher coverage area compare to the other four algorithms. Moreover, RL Algorithm also provides lower distortion compare VFA and Lloyd Algorithm due to its connectivity guarantee. Figs. 2.8a and 2.8b show similar performance comparison in WSN3. The trends in heterogeneous WSNs 2 and 3 are similar to those in homogeneous WSN1. In other words, RL Algorithm and DA Algorithm provide the performance improvement in both homogeneous and heterogeneous WSNs.

Furthermore, Table 2.2 provides the statistics of the performance for different algorithms

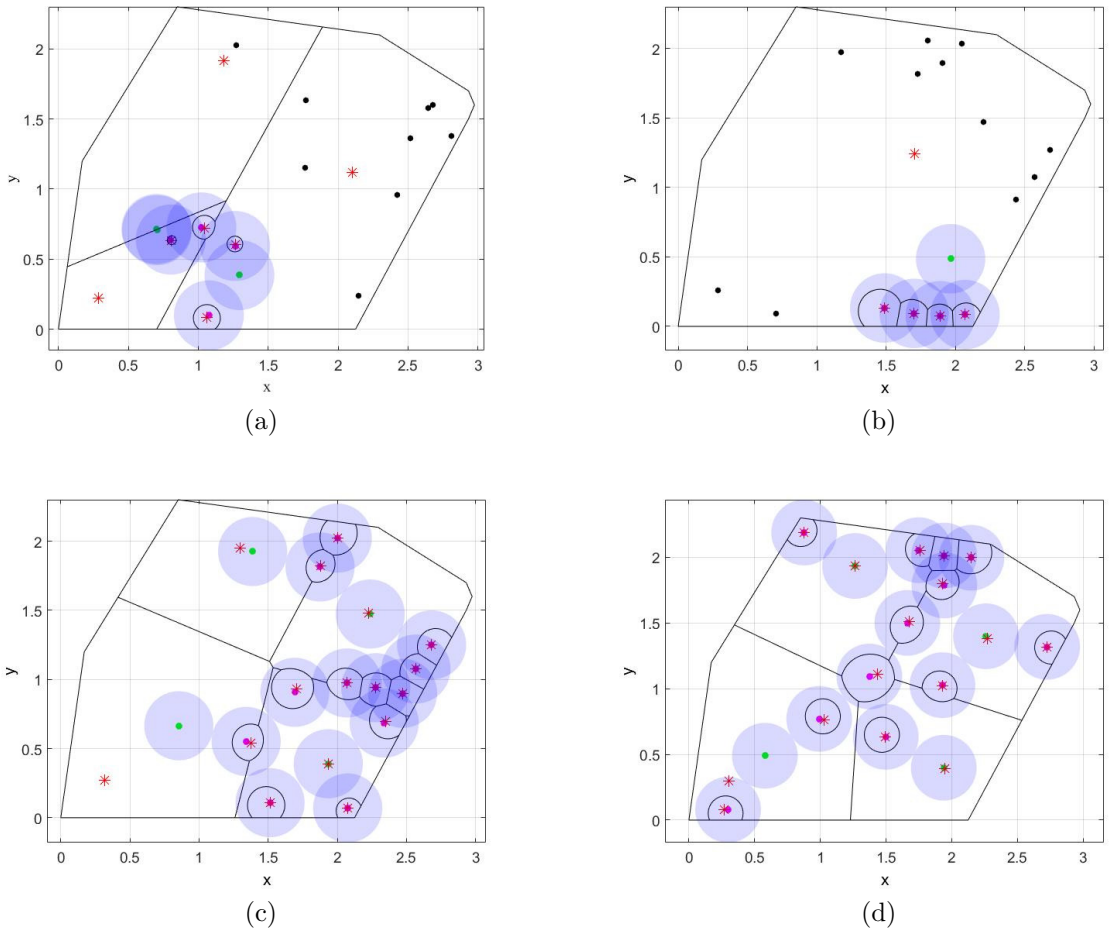


Figure 2.5: Sensor deployments in WSN2. (a) The initial sensor deployment and the corresponding weighted Voronoi regions. (b) The final deployment of Lloyd Algorithm. (c) The final deployment of RL Algorithm. (d) The final deployment of DA Algorithm.

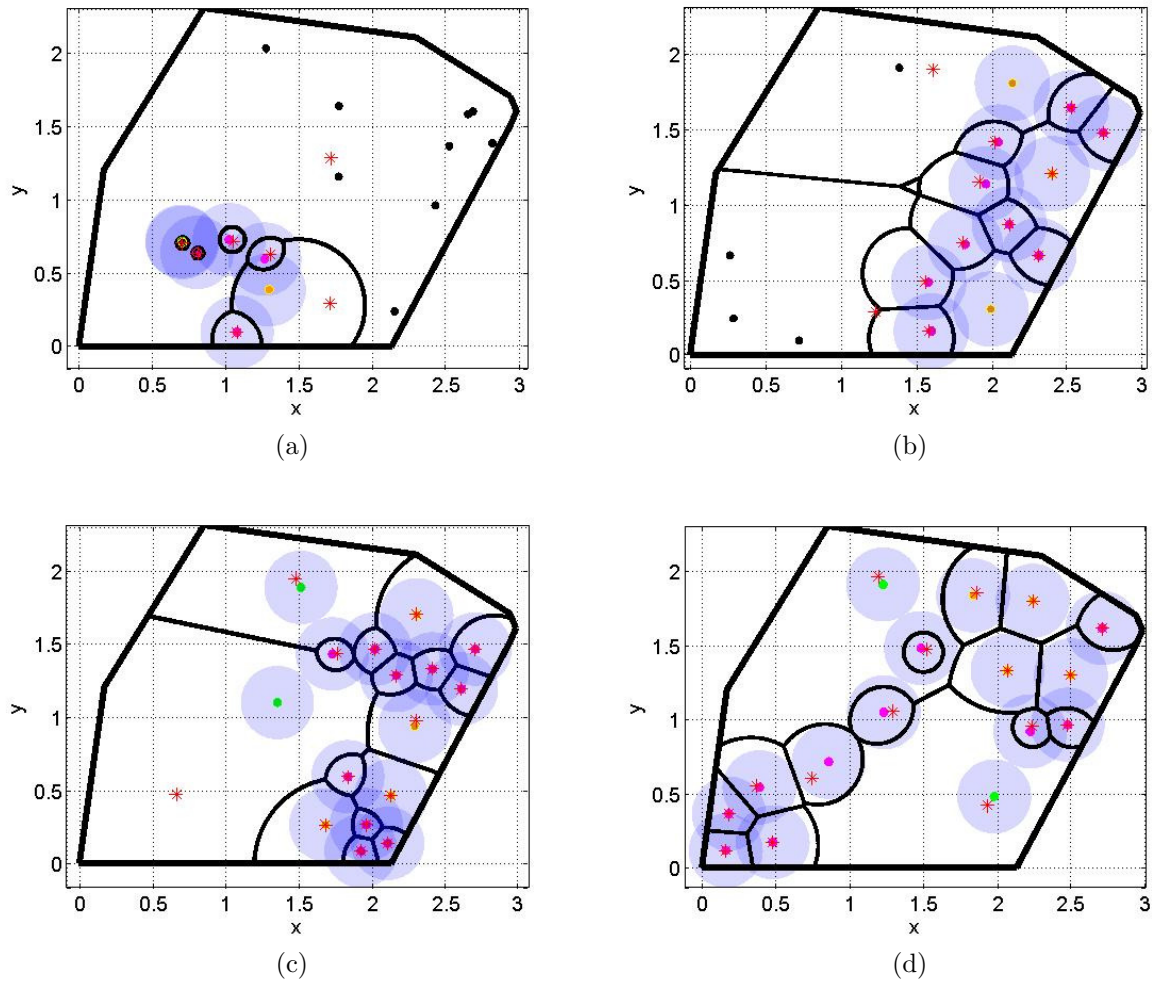
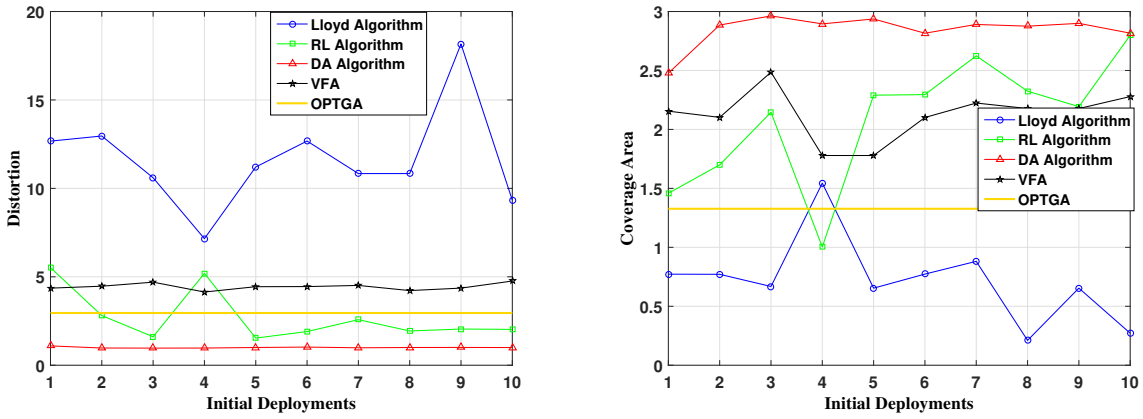


Figure 2.6: Sensor deployments in WSN3. Figure (a) The initial deployment and corresponding weighted Voronoi regions. (b) The final deployment of Lloyd Algorithm. (c) The final deployment of RL Algorithm. (d) The final deployment of DA Algorithm.

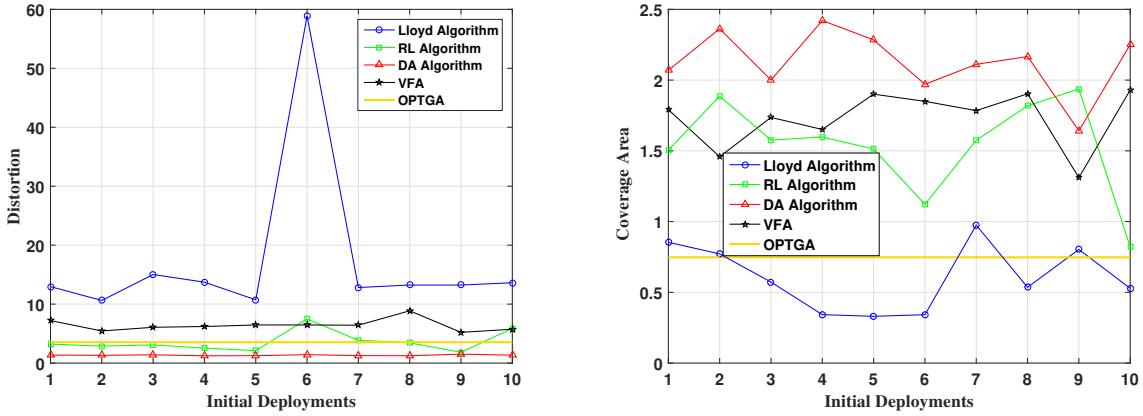
after 500 iterations. The statistics is reported for the 100 surviving deployments of the OPTGA and over the 10 different initial deployments for the other algorithms. For VFA, Lloyd, RL, and DA, a smaller standard deviation indicates less sensitivity to the initial condition. Again, DA Algorithm provides the best performance in both Homogeneous and Heterogeneous WSNs.

Fig. 2.9 shows the sensor deployments in a non-convex environment that includes two obstacles. Sensors can only move out of obstacles. Figs. 2.9a and 2.9b show one example of the initial and the finial deployments of Lloyd Algorithm in WSN2 with two obstacles.



(a) (b)

Figure 2.7: Comparison of performance in WSN2. (a) Comparison of distortion for different algorithms in WSN2. (b) Comparison of coverage for different algorithms in WSN2.



(a) (b)

Figure 2.8: Comparison of performance in WSN3. (a) Comparison of distortion for different algorithms in WSN3. (b) Comparison of coverage area for different algorithms in WSN3.

Table 2.2: Distortion Comparison
(μ: the average distortion, σ: the standard deviation)

Scenario	VFA		OPTGA		Lloyd		RL		DA	
	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
WSN1	1.08	0.15	1.18	0.01	3.65	1.04	0.91	0.68	**0.32**	**0.01**
WSN2	4.44	0.19	3.04	0.51	11.76	2.80	2.72	1.45	**1.00**	**0.04**
WSN3	6.41	1.04	3.54	0.11	17.49	14.61	3.63	1.75	**1.33**	**0.09**

Initially, two strong sensors and four weak sensors are consisted in the backbone network shown in Fig. 2.9a. In Fig. 2.9b, only one strong sensor and four weak sensors are included

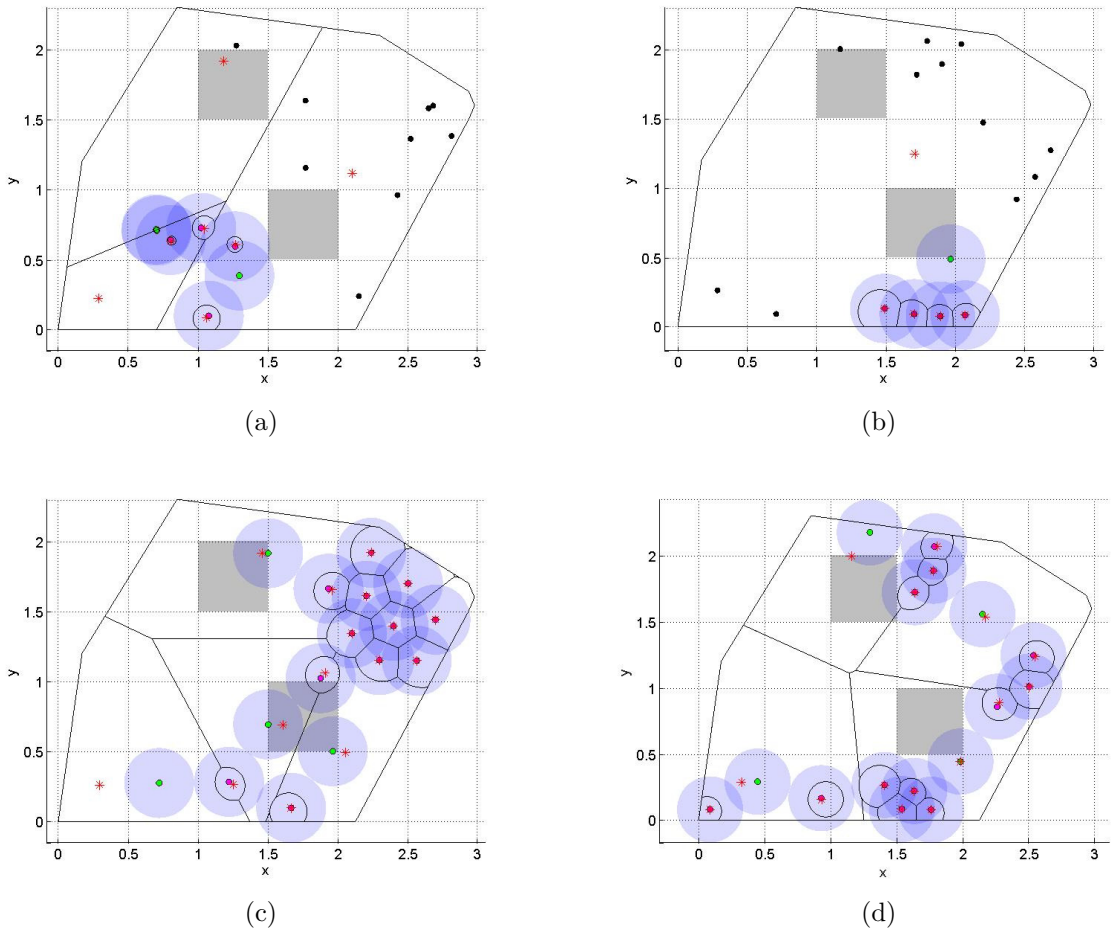
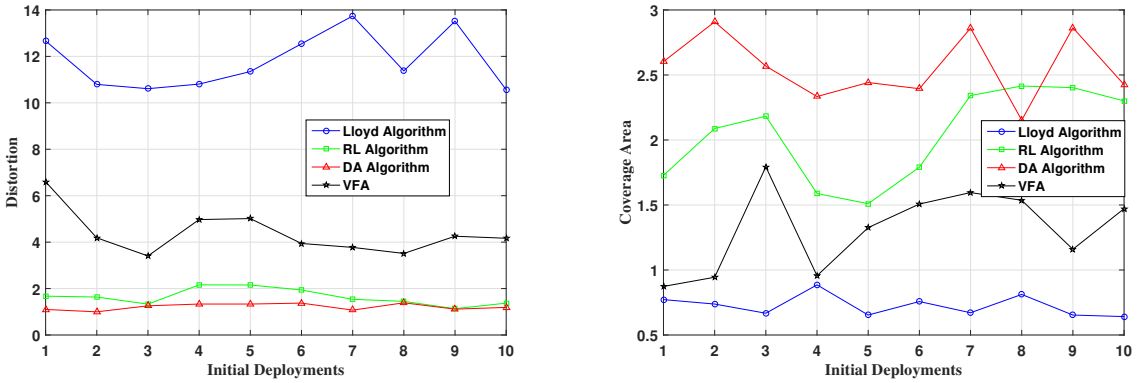


Figure 2.9: Sensor deployments in WSN2. (a) The initial sensor deployment and the corresponding weighted Voronoi regions. (b) The final deployment of Lloyd Algorithm. (c) The final deployment of RL Algorithm. (d) The final deployment of DA Algorithm. Two obstacles are shown by the gray region.

in the backbone network. Fig. 2.9c shows the outcome of RL Algorithm in WSN2 with two obstacles. After 500 iterations, the distortion is decreased from 13.15 to 1.67. Simultaneously, the coverage area is increased from 0.08 to 1.73 and the final deployment is connected. Fig. 2.9d shows the final deployment of DA Algorithm in WSN2 with two obstacles. After 500 iterations, the distortion is decreased from 13.15 to 1.10, which is better than that of RL Algorithm. Simultaneously, the coverage area is increased from 0.08 to 2.60 and full connectivity is provided.

Figs. 2.10a and 2.10b illustrate the distortion and coverage area of four different algorithms (VFA, Lloyd Algorithm, RL Algorithm, and DA Algorithm) for 10 random initial deploy-

ments in WSN2 with two obstacles. The DA Algorithm provides both lower distortion and higher coverage area compare to the other three algorithms.



(a) (b)

Figure 2.10: Comparison of performance in WSN2. (a) Comparison of distortion for different algorithms in WSN2. (b) Comparison of coverage for different algorithms in WSN2.

Chapter 3

3-Dimension Node Deployment in Wireless Sensor Networks

In many quantization problems, the distortion function is given by the Euclidean metric to measure the distance of a source sample to any given reproduction point of the quantizer. We will in this work regard distortion functions, which are additively and multiplicatively weighted for each reproduction point resulting in a heterogeneous quantization problem, as used for example in deployment problems of sensor networks. Whereas, normally in such problems, the average distortion is minimized for given weights (parameters), we will optimize the quantization problem over all weights, i.e., we tune or control the distortion functions in our favor. For a uniform source distribution in one-dimension, I derive the unique minimizer, given as the uniform scalar quantizer with an optimal common weight. By numerical simulations, I demonstrate that this result extends to two-dimensions where asymptotically the parameter optimized quantizer is the hexagonal lattice with common weights. As an application, I will determine the optimal deployment of unmanned aerial vehicles (UAVs) to provide a wireless communication to ground terminals under a minimal communication power cost. Here, the optimal weights relate to the optimal flight heights of the UAVs.

3.1 System Model and Problem Formulation

By $[N] = \{1, 2, \ldots, N\}$ I denote the first N natural numbers, $\mathbb{N}$. We will write real numbers in $\mathbb{R}$ by small letters and row vectors by bold letters. The Euclidean norm of $\mathbf{x}$ is given by $\|\mathbf{x}\| = \sqrt{\sum_n x_n^2}$. We denote by $\boldsymbol{V}^c$ the complement of the set $\boldsymbol{V} \subset \mathbb{R}^d$. The positive real numbers are denoted by $\mathbb{R}_+ := \{a \in \mathbb{R} | a > 0\}$.

To motivate the concept of parameterized distortion measures, we will investigate the deployment of N UAVs positioned at $\mathbf{P} = \{\mathbf{p}_1, \ldots, \mathbf{p}_N\} \subset (\Omega \times \mathbb{R}_+)^N$ to provide a wireless communication link to ground terminals (GTs) in a given target region $\Omega \subset \mathbb{R}^d$. Here, the nth UAV's position, $\mathbf{p}_n = (\mathbf{q}_n, h_n)$, is given by its ground position $\mathbf{q}_n = (x_n, y_n) \in \Omega$, representing the quantization point, and its flight height h_n, representing its distortion parameter. The optimal UAV deployment is then defined by the minimum average communication power (distortion) to serve GTs distributed by a density function f in Ω with a minimum given data rate R_b. Hereby, each GT will select the UAV which requires the smallest communication power, resulting in so called generalized Voronoi (quantization) regions of Ω, as used in [8], [13], [17], [18], [32], [34], [53], [105]. We also assume that the communication between all users and UAVs is orthogonal, i.e., separated in frequency or time (slotted protocols).

In the recent decade, UAVs with directional antennas have been widely studied in the literature [106]–[111], to increase the efficiency of wireless links. However, in [106]–[111], the antenna gain was approximated by a constant within a 3dB beamwidth and set to zero outside. This ignores the strong angle-dependent gain of directional antennas, notably for low-altitude UAVs. To obtain a more realistic model, I will consider an antenna gain which depends on the actual radiation angle $\theta_n \in [0, \frac{\pi}{2}]$ from the nth UAV at $\mathbf{p}_n$ to a GT at $\boldsymbol{\omega}$, see Fig. 3.1. To capture the power falloff versus the line-of-sight distance d_n along with the random attenuation and the path-loss, I adopt the following propagation model [112, (2.51)]

$$PL_{dB} = 10 \log_{10} K - 10\boldsymbol{\alpha} \log_{10}(d_n/d_0) - \psi_{dB}, \tag{3.1}$$

where K is a unitless constant depending on the antenna characteristics, d_0 is a reference distance, $\boldsymbol{\alpha} \geq 1$ is the terrestrial path-loss exponent, and ψ_{dB} is a Gaussian random variable following $\mathcal{N}\left(0, \sigma^2_{\psi_{dB}}\right)$. This air-to-ground or terrestrial path-loss model is widely used for UAV basestations path-loss models [113]. Practical values of $\boldsymbol{\alpha}$ are between 2 and 6 and depend on the Euclidean distance of GT $\boldsymbol{\omega}$ and UAV $\mathbf{q}_n$

$$d_n(\boldsymbol{\omega}) = d(\mathbf{p}_n, (\boldsymbol{\omega}, 0)) = \sqrt{\|\mathbf{q}_n - \boldsymbol{\omega}\|^2 + h_n^2} = \sqrt{(x_n - x)^2 + (y_n - y)^2 + h_n^2}. \tag{3.2}$$

For common practical measurements, see for example [114]. Typically maximal heights for UAVs are $<$ 1000m, due to flight zone restrictions of aircrafts. Hence, the received power at UAV n can be represented as $P_{RX} = P_{TX} G_{TX} G_{RX} K d_0^{\boldsymbol{\alpha}} d_n^{-\boldsymbol{\alpha}}(\boldsymbol{\omega}) 10^{-\frac{\psi_{dB}}{10}}$, where G_{TX} and G_{RX} are the antenna gains of the transmitter and the receiver, respectively. Here, we assume perfect omnidirectional transmitter GT antennas with an isotropic gain and directional receiver UAV antennas. The angle dependent antenna gains are

$$G_{\mathrm{GT}} > 0 \quad , \quad G_{\mathrm{UAV}} = \cos(\theta_n) = h_n / d_n(\boldsymbol{\omega}), \tag{3.3}$$

see [115, p.52]. The combined antenna intensity is then proportional to $G = G_{\mathrm{UAV}} G_{\mathrm{GT}} K$, see Fig. 3.1. Accordingly, the received power can be rewritten as

$$P_{RX} = P_{TX} h_n G_{\mathrm{GT}} K d_0^{\boldsymbol{\alpha}} d_n^{-\boldsymbol{\alpha}-1}(\boldsymbol{\omega}) 10^{-\frac{\psi_{dB}}{10}}. \tag{3.4}$$

To achieve a reliable communication between GT and UAV with bit-rate at least R_b for a channel bandwidth B and noise power density N_0, the Shannon formula requires $B \log_2\left(1 + \frac{P_{RX}}{B N_0}\right) \geq R_b$. The minimum transmission power to UAV $\mathbf{q}_n$ is then given by

$$P_{TX} = \left(2^{\frac{R_b}{B}} - 1\right) B N_0 d(\mathbf{p}_n, (\boldsymbol{\omega}, 0))^{\boldsymbol{\alpha}+1} 10^{\frac{\psi_{dB}}{10}} (h_n G_{\mathrm{GT}} K d_0^{\boldsymbol{\alpha}})^{-1} \tag{3.5}$$

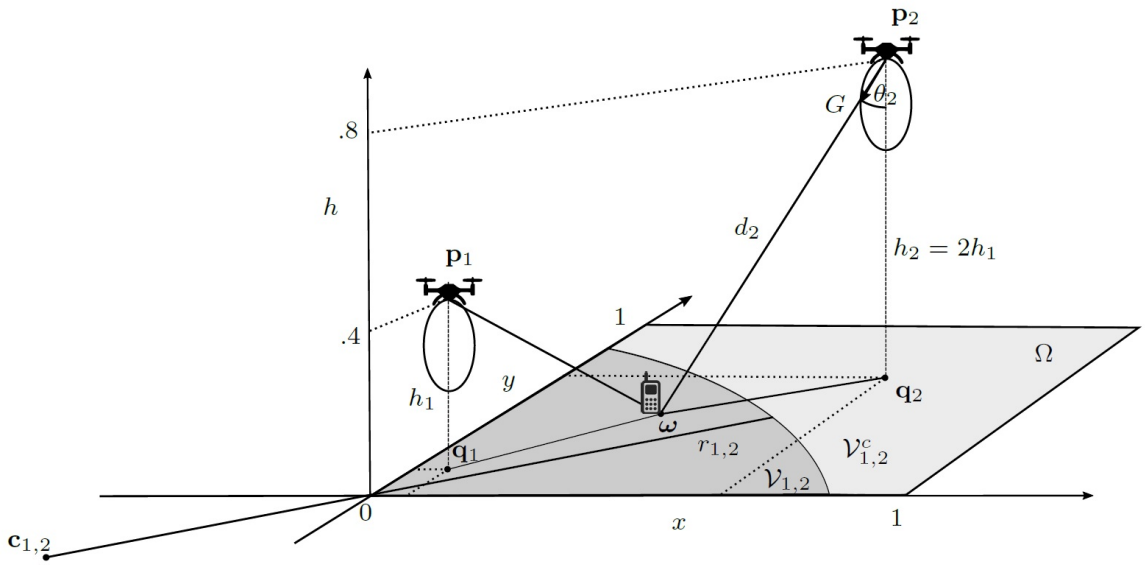


Figure 3.1: UAV deployment with directed antenna beam and associated GT cells for $\boldsymbol{\alpha} = 2$ and $N = 2$ for a uniform GT distribution.

with expectation

$$\mathbb{E}[P_{TX}] = \frac{(2^{\frac{R_b}{B}} - 1)N_0}{h_n G_{\mathrm{GT}} K d_0^{\boldsymbol{\alpha}}} \frac{d_n^{\boldsymbol{\alpha}+1}(\boldsymbol{\omega})}{\sqrt{2\pi}\sigma_{\psi_{dB}}} \int_{\mathbb{R}} \exp\left(-\frac{\psi_{dB}^2}{2\sigma_{\psi_{dB}}^2} + \ln(10)\frac{\psi_{dB}}{10}\right) d\psi_{dB} = \frac{\beta}{h_n} d_n^{2\gamma}(\boldsymbol{\omega}) \quad (3.6)$$

where the independent and fixed parameters are given by

$$\beta = (2^{\frac{R_b}{B}} - 1)BN_0 \exp\big(-\frac{\sigma_{\psi_{dB}}^2 (\ln 10)^2}{200}\big)(G_{\mathrm{GT}} K)^{-1} d_0^{-\boldsymbol{\alpha}} \quad \text{and} \quad \gamma = \frac{\boldsymbol{\alpha}+1}{2}. \quad (3.7)$$

Since the goal is to minimize the average transmission power (3.6) I define the nth *parameter distortion function* as

$$\mathcal{D}(\boldsymbol{\omega}, \mathbf{q}_n, a_n, b_n) = \beta \cdot \left(a_n \left\|\mathbf{q}_n - \boldsymbol{\omega}\right\|_2^2 + b_n\right)^{\gamma} \quad (3.8)$$

where $a_n = h_n^{-1/\gamma}$ and $b_n = h_n^{2-1/\gamma}$. As can be seen from (3.8), the distortion is a function of the parameter h_n in addition to the distance between the reproduction point $\mathbf{q}_n$ and the represented point $\boldsymbol{\omega}$. From a quantization point of view, one can start with the distortion function (3.8) without knowing the UAV power consumption formulas in this section. This

is what I will do in the next section. For simplicity, I will set from here on $\beta = 1$, since it will not affect the quantizer.

3.2 Optimizing Quantizers with parameterized distortion measures

The communication power cost (3.8) defines, with h_n and fixed $\gamma \geq 1$, a parameter-dependent distortion function for $\mathbf{q}_n$. For a given source sample GT density f in Ω and UAV deployment, the average power is the average distortion for given *quantization and parameter points* $(\mathbf{Q}, \mathbf{h})$ with quantization sets $\mathcal{R} = \{\mathcal{R}_n\}$, which is called the *average distortion* of the quantizer $(\mathbf{Q}, \mathbf{h}, \mathcal{R})$

$$\overline{\mathcal{D}}(\mathbf{Q}, \mathbf{h}, \mathcal{R}) = \sum_{n=1}^{N} \int_{\mathcal{R}_n} \mathcal{D}(\boldsymbol{\omega}, \mathbf{q}_n, h_n) f(\boldsymbol{\omega}) d\boldsymbol{\omega}. \tag{3.9}$$

The N quantization sets, which minimize the average distortion for given quantization and parameter points $(\mathbf{Q}, \mathbf{h})$, define a generalized Voronoi tessellation $\boldsymbol{V} = \{\boldsymbol{V}_n(\mathbf{Q}, \mathbf{h})\}$

$$\overline{\mathcal{D}}(\mathbf{Q}, \mathbf{h}, \boldsymbol{V}) := \int_{\Omega} \min_{n \in [N]} \{\mathcal{D}(\boldsymbol{\omega}, \mathbf{q}_n, h_n)\} f(\boldsymbol{\omega}) d\boldsymbol{\omega} = \sum_{n=1}^{N} \int_{\boldsymbol{V}_n(\mathbf{Q}, \mathbf{h})} \mathcal{D}(\boldsymbol{\omega}, \mathbf{q}_n, h_n) f(\boldsymbol{\omega}) d\boldsymbol{\omega}, \tag{3.10}$$

where the *generalized Voronoi regions* $\boldsymbol{V}_n(\mathbf{Q}, \mathbf{h})$ are defined as the set of sample points $\boldsymbol{\omega}$ with smallest distortion to the nth quantization point $\mathbf{q}_n$ with parameter h_n. Minimizing the *average distortion* $\overline{\mathcal{D}}(\mathbf{Q}, \mathbf{h}, \boldsymbol{V})$ over all parameter and quantization points can be seen as an *$N-$facility locational-parameter optimization problem* [18], [32], [33], [53]. By the definition of the Voronoi regions (3.10), this is equivalent to the minimum average distortion over all

$N-$level parameter quantizers

$$\overline{\mathcal{D}}(\mathbf{Q}^*, \mathbf{h}^*, \boldsymbol{V}^*) = \min_{(\mathbf{Q},\mathbf{h})\in\Omega^N\times\mathbb{R}_+^N} \overline{\mathcal{D}}(\mathbf{Q}, \mathbf{h}, \boldsymbol{V}) = \min_{(\mathbf{Q},\mathbf{h})\in\Omega^N\times\mathbb{R}_+^N} \min_{\mathcal{R}=\{\mathcal{R}_n\}\subset\Omega} \overline{\mathcal{D}}(\mathbf{Q}, \mathbf{h}, \mathcal{R}), \quad (3.11)$$

which I call the *$N-$level parameter optimized quantizer*. To find local extrema of (3.10) analytically, I will need that the objective function $\overline{\mathcal{D}}$ be continuously differentiable at any point in $\Omega^N \times \mathbb{R}_+^N$, i.e., the gradient should exist and be a continuous function. Such a property was shown to be true for piecewise continuous non-decreasing distortion functions in the Euclidean metric over Ω^N [6, Thm.2.2] and weighted Euclidean metric [32]. Then the necessary condition for a local extremum is the vanishing of the gradient at a critical point[1]. First, I will derive the generalized Voronoi regions for convex sets Ω in d dimensions for any parameters $h_n \in \mathbb{R}_+$ for the quantization points $\mathbf{q}_n$, which are special cases of *Möbius diagrams (tessellations)*, introduced in [116].

Lemma 5. *Let $\mathbf{Q} = \{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_N\} \subset \Omega^N \subset (\mathbb{R}^d)^N$ for $d \in \{1, 2\}$ be the quantization points and $\mathbf{h} = (h_1, \ldots, h_N) \in \mathbb{R}_+^N$ the associated parameters. Then the average distortion of $(\mathbf{Q}, \mathbf{h})$ over all samples in Ω distributed by f for some exponent $\gamma \geq 1$*

$$\overline{\mathcal{D}}\,(\mathbf{Q}, \mathbf{h}, \boldsymbol{V}) = \sum_{n=1}^{N} \int_{\boldsymbol{V}_n} \frac{(\|\mathbf{q}_n - \boldsymbol{\omega}\|^2 + h_n^2)^\gamma}{h_n} f(\boldsymbol{\omega}) d\boldsymbol{\omega} \quad (3.12)$$

has generalized Voronoi regions $\boldsymbol{V}_n = \boldsymbol{V}_n(\mathbf{Q}, \mathbf{h}) = \bigcap_{m\neq n} \boldsymbol{V}_{nm}$, where the dominance regions of quantization point n over m are given by

$$\boldsymbol{V}_{nm} = \begin{cases} \{\boldsymbol{\omega} \in \Omega \mid \|\mathbf{q}_n - \boldsymbol{\omega}\| \leq \|\mathbf{q}_m - \boldsymbol{\omega}\|\} & , h_m = h_n \\ \{\boldsymbol{\omega} \in \Omega \mid \|\boldsymbol{\omega} - \mathbf{c}_{nm}\| \leq r_{nm}\} & , h_n < h_m \\ \{\boldsymbol{\omega} \in \Omega \mid \|\boldsymbol{\omega} - \mathbf{c}_{nm}\| \geq r_{nm}\} & , h_n > h_m \end{cases} \quad (3.13)$$

[1]Note, if $\nabla\bar{P}$ is not continuous in $\mathcal{P}^N$ than any jump-point is a potential critical point and has to be checked individually.

where $\mathbf{c}_{nm}$ *and* r_{nm} *are given by*

$$\mathbf{c}_{nm} = \frac{\mathbf{q}_n - h_{nm}\mathbf{q}_m}{1 - h_{nm}} \quad \textit{and} \quad r_{nm} = \left(\frac{h_{nm}}{(1 - h_{nm})^2} \left\| \mathbf{q}_n - \mathbf{q}_m \right\|^2 + h_n^2 \frac{h_{nm}^{1-2\gamma} - 1}{1 - h_{nm}} \right)^{\frac{1}{2}}. \tag{3.14}$$

Here, I introduced the parameter ratio of the n*th and* m*th quantization point as* $h_{nm} = (h_n/h_m)^{\frac{1}{\gamma}}$.

Remark. It is also possible that two quantization points are equal, but have different parameters. If the parameter ratio is very small or very large, one quantization point can become redundant, i.e., if its optimal quantization set is empty. In fact, if one optimizes over all quantizer points, such a case will be excluded, which I show for one-dimension in Lemma 7.

Proof. The minimization of the distortion functions over Ω defines an assignment rule for a generalized Voronoi diagram $\boldsymbol{V}(\mathbf{Q}, \mathbf{h}) = \{\boldsymbol{V}_1, \boldsymbol{V}_2, \ldots, \boldsymbol{V}_N\}$ where

$$\boldsymbol{V}_n = \boldsymbol{V}_n(\mathbf{Q}, \mathbf{h}) := \left\{ \boldsymbol{\omega} \in \Omega \;\middle|\; a_n \left\| \mathbf{q}_n - \boldsymbol{\omega} \right\|^2 + b_n \leq a_m \left\| \mathbf{q}_m - \boldsymbol{\omega} \right\|^2 + b_m, m \neq n \right\} \tag{3.15}$$

is the nth generalized Voronoi region, see for example [33, Cha.3]. Here I denoted the weights by the positive numbers

$$a_n = h_n^{-\frac{1}{\gamma}}, \quad b_n = h_n^{2-\frac{1}{\gamma}} \tag{3.16}$$

which define a *Möbius diagram* [116], [117]. The bisectors of Möbius diagrams are circles or lines in $\mathbb{R}^2$ as I will show below. The nth Voronoi region is defined by $N-1$ inequalities, which can be written as the intersection of the $N-1$ *dominance regions* of $\mathbf{q}_n$ over $\mathbf{q}_m$, given by

$$\boldsymbol{V}_{nm} = \left\{ \boldsymbol{\omega} \in \Omega \;\middle|\; a_n \left\| \mathbf{q}_n - \boldsymbol{\omega} \right\|^2 + b_n \leq a_m \left\| \mathbf{q}_m - \boldsymbol{\omega} \right\|^2 + b_m \right\}. \tag{3.17}$$

If $h_n = h_m$ then $a_n = a_m$ and $b_n = b_m$, such that $\boldsymbol{V}_{nm} = \mathcal{H}(\mathbf{q}_n, \mathbf{q}_m)$, the left half-space between $\mathbf{q}_n$ and $\mathbf{q}_m$. For $a_n > a_m$ we can rewrite the inequality as

$$\|\boldsymbol{\omega}\|^2 - 2\langle \mathbf{c}_{nm}, \boldsymbol{\omega}\rangle + \frac{a_n^2\|\mathbf{q}_n\|^2 + a_m^2\|\mathbf{q}_m\|^2 - a_n a_m(\|\mathbf{q}_n\|^2 + \|\mathbf{q}_m\|^2)}{(a_n - a_m)^2} + \frac{b_n - b_m}{a_n - a_m} \leq 0$$

where the center point is given by

$$\mathbf{c}_{nm} = \frac{a_n\mathbf{q}_n - a_m\mathbf{q}_m}{a_n - a_m} = a_n\frac{\mathbf{q}_n - h_{nm}\mathbf{q}_m}{a_n - a_m} = \frac{\mathbf{q}_n - h_{nm}\mathbf{q}_m}{1 - h_{nm}} \tag{3.18}$$

where I introduced the *parameter ratio* of the nth and mth quantization point as

$$h_{nm} := a_m/a_n = (h_n/h_m)^{\frac{1}{\gamma}} > 0. \tag{3.19}$$

If $0 < a_n - a_m$, which is equivalent to $h_n < h_m$, then this defines a ball (disc) and for $h_n > h_m$ its complement. Hence we get

$$\boldsymbol{V}_{nm} = \begin{cases} \{\boldsymbol{\omega} \in \Omega \mid \|\boldsymbol{\omega} - \mathbf{c}_{nm}\| \leq r_{nm}\}, & h_n < h_m \\ \{\boldsymbol{\omega} \in \Omega \mid \|\boldsymbol{\omega} - \mathbf{q}_n\| \leq \|\boldsymbol{\omega} - \mathbf{q}_m\|\}, & h_n = h_m \\ \{\boldsymbol{\omega} \in \Omega \mid \|\boldsymbol{\omega} - \mathbf{c}_{nm}\| \geq r_{nm}\}, & h_n > h_m \end{cases} \tag{3.20}$$

where the radius square is given by

$$r_{nm}^2 = a_n a_m \frac{\|\mathbf{q}_n - \mathbf{q}_m\|^2}{(a_n - a_m)^2} + \frac{b_m - b_n}{a_n - a_m} = \frac{a_n}{a_m}\frac{\|\mathbf{q}_n - \mathbf{q}_m\|^2}{(1 - \frac{a_n}{a_m})^2} + \frac{b_m - b_n}{a_n - a_m}. \tag{3.21}$$

The second summand can be written as

$$\frac{b_m - b_n}{a_n - a_m} = \frac{h_m^{2-\frac{1}{\gamma}} - h_n^{2-\frac{1}{\gamma}}}{h_n^{-\frac{1}{\gamma}} - h_m^{-\frac{1}{\gamma}}} = \frac{h_n^2\left((h_n/h_m)^{\frac{1}{\gamma}-2} - 1\right)}{1 - (h_n/h_m)^{\frac{1}{\gamma}}} = h_n^2\frac{h_{nm}^{-\boldsymbol{\alpha}} - 1}{1 - h_{nm}}. \tag{3.22}$$

For any $\gamma \geq 1$, we have $h_{nm} = (h_n/h_m)^{1/\gamma} < 1$ if $h_n < h_m$ and $h_{nm} \geq 1$ else. In both cases (3.22) is positive, which implies a radius $r_{nm} > 0$ whenever $\mathbf{q}_n \neq \mathbf{q}_m$. Inserting (3.22) in (3.21) yields the result. □

Example 3.1. *We plotted in Fig. 3.1, for $N = 2$ and $\Omega = [0,1]^2$, the GT regions for a uniform distribution with UAVs placed on*

$$\mathbf{q}_1 = (0.1, 0.2), h_1 = 0.5, \quad \textit{and} \quad \mathbf{q}_2 = (0.6, 0.6), h_2 = 1. \tag{3.23}$$

If the second UAV reaches an altitude of $h_2 \geq 2.3$, its Voronoi region $\boldsymbol{V}_2 = \boldsymbol{V}_{2,1}$ will be empty and hence become "inactive".

3.2.1 Local optimality conditions

To find the optimal $N-$level parameter quantizer (3.10), we have to minimize the average distortion (3.9) over all possible quantization-parameter points, i.e., we have to solve a non-convex $N-$*facility locational-parameter optimization problem*,

$$\overline{\mathcal{D}}(\mathbf{Q}^*, \mathbf{h}^*, \boldsymbol{V}^*) = \min_{\mathbf{Q}\in\Omega^N, \mathbf{h}\in\mathbb{R}_+^N} \sum_{n=1}^{N} \int_{\boldsymbol{V}_n(\mathbf{Q},\mathbf{h})} h_n^{-1}(\|\mathbf{q}_n - \boldsymbol{\omega}\|^2 + h_n^2)^{\gamma} f(\boldsymbol{\omega}) d\boldsymbol{\omega} \tag{3.24}$$

where $\boldsymbol{V}_n(\mathbf{Q}, \mathbf{h})$ are the Mbius regions given in (3.13) for each fixed $(\mathbf{Q}, \mathbf{h})$. A point $(\mathbf{Q}^*, \mathbf{h}^*)$ with Mbius diagram $\boldsymbol{V}^* = \boldsymbol{V}(\mathbf{Q}^*, \mathbf{h}^*) = \{\boldsymbol{V}_1^*, \ldots, \boldsymbol{V}_N^*\}$ is a critical point of (3.24) if all partial derivatives of $\overline{\mathcal{D}}$ are vanishing, i.e., if for each $n \in [N]$ it holds

$$0 = \int_{\boldsymbol{V}_n^*} (\mathbf{q}_n^* - \boldsymbol{\omega})(\|\mathbf{q}_n^* - \boldsymbol{\omega}\|^2 + h_n^{*2})^{\gamma-1} f(\boldsymbol{\omega}) d\boldsymbol{\omega} \tag{3.25}$$

$$0 = \int_{\boldsymbol{V}_n^*} (\|\mathbf{q}_n^* - \boldsymbol{\omega}\|^2 + h_n^{*2})^{\gamma-1} \cdot (\|\mathbf{q}_n^* - \boldsymbol{\omega}\|^2 - (2\gamma - 1)h_n^{*2}) f(\boldsymbol{\omega}) d\boldsymbol{\omega}. \tag{3.26}$$

For $N = 1$ the integral regions will not depend on $\mathbf{Q}$ or $\mathbf{h}$ and since the integral kernel is continuously differentiable, the partial derivatives will only apply to the integral kernel.

Similar to [6], for $N > 1$, the conservation-of-mass law, can be used to show that the derivatives of the integral domains will cancel each other out.

Remark. The shape of the regions depend on the parameters, which if different for each quantization point (heterogeneous), generate spherical and not polyhedral regions. We will show later, that homogeneous parameter selection with polyhedral regions will be the optimal regions for $d = 1$.

