

Coverage-Awareness Scheduling Protocols for Wireless Sensor Networks

by

Xin Fei

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
For the Ph.D. degree in
Computer Science

School of Information Technology and Engineering
Faculty of Engineering
University of Ottawa

© Xin Fei, Ottawa, Canada, 2012

Abstract

The coverage and energy issues are the fundamental problems which prevent the development of wireless sensor networks. In order to accurately evaluate the monitoring quality (coverage), one needs to model the interactive of sensors, phenomenons and the environment. Furthermore, in collaborative with scheduling algorithm and computer optimization, protocols can improve the overall monitoring quality and prolong the lifetime of network.

This thesis is an investigation of coverage problem and its relative applications in the wireless sensor networks. We first discuss the realistic of current boolean sensing model and propose an irregular sensing model used to determine the coverage in the area with obstacles. We then investigate a joint problem of maintaining the monitoring quality and extending the lifetime of network by using scheduling schemes. Since the scheduling problem is NP hard, genetic algorithm and Markov decision process are used to determine an achievable optimal result for the joint problem of coverage-preserving and lifetime-prolong. In order to avoid the cost of centralized or distributed scheduling algorithms, a localized coverage-preserving scheduling algorithm is proposed by exploring the construction process of Voronoi diagram. Besides exploring the coverage characteristic in a static wireless sensor network, we investigate the coverage problem when the mobile elements are introduced into network. We consider the single-hop mobile data gathering problem with the energy efficiency and data freshness concerns in a wireless sensor network where the connectivity cannot be maintained. We first investigate the upper/lower bound of the covering time for a single collector to cover the monitoring area. Through our investigation we show that for a bounded rectangle area a hexagon walk could explore the area more efficiently than a random walk when the edges of area are known. We then propose a virtual force mobile model (VFM) in which the energy consumption for data transmission is modeled as a virtual elastic force and used to guide of mobile collectors to move to optimal positions for energy saving.

Acknowledgements

First and foremost I would like to express my sincere gratitude to my supervisor Prof. Azzedine Boukerche for the continuous support of my Ph.D study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my Ph.D study. I feel very privileged to have had the opportunity to learn from, and work with him. He not only has always believed in me, he has always been there whenever I have needed advice. For that, I am forever grateful.

My sincere thanks also go to Prof. Regina B.Araujo and Prof. Richard Liu for helping me explore my vision on the research. Their comments and suggestions have not only been crucial in bringing many aspects of this work to fruition, but they have also helped in my continued endeavors to personal and professional improvement as a researcher.

Furthermore, I would like to thank all my colleges throughout all the years who offered help in many aspects of what is presented here.

Last but not the least, I would like to thank my family who stood beside me, provided support and encouragement: my parents, Haosheng Fei, Jie Zhang, my wife, Fan Wan and my son, Ethan.

Glossaries

AGP Art Gallery Problem	MDP Markov Decision Process
ASM Association Sponsor Method	MSP Maximal Support Path
ALT-E Alternate Election Algorithm	NsD Non-similar Degree
BCC Breach Area Considered as Covered Area	PSM Probability Sensing Model
BSM Boolean Sensing Model	QoS Quality of Services
CAM Central Angle Method	RW Random Walk
CCB Covered Area Considered as Breach Area	SPT Shortest Path Tree
DT Delaunay Triangulation	TSP Travel Salesman Problem
EOFS Environment Observation and Forecasting System	TSPN Travel Salesman Problem with Neighborhoods
GA Genetic Algorithm	UCT Unit Circle Test
IPM The Intersection Point Method	VIP Vertex Intersection Point of Polygons
ISM The Irregular Sensing Model	VFM Virtual Force Model
LIP Line Intersection Point of Polygons	VFMDG Virtual Force based Mobile Data Gathering
MBP Maximal Breach Path	WRW Weighted Random Walk
MDC Mobile Data Collector	WSN Wireless Sensor Networks

Contents

1	Introduction	1
1.1	Overview	1
1.2	Problem Statements	2
1.3	Summary of Contributions	4
1.4	Organization of Dissertation	6
2	Related Works	7
2.1	The Coverage Problem of WSNs	7
2.2	Coverage Evaluation Methods	9
2.2.1	Sensing Models	9
2.2.2	Network Coverage Models	11
2.3	Coverage-Preserving Algorithms	19
2.3.1	Coverage-Preserving Deployment methods	19
2.3.2	Energy-efficient Coverage	21
2.3.3	Connectivity-preserving Coverage	26
3	Coverage Estimation Based on Irregular Sensing Range Model	30
3.1	Analysis of the Sensing Range Modeling	31
3.2	Shaping Based Irregular Sensing Model	34
3.3	Determining the Sensing Range Using α -Shape	35
3.3.1	Review of α -shape	36

3.3.2	Determination of α	37
3.4	Revised α -Shape Algorithms for Sensing Range Estimation	39
3.4.1	Weighted α -shape	39
3.4.2	Boundary Ring Based α -shape	41
3.4.3	Complexity of Boundary-ring Based α Shape	45
3.5	The Distributed Coverage Evaluation Protocol	46
3.6	Simulation Experiments	47
3.6.1	Boundary Recognition	47
3.6.2	Coverage Rate and Coverage Errors	48
3.6.3	Times of Insertion	52
3.7	Conclusion	52
4	Localized Scheduling Protocol for WSN with Irregular Sensing Range	53
4.1	Introduction	53
4.2	The Intersection Point Method(IPM)	54
4.2.1	Basic Definition	54
4.2.2	The Intersection Points Method	56
4.2.3	The Unit Circle Test	58
4.2.4	Non-Similar Degree	60
4.2.5	Comparison of IPM with Other Methods	62
4.3	Performance Evaluation	63
4.3.1	Experimental Environment	63
4.3.2	Experimental Results	64
4.4	Conclusion	69
5	GA Based Coverage-preserving Scheduling Protocol	71
5.1	Introduction	71
5.2	Definitions and Problem Statement	71
5.2.1	Definitions of Coverage	72

5.2.2	Problem Statement of Coverage Preserving Scheduling	73
5.3	Related Work of GA for Coverage	74
5.4	Grid-cover Vs. Set-Cover	77
5.4.1	Grid Based Fitness Function	77
5.4.2	Set-Covering Based Fitness Function	79
5.4.3	Time Complexity of Fitness Functions	80
5.5	Genetic Algorithm for Coverage-aware Scheduling	81
5.5.1	Encoding and Initialization	81
5.5.2	Selection and Fitness Function	82
5.5.3	Reproduction and Termination	82
5.6	Experimental Results	83
5.6.1	The Node Distributions	88
5.7	Conclusion	89
6	MDP Based Coverage-preserving Scheduling Protocol	90
6.1	Introduction	90
6.2	Restless Bandit Formulation and Solution	91
6.2.1	The Restless Bandit Problem	91
6.2.2	System Formulation	92
6.2.3	Solving the Restless Bandit Problem by LP Relaxation	93
6.3	Problem Statement and Modeling	95
6.3.1	The Coverage Model	95
6.3.2	The Energy Model	95
6.3.3	The Cost Model	96
6.4	The Scheduling Algorithm	97
6.5	Simulation Experiments	98
6.5.1	The Simulation Environment	98
6.5.2	Experimental Results	100

6.6	Conclusion	101
7	Mobile Coverage and Mobile Data Gathering Protocols for WSN	102
7.1	Related Works of Mobile Data Gathering	103
7.1.1	Static Data Gathering	104
7.1.2	Mobile Data Gathering	104
7.2	Problem Statement and Analysis	110
7.2.1	Energy Efficiency of Single-hop Data Gathering	111
7.2.2	Data Freshness and Area Exploration	115
7.3	A Localized Mobile Data Gathering Protocol Based on VFM	118
7.3.1	Time Arrangement for Data Transmission	120
7.3.2	Trap and Hexagon Exploration	121
7.3.3	Analysis of Multi-collector	122
7.4	Experiment and Discussions	123
7.4.1	The Simulation Environment	124
7.4.2	Experimental Results	125
7.5	Conclusion	133
8	Conclusion and Future Work	134
8.1	Our Contributions	134
8.2	Future Work	135

List of Tables

4.1	Comparison of Fully Sponsored Discovery Methods	62
5.1	Parameters of Network	84
5.2	Parameters of Genetic Algorithm	84
6.1	Transition Matrices for Coverage Model	99
6.2	Transition Matrices for Energy Model	99
7.1	Radio Parameters	124

List of Figures

1.1	The coverage challenge of WSN	2
1.2	Thesis works	4
2.1	Category of coverage problems	8
2.2	Central angle method	22
2.3	Association sponsor method	22
3.1	Electric nose surround by non-penetrated obstacles	31
3.2	Electric nose in windy grassland	32
3.3	The indoor polygon sensing range	34
3.4	The outdoor polygon sensing range	34
3.5	Alpha shape	38
3.6	Inappropriately selected alpha	38
3.7	Delaunay triangulation, 300 samples	40
3.8	Weight of edge	41
3.9	Boundary ring	43
3.10	Boundary recognition of U shape	48
3.11	Boundary recognition of star shape	48
3.12	Coverage rate without obstacles	49
3.13	Coverage rate with obstacles	49
3.14	CCB errors	50
3.15	BCC errors	50

3.16	Insertion at boundary ring	51
3.17	Times of insertion	51
4.1	The breach intersection polygon	56
4.2	Unit circle test for polygon P_S	59
4.3	Hole-tolerance	60
4.4	The non-similar degree	61
4.5	Candidates Vs. deployed nodes,disk sensing range	64
4.6	Candidates Vs. deployed nodes, equilateral triangle sensing range	65
4.7	Errors Vs. deployed nodes, equilateral triangle sensing range	65
4.8	Candidates Vs. the number of deployed nodes, star sensing range	66
4.9	Errors Vs. deployed nodes, star sensing range	66
4.10	Coverage ratio Vs. deployed nodes, $n = 100$, star sensing range	67
4.11	The number of candidates Vs. radius of test circle, $n=100$	67
4.12	Holes Vs. radius of test circle, $n=100$	68
4.13	Off-duty sensors Vs. deployed nodes, star sensing range	68
5.1	K-coverage	72
5.2	Network topology	73
5.3	Genetic algorithm	75
5.4	Coverage rate example	78
5.5	Test points	80
5.6	Coverage Vs. network lifetime	85
5.7	Overall coverage Vs. network lifetime	86
5.8	The number of active sensors Vs. network lifetime	86
5.9	Honeycomb distribution, 1-cover	87
5.10	Distribution for grid based GA	87
5.11	Distribution for set-covering based GA	88
6.1	Coverage of network	100

6.2	The number of active nodes	101
7.1	Potential energy.	115
7.2	The flow charts of VFMDG.	119
7.3	Trapped collector.	121
7.4	Data Freshness with Single MDC.	126
7.5	Data Freshness with 4 MDCs.	126
7.6	Coverage of single MDC.	127
7.7	Area coverage of multiple-MDC after 200 simulation seconds.	128
7.8	Node coverage.	128
7.9	The movement trajectory.	129
7.10	The number of visited sensors.	130
7.11	Energy distribution of VFMDG and SPT.	131
7.12	The average energy usage for single MDC	131
7.13	Travel length with single MDC.	132

Chapter 1

Introduction

1.1 Overview

In recent years, wireless sensor networks (WSN) have been used in many situations, such as habitat monitoring, EOFS (Environment Observation and Forecasting System) applications, health applications, battlefield monitoring, museums security, weather detection, wild animal protection and some other human inaccessible critical areas. The critical environments make the replacement of power be impossible and arise the joint issue of quality-preserving and energy-saving to the front line of research. The most challenging research brought by the applications addressed above is: “How to keep the network in a good monitoring quality and live as long as possible”. Such challenge is often related to the QoS issue and known as coverage-preserving energy-saving problem which includes many perspectives such as: How to keep good monitoring quality in face of high rates of failure? What is the optimal number of nodes that meets coverage quality? What are the breach points in a sensor field? When tracking of people or objects is required, as in a search for survivors during a critical condition, what is the optimal number of nodes necessary for target detection? How accurate is the location? Moreover, for the supervision and control applications, different types of sensors with different processing power and communication properties (radio range) might be integrated with themselves

and with actuators (more resourceful nodes with capability to respond to events in real-time) in the same WSN; How can coverage be met with heterogeneous sensors? How to manage the mobile element to cover the monitoring area or the subjects?

The solutions attacking these issues pose great challenges and show the research interests in all the layers of WSN as shown in Figure 1.1.

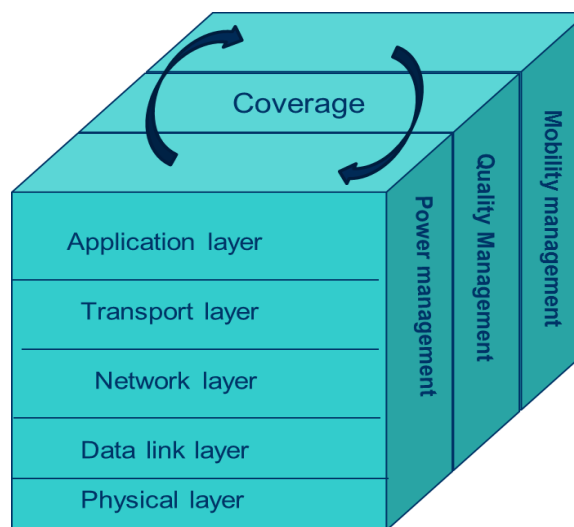


Figure 1.1: The coverage challenge of WSN

1.2 Problem Statements

We focus our researches in the following three parts:

1. Coverage Evaluation

The accuracy and reality requirement makes precise coverage evaluation methods a challenge of coverage problem. The challenge is two-folded. The first is the modeling of sensing range. The most common sensor model which is used by most protocols is boolean sensing model which assumes that a sensor can cover a disk area centered at itself with a radius equal to the sensing range. However, in most cases, the sensing system can not really have a disk sensing range. The sensing range

is usually irregular and location-dependent (for example, the photo sensor uses a square sensing range). Moreover, if we consider the effect of reflection caused by boundaries and obstacles, the sensor can not maintain its disk sensing range unless it is working on a completely flat and boundary-free area. Problems with such irregular sensing range have been discussed by Huang and colleagues[1]. Authors pointed out that their K-coverage scheme can work under an irregular polygon sensing range assumption; however, they did not literally prove this. Apparently, when the irregular polygon sensing range problem is applied to region coverage, the coverage evaluation became more complicated. The issues mentioned above lead to the challenges that motivate our research: “How to estimate the coverage of sensors with irregular sensing range? Can a distributed scheme solves the coverage problem under a polygon sensing range without relying on the GPS system?”

2. Coverage-preserving and Energy-efficiency

Maintaining the coverage quality while saving energy of wireless sensor network is one of the major challenges in wireless sensor network. A frequently used energy saving protocol in the wireless sensor network is coverage preserving duty-cycle scheme. By means of computing geometry and computer graphics some solutions proposed for the wired sensor network were transplant to the wireless sensor network. However, the heavy message overhead and the centralized model made most of them can not gracefully handle the restrict conditions in the wireless sensor network to resolve coverage problem. A localized, light message overhead algorithm has been expected. Besides, we prefer to have an achievable optimal result to serve as a base for the performance comparison. Since the scheduling problem is NP hard, it is difficult to be resolved in linear time. By modeling the coverage-preserving scheduling problem using tools such as genetic algorithm and Markov decision process, we show some optimized solutions.

3. Mobile Coverage

With the advances of wireless networks and the widespread use of thin mobile devices such as cellular phones and personal digital assistants (PDAs), researchers are investigating the use of hybrid network which contains both static sensors and mobile elements. Algorithmic work has included self-deployment, network reconfiguration and path planning. In the algorithms, mobile elements adjust their location online to maintain the surveillance quality. We are targeting the problem where the mobile elements collaborate with each other to move and cover the area or specific spots. The covering time, energy consumption and through output are our concerns.

1.3 Summary of Contributions

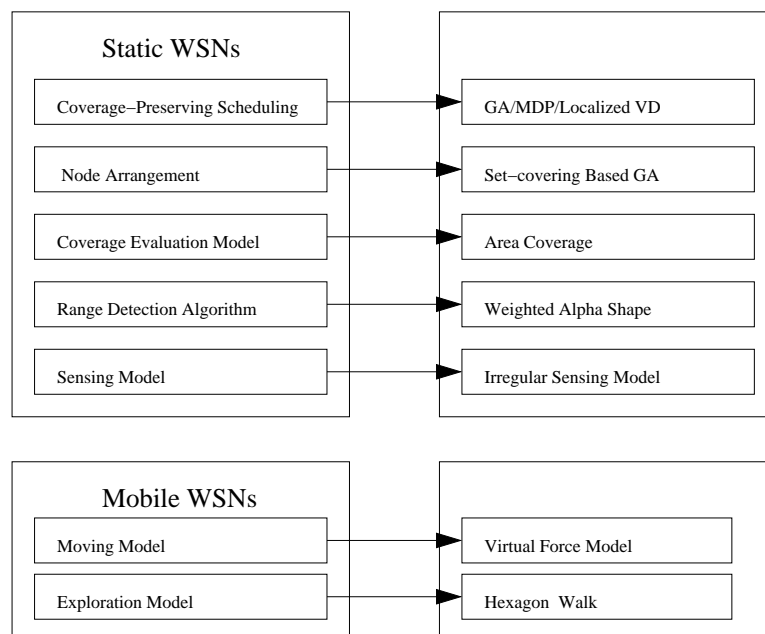


Figure 1.2: Thesis works

In this thesis, we consider the coverage problem in both static WSN and mobile network. We investigate the coverage problem from the fundamental issue (sensing model),

up to the optimization of coverage-preserving scheduling problem as shown in Figure 1.2. The main contribution of this thesis is three-folded:

1. We investigate the coverage problem in the wireless sensor networks and propose an irregular sensing range model. Based on the α -shape, we develop a new set of algorithm to estimate the coverage quality of network with irregular sensing range model.
2. By investigating the joint problem of coverage-preserving and energy-efficient, we propose three novel scheduling algorithms. Since the scheduling problem is NP hard, a set-covering based genetic algorithm is developed to find an achievable optimal result for static wireless sensors network. Markov decision process is used to derive the optimal result when the dynamic of network is considered. In order to avoid the cost of centralized and/or distributed scheduling algorithm, we propose an efficient localized scheduling algorithm based on our irregular sensing range model.
3. Beside investigating the coverage problem in a static wireless sensor network. We extend our research to the mobile coverage in a wireless sensor network where the network connectivity can not be maintained. We consider the single-hop mobile data gathering problem with the energy efficiency and data freshness concerns. We first investigate the upper/lower bound of the covering time for a single collector to cover the monitoring area. Through our investigation we show that for a bounded rectangle area a hexagon walk could explore the area more efficiently than a random walk when the edges of area is known. We then propose a virtual force mobile model (VFM) in which the energy consumption for data transmission is modeled as a virtual elastic force and used to guide mobile collectors to move to optimal positions for energy saving.