3.2.2 The optimal N-level parameter quantizer in one-dimension for uniform density

In this section, I discuss the parameter optimized quantizer for a one-dimensional convex source $\Omega \subset \mathbb{R}$, i.e., for an interval $\Omega = [s, t]$ given by some real numbers $s < t$. Under such circumstances, the quantization points are degenerated to scalars, i.e., $\mathbf{q}_n = x_n \in [s, t], \forall n \in [N]$. If we shift the interval Ω by an arbitrary $a \in \mathbb{R}$, then the average distortion, i.e., the objective function, will not change if we shift all quantization points by the same number a. Hence, if we set $a = -s$, we can shift any quantizer for $[s, t]$ to $[0, A]$ where $A = t - s$ without loss of generality. Let us assume a uniform distribution on Ω, i.e. $f(\boldsymbol{\omega}) = 1/A$. To derive the unique $N-$level parameter optimized quantizer for any N, I will first investigate the case $N = 1$.

Lemma 6. *Let $A > 0$ and $\gamma \geq 1$. The unique $1-$level parameter optimized quantizer (x^*, h^*) with distortion function* (3.8) *is given for a uniform source density in* $[0, A]$ *by*

$$x^* = \frac{A}{2}, h^* = \frac{A}{2} g(\gamma) \quad \textit{and the minimum average distortion} \quad \overline{\mathcal{D}}(x^*, h^*) = \left(\frac{A}{2}\right)^{2\gamma-1} g(\gamma)$$

where $g(\gamma) = \arg\min_{u>0} F(u, \gamma) < 1/\sqrt{2\gamma - 1}$ is the unique minimizer of

$$\Psi(u, \gamma) = \int_0^1 \psi(\boldsymbol{\omega}, u, \gamma) d\boldsymbol{\omega} \quad \textit{with} \quad \psi(\boldsymbol{\omega}, u, \gamma) = \frac{(\boldsymbol{\omega}^2 + u^2)^\gamma}{u} \tag{3.27}$$

which is for fixed γ a continuous and convex function over $\mathbb{R}_+$. For $\gamma \in \{1,2,3\}$ the minimizer can be derived in closed form as

$$g(1) = \sqrt{1/3}, \quad g(2) = \sqrt{(\sqrt{32/5}-1)/9}, \quad g(3) = \sqrt{\Big((32/7)^{1/3}-1\Big)/5}. \tag{3.28}$$

Proof. See Appendix A.2. □

Remark. The convexity of $\Psi(\cdot,\gamma)$ can be also shown by using extensions of the Hermite-Hadamard inequality [118], which allows to show convexity over any interval. Let us note here that for any fixed parameter $h_n > 0$, the average distortion $\overline{\mathcal{D}}(x_n^* \pm \epsilon, h_n)$ is strictly monotone increasing in $\epsilon > 0$. Hence, x_n^* is the unique minimizer for any $h_n > 0$. We will use this decoupling property repeatedly in the proofs [55].

To derive the main result, we need some general properties of the optimal regions.

Lemma 7. *Let $\Omega = [0, A]$ for some $A > 0$. The $N-$level parameter optimized quantizer $(\mathbf{Q}^*, \mathbf{h}^*) \in \Omega^N \times \mathbb{R}_+^N$ for a uniform source density in Ω has optimal quantization regions $\boldsymbol{V}_n(\mathbf{Q}^*, \mathbf{h}^*) = [b_{n-1}^*, b_n^*]$ with $0 \leq b_{n-1}^* < b_n^* \leq A$ and optimal quantization points $x_n^* = (b_n^* + b_{n-1}^*)/2$ for $n \in [N]$, i.e., each region is a closed interval with positive measure and centroidal quantization points.*

Proof. See Appendix A.3. □

Remark. Hence, for an $N-$level parameter optimized quantizer, all quantization points are used, which is intuitively, since each additional quantization point should reduce the distortion of the quantizer by partitioning the source in non-zero regions.

Theorem 3.1. *Let $N \in \mathbb{N}$, $\Omega = [0, A]$ for some $A > 0$, and $\gamma \geq 1$. The* unique $N-$level parameter optimized quantizer $(\mathbf{Q}^*, \mathbf{h}^*, \mathcal{R}^*)$ *is the uniform scalar quantizer with identical parameter values, given for $n \in [N]$ by*

$$\mathbf{q}_n^* = x_n^* = \frac{A}{2N}(2n-1), \quad h^* = h_n^* = \frac{A}{2N} g(\gamma), \quad \mathcal{R}_n^* = \left[\frac{A}{N}(n-1), \frac{A}{N} n\right] \tag{3.29}$$

with minimum average distortion

$$\overline{\mathcal{D}}(\mathbf{Q}^*, \mathbf{h}^*, \mathcal{R}^*) = \left(\frac{A}{2N}\right)^{2\gamma-1} \int_0^1 \frac{\left(\boldsymbol{\omega}^2 + g^2(\gamma)\right)^\gamma}{g(\gamma)} d\boldsymbol{\omega}. \tag{3.30}$$

For $\gamma \in \{1, 2, 3\}$, the closed form $g(\gamma)$ is provided in (3.28).

Proof. See Appendix A.4. □

Example 3.2. *We plot the optimal heights and optimal average distortion for a uniform GT density in $[0, 1]$ over various $\boldsymbol{\alpha}$ and $N = 2$ in Fig. 3.2. Note that the factor $A/2N = 1/4$ will play a crucial role for the height and distortion scaling. Moreover, the distortion decreases exponentially in $\boldsymbol{\alpha}$ if $A/2N < 1$.*

Let us set $\beta = 1 = A$. Then, the optimal UAV deployment is pictured in Fig. 3.3 for $N = 2$ and $N = 4$. The maximum elevation angle θ_{max} is hereby constant for each UAV and does not change if the number of UAVs, N, increases. Moreover, it is also independent of A and β, since with (3.29) *we have $\mu_n^* = x_n^* - x_{n-1}^* = A/N$ and*

$$\cos(\theta_{max}) = \cos(\theta_n) = \frac{h^*}{\mu_n^*/2} = \frac{2N}{A}\frac{A}{2N} g(1) = \frac{1}{\sqrt{3}}. \tag{3.31}$$

3.3 Llyod-like Algorithms and Simulation Results

In this section, I introduce two Lloyd-like algorithms, Lloyd-A and Lloyd-B, to optimize the quantizer for two-dimensional scenarios. The proposed algorithms iterate between two steps:

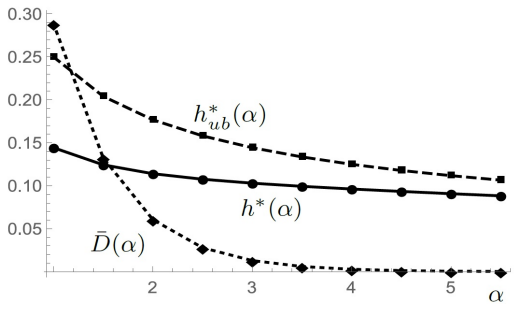


Figure 3.2: Optimal height (solid) with bound (dashed) and average distortion (dotted) for $N = 2, A = 1$ and uniform GT density.

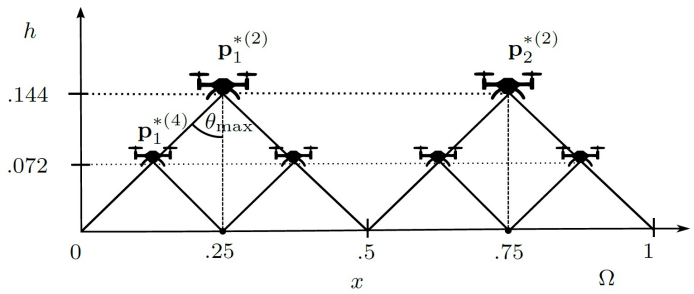
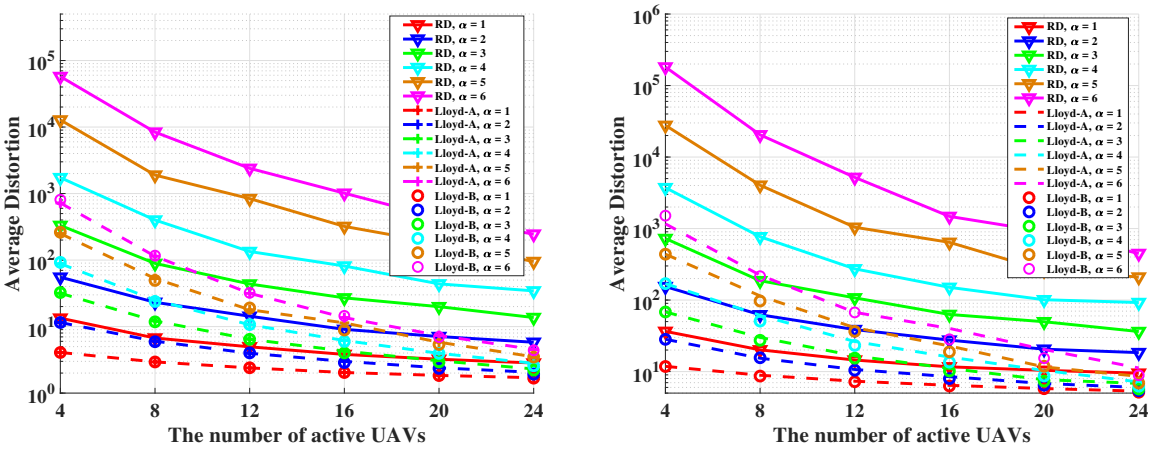


Figure 3.3: Optimal UAV deployment in one dimension for $A = 1, \boldsymbol{\alpha} = 1$ and $N = 2, 4$ over a uniform GT density by (3.29).

(1) The reproduction points are optimized through gradient descent while the partitioning is fixed; (ii) The partitioning is optimized while the reproduction points are fixed. In Lloyd-A, all UAVs (or reproduction points) share the common flight height while Lloyd-B allows UAVs at different flight heights.

In what follows, I provide the simulation results over the two-dimensional target region $\Omega = [0, 10]^2$ with uniform and non-uniform density functions. The non-uniform density function is a Gaussian mixture of the form $\sum_{k=1}^{3} \frac{A_k}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{\|\boldsymbol{\omega}-\mathbf{c}_k\|^2}{2\sigma_k}\right)$, where the weights, A_k, $k = 1, 2, 3$ are 0.5, 0.25, 0.25, the means, $\mathbf{c}_k$, are $(3, 3)$, $(6, 7)$, $(7.5, 2.5)$, the standard deviations, σ_k, are 1.5, 1, and 2, respectively.



(a) (b)

Figure 3.4: The performance comparison of Lloyd-A, Lloyd-B and Random Deployment (RD). (a) Uniform density. (b) Non-uniform density.

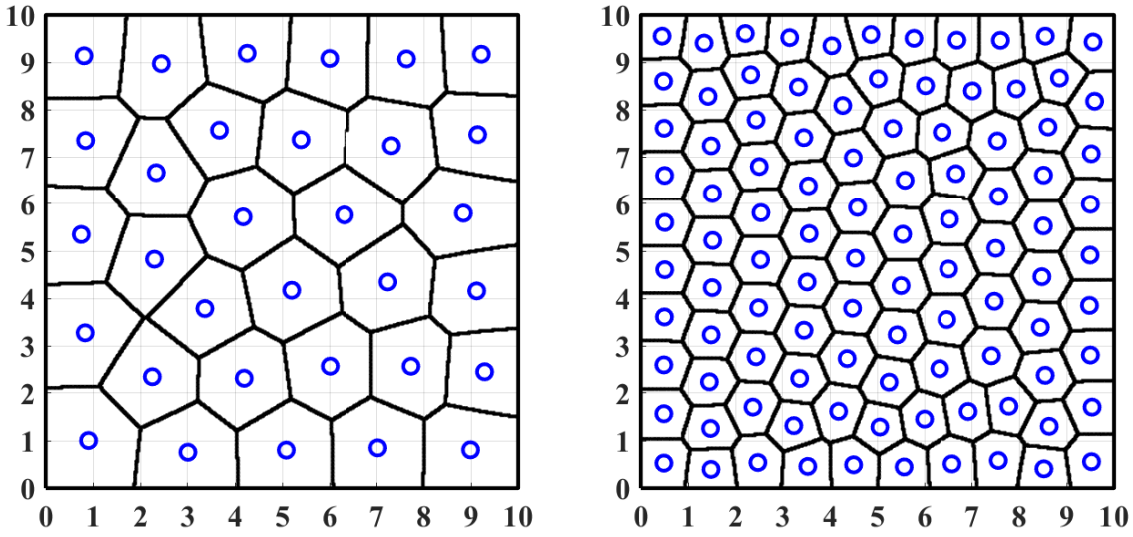
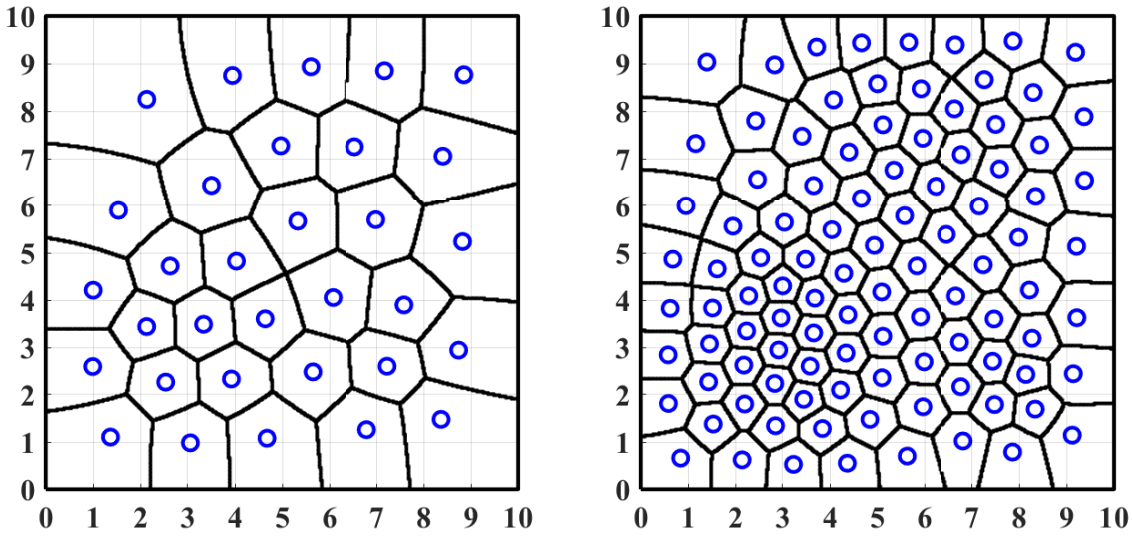


Figure 3.5: The UAV projections on the ground with generalized Voronoi Diagrams where $\boldsymbol{\alpha} = 2$ and the source distribution is uniform. (a) 32 UAVs. (b) 100 UAVs.

To evaluate the performance of the proposed algorithms, I compare them with the average distortion of 100 random deployments (RDs). Figs. 3.4a and 3.4b, show that the proposed algorithms outperform the random deployment on both uniform and non-uniform distributed target regions. From Fig. 3.4a, one can also find that the distortion achieved by Lloyd-A and Lloyd-B are very close, indicating that the optimality of the common height, as proved for the one-dimensional case in Section 3.2, might be extended to the two-dimensional case when the density function is uniform. However, one can find a non-negligible gap between Lloyd-A and Lloyd-B in Fig. 3.4b where the density function is non-uniform. For instance, given 16 UAVs and the path-loss exponent $\boldsymbol{\alpha} = 6$, Lloyd-A's distortion is 40.17 while Lloyd-B obtains a smaller distortion, 28.25, by placing UAVs at different heights. Figs. 3.5a and 3.5b illustrate the UAV ground projections and their partitions on a uniform distributed square region. As the number of UAVs increases, the UAV partitions approximate hexagons which implies that the optimality of congruent partition (Theorem 3.1) might be extended to uniformly distributed users for two-dimensional sources. 3.6a and 3.6b show that congruent partition is no longer a necessary condition for the optimal quantizer when distribution is non-uniform.



(a) (b)

Figure 3.6: The UAV projections on the ground with generalized Voronoi Diagrams where $\boldsymbol{\alpha} = 2$ and the source distribution is non-uniform. (a) 32 UAVs. (b) 100 UAVs.

Chapter 4

Energy Efficiency in Two-Tiered Wireless Sensor Networks

Multi-tier networks have many applications in different fields. In this Chapter, I define a novel two-tier quantizer that can be applied to different node deployment problems including the energy conservation in two-tier wireless sensor networks (WSNs) consisting of N access points (APs) and M fusion centers (FCs). We aim at finding an optimal deployment of APs and FCs to minimize the average weighted total, or Lagrangian, of sensor and AP powers. For one fusion center, $M = 1$, I show that the optimal deployment of APs is simply a linear transformation of the optimal N-level quantizer for density f, and the sole FC should be located at the geometric centroid of the sensing field. I also provide the exact expression of the AP-Sensor power function and prove its convexity. For more than one fusion center, $M > 1$, I provide a necessary condition for the optimal deployment. Furthermore, to numerically optimize the AP and FC deployment, I propose three Lloyd-like algorithms and analyze their convergence.

4.1 System Model and Problem Formulation

We consider a two-tier WSN consisting of three kinds of nodes: sensors, APs, and FCs. This is a canonical application of the two-tier quantizer. A similar network architecture has been studied in [68] where the authors ignore the energy consumption of the sensor nodes. We model the energy consumption by radio communication in two-tier WSNs.

Let Ω be a convex polygon in $\mathbb{R}^2$ including its interior. Given the target area Ω, N APs and M FCs are deployed to collect data. Without loss of generality, we assume that $N > M$. $\mathcal{I}_{\mathcal{A}} = \{1, \ldots, N\}$, and $\mathcal{I}_{\mathcal{B}} = \{1, \ldots, M\}$ denote the sets of AP indices and FC indices, respectively. AP deployment and FC deployment are, respectively, defined by $\boldsymbol{P} = (\boldsymbol{p}_1, \cdots, \boldsymbol{p}_N)$ and $\boldsymbol{Q} = (\boldsymbol{q}_1, \cdots, \boldsymbol{q}_M)$, where $\boldsymbol{p}_n \in \mathbb{R}^2$ is the location of AP n and $\boldsymbol{q}_m \in \mathbb{R}^2$ is the location of FC m. An AP partition $\mathcal{R}^{\mathcal{A}} = \{\mathcal{R}_n^{\mathcal{A}}\}_{n \in \mathcal{I}_{\mathcal{A}}}$ is a collection of disjoint subsets of $\mathbb{R}^2$ whose union is Ω. Let $\mathbf{T} : \mathcal{I}_{\mathcal{A}} \to \mathcal{I}_{\mathcal{B}}$ be an index map for which $\mathbf{T}(n) = m$ if and only if AP n is connected to FC m. A continuous and differentiable function $f(\cdot) : \Omega^2 \to \mathbb{R}_+$ is used to denote the density of the data rate from the densely distributed sensors [68]. Then, $\int_{\mathcal{R}} f(\boldsymbol{\omega}) d\boldsymbol{\omega}$ is the total amount of data generated from the sensors in region $\mathcal{R}$ in one time unit.

Usually, FCs have access to reliable energy sources and their energy consumption is not the main concern in this paper. Therefore, we focus on the energy consumption of sensors and APs. In fact, the energy consumed by sensors and APs comes from three parts: (i) Sensors transmit bit streams to APs; (ii) APs transmit bit streams to FCs; (iii) APs receive bit streams from sensors. The transmitting powers (Watt) of nodes, e.g., sensors and APs, mainly depend on two factors: (i) the *instant-transmission-power* (Joules/second); and (ii) the *channel-busy-ratio*: the percentage of time that the transmitter forwards data. The *average-transmitting-power* of AP n is defined as $\mathcal{P}_{t_n}^{\mathcal{A}} = E_{t_n}^{\mathcal{A}} \Gamma_n^{\mathcal{A}}, n \in \mathcal{I}_{\mathcal{A}}$, where $E_{t_n}^{\mathcal{A}}$ is the *instant-transmission-power* of AP n and $\Gamma_n^{\mathcal{A}}$ is the *channel-busy-ratio* for the channel from AP n to the corresponding FC. In order to achieve the required SNR thresholds at the receivers, the *instant-transmission-power* $E_{t_n}^{\mathcal{A}}$ should be set to a value that is determined by the distance, the antenna gain, and the SNR threshold [119]. More realistically, the

transmission power is proportional to distance squared or another power of distance in the presence of obstacles [66]. Taking path-loss into consideration, it is reasonable to model the *instant-transmission-power* from AP n to FC $\mathbf{T}(n)$ by $E^{\mathcal{A}}_{t_n} = \eta^{\mathcal{A}}_{n,\mathbf{T}(n)} \|\boldsymbol{p}_n - \boldsymbol{q}_{\mathbf{T}(n)}\|^{\alpha}$, where $\|\cdot\|$ denotes the Euclidean distance, $\eta^{\mathcal{A}}_{n,\mathbf{T}(n)}$ is a constant determined by the antenna gain of AP n and the SNR threshold of FC $\mathbf{T}(n)$, and $\alpha \in [2,4]$ is the path-loss parameter. We consider an environment without obstacles, i.e., $\alpha = 2$. Let $\xi^{\mathcal{A}}_{n\mathbf{T}(n)}$ be the AP n's *instant-transmitter-data-rate* which is determined by the SNR at the corresponding FC $\mathbf{T}(n)$. In this chapter, we focus on homogeneous sensors, APs, and FCs. Hence, we consider identical sensor antenna gains, identical AP antenna gains, identical AP SNR thresholds, and identical FC SNR thresholds. Without loss of generality, we set $\eta^{\mathcal{A}}_{n,\mathbf{T}(n)} = \eta^{\mathcal{A}}$ and $\xi^{\mathcal{A}}_{n\mathbf{T}(n)} = \xi^{\mathcal{A}}$. In such a homogeneous scenario, the total amount of data is proportional to the number of sensors, and $f(\cdot)$ is just a scaled sensor density function. Note that AP n collects data from the sensors in $\mathcal{R}^{\mathcal{A}}_n$, indicating that the *average-receiver-data-rate* - the amount of data received from sensors in the unit time - is $\int_{\mathcal{R}^{\mathcal{A}}_n} f(\boldsymbol{\omega}) d\boldsymbol{\omega}$. It is reasonable to assume that the AP transmitters have idle time and forward data only when the collected sensing data comes into the buffer. Hence, the *channel-busy-ratio* is proportional to the *average-receiver-data-rate*, and can be written as $\Gamma^{\mathcal{A}}_n = \frac{\int_{\mathcal{R}^{\mathcal{A}}_n} f(\boldsymbol{\omega}) d\boldsymbol{\omega} \cdot T/\xi^{\mathcal{A}}}{T} = \frac{\int_{\mathcal{R}^{\mathcal{A}}_n} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}{\xi^{\mathcal{A}}}$, where $\xi^{\mathcal{A}}$ is AP n's *instant-transmitter-data-rate*. Therefore, the *average-transmitter-power* of AP n can be rewritten as $\mathcal{P}^{\mathcal{A}}_{t_n} = E^{\mathcal{A}}_{t_n} \Gamma^{\mathcal{A}}_n = \frac{\eta^{\mathcal{A}}}{\xi^{\mathcal{A}}} \|\boldsymbol{p}_n - \boldsymbol{q}_{\mathbf{T}(n)}\|^2 \int_{\mathcal{R}^{\mathcal{A}}_n} f(\boldsymbol{\omega}) d\boldsymbol{\omega}$.

The power consumption at AP receivers is a constant and does not affect energy optimization [54]. Without loss of generality, we drop it from the total energy consumption. Therefore, to consider the total energy consumption, we use the sum of the APs' *average-transmitter-powers*, $P^{\mathcal{A}}_{t_n}$, which is calculated as

$$\overline{\mathcal{P}}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) = \sum_{n=1}^{N} \mathcal{P}^{\mathcal{A}}_{t_n} = \sum_{n=1}^{N} \int_{\mathcal{R}^{\mathcal{A}}_n} \frac{\eta^{\mathcal{A}}}{\xi^{\mathcal{A}}} \|\boldsymbol{p}_n - \boldsymbol{q}_{\mathbf{T}(n)}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega}. \tag{4.1}$$

Next, we discuss sensors' total power consumption. Similar to the power consumption at AP receivers, the power consumption at sensor receivers is also a constant and thus dropped from the objective function. In what follows, we focus on power consumption at sensor transmitters. Since f should be approximately uniform on an infinitesimal region $[\boldsymbol{\omega}, \boldsymbol{\omega}+d\boldsymbol{\omega}]$, the total amount of data generated from $[\boldsymbol{\omega}, \boldsymbol{\omega}+d\boldsymbol{\omega}]$ in one time unit is $f(\boldsymbol{\omega})d\boldsymbol{\omega}$. As a result, the sum of *channel-busy-ratios* of the sensors in $[\boldsymbol{\omega}, \boldsymbol{\omega}+d\boldsymbol{\omega}]$ is $\Gamma^{\mathcal{S}} = \frac{f(\boldsymbol{\omega})d\boldsymbol{\omega}\cdot T/\xi^{\mathcal{S}}}{T} = \frac{f(\boldsymbol{\omega})d\boldsymbol{\omega}}{\xi^{\mathcal{S}}}$, where $\xi^{\mathcal{S}}$ is sensors' *instant-transmitter-data-rate*. In addition, sensors within $[\boldsymbol{\omega}, \boldsymbol{\omega}+d\boldsymbol{\omega}]$ are supposed to have approximately the same distance to the corresponding FC q, and then have the same *instant-transmission-power* $E_t^{\mathcal{S}} = \eta^{\mathcal{S}}\|\boldsymbol{\omega}-q\|^2$, where $\eta^{\mathcal{S}}$ is a constant determined by the antenna gain of sensors and the SNR threshold of the corresponding AP. Hence, the sum of the sensors' *average-transmitter-powers* within $[\boldsymbol{\omega}, \boldsymbol{\omega}+d\boldsymbol{\omega}]$ is $\frac{\eta^{\mathcal{S}}}{\xi^{\mathcal{S}}}\|q-\boldsymbol{\omega}\|^2 f(\boldsymbol{\omega})d\boldsymbol{\omega}$. Note that sensors in $\mathcal{R}^{\mathcal{A}}$ transfer data to AP n. Therefore, the sum *average-transmitter-powers* of the sensors in the whole region Ω is calculated as

$$\overline{\mathcal{P}}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}}) = \sum_{n=1}^{N} \int_{\mathcal{R}_n^{\mathcal{A}}} \frac{\eta^{\mathcal{S}}}{\xi^{\mathcal{S}}} \|\boldsymbol{p}_n - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega}. \tag{4.2}$$

We define the scaled AP power and the scaled sensor power, respectively, as

$$\mathcal{P}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) \triangleq \frac{\xi^{\mathcal{A}}}{\eta^{\mathcal{A}}} \overline{\mathcal{P}}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) = \sum_{n=1}^{N} \int_{\mathcal{R}_n^{\mathcal{A}}} \|\boldsymbol{p}_n - \boldsymbol{q}_{\mathbf{T}(n)}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega}, \tag{4.3}$$

$$\mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}}) \triangleq \frac{\xi^{\mathcal{S}}}{\eta^{\mathcal{S}}} \cdot \overline{\mathcal{P}}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}}) = \sum_{n=1}^{N} \int_{\mathcal{R}_n^{\mathcal{A}}} \|\boldsymbol{p}_n - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega}. \tag{4.4}$$

In what follows, we focus on the scaled sensor power and the scaled AP power which are also, respectively, referred to as Sensor-power and AP-power for simplicity. On one hand, Sensors' densely deployment prevent the network from breaking by few sensor failures. On the other hand, the limited number of APs makes the recharging possible for APs. Under such

scenario, saving the total energy consumption becomes an important requirement. Therefore, one objective could be minimizing the scaled total AP power consumption in (4.3) given a constraint on the total sensor power (4.4) or vice versa. Like rate-distortion function $R(D)$ and the distortion-rate function $D(R)$, we define the AP-sensor power function and Sensor-AP power function, respectively, as

$$A(S) \triangleq \inf_{(\boldsymbol{P},\boldsymbol{Q},\mathcal{R}^{\mathcal{A}},\mathbf{T}):\mathcal{P}^{\mathcal{S}}(\boldsymbol{P},\mathcal{R}^{\mathcal{A}})\leq S} \mathcal{P}^{\mathcal{A}}(\boldsymbol{P},\boldsymbol{Q},\mathcal{R}^{\mathcal{A}},\mathbf{T}) \tag{4.5}$$

$$S(A) \triangleq \inf_{(\boldsymbol{P},\boldsymbol{Q},\mathcal{R}^{\mathcal{A}},\mathbf{T}):\mathcal{P}^{\mathcal{A}}(\boldsymbol{P},\boldsymbol{Q},\mathcal{R}^{\mathcal{A}},\mathbf{T})\leq A} \mathcal{P}^{\mathcal{S}}(\boldsymbol{P},\mathcal{R}^{\mathcal{A}}). \tag{4.6}$$

We can then define the Lagrangian objective function (two-tier distortion) to be minimized as

$$\begin{aligned}&\mathcal{D}(\boldsymbol{P},\boldsymbol{Q},\mathcal{R}^{\mathcal{A}},\mathbf{T}) = \mathcal{P}^{\mathcal{S}}(\boldsymbol{P},\mathcal{R}^{\mathcal{A}}) + \beta\mathcal{P}^{\mathcal{A}}(\boldsymbol{P},\boldsymbol{Q},\mathcal{R}^{\mathcal{A}},\mathbf{T})\\ =&\sum_{n=1}^{N}\int_{\mathcal{R}_n^{\mathcal{A}}} \left[\|\boldsymbol{p}_n-\boldsymbol{\omega}\|^2 + \beta\|\boldsymbol{p}_n-\boldsymbol{q}_{\mathbf{T}(n)}\|^2\right] f(\boldsymbol{\omega})d\boldsymbol{\omega}.\end{aligned} \tag{4.7}$$

Our main goal is to minimize the two-tier distortion over the following four variables: (i) AP deployment $\boldsymbol{P}$; (ii) FC deployment $\boldsymbol{Q}$; cell partition $\mathcal{R}^{\mathcal{A}}$; (iv) index map T. There are many applications that result in a two-tier quantization set-up and as long as the corresponding distortion is a function of the distance, the results will hold for those applications as well. Another interpretation is to consider (4.1) as the sensor energy consumption and (4.2) as the sensing quality measured by the mean square error (MSE) in a one-tier WSN scenario in which dots in Fig. 1.1b are sensors, stars are APs and the density function $f(\cdot)$ is replaced by the event probability density function. In other words, the proposed optimization problem can also provide the trade-off between energy efficiency and sensing quality in one-tier WSNs.

4.2 The Best Possible Distortion for the Two-tier WSNs with One FC

As discussed earlier, the node deployment problem in two-tier WSNs can be interpreted as a two-tier quantizer design problem whose reproduction points are APs and FCs. Similarly, if one only considers the energy consumption of sensors, the corresponding optimization problem becomes a "regular" (or one-tier) quantization problem with distortion

$$D_r(X, \mathbf{R}) = \sum_{n=1}^{N} \int_{R_n} \|x_n - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega}. \tag{4.8}$$

Let $(X^*, \mathbf{R}^*) = \arg\min_{(X,\mathbf{R})} D_r(X, \mathbf{R})$ be the optimal one-tier quantizer. In some cases [37], [68]–[70], only one FC or fusion center is deployed to collect data from the entire WSNs. In the case of one FC and multiple APs, the following proposition holds.

Proposition 2. *For a two-tier WSN with one FC, We have the following:*

(i) The optimal FC location is the centroid of the target region, i.e., $\boldsymbol{q}^* = \frac{\int_\Omega \boldsymbol{\omega} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}{\int_\Omega f(\boldsymbol{\omega}) d\boldsymbol{\omega}}$.

(ii) The optimal AP locations of the two-tier WSNs are linear transformations of the optimal reproduction points $X^* = (x_1^*, \ldots, x_N^*)$*, i.e.,* $\boldsymbol{p}_n^* = \frac{x_n^* + \beta \boldsymbol{q}^*}{1+\beta}, n \in \mathcal{I}_{\mathcal{A}}$.

(iii) The optimal AP partition is the same as the optimal one-tier quantizer partition $\mathbf{R}^* = (R_1^*, \ldots, R_N^*)$*, i.e.,* $\mathcal{R}^{\mathcal{A}*} = \mathbf{R}^*$.

(iv) The best possible distortion is $\frac{1}{1+\beta} D_r(X^*, R^*) + \frac{\beta}{1+\beta} \int_\Omega \|\boldsymbol{\omega} - \boldsymbol{q}^*\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega}$.

The proof of the proposition is provided in Appendix A.5. By Proposition 2, one can obtain the optimal solution for the two-tier quantization by shrinking the optimal reproduction points of the one-tier quantizer towards the corresponding FCs. Note that the algorithms AC and DC in [75] identify the optimal reproduction of the N-level one-tier quantizer as the AP deployment and then determine the FC deployment in terms of the identified AP deployment, and vice versa. In what follows, I provide an example to compare the quantizer generated

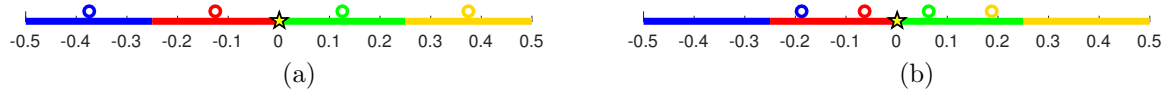


Figure 4.1: Two deployment examples in a 1-dimensional space with one FC. (a) The AC/DC two-tier quantizer. (b) The optimal quantizer. AP and FC locations are denoted by circles and star. The optimal partition cells are denoted by intervals. Each AP and its corresponding cell are illustrated by the same color.

by AC/DC with the proposed optimal quantizer, respectively. Consider one FC, N APs, and a uniform distribution over the 1-dimensional target region $\Omega = [s, t]$. The reproduction points and the cells of an optimal one-tier quantizer are given by $x_n^* = s + \frac{(2n-1)(t-s)}{2N}, n \in \mathcal{I}_{\mathcal{A}}$, and $\mathbf{R}_n^* = \left[s + \frac{(n-1)(t-s)}{N}, s + \frac{n(t-s)}{N}\right], n \in \mathcal{I}_{\mathcal{A}}$. In this case, the quantizers generated by AC and DC are identical. There is no surprise that the AP deployment and cell partition of AC/DC are exactly the reproduction points and partition of the optimal one-tier quantizer, i.e., $\boldsymbol{p}_n' = x_n^*$, $\mathcal{R}_n^{\mathcal{A}'} = R_n^*, n \in \mathcal{I}_{\mathcal{A}}$. FC location is the geometric centroid $\boldsymbol{q}' = (t+s)/2$. However, according to Proposition 2, the optimal AP deployment and cell partition are, respectively, $\boldsymbol{p}_n^* = s + \frac{\beta(t-s)}{2(1+\beta)} + \frac{(2n-1)(t-s)}{2N(1+\beta)}$ and $\mathcal{R}_n^{\mathcal{A}*} = R_n^*, n \in \mathcal{I}_{\mathcal{A}}$. Similar to the solution of AC/DC, the optimal FC location is $\boldsymbol{q}^* = \boldsymbol{q}' = (t+s)/2$. The corresponding minimum distortion is $\frac{(t-s)^2}{12(1+\beta)N^2} + \frac{\beta(t-s)^2}{12(1+\beta)}$. In particular, for $\beta = 1$ and $\Omega = [-\frac{1}{2}, \frac{1}{2}]$ with 1 FC and 4 APs, the solution of AC/DC, shown in Fig. 4.1a, is $\boldsymbol{q}' = 0$, $\mathcal{R}_n^{\mathcal{A}'} = \left[\frac{n-3}{4}, \frac{n-2}{4}\right]$, and $\boldsymbol{p}_n' = \frac{2n-5}{8}$, $n \in \mathcal{I}_{\mathcal{A}}$, indicting a distortion of $\frac{61}{768}$. But, the optimal FC location is $q^* = 0$, the optimal cells are $\mathcal{R}_n^{\mathcal{A}*} = \left[\frac{n-3}{4}, \frac{n-2}{4}\right], n \in \mathcal{I}_{\mathcal{A}}$, the optimal AP locations are $\boldsymbol{p}_n^* = \frac{2n-5}{16}, n \in \mathcal{I}_{\mathcal{A}}$, and the best possible distortion is $\frac{17}{384} < \frac{61}{768}$. The optimal two-tier quantizer is shown in Fig. 4.1b. Furthermore, the best possible distortion can be exactly determined in the high resolution regime $N \to \infty$. In fact, the best possible distortion $D_r(X^*, \mathbf{R}^*)$ of a one-tier quantizer is given by [71]

$$\kappa_d d N^{-\frac{2}{d}} \|f\|_{\frac{d}{d+2}} + o(N^{-\frac{2}{d}}), N \to \infty, \tag{4.9}$$

where $\|f\|_q \triangleq (\int_{\mathbb{R}^d} f^q(x)dx)^{\frac{1}{q}}$, and κ_d depends only on the dimension d. For example, we have $\kappa_1 = \frac{1}{12}$ and $\kappa_2 = \frac{5}{18\sqrt{3}}$ [71], [73]. By using Proposition 2, the best possible distortion

in high resolution regime $N \to \infty$ with one FC is

$$\frac{1}{1+\beta}\kappa_d d N^{-\frac{2}{d}} \|f\|_{\frac{d}{d+2}} + \frac{\beta}{1+\beta}\int_{\Omega} \|\boldsymbol{\omega} - \mathbf{c}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} + o(N^{-\frac{2}{d}}). \tag{4.10}$$

4.3 The Optimal Deployment in two-tier WSNs with Multiple FCs

In this section, I extend the analysis of the optimal deployment to WSNs with multiple FCs. Given multiple FCs, APs are divided into clusters in terms of the index map T, and the number of clusters is M. In particular, the M-th cluster is defined as $\mathcal{N}_m \triangleq \{n : \mathbf{T}(n) = m\}$. Let N_m be the number of elements in $\mathcal{N}_m$, and $W_m = \bigcup_{n\in\mathcal{N}_m} \mathcal{R}_n^{\mathcal{A}}$ be the m^{th} cluster region. Distortion in this general case is determined by (i) AP deployment, (ii) FC deployment, (iii) AP cell partition, and (iv) Clustering (or the index map from APs to FCs). Before we discuss the optimal AP and FC deployment, we need to know (a) the best index map $\mathbf{T}$ given $\boldsymbol{P}$, $\boldsymbol{Q}$, and $\mathcal{R}^{\mathcal{A}}$, and (b) the best AP partition $\mathcal{R}^{\mathcal{A}}$ given $\boldsymbol{P}$, $\boldsymbol{Q}$, and $\mathbf{T}$.

The index map T only influences the second term in (4.7). To minimize the second term, each AP should transfer data to the closest FC. Thus, the best index map is $\mathbf{T}^E_{[\boldsymbol{P},\boldsymbol{Q}]}(n) = \arg\min_m \|\boldsymbol{p}_n - \boldsymbol{q}_m\|$. However, given $\boldsymbol{P}$, $\boldsymbol{Q}$, and $\mathbf{T} = \mathbf{T}^E_{[\boldsymbol{P},\boldsymbol{Q}]}$, AP cell partition $\mathcal{R}^{\mathcal{A}}$ affects both terms in (4.7). The best AP cell partitions, called the energy Voronoi diagrams (EVDs), are

$$\boldsymbol{V}_n^E(\boldsymbol{P},\boldsymbol{Q}) = \{\boldsymbol{\omega} \big| \|\boldsymbol{p}_n - \boldsymbol{\omega}\|^2 + \beta\|\boldsymbol{p}_n - \boldsymbol{q}_{\mathbf{T}^E_{[\boldsymbol{P},\boldsymbol{Q}]}(n)}\|^2 \le \|\boldsymbol{p}_l - \boldsymbol{\omega}\|^2 + \beta\|\boldsymbol{p}_l - \boldsymbol{q}_{\mathbf{T}^E_{[\boldsymbol{P},\boldsymbol{Q}]}(l)}\|^2, \forall l \in \mathcal{I}_{\mathcal{A}}\}. \tag{4.11}$$

Now, let $\boldsymbol{V}^E(\boldsymbol{P},\boldsymbol{Q}) = \{\boldsymbol{V}_n^E(\boldsymbol{P},\boldsymbol{Q})\}_{n\in\mathcal{I}_{\mathcal{A}}}$ be the energy Voronoi partition. Putting the best

index map $\mathbf{T}^E_{[\boldsymbol{P},\boldsymbol{Q}]}(\cdot)$ and the best AP partition $\boldsymbol{V}^E(\boldsymbol{P},\boldsymbol{Q})$ into (4.7), the distortion is

$$\widetilde{\mathcal{D}}(\boldsymbol{P},\boldsymbol{Q})=\mathcal{D}(\boldsymbol{P},\boldsymbol{Q},\boldsymbol{V}^E(\boldsymbol{P},\boldsymbol{Q}),\mathbf{T}^E_{[\boldsymbol{P},\boldsymbol{Q}]})=\sum_{n=1}^{N}\int_{\boldsymbol{V}^E_n(\boldsymbol{P},\boldsymbol{Q})}\left(\|\boldsymbol{p}_n-\boldsymbol{\omega}\|^2+\beta\min_m\|\boldsymbol{p}_n-\boldsymbol{q}_m\|^2\right)f(\boldsymbol{\omega})d\boldsymbol{\omega}. \tag{4.12}$$

Let $\boldsymbol{P}^*=(\boldsymbol{p}^*_1,\ldots,\boldsymbol{p}^*_N)$ and $\boldsymbol{Q}^*=(\boldsymbol{q}^*_1,\ldots,\boldsymbol{q}^*_M)$ be, respectively, the optimal AP and FC deployments. Let $\mathbf{v}_n(\boldsymbol{P}^*,\boldsymbol{Q}^*)=\int_{\boldsymbol{V}^E_n(\boldsymbol{P}^*,\boldsymbol{Q}^*)}f(\boldsymbol{\omega})d\boldsymbol{\omega}$ be the Lebesgue measure (volume) of $\boldsymbol{V}^E_n(\boldsymbol{P}^*,\boldsymbol{Q}^*)$. Without loss of generality, we may assume $\mathbf{v}_n(\boldsymbol{P}^*,\boldsymbol{Q}^*)>0$, as quantization cells with zero probability do not affect the overall distortion.

Proposition 3. *Let $\alpha=2$ and $N>M$. The necessary conditions for the optimal deployments in the two-tier WSN with distortion defined by (4.7) are*

$$\boldsymbol{p}^*_n=\frac{\mathbf{c}_n(\boldsymbol{P}^*,\boldsymbol{Q}^*)+\beta\boldsymbol{q}^*_{\mathbf{T}^E_{[\boldsymbol{P}^*,\boldsymbol{Q}^*]}(n)}}{1+\beta},n\in\mathcal{I}_{\mathcal{A}}, \tag{4.13}$$

$$\boldsymbol{q}^*_m=\frac{\sum_{n:\mathbf{T}^E_{[\boldsymbol{P}^*,\boldsymbol{Q}^*]}(n)=m}\mathbf{c}_n(\boldsymbol{P}^*,\boldsymbol{Q}^*)\mathbf{v}_n(\boldsymbol{P}^*,\boldsymbol{Q}^*)}{\sum_{n:\mathbf{T}^E_{[\boldsymbol{P}^*,\boldsymbol{Q}^*]}(n)=m}\mathbf{v}_n(\boldsymbol{P}^*,\boldsymbol{Q}^*)},m\in\mathcal{I}_{\mathcal{B}}, \tag{4.14}$$

where $\boldsymbol{p}^*_n$ is the optimal location for AP n and $\boldsymbol{q}^*_m$ is the optimal location for FC m, $\mathbf{T}^E_{[\boldsymbol{P}^*,\boldsymbol{Q}^*]}(n)=\arg\min_m\|\boldsymbol{p}^*_n-\boldsymbol{q}^*_m\|$ is the best index map, $\mathbf{v}_n(\boldsymbol{P}^*,\boldsymbol{Q}^*)=\int_{\boldsymbol{V}^E_n(\boldsymbol{P}^*,\boldsymbol{Q}^*)}f(\boldsymbol{\omega})d\boldsymbol{\omega}$ is the Lebesgue measure (volume) of $\boldsymbol{V}^E_n(\boldsymbol{P}^*,\boldsymbol{Q}^*)$ and $\mathbf{c}_n(\boldsymbol{P}^*,\boldsymbol{Q}^*)$ is the geometric centroid of $\boldsymbol{V}^E_n(\boldsymbol{P}^*,\boldsymbol{Q}^*)$.