1.4 Organization of Dissertation

The rest of the dissertation is organized as follows. In Chapter 2, we review the proposed coverage protocols in the recent years. In Chapter 3, we investigate the sensing models used for coverage evaluation and propose our irregular sensing range model. Revised alpha-shape algorithms are developed to estimate the sensing polygon of sensors. In Chapter 4, based on the irregular sensing range model we investigate the coverage-preserving scheduling problem of wireless sensor network. we propose three methods for sensors to detect the breach area and identify the fully sponsored sensors locally. Sensors are scheduled to work in different time slots for maintaining the coverage quality and saving the energy of network. Since the scheduling problem is NP hard, in Chapter 5, genetic algorithm is used to derive the achievable optimal results. Instead of using a grid based coverage, we propose a set-covering based fitness function to reduce the time complexity of algorithm. In Chapter 6, we consider the scheduling problem in a stochastic system. The scheduling problem is modeled as restless bandit problem and resolved by using Markov decision process. In Chapter 7, we consider the single-hop mobile data gathering problem in a sparsely deployed wireless sensor network with the energy efficiency and data freshness concerns. We first analyze the covering time for a mobile collector to cover a WSN where the connectivity of network cannot be maintained. Through our analysis we show that for a bounded rectangle area a hexagon walk could explore the area more efficiently than a random walk. We then investigate the energy efficiency of radio propagation model of WSN and propose a virtual force model (VFM) for guiding the mobile collectors to move to the energy-efficient positions. Finally, in Chapter 8, we conclude this dissertation with a summary of the main contributions and the potential extensions to the conducted research.

Chapter 2

Related Works

Wireless sensor networks (WSNs) consist of a large number of small sensor devices which are powered by a battery and easy to be deployed in a monitoring field. The deployed sensors are organized into a network to monitor the environment and report the collected information to a central server for further analysis. Because of the flexibility and the low cost of WSNs, they have been used in many applications especially for those applications in the human inaccessible areas such as forest fire monitoring, deep sea exploration and battlefield rescue. In order to meet the QoS requirement of applications, we measure the performance of network and investigate its characteristic for the further improvement. A key measurement for the performance of wireless sensor network is the coverage which presents how well the sensing area is monitored by the sensor networks. In the following sections, we investigate the current work of coverage evaluation methods, survey some of proposed algorithms and categorize the algorithms in terms of their goals. The algorithms in the same category are compared with each other in terms of their methodologies.

2.1 The Coverage Problem of WSNs

The coverage problem was first introduced into wireless sensor network by the art gallery problem (AGP) [2], which is about how to determine the minimum number of guards

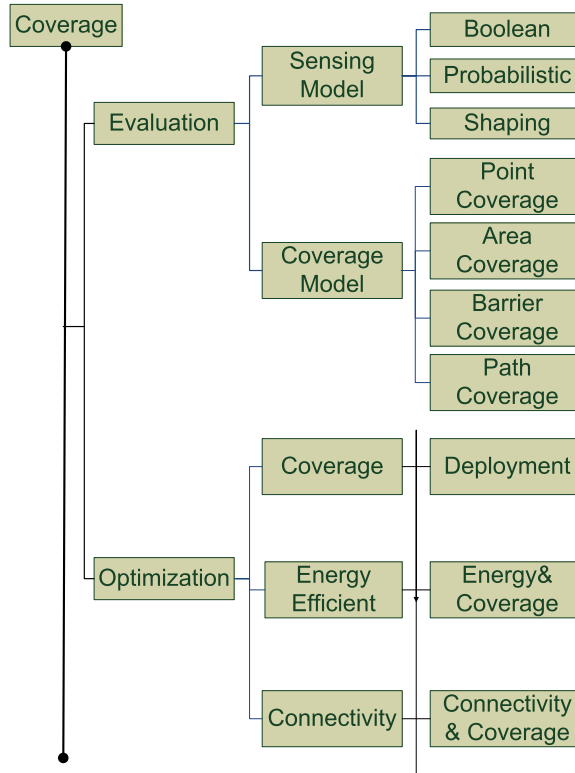


Figure 2.1: Category of coverage problems

required to cover the interior of an art gallery (represented by a polygon). The algorithms shown in [2] solve the AGP optimally in two-dimension by decomposing simple polygons into sub-polygons such as triangles, convex polygons, etc. After the AGP, the coverage problem quickly attracts a lot of attention of researchers and because a research hot spot of wireless sensor network.

As shown in Figure 2.1, we classify the research of coverage problem into two groups, coverage evaluation and coverage optimization. In the category of coverage evaluation, we focus on the definitions and the measurement methods of coverage for different scenarios. Generally we can define the coverage as the degree of a subject to be observed, analyzed, and reported. However, the definition of coverage in the WSNs has many variations which are slight different with each other in order to satisfy the requirements of applications. We first introduce the currently proposed sensing models because the way sensor observed

the environment highly effects the performance measurements of entire system. And then, we describe different coverage measurements for the entire network.

In the category of coverage optimization, we survey current coverage-preserving algorithms which optimize the coverage performance of network with or without other performance measurements such as energy-efficiency, network connectivity and so on. Based on the different combination of optimization goals, the mains interests of coverage-preserving algorithms can be classified into three sub-categories, coverage-preserving deployment, energy-efficient coverage and connectivity-preserving coverage. The goal of coverage-preserving deployment is to find the best node arrangement which satisfies certain coverage requirement. In the energy-efficient coverage, researchers are considering to extend the lifetime of network while maintaining the coverage quality. In the connectivity-preserving coverage, the coverage-preserving node arrangements have to satisfy the constraint of network connectivity for guaranteeing that the sensed data could be reported to the sink.

2.2 Coverage Evaluation Methods

The coverage evaluation methods are used to determine the monitoring capability and quality of sensor networks. They rely on the sensing model of sensors and are subject to different interpretations for different application scenarios. This section introduces some common sensor models and evaluation methods including their motivations, formulations, interpretations, and applications.

2.2.1 Sensing Models

A sensing model measures the sensing capability and quality of sensors by capturing the geometric relation between a space point and sensors. Because a sensor gathers signals from environment and changes them into another energy form, it can be considered as a type of transducer. According to the energy type of signal, sensors can be classified into

different categories, such as temperature sensors, electromagnetic sensors, mechanical sensors, chemical sensors, optical radiation sensors, acoustic sensors, and so on. The sensing model of each type of sensor highly depends on its category. In the context of coverage problem for wireless sensor networks, we only concern those sensors whose sensing capability and quality are related to the Euclidean distances. Generally, in a 2D space, a sensing model can be formulated as a function of the Euclidean distance between the space points and sensors. The result of such function is a non-negative real number which refers to the coverage degree of the point. The definition of the function varies for the different scenarios.

The Boolean Model

The boolean coverage model is the most widely used sensing model for the coverage problem of wireless sensor network. It is a simplification of the coverage problem where the sensing area of a sensor is assumed to be a circle with a fixed radius R_s . Depending on whether the event is located in the sensing range, sensors give a boolean value to determine the success or failure of detections.

$$p(d) = \begin{cases} 1 & d \leq R_s \\ 0 & d > R_s \end{cases}$$

Therefore, the coverage problem can be optimally done by resolving a classic disk covering problem in the set theory or a tessellation problem in graph theory. However, this model ignores the the condition effects of the environment. For example, the obstacles in the field may changed the strength of the emitted signal on the task of sensing.

The Probabilistic Model

Probability sensing model is more realistic than boolean model. It catches the distance effect that the strength of sensing signal may drop along the path. In the probability sensing model, the sensing ability of sensors is not uniform in all the directions. A general

probability sensing model can be formulated as following, where $f(d)$ is defined by the chosen path loss model.

$$p(d) = \begin{cases} 1 & d \leq r_1 \\ f(d) & r_1 \leq d \leq r_{\max} \\ 0 & d > r_{\max} \end{cases}$$

For example, assuming the path loss of sensing is similar to the shadowing in radio wave propagation, by apply a log-normal shadowing path loss model, the probability of a successful detection at a distance d is given as following by [3]:

$$P(d) = Q\left(\frac{10n \log_{10}(d/d_0)}{\sigma}\right)$$

$$Q(Z) = \int_Z^{\infty} \exp(-x^2/2) dx / \sqrt{2\pi}$$

Where, n is the path loss exponent; d_0 is the sensing radius without fading, σ is the fading parameter and x is a Gaussian random variable with a zero mean and a variance σ^2 .

2.2.2 Network Coverage Models

Based on the different sensing models the network coverage models are defined and evaluated in the context of different applications. Gage first employed the concept of coverage to evaluate multi-robot systems in [4]. He defined three basic types of coverage: blanket coverage, where the objective is to achieve a static arrangement of nodes that maximizes the total detection area; barrier coverage, where the objective is to minimize the probability of undetected penetration through the barrier; and sweep coverage, which is more-or-less equivalent to a moving barrier. After that, researchers completed the definition of those three coverage models and further extended the definitions to other scenarios. In this section, we focus on the definition of coverage models and their evaluation methods. The researches on the joint problems of coverage and other issues are presented in the next section.

Fundamental Concepts of Coverage

Two fundamental concepts of network coverage model are coverage degree and coverage rate. Coverage degree describes how well a space point is covered by sensors. For example, a point is called k -covered if it is within k distinct sensors' coverage disks. The similar definition can also be applied to other network coverage models.

Coverage ratio measures how much area of a sensing field or how many targets satisfy the requirement of coverage degree. For example, if three out of ten targets are covered, then the coverage ratio is 30%. A complete coverage (full coverage) usually refers to the scenario where every point/target within the sensing field covered by at least one sensor. Maintaining the complete coverage is high cost. Sensors have to be deployed redundantly. As a trade-off between monitoring quality and cost, in some scenario we use partial coverage where some points or targets are allowed to be uncovered.

Area Coverage

The area coverage or the physic coverage is the first type of coverage evaluation metric introduced into wireless sensor network by the art gallery problem (AGP) [2]. The main objective of area coverage is to evaluate how well the entire region is monitored by a wireless sensor network. Because in the nondeterministic high density wireless sensor network the monitor area is always assumed fully covered by sensors, the research of region coverage problem is always included in some other issues such as coverage quality, network connectivity, energy issues and deployment methods.

Definition 1 Polygon:

A polygon P is defined by a set of connected edges

$$E = \left\{ \overline{e_1e_2}, \overline{e_2e_3}, \overline{e_3e_4}, \dots, \overline{e_{n-1}e_n} : \overline{e_i e_j} \in P \text{ for } i, j = 1, \dots, n \right\}$$

Definition 2 Art Gallery Problem:

Given a simple polygon P , find the number of guards needed to cover P .

In 1975, Chvatal established the "Chvatal's Art Gallery Theorem" where $n/3$ guards are occasionally necessary and always sufficient for any polygon with n vertices. A typical proof of art gallery theorem was Fisk's method [5]. The main steps of Fisk's method is described in Algorithm 1.

Algorithm 1 Fisk's method

- Triangulate the given simple polygon P ;
 - 3-color the triangulated simple polygon;
 - Choose the color used least in the 3 - coloring;
 - Place guards at the vertices of this color.
-

the first step of Fisk's proof triangulates the polygon into partitions of triangles. Based on triangulation graphs of polygons, 3-coloring is applied to the triangulated polygon. The least frequently used color is picked; guards is placed at vertices of that color. The location and the number of guards that is sufficient to guard an art gallery can be yielded. The solutions of AGP become the foundations of most region coverage algorithms in wireless sensor network.

Huang et al.[1] introduced a k -coverage problem (where k was a predefined constant) to determine if every point in a given area was sufficiently covered by at least k sensors. To determine which areas are insufficiently covered, [1] assumed that there is a central controller in the sensor network, which broadcasts a desired value k to all sensors. Each sensor communicates with its neighboring sensors and then determines which segments of its perimeter are less than k -perimeter-covered. The results are sent back to the central controller. By putting all segments together, the central controller can precisely determine which areas are less than k -covered. The algorithms were proved that they can be applied under both disk sensing range and irregular sensing range. A $O(n \log n)$ algorithm was proposed to determine whether a sensor s_i is k -perimeter-covered or not, where n was the number of sensors which had intersection with s_i . The distance between two sensors s_i and s_j was denoted by $d(s_i, s_j)$. s_j contributed coverage to s_i if and only if $d(s_i, s_j) \leq 2r$. The algorithm to determine the perimeter coverage of s_i is described in

the Algorithm 2.

Algorithm 2 Perimeter Coverage Algorithm

for (each s_j that $d(s_i, s_j) \leq 2r$) **do**

determine the intersection angle that is s_i 's perimeter covered by s_j ;

end for

for (each s_i) **do**

s_i sorts and combines those intersection angles to get to get a coverage range;

end for

Let m be the number of neighboring sensors ($m \leq n$ where n is the number of deployed sensors). The complexities of steps 1 and 2 are $O(m)$ and $O(m \log m)$, respectively. Since the intersection points divide the $[0, 2\pi]$ into as many as $2m + 1$ segments, the complexity of step 3 is $O(m)$. So the overall complexity is $O(m \log m)$.

Point Coverage

The objective of point coverage problem is to investigate that how well a wireless sensing network covers a set of points. The requirement is that every target must be monitored at all times by at least one sensor (1-coverage). Kar and Banerjee [6] develop an algorithm to deploy a connected sensor network so as to cover a set of points in Euclidean space. This algorithm, which assumes that $R_s(\text{Sensingrange}) = R_t(\text{Transmissionrange})$, uses at most 7.256 times the minimum number of sensors needed to cover the given point set. It is easy to see that the constructed deployment covers all of the given points and is a connected network.

Grid coverage is another version of the point coverage problem. In [7], Chakrabarty et al. presented such question as a two or three-dimensional grid of points where sensor locations were restricted to these grid points and each grid point has to be covered by at least k , $k \geq 1$, sensors. In such problem, the sensors were assumed to have

a communication range large enough to reach the base station from any grid position and do not communicate with one another. The objective was to find a least sensor deployment that provides k -coverage. Based on a K -coverage scheme we are able to monitor all grid points so long as no more than $k - 1$ sensors fail.

Barrier Coverage

Barrier coverage was firstly defined by Gage in in [4] to evaluate multi-robot systems. the objective of Barrier coverage is to minimize the probability of undetected penetration through the barrier.

In order to enhance the monitoring quality, Kumar et al.[8] combined the k -coverage problem and barrier coverage and extended barrier coverage problem to a k -barrier coverage problem where a wireless sensor network was deployed as a belt to guarantee that all crossing paths through the belt were k -covered by the sensor network. Based on probability analysis an efficient algorithm was proposed and several key results were derived, such as the optimal number of sensors for k -barrier coverage. Two notions of barrier coverage, weak and strong barrier coverage, were introduced in [8] and studied with high probability.

Definition 3 weakly k -barrier covered

A belt region is said to be weakly k -barrier covered with high probability if and only if $\forall i : \limPr[\forall j \in L(i) : j \text{ is } k\text{-covered}] = 1$, where i is a crossing path through a belt region and $L(i)$ is the set of all crossing paths congruent to i .

Definition 4 strongly k -barrier covered

A belt region is said to be strongly k -barrier covered with high probability if and only if $\limPr[\forall i : i \text{ is } k\text{-covered}] = 1$, where i is a crossing path through a belt region.

Authors pointed out that even full k -coverage could be determined locally [1] it was impossible to locally determine that a given region was k -barrier covered. Equivalence conditions were given between k -barrier coverage and the existence of k node-disjoint

paths . Based on such conditions, existed algorithms for examining k node-disjoint paths could be used to identify a k -barrier covered region. It was proved that the condition for k -barrier-covering a open belt area is equal to the problem of determining whether there was k node-disjoint paths between a pair of vertices in a graph; and the condition for a k -barrier covered closed belt region can be equal to the problem of determining whether there exist k node-disjoint cycles, each of which loops around the entire belt region. An optimal deployment pattern to achieve k -barrier coverage was showed by deploying k rows of sensors on the shortest path across the length of the belt region such that consecutive sensors' sensing disks abut each other.

Probabilistic Coverage

When nondeterministic wireless sensor network was applied in industry more and more, without clear location information some original analysis methods for coverage of deterministic network became unsatisfied in the nondeterministic wireless sensor network. In order to overcome the problem of gathering location information and reduce the cost of resolving such problem, some probability analysis methods were introduced to evaluate the performance of random deployed wireless sensor network. Such works mostly focus on the determination of coverage of random deployed wireless sensor network.

A new notation, "information coverage" was proposed by Wang et al.[9]. A point is said to be completed information-covered if there exists enough sensors to keep the estimation error lower than a predefined threshold.

Definition 5 K -sensor ϵ -error information coverage:

A point is said to be (K, ϵ) -covered if information gathered on this point can be estimated by at least K sensors such that the probability is equal to or larger than ϵ where $0 \leq \epsilon \leq 1$ and the absolute value of the estimation error is equal to or less than a constant.

A region is said to be complete (K, ϵ) -covered if all the points of the region are (K, ϵ) -covered.

To measure an information θ by K distributed wireless sensors, let $d_k, k = 1, 2, \dots, K$ denotes the distance between a sensor k and measurement location of information θ . The information θ is assumed to decay with distance. So the estimator can be expressed as following:

$$x_k = \frac{\theta}{d_k^\alpha} + n_k, k = 1, 2, \dots, K$$

where n_k is the additive noise to the measurement and $\alpha > 0$ is the decay exponent. Such estimation error could achieve a minimum mean squared error with K measurements by applying best linear unbiased estimator. Further theory and simulation proves were provided in the paper to demonstrate that information coverage can provide an approximate coverage with less density.

Worst and Best Coverage

Worst and best case coverage are used to evaluate the service quality of the WSN. In worst-case coverage, attempts are made to quantify the quality of service by finding areas of lower observability and detecting breach regions. In best-case coverage, finding areas of high observability and identifying the best support and guidance regions are of primary concern.

In [10], Meguerdichian et al. proposed an optimal polynomial time algorithm which combined graph theory and computational geometry (Voronoi diagram) to resolve the best and worse case coverage by investigating the breach paths. Authors addressed the problem as: given a field instrumented with sensors and the initial and final locations of a mobile agent to determine a maximal breach path (MBP) and the maximal support path (MSP) of such agent. The MBP (MSP) corresponds to the worst (best) case coverage and has a property that, for any point on the path, the distance to the closest sensor is maximized (minimized). The algorithm assumes that sensor nodes are homogeneous and know sensor locations. Based on the observation that MBP lies on the *Voronoi* diagram lines and MSP lies on *Delaunay* triangulation lines, the proposed algorithm starts by

generating the Voronoi diagram (or Delaunay triangulation diagram) and assigns every segment a weight equal to the distance to the closest sensor (or equal with the segment length). The path is finally computed by a binary-search or Breadth-First-Search.