The proof of the proposition is provided in Appendix A.6. According to (4.13), the optimal location of AP n, connected to FC m, should be on the segment $\overline{\mathbf{c}_n(\boldsymbol{P},\boldsymbol{Q})\boldsymbol{q}_m}$. According to (4.14), the best location of FC m should be the geometric centroid of the m^{th} cluster region $\bigcup_{n:\mathbf{T}(n)=m}\boldsymbol{V}^E_n(\boldsymbol{P}^*,\boldsymbol{Q}^*)$. Obviously, the optimal deployment and the optimal partition in Proposition 2 also satisfy the necessary conditions in Proposition 3. In the next section, using Proposition 3, I design Lloyd-like algorithms to determine the optimal deployment.

First, note that when the AP cell partition is fixed, the geometric centroid $\mathbf{c}_n, n\in\mathcal{I}_{\mathcal{A}}$, and

the volume of the cells $\mathbf{v}_n, n \in \mathcal{I}_{\mathcal{A}}$, are fixed. Second, the index map $\mathbf{T}^E_{[\boldsymbol{P},\boldsymbol{Q}]}$ represents the best connection between APs and FCs (or clustering) if and only if $\boldsymbol{P}$ and $\boldsymbol{Q}$ are given. We now find the optimal deployment, the optimal partition, the optimal index map and the best possible distortion for a uniform density in one-dimensional space.

Theorem 4.1. *Let $\Omega = [s,t]$ with length $\mu(\Omega) = t - s$. Also, let*

$$\ell_a = \left(\beta + \lceil \tfrac{N}{M} \rceil^{-2}\right)^{-\frac{1}{2}}, \qquad \ell_b = \left(\beta + \lfloor \tfrac{N}{M} \rfloor^{-2}\right)^{-\frac{1}{2}} \tag{4.15}$$

$$M_a = (N \ mod \ M), \qquad M_b = M - (N \ mod \ M). \tag{4.16}$$

Then, given a uniform distribution on Ω with M FCs and N APs, the minimum distortion is

$$\frac{\mu^2(\Omega)}{12(1+\beta)} \left(M_a \ell_a + M_b \ell_b\right)^{-2}. \tag{4.17}$$

The minimum is achieved if and only if

(i) M_a of the clusters consist of $\lceil \frac{N}{M} \rceil$ APs each and have length $\ell_a \mu(\Omega)/(M_a \ell_a + M_b \ell_b)$,

(ii) M_b of the clusters consist of $\lfloor \frac{N}{M} \rfloor$ APs each and have length $\ell_b \mu(\Omega)/(M_a \ell_a + M_b \ell_b)$,

(iii) FCs are deployed at the centroids of the cluster regions,

(iv) AP cells are uniform partitions of the cluster, and

(v) AP n is deployed at $\frac{\mathbf{c}_n + \beta \boldsymbol{q}_{\mathbf{T}(n)}}{1+\beta}$, $n \in \mathcal{I}_{\mathcal{A}}$, where $\mathbf{c}_n$ is the geometric centroid of AP n's cell.

The proof of the theorem provided in Appendix A.7.

In particular, when $K = \frac{N}{M}$ is a positive integer, the optimal FC locations are $\boldsymbol{q}^*_m = s + \frac{(2m-1)(t-s)}{2M}, m \in \mathcal{I}_{\mathcal{B}}$, and the optimal index map is $\mathbf{T}^*(n) = \left\lceil \frac{n}{K} \right\rceil, n \in \mathcal{I}_{\mathcal{A}}$. The

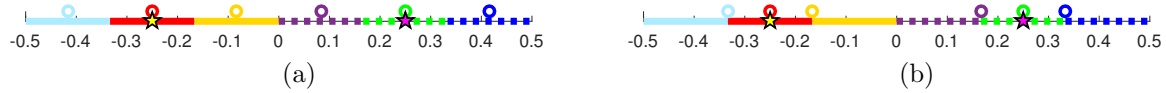


Figure 4.2: Two deployment examples in a 1-dimensional space with two FCs. (a) The AC/DC two-tier quantizer. (b) The optimal two-tier quantizer. AP and FC locations are denoted by circles and stars. The optimal partition cells are denoted by intervals. Each AP and its corresponding cell are illustrated by the same color. The two clusters are denoted by solid and dashed lines.

optimal AP locations are $\boldsymbol{p}_n^* = s + \frac{(t-s)}{(1+\beta)}\left(\frac{(2n-1)}{2N} + \beta\frac{(2\lceil\frac{n}{K}\rceil-1)}{2M}\right), n \in \mathcal{I}_{\mathcal{A}}$, and the optimal AP cell partitions are $\mathcal{R}_n^{\mathcal{A}*} = \left[s + \frac{(n-1)(t-s)}{N}, s + \frac{n(t-s)}{N}\right], n \in \mathcal{I}_{\mathcal{A}}$. The corresponding minimum distortion is $\frac{(t-s)^2}{12(1+\beta)M^2}\left(\frac{1}{K^2} + \beta\right)$. However, in such a uniform distributed 1-dimensional scenario, the quantizers generated by AC [75] and DC [75] are identical but different from the above optimal solution. The corresponding FC and AP deployments of AC/DC are, respectively, the optimal M-level and N-level one-tier reproduction points, i.e., $\boldsymbol{p}'_n = s + \frac{(2n-1)(t-s)}{2N}, \boldsymbol{q}'_m = s + \frac{(2m-1)(t-s)}{2M}, n \in \mathcal{I}_{\mathcal{A}}, m \in \mathcal{I}_{\mathcal{B}}$. The corresponding AP cell partitions are $\mathcal{R}_n^{\mathcal{A}'} = \left[s + \frac{(n-1)(t-s)}{N}, s + \frac{n(t-s)}{N}\right], n \in \mathcal{I}_{\mathcal{A}}$. In what follows, I provide an example to elucidate the gap between AC/DC quantizer and the optimal quantizer. For $\beta = 1$ and $\Omega = [-\frac{1}{2}, \frac{1}{2}]$ with 2 FC and 6 APs, the AC/DC quantizer, illustrated by Fig. 4.2a, is $\boldsymbol{Q}' = \{-\frac{1}{4}, \frac{1}{4}\}$, $\boldsymbol{P}' = \{-\frac{5}{12}, -\frac{3}{12}, -\frac{1}{12}, \frac{1}{12}, \frac{3}{12}, \frac{5}{12}\}$, and $\mathcal{R}_n^{\mathcal{A}'} = \left[\frac{n-4}{6}, \frac{n-3}{6}\right]$, $n \in \mathcal{I}_{\mathcal{A}}$. The corresponding distortion is thus $\frac{49}{2592}$. Nonetheless, according to Theorem 4.1, the optimal FC deployment is $\boldsymbol{Q}^* = \{-\frac{1}{4}, \frac{1}{4}\}$, the optimal AP deployment is $\boldsymbol{P}^* = \{-\frac{1}{3}, -\frac{1}{4}, -\frac{1}{6}, \frac{1}{6}, \frac{1}{4}, \frac{1}{3}\}$, the optimal AP cells are $\mathcal{R}_n^{\mathcal{A}*} = \left[\frac{n-4}{6}, \frac{n-3}{6}\right], n \in \mathcal{I}_{\mathcal{A}}$, and the optimal index map is $\mathbf{T}^*(n) = \lceil\frac{n}{3}\rceil, n \in \mathcal{I}_{\mathcal{A}}$. The optimal quantizer, shown in Fig. 4.2b, gains the minimum distortion, $\frac{5}{432}$ which is much smaller than $\frac{49}{2592}$. The distortion gap holds for more complicated circumstances, e.g., non-uniform distributed 2-dimensional space. Therefore, in order to approach the optimal distortion, it is necessary to designed a better two-tier quantizer. This is the topic of the discussion in Section 4.5.

4.4 AP-Sensor Power Function

As explained before, minimizing the distortion in (4.7) is the unconstrained version of minimizing the sum of the AP powers when the sum of the sensor powers is constrained or vise versa. In this section, we study the minimization of the sum of the AP powers when the sum of the sensor powers is constrained via analyzing the AP-Sensor power function in (4.5). An AP-Sensor power pair (x, y) is said to be achievable if there exists a solution $(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T})$ with Sensor-power $\mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}}) \leq x$ and AP-power $\mathcal{P}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) = y$. $(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T})$ is said to be a feasible solution for the AP-Sensor power pair $(S, A(S))$ if and only if $\mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}}) \leq S$ and $\mathcal{P}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) = A(S)$. Let $F(S)$ be the set of the feasible solutions for the point $(S, A(S))$, and $\hat{F} = \bigcup_{S} F(S)$ be the set of all feasible solutions for all points on the curve $A(S)$. It is self-evident that every point above the curve $A(S)$ is achievable. We now discuss the convexity of the AP-Sensor power function. By parallel axis theorem, (4.2) can be rewritten as

$$\mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}}) = \sum_{n=1}^{N} \int_{\mathcal{R}_n^{\mathcal{A}}} \|\mathbf{c}_n - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} + \sum_{n=1}^{N} \|\boldsymbol{p}_n - \mathbf{c}_n\|^2 \mathbf{v}_n, \tag{4.18}$$

where $\mathbf{c}_n = \frac{\int_{\mathcal{R}_n^{\mathcal{A}}} \boldsymbol{\omega} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}{\int_{\mathcal{R}_n^{\mathcal{A}}} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}$ is the centroid of $\mathcal{R}_n^{\mathcal{A}}$, $\mathbf{v}_n = \int_{\mathcal{R}_n^{\mathcal{A}}} f(\boldsymbol{\omega}) d\boldsymbol{\omega}$ is the volume of $\mathcal{R}_n^{\mathcal{A}}$. Note that $\mathbf{c}_n$ and $\mathbf{v}_n$ are functions of $\mathcal{R}^{\mathcal{A}}$. $\mathbf{c}_n$ and $\mathbf{v}_n$ are constants if and only if the AP cell partition $\mathcal{R}^{\mathcal{A}}$ is fixed. Furthermore, AP-power, (4.1), can be rewritten as

$$\mathcal{P}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) = \sum_{n=1}^{N} \|\boldsymbol{p}_n - \boldsymbol{q}_{\mathbf{T}(n)}\|^2 \mathbf{v}_n. \tag{4.19}$$

The representations (4.18) and (4.19) are convenient for the following convexity analysis.

Lemma 8. *Let $D_R(k)$ be the minimum distortion of the k-level one-tier quantizer on space Ω. Let M and N be, respectively, the number of FCs and the number of APs, where $N > M$. The AP-Sensor power function is a non-increasing function with the domain $[D_R(N), +\infty)$ such that $A(S) > 0$ when $D_R(N) \leq x < D_R(M)$ and $A(S) = 0$ when $x \geq D_R(M)$.*

The proof of the Lemma is provided in Appendix A.8.

4.4.1 Closed-form formulas and convexity for one FC

In this section, we assume only one FC and derive a closed-form analytical formula for the AP-Sensor power function. We also prove that the AP-sensor power function is convex.

Lemma 9. *When one FC $\boldsymbol{Q} = (\boldsymbol{q})$ and multiple APs $\boldsymbol{P} = (\boldsymbol{p}_1, \ldots, \boldsymbol{p}_N)$ are provided, and $\mathcal{R}^{\mathcal{A}}$ is fixed, the minimum $\mathcal{P}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T})$ with the constraint $\mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}}) \leq S$ is defined by*

$$\widehat{A}(S, \mathcal{R}^{\mathcal{A}}) \triangleq \inf_{(\boldsymbol{P}, \boldsymbol{Q}, \mathbf{T}): \mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}}) \leq S} \mathcal{P}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}). \tag{4.20}$$

We have the following results: (i) The domain of $\widehat{A}(S, \mathcal{R}^{\mathcal{A}})$ is $\{(S, \mathcal{R}^{\mathcal{A}}) | H(\mathcal{R}^{\mathcal{A}}) \leq S\}$; (ii) When $(S, \mathcal{R}^{\mathcal{A}}) \in \{(S, \mathcal{R}^{\mathcal{A}}) | H(\mathcal{R}^{\mathcal{A}}) \leq S, D_R(N) \leq S < D_R(M)\}$, we have

$$\begin{cases} \widehat{A}(S, \mathcal{R}^{\mathcal{A}}) = \left[\sqrt{(S - H(\mathcal{R}^{\mathcal{A}}))} - \sqrt{D_R(1) - H(\mathcal{R}^{\mathcal{A}})}\right]^2, & S \in [D_R(N), D_R(1)) \\ 0, & S \in [D_R(1), +\infty) \end{cases} \tag{4.21}$$

where $H(\mathcal{R}^{\mathcal{A}}) = \sum_{n=1}^{N} \int_{\mathcal{R}_n^{\mathcal{A}}} \|\mathbf{c}_n - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega}$, $\mathbf{c}_n = \frac{\int_{\mathcal{R}_n^{\mathcal{A}}} \boldsymbol{\omega} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}{\int_{\mathcal{R}_n^{\mathcal{A}}} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}$ is the centroid of $\mathcal{R}_n^{\mathcal{A}}$, and $\mathbf{v}_n = \int_{\mathcal{R}_n^{\mathcal{A}}} f(\boldsymbol{\omega}) d\boldsymbol{\omega}$ is the volume of $\mathcal{R}_n^{\mathcal{A}}$.

The proof is provided in Appendix A.9.

Theorem 4.2. *When one FC and multiple APs are provided, i.e., $M = 1$ and $N > 1$, the AP-Sensor power function $A(S)$ is a convex function with the expression*

$$A(S) = \begin{cases} \left[\sqrt{S - D_R(N)} - \sqrt{D_R(1) - D_R(N)}\right]^2, & S \in [D_R(N), D_R(1)) \\ 0, & S \in [D_R(1), +\infty) \end{cases} \tag{4.22}$$

The proof of the theorem is provided in Appendix A.10. When $M = 1$, as an inverse function of $A(S)$, the Sensor-AP power function is also a convex function with the expression

$$S(A) = \begin{cases} D_R(N) + \left[\sqrt{D_R(1) - D_R(N)} - \sqrt{A}\right]^2, & A \in [0, D_R(1) - D_R(N)) \\ D_R(N), & A \in [D_R(1) - D_R(N), +\infty) \end{cases} \tag{4.23}$$

Therefore, for $M = 1$, finding the solution of $A(S)$ on $S \in [D_R(N), D_R(1))$ is equivalent to (i) finding the minimizer of $\mathcal{P}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) + \lambda \mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \boldsymbol{Q})$, where the Lagrange Multiplier $\lambda = -\left(\frac{\partial A(S)}{\partial S}\right) = \sqrt{\frac{D_R(1) - D_R(N)}{S - D_R(N)}} - 1$ is the negative derivative at $(S, A(S))$ and (ii) finding the minimizer of $\mathcal{D}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) = \mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \boldsymbol{Q}) + \beta \mathcal{P}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T})$, where the Lagrange Multiplier $\beta = -\left(\frac{\partial A(S)}{\partial S}\right)^{-1} = \frac{1}{\sqrt{\frac{D_R(1) - D_R(N)}{S - D_R(N)}} - 1}$ is the negative reciprocal of the derivative at $(S, A(S))$.

In what follows, I provide an example for the 1-dimensional space $[s, t]$ and a uniform distribution. On one hand, due to the quantization theorem [59], we have $D_R(1) = \frac{(t-s)^2}{12}$ and $D_R(N) = \frac{(t-s)^2}{12N^2}$. Therefore, (4.22) becomes

$$A(S) = S - \frac{(t-s)^2}{6N^2} + \frac{(t-s)^2}{12} - 2\sqrt{\left(S - \frac{(t-s)^2}{12N^2}\right)\left(\frac{(t-s)^2}{12} - \frac{(t-s)^2}{12N^2}\right)}. \tag{4.24}$$

On the other hand, let $(\boldsymbol{P}^*, \boldsymbol{Q}^*, \mathcal{R}^{\mathcal{A}*}, \mathbf{T}^*)$ be an optimal solution for $\mathcal{D}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T})$ with $\beta = \frac{1}{\sqrt{\frac{D_R(1)-D_R(N)}{S-D_R(N)}-1}}$. By Proposition 2, we have $\mathcal{P}^{\mathcal{S}}(\boldsymbol{P}^*, \mathcal{R}^{\mathcal{A}*}) = S$ and

$$\begin{aligned}\mathcal{P}^{\mathcal{A}}(\boldsymbol{P}^*, \boldsymbol{Q}^*, \mathcal{R}^{\mathcal{A}*}, \mathbf{T}^*) &= S - \frac{(t-s)^2}{6N^2} + \frac{(t-s)^2}{12} \\ &-2\sqrt{\left(S - \frac{(t-s)^2}{12N^2}\right)\left(\frac{(t-s)^2}{12} - \frac{(t-s)^2}{12N^2}\right)},\end{aligned} \tag{4.25}$$

which is the same as (4.24).

4.5 Node deployment Algorithms

We introduce three algorithms, One-Tiered Lloyd (OTL), Two-Tiered Lloyd (TTL), and Combining Lloyd (CL), to minimize the distortion in two-tier WSNs. First, we quickly review the conventional Lloyd algorithm. Lloyd Algorithm has two basic steps in each iteration: (i) The node deployment is optimized while the partitioning is fixed; (ii) The partitioning is optimized while the node deployment is fixed. As shown in [32], Lloyd algorithm, which provides good performance and is simple enough to be implemented distributively, can be used to solve one-tier quantizers or one-tier node deployment problems. However, the conventional Lloyd Algorithm cannot be applied to two-tier WSNs where two kinds of nodes are deployed. We thus introduce three Lloyd-based algorithms to solve the optimal deployment problem in two-tier WSNs.

4.5.1 One-tier Lloyd Algorithm

OTL combines two independent Lloyd Algorithms. Using the Lloyd algorithm, an M-level one-tier quantizer is designed and its reproduction points are used as $\boldsymbol{Q}$. Another N-level one-tier quantizer is designed and its partition is used as $\mathcal{R}^{\mathcal{A}}$. The index map is determined by $\mathbf{T}(n) = \arg\min_m \|\boldsymbol{p}'_n - \boldsymbol{q}_m\|$ and the deployment $\boldsymbol{P}$ is determined by $\boldsymbol{p}_n = \min_m \frac{\boldsymbol{p}'_n + \beta \boldsymbol{q}_m}{1+\beta}, n \in \mathcal{I}_{\mathcal{A}}$, where $\boldsymbol{p}'_n$ is the n^{th} reproduction point obtained by the N-level

quantizer. Using Proposition 2, it is easy to show that, for the networks with one FC, the distortion of OTL converges to the minimum as long as the second Lloyd Algorithm provides the optimal N-level quantizer.

4.5.2 Two-tier Lloyd Algorithm

Before introducing the details of TTL, I introduce two concepts: (i) AP local distortion and (ii) FC local distortion. The AP local distortion is defined as

$$\mathcal{D}_n^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) = \int_{\mathcal{R}_n^{\mathcal{A}}} \left[\|\boldsymbol{p}_n - \boldsymbol{\omega}\|^2 + \beta \|\boldsymbol{p}_n - \boldsymbol{q}_{\mathbf{T}(n)}\|^2 \right] f(\boldsymbol{\omega}) d\boldsymbol{\omega}.$$

The total distortion is the sum of the AP local distortions, i.e.,

$$\mathcal{D}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) = \sum_{n=1}^{N} \mathcal{D}_n^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}).$$

Similarly, for $\mathcal{N}_m \triangleq \{n : \mathbf{T}(n) = m\}$, the FC local distortion is defined as

$$\mathcal{D}_m^{\mathcal{B}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) = \sum_{n \in \mathcal{N}_m} \int_{\mathcal{R}_n^{\mathcal{A}}} \left(\|\boldsymbol{p}_n - \boldsymbol{\omega}\|^2 + \beta \|\boldsymbol{p}_n - \boldsymbol{q}_m\|^2 \right) f(\boldsymbol{\omega}) d\boldsymbol{\omega}.$$

The total distortion is the summation of these FC local distortions, i.e., $\mathcal{D}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) = \sum_{m=1}^{M} \mathcal{D}_m^{\mathcal{B}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T})$. Let $\mathbf{c}_n$ and $\mathbf{v}_n$ be, respectively, the geometric centroid and the volume of the current AP cell partition. TTL iterates over fthe steps: (i) AP n moves to $\frac{\mathbf{c}_n + \beta \boldsymbol{q}_{\mathbf{T}(n)}}{1+\beta}$; (ii) AP partitioning is done by assigning the corresponding EVD to each AP node; (iii) FC m moves to $\frac{\sum_{n \in \mathcal{N}_m} \boldsymbol{p}_n \mathbf{v}_n}{\sum_{n \in \mathcal{N}_m} \mathbf{v}_n}$; (iv) Clustering is done by assigning the nearest FC to each AP. Furthermore, to avoid the APs with a zero-measure partition, we move such an AP towards a randomly selected FC q until the distance to the q is $min_n \|\boldsymbol{p}_n - q\|$ after (ii).
In what follows, I show that the distortion with TTL converges. First, due to the parallel

axis theorem and (A.52), the local distortion of AP n can be rewritten as

$$\begin{aligned}\mathcal{D}_n^{\mathcal{A}}(\boldsymbol{P},\boldsymbol{Q},\mathcal{R}^{\mathcal{A}},\mathbf{T}) &= \frac{1}{1+\beta}\int_{\mathcal{R}_n^{\mathcal{A}}}\|\mathbf{c}_n-\boldsymbol{\omega}\|^2 f(\boldsymbol{\omega})d\boldsymbol{\omega}\\ &+(1+\beta)\|\boldsymbol{p}_n-\widehat{\boldsymbol{p}}_n\|^2\mathbf{v}_n+\frac{\beta}{1+\beta}\int_{\mathcal{R}_n^{\mathcal{A}}}\|\boldsymbol{\omega}-\boldsymbol{q}_{\mathbf{T}(n)}\|^2 f(\boldsymbol{\omega})d\boldsymbol{\omega},\end{aligned} \tag{4.26}$$

where $\widehat{\boldsymbol{p}}_n=\frac{\mathbf{c}_n+\beta\boldsymbol{q}_{\mathbf{T}(n)}}{1+\beta}$. When $\boldsymbol{Q}$, $\mathcal{R}^{\mathcal{A}}$, and T are given, the first term and the third term of (4.26) are constants. In other words, the AP local distortion becomes a function of $\|\boldsymbol{p}_n-\widehat{\boldsymbol{p}}_n\|$. Therefore, Step (i) does not increase the AP local distortions and then the total distortion. Second, given $\boldsymbol{P}$, $\boldsymbol{Q}$ and $\mathbf{T}$, EVDs minimize the total distortion, indicating that the total cost is not increased by Step (ii). Third, observe that for $\widehat{\boldsymbol{q}}_m=\frac{\sum_{n\in\mathcal{N}_m}\boldsymbol{p}_n\mathbf{v}_n}{\sum_{n\in\mathcal{N}_m}\mathbf{v}_n}$, we have $\sum_{n\in\mathcal{N}_m}\mathbf{v}_n\|\boldsymbol{p}_n-\boldsymbol{q}_m\|^2=\sum_{n\in\mathcal{N}_m}\mathbf{v}_n\left[\|\boldsymbol{p}_n-\widehat{\boldsymbol{q}}_m\|^2+\|\boldsymbol{q}_m-\widehat{\boldsymbol{q}}_m\|^2\right]$. Therefore, the local distortion of FC m can be rewritten as

$$\begin{aligned}\mathcal{D}_n^{\mathcal{B}}(\boldsymbol{P},\boldsymbol{Q},\mathcal{R}^{\mathcal{A}},\mathbf{T}) &= \sum_{n\in\mathcal{N}_m}\int_{\mathcal{R}_n^{\mathcal{A}}}\|\boldsymbol{p}_n-\boldsymbol{\omega}\|^2 f(\boldsymbol{\omega})d\boldsymbol{\omega}\\ &+\beta\left(\sum_{n\in\mathcal{N}_m}\mathbf{v}_n\right)\|\boldsymbol{q}_m-\widehat{\boldsymbol{q}}_m\|^2+\beta\sum_{n\in\mathcal{N}_m}\left(\mathbf{v}_n\|\boldsymbol{p}_n-\widehat{\boldsymbol{q}}_m\|^2\right).\end{aligned} \tag{4.27}$$

When $\boldsymbol{P}$, $\mathcal{R}^{\mathcal{A}}$, and $\mathbf{T}$ are given, the first term and the third term in (4.27) are constants. In other words, the FC local distortion becomes a function of $\|\boldsymbol{q}_m-\widehat{\boldsymbol{q}}_m\|$. Therefore, Step (iii) does not increase the FC local distortions and then the total distortion. Last, given $\boldsymbol{P}$, $\boldsymbol{Q}$, and $\mathcal{R}^{\mathcal{A}}$, $\mathbf{T}^E_{[\boldsymbol{P},\boldsymbol{Q}]}(n)=\arg\min_m\|\boldsymbol{p}_n-\boldsymbol{q}_m\|$ minimizes the total distortion, indicating that the total distortion is not increased by Step (iv). In other words, the algorithm generates a positive non-increasing sequence of distortion values and therefore will converge.

4.5.3 Combining Lloyd Algorithm

Note that there is no guarantee to achieve a minimum distortion with OTL when $M>1$. The distortion with TTL converges to the local minimum, however, depends on the initial

deployments. Our simulations show that starting with some initial deployments, TTL ends with large minimum distortions. A natural idea to avoid these issues is to combine the two algorithms. First, OTL is applied to obtain a deployment with small distortion as the initial deployment for TTL. Second, TTL is applied to further decrease the distortion. We refer to such an algorithm as Combining Lloyd (CL) Algorithm.

4.6 Performance Evaluation

We provide the simulation results in two two-tier WSNs: (i) WSN1: A two-tier WSN including one FC and 20 APs; (ii) WSN2: A two-tier WSN including 4 FCs and 20 APs. Similar to [6], [32], the target region is set to $\Omega = [0, 10]^2$ and β is set to 1. The traffic density function is the sum of five Gaussian functions of the form $5\exp(0.5(-(x-x_c)^2-(y-y_c)^2))$, where centers (x_c, y_c) are (8,1), (4,9), (7.6,7.6), (9.4,5), and (2,2). We generate 50 initial AP and FC deployments on Ω randomly, i.e, every node location is generated with uniform distribution on Ω. For each initial AP and FC deployments, we connect every AP to its closest FC and then assign the corresponding EVD to the AP node. The maximum number of iterations is set to 100. FCs and APs are denoted, respectively, by colored five-pointed stars and colored circles. The corresponding geometric centroid of AP cells are denoted by colored crosses. Each FC and its connected APs form a cluster. To make clusters more visible, the symbols in the same cluster are filled with the same color. As discussed in Section 4.1, the distortion is the weighted power. We compare the distortion of my three algorithms (OTL, TTL, and CL) with Minimum Energy Routing (MER), Agglomerative Clustering (AC) [75], and Divisive Clustering (DC) [75] algorithms. AC and DC are two clustering methods applied to multi-tier networks. MER combines Voronoi Partition [33] and Bellman-Ford algorithms. On one hand, Voronoi Partition is the optimal cell partition in the first tier (one-tier WSNs). On the other hand, when edge costs are set as the AP powers, Bellman-Ford Algorithm provides the flow with the minimum energy consumption

[70], [120], indicating the optimal routing protocol in one tier network. Note that multi-hop communication among sensors is unnecessary because of the small distance between a sensor and its corresponding AP [68]. Therefore, the existing routing protocol, Bellman-Ford, is only applied to the second tier.

Figs. 4.3a, 4.3b, 4.3c, and 4.3d show one example of the final deployments of the four algorithms (MER, AC, DC, and CL) in WSN1. For MER, the multi-hop paths are denoted by black dotted lines in 4.3a. From Fig. 4.3d, we can find that APs are placed on the line between the corresponding FCs and cell centroids, as expected from Proposition 3. Compared to MER, CL saves 41.90% of the weighted total power in WSN1. However, AC and DC consume more energy compared to MER. Figs. 4.4a, 4.4b, 4.4c, and 4.4d show examples of the final deployments of MER, AC, DC, and CL, in WSN2, respectively. Compared to MER, the AC, DC, and CL save, respectively, 60.69%, 69.92%, and 81.55% of the weighted total power. Moreover, in Figs. 4.3 and 4.4, the FC deployments of the three algorithms, AC, DC, and CL, are approximately the same, but the AP deployments of AC and DC are more uniform than that of CL. Intuitively, compared to the MER, AC, and DC Algorithms, the APs are deployed closer to FCs and then produce smaller AP powers in CL Algorithm, which justifies my observation that CL saves more power compared to the other three algorithms.

Figs 4.5a and 4.5b illustrate the weighted total power (4.7) of different algorithms in WSN1 and WSN2. Our algorithms, OTL, TTL, and CL, outperform MER, AC, and DC in both networks, especially when β is large (the energy consumed on APs is significant). In WSN1, the three proposed algorithms have almost the same weighted power savings. Although it cannot be seen from the figure, for WSN2, TTL's power saving relative to MER is about 1% better than that of OTL. Comparing the two schemes directly, the power saving of TTL over that of OTL is about 1-3%. Besides, CL, whose performance is better than OTL and TTL in WSN2, requires the shortest running time among the three algorithms. When $\beta = 1$, the running time of repeating each algorithm 50 times is provided in Table 4.1. Note that

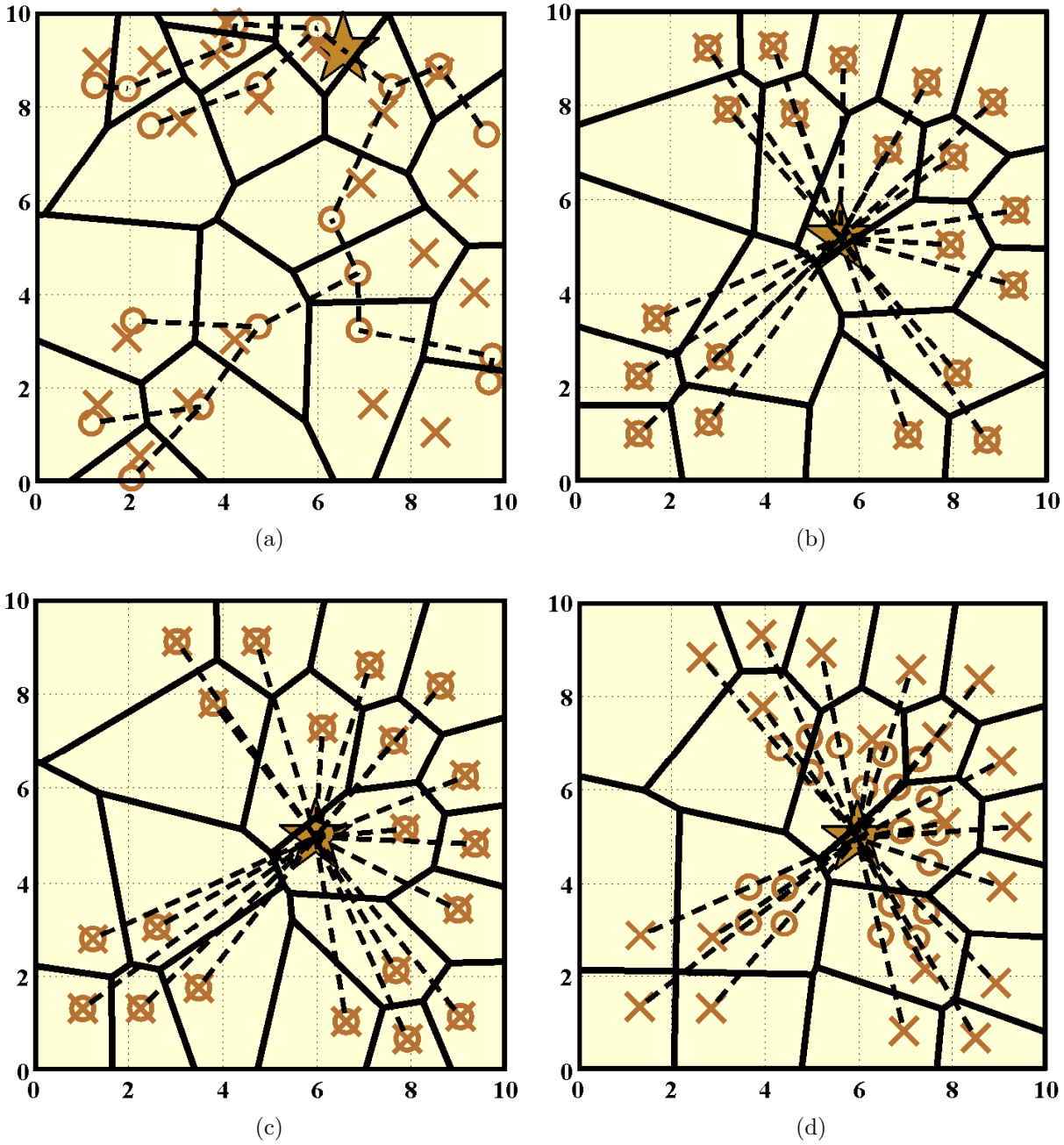


Figure 4.3: AP and FC deployments of different algorithms in WSN1. (a) MER. (b) AC (c) DC. (d) CL.

the running time is dominated by the calculation of the partition. OTL calculates Voronoi partitions for both FCs and APs. TTL calculates energy Voronoi partitions for APs only. Therefore, the running time of OTL increases as we increase the number of FCs. On the contrary, the running times of TTL are almost the same because WSN1 and WSN2 have

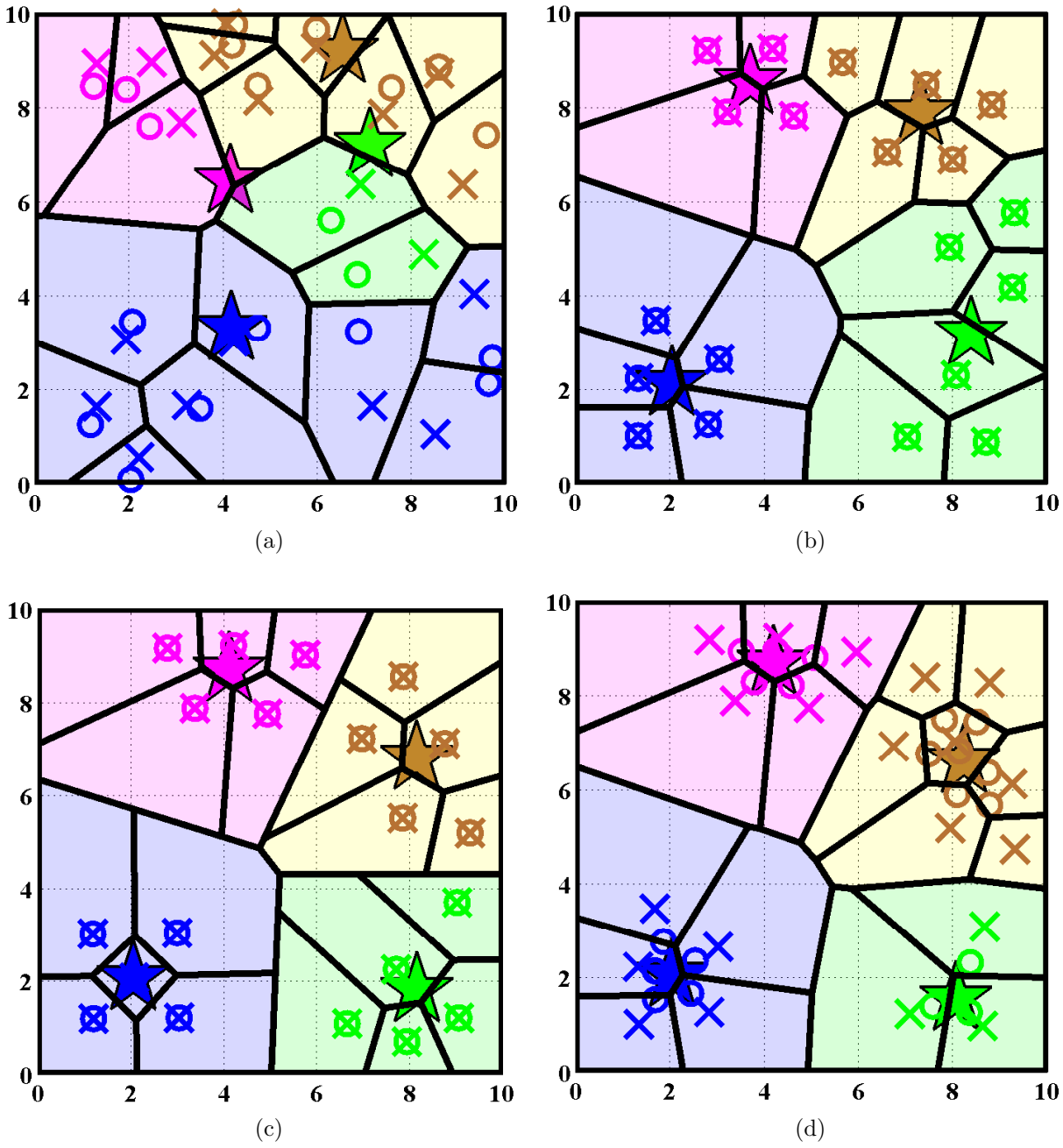


Figure 4.4: AP and FC deployments of different algorithms in WSN2. (a) MER. (b) AC (c) DC. (d) CL.

the same number of APs. CL, as a combination of OTL and TTL, is associated with the running time that is related to the number of FCs. Besides, TTL spends a lot of time to relocate APs with zero-measure partitions because of the bad initial deployments. However, CL attains a good deployment by operating OTL before TTL which greatly reduces the time

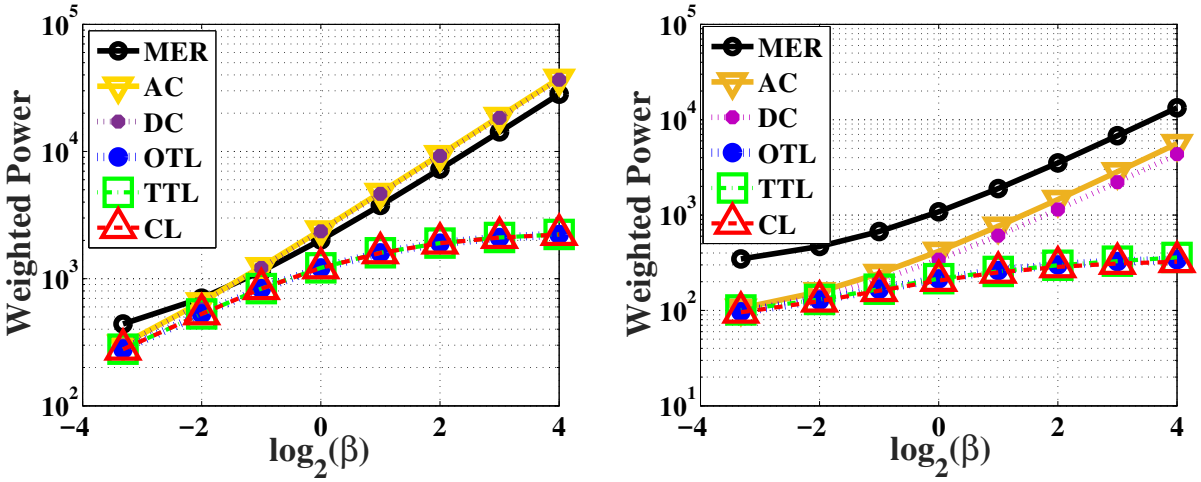


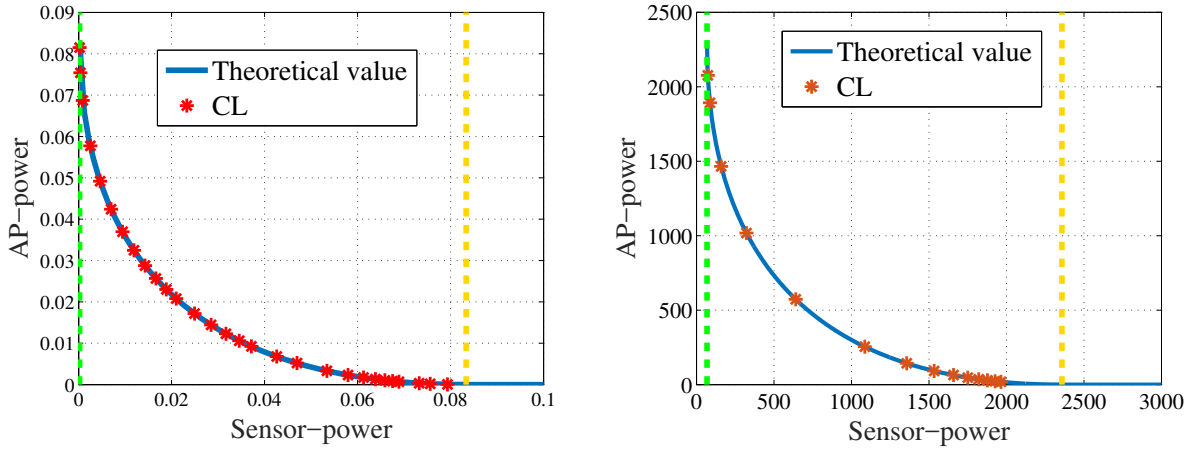
Figure 4.5: The weighted power comparison of different Algorithms. (a) WSN1. (b) WSN2.

consumed by AP relocation. Consequently, as shown in Table 4.1, CL requires the shortest running time among the three algorithms.

Figs. 4.6a and 4.6b illustrate the comparison between the AP Sensor power pair $(\mathcal{P}^{\mathcal{S}}, \mathcal{P}^{\mathcal{A}})$ using CL with different values of β and the AP-Sensor power function $A(S)$ in (4.5). One FC and twenty APs are provided, i.e., $M = 1$ and $N = 20$. Fig. 4.6a is based on a one-dimensional region uniformly distributed in $[0, 1]$ and Fig. 4.6b is based on WSN1. The green dotted line shows the value of $D_R(20)$ and the orange dotted line shows the value of $D_R(1)$. $D_R(1)$ and $D_R(20)$ in Fig. 4.6a are theoretical values derived from the quantization theory, i.e., $D_R(1) = \frac{1}{12}$ and $D_R(20) = \frac{1}{12*20^2}$. $D_R(1)$ and $D_R(20)$ in Fig. 4.6b are obtained from repeating Lloyd Algorithm and choosing the smallest distortion. From Figs. 4.6a and 4.6b, we find that the AP Sensor power pair $(\mathcal{P}^{\mathcal{S}}, \mathcal{P}^{\mathcal{A}})$ obtained from CL matches the theoretical AP-Sensor power function very well.

Table 4.1: Running times(s)

Algorithms	WSN1	WSN2
OTL	1931.41	2196.03
TTL	7078.12	7047.40
CL	1858.88	1968.30



(a) (b)

Figure 4.6: The comparison between the AP-Sensor power function and the performance of TTL: (a) One-dimensional region uniformly distributed in [0,1]. (b) WSN1.

Chapter 5

Movement-efficient Sensor Deployment in Wireless Sensor Networks

In this Chapter, I study a mobile wireless sensor network (MWSN) consisting of multiple mobile sensors or robots. To take all the three factors (sensing quality, energy consumption, and connectivity) into consideration, I model the sensor deployment problem as a constrained optimization problem. The main goal is to find an optimal sensor deployment (or relocation) to optimize the sensing quality with a limited communication range and a specific network lifetime constraint. Then, the necessary conditions for the optimal sensor deployment in both homogeneous and heterogeneous MWSNs are derived. Using these necessary conditions, both centralized and distributed algorithms are provide to make a flexible and explicit trade-off between sensing uncertainty and network lifetime. The proposed algorithms are successfully extended to more applications, such as area coverage and target coverage, via properly selected density functions.

5.1 System Model and Problem Formulation

Let $\Omega \in \mathbb{R}^2$ be a convex target region including its interior. Given N sensors in the target area Ω, sensor deployment before and after the relocation are, respectively, defined by $\mathbf{P}^0 = (\mathbf{p}_1^0, \ldots, \mathbf{p}_N^0) \subset \Omega^N$ and $\mathbf{P} = (\mathbf{p}_1, \ldots, \mathbf{p}_N) \subset \Omega^N$, where $\mathbf{p}_n^0$ is Sensor n's initial location and $\mathbf{p}_n$ is Sensor n's final location. Let $\mathcal{I}_\Omega = \{1, \ldots, N\}$ be the set of sensors in the MWSN. A cell partition $\mathcal{R}(\mathbf{P})$ is defined as a collection of N disjoint subsets, $\{\mathcal{R}_n(\mathbf{P})\}_{n \in \mathcal{I}_\Omega}$, whose union is Ω. We assume that Sensor n only monitors the events that occurred in its cell partition $\mathcal{R}_n(\mathbf{P})$, $\forall n \in \mathcal{I}_\Omega$. Let $\|\cdot\|$ denote the Euclidean distance, $card(\mathcal{A})$ be the number of elements in set $\mathcal{A}$, ∂W be the boundary of a region $W \subset \Omega$, and $\mathbb{B}(c, r) = \{\boldsymbol{\omega} |\ \|\boldsymbol{\omega} - c\| \leq r\}$ be a ball centered at c with radius r.