Relative neighborhood graph was introduced in [11], where Li et al. proposed a distributed algorithm for MSP computation. The authors considered two extensions, namely MSP with least energy consumption and MSP with smallest path distance. Liu and Towsley [12] addressed this penetration path detectability problem in the context of random grid-based sensor networks and proposed a critical density for a given sensor network based on percolation theory.

Differently from work in the [10], Meguerdichian et al. introduced another *exposure-based model* in [13] where another important factor (the sensing time-exposure) is mentioned that the longer the exposure time, the greater the sensing ability. A 2D sensing model is defined as $S(s, p) = \frac{\lambda}{[d(s, k)]^k}$ where $d(s, p)$ is the Euclidean distance between the sensor s and the point p , and λ and k are sensor technology dependent parameters. Another characteristic is the intensity of the sensor field. For example, n sensors s_1, s_2, \dots, s_n in the field F , all-sensor field intensity for a point p is $I(F, p) = \sum_1^n S(S_i, p)$. The exposure of an object moving in the sensor field during the interval $[t_1, t_2]$ along the path $p(t)$ is defined as $E(p(t), t_1, t_2) = \int_{t_1}^{t_2} I(F, p(t)) \left| \frac{dp(t)}{dt} \right| dt$. Because the exposure analytical computation is intractable, the solution proposed uses a grid-based approach to transform the problem domain to a tractable discrete domain. The minimal exposure path in each grid square is restricted to line segments connecting any two vertices. And the grid is transformed into a weighted graph, where the weight (exposure) of an edge is approximated using numerical techniques. Finally, the Dijkstras single-source-shortest-path algorithm is used to find the minimal exposure path between any arbitrary starting and ending points on the grid. The approximation quality improves by increasing the grid divisions, at a cost of higher storage and run-time.

Another aspect of the exposure-based model was pointed out in [14]. To estimate the sensor node deployment density, one should consider both the sensor characteristics

as well as target specifications. For example, detection of an enemy tank requires less nodes due to the strong the acoustic signal, compared with soldier detection that might require more sensors. Authors assumed the target moves in a straight-line path between two given points with a constant speed. Two radiuses were associated for a given sensor : radius of complete influence defined as the distance from the sensor such that all targets originating within this radius are detected, and radius of no influence with the property that any target originating beyond it cannot be detected. Using the sensing and exposure model described above, and knowing the threshold energy $E_{threshold}$ required to detect a target, authors proposed a solution to calculate the influence radii as well as the sensor nodes deployment density.

2.3 Coverage-Preserving Algorithms

Since coverage is a fundamental criteria of wireless sensor network it is always investigated with other factors, such as energy, connectivity and through output to improve the performance of network. In this section, we review some researches which focus on optimizing the joint performance of WSN.

2.3.1 Coverage-Preserving Deployment methods

Optimal deployment is an foundation research area in the static WSN and mostly related to auto deployment or self-deployment in the dynamic WSN. The manually optimal deployment algorithms had been addressed by a lot of papers and mostly proposed in the computer geometry and computer graphics domain. In this section, we will only focus on the critical boundary of deployment and automatic optimal deployment.

Critical Boundary

A computable deployment threshold is highly required for the increasing density. Gao et al.[15] analyze the redundancy problem in WSN and provide an easy and relatively

accurate estimation on the degree of redundancy without the knowledge of location or directional information. A theoretical analysis shows that under disk sensing range and R_t (transmission range) = R_s (Sensing range), if a sensor C was fully covered, at least three neighbors and at most five neighbors are needed to cover the sensing area of C. Based on this result the probability of a completely redundant sensor on a random deployment was given as $1 - n0.609^{n-1} \leq Pr\{A\} \leq 1 - n0.609^{n-1} + \varepsilon$ where $\varepsilon = (0.276)^{n-1}n(n-1)/2$. In Boukerche et al.[16], it was proved that such result is not satisfied under R_t (transmission range) = $2 * R_s$ (Sensing range) where the low boundary of fully sponsored neighbors can be reduced to two.

To keep a sensing network k-covered, Kumar et al.[17] investigated the boundary value under grid, random and Poisson distribution. A RIS(Random Independent Sleeping) scheme was proposed based on an assigned probability p and it was shown that the network lifetime can be increased by a factor $1/p$. Other works on WSN critical density computation include Adlakha et al.[14] and Ghosh[18] that present a method to deterministically estimate the exact amount of coverage holes under random deployment using Voronoi diagrams. The method uses the static nodes to collaborate and estimate the number of additional mobile nodes needed to be deployed and relocated to optimal positions to maximize coverage.

Automatic Optimal Deployment

In [19], a greedy and incremental self-deployment algorithm for mobile sensors and self deploy is addressed. A collection of mobile sensors are placed into an unknown and potentially hazardous environment. Following the initial placement, the sensors relocate so as to obtain maximum coverage of the unknown environment. They communicate the information they gather to a base station outside of the environment being sensed.

Virtual force algorithm (VFA) is one of the major solutions for the automatic deployment problem. For a given number of sensors, VFA attempts to maximize the sensor field coverage using a combination of attractive and repulsive forces according to the

distance between each other. During the execution of the force-directed VFA algorithm, sensors do not physically move but a sequence of virtual motion paths is determined for the randomly placed sensors. Once the effective sensor positions are identified, a one-time movement is carried out to redeploy the sensors at these positions. A virtual-force algorithm to redeploy sensors so as to maximize coverage also is developed by Zou and Chakrabarty [20]. Poduri and Sukhatme [21] develop a distributed self-deployment algorithm that is based on artificial potential fields and which maximizes coverage while ensuring that each sensor has at least k other sensors within its communication range. Another distributed potential-field-based algorithm to self deploy mobile sensors under the stated assumptions is developed in [22].

2.3.2 Energy-efficient Coverage

In the dynamic wireless sensor network the sensors can not be settled manually. To keep sensing quality the density of network is highly increased and leads to higher communication overhead and energy consumption. A set of power saving coverage solutions were developed facing such requirements. Because the random deployment of sensors might lead to redundant nodes that share one same area, in order to extend network lifetime while meeting coverage, a common solution, *Duty-cycle scheme*, is proposed to have an optimal number of nodes to be active and turn off redundant neighbor nodes.

Methods for Identifying Fully-sponsored Sensors

One of important research areas of off-duty scheme is how to identify the fully sponsored sensor. In paper [1] and [23], authors present a simple method to determine a fully sponsored node by calculating the center angles of its neighbors. If such center angles can cover the whole 360° (as the gray cycles B, C and D shown in Figure 2.2), node A can then be determined as *fully sponsored*. Because the CAM presented in the [23] can not identify all fully sponsored sensors by only considering those neighbors in the sensing range, an extended method is presented in [16], named association sponsor method(ASM). ASM

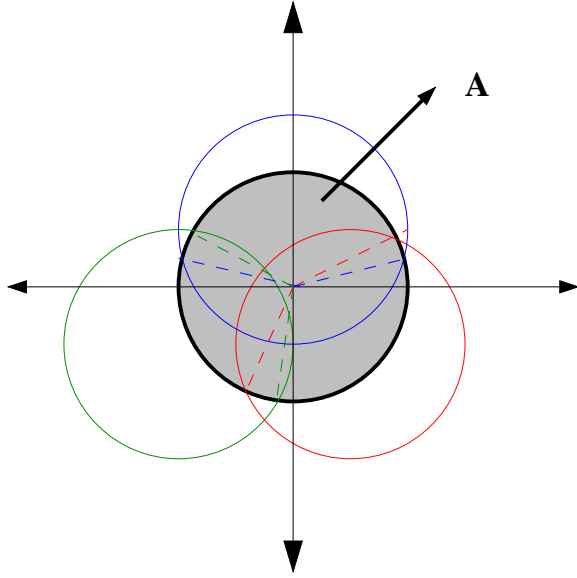


Figure 2.2: Central angle method

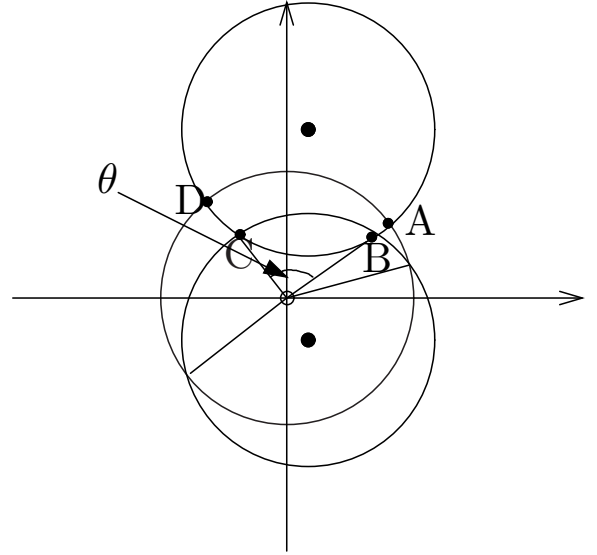


Figure 2.3: Association sponsor method

remarkably increases the number of off-duty sensors by increasing the transmission range to twice of the sensing range. So all the neighbors which have overlap area will be considered. As shown in figure 2.3, ASM establishes an association relationship between the high overlapped and low overlapped sponsors to identify more fully sponsored sensors.

Node Selection

The main object of off-duty scheme in the wireless sensor network is to divide the deployed sensors into disjoint sets while every set completely covers all targets. These disjoint sets are activated successively, such that at any moment only one set is active. The goal of this approach is to determine a maximum number of disjoint sets, so that the time interval between two activations for any given sensor is longer. By decreasing the fraction of active time for sensor, the lifetime of sensors increases, and the lifetime of application is extended proportionally by a factor equal to the number of disjoint sets.

A probing-based, node-scheduling solution for the energy-efficient coverage problem was proposed in [24] by Ye et al. In such protocol, all sensors are characterized by the

same sensing range, and coverage is seen as the ratio between the area under monitoring and total size of the network field. The off-duty eligibility rule is based on a probing mechanism. The protocol works as follows: A sensor broadcasts a probing message PRB within a probing range r_p . Any active node that hears such message will reply with a PRB_RPY message. If at least one reply is received, the node enters the sleep mode. Probing range is selected based on the desired working node density (number of sensors per unit area) or based on the desired coverage redundancy, whereas the wake-up time is based on the tolerable sensing intermittence. This protocol is distributed, localized, and has low complexity but does not preserve the original coverage area.

In [25], Cardei and Du proposed a protocol, which partitioned the set of available sensors into disjoint dominating sets such that each set covered all targets in each round. Such dominating sets are modeled in an undirected graph, where sensors form the vertex set and any two neighbors (sensors within each others sensing range) are joined by an edge. Let T be the set of targets to be monitored and let S_i denote the subset of T in the range of sensor i . Let $\{p_1, \dots, p_k\}$ be disjoint partitions of the set of n sensors, if $\cup_{j \in p_i} S_j = T$ the set of sensors in each P_i covers all targets. We refer to the set $\{p_1, \dots, p_k\}$ as a disjoint set whose cover size is k . As shown by the authors the maximum disjoint dominating sets computation is NP-complete, and any polynomial-time approximation algorithm has a lower bound of 1.5. A graph-coloring mechanism is proposed for computing the disjoint dominating sets. Disjoint sets are firstly formed by coloring all nodes, and then applying the sequential coloring algorithm. Then, each non-dominating set is considered in an increasing color number and transformed into a dominating set by recoloring a smallest number of higher-color vertices. When this process ends and no more dominating sets can be formed, the remaining nodes are added to the sets where they have the greatest contribution in covering parts of the uncovered given area. Simulations have shown that the number of sets computed is between 1.5 and 2 times greater than the algorithm in [26], with lapses in area coverage less than 5%, on average. However, the algorithm of Cardei and Du takes more time to execute and

has to fetch the global location information through flooding across the whole network. To avoid the requirement for global information, some interesting works were reported based on node scheduling.

Another energy-efficient node-scheduling-based coverage mechanism was proposed by Tian and Georganas in [23]. The protocol proposed is distributed and localized. The node scheduling scheme is divided into rounds, where each round has a self-scheduling phase followed by a sensing phase. In each self-scheduling phase, the nodes investigate its status by computing its sponsored angles with centre angle methods while an eligibility rule is applying to resolve the conflicts between the candidates. The eligible sensors will be turned off and wakeup until next rounds. To avoid eligible sensors make decision simultaneously which will lead to the blind points, a back-off scheme is used, where every node starts the evaluation rule after a random time, and then broadcasts a status advertisement message (SAM) to announce if it is available for turning off. Before turning off, a node waits another T_w time to listen the SAM from neighboring nodes. The whole work of such scheme is based on that global time synchronization mechanisms is available. It is implemented as an extension of LEACH protocol[27], and simulation results show an increase of 1.7 on average in system lifetime. However, the backup scheme can not guarantee that all breach holes are avoided.

To avoid those holes caused by inappropriate time schedule A. Boukerche and X. Fei proposed an alternate election algorithm based on their association sponsor method and local information exchanging [28]. In *alternate election* algorithm only candidates who receive stable status announcement messages will be involved in the election. The algorithm is separated into *odd* and *even* rounds and begins with the SAM (status acknowledge message) from the on-duty nodes (edge nodes) after identifying all of the candidates and On-duty nodes in the initial phase. Two new status, *On-duty candidate* and *Off-duty candidate*, are introduced as temporary statuses.

An odd round is triggered when a *P-Off* node A receives either a SAM message from all *P-On* neighbors, or SAM messages from all neighbors at the zero level. If node A is

the node selected by the election algorithm, it will be set to *Off-duty*; otherwise, it will be set as *P-On*. In the odd round, only the Off-duty election will be performed. Any candidate A that receives a SAM will check the distance between itself and the sender. If this distance is smaller than sensing range r , the status of the sensor is set to *Off-duty Candidate*; otherwise, it is set to *On-duty Candidate*.

An even round will be triggered when a *P-On* candidate A receives a SAM message from all of its *Off-duty* candidate neighbors. The center angles of candidate A are recalculated using the new status information gathered from its *P-Off* neighbors. If candidate A cannot keep its candidate status, it will be set as *On-duty*; otherwise, it will be set as *P-Off*. In the Even round, any candidate that receives a SAM will update its status to *P-On* and execute the *Extended Coverage Calculation Method* to find out whether the candidate is a qualified On-duty node. The status update rule is that the status only updates by means of the high-level SAM message.

Wake-up Strategy

As an inseparable part of an off-duty scheme wakeup strategy is an important issue in the coverage problem. It is mostly concerned as a research on the physical level even it highly effects the performance of protocols on the energy saving issue. Some power management schemes require that each node wakes up periodically to listen to the radio channel such as Hui et al.[29]. When an event of interest happens, some nodes detect the event and send power management messages to the network. The nodes in the listening mode wake up after received such power management messages. By choosing a good wake-up/sleep schedule, the network may save much energy without compromising the system functionality. The implementation of the wake-up/sleep scheduling often involves a timer that wakes up the CPU via an interrupt. However, the design of wake-up/sleep schedule is often application-dependent and complicated. The time synchronization service is always required.

Another approach is to use a low-power stand-by hardware component such as a radio-

transceiver subsystem listening a special radio channel when nodes go sleep. If the stand-by radio transceiver receives a wake-up radio signals, it wakes the node up. PicoRadio has such functionality [30]. It separates the radio hardware for data communication and channel monitoring. A separated low-power radio, called the *wake-up radio*, monitors the radio channel and wakes up the node when a power management beacon is received. The disadvantage of a stand-by component is that it uses extra energy to keep listen the power management messages.

[31] proposed another power management approach, which used the energy in the event (such as the wakeup message) to trigger the transition of the system from sleep mode to wake-up mode. A special hardware component, radio-triggered circuit, is connected to one of the interrupt inputs of the processor. The circuit itself does not have any power supply. When a power management message is sent by another node (possibly a sentry node) within a certain distance, the radio-triggered circuit collects enough energy to trigger the interrupt to wake up the network node. So, the node can enter sleep mode without periodic wake-up.

2.3.3 Connectivity-preserving Coverage

Besides the coverage, connectivity is another important factor for evaluating the quality of WSN. Several solutions have been proposed to guarantee connectivity (Ascent [32], Span [33], Gaf [34]). However, connectivity alone does not guarantee coverage. We often desire that once the sensors are deployed, they organize into a network that must be connected and cover the monitor area as well as possible so that the information collected by sensors can be relayed back to data sinks. An important and frequently addressed issue is to determine a minimal number of working sensors required to maintain the initial coverage area as well as connectivity. So, the power consumption is reduced by less transmission and network lifetime is prolonged. In this section we will present several connected coverage schemes.

Zhang and Hou [35] have established the following necessary and sufficient condition

for coverage to imply connectivity.

Theorem 1 *When the sensor density (number of sensors per unit area) is finite, R_t (transmission range) $\geq 2R_s$ (sensing range) is a necessary and sufficient condition for coverage to imply connectivity.*

Another important issue mentioned by [35] is that an area is completely covered if there are at least two disks that intersect and all crossings are covered. Here, a disk refers to a nodes sensing area, and a crossing is an intersection point of the circle boundaries of two disks. In the ideal case, in which node density is sufficiently high, the full coverage can be obtained by optimally placing the subset of working nodes at the vertices of regular hexagonal plane tiling.

Wang et al. [36] presented a similar result for the cases of k -coverage (each point is covered by at least k sensors) and k -connectivity (the communication graph for the deployed sensors is k connected).

Theorem 2 *When R_t (transmission range) $\geq 2R_s$ (sensing range), k -coverage of a convex region implies k -connectivity.*

Based on the Theory 2, authors proposed a distributed, localized algorithm, called optimal geographical density control (OGDC). In the OGDC a node can be in one of the three states: *UNDECIDED*, *ON* and *OFF*. The algorithm runs in rounds, and at the beginning of each round a set of starting nodes is selected as working nodes. After a back-off time, a starting node changes its state to ON and broadcasts a power-on message which contains location of the sender and a direction along which a working node should be located. The direction indicated by the power-on message of a starting node is randomly distributed. Having starting nodes randomly selected at the beginning of each round ensures uniform power consumption across the network. To avoid packet collisions a back-off mechanism was developed.