In the binary disk communication model [2]–[9], [20]–[22], two sensor nodes can establish reliable communications within one hop if and only if the distance between the two is smaller than R_c, where R_c is referred to as the communication range. We define the access point (AP) as the sensor node that can communicate with the outside information world. Without loss of generality, we assume that Sensor 1 acts as the AP. Other sensor nodes can transfer data outside if and only if there exist paths from the sensors to the AP. Each path consists of a sequence of sensor nodes where each hop distance is smaller than R_c. Sensor nodes that are connected to the AP via one-hop or multi-hop communications construct the backbone network. Let $\mathcal{S}(\mathbf{P})$ be the backbone network when the sensor deployment is $\mathbf{P}$. The sensors in $\mathcal{S}(\mathbf{P})$ are referred to as active sensors while the sensors outside of $\mathcal{S}(\mathbf{P})$ are referred to as inactive sensors. Accordingly, I define the active sensor deployment, $\mathcal{H}(\mathbf{P})$, as the vector of locations of active sensors. In particular, we have $\mathbf{P} = \mathcal{H}(\mathbf{P})$ and $card(\mathcal{S}(\mathbf{P})) = n$ when $\mathcal{S}(\mathbf{P})$ includes all sensor nodes. If all sensors are included in the backbone network, we call the network fully connected. Otherwise, the network is divided into several disconnected sub-graphs. For convenience, we assume that the initial sensor deployment constructs a fully connected network, i.e., $\mathbf{P}^0 = \mathcal{H}(\mathbf{P}^0)$.

To evaluate the sensing uncertainty in heterogeneous MWSNs, we consider the distortion function [32]–[34] defined by

$$\mathcal{D}(\mathbf{P}) = \sum_{n=1}^{N} \int_{\mathcal{R}_n(\mathbf{P})} \eta_n \|\mathbf{p}_n - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega}, \tag{5.1}$$

where the sensing cost parameters $\eta_n \in (0, 1]$ are constants that depend on Sensor n's characteristics, e.g. sensitivity, and $f(\boldsymbol{\omega}) : \Omega \to \mathbb{R}_+$ is a spatial density function that reflects the frequency of random events taking place over the target region. In homogeneous MWSNs, sensors have identical parameters, i.e., $\eta_n = \eta, \forall n \in I_\Omega$. Note that the sensing uncertainty is only determined by the final deployment $\mathbf{P}$.

However, as explained previously, when the communication range R_c is limited, some sensor nodes cannot transfer their data back to the AP. As a result, only the sensor nodes in the backbone network can contribute to the sensing and therefore the performance should be revised as

$$\mathcal{D}(\mathbf{P}) = \sum_{n \in \mathcal{S}(\mathbf{P})} \int_{\mathcal{R}_n(\mathcal{H}(\mathbf{P}))} \eta_n \|\mathbf{p}_n - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega}, \tag{5.2}$$

The optimal partition for the performance function (5.2) is Multiplicatively Weighted Voronoi Diagram (MWVD) [32], which can be applied to both homogeneous and heterogeneous MWSNs. The MWVD of Ω generated by $\mathbf{P}$ is the collection of sets $\{\boldsymbol{V}_n^H(\mathbf{P})\}_{n \in I_\Omega}$ defined by

$$\boldsymbol{V}_n^H(\mathbf{P}) = \{\boldsymbol{\omega} \in \Omega | \eta_n \|\boldsymbol{\omega} - \mathbf{p}_n\|^2 \leq \eta_m \|\boldsymbol{\omega} - \mathbf{p}_m\|^2, \forall m \in I_\Omega\}. \tag{5.3}$$

In particular, the MWVD for homogeneous MWSNs degenerates to the Voronoi Diagram [33]. From now on, we use $\boldsymbol{V}^H(\mathbf{P}) = \{\boldsymbol{V}_n^H(\mathbf{P})\}_{n \in \mathcal{I}_\Omega}$ to replace partition $\mathcal{R}(\mathbf{P}) = \{\mathcal{R}_n(\mathbf{P})\}_{n \in \mathcal{I}_\Omega}$. Placing (5.3) back to (5.2), we can rewrite the distortion as

$$\mathcal{D}(\mathbf{P}) = \sum_{n \in \mathcal{S}(\mathbf{P})} \int_{\boldsymbol{V}_n^H(\mathcal{H}(\mathbf{P}))} \eta_n \|\mathbf{p}_n - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega}. \tag{5.4}$$

The same distortion can also be applied to formulate the communication energy consumption among densely deployed sensors where the $f(\cdot)$ presents the sensor density function [54]. Next, we review a classic energy consumption model for the mobile sensor networks. Since the sensor movement dominates the power consumption, we only consider the power consumption for sensor movement. As I mentioned in Section 1.1, the energy consumption for one-step movement is linearly related to the moving distance. Therefore, the energy consumption for Sensor n moving from a to b can be defined [21]–[23], [35], [57], [58] as

$$\mathscr{E}_n(a, b) = \xi_n \|b - a\|, \tag{5.5}$$

where the moving cost parameter ξ_n is a predetermined constant that depends on Sensor n's energy efficiency.

5.2 Centralized sensor deployment with a network lifetime constraint

In a centralized sensor deployment scenario, a fusion center or base station collects global information (all sensor locations and parameters) and then computes and determines the final destinations for the sensors. After receiving the decisions from the fusion center, sensors move to their final destinations directly. It is self-evident that this point-to-point relocation is the most efficient strategy in terms of energy consumption.

5.2.1 Problem formulation

Since sensors move to their final destinations directly, the energy consumption for Sensor n is formulated as

$$E_n(\mathbf{P}) = \mathscr{E}_n(\mathbf{p}_n^0, \mathbf{p}_n) = \xi_n \|\mathbf{p}_n - \mathbf{p}_n^0\|, \tag{5.6}$$

where $\mathbf{p}_n$ is Sensor n's final destination. Our main goal is minimizing the sensing uncertainty defined by (5.4) given a constraint on the network lifetime T. To guarantee a required network lifetime, each sensor should be assigned an energy threshold (or maximum movement distance) for relocation [28]–[31]. Therefore, the corresponding constrained optimization problem, which is referred to as Problem $\mathcal{A}$, is

$$\underset{\mathbf{P}}{\text{minimize}} \qquad \mathcal{D}(\mathbf{P}) \tag{5.7}$$

$$\text{s.t.} \quad E_n(\mathbf{P}) \leq \gamma_n, n \in I_\Omega, \tag{5.8}$$

where γ_n is the maximum energy consumption of Sensor n. Let e_n be the battery energy of Sensor n at the initial time and β (watt) be Sensor n's power consumption (which is dominated by communication, sensing, and computation) after the relocation. To ensure the network lifetime, T, we have $\min_n (e_n - E_n(\mathbf{P})) \geq \beta T$, and thus $\gamma_n = e_n - \beta T, n \in I_\Omega$.

5.2.2 The Optimal Sensor Deployment

Lemma 10. *Given a fully connected initial deployment, i.e.,* $\mathbf{P}^0 = \mathcal{H}(\mathbf{P}^0)$*, the optimal deployment* $\mathbf{P}^*$ *for Problem* $\mathcal{A}$ *in a homogeneous MWSN is also fully connected, i.e.,* $\mathbf{P}^* = \mathcal{H}(\mathbf{P}^*)$.

The proof is provided in Appendix A.11.

According to Lemma 10, homogeneous networks should keep connectivity after optimal sensor movements as long as the initial deployment is fully connected. To analyze the network connectivity, I introduce two important concepts: desired region (DR) and feasible region (FR). Let $\mathcal{I} \subseteq \mathcal{I}_\Omega$ be an arbitrary sensor set. For convenience, the sensors in $\mathcal{I}$ and $\mathcal{I}_\Omega - \mathcal{I}$ are referred to as internal and external sensors, respectively. For each sensor, n, the set of all locations of n that result in a connected $\mathcal{I}$ is called the DR of n. An internal sensor's DR for sensor set $\mathcal{I}$ is defined as the region in which if the sensor is placed, the sensors in $\mathcal{I}$ are connected. As a special case, if an internal sensor's DR is empty, the sensors in $\mathcal{I}$ cannot construct a connected network. Without the internal sensor, $n \in \mathcal{I}$, the rest of the internal sensors,

$\mathcal{I}-\{n\}$, consists of $K_n(\mathbf{P},\mathcal{I})$ disjoint components: $U_{n1}(\mathbf{P},\mathcal{I}), U_{n2}(\mathbf{P},\mathcal{I}), \cdots, U_{nK_n(\mathbf{P},\mathcal{I})}(\mathbf{P},\mathcal{I})$, where the sensors in each component are connected and $\bigcup_{k=1}^{K_n(\mathbf{P},\mathcal{I})} U_{nk}(\mathbf{P},\mathcal{I}) = \mathcal{I} - \{n\}$. The internal sensors are connected if and only if Sensor n connects to all $\{U_{nk}(\mathbf{P},\mathcal{I})\}$s. Thus, internal sensors' DRs for set $\mathcal{I}$ are formulated as

$$\mathbb{D}_n(\mathbf{P},\mathcal{I}) = \bigcap_{k=1}^{K_n(\mathbf{P},\mathcal{I})} \left[\bigcup_{j\in U_{nk}(\mathbf{P},\mathcal{I})} \mathbb{B}\left(\mathbf{p}_j, R_c\right) \right], \forall n \in \mathcal{I}. \tag{5.9}$$

Although we represent DRs as functions of $\mathbf{P}$ for convenience, Sensor n's DR is in fact determined by all sensors except itself. For an internal sensor $n \in \mathcal{I}$, the condition $\mathbf{p}_n \in \mathbb{D}_n(\mathbf{P},\mathcal{I})$[1] guarantees that all internal sensors can communicate with each other. In particular, if the AP is included in $\mathcal{I}$, we have $\mathcal{I} \subseteq \mathcal{S}(\mathbf{P})$. In addition, it is trivial to show that for two sensors $m, n \in \mathcal{I}$, $\mathbf{p}_m \in \mathbb{D}_m(\mathbf{P},\mathcal{I})$ is equivalent to $\mathbf{p}_n \in \mathbb{D}_n(\mathbf{P},\mathcal{I})$.

An example for 12 sensors with $R_c = 1$ is illustrated in Fig. 5.1a. The internal sensor set $\mathcal{I}$ is defined as all sensors, i.e., $\mathcal{I} = \mathcal{I}_\Omega = \{1,\ldots,12\}$. Consider $n = 1$, to calculate $\mathbb{D}_1(\mathbf{P},\mathcal{I}_\Omega)$, the rest of the sensors are divided into $K_1 = 2$ components $U_{11} = \{2,3,4,5,6,7\}$ and $U_{12} = \{8,9,10,11,12\}$. According to the definition of DR, the green overlap between the cyan region $\left[\bigcup_{j=2}^{7} \mathbb{B}\left(\mathbf{p}_j, R_c\right)\right]$ and the yellow region $\left[\bigcup_{j=8}^{12} \mathbb{B}\left(\mathbf{p}_j, R_c\right)\right]$ in Fig. 5.1a constructs Sensor 1's DR, $\mathbb{D}_1(\mathbf{P},\mathcal{I}_\Omega)$. Obviously, if $\mathbf{p}_1$ is placed within $\mathbb{D}_1(\mathbf{P},\mathcal{I}_\Omega)$, all 12 sensors can communicate with each other. However, if the internal sensor set $\mathcal{I}$ is defined as $\{1,4,5,6,9,10,11\}$, the corresponding DR for Sensor 1 will be empty, indicating that the sensors $\{1,4,5,6,9,10,11\}$ cannot construct a connected network.

Next, I define $\mathcal{W}(\mathbf{P},\mathcal{I}) \triangleq \bigcup_{n\in\mathcal{I}_\Omega-\mathcal{I}} \mathbb{B}\left(\mathbf{p}_n, R_c\right)$. It is self-evident that the internal sensors placed in $\mathcal{W}(\mathbf{P},\mathcal{I})$ connect to at least one external sensor. As a result, the sensor set $\mathcal{I}$ is the exact backbone network if and only if $1 \in \mathcal{I}$ and $\mathbf{p}_n \in \mathbb{D}_n(\mathbf{P},\mathcal{I}) \bigcap \mathcal{W}^c(\mathbf{P},\mathcal{I})$, where $\mathcal{W}^c(\mathbf{P},\mathcal{I}) = \Omega - \mathcal{W}(\mathbf{P},\mathcal{I})$ is the complement of $\mathcal{W}(\mathbf{P},\mathcal{I})$. In what follows, I take the energy constraints (5.8) into account, and propose the concept of FR defined by

[1]Remark: When Sensor n is placed in its DR for $\mathcal{I}$, some external sensors, $m \in \mathcal{I}_\Omega - \mathcal{I}$, may also connect to internal sensors.

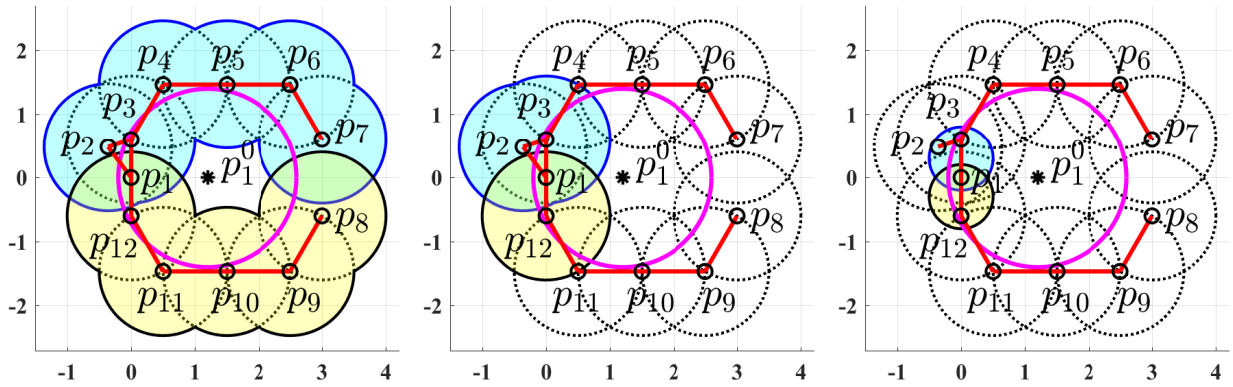


Figure 5.1: Example1: (a) DR and FR for Sensor 1; (b) ADR and AFR for Sensor 1; (c) SDR and SFR for Sensor 1; DR, ADR, and SDR are shown by green. FR, AFR, and SFR are shown by the intersections of green regions and the magenta circles. Communication ranges, movement range, and Connections are, respectively, denoted by black doted curves, magenta solid curve, and red lines.

$$\mathbb{F}_n(\mathbf{P},\mathcal{I}) \triangleq \mathbb{D}_n(\mathbf{P},\mathcal{I}) \bigcap \mathbb{B}\left(\mathbf{p}_n^0, \frac{\gamma_n}{\xi_n}\right), n \in \mathcal{I}, \tag{5.10}$$

where the energy constraint, $\xi_n \|\mathbf{p}_n - \mathbf{p}_n^0\| \leq \gamma_n$, is satisfied by the condition $\mathbf{p}_n \in \mathbb{B}\left(\mathbf{p}_n^0, \frac{\gamma_n}{\xi_n}\right)$. The example of FR for Sensor 1 is illustrated in Fig. 5.1a. The magenta circle demonstrates Sensor 1's movement range $\mathbb{B}\left(\mathbf{p}_1^0, \frac{\gamma_n}{\xi_n}\right)$. Then, the intersection of green regions and the magenta circle in Fig. 5.1a is Sensor 1's FR, $\mathbb{F}_1(\mathbf{P},\mathcal{I}_\Omega)$. Obviously, if $\mathbf{p}_1$ is placed within $\mathbb{F}_1(\mathbf{P},\mathcal{I}_\Omega)$, we have (a) all 12 sensors can communicate with each other and (b) Sensor 1's energy constraint is also satisfied.

Note that $\mathbf{p}_n \in \mathbb{F}_n(\mathbf{P},\mathcal{I})$ implicitly implies that $\mathbb{F}_n(\mathbf{P},\mathcal{I}) \neq \emptyset$. Accordingly, the set of deployments that not only constructs the backbone network $\mathcal{S}(\mathbf{P}) = \mathcal{I}$ but also satisfies the energy constraints can be formulated as

$$\Gamma(\mathcal{I}) = \{\mathbf{P} | 1 \in \mathcal{I}, \mathbf{p}_n \in \mathbb{F}_n(\mathbf{P},\mathcal{I}) \bigcap \mathcal{W}^c(\mathbf{P},\mathcal{I}), \forall n \in \mathcal{I}\}. \tag{5.11}$$

Based on the aforementioned concepts, I propose the following necessary condition for the optimal deployment.

Theorem 5.1. *Let* $\mathbf{P}^*=(\mathbf{p}_1^*,\ldots,\mathbf{p}_N^*)$ *be the optimal deployment for Problem* $\mathcal{A}$*. The necessary conditions for the optimal deployment are*

(i) $\mathbf{c}_n(\mathbf{P}^*) \notin \left[\mathbb{F}_n(\mathbf{P}^*,\mathcal{S}(\mathbf{P}^*)) \bigcap \mathcal{W}(\mathbf{P}^*,\mathcal{S}(\mathbf{P}^*))\right], \forall n \in \mathcal{S}(\mathbf{P}^*)$

$$\textit{(ii)}\mathbf{p}_n^* = \begin{cases} \mathbf{c}_n(\mathbf{P}^*), & \textit{if } \mathbf{c}_n(\mathbf{P}^*) \in \mathbb{F}_n(\mathbf{P}^*,\mathcal{S}(\mathbf{P}^*)) \bigcap \mathcal{W}^c(\mathbf{P}^*,\mathcal{S}(\mathbf{P}^*)) \\ \arg\min\limits_{q\in\partial\mathbb{F}_n(\mathbf{P}^*,\mathcal{S}(\mathbf{P}^*))} \|q-\mathbf{c}_n(\mathbf{P}^*)\|, & \textit{if } \mathbf{c}_n(\mathbf{P}^*) \in \Omega - \mathbb{F}_n(\mathbf{P}^*,\mathcal{S}(\mathbf{P}^*)) \end{cases}, \forall n \in \mathcal{S}(\mathbf{P}^*)$$

The proof is provided in Appendix A.12.

According to Theorem 5.1, if n is a sensor in the backbone network $\mathcal{S}(\mathbf{P}^*)$, its optimal location, $\mathbf{p}_n^*$, is either at the centroid $\mathbf{c}_n(\mathbf{P}^*)$ or on the boundary of $\mathbb{F}_n(\mathbf{P}^*,\mathcal{S}(\mathbf{P}^*))$. Note that Theorem 5.1 only provides the necessary conditions for the sensors in the backbone network $\mathcal{S}(\mathbf{P}^*)$ because sensors that are not in the backbone network make no contribution to the distortion (5.4). In particular, by Lemma 10, all sensors in homogeneous MWSNs should be included in the backbone network, i.e., $\mathcal{S}(\mathbf{P}^*) = \mathcal{I}_\Omega$, and therefore the necessary conditions in Theorem 5.1 can be extended to all sensors in homogeneous MWSNs. Since there are no inactive sensors in homogeneous MWSNs, we have $\mathcal{W}(\mathbf{P}^*,\mathcal{S}(\mathbf{P}^*)) = \emptyset$. Then, the necessary conditions for the optimal deployment in homogeneous MWSNs can be refined as

$$\mathbf{p}_n^* = \begin{cases} \mathbf{c}_n(\mathbf{P}^*), & \text{if } \mathbf{c}_n(\mathbf{P}^*) \in \mathbb{F}_n(\mathbf{P}^*,\mathcal{I}_\Omega) \\ \arg\min\limits_{q\in\partial\mathbb{F}_n(\mathbf{P}^*,\mathcal{I}_\Omega)} \|q-\mathbf{c}_n(\mathbf{P}^*)\|, & \text{if } \mathbf{c}_n(\mathbf{P}^*) \notin \mathbb{F}_n(\mathbf{P}^*,\mathcal{I}_\Omega) \end{cases}, \forall n \in \mathcal{I}_\Omega \tag{5.12}$$

With the help of the necessary conditions in Theorem 5.1, I design centralized Lloyd-like algorithms to find the optimal sensor deployment with a network lifetime constraint in the next subsection.

5.2.3 Centralized Lloyd-like Algorithms

To optimize the sensor deployments in homogeneous and heterogeneous MWSNs, I propose two centralized Lloyd-like algorithms: Centralized Constrained Movement Lloyd (CCML)

Algorithm and Backward-stepwise Centralized Constrained Movement Lloyd (BCCML) Algorithm. CCML Algorithm, which is designed for homogeneous MWSNs, keeps all sensors in the backbone network. Based on CCML Algorithm, BCCML Algorithm recursively selects the optimal sensor set to construct the backbone network for heterogeneous MWSNs.

CCML Algorithm

According to our analysis in Section 5.2.2, Sensor n's movement should be restrained within its desired region, $\mathbb{D}_n(\mathbf{P}, \mathcal{I}_\Omega)$, in order to keep full-connectivity, i.e., $\mathcal{I} = \mathcal{I}_\Omega$. Since the desired region is primarily influenced by the neighboring sensor nodes, we can approximate it by

$$\tilde{\mathbb{D}}_n(\mathbf{P}, \mathcal{I}_\Omega) = \bigcap_{k=1}^{K_n(\mathbf{P},\mathcal{I})} \left[\bigcup_{j \in U_{nk}(\mathbf{P},\mathcal{I}_\Omega) \bigcap \mathcal{N}_n(\mathbf{P})} \mathbb{B}\left(\mathbf{p}_j, R_c\right) \right], \tag{5.13}$$

where $\mathcal{N}_n(\mathbf{P})$ is the set of Sensor n's neighbors. Then, FR is approximated by

$$\tilde{\mathbb{F}}_n(\mathbf{P}, \mathcal{I}_\Omega) = \tilde{\mathbb{D}}_n(\mathbf{P}, \mathcal{I}_\Omega) \bigcap \mathbb{B}\left(\mathbf{p}_n^0, \frac{\gamma_n}{\xi_n}\right) \tag{5.14}$$

Note that the approximation in (5.14) can be calculated locally, but to calculate the exact feasible region, one needs global information. The above two approximations are referred to as approximated desired region (ADR) and approximated feasible region (AFR). The examples of ADR, and AFR for Sensor 1 are illustrated in Fig. 5.1b. Different from the DR shown in Fig. 5.1a, the ADR only considers Sensor 1's neighbors, $\mathcal{N}_1 = \{2, 3, 12\}$. Thus, the green overlap between cyan region $\left[\bigcup_{j=2}^{3} \mathbb{B}\left(\mathbf{p}_j, R_c\right)\right]$ and yellow region $\mathbb{B}\left(\mathbf{p}_{12}, R_c\right)$ in Fig. 5.1b construct Sensor 1's ADR, $\tilde{\mathbb{D}}_1\left(\mathbf{P}, \mathcal{I}_\Omega\right)$. Then, the intersection of the green region and the magenta circle in Fig. 5.1b is Sensor 1's AFR, $\tilde{\mathbb{F}}_1\left(\mathbf{P}, \mathcal{I}_\Omega\right)$. Note that Sensor 1's FR in Fig. 5.1a consists of two disconnected regions while Sensor 1's AFR in Fig. 5.1b is a connected region.

Now, I provide the details of CCML Algorithm. Like Lloyd Algorithm, the proposed algorithm iterates between two steps: (1) Partition optimization: Partitioning is done by MWVDs; (2) Location optimization: each sensor moves to the closest point to its centroid $\mathbf{c}_n(\mathbf{P})$ within $\tilde{\mathbb{F}}_n(\mathbf{P}, \mathcal{I}_\Omega)$. More details about CCML Algorithm are shown in Algorithm 3.

Algorithm 3 Centralized Constrained Movement Lloyd Algorithm

Input: Target area Ω; Probability density function $f(\cdot)$; the initial sensor deployment $\mathbf{P}^0$; the required network lifetime T; the stop threshold ϵ; the communication range R_c.
Output: Sensors deployment $\mathbf{P}$; Distortion $\mathcal{D}(\mathbf{P})$.

```
Calculate the energy constraints {γ_n}_{n∈I_Ω} in terms of T
Initialize sensor deployment P = P^0
do
    Calculate the old distortion D_old = D(P)
    Do multiplicatively weighted Voronoi partition
    for n = 1 to N do
        Calculate the feasible region {F_n(P, I_Ω)}
        Calculate the critical point q, closest point to c_n(P) within {F_n(P, I_Ω)}
        Update sensor deployment p_n = q
        Calculate the new distortion D_new = D(P)
    end for
while (D_old − D_new)/D_old > ε
```

Theorem 5.2. *CCML Algorithm is an iterative improvement algorithm, i.e., the distortion decreases at each iteration and converges.*

Proof. CCML Algorithm is an iterative improvement algorithm only if both steps in CCML Algorithm do not increase the distortion (5.4) subject to the constraints (5.8). In Section 5.1, I have proved that MWVD is the optimal cell partition for a given deployment. Therefore, Step (1) of CCML will not increase the distortion. During Step (2) of CCML, the cell partition is fixed as MWVD. In Appendix A, I show that Sensor n's optimal location should minimize its distance to the centroid $\mathbf{c}_n(\mathbf{P})$ when the cell partition is fixed. In addition, by the analysis in Section 5.2.2, Sensor n's movement should be restricted in $\tilde{\mathbb{F}}_n(\mathbf{P}, \mathcal{I}_\Omega)$ in order to guarantee both (i) energy constraints (5.8) and (ii) full-connectivity which has been proved (in Lemma 10) as a necessary condition for the optimum solution. Accordingly, Step (2) of CCML will not increase the distortion. Therefore, CCML Algorithm is an iterative

improvement algorithm. Furthermoer, the distortion has a lower bound 0. As a result, the distortion of CCML Algorithm is non-increasing with a lower bound, indicating that the distortion converges. □

BCCML Algorithm

Sensors with low-battery energy have small energy to spend on motion, which results in small movement ranges. To keep the connection with a low-battery node, e.g., Sensor n, the neighboring sensors' movements will be restricted by the limited communication range of Sensor n, even if Sensor n's neighbors have access to large battery energy. In this case, if Sensor n is not used in the MWSN, the neighboring sensors will have more freedom to move and probably further decrease the overall distortion. Given the current sensor set $\mathcal{I}$, sensors that decrease the distortion when removed from $\mathcal{I}$ are referred to as bottleneck sensors. To select the optimal sensor set as our backbone network, BCCML Algorithm starts with all sensors and repeatedly eliminates the least significant bottleneck sensor in terms of reducing the distortion until no bottleneck sensor is left. Intuitively, a bottleneck sensor n should satisfy the following conditions: (i) After eliminating n, the rest of sensors in $\mathcal{I}$ should be connected, i.e., $K_n(\mathbf{P}, \mathcal{I}) = 1$ and $U_{n1}(\mathbf{P}, \mathcal{I}) = \mathcal{I} - \{n\}$. In other words, Sensor n is a leaf node in the network. Otherwise, the network will be divided into multiple sub-graphs after eliminating n, and then fewer sensors will be used in the sensing task. (ii) Sensor n should already run out of its movement energy, i.e., $\xi_n \|\mathbf{p}_n - \mathbf{p}_n^0\| = \gamma_n$. (iii) At least one of its neighbors has redundant energy, i.e., $\exists m \in \mathcal{N}_n, \xi_m \|\mathbf{p}_m - \mathbf{p}_m^0\| < \gamma_m$. The above three conditions are referred to as the bottleneck criterion. To speed up the computation, BCCML Algorithm merely eliminates bottleneck sensors satisfying the bottleneck criterion. The details of BCCML Algorithm are shown in Algorithm 4.

Algorithm 4 Backwards-stepwise Centralized Constrained Movement Lloyd Algorithm

Input: Target area Ω; probability density function $f(\cdot)$; the initial sensor deployment $\mathbf{P}^0$; the required network lifetime T; the stop threshold ϵ; the communication range R_c.
Output: Sensors deployment $\mathbf{P}$; Distortion $\mathcal{D}(\mathbf{P})$.

Calculate the energy constraints $\{\gamma_n\}_{n\in\mathcal{I}_\Omega}$ in terms of T
Initialize sensor set $\mathcal{I} = \mathcal{I}_\Omega$
Run CCML: $[\mathbf{P}, \mathcal{D}(\mathbf{P})] = \text{CCML}(\Omega, \mathbf{P}^0, R_c, \{\gamma_n\}_{n\in\mathcal{I}_\Omega}, f(\cdot), \epsilon, \mathcal{I})$
for $k = 1$ to $N - 1$ **do**
    Identify bottleneck sensor set $\mathcal{I}_b$ by checking the bottleneck criterion
    $LSS = null$
    for $i \in \mathcal{I}_b$ **do**
        Generate a temporary sensor set $\hat{\mathcal{I}} = \mathcal{I} - i$
        Run CCML: $[\hat{\mathbf{P}}, \mathcal{D}(\hat{\mathbf{P}})] = \text{CCML}(\Omega, \mathbf{P}^0, R_c, \{\gamma_n\}_{n\in\mathcal{I}}, f(\cdot), \epsilon, \hat{\mathcal{I}})$
        if $\mathcal{D}(\hat{\mathbf{P}}) < \mathcal{D}(\mathbf{P})$ **then**
            Update the least significant sensor (LSS): $LSS = i$, $\mathbf{P} = \hat{\mathbf{P}}$
        end if
    end for
    if $LSS \neq null$ **then**
        Eliminate the least significant sensor: $\mathcal{I} = \mathcal{I} - LSS$
    else
        break
    end if
end for

5.3 Distributed sensor deployment with a network lifetime constraint

5.3.1 Problem formulation

In the distributed scenario, there is no fusion center, and sensors determine their own destinations. In general, sensors are supposed to only collect neighboring information (the locations and parameters of its neighbors and itself). As I discussed in Section 5.2, the most energy-efficient relocation strategy is moving a sensor from its initial location to the final destination in one step. Unfortunately, this one-step relocation strategy requires global information and cannot be implemented in a distributed scenario. In fact, the distributed sensor relocation methods in the literature are categorized into continuous and discrete time

systems [3]–[10], [13], [15]–[17], [32]. In the continuous time systems[2], sensors keep communicating with their neighbors during continuous movements. Then, dynamic systems are widely used to control sensors's first order dynamics, velocity, and/or second order dynamics, acceleration [7], [8], [13]. However, in the discrete time system, sensors only communicate with their neighbors at some discrete time instances, and their relocation is divided into multiple steps [10]. Regarding the discrete nature of the relocation, the sensors should be synchronized with each other in some fashion. Some iterative algorithms, such as Lloyd-like algorithms and virtual-force based algorithms, have been applied to this scenario [9], [10], [15]–[17], [32], [34]. To reduce the communication costs during the relocation, I use a discrete time system to control sensors' movements.

In what follows, I concentrate on the sensor relocation with multiple stops. The sensor deployment at the k-th stop is defined by $\mathbf{P}^k = (\mathbf{p}_1^k, \ldots, \mathbf{p}_N^k) \subset \Omega^N$, where $\mathbf{p}_n^k$ is Sensor n's location at the k-th stop. Let K be the maximum number of stops (iterations) for each sensor. For convenience, each sensor is extended to have K stops. For a sensor with J physical stops, e.g. Sensor n, its redundant stops are extended as $\mathbf{p}_n^k = \mathbf{p}_n^J, \forall k \in \{J+1, \ldots, K\}$. In particular, $\mathbf{P}^0 = (\mathbf{p}_1^0, \ldots, \mathbf{p}_N^0) \subset \Omega^N$ and $\mathbf{P}^K = (\mathbf{p}_1^K, \ldots, \mathbf{p}_N^K) \subset \Omega^N$ are the initial and final deployments, respectively. Sensor n's total movement distance is $\sum_{k=1}^{K} \|\mathbf{p}_n^k - \mathbf{p}_n^{k-1}\|$, and therefore Sensor n's individual energy consumption is formulated as

$$\sum_{k=1}^{K} \mathscr{E}(\mathbf{p}_n^{k-1}, \mathbf{p}^k) = \xi_n \sum_{k=1}^{K} \|\mathbf{p}_n^k - \mathbf{p}_n^{k-1}\|. \tag{5.15}$$

Now, we discuss the distributed realization for the node deployment with (i) network lifetime constraint and (ii) limited communication range. According to the analysis in [10], the movement distance should be constrained at each iteration in order to avoid zigzag movements.

[2]Remark: The energy formulation (5.5) works for a one-step movement where the optimal velocity and acceleration is determined by the distance [57]. However, the movement in continuous time systems is not step-wise and the corresponding motion energy is a function of velocity [13]. Thus, the energy model in this paper cannot be applied to continuous time systems. The continuous sensor relocation in MWSNs is an interesting future work.

Therefore, we limit Sensor n's movement distance in the k-th iteration by an upper bound d_n^k. Note that d_n^k can be a constant or a function of the previous and current deployments. For instance, $d_n^k = \alpha\|\mathbf{p}^{k-1} - \mathbf{c}^{k-1}\|$ in Lloyd-α [10], where $\alpha \in (0, 1]$. Moreover, to guarantee the required network lifetime, another constraint $\xi_n \sum_{i=1}^{k} \|\mathbf{p}_n^i - \mathbf{p}_n^{i-1}\| \leq \gamma_n$ should be taken into account. Furthermore, full-connectivity, which is ignored in most distributed sensor relocation algorithms, is definitely required to obtain neighboring information. Then, another constraint $\mathcal{H}\left(\mathbf{P}^k\right) = \mathbf{P}^k$ should also be considered. With the above constraints, each sensor in the distributed scenario optimizes its next stop, $\mathbf{P}^k$, in terms of the previous and current neighboring information, $\mathbf{P}^i, \forall i < k$. In particular, the cell partition in Lloyd-like algorithms is generated by the current deployment [9], [10], [15]–[17], [32], [34], $\mathcal{R}_n = \boldsymbol{V}_n^H(\mathbf{P}^{k-1})$. The corresponding optimization problem, which is referred to as Problem $\mathcal{B}$, is thus represented as

$$\underset{\mathbf{P}^k}{\text{minimize}} \qquad \sum_{n=1}^{N} \int_{\boldsymbol{V}_n^H(\mathbf{P}^{k-1})} \eta_n \|\mathbf{p}_n^k - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} \tag{5.16}$$

$$\text{s.t.} \qquad \mathcal{H}\left(\mathbf{P}^k\right) = \mathbf{P}^k \tag{5.17}$$

$$\|\mathbf{p}_n^k - \mathbf{p}_n^{k-1}\| \leq \min\left(\frac{\tilde{e}_n^k}{\xi_n}, d_n^k\right), n \in I_\Omega \tag{5.18}$$

where $\tilde{e}_n^k = \gamma_n - \xi_n \sum_{i=1}^{k-1} \|\mathbf{p}_n^i - \mathbf{p}_n^{i-1}\|$ is the residual energy at the k-th iteration. Since full-connectivity is guaranteed by constraint (5.17), all sensors contribute to the distortion (5.16).

5.3.2 Semi-desired Region and Semi-feasible Region

Before studying the optimal solution for (5.16), we analyze the constraints (5.17) and (5.18). In the distributed scenario, sensors are supposed to relocate simultaneously. However, the full-connectivity strategy used in CCML and BCCML requires a one-by-one relocation scheme, which is not possible in large-scale distributed networks. To follow full-connectivity constraint (5.17) in a large-scale distributed network, I introduce another important concept, semi-desired region (SDR), which is a shrunk version of the approximated desired region

$\tilde{\mathbb{D}}_n(\mathbf{P}, \mathcal{I}_\Omega)$. Let $\mathcal{G}(\mathbf{P}) = (\mathcal{V}(\mathbf{P}), \mathcal{E}(\mathbf{P}))$ be the undirected connectivity graph comprising a set of vertices $\mathcal{V} = \{\mathbf{p}_1, \ldots, \mathbf{p}_N\}$ and a set of edges $\mathcal{E} = \{e_{ij}\}$. The edge cost, $w_{ij} = \|\mathbf{p}_i - \mathbf{p}_j\|$, is defined as the Euclidean distance between the end vertices of the edge, e_{ij}, and there exists an edge between $\mathbf{p}_i$ and $\mathbf{p}_j$ in $\mathcal{G}(\mathbf{P})$ if and only if $w_{ij} \leq R_c$. For a fully connected graph, $\mathcal{G}(\mathbf{P})$, the corresponding minimum spanning tree (MST) is defined as $\tilde{\mathcal{G}}(\mathbf{P}) = \left(\mathcal{V}(\mathbf{P}), \tilde{\mathcal{E}}(\mathbf{P})\right)$, where $\tilde{\mathcal{E}}(\mathbf{P}) \subset \mathcal{E}(\mathbf{P})$ is a subset of size $|\mathcal{V}(\mathbf{P})| - 1$. Let $\mathcal{N}_n^s(\mathbf{P}) = \{m | \tilde{e}_{nm} \in \tilde{\mathcal{E}}(\mathbf{P}) \text{ or } \tilde{e}_{mn} \in \tilde{\mathcal{E}}(\mathbf{P})\}$ be the set of Sensor n's neighbors in MST. Then, the SDRs are defined as

$$\mathbb{D}_n^s(\mathbf{P}) = \bigcap_{m \in \mathcal{N}_n^s(\mathbf{P})} \mathbb{B}\left(\frac{\mathbf{p}_m + \mathbf{p}_n}{2}, \frac{R_c}{2}\right), \forall n \in \mathcal{I}_\Omega. \tag{5.19}$$

An example of SDR is illustrated in Fig. 5.1c. In this example, 12 sensors with communication range $R_c = 1$ are deployed on the plane, indicating that $\mathcal{I}_\Omega = \{1, \ldots, 12\}$. The edges in MST are denoted by red lines in Fig. 5.1c. Also, Sensor 1's movement range, $\mathbb{B}(\mathbf{p}_1^0, d_n)$, is demonstrated by a magenta circle. According to the definition of semi-desired region, the green overlap between cyan region $\mathbb{B}\left(\frac{\mathbf{p}_1+\mathbf{p}_2}{2}, \frac{R_c}{2}\right)$ and yellow region $\mathbb{B}\left(\frac{\mathbf{p}_1+\mathbf{p}_{12}}{2}, \frac{R_c}{2}\right)$ in Fig. 5.1c constructs Sensor 1's semi-desired region, $\mathbb{D}_1^s(\mathbf{P})$. From Figs. 5.1a, 5.1b, and 5.1c, it is also clear that the semi-desired region is a subset of the approximated desired region and the desired region, i.e., $\mathbb{D}_1^s(\mathbf{P}) \subseteq \tilde{\mathbb{D}}_1(\mathbf{P}, \mathcal{I}_\Omega) \subseteq \mathbb{D}_1(\mathbf{P}, \mathcal{I}_\Omega)$.

Theorem 5.3. *Starting with a fully connected network ($\mathcal{S}(\mathbf{P}^k) = \mathcal{I}_\Omega$), the network is still fully connected ($\mathcal{S}(\mathbf{P}^{k+1}) = \mathcal{I}_\Omega$) if sensors simultaneously move within their respective semi-desired regions i.e., $\mathbf{p}_n^{k+1} \in \mathbb{D}_n^s\left(\mathbf{P}^k\right), \forall n \in \mathcal{I}_\Omega$.*
The proof is provided in Appendix A.13.
Then, I define the semi-feasible regions as

$$\mathbb{F}_n^s(\mathbf{P}, \tilde{e}_n^k, d_n^k) = \mathbb{D}_n^s(\mathbf{P}) \bigcap \mathbb{B}\left(\mathbf{p}_n^0, \min\left(\frac{\tilde{e}_n^k}{\xi_n}, d_n^k\right)\right), \forall n \in \mathcal{I}_\Omega, \tag{5.20}$$

It is trivial to show that both (5.17) and (5.18) are satisfied if sensors move within their semi-feasible regions. Let $\mathcal{P}^k = \{\mathbf{P}^k | \mathcal{H}\left(\mathbf{P}^k\right) = \mathbf{P}^k, \|\mathbf{p}_n^k - \mathbf{p}_n^{k-1}\| \leq \min\left(\frac{\tilde{e}_n^k}{\xi_n}, d_n^k\right), \forall n \in \mathcal{I}_\Omega\}$

be the set of deployments that follow constraints (5.17) and (5.18), and $\widehat{\mathcal{P}}^k = \{\mathbf{P}^k | \mathbf{p}_n^k \in \mathbb{F}_n^s(\mathbf{P}, \tilde{e}_n^k, d_n^k), \forall n \in \mathcal{I}_\Omega\}$ be the set of deployments that are placed within semi-feasible regions. Then, we have $\widehat{\mathcal{P}}^k \subseteq \mathcal{P}^k$. To simplify the problem, we replace $\mathcal{P}^k$ by $\widehat{\mathcal{P}}^k$, and the optimization problem is represented as N independent problems:

$$\underset{\mathbf{P}^k}{\text{minimize}} \qquad \int_{\boldsymbol{V}_n^H(\mathbf{P}^{k-1})} \eta_n \|\mathbf{p}_n^k - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} \tag{5.21}$$

$$\text{s.t.} \qquad \mathbf{p}_n^k \in \mathbb{F}_n^s(\mathbf{P}, \tilde{e}_n^k, d_n^k) \tag{5.22}$$

where $n \in \mathcal{I}_\Omega$. By parallel axis theorem[121], (5.21) can be rewritten as

$$\int_{\boldsymbol{V}_n^H(\mathbf{P}^{k-1})} \eta_n \|\mathbf{c}_n^{k-1} - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} + \eta_n \|\mathbf{p}_n^k - \mathbf{c}_n^{k-1}\|^2 \mathbf{v}_n^{k-1}, \tag{5.23}$$

where $\mathbf{v}_n^{k-1} = \int_{\boldsymbol{V}_n^H(\mathbf{P}^{k-1})} f(\boldsymbol{\omega}) d\boldsymbol{\omega}$ and $\mathbf{c}_n^{k-1} = \frac{\int_{\boldsymbol{V}_n^H(\mathbf{P}^{k-1})} \boldsymbol{\omega} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}{\int_{\boldsymbol{V}_n^H(\mathbf{P}^{k-1})} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}$. Since the first term in (5.23) is a constant, Sensor n's distortion is an increasing function of the distance from $\mathbf{p}_n^k$ to $\mathbf{c}_n^{k-1}$. Accordingly, the optimal solution for (5.21) with constraint (5.22) is the point closet to $\mathbf{c}_n^{k-1}$ within $\mathbb{F}_n^s(\mathbf{P}, \tilde{e}_n^k, d_n^k)$, i.e., $\mathbf{p}_n^k = \arg\min_{q \in \mathbb{F}_n^s(\mathbf{P}, \tilde{e}_n^k, d_n^k)} \|q - \mathbf{c}_n^{k-1}\|$. By moving sensors to $\arg\min_{q \in \mathbb{F}_n^s(\mathbf{P}, \tilde{e}_n^k, d_n^k)} \|q - \mathbf{c}_n^{k-1}\|$ at each iteration, we will have a distributed realization, Distributed Constrained Movement Lloyd (DCML) Algorithm. According to the above analysis, DCML Algorithm will result in a deployment that guarantees both connectivity and the required network lifetime. Like CCML Algorithm, DCML Algorithm is an iterative improvement algorithm in which the distortion is non-increasing and converges. The proof is similar to that of Theorem 5.2 and therefore omitted here. More details about DCML Algorithm are shown in Algorithm 5.

Algorithm 5 Distributed Constrained-Movement Lloyd Algorithm

Input: Target area Ω; probability density function $f(\cdot)$; the initial sensor deployment $\mathbf{P}^0$; the required network lifetime T; The number of stops K; the communication range R_c.

Output: Sensors deployments at K stops $\{\mathbf{P}^k\}_{k\in\{1,\dots,K\}}$; Distortion at the final deployment $\mathcal{D}(\mathbf{P}^K)$.