At the beginning of each round, the statue of every node is UNDECIDED and will change to ON or OFF state by the power-on messages received. When a node receives a

power-on message, it checks whether its sensing area was fully sponsored by its neighbors, and if so, it will change to OFF state. Otherwise, it will change to the ON state if it is the closest node to the optimal location of an ideal working node selected to cover the crossing points of the coverage areas of two working neighbors. The simulation experiments based on NS-2 show good results in term of coverage rate and system lifetime.

Based on the theory 1, [36] integrated their coverage configuration protocol (CCP) with SPAN [33] to support both coverage and connectivity. SPAN is a well-known distributed algorithm that conserves energy by applying off-duty scheme while maintaining connectivity. The combined eligibility rule is as follows:

- (1) A sleeping node wake-up if it satisfies the eligibility rules of SPAN and CCP;
- (2) An active node go to sleep if it satisfies neither the eligibility rules.

With the eligible rules applications can obtain k-coverage through CCP and 1-connectivity through SPAN.

In the paper [37], author presented another connectivity-aware coverage solution in which coverage was achieved through a probing mechanism which controlled the network density. In this algorithm, a node can be in one of three states: sleeping, wake-up and working - when a sleeping node wakes up (after an exponentially distributed period of time), it broadcasts a probing message within a certain range and waits for a reply. If no reply is received within a timeout, it will take over the surveillance task continuously (until it runs out of battery). In this solution, the probing range and wakeup rate can be adjusted to affect the degree of coverage indirectly. However, this solution does not guard against blind points since there is no guarantee on sensing coverage [38].

Because in some cases, the $R_t = R_s$ assumption can efficiently reduce the complexity of algorithms, Carle and Simplot [39] proposed another mechanism for energy-efficient connected area coverage. The main idea of such algorithm is to select a connected area-dominating set of nodes of minimum cardinality to covers the given area. Carle's algorithm relies on other protocols such as Tian and Georganas's work [23] to decide the status of sensors. When a sensor decides its status a status message is sent to all its

neighbors. After that, the protocol runs as follows:

- (1) A node p determine whether its monitoring area is fully sponsored by its neighbors;
- (2) After a back-off interval, node p computes a subgraph of its one-hop active neighbors;
- (3) If the subgraph is connected and fully covers ps area, then node p will be in the sleep mode, otherwise it will be in the active mode during the next round.

Chapter 3

Coverage Estimation Based on Irregular Sensing Range Model

Sensors can be considered as a type of transducer because they gather signals from environment and change them into another energy form. Therefore, we can classify sensors according to the energy type of signal, such as temperature sensors, electromagnetic sensors, mechanical sensors, chemical sensors, optical radiation sensors, acoustic sensors, and so on. The sensing model of each sensor highly depends on its category. In the majority of the research concerning the coverage problem of WSN, two sensing models are well adopted, boolean sensing model (BSM) and probability sensing model (PSM). The former model assumes that a sensor can cover a disk area centered at itself with a radius equal to its sensing range. The BSM highly simplifies the coverage problem, however, if we consider the effect of path loss and absorption caused by obstacles, the sensor can not maintain its disk sensing range unless it is working on a completely flat and obstacle-free area. Therefore, BSM based application will lead to unpredictable detection errors in the real scenarios.

The probability sensing model reflect the reality better than the boolean disk sensing model. However, probability sensing model mainly focuses on digging the sensing ability of sensors; the weakening caused by environment factors such as obstacles is not

concerned. For examples, the view of image sensor could be changed by obstacles in the range. In the probability sensing model, path loss is defined by Gaussian random variable. In order to reflect the characters of environments, β and σ of Gaussian distribution have to be assigned after careful in-field experiments. Therefore, when the environment, such as volcano or battle fields, changes frequently, those predefined parameters will lose their justices. Especially, for some disposable applications such as emergency systems, there is no way to predetermine the parameters of probability model. The presented issues motivate our research: “Can we ignore the characteristic of sensing module and determine the sensing range online? Therefore, the coverage related applications can notice and adapt to the changes of environment.”

In the following of this chapter, we present a new irregular sensing model(ISM) and the α -shape based method to determine the sensing range online.

3.1 Analysis of the Sensing Range Modeling

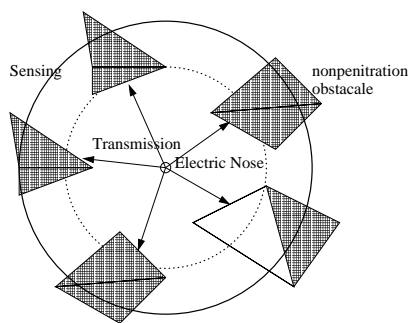


Figure 3.1: Electric nose surround by non-penetrable obstacles

In the majority of researches that concern the coverage problem of WSN, we assume that in the ideal environment the sensing range of wireless sensor is circle. Some papers even extended this assumption by equaling the sensing range to the transmission range. However, an important fact is ignored: the characteristics of sensing and transmission equipment are completely different and rely on application requirements. Let us use the

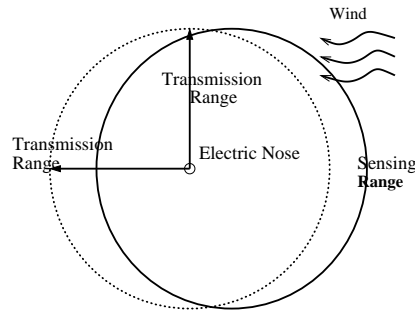


Figure 3.2: Electric nose in windy grassland

electric nose as an example, which is used to detect the chemicals volatilized in the air. We assume that the sensing range of the electric nose in the ideal environment is circular. When such an electric nose is deployed in a mountain area and is surrounded by huge stones, as show in the Figure 3.1, there is not doubt that the transmission range will be seriously affected while the sensing range will not be affected. When we move this scenario to a windy grassland, as show in the Figure 3.2, wind become a primary factor in affecting the sensing range of the electric nose. Wind will evidently change the sensing range of the electric nose but will have not effect on its transmission range. When the wind comes from different directions the sensing range of the electric nose will have a different shape.

It is obviously that the factors that affect the sensing range are different from the factors that affect the transmission range. Therefore, in the real world, analysis under the assumption that the sensing range is equal to transmission range, is not suitable. In other words, we must redefine the factors that will affect the sensing range and distinguish them from the factors that affect transmission. In order to simplify the analysis, we abstract the reasons for weakness on the sensing range as Fading Factors and denote the Fading Factors on certain angles as [fade strength, angle]. The sensing range will be decreased in the direction in which such fading factors exist. By detecting the fading factors on the different directions, we can use a set of such factors to describe the sensing range in the real world. Such a set of fading factor is denoted by FF . Depending on the detection

technology of the sensing equipment, the real sensing range described by the FF is more similar to a polygon than to a disk.

Because the detection methods of fading factors are changed for different sensing applications and rely on the related technology, we select typical localization sensor such as ultrasonic sensor and laser scanner as our sensors and consider the scenario where the main reason for fading on the sensing range is caused by non-penetration.

Ultrasonic sensors determine distances to objects by sending and receiving ultrasonic signals. To estimate the distance between objects and sensor two different approaches were commonly used. The time-of-flight method uses pulses of energy, for which the time between sending and reception is estimated. The reflected signal is compared to the reference signal. And the distance is computed by the relative phase shift of those two signals. Since ultrasonic waves are likely to be reflected by smooth surfaces, the distance measurements are not very accurate because of multiple reflections.

Laser scanners work in a manner similar to ultrasonic sensors but use a laser beam instead of ultrasonic waves. In particular, the two distance measuring techniques for ultrasonic sensors described above also exist for laser scanners: the time-of-flight technique and the phase-shift measurement technique. Furthermore, a frequency modulated technique also exists as an alternative. In this model the frequency of the transmitted continuous wave varies linearly with time. The three methods are described in detail by Adams [40]. Because of shorter wave lengths, laser scanners are normally more precise than ultrasonic sensors in distance measurements. So, laser scanners are extremely suitable for localization applications which are based on range sensing. A typical laser scanner takes a 180° scan of its environment with an angle increment of 0.5° within a range of approximately $80m$. The typical resolution scan points is about $10mm$.

When such two kinds of sensors are deployed indoor or outdoor, they can detect the fading of the sensing range at every angles to meet the reality, as shown in the Figure 3.3 and 3.4. Based on the above analysis we may say a coverage scheme based on a polygon sensing range is very useful for the reality analysis.

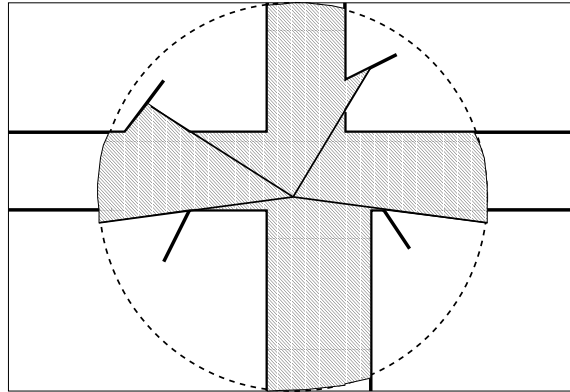


Figure 3.3: The indoor polygon sensing range

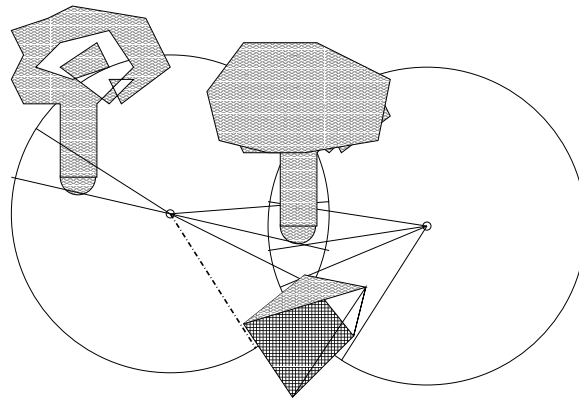


Figure 3.4: The outdoor polygon sensing range

3.2 Shaping Based Irregular Sensing Model

Sensor collects the energy in the environment and converts them into logical samples. If we assume that all samples observed by sensors contain the location information of event and all the obstacles in the field are as a set of simple polygons $S_{ob} = \bigcup_i P_i$, then we can determine the range of sensors using shaping methods of geometry. The general form of irregular sensing model can be presented as following:

For a sample $x = (d, \theta)$,

$$S_{ob} = \bigcup_i P_i$$

$$P(x) = \begin{cases} P(d) & x \notin S_{ob} \\ 0 & x \in S_{ob} \end{cases}$$

Or,

$$P(x) = \begin{cases} P(d) & x \in SP_j \\ 0 & x \notin SP_j \end{cases}$$

where SP_j is the boundary polygon of sensing range of sensor j .

3.3 Determining the Sensing Range Using α -Shape

In both BSM and PSM, algorithms have to know the capability of sensors and characteristic of environments to investigate the coverage of WSN. In order to remove such a constrain, instead of using the capability awareness assumption, we assume that sensors have the capability to know the position of gathered events. Through adding such an assumption, we can determine the holes caused by obstacles or some other reasons without the knowledge of sensing capability and the characteristic of environment. The range detection problem is converted into a boundary recognition problem or shaping problem (a classic computer geometry problem which determines the shape from a set of sampled points).

α -shape was introduced as a tool for reasoning the shape of a set of points by Edelsbrunner in 1983 [41]. In the following sections, we first review the concept of α -shape and then revise it to efficiently generate the boundary of sensing range. The following definitions serve as the fundamental of our algorithms.

Definition 6 *Obstacle: An obstacle for wireless sensor network refers to a closed point set O on 2D plane. Any energy signal propagating through such a point set will be 100% absorbed.*

Definition 7 *Face:* A face of obstacle is a polygon by which an obstacle is bound.

Definition 8 *Boundary Set:* A boundary set R_b is a closed subset of R that for each point x belong to R_b there is at least one neighborhood point of x is not belong to R .

Definition 9 *Internal Boundary and External Boundary:* If the center of boundary polygon and at least one generalized disc in the same side of boundary edges, we name such a boundary polygon as an internal boundary, otherwise, it is an external boundary.

3.3.1 Review of α -shape

An α shape is a concrete geometric object that is uniquely defined for a particular point set. It is the generalizations of the convex hull and a subgraph of the Delaunay triangulation. Given a finite point set S , and a real parameter α , the α -shape of S is a polytope which is neither necessarily convex nor necessarily connected. The real parameter, “ α ”, controls the desired level of detail. The set of all real α values leads to a whole family of shapes capturing the shapes of a point set. The definitions of terms of α -shape and pseudo code of the construction process are shown as following:

Algorithm 3 α -shape Construction Algorithm

- 1: **procedure** α -SHAPE(x) $\triangleright x$ is the location of sample
 - 2: Incrementally construct Delaunay Triangulation(DT)
 - 3: select the α
 - 4: generate the generalized discs
 - 5: Scan the edges of delaunay triangulation
 - 6: Determine the α -extreme points of S
 - 7: Determine the α -neighbors of S
 - 8: Output the *alpha*-shape
 - 9: **end procedure**
-

Definition 10 *generalized disc*: A generalized disc of arbitrary real value α is the standard disc of radius $1/\alpha$ if $\alpha > 0$, or the complement of a disc of radius $1/\alpha$ if $\alpha < 0$. If $\alpha = 0$, the generalized disc is the half plane.

Definition 11 α -hull: The α -hull of a planar set S is composed by the intersection of all closed generalized discs of radius $1/\alpha$ that contain all the points of S .

Definition 12 α -extreme: A point s is defined as an α -extreme if s lies on the boundary of a closed generalized disc of radius $1/\alpha$ which contains all other points of S .

Definition 13 α -neighbors: A pair of α -extreme points s and s' are α -neighbors if s and s' are on the boundary of a closed generalized disc of radius $1/\alpha$ containing all other points of S .

Definition 14 α -shape: The α -shape is the line graph where the vertices are the α -extremes and the edges connected α -neighbors.

The α -shape describes the shape of obstacles better than the standard convex hull because it can show internal boundaries and the breaches on the external boundary. However, the quality of α -shape depends on the selection of α . Considering the Figure 3.6, when $1/\alpha$ is small ($\alpha < 0$) some boundary points are isolated and excluded from the boundary polygon. In order to avoid the disconnection, the selection of α is a key challenge of α -shape.

For simplifying the problem, we only consider the situation where $\alpha \leq 0$. Based on the definition of α -extreme point, when $\alpha = 0$ the external boundary is a convex hull of the point set S while there is no internal boundary. By decreasing the $1/\alpha$, we can shrink the generalized disc to find the internal boundary.

3.3.2 Determination of α

In order to find the α which can determine the internal boundary and do not isolate samples, we need review the construction process of α -shape. When the generalized

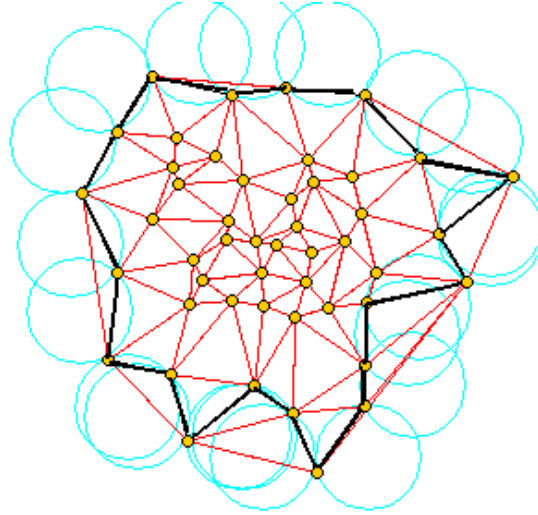


Figure 3.5: Alpha shape

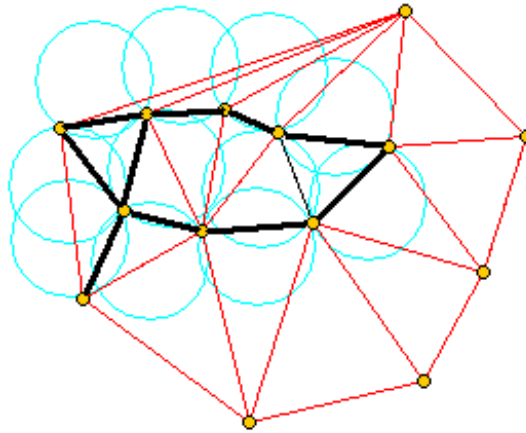


Figure 3.6: Inappropriately selected alpha

disc scans every Delaunay edges, edges will not be outputted if none of their ends is an α -extreme. Therefore, the construction process of α -shape can be considered as an edge-removing process. If we define the edge set adjacent with boundary point i as EA_i , then in order to avoid isolating samples, the radius of generalized discs $1/\alpha$ have to satisfy the following condition.

$$\frac{1}{\alpha} \geq \text{Max}(\text{Min}(EA_1), \text{Min}(EA_2), \dots, \text{Min}(EA_k)) \quad (3.1)$$

Assuming that an internal hole h exists. Considering any edge e on the internal boundary polygon of h . In order to keep both ends of e as α -neighbor, the α have to satisfy to following condition:

$$1/\alpha_e \leq \text{Max}(r_{left}, r_{right}) \quad (3.2)$$

where r_{left} and r_{right} are the radius of left and right circumstances of e . In order to avoid isolated boundary samples while detecting the hole h , the α have to satisfy both conditions, however, the length of EA_i , r_{left} and r_{right} highly rely on the distribution of samples. The following condition can not always be maintained. An optimized α has to trade off detection accuracy on the internal and external boundary.

$$\text{Max}(\text{Min}(EA_1), \text{Min}(EA_2), \dots, \text{Min}(EA_k)) \leq 1/\alpha_e \leq \text{Max}(r_{left}, r_{right}) \quad (3.3)$$

3.4 Revised α -Shape Algorithms for Sensing Range Estimation

Edelsbrunner had pointed out that we can increase the α on some area where the density of samples higher than other place [41]; however, he did not present how the weighted α -shape works. In this section, we propose our revised weighted α -shape solution including a weighted α -shape algorithm and a boundary ring based α -shape algorithm.

3.4.1 Weighted α -shape

Let us consider the Delaunay triangulation in the Figure(3.7) where the internal polygon presents an obstacle in the field. The delaunay triangles on the edges of obstacle and boundaries, such as A and B in the Figure (3.7), appear different from others. Considering an edge e of internal boundary. Because e is a boundary edge, its two Delaunay triangles site on its both sides and one of its Delaunay triangles will cross the breach area. Since the samples falling in the breach area is unobservable, when more samples are

gathered the size of the Delaunay triangle crossing the breach area will decrease slower than the other one. In other words, the ratio of the size of two Delaunay triangles of edge e indicates whether e is a boundary edge. Therefore, instead of looking for an optimized global α or dividing area into grids, we are looking for those edges whose two Delaunay triangles have big difference in terms of size. We give a weight for such edges and adjust the α locally.

Definition 15 Weight of Edge:

The weight of an edge e is the ratio of $r_{e_{\text{circum-left}}}$ and $r_{e_{\text{circum-right}}}$.

$$weight_e = \frac{\max(r_{e_{\text{circum-left}}}, r_{e_{\text{circum-right}}})}{\min(r_{e_{\text{circum-left}}}, r_{e_{\text{circum-right}}})} \quad (3.4)$$

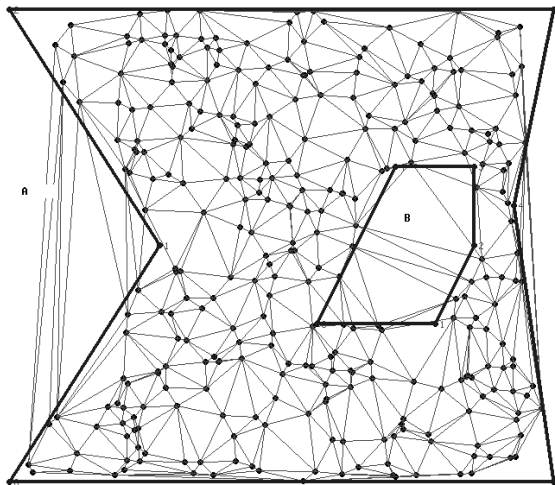
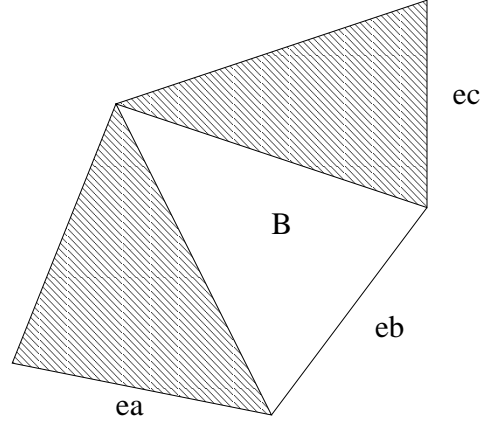


Figure 3.7: Delaunay triangulation, 300 samples

If $weight_e \geq W$ where W a predefined parameter, then we consider the edge e as a potential boundary edge. The radius of generalized disc for edge e is $1/(|\alpha * weight_e|)$.

The weighted edge method can quickly identify those boundary edges with large weight when the number of event goes high; however, some boundary edges do not change their weight fast especially for those boundary edges around the corners of obstacles. In order to avoid such problem, we add the effects of adjacent boundary edges to the weight of edge as shown in Figure 3.8.



- eb boundary edge
- ea $\text{weight}=\text{weight_ea}*\text{weight_eb}$
- ec $\text{weight}=\text{weight_ec}*\text{weight_eb}$

Figure 3.8: Weight of edge

Definition 16 Revised Weight of Edge:

The weight of an edge e is the ratio of $re_{\text{circum-left}}$ and $re_{\text{circum-right}}$ multiple the weight of adjacent boundary edge.

$$\text{weight}_e = \frac{\max(re_{\text{circum-left}}, re_{\text{circum-right}})}{\min(re_{\text{circum-left}}, re_{\text{circum-right}})} * \text{weight}_{\text{boundary-neighboring}}$$

Because such edge weight based method only rely on the information of edge and its adjacent edges, it could easily changed to an distributed algorithm.

3.4.2 Boundary Ring Based α -shape

The complexity of constructing an α -shape is mainly contributed by the complexity of algorithm for constructing DT, which varies from $O(n \log n)$ to $O(n^2)$. It is obviously that when we gather more samples for a better boundary detection accuracy the computing time will increase with the number of samples. Fortunately in some special cases, even we can not reduce the complexity of algorithms we can reduce the input size (n) while

Algorithm 4 Weighted α -shape Construction Algorithm

```

procedure WEIGHTED  $\alpha$ -SHAPE( $x$ )  $\triangleright x$  is the location of samples
2:   Construct Delaunay Triangulation(DT)
      Compute the lowest  $\alpha$  for not isolating external boundary samples
4:   Computing weight of every edges
      Scan edges with  $r_{gen} = 1/(|\alpha * weight_e|)$ 
6:   Determine the  $\alpha$ -extreme points of S
      Determine the  $\alpha$ -neighbors of S
8:   Output the weighted alpha-shape
end procedure

```

maintaining the accuracy of boundary detection. In here we only consider the effects of absorbing and path loss in the signal propagation. An obstacle appearing in the area will absorb any phenomenon signal passing through it. If the obstacles are considered as simple polygons, the boundary of sensing range is a simple polygon that does't not contain any holes.

Since every α shape is a subset of a Delaunay triangulation the construction of α shape starts from building a Delaunay triangulation or Voronoi diagram. Considering the incremental characteristic of sampling process, the incremental construction algorithm of DT is more suitable than other construction algorithms. Therefore, we select the incremental algorithm to construct the Delaunay triangulation for boundary detection and try to reduce the computing time by reducing the size of input. A classical incremental algorithm for Delaunay triangulation starts from point insertion. The old edges are verified by an circumcircle. The unsatisfied edges will be flipped. In the worst case, all the old edges are violated the empty circumcircle condition and cause $O(n)$ flips where n equal to the number of edges. However, in practice, the number of average edge tests per insertion is far less than $O(n)$. Considering the α shape shown in Figure 3.9. Since there is no inner hole the insertion of most samples in the sensing disk does not change the outer boundary of α shape. Therefore, we do not have to construct the De-

launay triangulation for all the samples in the sensing disk. We can ignore some samples in the middle of shape and use only a set of boundary samples to build the sensing range polygon.

Our algorithm bases on the following three definitions.

Definition 17 Boundary Triangle: *A boundary triangle is a Delaunay triangle having at least one vertex or edge belong to the boundary polygon.*

Definition 18 Boundary Ring: *A boundary ring is a subset of Delaunay triangles that are boundary triangles or adjacent with boundary triangles.*

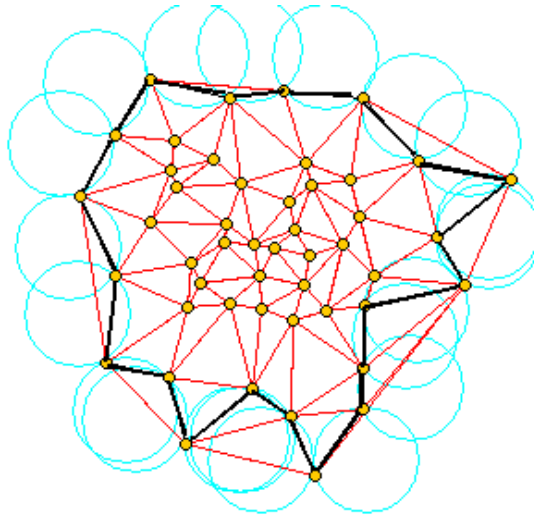


Figure 3.9: Boundary ring

Definition 19 Width of Boundary Ring (R_w): *The width of boundary ring is defined as $R_w = R_{farthest} - R_{nearest}$ where $R_{farthest}$ and $R_{nearest}$ are the distance from sensor to the farthest sample points and nearest sample points in the boundary ring.*

The boundary ring based α - shape starts from constructing an alpha hull and then constructing the boundary ring. Each time when a new sample is inserted, we will check whether such insertion will cause a flip in the boundary ring. If there is no flip in the

boundary ring the sample will be discarded. The number of Delaunay triangles will remain same. Otherwise, the sample will be inserted and the Delaunay triangles and the boundary-ring will be updated.

As we know the algorithm of creating α -shape is a global partition process which removes ineligible DT edges from entire graph. When samples are not normally distributed in the sensing disk or the number of samples is not high enough, parts of boundary may have enough samples to detect the breach, others may not. If we apply an α on the entire sensing disk, we may lost the connectivity of boundary points. Therefore, we introduce the concept of weight of boundary edge which only let the area that have sufficient samples do the job of removing DT edges.

Definition 20 Weight of Boundary Edge:

The weight of a boundary edge e_b is the average length of the Delaunay edges e_d which incident to e_b 's vertices and are not on the boundary polygon.

Because samples falling in the area of obstacles will not be sensed by any sensor, an interesting phenomenon can be easily observed in the simulations, that the samples surrounded the obstacles keep dividing those boundary Delaunay triangles into smaller pieces. Therefore, the weight of boundary edge e_b will keep decreasing. Based on such observations, we can use the weight of boundary edge to determine whether a boundary edge should be removed. The algorithm for removing boundary edges can be presented as following:

If the insertion of a sample will change the Delaunay triangle containing a boundary edge e_b , we compute the weight of e_b .

If the weight of e_b is less than $edge/\delta < 1/2$ while the remove will not lose boundary points, e_b will be removed.

If the weight of e_b is less than $edge/(2 * \delta)$ while the remove will cause the degree of an boundary vertex less than two, e_b will be removed and the boundary vertex will be abandoned.

3.4.3 Complexity of Boundary-ring Based α Shape

Comparing to the original α shape the complexity of boundary-ring based α shape relies on the number of sensors fall in the boundary-ring. Assuming that distribution of sample X follows normal distribution; the probability of $distance(X, sensor) < r$ is proportional to the area of sensing disk. The distribution function ($F(r) = P(X \leq r)$) of samples can be presented as following:

If $r < 0$, $F(r) = 0$; If $r \geq R$, $F(r) = 1$;

If $0 \leq r < R$, $F(r) = P(x \leq r) = k\pi r^2$;

Because $F(R) = k\pi R^2 = 1$, we have $k = 1/(\pi R^2)$.

The distribution function of sample X is:

$$F(r) = \begin{cases} 0, & r < 0; \\ \frac{r^2}{R^2} & 0 \leq r < R; \\ 1 & r \geq R \end{cases}$$

By dividing the whole sensing disk into k rings where $k = R/R_w$ we have the probability of sample falling into the $k - i$ ring :

$$\begin{aligned} P\left(\frac{i}{k}R < X < \frac{i+1}{k}R\right) &= F\left(\frac{i+1}{k}R\right) - F\left(\frac{i}{k}R\right) \\ &= \left(\frac{i+1}{k}\right)^2 - \left(\frac{i}{k}\right)^2 = \frac{2i+1}{k^2}(i = 0 \dots k-1) \end{aligned}$$

Then the probability of the sample falling in the k -th ring (boundary ring) is :

$$\frac{2i+1}{k^2} = \frac{2k-1}{k^2} = \frac{2R_w R - R_w^2}{R^2} \quad (i = 0 \dots k-1)$$

Since the probability of insertion decreases with the width of boundary ring the probability of insertion approaches zero when the R_w is very small.

Boundary ring based α -shape has lower complexity than the original α -shape. It is suitable for the range detection when the inner hole of sensing disk is not concerned.

Algorithm 5 A Distributed Coverage Evaluation Protocol

procedure SENSORS()

Gather events

Generate the boundary-ring

 Determine α

Exchange the boundary edge set with single hop neighbors

Add the negative event from neighbors into event set

 Refine the α -shape

Breach detection

send the boundary edge set to sink

end procedure**procedure** SINK()

Reassemble the edges from sensors

end procedure

3.5 The Distributed Coverage Evaluation Protocol

Using our shaping based sensing model, each sensor determine its external boundary through gathered events and broadcasts its boundary edge set to the single-hop neighbors. When an event happens in an obstacles-enabled overlapping area, the sensors on the other side of obstacles can not received signals; however, the sensors at the same side of event will detect it and report such event to the sink. Therefore, if sink does not receive the event reported from other sensors in the same area, the location of such an event will be marked as a lost point and sent to the sensors who do not receive the signals because of the obstacles. When a sensor receives the lost point it will insert such point in to boundary ring. A test path from sensor to the lost point is drawn and extended to the outside of boundary polygon. Any Delaunay edges after the lost point will be removed if they have intersection with test path. The lost point will be removed from the boundary ring until an event sample is inserted to its Delaunay triangle.

A global rebuild process only happens after t seconds or a global weakening happens. In order to discover a global weakening we assume every sensor carries an event generator module. Each sensor periodically generates a detection event and message. Each sensor partitions its sensing range by its neighborhood. When a sensor receives a detection message it will try to detect the event generated by sender. If the detection fails frequently the samples in the partition will be discarded and the boundary ring at the partition will be reconstructed. The pseudo code of protocol is shown as in Algorithm 5.

3.6 Simulation Experiments

In order to evaluate the performance of our sensing model and detection protocol, we establish an obstacle model to simulate our work. The obstacle is modeled as an irregular simple polygon. It absorbs any phenomenon signals passing through it. As shown in Figure 3.10 and 3.11, two types of polygon are selected to show the performance of boundary shaping in our simulation. Sensors are randomly deployed in a $40m * 40m$ square area. The number of sensors varies from 5 to 20. The ideal sensing range of sensors is 10m. Uniformly distributed events are inserted into the area sequently. The number of events varies from 50 to 2000.

The performance of our protocols are evaluated in terms of four concepts, boundary recognition, coverage rate, coverage errors and times of insertion.

3.6.1 Boundary Recognition

Instead of composing the α -shape for entire space, irregular sensing range model construct boundary ring for each sensor. In the following figures, we only keep the samples in the boundary ring of each sensors. As shown in the Figure 3.10 and 3.11, the shapes in the middle are well recognized; however, we also notice that the the vertex of obstacles are blurred. On the right of up-vertex of start shape, we also notice that a small piece of area is recognized as uncovered because an edge is removed.

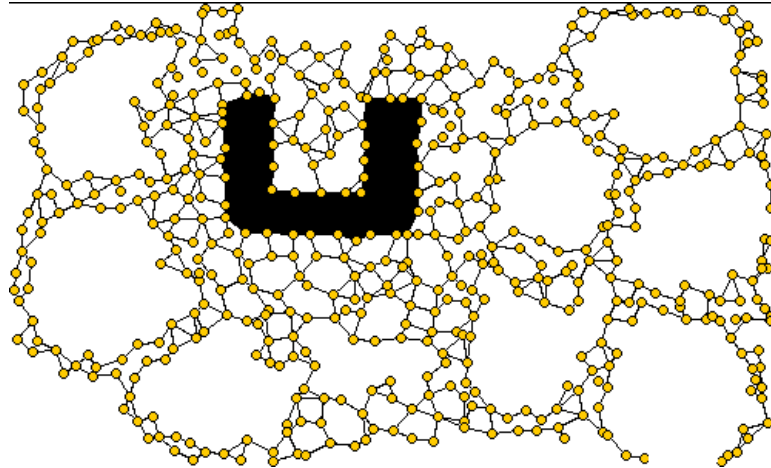


Figure 3.10: Boundary recognition of U shape

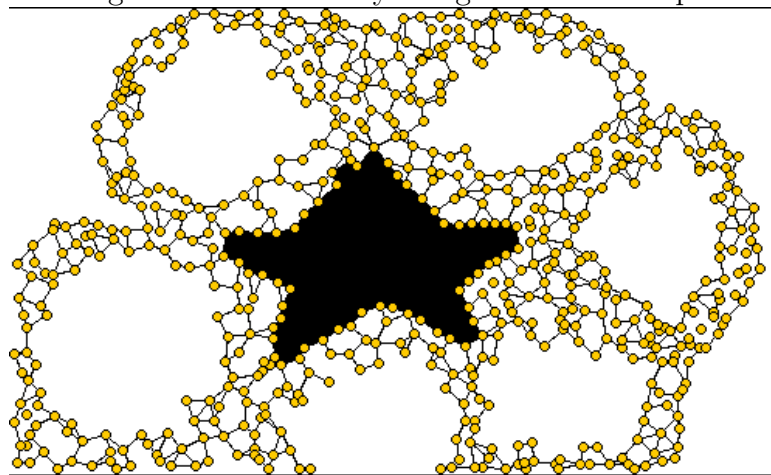


Figure 3.11: Boundary recognition of star shape

3.6.2 Coverage Rate and Coverage Errors

In order to evaluate the coverage rate, the monitoring area is divided into $1m * 1m$ cells. A U-shape obstacle is deployed in the middle of monitoring area. The size of obstacle is 10% of the entire area. We compare the coverage rates determined by binary disk model and our irregular sensing model with or without obstacles. In the idea area (a flat area without obstacle), the coverage rate determined by BSM is much higher than ISM; however, when the number of samples reaches 500 the coverage rates of two protocols

becomes the same. When obstacles are applied to the area, because the binary disk model is not aware of such changes, it keeps the same coverage rate. The coverage rate determined by ISM is quickly changed and approaches the real coverage rate when the number of samples reaches a certain level.