1: Calculate energy constraints γ_n and initialize the residual energy $\tilde{e}_n^1 = \gamma_n, \forall n \in \mathcal{I}_\Omega$
2: **for** $k = 1$ to K **do**
3: Determine the maximum movement distances $d_n^k = \min\left(\frac{\tilde{e}_n^k}{\xi_n}, \frac{2\tilde{e}_n^k}{K-k+1}\right), n \in \mathcal{I}_\Omega$
4: Sensors obtain their MST neighbors $\{\mathcal{N}_n^s\}_{n\in\mathcal{I}_\Omega}$ by GHS Algorithm [122]
5: Do multiplicatively weighted Voronoi partition, and then update centroid $\{\mathbf{c}_n(\mathbf{P}^{k-1})\}_{n\in\mathcal{I}_\Omega}$
6: Calculate semi-desired regions $\{\mathbb{D}_n^s(\mathbf{P}^{k-1}, d_n^k)\}_{n\in\mathcal{I}_\Omega}$
7: Calculate the critical point q_n^k, closest point to $\mathbf{c}_n(\mathbf{P}^{k-1})$ within $\mathbb{D}_n^s(\mathbf{P}^{k-1}, d_n^k)$, $\forall n \in \mathcal{I}_\Omega$
8: Each sensor moves to its next stop $\mathbf{p}_n^k = \mathbf{p}_n^{k-1} + \min\left(\frac{\gamma_n}{2K\xi_n}, \|q_n^k - \mathbf{p}_n^{k-1}\|\right)\frac{q_n^k - \mathbf{p}_n^{k-1}}{\|q_n^k - \mathbf{p}_n^{k-1}\|}$, $\forall n \in \mathcal{I}_\Omega$
9: Update residual energy $\tilde{e}_n^{k+1} = \tilde{e}_n^k - \xi_n\|\mathbf{p}_n^k - \mathbf{p}_n^{k-1}\|, \forall n \in \mathcal{I}_\Omega$
10: **end for**

5.4 Algorithm complexity and communication overhead

5.4.1 Algorithm Complexity

Before I calculate the complexity of different algorithms, I need to study the computational complexity of FR, AFR, SFR, $\mathbf{c}_n(\mathbf{P})$ and $\boldsymbol{V}_n^H(\mathbf{P})$. Finding Sensor n's FR, $\mathbb{F}_n(\mathbf{P}, \mathcal{I})$, includes the following four stages: (i) Determine $K_n(\mathbf{P}, \mathcal{I})$ disjoint components $\{U_{nk}(\mathbf{P}, \mathcal{I})\}_{k=1,\dots,K_n(\mathbf{P},\mathcal{I})}$ by Breadth First Search (BFS) or Depth First Search (DFS) with time complexity $O(card(\mathcal{I}))$[3]. (ii) Calculate the union of communication balls in each component, i.e., $\bigcup_{j\in U_{nk}(\mathbf{P},\mathcal{I})} \mathbb{B}(\mathbf{p}_j, R_c)$, $\forall k \in \{1, \dots, K_n(\mathbf{P}, \mathcal{I})\}$. Note that the complexity of calculating the union of two regions is a constant, $O(1)$. Thus, the complexity of Stage (ii) is $O(card(\mathcal{I}))$. (iii) Obtain $\mathbb{D}_n(\mathbf{P}, \mathcal{I})$ by calculating the intersection of $K_n(\mathbf{P}, \mathcal{I})$ regions obtained from the previous stage. Since the computational complexity of intersection is $O(1)$, the complexity in Stage (iii) is $O(K_n(\mathbf{P}, \mathcal{I}))$. (iv) Compute the intersection between $\mathbb{D}_n(\mathbf{P}, \mathcal{I})$ and $\mathbb{B}_n(\mathbf{p}_n^0, \frac{\gamma_n}{\xi_n})$ with complexity $O(1)$.

Therefore, the computational complexity of $\mathbb{F}_n(\mathbf{P}, \mathcal{I})$ is $O(card(\mathcal{I}) + K_n(\mathbf{P}, \mathcal{I}))$. In total, the

[3]The exact complexity of BFS and DFS is $O(N + E)$, where E is the number of connections in the communication graph. Since each sensor has very limited neighbours in the range of R_c, we have $O(N) = O(E)$ and thus $O(N + E) = O(N)$.

complexity of finding $card(\mathcal{I})$ sensors' FRs is $O(card(\mathcal{I})^2 + card(\mathcal{I}) \cdot K_n(\mathbf{P}, \mathcal{I}))$. In general, $K_n(\mathbf{P}, \mathcal{I}) \ll card(\mathcal{I})$, and then the corresponding complexity becomes $O(card(\mathcal{I})^2)$. The computations of AFR and SFR are similar to that of FR, and have the same complexity $O(card(\mathcal{I})^2)$.

Next, we study the computational complexity of $\mathbf{c}_n(\mathbf{P})$ and $\boldsymbol{V}_n^H(\mathbf{P})$. Let $\mu(R) = \int_R \boldsymbol{\omega} d\boldsymbol{\omega}$ be the volume of region R with the uniform distribution. Many integral algorithms, such as uniform sampling, stratified sampling, importance sampling, sequential Monte Carlo, and Risch algorithm are available in the literature [123], [124]. For simplicity, I assume the integrals in $\mathbf{c}_n(\mathbf{P})$ and $\boldsymbol{V}_n^H(\mathbf{P})$ are calculated by uniform sampling[4]. In this case, the computational complexity of $\mathbf{c}_n(\mathbf{P})$ and $\boldsymbol{V}_n^H(\mathbf{P})$ is proportional to the number of samples, $O(\frac{\mu(\boldsymbol{V}_n^H(\mathbf{P}))}{\epsilon})$, where ϵ is the sample size. Thus, the total complexity of computing all $\mathbf{c}_n(\mathbf{P})$s and $\boldsymbol{V}_n^H(\mathbf{P})$s is $O(\frac{\sum_{n=1}^{N} \mu(\boldsymbol{V}_n^H(\mathbf{P}))}{\epsilon}) = O(\frac{\mu(\Omega)}{\epsilon})$.

Now, we have enough materials to derive the complexity of different algorithms. Since CCML is deigned for homogeneous MWSN, the backbone network includes all sensors, i.e., $card(\mathcal{I}) = N$. Therefore, CCML's algorithm complexity is $O((N^2 + \frac{\mu(\Omega)}{\epsilon})K)$, where K is the number of iterations. Let Z be the number of sensors out of the final backbone network. BCCML's algorithms complexity is calculated as $O((\sum_{z=0}^{Z}(N-z)^2 + O(\frac{\mu(\Omega)}{\epsilon}))K)$. The complexity of the worst case, $Z = 1$, is then $O((N^2 + \frac{\mu(\Omega)}{\epsilon})NK)$. Different from CCML and BCCML, DCML is a distributed algorithm. Thus, we focus on Sensor n's complexity in DCML. Note that MST can be obtained by a distributed minimum spinning tree algorithm, GHS Algorithm [122], with complexity $O(N\log(N))$. At the k-th stop, DCML calculates (a) $\mathbf{c}_n(\mathbf{P})$s and $\boldsymbol{V}_n^H(\mathbf{P})$, (b) $\mathbb{F}_n(\mathbf{P}, \mathcal{I})$ and (c) MST with complexities $O(\frac{\mu(\boldsymbol{V}_n^H(\mathbf{P}^k))}{\epsilon})$, $O(N)$, and $O(N\log(N))$, respectively. In total, DCML's complexity during K stops is then $O(KN\log(N) + \sum_{k=1}^{K} \frac{\mu(\boldsymbol{V}_n^H(\mathbf{P}^k))}{\epsilon})$. Obviously, as a distributed algorithm, DCML's complexity is smaller than that of CCML and BCCML. i.e., $O(KN\log(N) + \sum_{k=1}^{K} \frac{\mu(\boldsymbol{V}_n^H(\mathbf{P}^k))}{\epsilon}) < O((N^2 + \frac{\mu(\Omega)}{\epsilon})K) < O((N^2 + \frac{\mu(\Omega)}{\epsilon})NK)$.

[4]The integral is approximated by the summation over uniform samples.

5.4.2 Communication Overhead

In centralized implementations, the relocation includes the following stages: (i) Sensors send their spatial information to AP via multi-hop communications; (ii) AP calculates the final deployment by running CCML/BCCML in terms of the collected information; (iii) AP sends final deployment to each sensor by multi-hop communications; (iv) Sensors are relocated to their destinations based on the received deployment. In the worst case, where sensors are connected with a line topology, the sensor with a hops to AP will transfer $N-a$ messages from itself and $N-1-a$ farther sensors in Stage (2). Thus, the communication overhead in Stage (ii) is $O(\sum_{a=1}^{N-1} a) = O(N^2)$. Similarly, the communication overhead in Stage (iv) is also $O(N^2)$. Therefore, the total communication overhead in centralized implementations (CCML and BCCML) is $O(N^2)$.

In what follows, we analyze the communication in the distributed implementation. At each stop, sensors need to compute (a) their own Voronoi Diagrams and (b) MST in terms of their neighbor information. Sensor n can compute its Voronoi Diagram by exchanging message with its one-hop neighbors [6], [10]. Since each sensor has very limited number of one-hop neighbours, the communication overhead of N sensors for Voronoi Diagram is $O(N)$. In addition, communication overhead for distributed MST for each stop is $O(N\log(N))$ [122]. Therefore, the total communication overhead in a K-stop distributed implementation (DCML) is $O(K(N\log(N)+N)) = O(KN\log(N))$.

5.5 Extension

In this section, I extend the proposed algorithms, CCML and DCML, to other sensing tasks: area coverage and target coverage. We employ the binary coverage model [2]–[12], [14], [32], [33] in which Sensor n can only detect the points within its sensing radius $R_{s,n}$. Intuitively, in order to decrease the sensing uncertainty, CCML and DCML deploy sensors into high-density regions, and thus the points with high density are more likely to be covered. To cover the objects in different tasks, the density function $f(\boldsymbol{\omega})$ in (5.2) should be predetermined to

highlight the points around the objects of interest. In the following three subsections, we introduce three kinds of coverage and propose the corresponding density functions.

5.5.1 Area Coverage

Without any prior information about the target region, the density function is chosen to be uniform, i.e., $f(\boldsymbol{\omega}) = \frac{1}{\int_\Omega d\boldsymbol{\omega}}, \forall \boldsymbol{\omega} \in \Omega$. Under such circumstances, maximizing the area covered by sensors is a primary task. To evaluate the corresponding sensing performance, we employ area coverage [6], [23], [32] (the proportion of covered area) defined by

$$C^{\mathcal{A}}(\mathbf{P}) = \frac{\int_{\bigcup_{n=1}^{N} \mathbb{B}(\mathbf{p}_n, R_{s,n})} d\boldsymbol{\omega}}{\int_\Omega d\boldsymbol{\omega}} = \frac{\sum_{n=1}^{N} \int_{\boldsymbol{V}_n^H(\mathbf{P}) \bigcap \mathbb{B}(\mathbf{p}_n, R_{s,n})} d\boldsymbol{\omega}}{\sum_{n=1}^{N} \int_{\boldsymbol{V}_n^H(\mathbf{P})} d\boldsymbol{\omega}}. \tag{5.24}$$

The experimental results in Section 5.6 show that CCML and DCML algorithms with uniform density function provide a large area coverage in addition to a small distortion.

5.5.2 Target Coverage

In another popular scenario, sensors are deployed to collect detailed information from the targets with known locations [24]–[27]. Let $\mathcal{T} = \{t_1, t_2, \ldots, t_L\}$ be the set of known targets, $\widehat{\mathcal{T}} = \{t | \min_n \left(\frac{\|t - \mathbf{p}_n\|}{R_{s,n}} \right) \leq 1, t \in \mathcal{T}\}$ be the set of targets that covered by at least one sensor. Then, area coverage $C^{\mathcal{T}}(\mathbf{P})$ - the proportion of covered target points - can be written as

$$C^{\mathcal{T}}(\mathbf{P}) = \frac{\mathbf{card}(\widehat{\mathcal{T}})}{\mathbf{card}(\mathcal{T})} = \frac{\sum_{n \in \mathcal{S}(\mathbf{P})} \int_{\boldsymbol{V}_n^H(\mathbf{P}) \bigcap \mathbb{B}(\mathbf{p}_n, R_{s,n})} \left[\sum_{l=1}^{L} \phi\left(\|\boldsymbol{\omega} - t_l\| \right) \right] d\boldsymbol{\omega}}{\sum_{n \in \mathcal{S}(\mathbf{P})} \int_{\boldsymbol{V}_n^H(\mathbf{P})} \left[\sum_{l=1}^{L} \phi\left(\|\boldsymbol{\omega} - t_l\| \right) \right] d\boldsymbol{\omega}}, \tag{5.25}$$

where $\phi(\cdot)$ is the unit impulse response, $\mathbf{card}(S)$ is the cardinality of the set S. To emphasize the importance of discrete targets, we model the density function as a Gaussian mixture centered at discrete targets. The corresponding density function can be written as

$$f(\boldsymbol{\omega}) = \sum_{l=1}^{L} A_l e^{-\frac{\|\boldsymbol{\omega} - t_l\|^2}{R_{s,n}^2}} \tag{5.26}$$

where A_l reflects the comparative importance of the target q_l. Similarly, CCML and DCML can also be extended to maximize barrier coverage [28]–[30].

5.6 Performance Evaluation

We provide the simulation results for two different MWSNs: (1) MWSN1: A homogeneous MWSN in which all sensors have the same characteristics. (2) MWSN2: A heterogeneous MWSN including sensors with different sensing, moving cost parameters, and battery energies. In addition, we employ uniform density function, $f(\boldsymbol{\omega}) = 1$, for MWSN1 while the non-uniform density function in [6], [32] is employed for MWSN2. The non-uniform density function is the sum of five Gaussian functions of the form $5exp(6(-(x-x_{center})^2-(y-y_{center})^2))$. The centers (x_{center}, y_{center}) are (2,0.25), (1,2.25), (1.9,1.9), (2.35,1.25) and (0.1,0.1). Moreover, the target region, Ω, which is also the same as in [6], [32], is determined by the polygon vertices (0,0), (2.125,0), (2.9325,1.5), (2.975,1.6), (2.9325,1.7), (2.295,2.1), (0.85,2.3), (0.17,1.2). Also, I set the power consumption after sensor relocation as $\beta = 1$. As a result, the energy constraints can be calculated by $\gamma_n = e_n - T$, where e_n is Sensor n's battery energy. Other parameters are provided in Table 5.1. Moreover, I generate initial sensor deployments randomly, i.e., every node location is generated with uniform distribution on Ω. To guarantee the initial full-connectivity, I sequentially generate random node locations, and only keep a node if it connects with at least one previous node. The maximum number of iterations is set to 100[5].

Table 5.1: Simulation Parameters of Mobile Wireless Sensor Networks

Parameters	N	$\eta_1-\eta_8$	$\eta_9-\eta_{32}$	$\xi_1-\xi_8$	$\xi_9-\xi_{32}$	$R_{s,1}-R_{s,8}$	$R_{s,9}-R_{s,32}$	e_1-e_{28}	$e_{29}-e_{32}$
MWSN1	32	1	1	1	1	0.2	0.2	2	2
MSWN2	32	1	4	2	1	0.3	0.15	2	0.8

To evaluate the performance, I compare the distortion (5.4) and area coverage (5.24) of CCML, BCCML, and DCML with those of VFA [36], Lloyd-α [10], and DEED [10]. We run the algorithms for: (i) the centralized scheme where each sensor's energy consumption for relocation is determined by the distance from the initial location and the final location; and (ii) the distributed scheme where each sensor's energy consumption for relocation is

[5]100 is large enough for the proposed algorithms to converge. Therefore, instead of using the stop threshold ϵ in CCML and BCCML, I run all algorithms 100 iterations in the experiments.

determined by the total distance of its specific (100-stop) movement path. Several important simulation details are provided as follows. Since network lifetime is not considered in VFA [36], it is impossible to apply the original VFA to satisfy the required network lifetime. Thus, I propose a variant of VFA in which each sensor stops moving after the predetermined energy, γ_n, is consumed. Furthermore, when the communication range R_c is limited, the lack of full-connectivity prevents VFA, Lloyd-α, and DEED from operating in a distributed scheme. To compare them with our DCML Algorithm, sensors need to have global information in VFA, Lloyd-α, and DEED. Another issue is that sensors may be divided into multiple disconnected sub-graphs after running VFA, Lloyd-α, and DEED because of the limited communication range. Under such circumstances, I compute the distortions associated with different sub-graphs and report the minimum one. In other words, I focus on the best performances that VFA, Lloyd-α, and DEED can reach in MWSNs when communication range is limited. Nonetheless, when I compute the distortion for our proposed algorithms, only sensors in the actual backbone network (the sub-graph including AP, i.e., Sensor 1), are taken into account, which gives our algorithm more advantage over the existing algorithms.

Simulation results for the centralized scheme are provided in Figs. 5.2 and 5.3. From Figs. 5.2a and 5.2c, I observe that both VFA and CCML algorithms generate fully connected final deployments for the required network lifetime, $T = 1.3$. By setting α to 0.2, Lloyd-α achieves a similar network lifetime $T = 1.31$. However, Lloyd-0.2 generates a disconnected network where 12 sensors are placed out of the backbone network. The corresponding distortions for VFA, Lloyd-0.2, and CCML are, respectively, 0.17, 0.78, and 0.14.

The centralized sensor relocations in heterogeneous MWSN (MWSN2) are illustrated in Figs. 5.2d, 5.2e, and 5.2f. In MWSN2, BCCML activates 30 sensors to sense the target region while the other two sensors are deactivated because of their low battery energy. VFA and Lloyd-α attempt to use all sensors to finish the sensing task, but AP can only collect information from the active sensors shown by red dots. BCCML's distortion, 0.29, is much smaller than that of VFA and Lloyd-α, which are 4.16 and 0.66, respectively.

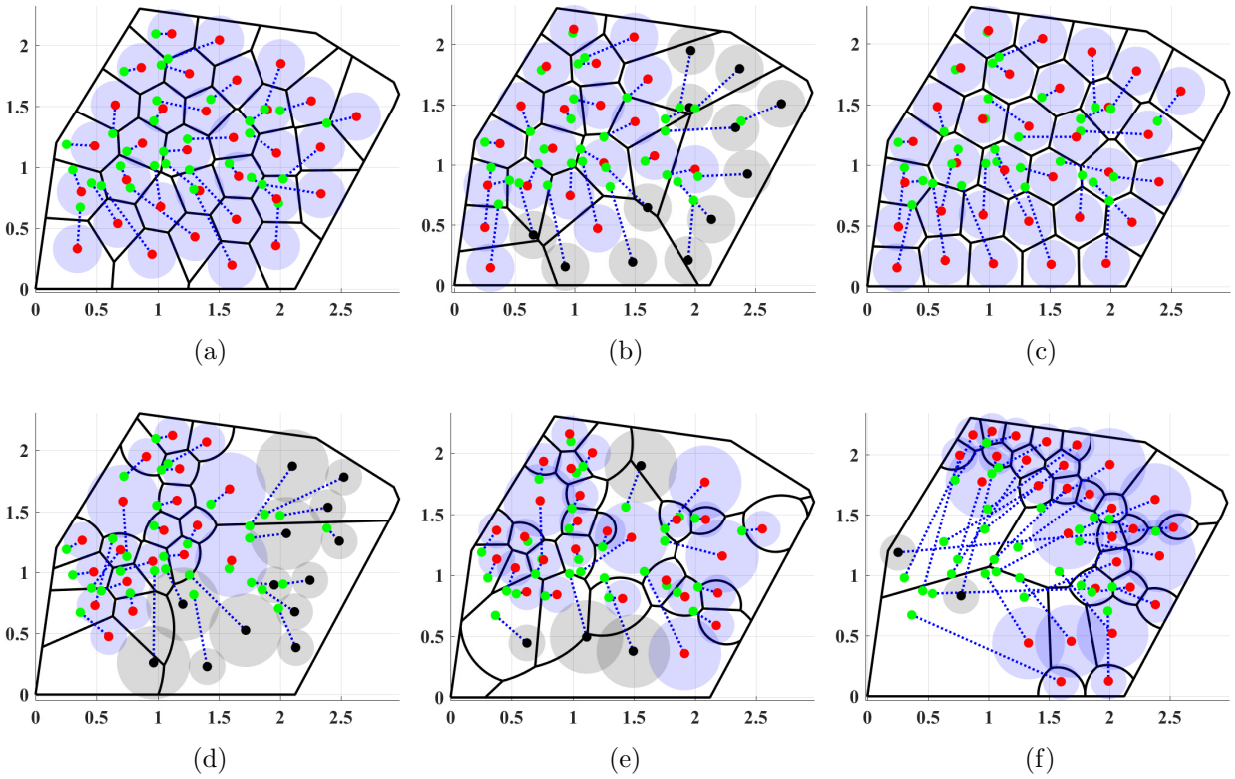


Figure 5.2: Centralized sensor deployments: (a) VFA in MWSN1; (b) Lloyd-α in MWSN1; (c) CCML in MWSN1; (d) VFA in MWSN2; (e) Lloyd-α in MWSN2; (f) BCCML in MWSN2. The initial sensor locations are denoted by green dots. The final locations of active and inactive sensors are denoted by red and black dots. The sensing regions of active and inactive sensors are denoted by blue and black. The movement paths are denoted by blue lines.

More detailed performance comparisons are provided in Figs. 5.3a, 5.3b, 5.3c, and 5.3d. In Fig. 5.3a, for any given network lifetime, CCML Algorithm provides a lower distortion compare to other algorithms in the homogeneous MWSN1. Similarly, BCCML Algorithm outperforms other algorithms in heterogeneous MWSNs shown in Fig. 5.3b. It is also noteworthy that low distortion is accompanied with high area coverage in MWSN1 where the density function is uniform. Unfortunately, such a relationship does not hold for MWSN2 where the density function is non-uniform. Consequently, one can approximately optimize the area coverage by CCML and BCCML with a uniform density function.

There are two primary reasons why the proposed CCML and BCCML perform better than the existing algorithms, VFA, Lloyd-α, and DEED. (i) The existing algorithms do not take connectivity into consideration. As a result, when the communication range is small, VFA,

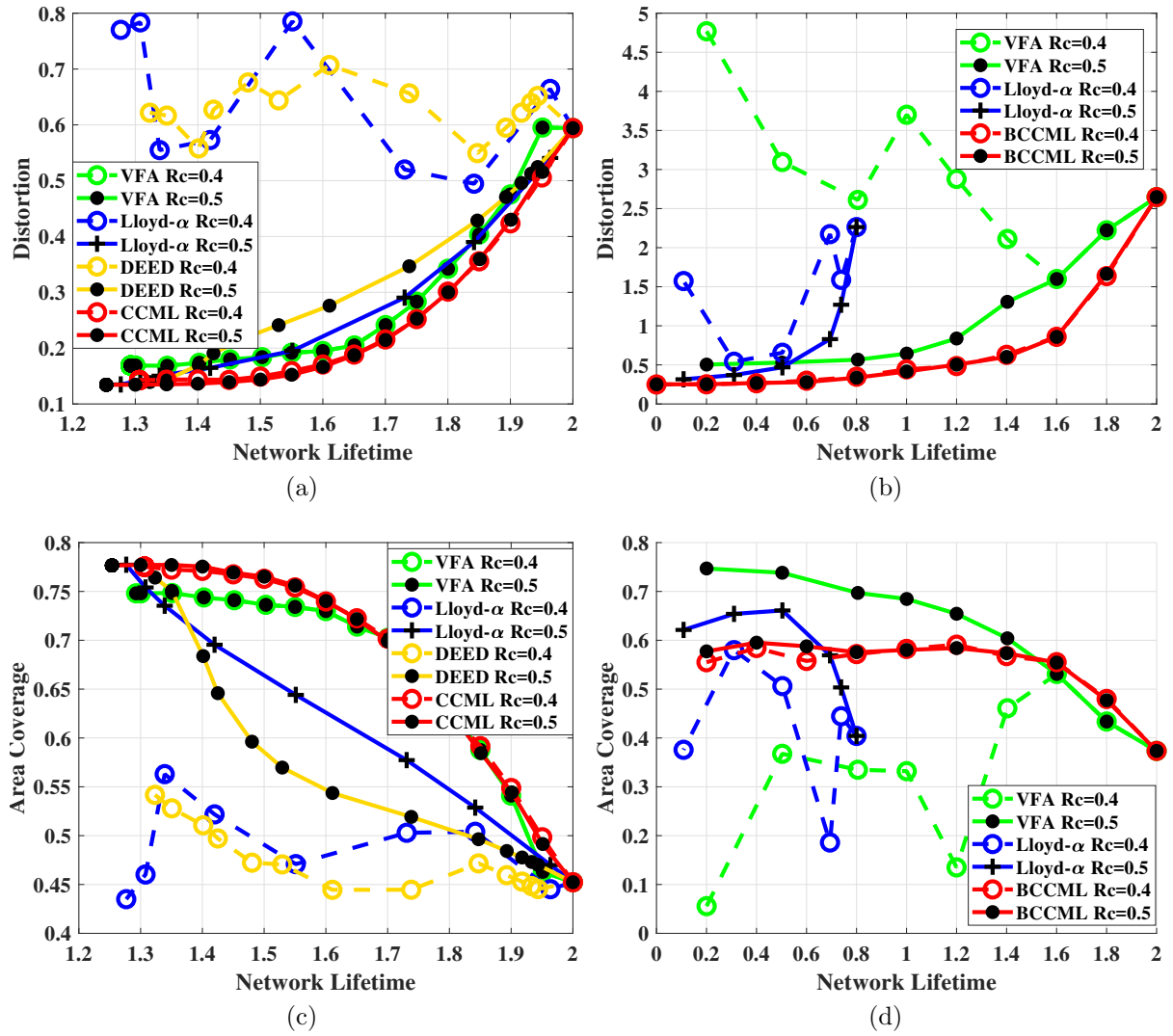


Figure 5.3: Performance comparison for centralized sensor deployment. (a) Distortion in MWSN1; (b) Distortion in MWSN2; (c) Area coverage in MWSN1; (d) Area coverage in MWSN2.

Lloyd-α, and DEED may generate disconnected networks and then large distortions. (ii) In the existing algorithms, each sensor attempts to save its own energy consumption which results in unbalanced energy consumption among sensors and then short network lifetime. However, CCML and BCCML determine the relocation considering all sensors' residual energy.

Besides, our proposed algorithms perform well in both homogeneous and heterogeneous MWSNs, but the existing algorithms, which are designed for homogeneous MWSNs, have very restricted performance in heterogeneous MWSNs or even cannot be applied to hetero-

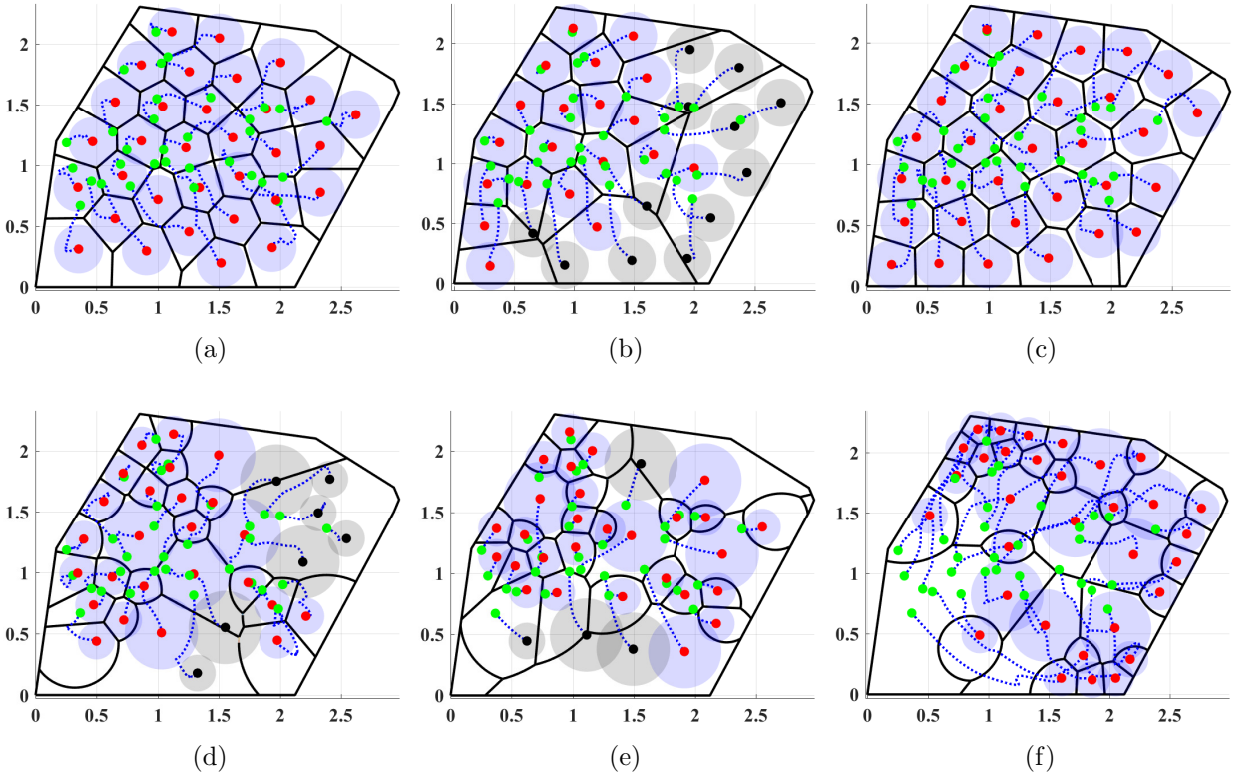


Figure 5.4: Distributed sensor deployments: (a) VFA in MWSN1; (b) Lloyd-α in MWSN1; (c) DCML in MWSN1; (d) VFA in MWSN2; (e) Lloyd-α in MWSN2; (f) DCML in MWSN2. The initial sensor locations are denoted by green dots. The final locations of active and inactive sensors are denoted by red and black dots. The sensing regions of active and inactive sensors are denoted by blue and black. The movement paths are denoted by blue lines.

geneous MWSNs. Note that the implementation of DEED needs both gradient and Hessian matrix of the objective function (5.4). To our best knowledge, the theoretical computation of Hessian matrix in heterogeneous MWSNs is still an open problem. Although one can approximate the second-order derivatives by numerical methods, the corresponding extreme time complexity prevents DEED from being a feasible solution. Therefore, DEED cannot be extended to heterogeneous MWSNs. Different from DEED, Lloyd-α, which only needs gradient, can be extended to heterogeneous MWSNs as the calculation of the gradient in heterogeneous MWSNs already proposed in our previous work [32]. Unfortunately, when sensors are equipped with variant battery energies, Lloyd-α can only achieve a short network lifetime. In MWSN2, where 4 sensors are equipped with a low battery energy, 0.8, Lloyd-α still uses all sensors to sense the target region. Note that the network will die after

the first node runs out of its battery energy. As a result, Lloyd-α cannot achieve a network lifetime larger than 0.8 in MWSN2 (see Fig. 5.3b). However, the proposed BCCML Algorithm appropriately selects a subset of sensors to finish the sensing task.

Another advantage of BCCML Algorithm over Lloyd-α and DEED is its tractability. BCCML Algorithm directly controls the network lifetime while Lloyd-α and DEED indirectly influence network lifetime by tuning the hyperparameters α and δ, respectively. There is no explicit relationship between α (or δ) and network lifetime. Thus, one has to attempt different values of α (or δ) in Lloyd-α (or DEED) to reach the required network lifetime.

In what follows, I analyze the impact of communication range, R_c, on the performance. The sensors with larger communication range, R_c, are more likely to construct or maintain a connected network. According to our simulation results, VFA, Lloyd-α, DEED, CCML and BCCML keep full-connectivity for the two considered MWSNs when the communication range is large, e.g., $R_c = 0.5$. However, when $R_c = 0.4$, VFA, Lloyd-α, DEED lose connectivity in some cases. In general, a shorter network lifetime implies that more energy can be used for relocation, and then smaller distortions can be achieved. We observe from Figs 5.3a and 5.3b that all algorithms provide non-decreasing Distortion-Lifetime functions for the cases of $R_c = 0.5$. However, when $R_c = 0.4$, the distortions of VFA, Lloyd-α, and DEED fluctuate because of the unpremeditated loss of connectivity in some cases. On the contrary, the distortions of CCML and BCCML are still non-decreasing functions of network lifetime because the backbone network is appropriately selected and maintained by CCML and BCCML. In sum, for VFA, Lloyd-α, and DEED algorithms, the distortions are significantly increased when R_c is decreased from 0.5 to 0.4. However, CCML/BCCML provides similar performance for both R_cs.

The above analysis also works for the distributed sensor relocation scheme. The distributed relocations in both MWSNs are illustrated in Fig. 5.4, and the corresponding performances are compared in Figs. 5.5a, 5.5b, 5.5c, and 5.5d. Different from the sensor relocation in the centralized implementation (Fig. 5.2) where movement paths are straight lines from initial

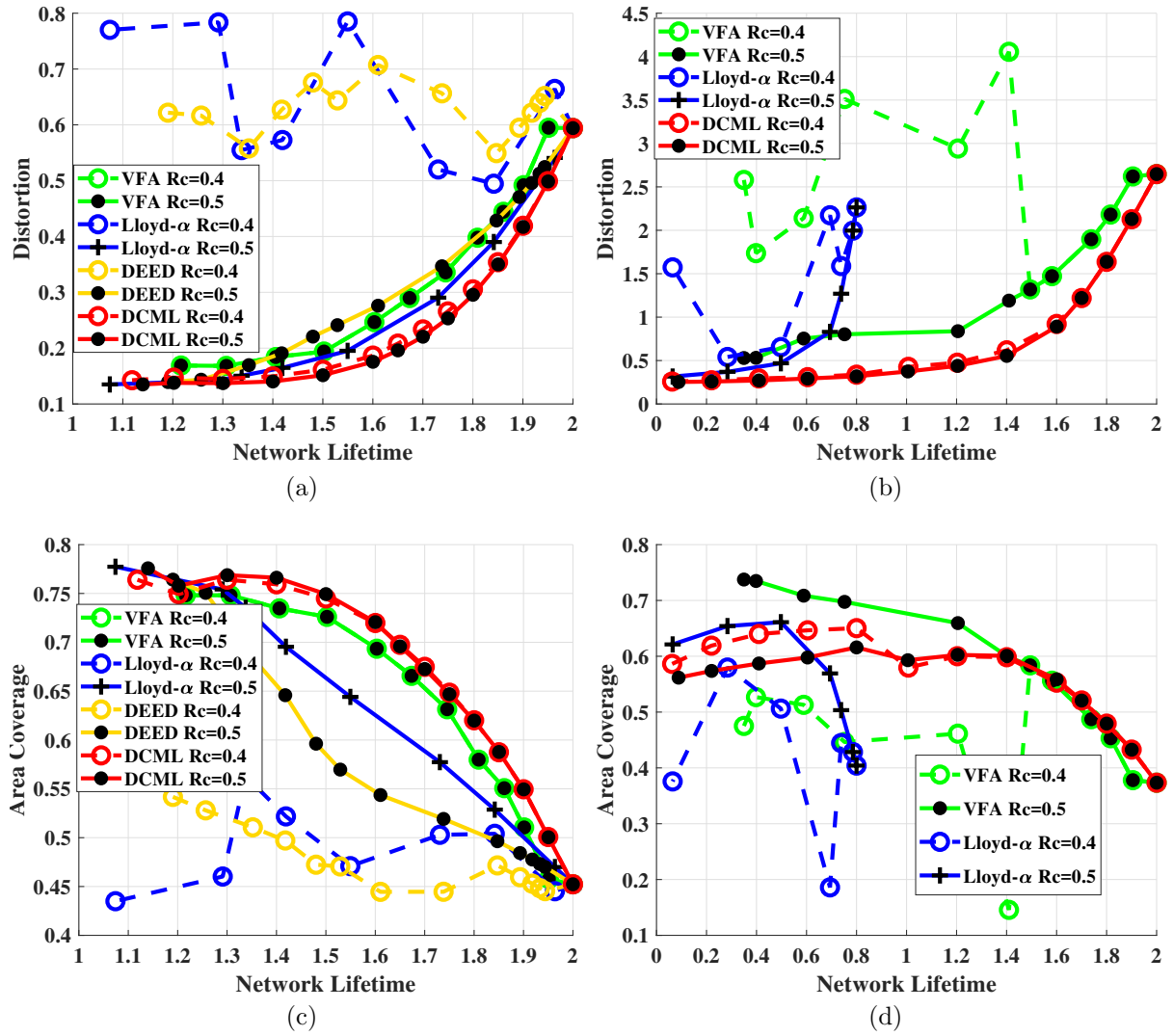
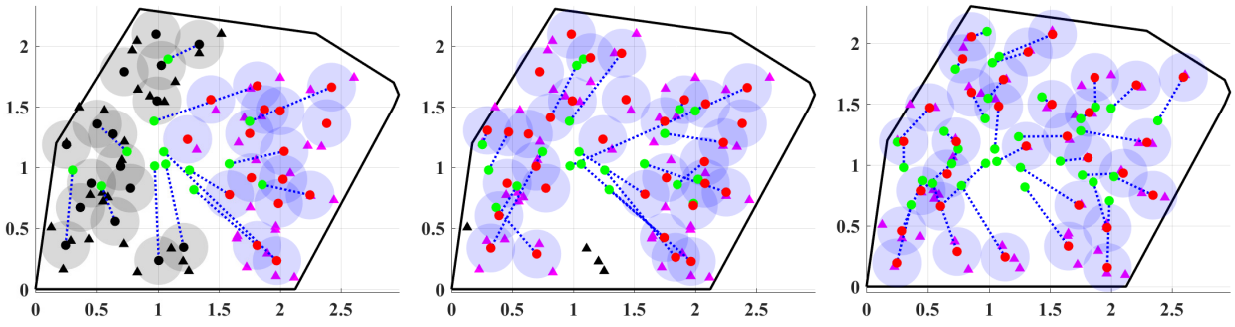


Figure 5.5: Performance comparison for distributed sensor deployment. (a) Distortion in MWSN1; (b) Distortion in MWSN2; (c) Area coverage in MWSN1; (d) Area coverage in MWSN2.

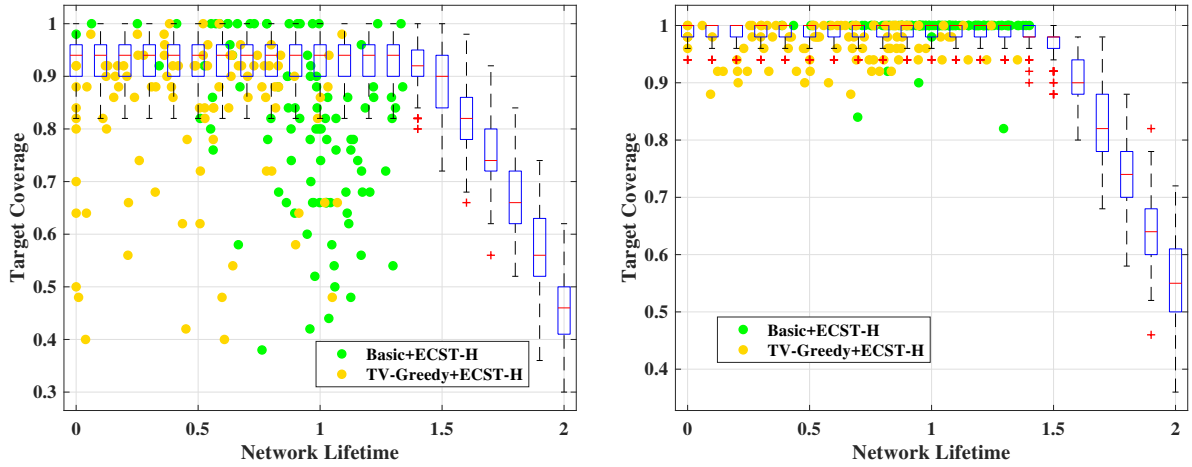
locations to final locations, the sensor relocation in the distributed implementation (Fig. 5.4) are represented by broken lines with multiple stops.

Last, I provide sensor relocation of three different algorithms (Basic+ECST-H [26], TV-Greedy+ECST-H [26], and CCML) in Figs. 5.6 and 5.7 to confirm that CCML Algorithm can be successfully extended to target coverage problems. Both Basic+ECST-H and TV-Greedy+ECST-H consist of three stages: (i) A subset of sensors are placed to cover all targets; and (ii) other sensors are placed to guarantee connectivity; (iii) Hungarian Algorithm is employed to reduce the total energy consumption. According to our experiments,



(a) (b) (c)

Figure 5.6: The target coverage in MWSN1. (a) Basic+ECST-H; (b) TV-Greedy+ECST-H; (c) CCML. The covered targets and uncovered targets are denoted by magenta triangles and black triangles, respectively.



(a) (b)

Figure 5.7: The target coverage in MWSN1. (a) $R_c = 0.4$, $R_s = 0.2$; (b) $R_c = 0.5$, $R_s = 0.25$. The target coverage of Basic+ECST-H, TV-Greedy+ECST-H, and CCML are, respectively, denoted by green dots, yellow dots and blue boxes.

Basic+ECST-H and TV-Greedy+ECST-H require more sensors than CCML to achieve both full-coverage and full-connectivity. In Fig. 5.6a, all sensors are scheduled to cover targets in Stage (i) of Basic+ECST-H, and then no sensor is available in Stage (ii). As a result, Basic+ECST-H with $R_c = 0.4$ is terminated with four disconnected subgraphs in which the largest sub-graph merely covers 48% of targets. In Fig. 5.6b, 32 sensors are not enough to achieve full-coverage in Stage (i) of TV-Greedy+ECST-H, and only 92% of targets are covered. However, with the same number of sensors, our proposed CCML Algorithm covers

all targets and ensures full-connectivity. To perform a statistical performance analysis, I run the relocation algorithms with 100 random target deployments[6]. The detailed comparisons[7] of Basic+ECST-H, TV-Greedy+ECST-H, and CCML are provided in Figs. 5.7a and 5.7b. Given a specific target coverage, CCML, on average, achieves a longer network lifetime.

[6]Each target deployment consists of 50 target locations.

[7]For each required network lifetime, a boxplot is employed to display minimum, first quartile, median, third quartile, and maximum distortions of CCML with 100 target deployments. However, the target coverages of Basic+ECST-H and TV-Greedy+ECST-H are represented by, in total, 200 dots since network lifetime is not explicitly controlled by Basic+ECST-H and TV-Greedy+ECST-H.

Chapter 6

Conclusion

The performance (sensing quality, connectivity, and energy consumption) optimization of wireless sensor network (WSN) is studied in this dissertation.

First, I studied the deployment of sensors for three kinds of model in heterogeneous wireless sensor networks. The optimal partitioning and gradient are supplied to search optimal sensor deployment. Similar to homogeneous WSNs, the necessary condition for optimal deployment in heterogeneous WSNs implies that every sensor node location should coincide with the centroid of its own optimal sensing region. Moreover, I considered connectivity and modeled the sensor deployment problem as a multi-center optimization problem with different performance functions. By defining an appropriate performance measure, I proposed a Restrained Lloyd algorithm and a Deterministic Annealing algorithm to optimize sensor deployment in both homogeneous and heterogeneous WSNs. The simulation results show that both DA and RL algorithms provide a fully connected network and outperform Lloyd algorithm, Genetic Algorithm, and Virtual Force Algorithm when communication range is limited. The DA is not sensitive to initial conditions. Moreover, both RL and DA have been applied to solve sensor deployment in non-convex environments. Our future work includes the optimal sensor deployment with the random connection model in heterogeneous WSNs.

Second, I studied quantizers with parameterized distortion measures for an application to

UAV deployments. Instead of using the traditional mean distance square as the distortion, I introduce a distortion function which models the energy consumption of UAVs in dependence of their heights. I derived the unique parameter optimized quantizer – a uniform scalar quantizer with an optimal common parameter – for uniform source density in one-dimensional space. In addition, two Lloyd-like algorithms are designed to minimize the distortion in two-dimensional space. Numerical simulations demonstrate that the common weight property extends to two-dimensional space for a uniform density.

Third, two-tier quantizer and its application, energy efficiency in a two-tier wireless sensor network (WSN), are studied in this dissertation, too. Different from one-tier WSNs, the two-tier WSN collects data from a large-scale wireless sensor network to fusion centers through access points. The necessary condition for optimal deployment implies that every AP location should be deployed between the centroid of its cell and its associated FC. In addition, the AP-Sensor power function is introduced and analyzed to provide the minimum AP power with a sensor power constraint. We also proposed Lloyd-like algorithms to minimize the distortion. According to the experiments, the algorithms proposed in this dissertation significantly save the weighted power or energy in two-tier WSNs.

Last, the trade-off between sensing quality and energy consumption, which is dominated by movement, is discussed in this dissertation. I explore the optimal sensor deployment to minimize sensing uncertainty with a network lifetime constraint in both homogeneous and heterogeneous mobile wireless sensor networks. According to the analysis, full-connectivity is necessary to minimize the sensing uncertainty in homogeneous MWSNs. The necessary condition for an optimal deployment implies that sensors should move towards the centroid within their own feasible regions, determined by both the battery energies and the communication range. With the help of these necessary conditions, two centralized sensor relocation algorithms, Centralized Constrained Movement Lloyd Algorithm and Backwards-stepwise Centralized Constrained Movement Lloyd Algorithm, are designed for homogeneous and heterogeneous MWSNs, respectively. Moreover, a distributed realization, Distributed

Constrained Movement Lloyd Algorithm, whose performance is similar to the centralized scheme, is also provided in this dissertation. Furthermore, by manually changing the density function, the proposed sensor relocation algorithms can be extended to solve target coverage. The simulation results show that the proposed algorithms outperform the existing algorithms in the literature (VFA, Lloyd-α, DEED) when a minimum network lifetime is given in homogeneous and heterogeneous MWSNs. Compared with the existing target coverage algorithms, such as Basic+ECST-H and TV-Greedy+ECST-H, CCML Algorithm provides a more flexible trade-off between target coverage and network lifetime.