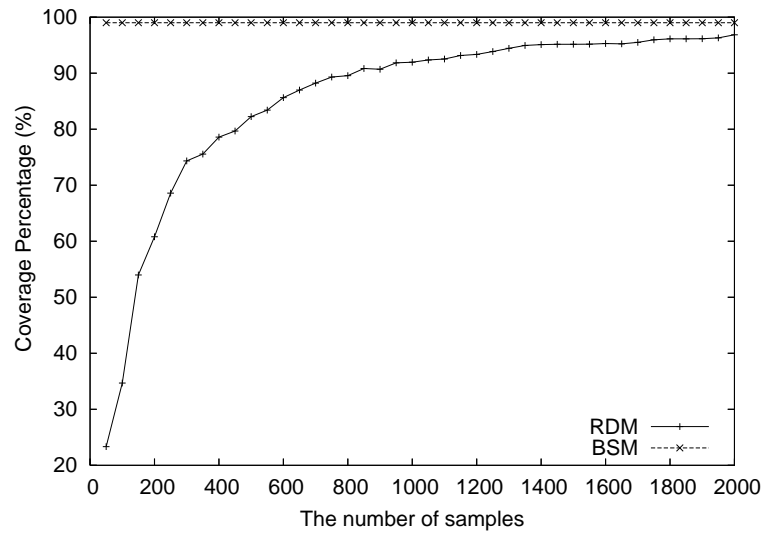


Figure 3.12: Coverage rate without obstacles

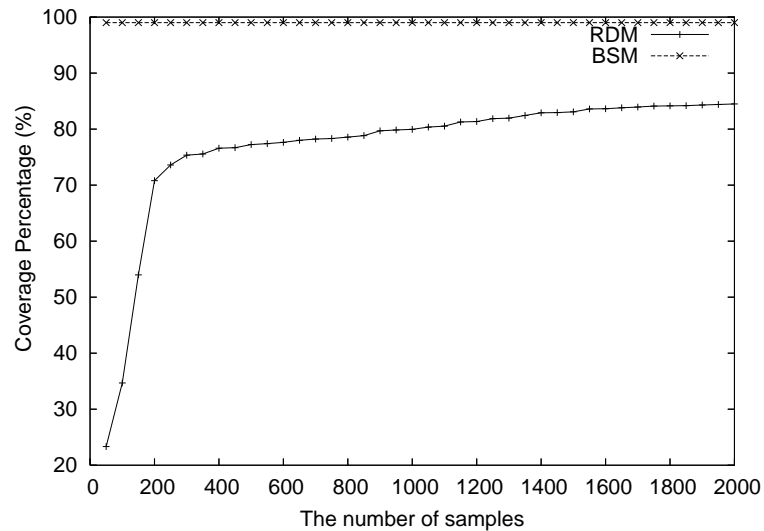


Figure 3.13: Coverage rate with obstacles

Comparing to the BSM, ISM is aware of changes of environment; however, when

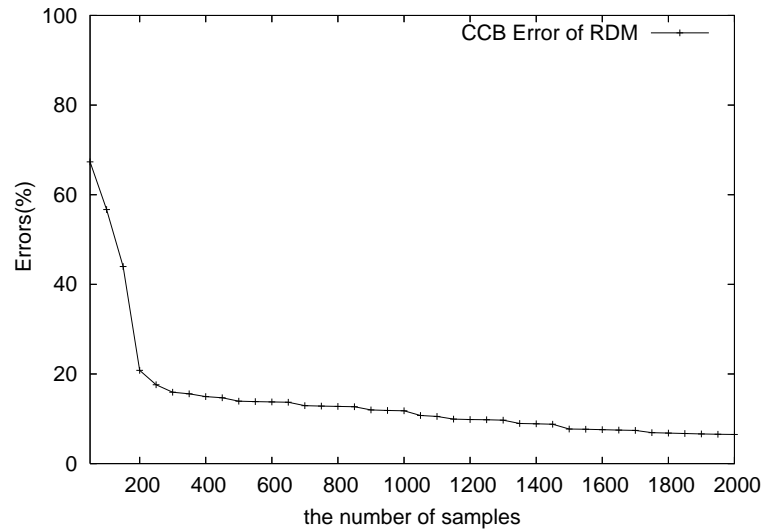


Figure 3.14: CCB errors

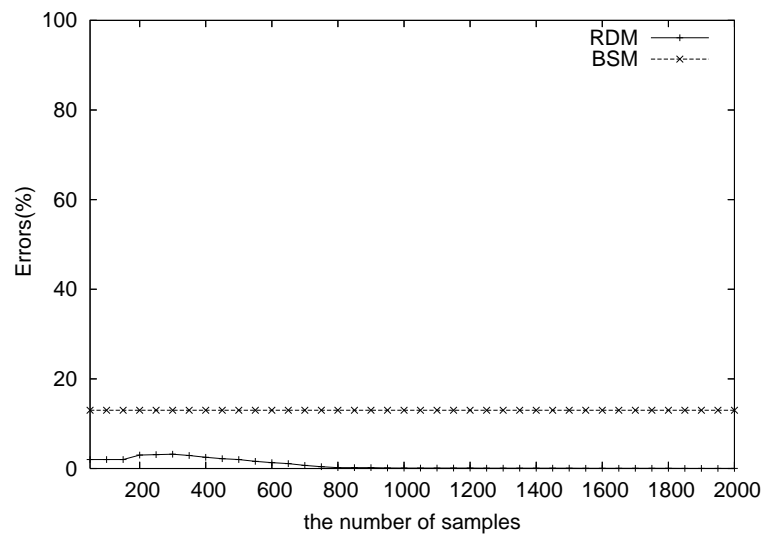


Figure 3.15: BCC errors

the sample density is not high enough the coverage determined by ISM contains two kinds of errors, covered area considered as breach area(CCB) and breach area considered as covered area(BCC). In the coverage evaluation based protocols such as duty-cycle scheme, such errors will lead to wrong elections result for on-duty sensors. As shown in Figure 3.14 and 3.15, when only 50 samples are deployed in the network the CCB error

of ISM is over 60%; however, when the number of samples reach 1000, only 14% error exists. Comparing to the result of ISM, when obstacles are applied to the environment, the BCC errors of ISM are reduced to zero when the number of samples reaches 800.

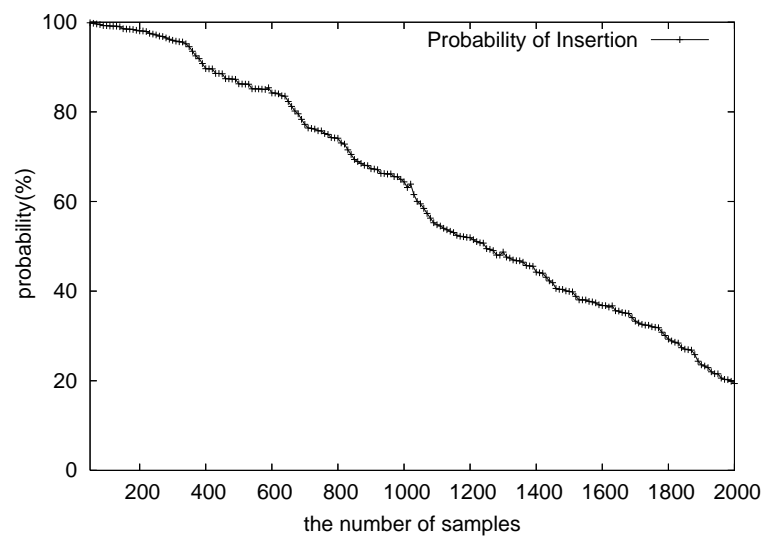


Figure 3.16: Insertion at boundary ring

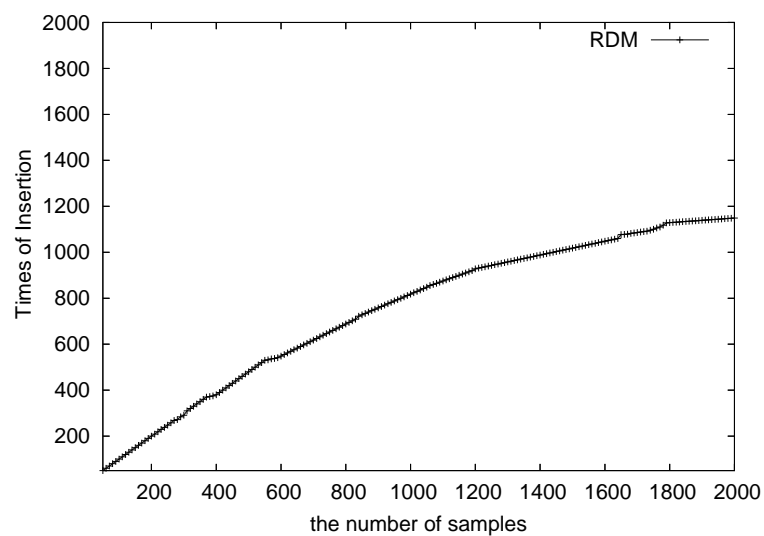


Figure 3.17: Times of insertion

3.6.3 Times of Insertion

The complexity of boundary construction algorithm is mostly contributed by the construction algorithm for Delaunay Triangulation. In ISM, we adopt the incremental construction algorithm to construct Delaunay triangulations. Therefore, we can consider insertion as a fundamental operation for evaluating the performance of our algorithm.

In the section of boundary ring, from the view of probability we show that the input size of algorithms can be reduced by only reconstructing the Delaunay triangulations in the boundary ring. Here, we vary the number of samples in the area from 50 to 2000 to prove it. Because the initial width of boundary ring is equal to radius of sensing disk we record the changes of probability after 50 samples are deployed. As shown in Figure 3.16, the probability of insertion operations is reduced when the number of deployed samples become high. The reason of such reduce is the shrink of boundary ring. In Figure 3.17, the times of insertion are recorded to evaluate the performance of algorithms in practice. When only 50 samples are deployed to the area they are all included in the boundary ring. Therefore, the number of insertion equals to the number of samples. After over 200 samples are deployed, some samples are excluded from the boundary ring. When the number of samples reaches 2000, only 1151 insertion happens while only 233 insertions happens between 1000 to 2000.

3.7 Conclusion

In this chapter, we present an irregular sensing range model for coverage algorithms of wireless sensor network in an area with obstacles. In order to adapt to the changes of environment, based on the gathered data the sensing range of sensors are determined by a revised geometry structure (α -shape). For reducing the complexity, the algorithm of computing α -shape is revised by constructing a boundary ring. The simulation results show that ISM can adapt to the range change caused by obstacles.

Chapter 4

Localized Scheduling Protocol for WSN with Irregular Sensing Range

4.1 Introduction

The most common sensing model which is used by most protocols, assumes that a sensor can cover a disk centered at itself with a radius equal to the sensing range. However, in most cases, sensing system can not really have a disk sensing range. Sensing range is location-dependent and usually irregular[1] (for example, the photo sensor uses a square sensing range). Moreover, if we consider the effect of reflection caused by boundaries and obstacles, the sensor can not maintain its disk sensing range unless it is working on a completely flat and boundary-free area. Problems with such irregular sensing range have been discussed by Huang and colleagues[1], who pointed out that their K-covered scheme could work under an irregular polygon sensing range assumption, but did not literally prove this. When this irregular polygon sensing range problem was applied to region coverage, it became more complicated. The issues mentioned above lead to a challenge that has motivated our research: “Can a distributed scheme, which solves the coverage problem under a polygon sensing range without relying on the GPS system, be devised?” Another interesting aspect of coverage is tolerance to holes (blind areas) since

high accurate coverage is not always necessary (the accuracy for tracing a person is not the same as that for tracing a tank). If we are able to control the size of holes that can be tolerated, coverage will not be jeopardized while the network lifetime can be remarkably extended by putting more sensors in the off-duty state[9].

In response to the challenge posed above, in this chapter, we propose an intersection point method (IPM) to help nodes discover the coverage holes without the limitation imposed by the disk sensing range. Moreover, a unit circle test (UCT) with adjustable test radius r_u , is developed to make our scheme detect holes in a highly precise, adjustable manner.

4.2 The Intersection Point Method(IPM)

In order to make the coverage scheme more realistic, we propose a new algorithm that is based on the irregular polygon sensing range in order to discover fully covered sensors. The solution, named Intersection Point Method(IPM), is based on the investigation of the intersection points of sensing polygons. Our scheme is devised under the following assumptions:

- (1) The sensors' density is sufficiently high that only part of the sensors can monitor the desired region R_m .
- (2) The sensing range is a closed simple polygon and can be detected by sensors.
- (3) The communication range is twice the maximum sensing range.

4.2.1 Basic Definition

The basic definitions necessary to understand the IPM scheme are described below and serve as a basis for our proposed solution.

Definition 21 (Transmission Neighboring Set): Consider a set of sensors $\{p_1 \dots p_n\}$ in a finite area δ . If we assume that r is the radius of a sensor, then the neighboring sensor set TNS_{p_i} of sensor p_i is defined as: $TNS_{p_i} = \{n \in \mathbb{N} | \text{distance}(p_i, p_j) < r, p_i \neq p_j\}$

Definition 22 (Sensing Neighboring Set): Consider a set of sensors $\{p_1 \dots p_n\}$ in a finite area δ . If we assume that SR is the sensing range of a sensor, then the neighboring sensor set SNS_{p_i} of sensor p_i is defined as: $SNS_{p_i} = \{n \in \aleph \mid SR_{p_i} \cap SR_{p_j} \neq \phi, p_i \neq p_j\}$

Definition 23 (Candidate-Fully Sponsored Sensor): We refer to a node A as a Candidate or as being fully sponsored by its neighbors if the sensing area, $S(A)$, is fully covered by $S(SNS_A)$, where SNS_A represents the sensing neighboring set of sensor A , denoted as $SNS_A \xrightarrow{FS} A$.

Definition 24 (Simple Polygon): A polygon P is said to be simple (or Jordan) if the only points of the plane belonging to two polygon edges of P are the polygon vertices of P .

We can simply consider a simple polygon to be a polygon without self-intersections.

Definition 25 (Intersection Point of Polygons): A point p is said to be an intersection point of two polygons if the two edges, which generate such a point, belong to different polygons. If there is no vertex of any other polygon located in exactly the same location, such a point p is called a line intersection point (LIP) of polygons. Otherwise, it is called the vertex intersection point (VIP) of Polygons.

Definition 26 (Intersection sub-polygon): A sub-polygon of polygon P is considered to be an intersection sub-polygon if its vertices belong to the intersection points set of P and P 's neighbors. If such a sub-polygon exists inside P , we consider it to be a breach intersection polygon.

Theorem 3 A polygon P is fully covered by its sponsors only if there is no intersection point $p' \in P$, which is generated by two sponsor polygons P_1, P_2 , and if it is covered only by polygon P .

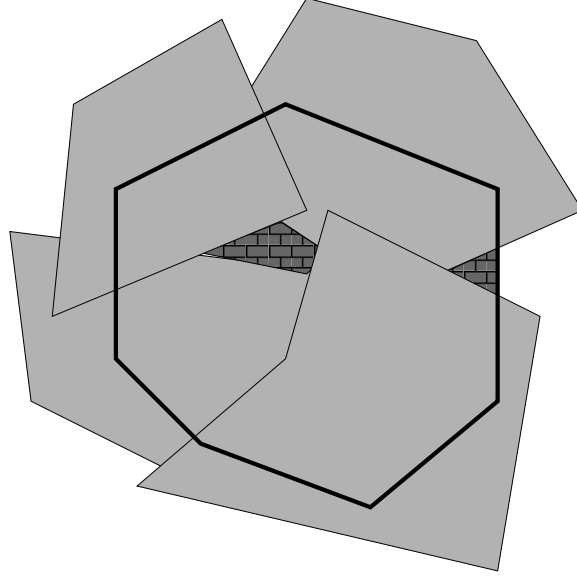


Figure 4.1: The breach intersection polygon

Proof of Lemma 1:

Let us assume that such an LIP' of P_1 and P_2 exists inside a fully sponsored polygon P . Considering another point p'' in P , which is excluded in P_1 and P_2 ; the distance $d_{p' \rightarrow p''}$ approaches 0. Because P is a fully sponsored Polygon and $p'' \notin \{P_1 \cup P_2\}$, p'' must be covered by another sponsor polygon P_3 . When $d_{p' \rightarrow p''}$ approaches 0, P_3 will intersect with P_1 and P_2 at p' or cover p' ; otherwise, any points between p' and p'' will be covered only by P , which will cause a conflict with our assumption that P is a fully sponsored polygon. Lemma 1 has been proven.

4.2.2 The Intersection Points Method

In the simple polygon world, the relation of polygons is fourfold: inside, overlapping, tilling and intersecting. Because we assume that no two sensors are in the same location, we will ignore the overlapping case and focus only on the three other cases. Based on the geometry graph theory, we can determine that polygon A is *inside polygon B* by checking the vertices of polygon A against polygon B to see if they are all inside polygon B. If no

vertex of A is outside the range of polygon B and no vertex of B is inside A, we can say that simple polygon A is inside simple polygon B.

The problem of identifying a sub-polygon is not fully sponsored is similar to the problem of discovering an intersection sub-polygon that is inside polygon B and only covered by B. The basic solution for discovering such a sub-polygon is to identify all of intersection sub-polygons and mapping their vertices against the sponsored polygon. Delaunay triangulation is the most common method for dealing with such a problem. It tills the polygon into the smallest sub-polygons with the vertices and intersection points. The complexity of computing all of the intersection points in order to discover all of the vertices is $O(C_{n+1}^2 * m^2)$, where m is the average number of edges of the polygons and n is the number of sponsors. The complexity of construction of Delaunay triangulation is $O(X \log X)$ when X is the number of vertices and intersection points. After a Delaunay triangulation is created, we can match each triangle with polygons to find out whether there is a triangle covered by less than one polygon. The complexity of such a process lies in $O(3t * m)$ where t is the number of triangles. The total complexity of such a process will be $O(C_{n+1}^2 * m^2 + X \log X + 3t * m)$. Compared to the processing capability of a sensor node, such computations can be too complex to identify all of the breaching holes under the assumption of polygon sensor range. Thus, we must reduce the complexity of such a solution to make it more realistic. Further research is encouraged by Lemma4.

Theorem 4 *If a polygon P intersects with its sponsors and is not fully sponsored, there must be an intersection polygon inside P whose vertices are composed by the intersection points of P and P's sponsors.*

In other words, if we can identify an intersection point in the polygon P adjoined to any area that belongs only to P, we can say that P is not fully sponsored. Actually, for the off-duty scheme we are more interested in the existence of such a breach polygon than in obtaining all of its vertex information. Thus, we develop the following algorithm to identify an un-fully sponsored sensor:

Algorithm 6 The Intersection Point Method

for (each node q) **do** Use a line sweep algorithm to investigate all of the intersection points generated by q and q 's neighbors; Remove intersection points uncovered by P ;

Find an intersection Point (IP) that adjoins a breach intersection polygon;

If there is no such IP, the tested sensor is fully sponsored.

end for

The key issue of this algorithm is to discover the intersection points adjoining the breach intersection polygon. In order to resolve this problem, based on Lemma 4 we devised a new method: the unit circle test.

4.2.3 The Unit Circle Test

Assuming that sensing polygons W , X , Y and Z intersect at an intersection point $IP_t \in P_S$, we draw a small enough test circle centered at IP_t to identify whether P_S has a hole closed to IP_t . The intersection edges of IP_t will cut such a test circle into small pieces as shown in Figure 4.2. If IP_t adjoin to a breach area, at least one piece generated by test circle will belong to a breach sub-polygon. Thus, we can pick up points from each piece, such as points A to G in Figure 4.2, and test whether they are inside polygons W , X , Y , and Z . If any point from A to G belongs only to polygon P_S , polygon P_S is not fully sponsored. In order to select proper points, we pick up the test points as follows:

As depicted in Figure 4.2, the test circle intersects with all intersection edges of IP_t . We denote such new intersection points generated by the Test Circle as p_{ui} . We connect p_{ui} clockwise and pick up the middle point of such a connection line segment as the test point (TP). As we know, we can look at each of the pieces generated by the unit circle test as a convex curve polygon, which means that test point will be located exactly in each of the pieces. Thus, if a TP belongs to polygon A, polygon A is an un-fully sponsored

polygon.

Algorithm 7 The Unit Circle Test

for (each node q) **do**

 Identify all of the intersection points generated by q and q 's neighbors;

 Remove the intersection points uncovered by q ;

 Run the Unit Circle Test;

 Get the Test points TPs;

 Check TPs against Intersection polygons;

 If no such TP, which exists only inside of q , q is considered to be a fully sponsored node;

end for

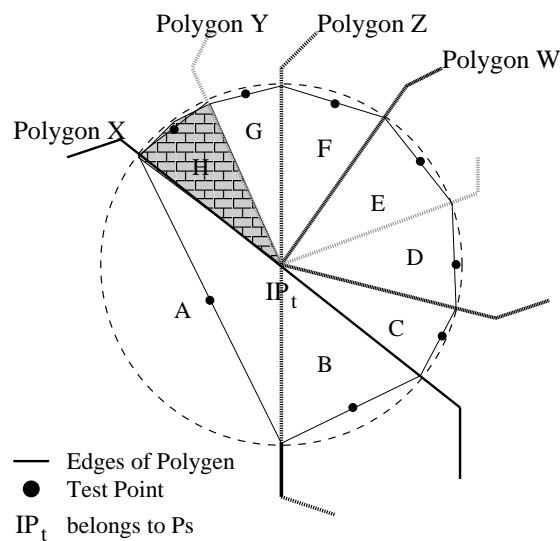


Figure 4.2: Unit circle test for polygon P_S

Tolerance to Holes:

In the application where sensors are deployed in high dense manner, holes can be more or less tolerable. Thus, it would be interesting if we could control the degree of hole tolerance according to the application requirements. Most existing coverage algorithms ignore this requirement or do not support hole tolerance. In IPM, hole tolerance control

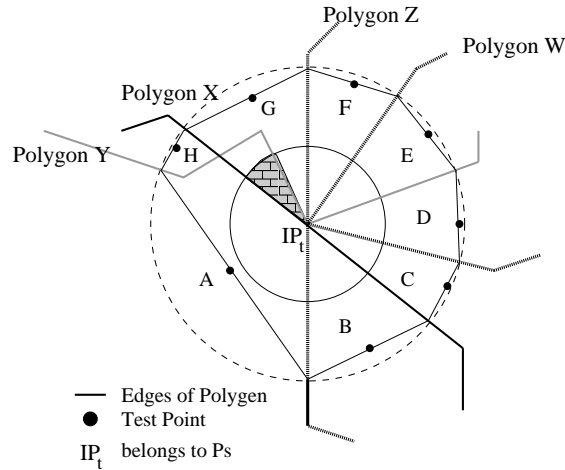


Figure 4.3: Hole-tolerance

can be supported by adjusting the radius r_u of the Test Circle; we can allow algorithms tolerant to holes no larger than the Test Circle. Figure 4.3 shows that when the Test Circle was enlarged from the solid line circle to the dash line circle, the breach polygon created by polygon X and Y and denoted by the black brick area is ignored.

4.2.4 Non-Similar Degree

Even though IPM is more realistic and accurate than CAM, an increase in complexity should not be ignored. A problem is presented in terms of maintaining the quality of coverage while decreasing the the average complexity of algorithm. By comparing the IPM and CAM algorithms we are able to discover that the increase of computing complexity is caused for most part by computing the intersection points of the edges of sensing polygons. It is obvious that if we try to maintain accuracy, we cannot decrease the complexity. Can we employ the same idea as hole-tolerant described in the Unit Circle Test to decrease the complexity by just sacrificing a small amount of accuracy?

By investigating our experiments as show in the next section, we find that the more a polygon sensing range similar to its Circumcircle the fewer errors are found. Therefore, if the polygon sensing range is replaced by its Circumcircle while the error caused by

such replacement can be tolerated, we can markedly reduce complexity and maintain the quality of sensing coverage. To find out which polygon range is suitable for such a replacement, we define a Non-Similar Degree.

Definition 27 (The Non-similar Degree): A Non-similar degree (NsD) of a polygon p is defined as the difference between the radius of p 's Circumcircle and that of Incircle.

$$NsD = r_{inc} - r_{cir} .$$

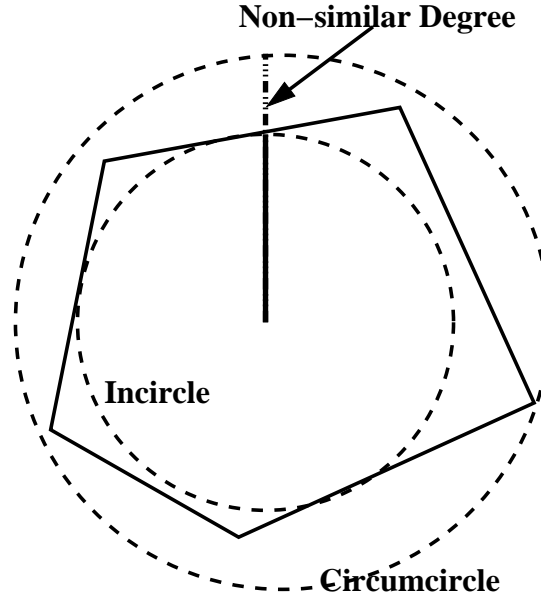


Figure 4.4: The non-similar degree

When this Non-similar Degree is smaller than the diameter of the test circle, $NsD \leq 2 * r_u$, the sensing polygon is replaced by its circumcircle. By applying the NsD to IPM, the process of calculating the intersection points is divided into three sections: polygon Vs. polygon, polygon Vs. Circumcircle, and Circumcircle Vs. Circumcircle. If we denote P_r as the probability of sensors suitable for NsD, the complexity of $IPM + NsD$ is

$$(X * k * m + (1 - P_r)n * m * \log((1 - P_r)n * m) + ((1 - P_r)n + 1) * m * \log((1 - P_r)n + 1) * m) + 2P_r n) =$$

$$X * k * m + (1 - P_r)n * m(\log((1 - P_r)n * m) + \log(((1 - P_r) * n + 1) * m)) + m * \log((1 - P_r)n + 1) * m) + 2P_r n).$$

4.2.5 Comparison of IPM with Other Methods

In order to illustrate the IPM's strength, we compare IPM with Probing [24]; the central angle method (CAM) [23]; the association sponsor method (ASM) [16], and Delaunay triangulation in terms of sensing range assumption, precision, hole-tolerance and computing complexity.

Probing is a basic method for the coverage problem. It does not consider the sensing range and turns on sensors only when there are no other sensors in the communication range. Probing runs very quickly and is easy to implement. However, it does not deal with holes at all. The central angle method finds the existence of holes by identifying fully sponsored sensors locally but it cannot completely determine holes. In order to maintain the network connectivity and take full advantage of the central angles method, ASM considers the overlap area of all neighbors by associating the neighbors with low overlap area and high overlap area.

In the comparison of Table 4.2.5, IPM shows good potential in most of the aspects except high computing complexity which is caused by the simple polygon assumption.

Table 4.1: Comparison of Fully Sponsored Discovery Methods

	Delaunay triangulation	IPM	ASM	C-PNSS	Probing
Sensing Range	Simple Polygon	Simple Polygon	Disk	Disk	Any
Methods	DT	UCT	AS	CA	Probing
Precision	high	radius of Unit Circle	X	X	X
Holes-Tolerance	N/A	Controlable	N/A	N/A	N/A
Complexity	$O(C_{n+1}^2 * m^2 + X \log X + 3t * m)$	$O(Xk^2m + nm \log(nm))$	$O(2n)$	$O(2n)$	$O(n)$

(Where n is the average number of neighbors, m is the average number of edges of a sensing polygon, and k is the average number of edges involved in an intersection point, and X is the total number of intersection points in the polygon P.)

DT: Delaunay Triangulation; UCT:Unit circle test; AS:Association sponsor; CA:Central angles.)

4.3 Performance Evaluation

In order to evaluate the performance of our scheme a set of simulation experiments have been carried out. The simulation result is compared to both of the association sponsor method and central angles method.

4.3.1 Experimental Environment

The sensors and sink are randomly deployed in the $50m \times 50m$ area. Each sensor's sensing range is considered to be a simple polygon and varies from 5 to 10 meters. The number of sensors and sink varies from 100 to 300.

In order to calculate coverage, the monitored area is divided into $1m \times 1m$ grids where events are generated at the cross points every 0.5 second. By determining the number of events detected by on-duty nodes, we can roughly calculate the coverage rate. If an event source cannot be detected by any on-duty node, we call the location of such an event source a *hole* or *blind point*. We assume the initial network coverage is 100% and consider only the holes generated by turned-off sensors. The numbers of blind points are compared between initial stage and final stage. How well the scheme prevents the occurrence of blind points after turning off sensors indicates the coverage-preserving ability of the scheme.

The simulation result shows that under the desire accuracy requirement IPM can efficiently identify all the candidates and discover all the breach points without error. Even our algorithm can be applied to any simple polygon sensing range the simulation results still could be affected by the shape of sensing range. In order to simplify the calculation and keep the validity of comparison, we limit the shape of sensing range to equilateral triangle sensing range(convex simply polygon) and star(concave decagon) sensing range. Because CAM and ASM are developed under disk sensing range, circumscribed disk sensing ranges are used to keep the fairness of comparison.

4.3.2 Experimental Results

Experimental results are compared and analyzed in terms of the following two aspects: The number of candidates & errors, coverage ratio & tolerated holes.

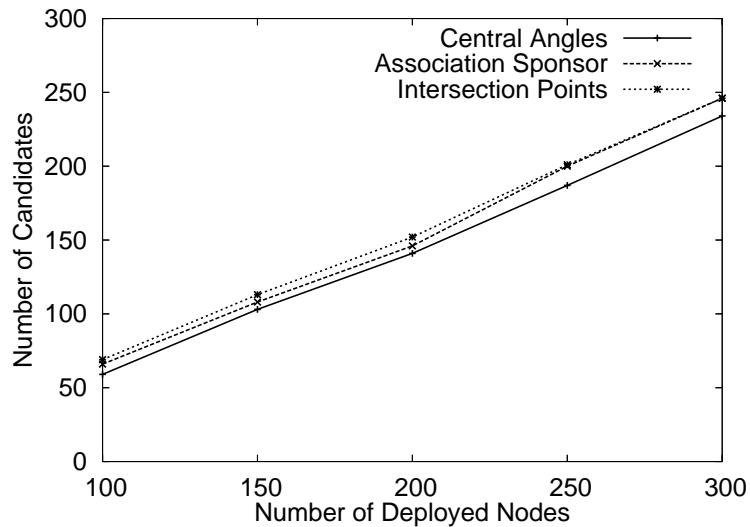


Figure 4.5: The number of candidates Vs. the number of deployed nodes, disk sensing range, $r=10m$

- **Candidate & Errors:**

In this section, we evaluate how well the IPM identifies the fully sponsored sensor. We run the algorithms separately under the equilateral triangle sensing range, star sensing range, and disk sensing range separately and evaluate the identified candidate by checking whether all the even sources in the sensing range of identified candidate are covered by the sensing range of other sensors. We will only consider the miss-identified candidates which will cause holes, and ignore the error of missing eligible candidates. As shown in Figure 4.5, the performance of all algorithms is good under the disk sensing range and IPM achieves the same result as ASM on average and in some cases it is even better than ASM. It is interesting to observe that in Figures 4.6 and 4.8 when the algorithms are applied to irregular sensing polygon such as triangles and stars, ASM and CAM identify an abnormally high number of candidates than IPM. We investigate the number of miss-

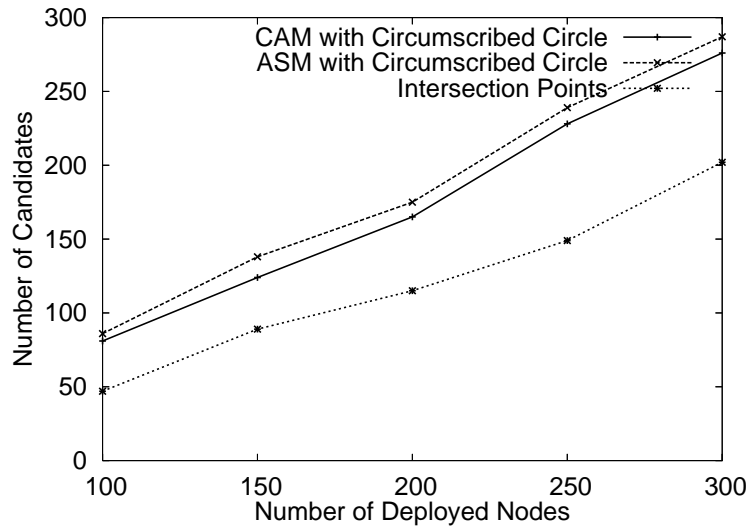


Figure 4.6: The number of candidates Vs. the number of deployed nodes, equilateral triangle sensing range, $r_u = 0.1m$

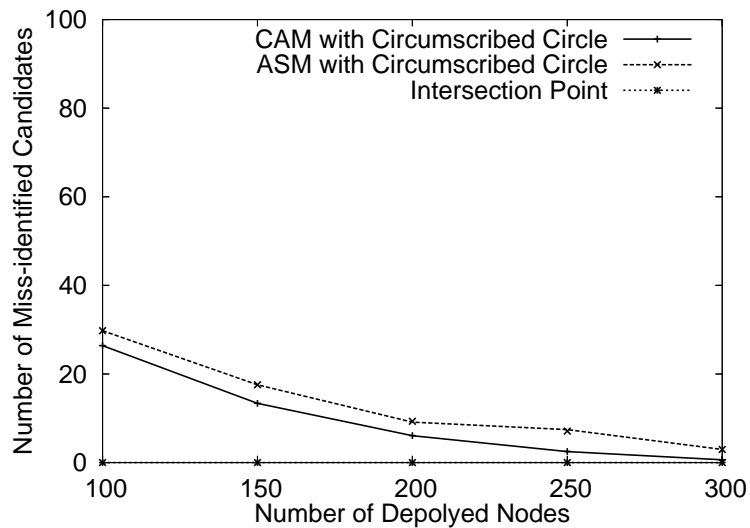


Figure 4.7: The number of errors of candidates Vs. the number of deployed nodes, equilateral triangle sensing Range, $r_u = 0.1m$

identified candidates by comparing the results of IPM, ASM, and CAM. In Figures 4.7 and 4.9 the latter two algorithms show over 20 errors when the number of deployed sensors is equal to 100. It is obvious that the algorithms based on central angle method

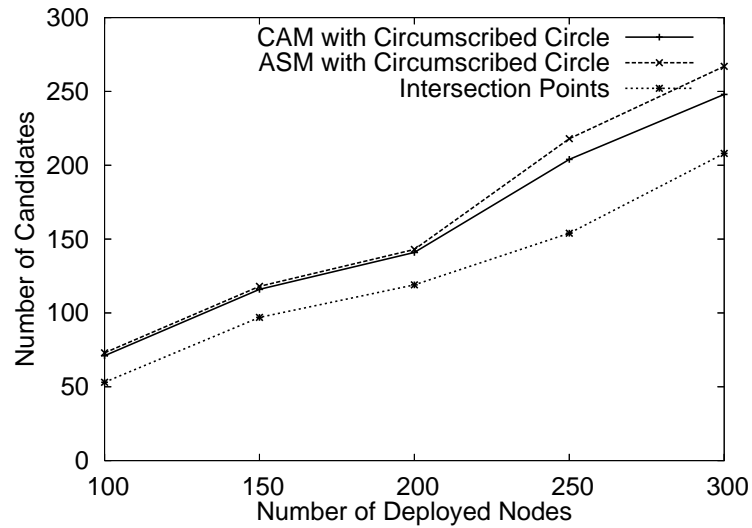


Figure 4.8: The number of candidates Vs. the number of deployed nodes, star sensing range, $r_u = 0.1m$

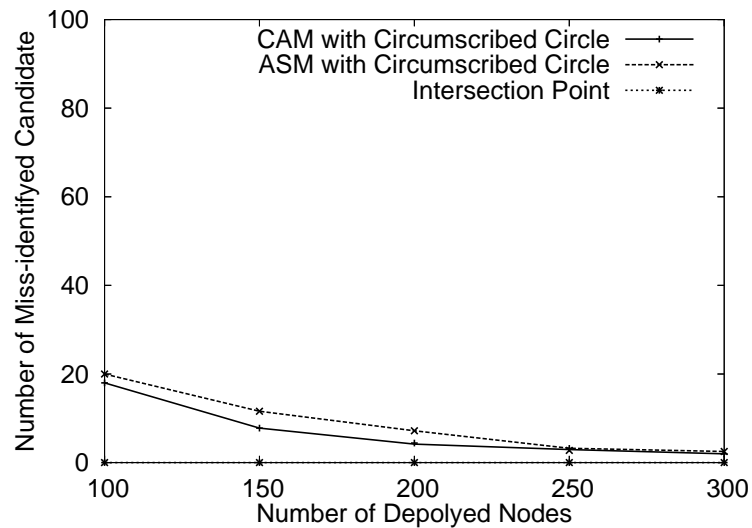


Figure 4.9: Errors Vs. number of Deployed Nodes, star sensing range, $r_u = 0.1m$

does not work well under the irregular polygon sensing range to identify fully sponsored sensors since they are not able to deal with a scenario where two polygons intersect at more than two points.

In Figures 4.7 and 4.9 we can observe that the number of miss-identified sensors are

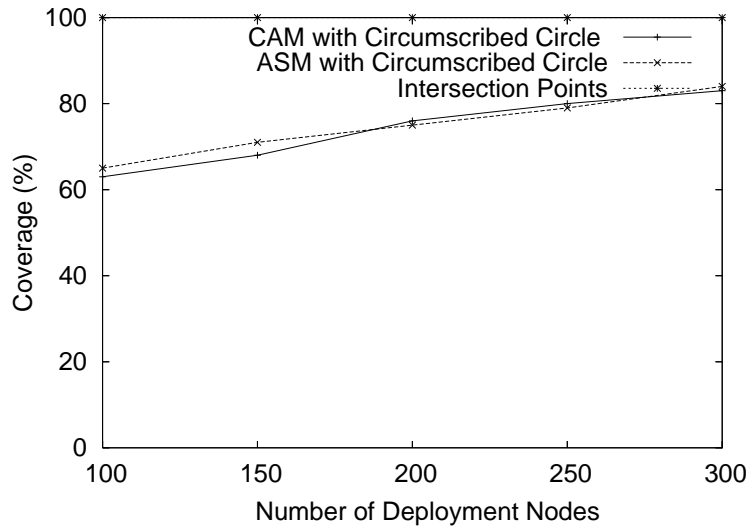


Figure 4.10: Coverage ratio Vs. deployed nodes, $n = 100$, star sensing range

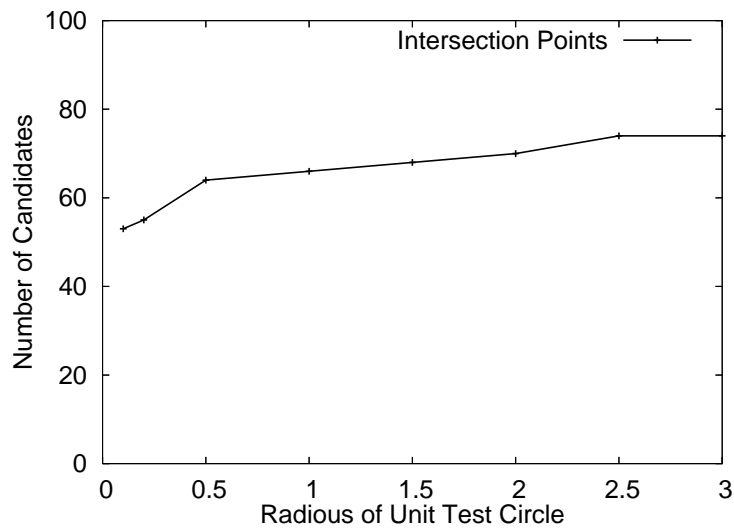


Figure 4.11: The number of candidates Vs. radius of test circle, $n=100$

decreased with the increasing of deployment density. the reason of such decreasing is that the whole monitor area is covered by sensors. In another word, every sensors in the middle of sensing area will be coverage by some other sensors. But when an off-duty scheme is applied the errors will show up again.