Bibliography

[1] M. Cardei and J. Wu, *Handbook of Sensor Networks*, 1st ed. Hoboken, NJ, USA: John Wiley & Sons, 2004.

[2] C. Zhu, C. Zheng, L. Shu, and G. Han, "A survey on coverage and connectivity issues in wireless sensor networks," *ELSEVIER J. Netw. Comput. Appl.*, vol. 35, no. 2, pp. 619–632, Mar. 2012.

[3] B. Wang, "Coverage problems in sensor networks: A survey," *ACM Comput. Surveys*, vol. 43, no. 4, pp. 1–53, Oct. 2011.

[4] M. R. Senouci, A. Mellouk, K. Asnoune, and F. Y. Bouhidel, "Movement-assisted sensor deployment algorithms: A survey and taxonomy," *IEEE Commun. Surveys Tut.*, vol. 17, no. 4, pp. 2493–2510, Mar. 2015.

[5] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Trans. on Robot. Autom.*, vol. 20, no. 2, pp. 243–255, Apr. 2004.

[6] J. Cortes, S. Martinez, and F. Bullo, "Spatially-distributed coverage optimization and control with limited-range interactions," *ESAIM: COCV*, vol. 11, no. 4, pp. 691–719, Oct. 2005.

[7] K. Sugimoto, T. Hatanaka, M. Fujita, and N. Huebel, "Experimental study on persistent coverage control with information decay," in *IEEE Annu. Conf. Soc. Instrum. Control Eng. Japan (SICE)*, Jul. 2015.

[8] M. T. Nguyen, L. Rodrigues, C. S. Maniu, and S. Olaru, "Discretized optimal control approach for dynamic multi-agent decentralized coverage," in *IEEE Int. Symp. Intell. Control (ISIC)*, Sep. 2016.

[9] F. Li, J. Luo, S. Xin, and Y. He, "Autonomous deployment of wireless sensor networks for optimal coverage with directional sensing model," *ELSEVIER Comput. Netw.*, vol. 108, no. 10, pp. 120–132, Oct. 2016.

[10] Y. Song, B. Wang, Z. Shi, K. Pattipati, and S. Gupta, "Distributed algorithms for energy-efficient even self-deployment in mobile sensor networks," *IEEE Trans. Mobile Comput.*, vol. 13, no. 5, pp. 1035–1047, Apr. 2014.

[11] G. Xing, X. Wang, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated coverage and connectivity configuration for energy conservation in sensor networks," *ACM Trans. Sensor Netw.*, vol. 1, no. 1, pp. 36–72, Aug. 2005.

[12] X. Bai, S. Kumar, D. Xuan, Z. Yun, and T. H. Lai, "Deploying wireless sensors to achieve both coverage and connectivity," in *ACM Int. Symp. Mobile Ad Hoc Netw. Comput. (MobiHoc)*, May 2006.

[13] M. Moarref and L. Rodrigues, "An optimal control approach to decentralized energy-efficient coverage problems," in *Elsevier The International Federation of Automatic Control (IFAC)*, vol. 47, Aug. 2014, pp. 6038–6043.

[14] H. Yousefi'zadeh, H. Jafarkhani, and J. Kazemitabar, "A study of connectivity in MIMO fading ad-hoc networks," *IEEE J. Commun. Netw.*, vol. 11, no. 1, pp. 47–56, Feb. 2009.

[15] X. Liu, K. Wub, Y. Zhu, L. Kong, and M. Wua, "Mobility increases the surface coverage of distributed sensor networks," *ELSEVIER Comput. Netw.*, vol. 57, no. 11, pp. 2348–2363, Aug. 2013.

[16] V. N. Had, "Decentralized control of three dimensional mobile robotic sensor networks," *arxiv*, Jun. 2016.

[17] E. Koyuncu, R. Khodabakhsh, N. Surya, and H. Seferoglu, "Deployment and trajectory optimization for UAVs: A quantization theory approach," in *IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2018. eprint: `1708.08832v5`.

[18] J. Guo, E. Koyuncu, and H. Jafarkhani, "A source coding perspective on node deployment in two-tier networks," *IEEE Trans. Commun.*, vol. 66, no. 7, pp. 3035–3049, Jul. 2018.

[19] K. Dantu, M. Rahimi, H. Shah, S. Babel, A. Dbariwal, and G. S. Sukhatme, "Robomote: Enabling mobility in sensor network," in *IEEE 4th Int. Symp. Inf. Process. Sensor Netw. (IPSN)*, Apr. 2005.

[20] J. Wu and S. Yang, "Optimal movement-assisted sensor deployment and its extensions in wireless sensor networks," in *IEEE 12th Int. Conf. Parallel Distrib. Syst. (ICPADS)*, Jul. 2006.

[21] W. Li, "On wireless sensors covering and movement problem," in *IEEE Int. Conf. Wireless Commun., Netw. Mobile Comput. (WiCOM)*, Sep. 2007.

[22] M. R. Senouci and A. Mellouk, "Localized movement-assisted sensor deployment algorithm for hole detection and healing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 5, pp. 1267–1276, May 2014.

[23] Y. Zou and K. Chakrabarty, "Sensor deployment and target localization in distributed sensor networks," *ACM Trans. Embedded Comput. Syst.*, vol. 3, no. 1, pp. 61–91, Feb. 2004.

[24] M. Rout and R. Roy, "Self-deployment of mobile sensors to achieve target coverage in the presence of obstacles," *IEEE Sensors J.*, vol. 16, no. 14, pp. 5837–5842, Jul. 2016.

[25] Z. Chen, X. Gao, F. Wu, and G. Chen, "A PTAS to minimize mobile sensor movement for target coverage problem," in *IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr. 2016.

[26] Z. Liao, J. Wang, S. Zhang, J. Cao, and G. Min, "Minimizing movement for target coverage and network connectivity in mobile sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 7, pp. 1971–1983, Jul. 2015.

[27] A. N. Njoya, W. Abdou, A. Dipanda, and E. Tonye, "Evolutionary-based wireless sensor deployment for target coverage," in *IEEE Int. Conf. Signal-Image Technol. & Internet-Based Syst. (SITIS)*, Nov. 2015.

[28] D. Z. Chen, Y. Gu, J. Li, and H. Wang, "Algorithms on minimizing the maximum sensor movement for barrier coverage of a linear domain," *Springer Discrete & Computational Geometry*, vol. 50, no. 2, 374–408, Sep. 2013.

[29] S. Li and H. Shen, "Minimizing the maximum sensor movement for barrier coverage in the plane," in *IEEE Int. Conf. Comput. Commun. (INFOCOM)*, May 2015.

[30] B. Z. Z. He, "An improved algorithm for minimizing the maximum sensor movement in linear barrier coverage," in *IEEE Global Telecommun. Conf. (GLOBECOM)*, Dec. 2016.

[31] A. S. S and J. D. Beegum, "Enhancing the life time of sensor network with energy awareness and clutter adaptability," in *IEEE Int. Conf. Control Commun. & Comput. India (ICCC)*, Mar. 2016.

[32] J. Guo and H. Jafarkhani, "Sensor deployment with limited communication range in homogeneous and heterogeneous wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 10, pp. 6771–6784, Oct. 2016.

[33] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, 2nd ed. New York, NY, USA: John Wiley & Sons, 2000.

[34] E. Koyuncu and H. Jafarkhani, "On the minimum average distortion of quantizers with index-dependent distortion measures," *IEEE Trans. Signal Process.*, vol. 65, no. 17, pp. 4655–4669, Sep. 2017.

[35] Z. Yong and W. Li, "A sensor deployment algorithm for mobile wireless sensor networks," in *IEEE Chinese Control and Decision Conf. (CCDC)*, Jun. 2009.

[36] Y. Yoon and Y.-H. Kim, "An efficient genetic algorithm for maximum coverage deployment in wireless sensor networks," *IEEE Commun. Surveys Tut.*, vol. 43, no. 5, pp. 1473–1483, Apr. 2013.

[37] M. Gani, "Optimal deployment control for a heterogeneous mobile sensor network," in *IEEE 9th Int. Conf. Control, Autom., Robot. Vision (ICARCV)*, Dec. 2006.

[38] S. Karimi-Bidhendi, J. Guo, and H. Jafarkhani, "Using quantization to deploy heterogeneous nodes in two-tier wireless sensor networks," in *IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2019. arXiv: 1901.06742.

[39] C. Hu, X. Wang, Z. Yang, J. Zhang, Y. Xu, and X. Gao, "A geometry study on the capacity of wireless networks via percolation," *IEEE Trans. Commun.*, vol. 58, no. 10, pp. 2916–2925, Oct. 2010.

[40] N. Bartolini, T. Calamoneri, T. F. L. Porta, and S. Silvestri, "Autonomous deployment of heterogeneous mobile sensors," *IEEE Trans. Mobile Comput.*, vol. 10, no. 6, pp. 753–766, Jun. 2011.

[41] M. Cardei, M. Thai, Y. Li, and W. Wu, "Energy-efficient target coverage in wireless sensor networks," in *IEEE 24th Annu. Joint Conf. Comput. Commun. Societies.*, Aug. 2005, pp. 1976–1984.

[42] A. Saipulla, C. Westphal, B. Liu, and J. Wang, "Barrier coverage of line-based deployed wireless sensor networks," in *IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr. 2009.

[43] S. He, X. Gong, J. Zhang, J. Chen, and Y. Sun, "Curve-based deployment for barrier coverage in wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 2, pp. 724–735, Feb. 2014.

[44] J. A. Cobb, "Improving the lifetime of non-penetrable barrier coverage in sensor networks," in *IEEE Int. Conf. Distrib. Comput. Syst. (ICDCSW)*, Jul. 2015.

[45] S. He, X. Gong, J. Zhang, J. Chen, and Y. Sun, "Curve-based deployment for barrier coverage in wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 5, no. 6, pp. 640–652, Jun. 2006.

[46] C. Wu, K. Lee, and Y. Chung, "A delaunay triangulation based method for wireless sensor network deployment," in *IEEE Int. Conf. on Parallel Distrib. Syst. (ICPADS)*, Jul. 2006.

[47] N. Aitsaadi, N. Achir, K. Bousseta, and B. Gavish, "A gradient approach for differentiated wireless sensor network deployment," in *2008 1st IFIP Wireless Days*, Nov. 2008.

[48] A. Ababnah and B. Natarajan, "LQR formulation of sensor deployment for decision fusion based detection," in *IEEE Global Telecommun. Conf. (GLOBECOM)*, Dec. 2010.

[49] S. Li, C. Xu, W. Pan, and Y. Pan, "Sensor deployment optimization for detecting maneuvering targets," in *IEEE Int. Conf. Inf. Fusion (ICIF)*, Jul. 2005.

[50] S. Temel, N. Unaldi, and O. Kaynak, "On deployment of wireless sensors on 3-D terrains to maximize sensing coverage by utilizing cat swarm optimization with wavelet transform," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 1, pp. 111–120, Jan. 2014.

[51] E. Onur, C. Ersoy, H. Delic, and L. Akarun, "Surveillance wireless sensor networks: Deployment quality analysis," *IEEE Netw.*, vol. 21, no. 6, pp. 48–53, Dec. 2007.

[52] X. Liu, "Coverage with connectivity in wireless sensor networks," in *IEEE 3rd Int. Conf. Broadband Commun., Netw. Syst.*, Oct. 2006.

[53] J. Guo and H. Jafarkhani, "Movement-efficient sensor deployment in wireless sensor networks," in *IEEE Int. Conf. Commun. (ICC)*, May 2018.

[54] J. Guo, E. Koyuncu, and H. Jafarkhani, "Energy efficiency in two-tiered wireless sensor networks," in *IEEE Int. Conf. Commun. (ICC)*, May 2017.

[55] J. Guo, P. Walk, and H. Jafarkhani, "Quantizers with parameterized distortion measures," *arxiv*, Nov. 2018. eprint: `1811.02554` (arxiv).

[56] H. J. H. Yousefi'zadeh and M. Moshfeghi, "Power optimization of wireless media systems with space-time code building blocks," *IEEE Trans. Image Process.*, vol. 13, no. 7, 873–884, Jul. 2004.

[57] G. Wang, M. J. Irwin, P. Berman, H. Fu, and T. L. Porta, "Optimizing sensor movement planning for energy efficiency," in *IEEE Int. Symp. Low Power Electron. Des.*, Aug. 2005.

[58] Y. Mei, Y. Lu, Y. C. Hu, and C. S. G. Lee, "Energy-efficient motion planning for mobile robots," in *IEEE Int. Conf. Robot. & Autom. (ICRA)*, Aug. 2004.

[59] R. M. Gray and D. L. Neuhoff, "Quantization," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2325–2383, Oct. 1998.

[60] G. Wang, G. Cao, and T. L. Porta, "Movement-assisted sensor deployment," *IEEE Trans. Mobile Comput.*, vol. 5, no. 6, pp. 640–652, Jun. 2006.

[61] O. Arslan and D. E. Koditschek, "Voronoi-based coverage control of heterogeneous disk-shaped robots," in *IEEE Int. Conf. Robot. (ICRA)*, May 2016.

[62] C. C. Coskun and E. Ayanoglu, "Energy-efficient base station deployment in heterogeneous networks," *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 593–596, Dec. 2014.

[63] T. Zhang, J. Zhao, L. An, and D. Liu, "Energy efficiency of base station deployment in ultra dense hetnets: A stochastic geometry analysis," *IEEE Wireless Commun. Lett.*, vol. 5, no. 2, pp. 184–187, Apr. 2016.

[64] H. Elsawy, E. Hossain, and M. Haenggi, "Stochastic geometry for modeling, analysis, and design of multi-tier and cognitive cellular wireless networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 996–1019, Jul. 2013.

[65] H. Jafarkhani, *Space-Time Coding: Theory and Practice*, 1st ed. Cambridge, United Kingdom: Cambridge Univ. Press, 2005.

[66] K. Akkaya and M. Younis, "A survey on routing protocols in wireless sensor networks," *Elsevier Ad Hoc Network*, vol. 3, no. 3, pp. 325–349, May 2005.

[67] L. Li and J. Y. Halpern, "Minimum-energy mobile wireless networks revisited," in *IEEE Int. Conf. Commun. (ICC)*, Jun. 2001.

[68] Y. T. Hou, Y. Shi, H. D. Sherali, and F. Midkiff, "On energy provisioning and relay node placement for wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 5, no. 5, pp. 2579–2590, Sep. 2005.

[69] W. Pawgasame, "A survey in adaptive hybrid wireless sensor network for military operations," in *IEEE 2th Asian Conf. Defence Technol. (ACDT)*, Jan. 2016.

[70] J.-H. Chang and L. Tassiulas, "Energy conserving routing in wireless ad-hoc networks," in *IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Mar. 2000.

[71] W. R. Bennett, "Spectra of quantized signals," *IEEE The Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 446–472, Jul. 1948.

[72] A. Gersho, "Asymptotically optimal block quantization," *IEEE Trans. Inf. Theory*, vol. 25, no. 4, pp. 373–380, Jul. 1979.

[73] P. L. Zador, "Asymptotic quantization error of continuous signals and the quantization dimension," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 139–148, Mar. 1982.

[74] E. Koyuncu and H. Jafarkhani, "On the minimum average distortion of quantizers with index-dependent distortion measures," *IEEE Trans. Signal Process.*, vol. 65, no. 17, pp. 4655–4669, Sep. 2017.

[75] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 1st ed. San Francisco, CA, USA: Morgan Kaufmann, 2011.

[76] H. Jafarkhani and V. Tarokh, "Design of successively refinable trellis coded quantizers," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1490–1497, Jul. 1999.

[77] H. Jafarkhani, H. Brunk, and N. Farvardin, "Entropy-constrained successively refinable scalar quantization," in *IEEE Data Compression Conf. (DCC)*, Mar. 1997.

[78] D. Mukherjee and S. K. Mitra, "Successive refinement lattice vector quantization," *IEEE Trans. Image Process.*, vol. 11, no. 12, pp. 1337–1348, Dec. 2002.

[79] N. Chaddha, P. A. Chou, and R. M. Gray, "Constrained and recursive hierarchical table-lookup vector quantization," in *IEEE Data Compression Conf. (DCC)*, Apr. 1996.

[80] A. Gersho and Y. Shoham, "Hierarchical vector quantization of speech with dynamic codebook allocation," in *IEEE Int. Conf. on Acoust., Speech, and Signal Process. (ICASSP)*, Mar. 1984.

[81] H. Jafarkhani and N. Farvardin, "Channel-matched hierarchical table-lookup vector quantization," *IEEE Trans. Inf. Theory*, vol. 46, no. 3, pp. 1121–1125, May 2000.

[82] S. S. Prakash and P. Niranjan, "Movement minimization of randomly deployed mobile nodes for complete coverage and connectivity," in *IEEE Int. Conf. Advanced Commun., Control Comput. Tech.*, May 2014.

[83] K. Shih, H. Chen, J. Tsai, and C. Li, "PALM: A partition avoidance lazy movement protocol for mobile sensor networks," in *IEEE Wireless Commun. Netw. Conf.*, Mar. 2007.

[84] S. Yang, M. Li, and J. Wu, "Scan-based movement-assisted sensor deployment methods in wireless sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 18, no. 8, pp. 1108–1121, Aug. 2007.

[85] S. Meguerdichian, F. Koushanfar, M. Potkinjak, and M. B. Srivastava, "Coverage problems in wireless ad-hoc sensor networks," in *IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr. 2001, pp. 1380–1387.

[86] M. Jin, G. Rong, H. Wu, L. Shuai, and X. Guo, "Optimal surface deployment problem in wireless sensor networks," in *IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Mar. 2012.

[87] L. Yang, J. Liang, and W. Liu, "Radar sensor (RS) deployment for multi-target detection," in *IEEE Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2014.

[88] S. Meguerdichian, F. Koushanfar, G. Qu, and M. Potkonjak, "Exposure in wireless ad-hoc sensor networks," in *ACM The 7th Annu. Int. Conf. Mobile Comput. Netw (MobiCom)*, Jul. 2001.

[89] V. Choudhary and K. R Chowdhary, "Probabilistic sensor deployment in wireless sensor network: A new approach," in *IEEE Int. Conf. Emerging Trends Netw Comput. Commun. (ETNCC)*, Apr. 2011.

[90] A. Gusrialdi and L. Zeng, "Distributed deployment algorithms for robotic visual sensor networks in non-convex environment," in *IEEE Int. Conf. Netw., Sens. Control (ICNSC)*, Apr. 2011.

[91] A. Breitenmoser, M. Schwager, J. C. Metzger, R. Siegwart, and D. Rus, "Voronoi coverage of non-convex environments with a group of networked robots," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2010.

[92] T. S. Helal, P. Mozumdar, and L. Akter, "Evaluating the performance of optimal control based sensor deployment algorithms for realistic terrain model," in *IEEE Int. Conf. Elect. Comput. Eng. (ICECE)*, Dec. 2014.

[93] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, 2nd ed. Boston, MA, USA: Kluwer, 1992.

[94] A. Gersho, "Asymptotically optimal block quantization," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, Mar. 1982.

[95] J. C. Kieffer, "Exponential rate of convergence for lloyd's method i," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 205–210, Mar. 1982.

[96] X. Wu, "On convergence of lloyd's method i," *IEEE Trans. Inf. Theory*, vol. 38, no. 1, pp. 171–174, Mar. 1982.

[97] Q. Du, M. Emelianenko, and L. Ju, "Convergence of the lloyd algorithm for computing centroidal voronoi tessellations," *Siam J. Numer. Anal.*, vol. 44, no. 1, pp. 102–119, Mar. 2006.

[98] J. Guo and H. Jafarkhani, "Sensor deployment in heterogeneous wireless sensor networks," in *IEEE Global Telecommun. Conf. (GLOBECOM)*, Dec. 2016.

[99] H. Yousefi'zadeh, H. Jafarkhani, and J. Kazemitabar, "Voronoi diagrams – a survey of fundamental geometric data structure," *ACM Comput. Surveys*, vol. 23, no. 3, pp. 345–405, Sep. 1991.

[100] S. Valeriu, *Introduction to the Axiomatic Theory of Convexity*, 1st ed. Russian: Stiinta, 1984.

[101] W. P. Soltan, *Abstract Convex Analysis*, 1st ed. New York, NY, USA: John Wiley & Sons, 1997.

[102] F. Aurenhammer and H. Edelsbrunner, "An optimal algorithm for constructing the weighted voronoi diagram in the plane," *Pattern Recognition*, vol. 17, no. 2, pp. 251–257, Dec. 1984.

[103] P. J. M. Laarhoven and E. H. L Aarts, *Simulated Annealing: Theory and Applications*, 1st ed. Norwell, MA, USA: Springer, 1987.

[104] S. Geman and D. Geman, "Stochastic relaxation, gibbs distortion and the bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Int.*, vol. 6, no. 6, pp. 721–741, Nov. 1984.

[105] E. Koyuncu and H. Jafarkhani, "On the minimum distortion of quantizers with heterogeneous reproduction points," in *IEEE Data Compression Conf. (DCC)*, Mar. 2016.

[106] B. Galkin, J. Kibilda, and L. A. DaSilva, "Backhaul for low-altitude UAVs in urban environments," in *IEEE Int. Conf. Commun. (ICC)*, May 2018.

[107] M. M. Azari, F. Rosas, and S. Pollin, "Reshaping cellular networks for the sky: Major factors and feasibility," *arxiv*, Oct. 2017.

[108] H. Shakhatreh, A. Khreishah, N. S. Othman, and A. Sawalmeh, "Maximizing indoor wireless coverage using UAVs equipped with directional antennas," in *IEEE 13th Malaysia Int. Conf. Commun. (MICC)*, Nov. 2017.

[109] K. Venugopal, M. C. Valenti, and R. W. Heath, "Device-to-device millimeter wave communications: Interference, coverage, rate, and finite topologies," *IEEE Trans. Wireless Commun.*, vol. 15, no. 9, pp. 6175–6188, Sep. 2016.

[110] H. He, S. Zhang, Y. Zeng, and R. Zhang, "Joint altitude and beamwidth optimization for UAV-enabled multiuser communications," *IEEE Commun. Lett.*, vol. 22, no. 2, Feb. 2018.

[111] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage," *IEEE Commun. Lett.*, vol. 20, no. 8, pp. 1647–1650, Aug. 2016.

[112] A. Goldsmith, *Wireless Communications*. Cambridge University Press, 2005.

[113] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Unmanned aerial vehicle with underlaid device-to-device communications: Performance and tradeoffs," *IEEE Trans. Wireless Commun.*, vol. 15, no. 6, pp. 3949–3963, Jun. 2016.

[114] A. Al-Hourani and K. Gomez, "Modeling Cellular-to-UAV path-loss for suburban environments," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 82–85, Feb. 2018.

[115] C. A. Balanis, *Antenna Theory: Analysis and Design*, 3rd ed. John Wiley & Sons, 2005.

[116] J.-D. Boissonnat, C. Wormser, and M. Yvinec, "Curved voronoi diagrams," in *Effective Computational Geometry for Curves and Surfaces*. Berlin, Heidelberg, Germany: Springer, 2007.

[117] J.-D. Boissonnat and M. I. Karavelas, "On the combinatorial complexity of euclidean voronoi cells and convex hulls of d-dimensional spheres," *INRIA*, Jul. 2002.

[118] X. M. Zhang and Y. M. Chu, "Convexity of the integral arithmetic mean of a convex function," *Rocky Mountain Journal of Mathematics*, vol. 40, no. 3, pp. 1061–1068, Jun. 2010.

[119] M. Skolnik, *Radar Handbook*, 1st ed. New York, NY, USA: McGraw-Hill, 2008.

[120] D. Bertsekas and R. Gallager, *Data Networks*, 1st ed. Englewood Cliffs, NJ, USA: McGraw-Hill, 1992.

[121] B. Paul, *Kinematics and Dynamics of Planar Machinery*, 1st ed. Aurora, IL, USA: Prentice-Hall, 1979.

[122] R. G. Gallager, P. A. Humblet, and P. M. Spira, "A distributed algorithm for minimum-weight spanning trees," *ACM Trans. Programming Languages Syst.*, vol. 5, no. 1, pp. 66–77, Jan. 1983.

[123] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes: The Art of Scientific Computing*, 3rd ed. New York, NY, USA: Cambridge University Press, 2007.

[124] K. O. Geddes, S. R. Czapor, and G. Labahn, *Algorithms for Computer Algebra*, 1st ed. Boston, MA, USA: Kluwer Academic Publishers, 1992.

[125] D. Zwillinger, *Standard Matehematical Tables and Formulae*, 31st ed. Boca Raton, FL, USA: CRC Press, 2003.

[126] J. H. Conway and N. J. A. Sloane, *Sphere packings, lattices and groups*, 1st ed. New York, NY, USA: Springer Science & Business Media, 1993.

Appendix A

Proofs

A.1 Proof of Proposition 1

At the beginning, it is necessary to study the shape of WVDs. Let $\boldsymbol{V}_{nk}^{H}(\mathbf{P}) = \{\boldsymbol{\omega}|\ \eta_n\|\boldsymbol{\omega} - \mathbf{p}_n\|^2 \leq \eta_k\|\boldsymbol{\omega} - \mathbf{p}_k\|^2\}$ be the pairwise Multiplicatively Weighted Voronoi Diagram (MWVD) of Sensor n when we only consider Sensors n and k. Then, the exact MWVD of Sensor n is the intersection of these pairwise WVDs, i.e., $\boldsymbol{V}_{n}^{H}(\mathbf{P}) = \bigcap_{k\neq n} \boldsymbol{V}_{nk}^{H}(\mathbf{P})$. We define the coordinates of $\boldsymbol{\omega} = (x, y)$, $\mathbf{p}_n = (\mathbf{p}_{nx}, \mathbf{p}_{ny})$ and $\mathbf{p}_k = (\mathbf{p}_{kx}, \mathbf{p}_{ky})$ and define $\eta = \sqrt{\eta_n/\eta_k} > 0$. Then, expanding the hyperplane equation $\eta_n\|\boldsymbol{\omega} - \mathbf{p}_n\|^2 = \eta_k\|\boldsymbol{\omega} - \mathbf{p}_k\|^2$ results in

$$(\eta^2 - 1)(x^2 + y^2) + 2(\mathbf{p}_{kx} - \eta^2\mathbf{p}_{nx})x + 2(\mathbf{p}_{ky} - \eta^2\mathbf{p}_{ny})y = \mathbf{p}_{kx}^2 - \eta^2\mathbf{p}_{nx}^2 + \mathbf{p}_{ky}^2 - \eta^2\mathbf{p}_{ny}^2. \quad \text{(A.1)}$$

When $\eta = 1$, the hyperplane equation is $2(\mathbf{p}_{kx}-\mathbf{p}_{nx})x+2(\mathbf{p}_{ky}-\mathbf{p}_{ny})y+\mathbf{p}_{nx}^2+\mathbf{p}_{ny}^2-\mathbf{p}_{kx}^2-\mathbf{p}_{ky}^2 = 0$. This hyperplane is the boundary of the half space $HS(\mathbf{p}_n, \mathbf{p}_k)$. When $\eta < 1$, $\boldsymbol{V}_{nk}^{H}$ is defined by

$$(\boldsymbol{\omega} - \mathbf{c}_{nk})^2 \geq r_{nk}^2. \quad \text{(A.2)}$$

When $\eta > 1$, Part_{nk} is defined by

$$(\boldsymbol{\omega} - \mathbf{c}_{nk})^2 \leq r_{nk}^2, \tag{A.3}$$

where $\mathbf{c}_{nk} = (\frac{\mathbf{p}_{kx} - \eta^2 \mathbf{p}_{nx}}{1-\eta^2}, \frac{\mathbf{p}_{ky} - \eta^2 \mathbf{p}_{ny}}{1-\eta^2})$, $r_{nk} = \frac{\eta}{|1-\eta^2|} \|\mathbf{p}_n - \mathbf{p}_k\|$. Therefore, we have

$$\begin{aligned} \boldsymbol{V}_n^H(\mathbf{P}) &= \{\boldsymbol{\omega} \in \Omega | \eta_n \|\boldsymbol{\omega} - \mathbf{p}_n\| \leq \eta_l \|\boldsymbol{\omega} - \mathbf{p}_l\|^2, \forall l \in [N]\} \\ &= \left[\bigcap_{i:\eta_i < \eta_n} \mathbb{B}(\mathbf{c}_{in}, r_{in})\right] \bigcap \left[\bigcap_{k:\eta_k = \eta_n} HS(p_n, p_k)\right] \bigcap \left[\bigcap_{j:\eta_j > \eta_n} [\mathbb{B}(\mathbf{c}_{nj}, r_{nj})]^c\right] \\ &= \left[\bigcap_{i:\eta_i < \eta_n} \mathbb{B}(\mathbf{c}_{in}, r_{in})\right] \bigcap \left[\bigcap_{k:\eta_k = \eta_n} HS(p_n, p_k)\right] \bigcap \left[\bigcup_{j:\eta_j > \eta_n} \mathbb{B}(\mathbf{c}_{nj}, r_{nj})\right]^c, \end{aligned} \tag{A.4}$$

where A^c denotes the complementary set of A.

Let $\bar{\boldsymbol{V}}_n^H(\mathbf{P})$ be the MWVD of Sensor n when we ignore sensors with larger cost parameters. Accordingly, $\bar{\boldsymbol{V}}_n^H(\mathbf{P})$ is defined by $\bar{\boldsymbol{V}}_n^H(\mathbf{P}) = \{\boldsymbol{\omega} \in \Omega | \eta_n \|\boldsymbol{\omega} - \mathbf{p}_n\|^2 \leq \eta_l \|\boldsymbol{\omega} - \mathbf{p}_l\|^2, \forall l, \eta_n \geq \eta_l\}$. Review the definition of $\boldsymbol{V}_n^H(\mathbf{P})$ in (2.4), we have

$$\begin{aligned} \boldsymbol{V}_n^H(\mathbf{P}) = \{\boldsymbol{\omega} \in \Omega |\ & \eta_n \|\boldsymbol{\omega} - \mathbf{p}_n\|^2 \leq \eta_i \|\boldsymbol{\omega} - \mathbf{p}_i\|^2, \eta_n \|\boldsymbol{\omega} - \mathbf{p}_n\|^2 \leq \eta_j \|\boldsymbol{\omega} - \mathbf{p}_j\|^2, \\ & \forall i, j, \eta_j > \eta_n, \eta_n \geq \eta_i\} \end{aligned} \tag{A.5}$$

According to (A.4), $\bar{\boldsymbol{V}}_n^H(\mathbf{P})$ is the intersection of convex regions and therefore star-shaped. The relationship between $\bar{\boldsymbol{V}}_n^H(\mathbf{P})$ and $\boldsymbol{V}_n^H(\mathbf{P})$ is

$$\bar{\boldsymbol{V}}_n^H(\mathbf{P}) = \boldsymbol{V}_n^H(\mathbf{P}) \bigcup W_n, \tag{A.6}$$

where $W_n = \{\boldsymbol{\omega} \in \Omega |\ \eta_n \|\boldsymbol{\omega} - \mathbf{p}_n\|^2 \geq \eta_j \|\boldsymbol{\omega} - \mathbf{p}_j\|^2, \eta_n \|\boldsymbol{\omega} - \mathbf{p}_n\|^2 \leq \eta_i \|\boldsymbol{\omega} - \mathbf{p}_i\|^2, \forall i, j, \eta_j > \eta_n, \eta_n \geq \eta_i\}$. For any point $\boldsymbol{\omega}$ such that $\eta_j \|\boldsymbol{\omega} - \mathbf{p}_j\|^2 \leq \eta_n \|\boldsymbol{\omega} - \mathbf{p}_n\|^2$ and $\eta_n \|\boldsymbol{\omega} - \mathbf{p}_n\|^2 \leq \eta_i \|\boldsymbol{\omega} - \mathbf{p}_i\|^2$, we have $\eta_j \|\boldsymbol{\omega} - \mathbf{p}_j\|^2 \leq \eta_i \|\boldsymbol{\omega} - \mathbf{p}_i\|^2$. So we can rewrite W_n as

$$\begin{aligned} W_n = \{\boldsymbol{\omega} \in \Omega |\ \eta_j \|\boldsymbol{\omega} - \mathbf{p}_j\|^2 \leq \eta_i \|\boldsymbol{\omega} - \mathbf{p}_i\|^2, \eta_j \|\boldsymbol{\omega} - \mathbf{p}_j\|^2 \leq \eta_n \|\boldsymbol{\omega} - \mathbf{p}_n\|^2, \\ \eta_n \|\boldsymbol{\omega} - \mathbf{p}_n\|^2 \leq \eta_i \|\boldsymbol{\omega} - \mathbf{p}_i\|^2 \text{ for any } i, j \text{ such that } \eta_j > \eta_n, \eta_n \geq \eta_i\}. \end{aligned} \tag{A.7}$$

Due to the definitions of $\bar{\boldsymbol{V}}_n^H(\mathbf{P})$ and $\boldsymbol{V}_n^H(\mathbf{P})$, we have

$$W_n = \left[\bigcup_{j:\eta_j > \eta_n} \boldsymbol{V}_j^H(\mathbf{P})\right] \bigcap \bar{\boldsymbol{V}}_n^H(\mathbf{P}) = \bigcup_{j:\eta_j > \eta_n} \left[\boldsymbol{V}_j^H(\mathbf{P}) \bigcap \bar{\boldsymbol{V}}_n^H(\mathbf{P})\right] \tag{A.8}$$

Replacing W_n from (A.8) in (A.6), we get the final result

$$\bar{\boldsymbol{V}}_n^H(\mathbf{P}) = \boldsymbol{V}_n^H(\mathbf{P}) \bigcup_{j:\eta_j > \eta_n} \left[\boldsymbol{V}_j^H(\mathbf{P}) \bigcap \bar{\boldsymbol{V}}_n^H(\mathbf{P})\right] \tag{A.9}$$

The elements in the right side are disjoint subsets. Without loss of generality, let us assume that the M disjoint cost parameters are ordered such that the k-level cost parameter is larger than the $k+1$-level cost parameter. We also call the set including the indices of all sensors with a k-level cost parameter Z_k. Then:

(1) For any level-1 sensor n, $\boldsymbol{V}_n^H(\mathbf{P})$ is a convex set. Accordingly, the intersection $\boldsymbol{V}_n^H(\mathbf{P}) \bigcap W$ is a convex set and therefore (2.17) holds due to Proposition 1.6 in [6].

(2) Assume the equation holds for any sensor whose level is smaller than or equal to k. Consider a sensor n whose level is $k+1$, by using the relationship between $\bar{\boldsymbol{V}}_n^H(\mathbf{P})$ and $\boldsymbol{V}_n^H(\mathbf{P})$ in (A.9), we rewrite the objective function as

$$\int_{\boldsymbol{V}_n^H(\mathbf{P}) \bigcap W} \varphi(\boldsymbol{\omega}, \mathbf{p}_n) d\boldsymbol{\omega} = \int_{\bar{\boldsymbol{V}}_n^H(\mathbf{P}) \bigcap W} \varphi(\boldsymbol{\omega}, \mathbf{p}_n) d\boldsymbol{\omega} - \sum_{l=1}^{k} \sum_{t \in Z_l} \left[\int_{\boldsymbol{V}_t^H(\mathbf{P}) \bigcap \bar{\boldsymbol{V}}_n^H(\mathbf{P}) \bigcap W} \varphi(\boldsymbol{\omega}, \mathbf{p}_n) d\boldsymbol{\omega}\right]. \tag{A.10}$$

Since $\boldsymbol{V}_n^H(\mathbf{P}) \bigcap W$ is a convex set and therefore star-shaped, the partial derivative of the first term can be solved by proposition 1.6 in [6]. Sensors' levels in the second term are smaller than k and thus the partial derivative of the second term can be solved by our assumption in Step (2). Therefore, for $n, m \in [N]$, the partial derivative becomes

$$\begin{aligned}
&\frac{\partial \int_{\boldsymbol{V}_n^H(\mathbf{P}) \bigcap W} \varphi_n(\boldsymbol{\omega}, \mathbf{p}_n) d\boldsymbol{\omega}}{\partial \mathbf{p}_m} \\
&= \int_{\bar{\boldsymbol{V}}_n^H(\mathbf{P}) \bigcap W} \frac{\partial \varphi_n(\boldsymbol{\omega}, \mathbf{p}_n)}{\partial \mathbf{p}_m} d\boldsymbol{\omega} + \int_{\partial\left[\bar{\boldsymbol{V}}_n^H(\mathbf{P}) \bigcap W\right]} \varphi_n(\gamma, \mathbf{p}_n) \mathbf{n}^t(\gamma) \frac{\partial \gamma}{\partial \mathbf{p}_m} d\gamma \\
&- \sum_{l=1}^{k} \sum_{t \in Z_l} \left[\int_{\boldsymbol{V}_t^H(\mathbf{P}) \bigcap \bar{\boldsymbol{V}}_n^H(\mathbf{P}) \bigcap W} \frac{\partial \varphi_n(\boldsymbol{\omega}, \mathbf{p}_n)}{\partial \mathbf{p}_m} d\boldsymbol{\omega} \right] \\
&- \sum_{l=1}^{k} \sum_{t \in Z_l} \left[\int_{\partial[\boldsymbol{V}_t^H(\mathbf{P}) \bigcap \bar{\boldsymbol{V}}_n^H(\mathbf{P}) \bigcap W]} \varphi_n(\gamma, \mathbf{p}_n) \mathbf{n}^t(\gamma) \frac{\partial \gamma}{\partial \mathbf{p}_m} d\gamma \right].
\end{aligned} \tag{A.11}$$

Note that $\bar{\boldsymbol{V}}_n^H(\mathbf{P}) \bigcap W = \left[\boldsymbol{V}_n^H(\mathbf{P}) \bigcap W\right] \bigcup G(\mathbf{P})$, where $G(\mathbf{P}) = \left[\bigcup_{l=1}^{k} \bigcup_{t \in Z_l} (\boldsymbol{V}_t^H(\mathbf{P}) \bigcap \boldsymbol{V}_n^H(\mathbf{P}) \bigcap W)\right]$, is a star-shaped set consisting of several disjoint subsets. By using Lemma 4, we have

$$\begin{aligned}
&\int_{\partial\left[\bar{\boldsymbol{V}}_n^H(\mathbf{P}) \bigcap W\right]} \varphi_n(\gamma, \mathbf{p}_n) \mathbf{n}^t(\gamma) \frac{\partial \gamma}{\partial \mathbf{p}_m} d\gamma - \sum_{l=1}^{k} \sum_{t \in Z_l} \left[\int_{\partial[\boldsymbol{V}_t^H(\mathbf{P}) \bigcap \bar{\boldsymbol{V}}_n^H(\mathbf{P}) \bigcap W]} \varphi_n(\gamma, \mathbf{p}_n) \mathbf{n}^t(\gamma) \frac{\partial \gamma}{\partial \mathbf{p}_m} d\gamma \right] \\
&= \int_{\partial[\boldsymbol{V}_n^H(\mathbf{P}) \bigcap W]} \varphi_n(\gamma, \mathbf{p}_n) \mathbf{n}^t(\gamma) \frac{\partial \gamma}{\partial \mathbf{p}_m} d\gamma.
\end{aligned} \tag{A.12}$$

Also, we have

$$\int_{\bar{\boldsymbol{V}}_n^H(\mathbf{P}) \bigcap W} \frac{\partial \varphi_n(\boldsymbol{\omega}, \mathbf{p}_n)}{\partial \mathbf{p}_m} d\boldsymbol{\omega} - \sum_{l=1}^{k} \sum_{t \in Z_l} \left[\int_{\boldsymbol{V}_t^H(\mathbf{P}) \bigcap \bar{\boldsymbol{V}}_n^H(\mathbf{P}) \bigcap W} \frac{\partial \varphi_n(\boldsymbol{\omega}, \mathbf{p}_n)}{\partial \mathbf{p}_m} d\boldsymbol{\omega} \right] = \int_{\boldsymbol{V}_n^H(\mathbf{P}) \bigcap W} \frac{\partial \varphi_n(\boldsymbol{\omega}, \mathbf{p}_n)}{\partial \mathbf{p}_m} d\boldsymbol{\omega}. \tag{A.13}$$

(2.17) is derived by replacing (A.12) and (A.13) in (A.11). In other words, (2.17) is correct for sensors whose level is smaller than or equal to $k+1$. In summary, (2.17) is correct for sensors in all levels of heterogeneous WSNs.

A.2 Proof of Lemma 6

To find the optimal 1−level parameter quantizer (x^*, h^*) for a uniform density $f(\boldsymbol{\omega}) = 1/A$, we need to satisfy (3.26), i.e., for[1] $\Omega = \boldsymbol{V}_1 = \boldsymbol{V}_1^* = [0, A]$

$$0 = \int_0^A (x^* - \boldsymbol{\omega})\big((x^* - \boldsymbol{\omega})^2 + h^{*2}\big)^{\gamma-1} d\boldsymbol{\omega}. \tag{A.14}$$

Substituting $x^* - \boldsymbol{\omega}$ by $\boldsymbol{\omega}$ we get

$$0 = \int_{x^*-A}^{x^*} \boldsymbol{\omega}\big(\boldsymbol{\omega}^2 + h^{*2}\big)^{\gamma-1} d\boldsymbol{\omega}. \tag{A.15}$$

Since the integral kernel is an odd function in $\boldsymbol{\omega}$ and $x^* \in [0, A]$, it must hold

$$0 = -\int_0^{x^*-A} \boldsymbol{\omega}(\boldsymbol{\omega}^2 + h^{*2})^{\gamma-1} d\boldsymbol{\omega} + \int_0^{x^*} \boldsymbol{\omega}(\boldsymbol{\omega}^2 + h^{*2})^{\gamma-1} d\boldsymbol{\omega} \tag{A.16}$$

by substituting $\boldsymbol{\omega}$ by $-\boldsymbol{\omega}$ we get

$$\int_0^{A-x^*} \boldsymbol{\omega}(\boldsymbol{\omega}^2 + h^{*2})^{\gamma-1} d\boldsymbol{\omega} = \int_0^{x^*} \boldsymbol{\omega}(\boldsymbol{\omega}^2 + h^{*2})^{\gamma-1} d\boldsymbol{\omega}. \tag{A.17}$$

Hence for any choice of h^* it must hold $x^* = A - x^*$, which is equivalent to $x^* = A/2$. To find the optimal parameter, we can just insert x^* into the average distortion

$$\overline{\mathcal{D}}(x^*, h) = \frac{1}{A}\int_0^A \frac{(x^* - \boldsymbol{\omega})^2 + h^2)^{\gamma}}{h} d\boldsymbol{\omega} = \frac{1}{A}\int_0^{A/2} \frac{(\boldsymbol{\omega}^2 + h^2)^{\gamma}}{h} d\boldsymbol{\omega} \tag{A.18}$$

where we substituted again and inserted $x^* = A/2$. By substituting $\boldsymbol{\omega}$ with $2\boldsymbol{\omega}/A$ and h with $u = 2h/A$ we get

$$= \int_0^1 \frac{2}{A}\frac{((A\boldsymbol{\omega}/2)^2 + (Au/2)^2)^{\gamma}}{u} d\boldsymbol{\omega} = \left(\frac{A}{2}\right)^{2\gamma-1} \int_0^1 \psi(\boldsymbol{\omega}, u, \gamma) d\boldsymbol{\omega} \tag{A.19}$$

[1] Note, there is no optimizing over the regions, since there is only one.

where for each $\gamma \geq 1$ the integral kernel ψ is a convex function in $\mathbf{x} = (\boldsymbol{\omega}, u)$ over $\mathbb{R}^2_+$. Let us rewrite f as

$$\psi(\boldsymbol{\omega}, u, \gamma) = \frac{(\boldsymbol{\omega}^2 + u^2)^\gamma}{u} = \frac{\|(\boldsymbol{\omega}, u)\|_2^{2\gamma}}{u}. \tag{A.20}$$

Clearly, $\|\mathbf{x}\|_2$ is a convex and continuous function in $\mathbf{x}$ over $\mathbb{R}^2$ and since $(\cdot)^{2\gamma}$ with $2\gamma \geq 2$ is a strictly increasing continuous function, the concatenation $\psi(\mathbf{x}, \gamma)$ is a strict convex and continuous function over $\mathbb{R}^2_+$. Hence, for any $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^2$ we have

$$\|\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2\|_2^{2\gamma} < \lambda \|\mathbf{x}_1\|_2^{2\gamma} + (1-\lambda)\|\mathbf{x}_2\|_2^{2\gamma} \tag{A.21}$$

for all $\lambda \in (0,1)$. But then we have also for any $u_1, u_2 \in \mathbb{R}^2_+$ and $\boldsymbol{\omega} \geq 0$

$$\psi(\lambda u_1 + (1-\lambda)u_2, \boldsymbol{\omega}, \gamma) < \frac{\lambda \|(\boldsymbol{\omega}, u_1)\|_2^{2\gamma} + (1-\lambda)\|(\boldsymbol{\omega}, u_2)\|_2^{2\gamma}}{\lambda u_1 + (1-\lambda)u_2}. \tag{A.22}$$

Considering the following inequality

$$\frac{1}{u_1} + \frac{1}{u_2} = \left(\frac{1}{u_1} + \frac{1}{u_2}\right) \frac{\lambda u_1 + (1-\lambda)u_2}{\lambda u_1 + (1-\lambda)u_2} = \frac{\left(\lambda + \frac{(1-\lambda)u_2}{u_1} + (1-\lambda) + \frac{\lambda u_1}{u_2}\right)}{\lambda u_1 + (1-\lambda)u_2} > \frac{1}{\lambda u_1 + (1-\lambda)u_2}$$

and (A.22), we will have

$$\psi(\lambda u_1 + (1-\lambda)u_2, \boldsymbol{\omega}, \gamma) < \lambda\psi(u_1, \boldsymbol{\omega}, \gamma) + (1-\lambda)\psi(u_2, \boldsymbol{\omega}, \gamma) \tag{A.23}$$

for every $\lambda \in (0,1)$. Hence, the integral kernel is a strictly convex function for every $\boldsymbol{\omega} \geq 0, \gamma \geq 1$, and since the infinite sum (integral) of convex functions is again a convex function, for $u > 0$, we have shown convexity of $\Psi(u, \gamma)$. Note, $\psi(u, \boldsymbol{\omega}, \gamma)$ is continuous in $\mathbb{R}^2_+$ since it is a product of the continuous functions $\|(u, \boldsymbol{\omega})\|_2^{2\gamma}$ and $1/(u + 0 \cdot \boldsymbol{\omega})$, and so is $\Psi(u, \gamma)$.

Therefore, the only critical point of $\Psi(\cdot,\gamma)$ will be the unique global minimizer

$$g(\gamma) = \arg\min_{u>0} \Psi(u,\gamma), \tag{A.24}$$

which is defined by the vanishing of the first derivative:

$$\Psi'(u) = \int_0^1 (\boldsymbol{\omega}^2+u^2)^{\gamma-1} \left((2\gamma-1) - \frac{\boldsymbol{\omega}^2}{u^2} \right) d\boldsymbol{\omega} = \frac{1}{u^2} \int_0^1 (\boldsymbol{\omega}^2+u^2)^{\gamma-1} \left((2\gamma-1)u^2 - \boldsymbol{\omega}^2 \right) d\boldsymbol{\omega}. \tag{A.25}$$

Hence, $\Psi'(u)$ can only vanish if $u < 1/\sqrt{2\gamma-1}$, which is an upper bound on $g(\gamma)$. The optimal parameter for minimizing the average distortion (A.18) is then

$$h^* = \frac{A}{2} g(\gamma) \quad \text{with} \quad \overline{\mathcal{D}}(x^*, h^*) = \left(\frac{A}{2} \right)^{2\gamma-1} g(\gamma). \tag{A.26}$$

Analytical solutions for $\Psi'(u) = 0$ are possible for integer valued γ. Let us set $0 < x = u^2$ in (A.25), then for $\gamma \in \mathbb{N}$, the integrand in (A.25) will be a polynomial in Ω of degree 2γ and in x of degree γ. For $\gamma \in \{1,2,3\}$ the integrand will be

$$(\boldsymbol{\omega}^2 + x)^0 (1x - \boldsymbol{\omega}^2) = x - \boldsymbol{\omega}^2 \tag{A.27}$$

$$(\boldsymbol{\omega}^2 + x)^1 (3x - \boldsymbol{\omega}^2) = 3x^2 + 2\boldsymbol{\omega}^2 x - \boldsymbol{\omega}^4 \tag{A.28}$$

$$(\boldsymbol{\omega}^2 + x)^2 (5x - \boldsymbol{\omega}^2) = 5x^3 + 9\boldsymbol{\omega}^2 x^2 + 3\boldsymbol{\omega}^4 x - \boldsymbol{\omega}^6 \tag{A.29}$$

which yield with the definite integrals to

$$0 = \boldsymbol{\omega}(x - \frac{\boldsymbol{\omega}^2}{3}) \Big|_{\Omega=1} \tag{A.30}$$

$$0 = \boldsymbol{\omega}(3x^2 + \frac{2\boldsymbol{\omega}^2 x}{3} - \frac{\boldsymbol{\omega}^4}{5}) \Big|_{\Omega=1} \tag{A.31}$$

$$0 = \boldsymbol{\omega}(5x^3 + 3\boldsymbol{\omega}^2 x^2 + \frac{3\boldsymbol{\omega}^4 x}{5} - \frac{\boldsymbol{\omega}^6}{7}) \Big|_{\Omega=1} \tag{A.32}$$

Solving (A.30) for x yields to the only feasible solution

$$x = \frac{1}{3} \quad \Rightarrow \quad g(1) = \frac{1}{\sqrt{3}} \approx 0.577. \tag{A.33}$$

The solutions of (A.31) are

$$x_{\pm} = -\frac{1}{9} \pm \sqrt{\frac{1}{81} + \frac{1}{15}} = \frac{\pm\sqrt{32/5} - 1}{9} \tag{A.34}$$

Since only positive roots are allowed, we get as the only feasible solution

$$g(3) = \frac{\sqrt{\sqrt{32/5} - 1}}{3} \approx 0.412. \tag{A.35}$$

Finally, the cubic equation (A.32) results in

$$5x^3 + 3x^2 + \frac{3}{5}x - \frac{1}{7} = 0 \tag{A.36}$$

The solution of a cubic equation can be found in [125, p. 2.3.2] by calculating the discriminant

$$\Delta = q^2 + 4p^3 \quad \text{with} \quad q = \frac{2b^3 - 9abc + 27a^2d}{27a^3}, p = \frac{3ac - b^2}{9a^2} \tag{A.37}$$

Let us identify $a = 5, b = 3, c = 3/5$ and $d = -1/7$, then we get

$$q = \frac{6 \cdot 9 - 9 \cdot 9 - 27 \cdot 5^2 \cdot 1/7}{27 \cdot 5^3} = -\frac{3}{3 \cdot 5 \cdot 25} - \frac{1}{5 \cdot 7} = -\frac{32}{25 \cdot 35} \tag{A.38}$$

$$\Delta = q^2 + 4\left(\frac{3 \cdot 3 - 9}{9 \cdot 5^2}\right)^3 = q^2 > 0 \tag{A.39}$$

which indicates only one real-valued root, given by

$$x = \alpha_+^{1/3} + \alpha_-^{1/3} - \frac{b}{3a} \quad \text{with} \quad \alpha_\pm = \frac{-q \pm \sqrt{\Delta}}{2} = \left\{0, \frac{32}{25 \cdot 35}\right\} \tag{A.40}$$

which computes to

$$x = \left(\frac{32}{5^3 \cdot 7}\right)^{1/3} - \frac{1}{5} = \frac{(\frac{32}{7})^{1/3} - 1}{5} \Rightarrow g(5) = \sqrt{\frac{(\frac{32}{7})^{1/3} - 1}{5}} \approx 0.363. \tag{A.41}$$

A.3 Proof of Lemma 7

Although, this statement seems to be trivial, it is not straight forward to show. We will use the quantization relaxation for the average distortion $\overline{\mathcal{D}}$ in (3.9) to show that the $N-$level parameter optimized quantizer has strictly smaller distortion than the $(N-1)-$level optimized quantizer (3.10). We define, as in quantization theory, see for example [59], an *$N-$level quantizer* for Ω, by a (disjoint) partition $\mathcal{R} = \{\mathcal{R}_n\}_{n=1}^N \subset \Omega$ of Ω and assign to each partition region $\mathcal{R}_n$ a quantization-parameter point $(\mathbf{q}_n, h_n) \in \Omega \times \mathbb{R}_+$. The assignment rule or *quantization rule* can be anything such that the regions are independent of the value of the quantization and parameter points. Minimizing over the quantizer, that is, over all partitions and possible quantization-parameter points will yield to the parameter optimized quantizer, which is by definition the optimal deployment which generate the generalized Voronoi regions as the optimal partition (tessellation[2]). This holds for any density function

[2]Since we take here the continuous case, the integral will not distinguish between open or closed sets.

$f(\boldsymbol{\omega})$ and target area Ω. To see this[3], let us start with any quantizer $(\mathbf{Q}, \mathbf{h}, \mathcal{R})$ for Ω yielding to the average distortion

$$\begin{aligned}\overline{\mathcal{D}}(\mathbf{Q}, \mathbf{h}, \mathcal{R}) &= \sum_{n=1}^{N} \int_{\mathcal{R}_n} \mathcal{D}(\mathbf{q}_n, h_n, \boldsymbol{\omega}) f(\boldsymbol{\omega}) d\boldsymbol{\omega} \geq \sum_{n=1}^{N} \int_{\mathcal{R}_n} \Big(\min_{m \in [N]} \mathcal{D}(\mathbf{q}_m, h_n, \boldsymbol{\omega}) \Big) f(\boldsymbol{\omega}) d\boldsymbol{\omega} \\ &= \int_{\Omega} \min_{m \in [N]} \mathcal{D}(\mathbf{q}_m, h_n, \boldsymbol{\omega}) f(\boldsymbol{\omega}) d\boldsymbol{\omega} = \sum_{n=1}^{N} \int_{\boldsymbol{V}_n(\mathbf{Q}, \mathbf{h})} \mathcal{D}(\mathbf{q}_n, h_n, \boldsymbol{\omega}) f(\boldsymbol{\omega}) d\boldsymbol{\omega} \quad \text{(A.42)}\end{aligned}$$

where the first inequality is only achieved if for any $\boldsymbol{\omega} \in \mathcal{R}_n$ we have chosen $(\mathbf{q}_n, h_n)$ to be the optimal quantization point with respect to $\mathcal{D}$, or vice versa, if every $(\mathbf{q}, h_n)$ is optimal for every $\boldsymbol{\omega} \in \mathcal{R}_n$, which is the definition of the generalized Voronoi region $\boldsymbol{V}(\mathbf{Q}, \mathbf{h})$. Therefore, minimizing over all partitions gives equality, i.e.

$$\min_{\mathcal{R}} \overline{\mathcal{D}}(\mathbf{Q}, \mathbf{h}, \mathcal{R}) = \overline{\mathcal{D}}(\mathbf{Q}, \mathbf{h}, \boldsymbol{V}(\mathbf{Q}, \mathbf{h})) \quad \text{(A.43)}$$

for any $(\mathbf{Q}, \mathbf{h}) \in \Omega^N \times \mathbb{R}_+^N$. Hence, we have shown that the parameterized distortion quantizer optimization problem is equivalent to the locational-parameter optimization problem

$$\min_{\mathbf{Q} \in \Omega^N, \mathbf{h} \in \mathbb{R}_+^N} \min_{\mathcal{R} \in \Omega^N} \overline{\mathcal{D}}(\mathbf{Q}, \mathbf{h}, \mathcal{R}) = \min_{\mathbf{Q} \in \Omega^N, \mathbf{h} \in \mathbb{R}_+^N} \overline{\mathcal{D}}(\mathbf{Q}, \mathbf{h}, \boldsymbol{V}(\mathbf{Q}, \mathbf{h})) = \overline{\mathcal{D}}(\mathbf{Q}^*, \mathbf{h}^*, \boldsymbol{V}^*). \quad \text{(A.44)}$$

We need to show that for the optimal $N-$level parameter-quantizer $(\mathbf{Q}^*, \mathbf{h}^*, \boldsymbol{V}^*)$ with $\boldsymbol{V}^* = \boldsymbol{V}(\mathbf{Q}^*, \mathbf{h}^*)$, we have $\mu(\boldsymbol{V}_n) > 0$ for all $n \in [N]$. Let us first show that each region is indeed a closed interval, i.e., $\boldsymbol{V}_n^* = [b_{n-1}^*, b_n^*]$ with $0 \leq b_{n-1}^* \leq b_n^* \leq A$. By the definition of the Mbius regions in Lemma 5, each dominance region is either a single interval (if it is a ball not contained in the target region or a halfspace) or two disjoint intervals (if its a ball contained in the target region), we can not have more than $K_n \leq 2N - 2$ disjoint closed intervals for each Mbius (generalized Voronoi) region. Therefore, the nth optimal Mbius region is given as $\boldsymbol{V}_n^* = \bigcup_{k=1}^{K_n} v_{n,k}$, where $v_{n,k} = [a_{n,k-1}, a_{n,k}]$ are intervals for some $0 \leq a_{n,k-1} \leq a_{n,k} \leq A$.

[3]We use the same argumentation as in the prove of [34, Prop.1].

Let us assume there are quantization points with disconnected regions, i.e. $K_n > 1$ for $n \in \mathcal{I}_d$ and some $\mathcal{I}_d \subset [N]$. Then, we will re-arrange the partition $\boldsymbol{V}^*$ by concatenating the K_n disconnected intervals $v_{n,k}$ to $\mathcal{R}_n = [b_{n-1}, b_n]$ for $n \in \mathcal{I}_d$ and move the connected regions appropriatly such that for all $n \in [N]$ it holds $\mu(\mathcal{R}_n) = \mu(\boldsymbol{V}_n^*) = b_n - b_{n-1}$ and $b_{n-1} \leq b_n$, where we set $b_0 = 0$ and $b_N = A$. For the new concatenated regions, we move each q_n^* to the center of the new arranged regions, i.e., $\tilde{q}_n = \frac{b_n + b_{n-1}}{2}$ for $n \in \mathcal{I}_d$. If for the connected regions $n \in [N] \setminus \mathcal{I}_d$, the quantization point q_n^* is not centroidal, by placing it at the center of the corresponding closed interval, we will obtain a strictly smaller distortion by Lemma 6. Hence, for the optimal quantizer, the quantization points must be centroidal and we can assume $\tilde{q}_n = (b_n + b_{n-1})/2$ for all $n \in [N]$. In this rearrangement, we did not change the parameters h_n^* at all. The rearranged partition $\mathcal{R} = \{\mathcal{R}_n\}$ and replaced quantization points $\tilde{\mathbf{q}} = (\tilde{q}_1, \ldots, \tilde{q}_N)$ provide the average distortion

$$\overline{\mathcal{D}}(\tilde{\mathbf{q}}, \mathbf{h}^*, \mathcal{R}) = \sum_{n=1}^{N} \int_{b_{n-1}}^{b_n} \frac{((\tilde{q}_n - \boldsymbol{\omega})^2 + h_n^{*2})^\gamma}{h_n^*} d\boldsymbol{\omega} = 2 \sum_{n=1}^{N} \int_0^{\frac{b_n - b_{n-1}}{2}} \frac{(\boldsymbol{\omega}^2 + h_n^{*2})^\gamma}{h_n^*} d\boldsymbol{\omega} \quad \text{(A.45)}$$

where we substituted $\boldsymbol{\omega}$ by $\tilde{q}_n - \boldsymbol{\omega}$. Since the function $(\boldsymbol{\omega}^2 + h_n^{*2})^\gamma$ is strictly monotone increasing in $\boldsymbol{\omega}$ for each $\gamma > 0$, for any $n \in \mathcal{I}_d$, we have

$$\overline{\mathcal{D}}_n(\tilde{q}_n, h_n^*, \mathcal{R}_n) = 2 \int_0^{\frac{b_n - b_{n-1}}{2}} \frac{(\boldsymbol{\omega}^2 + h_n^{*2})^\gamma}{h_n^*} d\boldsymbol{\omega} < \sum_{k=1}^{K_n} \int_{a_{n,k} - q_n^*}^{a_{n,k-1} - q_n^*} \frac{(\boldsymbol{\omega}^2 + h_n^{*2})^\gamma}{h_n^*} d\boldsymbol{\omega} \quad \text{(A.46)}$$

since the non-zero gaps in $\bigcup_k [a_{n,k} - q_n^*, a_{n,k-1} - q_n^*]$ will lead to larger $\boldsymbol{\omega}$ in the RHS integral and therefore to a strictly larger average distortion. Therefore, the points $(\tilde{\mathbf{q}}, \mathbf{h}^*)$ with closed intervals $\{\mathcal{R}_n\}$ have a strictly smaller average distortion, which contradicts the assumption that $(\mathbf{q}^*, \mathbf{h}^*)$ is the parameter-optimized quantizer (3.11). Hence, $K_n = 1$ for each $n \in [N]$ and every $\gamma \geq 1$. Moreover, the optimal quantization points must be centroids of the intervals, i.e. $x_n^* = (b_n^* + b_{n-1}^*)/2$.

Now, we have to show that the optimal quantization regions $\boldsymbol{V}_n^* = \{[b_{n-1}^*, b_n^*]\}_{n=1}^N$ are not points, i.e., it should hold $b_n^* > b_{n-1}^*$ for each $n \in [N]$. If $b_n^* = b_{n-1}^*$ for some n, then the nth average distortion $\overline{\mathcal{D}}_n$ will be zero for this quantization point, since the integral is vanishing. But, then we only optimize over $N-1$ quantization points. So we only need to show that an additional quantization point strictly decreases the minimum average distortion. Hence, take any non-zero optimal quantization region $\boldsymbol{V}_n^* = [b_{n-1}^*, b_n^*]$. We know by Lemma 6 that the optimal quantizer q_n^* for some closed interval $\boldsymbol{V}_n^*$ must be centroidal for any parameter h_n. Hence, if we split $\boldsymbol{V}_n^*$ with $\mu_n^* = b_n^* - b_{n-1}^*$ by a half and put two quantizers q_{n_1} and q_{n_2} with the same parameter h_n^* in the center, we will get by using (A.46)

$$\overline{\mathcal{D}}_{n_1} + \overline{\mathcal{D}}_{n_2} = \frac{1}{h_n^*} \left(\int_{b_{n-1}^*}^{b_{n-1}^* + \frac{\mu_n^*}{2}} ((q_{n_1} - \boldsymbol{\omega})^2 + h_n^{*2})^\gamma d\boldsymbol{\omega} + \int_{b_{n-1}^* + \frac{\mu_n^*}{2}}^{b_n^*} ((q_{n_2} - \boldsymbol{\omega})^2 + h_n^{*2})^\gamma d\boldsymbol{\omega} \right)$$

Substituting $q_{n_i} - \boldsymbol{\omega}$ by $\boldsymbol{\omega}$, we get

$$= \int_{-\frac{\mu_n^*}{4}}^{\frac{\mu_n^*}{4}} \frac{(\boldsymbol{\omega}^2 + h_n^{*2})^\gamma}{h_n^*} d\boldsymbol{\omega} + \int_{-\frac{\mu_n^*}{4}}^{\frac{\mu_n^*}{4}} \frac{(\boldsymbol{\omega}^2 + h_n^{*2})^\gamma}{h_n^*} d\boldsymbol{\omega} \tag{A.47}$$

$$= 2 \int_0^{\frac{\mu_n^*}{4}} \frac{(\boldsymbol{\omega}^2 + h_n^{*2})^\gamma}{h_n^*} d\boldsymbol{\omega} + w \int_0^{\frac{\mu_n^*}{4}} \frac{(\boldsymbol{\omega}^2 + h_n^{*2})^\gamma}{h_n^*} d\boldsymbol{\omega} \tag{A.48}$$

$$< 2 \int_0^{\frac{\mu_n^*}{4}} \frac{(\boldsymbol{\omega}^2 + h_n^{*2})^\gamma}{h_n^*} d\boldsymbol{\omega} + 2 \int_{\frac{\mu_n^*}{4}}^{\frac{\mu_n^*}{2}} \frac{(\boldsymbol{\omega}^2 + h_n^{*2})^\gamma}{h_n^*} d\boldsymbol{\omega} = 2 \int_0^{\frac{\mu_n^*}{2}} \frac{(\boldsymbol{\omega}^2 + h_n^{*2})^\gamma}{h_n^*} d\boldsymbol{\omega} = \overline{\mathcal{D}}_n. \tag{A.49}$$

Hence, the average distortion will strictly decrease if $\mu_n^* > 0$. Therefore, the $N-$level parameter optimized quantizer will have quantization boundaries $b_n > b_{n-1}$ for $n \in [N]$.

A.4 Proof of Theorem 3.1

We know by Lemma 7 that the optimal quantization regions are closed non-vanishing intervals $\boldsymbol{V}_n^* = [b_{n-1}^*, b_n^*]$ for some $b_{n-1}^* < b_n^*$ with quantization points

$$\mathbf{q}_n^* = x_n^* = \frac{b_n^* + b_{n-1}^*}{2} \tag{A.50}$$

for $n \in [N]$. Let us set $\mu_n^* = b_n^* - b_{n-1}^*$ for $n \in [N]$. By substituting $\frac{2(x_n^* - \boldsymbol{\omega})}{\mu_n} = \tilde{\boldsymbol{\omega}}$ and $h_n^* = \frac{u_n^* \mu_n^*}{2}$ in the average distortion, we get

$$\begin{aligned} \overline{\mathcal{D}}(\mathbf{Q}^*, \mathbf{h}^*, \boldsymbol{V}^*) &= \sum_{n=1}^{N} \int_{b_{n-1}^*}^{b_n^*} \frac{((x_n^* - \boldsymbol{\omega})^2 + {h_n^*}^2)^\gamma}{h_n^*} \frac{d\boldsymbol{\omega}}{A} \qquad (A.51) \\ = \sum_{n=1}^{N} \int_{1}^{-1} -\frac{(\mu_n^{*2}\tilde{\boldsymbol{\omega}}^2/4 + u_n^{*2}\mu_n^{*2}/4)^\gamma}{u_n^* \mu_n^*/2} \frac{\mu_n}{2A} d\tilde{\boldsymbol{\omega}} &= \frac{1}{2^{2\gamma-1}A} \sum_{n=1}^{N} \mu_n^{*2\gamma} \cdot \int_0^1 \frac{(\boldsymbol{\omega}^2 + u_n^{*2})^\gamma}{u_n^*} d\boldsymbol{\omega} \end{aligned}$$

where we used (A.50) to get for the integral boundaries $2(x_n^* - b_{n-1}^*)/\mu_n^* = 1 = -2(x_n^* - b_n^*)/\mu_n^*$. We do not know the value of u_n^* and μ_n^* but we know that $\mu_n^* > 0$ and $\sum_{n=1}^N \mu_n^* = A$ by Lemma 7. Furthermore, (A.51) is the minimum over all such $\mu_n > 0$ and $u_n > 0$. Hence, it must hold

$$\begin{aligned} \overline{\mathcal{D}}(\mathbf{Q}^*, \mathbf{h}^*, \boldsymbol{V}^*) &= \frac{1}{2^{2\gamma-1}A} \min_{u_n > 0} \min_{\substack{\mu_n > 0 \\ A = \sum_{n=1}^N \mu_n}} \sum_{n=1}^{N} \mu_n^{2\gamma} \cdot \left(\int_0^1 \frac{(\boldsymbol{\omega}^2 + u_n^2)^\gamma}{u_n} d\boldsymbol{\omega} \right) \\ &= \frac{g(\gamma)}{2^{2\gamma-1}A} \min_{\substack{\mu_n > 0 \\ A = \sum_{n=1}^N \mu_n}} \sum_{n=1}^{N} \mu_n^{2\gamma} \end{aligned}$$

where in the last equality we used Lemma 6. By the Hlder inequality we get for $p = 2\gamma, q = 2\gamma/(2\gamma - 1)$,

$$\sum_{n=1}^{N} \mu_n^{2\gamma} = \sum_{n=1}^{N} \mu_n^p = \sum_{n=1}^{N} \mu_n^p \cdot \Big(\sum_{n=1}^{N} (1/N)^q \Big)^{p/q} \cdot N \geq \Big(\sum_{n=1}^{N} \frac{\mu_n}{N} \Big)^p \cdot N = \left(\frac{A}{N} \right)^{2\gamma} N$$

where the equality is achieved if and only if $\mu_n^* = A/N$. Hence, the optimal parameter-quantizer is the uniform scalar quantizer $x_n^* = (2n-1)A/2N$ with identical parameters $h^* = (A/2N)g(\gamma)$ resulting in the minimum average distortion (3.30).

Let us note here, that for identical parameters, the Mbius regions are closed intervals and reduce to Euclidean Voronoi regions by Lemma 5, for which the optimal tessellation is known to be the uniform scalar quantizer, see for example [59].

A.5 Proof of Proposition 2

When there is only one FC, all APs transfer data to the unique FC located at $\boldsymbol{q}$, and the index map is simply given by $\mathbf{T}(n) = 1, n \in \mathcal{I}_{\mathcal{A}}$. The corresponding distortion is

$$\mathcal{D}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) = \sum_{n=1}^{N} \int_{\mathcal{R}_n^{\mathcal{A}}} \left[\|\boldsymbol{p}_n - \boldsymbol{\omega}\|^2 + \beta \|\boldsymbol{p}_n - \boldsymbol{q}\|^2 \right] f(\boldsymbol{\omega}) d\boldsymbol{\omega}.$$

Since

$$\|\boldsymbol{p}_n - \boldsymbol{\omega}\|^2 + \beta \|\boldsymbol{p}_n - \boldsymbol{q}\|^2 = (1+\beta) \left\| \boldsymbol{p}_n - \tfrac{(\boldsymbol{\omega} + \beta \boldsymbol{q})}{1+\beta} \right\|^2 + \tfrac{\beta \|\boldsymbol{\omega} - \boldsymbol{q}\|^2}{1+\beta}, \tag{A.52}$$

we obtain

$$\mathcal{D}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) = \frac{1}{1+\beta} \sum_{n=1}^{N} \int_{\mathcal{R}_n^{\mathcal{A}}} ((1+\beta)\boldsymbol{p}_n - \beta \boldsymbol{q}) - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} + \frac{\beta}{1+\beta} \int_{\Omega} \|\boldsymbol{\omega} - \boldsymbol{q}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega}, \tag{A.53}$$

The first term in (A.53) is the distortion of a one-tier quantizer with linear transformation of its reproduction points (AP locations). The minimum value of the first term is $D_r(X^*, \mathbf{R}^*)$ and can be achieved by choosing the optimal AP deployment $\boldsymbol{P}$ for any FC location $\boldsymbol{q}$. On the other hand, the second term in (A.53) is the distortion of another quantizer whose reproduction point is the FC, and is independent of the choice of AP locations and partition cells. In other words, the second term only depends on the FC location $\boldsymbol{q}$. As a result,

one can optimize (A.53) by finding the optimal $\boldsymbol{q}^*$ to minimize the second term and then calculate the optimal AP deployment $\boldsymbol{P}^*$ for $\boldsymbol{q}^*$. By parallel axis theorem, the second term achieves the minimum if and only if the FC is placed at the geometric centroid $c = \frac{\int_\Omega \boldsymbol{\omega} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}{\int_\Omega f(\boldsymbol{\omega}) d\boldsymbol{\omega}}$ of Ω, which proves (i). The best possible distortion is then the summation of $\frac{1}{1+\beta} D_r(X^*, \mathbf{R}^*)$ and $\frac{\beta}{1+\beta} \int_\Omega \|\boldsymbol{\omega} - \mathbf{c}\| f(\boldsymbol{\omega}) d\boldsymbol{\omega}$, which proves (iv). The two-tier quantizer achieves this minimum when $\boldsymbol{q}^* = \mathbf{c}$, $x_n^* = \left((1+\beta)\boldsymbol{p}_n^* - \beta \boldsymbol{q}^*\right), n \in \mathcal{I}_{\mathcal{A}}$, and $\mathcal{R}^{\mathcal{A}} = \mathbf{R}^*$, which proves (ii) and (iii).

A.6 Proof of Proposition 3

Let $(\boldsymbol{P}^*, \boldsymbol{Q}^*, \mathcal{R}^{\mathcal{A}*}, \mathbf{T}^*)$ be an optimal solution for (4.7). As we show at the beginning of Sec. 4.3, given the optimal deployment $(\boldsymbol{P}^*, \boldsymbol{Q}^*)$, the optimal partition and the optimal index map are, respectively, $\mathcal{R}^{\mathcal{A}*} = \boldsymbol{V}^E(\boldsymbol{P}^*, \boldsymbol{Q}^*)$ and $\mathbf{T}^* = \mathbf{T}^E_{[\boldsymbol{P}^*, \boldsymbol{Q}^*]}$. Thus, the optimal geometric centroid and the optimal Lebesgue measure (volume) of $\mathcal{R}^{\mathcal{A}*}$ can be represented as $\mathbf{c}_n(\boldsymbol{P}^*, \boldsymbol{Q}^*)$ and $\mathbf{v}_n(\boldsymbol{P}^*, \boldsymbol{Q}^*)$, where $\mathbf{c}_n(\boldsymbol{P}, \boldsymbol{Q}) = \frac{\int_{\boldsymbol{V}_n^E(\boldsymbol{P}, \boldsymbol{Q})} \boldsymbol{\omega} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}{\int_{\boldsymbol{V}_n^E(\boldsymbol{P}, \boldsymbol{Q})} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}$ and $\mathbf{v}_n(\boldsymbol{P}, \boldsymbol{Q}) = \int_{\boldsymbol{V}_n^E(\boldsymbol{P}, \boldsymbol{Q})} f(\boldsymbol{\omega}) d\boldsymbol{\omega}$. According to the parallel axis theorem, given the optimal partition $\boldsymbol{V}^E(\boldsymbol{P}^*, \boldsymbol{Q}^*)$ and the optimal index map $\mathbf{T}^E_{[\boldsymbol{P}^*, \boldsymbol{Q}^*]}$, the objective function in (4.7) can be expressed as

$$\begin{aligned} \mathcal{D}\left(\boldsymbol{P}, \boldsymbol{Q}, \boldsymbol{V}^E(\boldsymbol{P}^*, \boldsymbol{Q}^*), \mathbf{T}^E_{[\boldsymbol{P}^* \boldsymbol{Q}^*]}\right) = & \sum_{n=1}^{N} \int_{\boldsymbol{V}_n^E(\boldsymbol{P}^*, \boldsymbol{Q}^*)} \|\mathbf{c}_n(\boldsymbol{P}^*, \boldsymbol{Q}^*) - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} \\ + \|\boldsymbol{p}_n - \mathbf{c}_n(\boldsymbol{P}^*, \boldsymbol{Q}^*)\|^2 \mathbf{v}_n(\boldsymbol{P}^*, \boldsymbol{Q}^*) + & \sum_{n=1}^{N} \left[\beta \|\boldsymbol{p}_n - \boldsymbol{q}_{\mathbf{T}^E_{[\boldsymbol{P}^*, \boldsymbol{Q}^*]}}\|^2 \mathbf{v}_n(\boldsymbol{P}^*, \boldsymbol{Q}^*)\right]. \end{aligned} \tag{A.54}$$

The partial derivatives of (A.54) are

$$\begin{aligned} & \frac{\partial \mathcal{D}(\boldsymbol{P}, \boldsymbol{Q}, \boldsymbol{V}^E(\boldsymbol{P}^*, \boldsymbol{Q}^*), \mathbf{T}^E_{[\boldsymbol{P}^*, \boldsymbol{Q}^*]})}{\partial \boldsymbol{p}_n} \\ = & 2\left[(\boldsymbol{p}_n - \mathbf{c}_n(\boldsymbol{P}^*, \boldsymbol{Q}^*)) + \beta\left(\boldsymbol{p}_n - \boldsymbol{q}_{\mathbf{T}^E_{[\boldsymbol{P}^*, \boldsymbol{Q}^*]}(n)}\right)\right] \mathbf{v}_n(\boldsymbol{P}^*, \boldsymbol{Q}^*), n \in \mathcal{I}_{\mathcal{A}}, \end{aligned}$$

and

$$\frac{\partial \mathcal{D}(\boldsymbol{P}, \boldsymbol{Q}, \boldsymbol{V}^E(\boldsymbol{P}^*, \boldsymbol{Q}^*), \mathbf{T}^E_{[\boldsymbol{P}^*, \boldsymbol{Q}^*]})}{\partial \boldsymbol{q}_m} = \sum_{n: \mathbf{T}^E_{[\boldsymbol{P}^*, \boldsymbol{Q}^*]}(n) = m} 2\beta\left(\boldsymbol{q}_m - \boldsymbol{p}_n\right) \mathbf{v}_n\left(\boldsymbol{P}^*, \boldsymbol{Q}^*\right), m \in \mathcal{I}_{\mathcal{B}}.$$

Since (A.54) is a convex function of P and Q, the optimal deployment $(\boldsymbol{P}^*, \boldsymbol{Q}^*)$ satisfies zero gradient. Solving for $\boldsymbol{p}_n^*$ and $\boldsymbol{q}_m^*$, we obtain

$$\boldsymbol{p}*_n = \frac{\mathbf{c}_n(\boldsymbol{P}^*, \boldsymbol{Q}^*) + \beta q^*_{\mathbf{T}^E_{[\boldsymbol{P}^*, \boldsymbol{Q}^*]}(n)}}{1+\beta}, n \in \mathcal{I}_{\mathcal{A}} \tag{A.55}$$

$$q_m^* = \frac{\sum_{n:\mathbf{T}^E_{[\boldsymbol{P}^*, \boldsymbol{Q}^*]}(n)=m} \boldsymbol{p}_n^* \mathbf{v}_n(\boldsymbol{P}^*, \boldsymbol{Q}^*)}{\sum_{n:\mathbf{T}^E_{[\boldsymbol{P}^*, \boldsymbol{Q}^*]}(n)=m} \mathbf{v}_n(\boldsymbol{P}^*, \boldsymbol{Q}^*)}, m \in \mathcal{I}_{\mathcal{B}} \tag{A.56}$$

Substituting (A.55) to (A.56), we have

$$\boldsymbol{q}_m^* = \frac{\sum_{n:\mathbf{T}^E_{[\boldsymbol{P}^*, \boldsymbol{Q}^*]}(n)=m} \mathbf{c}_n(\boldsymbol{P}^*, \boldsymbol{Q}^*) \mathbf{v}_n(\boldsymbol{P}^*, \boldsymbol{Q}^*)}{\sum_{n:\mathbf{T}^E_{[\boldsymbol{P}^*, \boldsymbol{Q}^*]}(n)=m} \mathbf{v}_n(\boldsymbol{P}^*, \boldsymbol{Q}^*)}, m \in \mathcal{I}_{\mathcal{B}}. \tag{A.57}$$

$$d(N, \Omega) \overset{(a)}{=} \sum_{n=1}^{N} \int_{R_n} \|x_n - \boldsymbol{\omega}\|^2 \frac{d\boldsymbol{\omega}}{\mu(\Omega)} \tag{A.58}$$

$$\overset{(b)}{=} \sum_{n=1}^{N} \frac{\mu(R_n)}{\mu(\Omega)} d(1, R_n) \tag{A.59}$$

$$\overset{(c)}{\geq} \sum_{n=1}^{N} \frac{1}{\mu(\Omega)} \frac{\mu(R_n)^3}{12} \tag{A.60}$$

$$\overset{(d)}{\geq} \frac{1}{12\mu(\Omega)} \left(\sum_{n=1}^{N} \mu(R_n) \right)^3 N^{-2} \tag{A.61}$$

$$\overset{(e)}{=} \frac{\mu(\Omega)^2}{12N^2}, \tag{A.62}$$

A.7 Proof of Theorem 4.1

Before we discuss the best possible distortion in the uniformly distributed 1-dimensional space, we need to present the following concepts and Lemmas. Let $\mu(W)$ be the (Lebesgue) measure of the set W. Let $d(N, \Omega) = \min_{x_1, \ldots, x_N} \int_\Omega \min_n \|x_n - \boldsymbol{\omega}\|^2 \frac{d\boldsymbol{\omega}}{\mu(\Omega)}$ be the minimum distortion of the N-level one-tier quantizer for a uniform distribution on $\Omega \subset \mathbb{R}$.

Lemma 11. *We have* $d(N,\Omega) \geq \frac{\mu(\Omega)^2}{12N^2}$ *with equality if and only if* $\boldsymbol{\omega}$ *is the union of* N *disjoint intervals, each with measure* $\frac{\mu(\Omega)}{N}$.

Proof. Let $x_1,\ldots,x_N \in \Omega$, and $R_1,\ldots,R_N \subset \mathbb{R}$ respectively denote the reproduction points and the quantization cells of the optimal one-tier quantizer that achieves $d(N,\Omega)$. Note that x_n is the centroid of R_n. We have (A.62), where (a) follows since for any $x_n \in \mathbb{R}$, we have $d(1,R_n) = \int_{R_n} \|x_n - \boldsymbol{\omega}\|^2 \frac{d\boldsymbol{\omega}}{\mu(R_n)}$ by definition, and (c) follows since for any $W \subset \mathbb{R}$, we have $d(1,W) \geq \frac{\mu(W)^2}{12}$ with equality if and only if W is an interval [126]. Also, (d) is the reverse Hölder's inequality, and (e) follows since $\sum_{n=1}^{N} \mu(R_n) = \mu(\Omega)$. Note that (c) is an equality if and only if $\boldsymbol{\omega}_n$s are intervals, and (d) is an equality if and only if $\mu(\Omega_n) = \frac{\mu(\Omega)}{N}, \forall n$. Therefore, (e) can be achieved if and only if $\boldsymbol{\omega}$ is the union of disjoint intervals with the same measure $\frac{\mu(\Omega)}{N}$. □

Lemma 12. *Let* N *and* M *be two positive integers such that* $N \geq M$. *We define a function* $D^{LB}(e_1,\ldots,e_M) = \left(\sum_{m=1}^{M}\left(\beta + \frac{1}{e_m^2}\right)^{-\frac{1}{2}}\right)^{-2}$ *with the domain* $R_c = \{(e_1,\ldots,e_M) | \sum_{m=1}^{M} e_m = N, e_m \in \mathbb{N}, \forall m\}$, *where* β *is a non-negative constant. Let* $M_a = (N \bmod M)$ *and* $M_b = M - M_a$. *Then,* $D^{LB}(e_1,\ldots,e_M)$ *attains the unique minimum*

$$\left(M_a\left(\beta + \frac{1}{\lceil \frac{N}{M} \rceil^2}\right)^{-\frac{1}{2}} + M_b\left(\beta + \frac{1}{\lfloor \frac{N}{M} \rfloor^2}\right)^{-\frac{1}{2}}\right)^{-2} \tag{A.63}$$

where M_a *of the* e_ms *are equal to* $\lceil \frac{N}{M} \rceil$ *and* M_b *of the* e_ms *are equal to* $\lfloor \frac{N}{M} \rfloor$. *In particular, when* $M_a = 0$, $D^{LB}(e_1,\ldots,e_M)$ *attains the unique minimum* $\left(\frac{\beta}{M^2} + \frac{1}{N^2}\right)$ *at* $\left(\frac{N}{M},\ldots,\frac{N}{M}\right)$.

Proof. Let $\mathbf{e} = (e_1,\ldots,e_M) \in R_c$, $D^{UB}(\mathbf{e}) = \sum_{m=1}^{M}\left(\beta + \frac{1}{e_m^2}\right)^{-\frac{1}{2}}$. Minimizing D^{LB} is equivalent to maximizing D^{UB}. Let $i,j \in \{1,\ldots,M\}$ be two arbitrary indices. When $e_k, \forall k \neq i,j$ and $\gamma = e_i + e_j$ are fixed, D^{UB} only depends on the difference between e_i and e_j. Without

loss of generality, suppose $e_i \geq e_j$. Let $\delta = e_i - e_j$, we have

$$\begin{aligned}\widetilde{D}_{ij}^{UB}(\delta) &= D^{UB}\left(e_1, \ldots, \frac{\zeta\gamma+\delta}{2}, \ldots, \frac{\gamma-\delta}{2}, \ldots, e_m\right) \\ &= \sum_{k\neq i,j}\left(\beta + \frac{1}{e_k^2}\right)^{-\frac{1}{2}} + \left(\beta + \frac{1}{\left(\frac{\gamma+\delta}{2}\right)^2}\right)^{-\frac{1}{2}} + \left(\beta + \frac{1}{\left(\frac{\gamma-\delta}{2}\right)^2}\right)^{-\frac{1}{2}},\end{aligned} \tag{A.64}$$

$$\frac{\partial \widetilde{D}_{ij}^{UB}(\delta)}{\partial \delta} = \frac{1}{2}\left[\left(1+\beta\left(\frac{\gamma+\delta}{2}\right)^2\right)^{-\frac{3}{2}} - \left(1+\beta\left(\frac{\gamma-\delta}{2}\right)^2\right)^{-\frac{3}{2}}\right]. \tag{A.65}$$

Let $g(y) = \frac{1}{2}y^{-\frac{3}{2}}, y \in (0,\infty)$, $y(x) = 1 + x^2$, $x \in [0,\infty)$. Since e_i and e_j are non-negative and $e_i \geq e_j$, we have $\delta \geq 0$, $\gamma \geq 0$, and then $x_1 = \frac{\gamma+\delta}{2} \geq \frac{\gamma-\delta}{2} = x_2$. Consequently, $y(x_1) \geq y(x_2) > 0$, and thus, $\frac{\partial \widetilde{D}^{UB}(\delta)}{\partial \delta} = g(y(x_1)) - g(y(x_2)) \leq 0$ with equality if and only if $\delta = 0$. Therefore, $\widetilde{D}_{ij}^{UB}(\delta)$ is a decreasing function for non-negative continuous δ.

Let $\mathbf{e}^* = (e_1^*, \ldots, e_M^*) \triangleq \arg\min_{(e_1,\ldots,e_M)\in R_c} D^{LB}(e_1, \ldots, e_M)$ be a minimizer of D^{LB} on R_c, and $\hat{\delta} \triangleq \min_{i\neq j} |e_i^* - e_j^*|$ be the minimum difference among e_m^*s. Since e_i^*s are positive integers, we have $\hat{\delta} \in \mathbb{N}$. In what follows, we show that $\hat{\delta} \in \{0, 1\}$. Suppose $\hat{\delta} \geq 2$. Then, we can find two indices $i, j \in \{1, \ldots, M\}$ such that $\delta = e_i^* - e_j^* \geq 2$. Let $\mathbf{e}' = (e_1^*, \ldots, e_i', \ldots, e_j', \ldots, e_M^*)$ be a new solution where $e_i' = e_i^* - 1$, and $e_j' = e_j^* + 1$. We have $\delta' = e_i' - e_j' = e_i^* - e_j^* - 2 = \delta - 2 < \delta$. Since $\widetilde{D}_{ij}^{UB}(\delta)$ is a monotonically decreasing function for non-negative continuous δ, we have $\widetilde{D}_{ij}^{UB}(\delta') > \widetilde{D}_{ij}^{UB}(\delta)$ where δ' and δ are non-negative integers. Thus, we have $D^{UB}(\mathbf{e}') > D^{UB}(\mathbf{e}^*)$ which contradicts the optimality of $\mathbf{e}^*$.

Therefore, $\hat{\delta} \in \{0, 1\}$, and e_m^*s can thus assume at most 2 distinct values. Suppose M_1 of the e_m^*s are equal to h and M_2 of the e_m^*s are equal to $h + 1$, where $h \geq 0$ is an integer. It is self-evident that at least one of M_1 or M_2 should be positive. Without loss of generality, suppose $M_1 > 0$ and $M_2 \geq 0$. Since $M_1 + M_2 = M$ and $M_1 > 0$, we have $0 < M_1 \leq M$ and $0 \leq M_2 < M$. From the equalities $M_1 + M_2 = M$ and $M_1 h + M_2(h + 1) = N$, we obtain $Mh + M_2 = N$. Solving the system $Mh + M_2 = N$ and $0 \leq M_2 < M$, we have $h = \lfloor \frac{N}{M} \rfloor$ and $M_2 = N \mod M = M_a$. Finally, using the equality $M_1 + M_2 = M$, we can determine $M_1 = M - (N \mod M) = M_b$. □

Now, we have enough tools to derive the best possible distortion in the uniformly distributed 1-dimensional space. We have

$$
\begin{aligned}
\mathcal{D}(\boldsymbol{P},\boldsymbol{Q},\mathcal{R}^{\mathcal{A}},\mathbf{T}) &= \sum_{m=1}^{M}\sum_{n\in\mathcal{N}_m}\int_{\mathcal{R}_n^{\mathcal{A}}}\left(\|\boldsymbol{p}_n-\boldsymbol{\omega}\|^2+\beta\|\boldsymbol{p}_n-\boldsymbol{q}_m\|^2\right)d\boldsymbol{\omega}\\
&\overset{(a)}{=}\sum_{m=1}^{M}\sum_{n\in\mathcal{N}_m}\int_{\mathcal{R}_n^{\mathcal{A}}}\left(\frac{1}{1+\beta}\left\|(1+\beta)\boldsymbol{p}_n-\beta\mathbf{q}-\boldsymbol{\omega}\right\|^2+\frac{\beta}{1+\beta}\|\boldsymbol{\omega}-\boldsymbol{q}\|^2\right)d\boldsymbol{\omega}\\
&\overset{(b)}{\geq}\sum_{m=1}^{M}\left[\frac{1}{1+\beta}d(N_m,W_m)+\frac{\beta}{1+\beta}d(1,W_m)\right]\frac{\mu(W_m)}{\mu(\Omega)}\\
&\overset{(c)}{\geq}\frac{1}{12(1+\beta)\mu(\Omega)}\sum_{m=1}^{M}\mu^3(W_m)\left(\beta+\frac{1}{N_m^2}\right)\\
&\overset{(d)}{\geq}\frac{1}{12(1+\beta)\mu(\Omega)}\left(\sum_{m=1}^{M}\mu(W_m)\right)^3\left(\sum_{m=1}^{M}\left(\beta+\frac{1}{N_m^2}\right)^{-\frac{1}{2}}\right)^{-2}\\
&\overset{(e)}{=}\frac{\mu^2(\boldsymbol{\omega})}{12(1+\beta)}\left(\sum_{m=1}^{M}\left(\beta+\frac{1}{N_m^2}\right)^{-\frac{1}{2}}\right)^{-2}\\
&\overset{(f)}{\geq}\frac{\mu^2(\boldsymbol{\omega})}{12(1+\beta)}\min_{\substack{N_1,\ldots,N_M\in\mathcal{N}\\ \sum_{m=1}^{M}N_m=N}}\left(\sum_{m=1}^{M}\left(\beta+\frac{1}{N_m^2}\right)^{-\frac{1}{2}}\right)^{-2}
\end{aligned}
\tag{A.66}
$$

where (a) follows from (A.52), the first inequality follows from the definition of $d(N,\Omega)$, the second inequality follows from Lemma 11, and the third inequality is the reverse Hölder's inequality. All these inequalities can be made tight with a specific choice of $\boldsymbol{p}_n$, $\boldsymbol{q}_m$, W_ms and N_ms. In fact, by Proposition 3, (b) is an equality if and only if $\boldsymbol{p}_n=\frac{\mathbf{c}_n+\beta\boldsymbol{q}_{\mathbf{T}(n)}}{1+\beta}$ and $\boldsymbol{q}_m$ is the centroid of W_m, indicating (iii) and (v) in Theorem 4.1. Also, according to Lemma 11, (c) is an equality if and only if $W_m, m\in\mathcal{I}_{\mathcal{B}}$, are intervals, and W_m is uniformly divided into N_m intervals. Therefore, (iv) in Theorem 4.1 is proved. According to the reverse Hölder's inequality, (d) is an equality if and only if $\exists\tau>0, \mu(W_m)=\tau\left(\beta+\frac{1}{N_m^2}\right)^{-\frac{1}{2}}, \forall m\in\mathcal{I}_{\mathcal{B}}$. Moreover, the sum of these measures is $\mu(\Omega)$, i.e., $\sum_{j=1}^{M}\mu(W_j)=\mu(\Omega)$. Therefore, the corresponding measure of the m^{th} cluster region is

$$
\mu(W_m)=\frac{N_m\left(1+\beta N_m^2\right)^{-\frac{1}{2}}\mu(\Omega)}{\sum_{j=1}^{M}N_j\left(1+\beta N_j^2\right)^{-\frac{1}{2}}}. \tag{A.67}
$$

Note that (e) is a function of $(N_1, \ldots, N_M)$ and (f) is just the minimum of (e). Therefore, the last inequality is an equality when we properly select the variables $N_1, \ldots, N_M$. Obviously, the above equality conditions are compatible, i.e., all equality conditions can be satisfied simultaneously. Therefore, (f) is an achievable lower bound, indicating the minimum distortion. The last thing is to determine the optimal $(N_1, \ldots, N_M)$ that attains the minimum of (e). By Lemma 12, (e) attains (f), if and only if M_a of the N_ms are equal to $\lceil \frac{N}{M} \rceil$ and M_b of the N_ms are equal to $\lfloor \frac{N}{M} \rfloor$. Substituting the optimal values for $N_m s$ to (e), we obtain the minimum distortion formula in (4.17). Substituting the optimal values for N_ms to (A.67), we get (i) and (ii) in Theorem 4.1.

A.8 Proof of Lemma 8

Since $\mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}})$ is the distortion of an N-level quantizer, we have $\mathcal{P}^{\mathcal{S}} \in [D_R(N), +\infty)$ which is then the domain of the function $A(S)$. Next, we justify the monotonicity of $A(S)$. Let $F(S)$ be the set of the feasible solutions for the point $(S, A(S))$. Thus, the AP-Sensor power function can be rewritten as $A(S) = \inf_{(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) \in F(S)} \mathcal{P}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T})$. For any two values S_1 and S_2 such that $D_R(N) \leq S_1 < S_2$, we have $F(S_1) \subseteq F(S_2)$ and then $A(S_1) \geq A(S_2)$. In other words, $A(S)$ is a non-increasing function. Note that $D_R(\cdot)$ is a mean-square-error distortion with continuous and differentiable source. Therefore, $D_R(\cdot)$ is a strictly decreasing function. Since $N > M$, we have $D_R(N) < D_R(M)$. Now, we discuss the values of $A(S)$ on $[D_R(M), +\infty)$. Let $(X^*, \mathbf{R}^*) = \arg\min_{(X, \mathbf{R})} \sum_{n=1}^{M} \int_{R_n} \|x - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega}$ be the optimal solution for the M-level one-tier quantizer. $X^* = (x_1^*, \ldots, x_M^*)$ and $\mathbf{R}^* = (R_1^*, \ldots, R_M^*)$ are, respectively, the optimal reproduction points and quantization regions. Let $\boldsymbol{P'} = (x_1^*, x_2^*, \ldots, x_M^*, x_1^*, \ldots, x_1^*)$ be a deployment including N points (APs), in which the last $N - M$ APs have the same location x_1^*. Afterwards, we define $\mathcal{R}'^{\mathcal{A}} = \mathbf{R}^*$, $\boldsymbol{Q'} = X^*$, and $\mathbf{T}'(n) = \mathbf{T}^E_{[\boldsymbol{P'}, \boldsymbol{Q'}]}(n) = \arg\min_m \|\boldsymbol{p}'_n - \boldsymbol{q}'_m\|$. Substituting $(\boldsymbol{P'}, \boldsymbol{Q'}, \mathcal{R}'^{\mathcal{A}}, \mathbf{T}')$ to (4.4) and (4.3), we obtain $\mathcal{P}^{\mathcal{S}}(\boldsymbol{P'}, \mathcal{R}'^{\mathcal{A}}) = D_R(M)$ and $\mathcal{P}^{\mathcal{A}}(\boldsymbol{P'}, \boldsymbol{Q'}, \mathcal{R}'^{\mathcal{A}}, \mathbf{T}') = 0$. In other words, $A(D_R(M)) \leq 0$. On one hand, since $A(S)$ is a non-negative non-increasing function, we

have $A(S) = 0$ on $[D_R(M), +\infty)$. On the other hand, by quantization theorem, when N APs are placed at K distinct locations, where $K \leq M$, we have $\mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}}) \geq D_R(M)$. In other words, when $\mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}}) < D_R(M)$, N APs have at least $M+1$ distinct locations. Under such circumstances, it is impossible to connect each AP to the FC with zero distance, indicating that $\mathcal{P}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) > 0$. Therefore, we have $A(S) > 0$ on $[D_R(N), D_R(M))$.

A.9 Proof of Lemma 9

At the beginning, we prove (i), the domain of $\widehat{A}(S, \mathcal{R}^{\mathcal{A}})$ shown in the Lemma is correct. An input $(S, \mathcal{R}^{\mathcal{A}})$ belongs to the domain if and only if there exists a P such that $\mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}}) \leq S$. When $\mathcal{R}^{\mathcal{A}}$ is fixed, the range of $\mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}})$ is $[H(\mathcal{R}^{\mathcal{A}}), +\infty)$, where $H(\mathcal{R}^{\mathcal{A}})$ is the minimum distortion of a one-tier quantizer with partition $\mathcal{R}^{\mathcal{A}}$. Therefore, the domain of $\widehat{A}(S, \mathcal{R}^{\mathcal{A}})$ can be represented as $\{(S, \mathcal{R}^{\mathcal{A}}) | H(\mathcal{R}^{\mathcal{A}}) \leq S\}$. Next, we prove (ii), the value of $\widehat{A}(S, \mathcal{R}^{\mathcal{A}})$ on the domain shown in the Lemma is correct. Note that one can simply achieve the minimum AP-power 0 with Sensor-power $D_R(1)$ by placing all APs at the centroid of the target area. Therefore, when $x \in [D_R(1), +\infty)$, $\widehat{A}(S, \mathcal{R}^{\mathcal{A}}) = 0$. In what follows, we focus on the case that $x \in [D_R(N), D_R(1))$. To calculate the value of $\widehat{A}(S)$ at $(S, \mathcal{R}^{\mathcal{A}})$, we assume that the AP cell partition is fixed as $\mathcal{R}^{\mathcal{A}}$. Therefore, the centroid, $\mathbf{c}_n = \frac{\int_{\mathcal{R}_n^{\mathcal{A}}} \boldsymbol{\omega} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}{\int_{\mathcal{R}_n^{\mathcal{A}}} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}$, and the volume of $\mathcal{R}_n^{\mathcal{A}}$, $\mathbf{v}_n = \int_{\mathcal{R}_n^{\mathcal{A}}} f(\boldsymbol{\omega}) d\boldsymbol{\omega}$, are constants. Since $M = 1$, the index map $\mathbf{T}(n) = 1, \forall n \in \mathcal{I}_{\mathcal{A}}$, is determined. Let $\boldsymbol{q}$ be the location of the unique FC. Therefore, the AP-power function becomes

$$\begin{aligned} &\mathcal{P}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) = \sum_{n=1}^{N} \int_{\mathcal{R}_n^{\mathcal{A}}} \|\boldsymbol{p}_n - \boldsymbol{q}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} \\ &= \sum_{n=1}^{N} \|\boldsymbol{p}_n - \boldsymbol{q}\|^2 \mathbf{v}_n = \sum_{n=1}^{N} \|\boldsymbol{p}_n \sqrt{\mathbf{v}_n} - \boldsymbol{q}\sqrt{\mathbf{v}_n}\|^2 = \|\widetilde{\mathbf{p}} - \widetilde{\mathbf{q}}\|^2, \end{aligned} \tag{A.68}$$

and the Sensor-power function becomes

$$\begin{aligned} &\mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}}) = \sum_{n=1}^{N} \int_{\mathcal{R}_n^{\mathcal{A}}} \|\boldsymbol{p}_n - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} = H(\mathcal{R}^{\mathcal{A}}) + \sum_{n=1}^{N} \|\boldsymbol{p}_n - \mathbf{c}_n\|^2 \mathbf{v}_n \\ &= H(\mathcal{R}^{\mathcal{A}}) + \sum_{n=1}^{N} \|\boldsymbol{p}_n \sqrt{\mathbf{v}_n} - \mathbf{c}_n \sqrt{\mathbf{v}_n}\|^2 = H(\mathcal{R}^{\mathcal{A}}) + \|\widetilde{\mathbf{p}} - \widetilde{\mathbf{c}}\|^2, \end{aligned} \tag{A.69}$$

where $\widetilde{\mathbf{p}} \triangleq [\boldsymbol{p}_1\sqrt{\mathbf{v}_1}, \ldots, \boldsymbol{p}_N\sqrt{\mathbf{v}_N}]$, $\widetilde{\mathbf{q}} \triangleq [\boldsymbol{q}\sqrt{\mathbf{v}_1}, \ldots, \boldsymbol{q}\sqrt{\mathbf{v}_N}]$, and $\widetilde{\mathbf{c}} \triangleq [\mathbf{c}_1\sqrt{\mathbf{v}_1}, \ldots, \mathbf{c}_N\sqrt{\mathbf{v}_N}]$. Since $H(\mathcal{R}^{\mathcal{A}})$ is a constant, (4.20) can be rewritten as

$$\widehat{A}(S, \mathcal{R}^{\mathcal{A}}) = \inf_{(\widetilde{\mathbf{p}}, \widetilde{\mathbf{q}}): \|\widetilde{\mathbf{p}} - \widetilde{\mathbf{c}}\|^2 \leq S - H(\mathcal{R}^{\mathcal{A}})} \|\widetilde{\mathbf{p}} - \widetilde{\mathbf{q}}\|^2. \tag{A.70}$$

When $\boldsymbol{q}$ is fixed, one should minimize the distance of $\widetilde{\mathbf{p}}$ to the fixed $\widetilde{\mathbf{q}}$ subject to the constraint that $\widetilde{\mathbf{p}}$ remain within a ball of radius-square $S - H(\mathcal{R}^{\mathcal{A}})$ centered at $\widetilde{\mathbf{c}}$. A simple geometric argument reveals that the optimal solution should then fall on a line between $\widetilde{\mathbf{p}}$ and $\widetilde{\mathbf{q}}$, i.e., $\widetilde{\mathbf{p}} = \frac{\widetilde{\mathbf{q}} + \lambda\widetilde{\mathbf{c}}}{1+\lambda}$ for some $\lambda \geq 0$. Going back to the original variables, we have the optimal AP locations $\boldsymbol{p}_n = \frac{\boldsymbol{q} + \lambda\mathbf{c}_n}{1+\lambda}, n \in \mathcal{I}_{\mathcal{A}}$. Substituting the optimal AP locations to (A.69), the constraint in (A.70) becomes $(1+\lambda)^2 \geq \frac{\sum_{n=1}^{N} \|q - \mathbf{c}_n\|^2 \mathbf{v}_n}{S - H(\mathcal{R}^{\mathcal{A}})}$. Specially, when $\lambda = 0$, we have $\boldsymbol{p}_n = \boldsymbol{q}, \forall n \in \mathcal{I}_{\mathcal{A}}$. In other words, all APs are placed at the same location which is equivalent to the scenario that only one AP is placed. Again, $D_R(1)$ is the minimum distortion for the one-tier quantizer and then the minimum sensor power when only one AP is placed. Thus, when $\lambda = 0$, we have $\mathcal{P}^{\mathcal{S}} \geq D_R(1)$. Since we only consider $S \in [D_R(N), D_R(1))$, to ensure the constraint $S \geq \mathcal{P}^{\mathcal{S}}$, λ cannot be 0. Thus, the possible range of λ is $(0, +\infty)$.

On the other hand, $\mathbf{T}$ is determined and the constraint in (4.20) is independent of $\boldsymbol{Q} = \{\boldsymbol{q}\}$ so that $\widehat{A}(S, \mathcal{R}^{\mathcal{A}})$ can be rewritten as

$$\begin{aligned}
&\widehat{A}(S, \mathcal{R}^{\mathcal{A}}) = \inf_{(\boldsymbol{P}, \boldsymbol{Q}, \mathbf{T}): \mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}}) \leq S} \mathcal{P}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) \\
&= \inf_{\boldsymbol{Q}} \inf_{(\boldsymbol{P}, \mathbf{T}): \mathcal{P}^{\mathcal{S}}(\boldsymbol{P}, \mathcal{R}^{\mathcal{A}}) \leq S} \mathcal{P}^{\mathcal{A}}(\boldsymbol{P}, \boldsymbol{Q}, \mathcal{R}^{\mathcal{A}}, \mathbf{T}) \\
&= \inf_{\boldsymbol{q}} \inf_{\lambda: \lambda > 0, (1+\lambda)^2 \geq \frac{\sum_{n=1}^{N} \|\boldsymbol{q} - \mathbf{c}_n\|^2 \mathbf{v}_n}{S - H(\mathcal{R}^{\mathcal{A}})}} \sum_{n=1}^{N} \left\| \frac{\boldsymbol{q} + \lambda\mathbf{c}_n}{1+\lambda} - \boldsymbol{q} \right\|^2 \mathbf{v}_n \\
&= \inf_{\lambda: \lambda > 0, (1+\lambda)^2 \geq \frac{\sum_{n=1}^{N} \|\boldsymbol{q} - \mathbf{c}_n\|^2 \mathbf{v}_n}{S - H(\mathcal{R}^{\mathcal{A}})}} \inf_{\boldsymbol{q}} Z(\boldsymbol{q}),
\end{aligned} \tag{A.71}$$

where $Z(\boldsymbol{q}) = \sum_{n=1}^{N} \|\frac{\boldsymbol{q} + \lambda\mathbf{c}_n}{1+\lambda} - \boldsymbol{q}\|^2 \mathbf{v}_n$. In what follows, we attempt to find the optimal $\boldsymbol{q}$ to minimize $Z(\boldsymbol{q})$. Note that $Z(\boldsymbol{q})$ is a quadratic function and convex. Therefore, the

unique minimum is associated with zero-gradient. Since $\frac{\partial Z(\boldsymbol{q})}{\partial \boldsymbol{q}} = \frac{2\lambda^2}{(1+\lambda)^2}\left(\boldsymbol{q} - \frac{\sum_{n=1}^{N}\mathbf{c}_n\mathbf{v}_n}{\sum_{n=1}^{N}\mathbf{v}_n}\right)$, we obtain the optimal FC location $\boldsymbol{q} = \frac{\sum_{n=1}^{N}\mathbf{c}_n\mathbf{v}_n}{\sum_{n=1}^{N}\mathbf{v}_n} = \frac{\sum_{n=1}^{N}\int_{\Omega}\boldsymbol{\omega}f(\boldsymbol{\omega})d\boldsymbol{\omega}}{\sum_{n=1}^{N}\mathbf{v}_n} = \mathbf{c}$, where $\mathbf{c}$ is the centroid of the target region $\boldsymbol{\omega}$. Substituting the optimal $\boldsymbol{q}$ to (A.71), we get $\widehat{A}(S,\mathcal{R}^{\mathcal{A}}) = \inf_{\lambda:\lambda>0,(1+\lambda)^2\geq\frac{J(\mathcal{R}^{\mathcal{A}})}{x-H(\mathcal{R}^{\mathcal{A}})}} \frac{\lambda^2}{(1+\lambda)^2}J(\mathcal{R}^{\mathcal{A}})$, where $J(\mathcal{R}^{\mathcal{A}}) = \sum_{n=1}^{N}\|\mathbf{c}-\mathbf{c}_n\|^2\mathbf{v}_n$. According to the parallel axis theorem, we have

$$\begin{aligned} D_R(1) &= \sum_{n=1}^{N}\int_{\mathcal{R}_n^{\mathcal{A}}}\|\mathbf{c}-\boldsymbol{\omega}\|^2 f(\boldsymbol{\omega})d\boldsymbol{\omega} = \sum_{n=1}^{N}\|\mathbf{c}-\mathbf{c}_n\|^2\mathbf{v}_n \\ &+ \sum_{n=1}^{N}\int_{\mathcal{R}_n^{\mathcal{A}}}\|\boldsymbol{\omega}-\mathbf{c}_n\|^2 f(\boldsymbol{\omega})d\boldsymbol{\omega} = J(\mathcal{R}^{\mathcal{A}}) + H(\mathcal{R}^{\mathcal{A}}). \end{aligned} \tag{A.72}$$

Therefore, $\widehat{A}(S,\mathcal{R}^{\mathcal{A}})$ can be rewritten as $\widehat{A}(S,\mathcal{R}^{\mathcal{A}}) = \inf_{\lambda:\lambda>0,(1+\lambda)^2\geq\frac{D_R(1)-H(\mathcal{R}^{\mathcal{A}})}{S-H(\mathcal{R}^{\mathcal{A}})}} \frac{\lambda^2}{(1+\lambda)^2}\left(D_R(1)-H(\mathcal{R}^{\mathcal{A}})\right)$. Solving the inequality $(1+\lambda)^2 \geq \frac{D_R(1)-H(\mathcal{R}^{\mathcal{A}})}{S-H(\mathcal{R}^{\mathcal{A}})}$ results in $\lambda \geq \sqrt{\frac{D_R(1)-H(\mathcal{R}^{\mathcal{A}})}{S-H(\mathcal{R}^{\mathcal{A}})}}-1$ or $\lambda \leq -\sqrt{\frac{D_R(1)-H(\mathcal{R}^{\mathcal{A}})}{S-H(\mathcal{R}^{\mathcal{A}})}}-1$. Since $\lambda > 0$, we have only one solution, i.e., $\lambda \geq \sqrt{\frac{D_R(1)-H(\mathcal{R}^{\mathcal{A}})}{S-H(\mathcal{R}^{\mathcal{A}})}}-1$. Since $\frac{\lambda^2}{(1+\lambda)^2}$ is monotonically increasing, $\widehat{A}$ attains the minimum at $\lambda = \sqrt{\frac{D_R(1)-H(\mathcal{R}^{\mathcal{A}})}{S-H(\mathcal{R}^{\mathcal{A}})}}-1$. Substituting $\lambda = \sqrt{\frac{D_{R(1)}-H(\mathcal{R}^{\mathcal{A}})}{S-H(\mathcal{R}^{\mathcal{A}})}}-1$ to $\widehat{A}(S,\mathcal{R}^{\mathcal{A}})$, we prove (4.21).

A.10 Proof of Theorem 4.2

By Lemma 8, we know $A(S) = 0$ on $[D_R(M),+\infty)$ and the domain of $A(S)$ is $[D_R(N),+\infty)$. In what follows, we study $A(S)$ on $[D_R(N),D_R(M))$. Let $\widehat{A}(S,\mathcal{R}^{\mathcal{A}})$ be the minimum AP-power with the Sensor-power constraint $\mathcal{P}^{\mathcal{S}} \leq S$ for the fixed AP cell partition $\mathcal{R}^{\mathcal{A}}$ when $M = 1$. Therefore, when $M = 1$, minimizing the AP-power with the Sensor-power constraint $\mathcal{P}^{\mathcal{S}}(\boldsymbol{P},\mathcal{R}^{\mathcal{A}}) \leq S$ is equivalent to minimizing $\widehat{A}(S,\mathcal{R}^{\mathcal{A}})$ on its domain. According to Lemma 9, the domain of $\widehat{A}(S,\mathcal{R}^{\mathcal{A}})$ is $\{(S,\mathcal{R}^{\mathcal{A}})|H(\mathcal{R}^{\mathcal{A}}) \leq S\}$. Therefore, the AP-Sensor power function can be rewritten as

$$
\begin{aligned}
A(S) &= \inf_{(\boldsymbol{P},\boldsymbol{Q},\mathcal{R}^{\mathcal{A}},\mathbf{T}):\mathcal{P}^{\mathcal{S}}(\boldsymbol{P},\mathcal{R}^{\mathcal{A}})\leq S} \mathcal{P}^{\mathcal{A}}(\boldsymbol{P},\boldsymbol{Q},\mathcal{R}^{\mathcal{A}},\mathbf{T}) \\
&= \inf_{\mathcal{R}^{\mathcal{A}}:H(\mathcal{R}^{\mathcal{A}})\leq S} \inf_{(\boldsymbol{P},\boldsymbol{Q},\mathbf{T}):\mathcal{P}^{\mathcal{S}}(\boldsymbol{P},\mathcal{R}^{\mathcal{A}})\leq S} \mathcal{P}^{\mathcal{A}}(\boldsymbol{P},\boldsymbol{Q},\mathcal{R}^{\mathcal{A}},\mathbf{T}) \\
&= \inf_{\mathcal{R}^{\mathcal{A}}:H(\mathcal{R}^{\mathcal{A}})\leq S} \widehat{A}(S,\mathcal{R}^{\mathcal{A}}) \\
&= \inf_{\mathcal{R}^{\mathcal{A}}:H(\mathcal{R}^{\mathcal{A}})\leq S} \left[\sqrt{(S-H(\mathcal{R}^{\mathcal{A}}))}-\sqrt{D_R(1)-H(\mathcal{R}^{\mathcal{A}})}\right]^2,
\end{aligned}
\tag{A.73}
$$

Note that $\mathcal{R}^{\mathcal{A}}$ affects $A(S)$ merely via $H(\mathcal{R}^{\mathcal{A}})$. Moreover, $H(\mathcal{R}^{\mathcal{A}})$ is the minimum distortion of the N-level quantizer with partition $\mathcal{R}^{\mathcal{A}}$, and $D_R(N)$ is the minimum distortion of the N-level quantizer over all possible partitions. Thus, $H(\mathcal{R}^{\mathcal{A}}) \geq D_R(N)$. Given the value of S, to satisfy the inequality constraint $H(\mathcal{R}^{\mathcal{A}}) \leq S$, the range of $H(\mathcal{R}^{\mathcal{A}})$ is then $[D_R(N), S]$. Therefore, the AP-Sensor power function can be rewritten as

$$
A(S) = \inf_{y:D_R(N)\leq y\leq S} \widetilde{A}(S,y), \tag{A.74}
$$

where $\widetilde{A}(S,y) = \left[\sqrt{(S-y)}-\sqrt{(D_R(1)-y)}\right]^2$. When $S \neq y$, we have

$$
\frac{\partial \widetilde{A}(S,y)}{\partial y} = \ -2+\frac{S+D_R(1)-2y}{\sqrt{(S-y)(D_R(1)-y)}}. \tag{A.75}
$$

Extending $(S+D_R(1)-2y)^2$, we get

$$
\begin{aligned}
&(S+D_R(1)-2y)^2 = ((S-y)+(D_R(1)-y))^2 \\
=&(S-y)^2+(D_R(1)-y)^2+2(S-y)(D_R(1)-y) \\
>&4(S-y)(D_R(1)-y) = \left[2\sqrt{(S-y)(D_R(1)-y)}\right]^2.
\end{aligned}
\tag{A.76}
$$

Again, we only consider the region $[D_R(N), D_R(1))$, indicating $D_R(N) \leq S < D_R(1)$. Taking the constraint $D_R(N) \leq y \leq S$ into consideration, we have $y \leq S < D_R(1)$. When $y \neq S$, the term $\frac{S+D_R(1)-2y}{\sqrt{(S-y)(D_R(1)-y)}}$ is positive. Therefore, we have $\frac{\partial \widetilde{A}(S,y)}{\partial y} > 0, \forall S \in [D_R(N), D_R(1))$ and $S \neq y$. Therefore, when $y \in [D_R(N), S)$, $\widetilde{A}(S,y)$ is increasing and has its unique minimum at $y = D_R(N)$. Taking the case $S = y$ into account, we have $A(S) = \min(\widetilde{A}(S, D_R(N)), \widetilde{A}(S,S))$. After straightforward calculation, we get $\widetilde{A}(S, D_R(N)) < \widetilde{A}(S,S)$

when $x \in [D_R(N), \hat{D}_R(1))$. Therefore, in order to minimize $\widetilde{A}(S, y)$, y should be minimized to $D_R(N)$. In other words, the AP-Sensor power function on $[D_R(N), D_R(1))$ can be rewritten as (4.22), which is a convex function. In sum, the AP-Sensor power function is convex on the domain $[D_R(N), +\infty)$.

A.11 Proof of Lemma 10

Lemma 10 focuses on homogeneous MWSNs where $\eta_n = \eta$, $\xi_n = \xi$, and $\gamma_n = \gamma, \forall n \in \mathcal{I}_\Omega$. Let $\mathbf{P}^0 = (\mathbf{p}_1^0, \ldots, \mathbf{p}_N^0)$ and $\mathbf{P}^* = (\mathbf{p}_1^*, \ldots, \mathbf{p}_N^*)$ be the initial and the optimal sensor deployments in an MWSN with performance function (5.4) and constraints (5.8), respectively. For convenience, let $\mathcal{I}_\Omega$ be the set of all sensors, $\mathcal{S}(\mathbf{P})$ be the set of sensors that can communicate with the AP when the sensor deployment is $\mathbf{P}$, and $\mathcal{N}_n(\mathbf{P})$ be Sensor n's neighbors given deployment $\mathbf{P}$. $\mathcal{S}(\mathbf{P})$ is also referred to as the backbone network in Section 5.1. In Lemma 10, we assumed that $\mathbf{P}^0$ provides a fully connected network, i.e., $S(\mathbf{P}^0) = \mathcal{I}_\Omega$. Now, we assume that the optimal deployment is associated with a disconnected network, i.e., $\mathcal{S}(\mathbf{P}^*) \neq \mathcal{I}_\Omega$. In this case, we can find a sensor, $n \in \mathcal{S}(\mathbf{P}^*)$, such that one of its neighbors in the initial deployment, $m \in \mathcal{N}_n(\mathbf{P}^0)$, is not in the final backbone network, $m \notin \mathcal{S}(\mathbf{P}^*)$. An alternative point is defined as

$$\mathbf{p}'_m = \mathbf{p}_m^0 + \min\left(0, \|\mathbf{p}_n^* - \mathbf{p}_m^0\| - R_c\right) \frac{\mathbf{p}_n^* - \mathbf{p}_m^0}{\|\mathbf{p}_n^* - \mathbf{p}_m^0\|}. \tag{A.77}$$

Replacing $\mathbf{p}_m^*$ by $\mathbf{p}'_m$, we get an alternative deployment $\mathbf{P'} = (\mathbf{p}_1^*, \ldots, \mathbf{p}'_m, \ldots, \mathbf{p}_N^*)$. Next, we check if $\mathbf{P}'$ satisfies the network lifetime constraints (5.8). First, since sensors m and n are neighbors, we have $\|\mathbf{p}_m^0 - \mathbf{p}_n^0\| \leq R_c$. Second, following the constraints (5.8), we have $\|\mathbf{p}_n^* - \mathbf{p}_n^0\| \leq \frac{\gamma}{\xi}$. Third, using the triangular inequality, we have $\|\mathbf{p}_n^0 - \mathbf{p}_m^0\| + \|\mathbf{p}_n^* - \mathbf{p}_n^0\| > \|\mathbf{p}_m^0 - \mathbf{p}_n^*\|$. Combining the above three inequalities, we obtain $\|\mathbf{p}_n^* - \mathbf{p}_m^0\| < R_c + \frac{\gamma}{\xi}$. According to (A.77), $\mathbf{p}'_m$ is placed between $\mathbf{p}_n^*$ and $\mathbf{p}_m^0$, and the moving distance is

$$\|\mathbf{p}'_m - \mathbf{p}_m^0\| = \min\left(0, \|\mathbf{p}_n^* - \mathbf{p}_m^0\| - R_c\right) < \frac{\gamma}{\xi}. \tag{A.78}$$

Thus, the deployment $\mathbf{P}'$ satisfies the network lifetime constraints.

In what follows, we verify that $\mathbf{P}'$ provides a smaller distortion compared to $\mathbf{P}^*$. The distance between $\mathbf{p}'_m$ and $\mathbf{p}^*_n$ can be calculated as

$$\|\mathbf{p}^*_n - \mathbf{p}'_m\| = \|\mathbf{p}^*_n - \mathbf{p}^0_m\| - \|\mathbf{p}'_m - \mathbf{p}^0_m\| = \begin{cases} \|\mathbf{p}^*_n - \mathbf{p}^0_m\|, & \text{if } \|\mathbf{p}^*_n - \mathbf{p}^0_m\| \leq R_c \\ R_c, & \text{otherwise} \end{cases}, \quad \text{(A.79)}$$

which means $\|\mathbf{p}^*_n - \mathbf{p}'_m\| \leq R_c$. In other words, when the deployment is $\mathbf{P}'$, Sensor m connects with n and then should be taken into the calculation of distortion (5.4). In our model, sensors are initially deployed in the target region $\boldsymbol{\omega}$, i.e., $\mathbf{p}^0_n \in \boldsymbol{\omega}$, where $\boldsymbol{\omega}$ is a convex region [32]. Moveover, it is self-evident that the optimal sensor locations are also in the target region, i.e., $\mathbf{p}^*_n \in \boldsymbol{\omega}, \forall n \in \mathcal{I}_\Omega$. By properties of a convex region, any point between $\mathbf{p}^*_n$ and $\mathbf{p}^0_m$ should be in the target region, e.g., $\mathbf{p}'_m \in \boldsymbol{\omega}$. Therefore, Sensor m is associated with a non-empty MWVD $\boldsymbol{V}^H_m(\mathcal{H}(\mathbf{P}')) = \{\boldsymbol{\omega} | \boldsymbol{\omega} \in \boldsymbol{\omega}, \|\boldsymbol{\omega} - \mathbf{p}'_m\| < \|\boldsymbol{\omega} - \mathbf{p}^*_i\|, \forall i \in \mathcal{S}(\mathbf{P}') i \neq m\}$. The difference between the distortions at $\mathbf{P}'$ and $\mathbf{P}^*$ lays on $\boldsymbol{V}^H_m(\mathcal{H}(\mathbf{P}'))$ and can be calculated as

$$\begin{aligned} \mathcal{D}(\mathbf{P}') - \mathcal{D}(\mathbf{P}^*) &= \int_{\boldsymbol{V}^H_m(\mathcal{H}(\mathbf{P}'))} \eta \|\boldsymbol{\omega} - \mathbf{p}'_m\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} - \sum_{n \in \mathcal{S}(\mathcal{H}(\mathbf{P}^*))} \int_{\boldsymbol{V}^H_m(\mathcal{H}(\mathbf{P}')) \bigcap \boldsymbol{V}^H_n(\mathcal{H}(\mathbf{P}^*))} \eta \|\boldsymbol{\omega} - \mathbf{p}'_m\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} \\ &= \int_{\boldsymbol{V}^H_m(\mathcal{H}(\mathbf{P}'))} \eta \left(\|\boldsymbol{\omega} - \mathbf{p}'_m\|^2 - \min_{n \in \mathcal{S}(\mathbf{P}^*)} \|\boldsymbol{\omega} - \mathbf{p}^*_n\|^2 \right) f(\boldsymbol{\omega}) d\boldsymbol{\omega} < 0 \end{aligned} \quad \text{(A.80)}$$

Consequently, $\mathbf{P}'$ is a better solution than $\mathbf{P}^*$, which contradicts our assumption[4].

A.12 Proof of Theorem 5.1

Let $\mathcal{R}^* = (\mathcal{R}^*_1, \ldots, \mathcal{R}^*_N)$ be the optimal partition and $\mathcal{I}^*$ be the optimal backbone network. For simplicity, let $\mathbf{c}^*_n = \frac{\int_{\mathcal{R}^*_n} \boldsymbol{\omega} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}{\int_{\mathcal{R}^*_n} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}$ and $\mathbf{v}^*_n = \int_{\mathcal{R}^*_n} f(\boldsymbol{\omega}) d\boldsymbol{\omega}$ be, respectively, the geometric

[4]Remark: In this proof, we ignore two special cases: (i) $\mathbf{p}'_m$ is placed on top of another sensor, i.e., $\mathbf{p}'_m = \mathbf{p}^*_i, i \neq m$. In this case, $\mathbf{p}'_m$ should be moved towards $\mathbf{p}^0_m$ a little bit to avoid overlap without breaking the constraints, and then we will have the same contradiction. (ii) After replacing $\mathbf{p}^*_m$ by $\mathbf{p}'_m$, more than one sensor joins the backbone network. In this case, the distortion at $\mathbf{P}'$ will be further reduced, and therefore we will have the same contradiction.

centroid and the Lebesgue measure (volume) of $\mathcal{R}_n^*$, $\forall n \in \mathcal{I}_\Omega$. Let $m \in \mathcal{I}^*$ be a specific sensor in the backbone network, $\mathbf{P}^*$. First, we assume that other sensor locations, $\{\mathbf{p}_n^*\}_{n\in(\mathcal{I}_\Omega-\{m\})}$, are known, and then derive the constraint for $\mathbf{p}_m^*$. Since $\mathcal{I}^*$ is the backbone network, we have[5]. $\mathbf{p}_m^* \in \mathbb{D}_m\left(\mathbf{P}^*, \mathcal{I}^*\right) \bigcap \mathcal{W}^c\left(\mathbf{P}^*, \mathcal{I}^*\right)$. Moreover, since Sensor m should satisfy the energy constraints (5.8), we have $\mathbf{p}_m^* \in \mathbb{B}(\mathbf{p}_m^0, \frac{\gamma_m}{\xi_m})$. In summary, Sensor m's location is restrained by

$$\mathbb{D}_m\left(\mathbf{P}^*, \mathcal{I}^*\right) \bigcap \mathcal{W}^c\left(\mathbf{P}^*, \mathcal{I}^*\right) \bigcap \mathbb{B}\left(\mathbf{p}_m^0, R_c\right) = \mathbb{F}_m\left(\mathbf{P}^*, \mathcal{I}^*\right) \bigcap \mathcal{W}^c\left(\mathbf{P}^*, \mathcal{I}^*\right). \tag{A.81}$$

Second, the minimum distortion can be rewritten as

$$\begin{aligned}
\mathcal{D}(\mathbf{P}^*) &= \sum_{n\in\mathcal{I}^*} \int_{\mathcal{R}_n^*} \eta_n \|\mathbf{p}_n^* - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} && \text{(A.82)}\\
&= \int_{\mathcal{R}_m^*} \eta_m \|\mathbf{p}_m^* - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} + \sum_{n\in(\mathcal{I}^*-\{m\})} \int_{\mathcal{R}_n^*} \eta_n \|\mathbf{p}_n^* - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} && \text{(A.83)}\\
&= \eta_m \|\mathbf{c}_m^* - \mathbf{p}_m^*\|^2 \mathbf{v}_m^* + \int_{\mathcal{R}_m^*} \eta_m \|\mathbf{c}_m^* - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega} + \sum_{n\in(\mathcal{I}^*-\{m\})} \int_{\mathcal{R}_n^*} \eta_n \|\mathbf{p}_n^* - \boldsymbol{\omega}\|^2 f(\boldsymbol{\omega}) d\boldsymbol{\omega}, && \text{(A.84)}
\end{aligned}$$

where the third equation follows from the parallel axis theorem. Given the optimal partition $\mathcal{R}^*$ and optimal locations $\{\mathbf{p}_n^*\}_{n\in(\mathcal{I}_\Omega-\{m\})}$, the second and third terms in (A.84) are constants. Therefore, $\mathbf{p}_m^*$ should be a minimizer of the first term with the constraint (A.81), i.e.,

$$\mathbf{p}_m^* = \arg \min_{p_m : p_m \in \mathbb{F}(\mathbf{P}^*, \mathcal{I}^*) \bigcap \mathcal{W}^c(\mathbf{P}^*, \mathcal{I}^*)} \eta_m \|\mathbf{p}_m - \mathbf{c}_m^*\|^2 \mathbf{v}_m^*. \tag{A.85}$$

The equation (A.85) implies that the optimal solution should minimize the distance to $\mathbf{c}_n^*$ within $\mathbb{F}(\mathbf{P}^*, \mathcal{I}^*) \bigcap \mathcal{W}^c\left(\mathbf{P}^*, \mathcal{I}^*\right)$. In what follows, we discuss 3 different cases of $\mathbf{c}_m^*$.

(a) If $\mathbf{c}_m^* \in [\mathbb{F}(\mathbf{P}^*, \mathcal{I}^*) \bigcap \mathcal{W}\left(\mathbf{P}^*, \mathcal{I}^*\right)]$, we have $\mathbf{p}_m^* \neq \mathbf{c}_m^*$ because $\mathbf{p}^* \in [\mathbb{F}\left(\mathbf{P}^*, \mathcal{I}^*\right) \bigcap \mathcal{W}^c]$. However, replacing $\mathbf{p}_m^*$ by $\mathbf{c}_m^*$, one can get a better solution $\mathbf{P}' = \left(\mathbf{p}_1^*, \ldots, \mathbf{p}_{m-1}^*, \mathbf{c}_m^*, \mathbf{p}_{m+1}^*, \ldots, \mathbf{p}_N^*\right)$ which not only follows the energy constraints but also provides a smaller distortion. As a

[5]The definitions of $\mathbb{D}_m\left(\mathbf{P}, \mathcal{I}\right)$ and $\mathcal{W}\left(\mathbf{P}, \mathcal{I}\right)$ and their relationships to backbone network are already provided in Section 5.2

result, $\mathbf{c}_m^* \in [\mathbb{F}(\mathbf{P}^*, \mathcal{I}^*) \bigcap \mathcal{W}(\mathbf{P}^*, \mathcal{I}^*)]$ is an impossible case for the optimal deployment. In other words, we have the condition

$$\mathbf{c}_m^* \notin \left[\mathbb{F}(\mathbf{P}^*, \mathcal{I}^*) \bigcap \mathcal{W}(\mathbf{P}^*, \mathcal{I}^*)\right] \tag{A.86}$$

Since MWVD is the optimal partition, we have $\mathcal{R}^* = \boldsymbol{V}^H(\mathbf{P}^*)$ and then

$$\mathbf{c}_m^* = \frac{\int_{\boldsymbol{V}_m^H(\mathbf{P}^*)} \boldsymbol{\omega} f(\boldsymbol{\omega}) d\boldsymbol{\omega}}{\int_{\boldsymbol{V}_n^H(\mathbf{P}^*)} f(\boldsymbol{\omega}) d\boldsymbol{\omega}} = \mathbf{c}_m(\mathbf{P}^*), \forall n \in I_\Omega. \tag{A.87}$$

Moreover, the backbone network $\mathcal{I}^*$ is a function of deployment $\mathbf{P}^*$ and can be represented by

$$\mathcal{I}^* = \mathcal{S}(\mathbf{P}^*). \tag{A.88}$$

Substituting (A.87) and (A.88) to (A.86), we get condition (i).

(b) If $\mathbf{c}_m^* \in [\mathbb{F}(\mathbf{P}^*, \mathcal{I}^*) \bigcap \mathcal{W}^c(\mathbf{P}^*, \mathcal{I}^*)]$, Sensor m should be placed at $\mathbf{c}_m^*$, i.e., Replacing $\mathbf{c}_m^*$ by (A.87) , we get the first case in condition (ii).

(c) If $\mathbf{c}_m^* \notin \mathbb{F}(\mathbf{P}^*, \mathcal{I}^*)$, a simple geometric argument reveals that the optimal solution should be on the boundary of $\mathbb{F}(\mathbf{P}^*, \mathcal{I}^*)$. Therefore, the optimal location can be represented as $\mathbf{p}_m^* = \arg \min_{q \in \partial[\mathbb{F}_m(\mathbf{P}^*, \mathcal{I}^*) \bigcap \mathcal{W}^c(\mathbf{P}^*, \mathcal{I}^*)]} \|q - \mathbf{c}_m^*\|$. Replacing $\mathbf{c}_m^*$ and $\mathcal{I}^*$ by (A.87) and (A.88), respectively, we get the second case in condition (ii).

A.13 Proof of Theorem 5.3

Let $\mathbf{P}^k = \left(\mathbf{p}_1^k, \ldots, \mathbf{p}_N^k\right)$ and $\mathbf{P}^k = \left(\mathbf{p}_1^{k+1}, \ldots, \mathbf{p}_N^{k+1}\right)$ be the current and next sensor deployment. Let m and n be two sensors such that they are MST neighbors at $\mathbf{P}^k$. Then, the distance between $\mathbf{p}_m^k$ and $\mathbf{p}_n^k$ is no larger than the communication range R_c, i.e., $\|\mathbf{p}_m^k - \mathbf{p}_n^k\| \leq R_c$. According to the definition of semi-desired region (5.19), we have $\mathbb{D}_m^s(\mathbf{P}) \subset \mathbb{B}\left(\frac{\mathbf{p}_m^k + \mathbf{p}_n^k}{2}, \frac{R_c}{2}\right)$ and $\mathbb{D}_n^s(\mathbf{P}) \subset \mathbb{B}\left(\frac{\mathbf{p}_m^k + \mathbf{p}_n^k}{2}, \frac{R_c}{2}\right)$. Moreover, m and n move within their semi-desired regions, i.e., $\mathbf{p}_m^{k+1} \in \mathbb{D}_m^s\left(\mathbf{P}^k\right)$ and $\mathbf{p}_n^{k+1} \in \mathbb{D}_n^s\left(\mathbf{P}^k\right)$. Thus, both $\mathbf{p}_m^{k+1}$ and $\mathbf{p}_n^{k+1}$ lie in the circle

$\mathbb{B}\left(\frac{\mathbf{p}_m^k+\mathbf{p}_n^k}{2}, \frac{R_c}{2}\right)$. Therefore, the distance between $\mathbf{p}_m^{k+1}$ and $\mathbf{p}_n^{k+1}$ is no larger than the communication range, i.e., $\|\mathbf{p}_m^{k+1}-\mathbf{p}_n^{k+1}\| \leq R_c$. In other words, Sensors m and n are still connected with each other after the relocation. Consequently, the network at $\mathbf{P}^{k+1}$, $\mathcal{G}\left(\mathbf{P}^{k+1}\right)$, retains all edges in the MST at $\mathbf{P}^k$, indicating that the network, $\mathcal{G}\left(\mathbf{P}^{k+1}\right)$, is fully connected.