• Coverage Ratio & Hole Tolerance

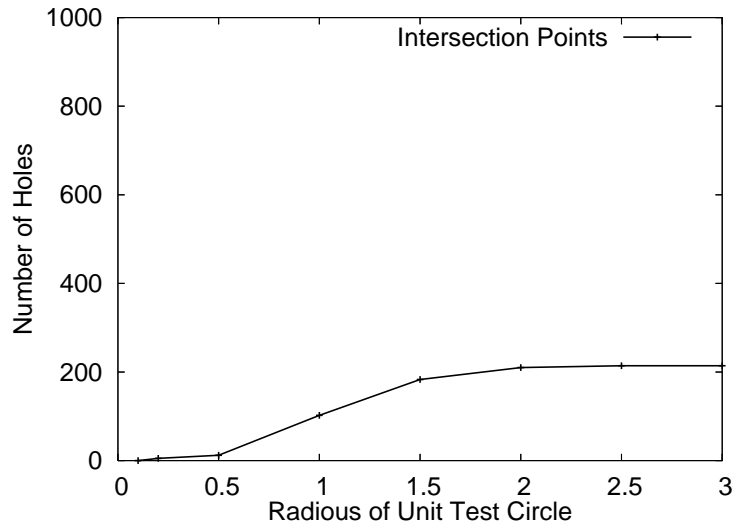


Figure 4.12: Holes Vs. radius of test circle, n=100

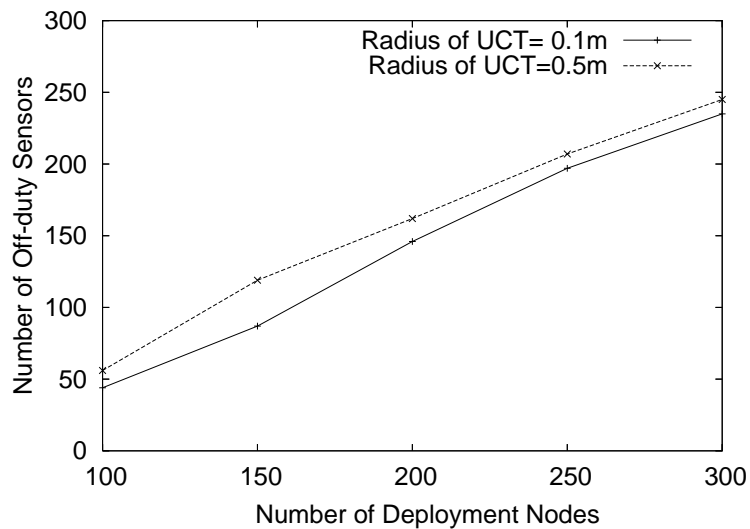


Figure 4.13: Off-duty sensors Vs. deployed nodes, star sensing range

The purpose of identifying the candidates is to turn-off the eligible candidate to save the energy. We apply the the same simple election algorithm on CAM, ASM, and IPM to evaluate coverage quality after turn-off sensors. The result is shown in Figure 4.10. It is interesting that the coverage of both CAM and ASM can not reach 100% after turn-off sensors. Even we increase the deployment density from 100 to 300, the coverage still

low than 85% after applying off-duty scheme. So we can see CAM and ASM can not guarantee the initial coverage under polygon sensing range even under high deployment density.

As shown in Figure 4.6 and 4.8, A problem of IPM is that IsP identifies less candidates than CAM and ACM. It may lead to the result that less eligible sensors are turned off. In a long run, the lifetime of network employed IPM is less than the lifetime of network employed ACM and CAM. The reason is that IPM turn on some sensors to cover some tiny holes. But in some cases, small holes can be tolerated. By applying the unit circle test we increase the number of candidate and tolerate the holes lower than the requirement.

In the Figure 4.11 and Figure 4.12, we adjust the radius r_u of the test circle from 0.1m to 3m and investigate the increase of the number of candidates. The result shows that when the radius reaches 0.5m, the number of identified candidates increases from 52 to 64 and the number of holes that generated by turning off sensors is only 5. All new holes having a radius lower than 0.5m can be tolerated. When the r_u is over 0.5m the increase of the number of candidates is slow down and the number of generated holes increases fast until $r_u > 1.5m$. So by adjusting r_u to 0.5m IPM can identify more candidates and generate only a very low number of holes which can be tolerated in most cases. In the Figure 4.13 we adjust the r_u to 0.5m on all the scenario. We can see the number of off-duty sensors increases about 10%, and the quality of coverage is not hurt by tolerating holes whose radius lower than 0.25m.

4.4 Conclusion

We presented a new fully sponsored sensor discovery scheme, the Intersection Point Method (IPM), which works under the irregular sensing range and can efficiently increase the accuracy of the discovery method through a Unit Circle Test. By adjusting the radius r_u of this Unit Circle Test, the scheme can be made tolerant to holes of a certain size,

making the solution flexible when the degree of accuracy must be controlled. IPM is compared to the ASM and CAM algorithms under the triangle and star sensing range. The error rates obtained show that the central angles method does not meet coverage quality requirements under the irregular sensing range even when a circumscribed circle is assumed. By means of an adjustable test circle, we show that by tolerating holes in a controllable manner, we can efficiently increase the number of off-duty sensors which will extend network lifetime under a high coverage rate. Moreover, a controllable tolerance to holes makes IPM a flexible solution when accuracy can be relaxed. Our solution is compared with CAM and ASM schemes in terms of correctness in identifying candidates. The results show the superior potential of our solution in terms of maintaining a high coverage rate in WSN under an irregular polygon sensing range.

Chapter 5

GA Based Coverage-preserving Scheduling Protocol

5.1 Introduction

In this section, we proposed our centralized scheduling scheme which resolves the scheduling problem using genetic algorithm (GA). In our scheme, the overall network coverage is maximized for a given network lifetime. Through running sensors on the optimized schedule, we can save the energy of network while maintaining the coverage quality. Different from other genetic algorithm based coverage-preserving scheduling protocols, our works use the set-covering method as the fitness function instead of using grid-covering to estimate the coverage of network. The new fitness function reduces the the running time of GA and improves the estimation quality.

5.2 Definitions and Problem Statement

In this section, we first give some fundamental definitions of coverage-aware scheduling and then formally define the problem.

5.2.1 Definitions of Coverage

Definition 28 *The coverage degree of a point p in the wireless sensor network is the number of sensors covering p .*

Definition 29 *The coverage rate of a wireless sensor network is the percentage of area whose coverage degree is greater than or equal to K .*

Definition 30 *The deployment density of a network is a ratio of the summary size of sensing disks to the size of area, $DD = \frac{\sum \pi r^2}{M}$.*

Definition 31 *A wireless sensor network is K -covered when every point in the network is monitored by at least K sensors.*

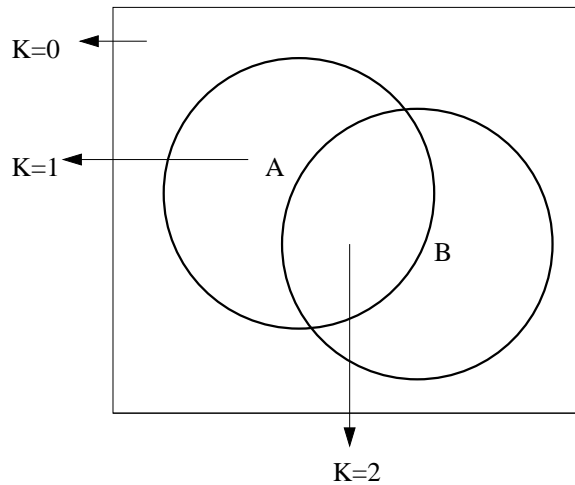


Figure 5.1: K -coverage

Definition 32 *A scheduling process in the wireless sensor network is a function to determine a strategy that assigns jobs to the sensors in order to optimize the performance of networks (such as energy consumption).*

Definition 33 *The overall coverage-aware scheduling in wireless sensor network is the process to schedule sensors in different time slots while maximizing the overall coverage of network for a given time span T and coverage degree K .*

By scheduling sensors in different time slots, we can increase the overall coverage of network. Considering the two sensors in the Figure 5.1, they divide the monitoring area into three sets, $\{K=0\}\{K=1\}\{K=2\}$, by the coverage degree. If we want to maximize the coverage rate and satisfy the 2 – cover requirement, we have to schedule both sensor A and B in the same time slot, however, if only 1 – cover is required, then we can schedule sensor A and B in different time slots. Although the coverage rate in each time slot is reduced, the overall coverage increases due to lower coverage redundancy and network lifetime is extended.

5.2.2 Problem Statement of Coverage Preserving Scheduling

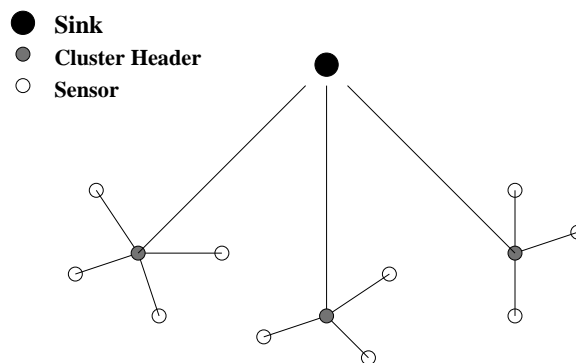


Figure 5.2: Network topology

Consider a densely deployed heterogeneous wireless sensor network based on clusters such as the one shown in Figure 5.2. Sensors periodically monitor the area and report the events to the cluster header. Assuming cluster headers have not energy issues and the sensors only communicate with their cluster header. We will only consider the energy usage and the coverage quality of on-duty sensors. In order to simplify the problem we assume that, in each working interval, every on-duty sensor costs the same amount of energy for sensing and communication. Instead of maximizing the coverage rate in a particular working interval, our optimization goal is to improve the overall network coverage for a given time span. The optimization problem is formulated as following:

Given N sensors deployed in an area M ; sensors only work at their on-duty time slots t_j ; each time slot lasts x minutes. For a given working time span T and coverage degree K , $z \in Z$ is a working schedule for entire network, $z_{ij} = 1$ if sensor i is scheduled to work at time slot t_j . the optimization goal is to maximize the overall k -coverage for a given time span T :

$$\arg \max_{z \in Z} \left(\sum_{j=1}^T \text{Coverage}_j(z, k, x, L, S, M) \right)$$

Subject to:

$$\begin{aligned} \sum_{j=1}^T x z_{ij} &\leq L_i \\ \left(L_i / x \right) &\leq T \\ 0 &< k < |S| \\ z_{ij} &\in \{0, 1\} \end{aligned}$$

where z is a solution, Z is the decision space and $\text{Coverage}(z)$ is the function for computing the network coverage of solution z .

5.3 Related Work of GA for Coverage

The scheduling problem is a NP problem which is hard to find a optimal solution in the linear time. Because GAs are particularly well-suited to solving problems where the solution space is too big to be searched exhaustively in reasonable time, they have been widely used in many problems in the wireless sensor network such as the coverage-preserving scheduling problem.

The earliest instances of genetic algorithms (GAs) were introduced by evolutionary biologists in 1950s to model natural evaluation. Until 1975, based on the earlier research

the book *Adaptation in Natural and Artificial Systems* first systematically described the concepts of genetic algorithm, such as mutation, selection, crossover and so on. GAs simulate an evolution process by taking an initial population of individuals and apply GA operations on each reproduction. Each individual(chromosome) represents a potential solution to the given problem. The fitness of individuals is evaluated by a fitness function. The individuals with high rank have more opportunities to be selected to crossover and reproduce new individuals. The offspring inherit the genes from both parents. In order to introduce new genes into populations, mutation functions are always employed by altering part of genes. The offspring can replace either the whole population or the less fit chromosomes. GAs repeat such processes as shown in Figure 5.3 until the desire solutions are found or reach the generation limitation.

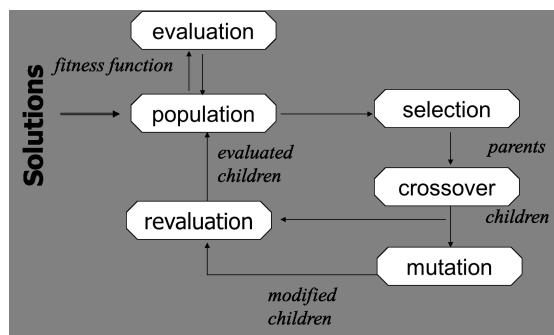


Figure 5.3: Genetic algorithm

In paper [42], Jourdan etc. presented a Multi-Objective Genetic Algorithm to optimize the layout of WSN. The algorithm aims at maximizing the coverage and lifetime of the network, yielding a Pareto Front from which the user can choose. They also investigated the influence of the ratio between sensing range and communication range on the optimal layout with best coverage. When this ratio is below $1/2$, this layout is formed of polygons and resembles a beehive, whereas for ratios above $1/2$, hub-and-spoke layouts become optimal. An analysis showed how the number of spokes in the hub-and-spoke layouts varies with this ratio, from 6 to 2; the optimality of the results provided by the MOGA is thus verified in this simple idealized model. The true value of the MOGA will

become apparent in more realistic conditions, e.g. with mixed sensors, uneven terrain and non-ideal sensing and communication boundary conditions.

In paper [43], authors present a genetic algorithm for solving the constrained coverage problem in continuous space. The genetic operators are novel operators and specially designed to solve the coverage problem. The new algorithm has a high convergence rate and finds the global optimum by a high probability. The algorithm is tested by several benchmark problems, the results of which demonstrate the power of algorithm. The presented new method combines the genetic algorithm with an elegant geometric approach to constrained coverage problem. Presented genetic operators are new which are suggested regarding the coverage problem goal. In addition, the selection operator that is used here is a new type of tournament selection operators. This operator can be used in any genetic algorithm. The proposed algorithm can be used for solving the discrete and weighted coverage problem. With respect to the improvements being made to the weighted Voronoi diagram, naturally one of the future works can be presenting a compatible version of these algorithms in the non-uniform density space.

In paper [44], authors consider the problem of maximizing the network lifetime (in terms of rounds) while maintaining a high quality of service (QoS), which includes target coverage and network connectivity. The paper generalises the searching procedure to maximise the total number of rounds. A novel searching algorithm is proposed on the basis of improved NSGA-II to select the optimal coverage set which includes two competing objectives, the coverage rate and the number of working nodes. As another new contribution, we apply the novel algorithm in the k-disjoint coverage sets problem. All the sensors are divided into k-disjoint sets, guaranteeing each set with full coverage. By alternating coverage subsets and using only one at each round, the maximum network lifetime is achieved. Numerical and simulation results are provided to examine our analysis for wireless sensor networks.

In paper [45], on the opposite to traditional coverage approaches that aim at guaranteeing a uniform density distribution, authors place the sensor nodes in a manner that

increases the coverage degree according to their proximity to a sink node. To reduce the complexity of the optimization process, we consider a discrete search space by structuring the monitored into a uniform grid. An evolutionary algorithm is then used to choose whether to activate or not sensor nodes within every cell of the grid. Authors conducts a set of simulations in order to evaluate the performance of the proposed strategy, mainly in ensuring multi-target tracking.

5.4 Grid-cover Vs. Set-Cover

Since the fitness function is called at every generation to evaluate all the chromosomes the complexity of fitness function is highly affect the performance of GA algorithm. Most current genetic algorithms targeting the coverage-awareness scheduling problem share the same fitness function based on the grid coverage [43, 45, 44]. In this section, we first review the classic grid-cover based fitness function and then present a new set-covering based fitness function to reduce the time complexity of genetic algorithm for coverage-preserving scheduling problem. It has less computing complexity and better accuracy than the grid-cover based fitness function. The performance comparison of two fitness function is shown in the simulation experiments section.

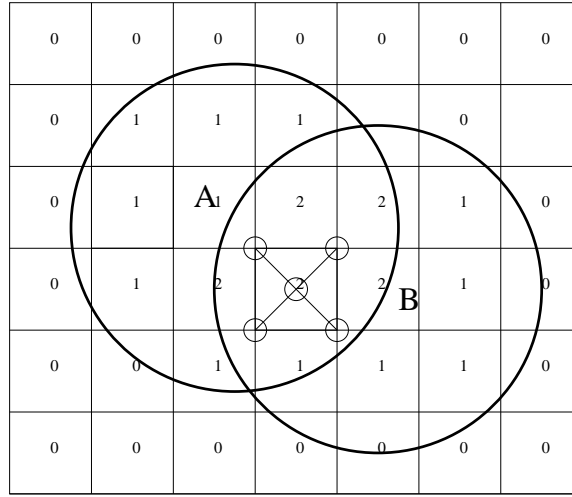
5.4.1 Grid Based Fitness Function

In order to estimate the coverage of network, usually the surveillance area is divided into grids[references]. If the four vertexes of a grid fall are in the same sensing disk of a sensor, the gird is covered by the sensor. If a grid is covered by at least k' sensors then it is rated as K'/K where K is the K -cover requirement. If we denote the number of total grids as G , the lifetime of network is T , then the fitness function can be expressed as following:

$$\text{Fitness} = \frac{\sum_{i=1}^T f_i(k)}{T * G} \quad (5.1)$$

$$k'_{ij} = \left\{ \begin{array}{ll} k'_{ij} & \text{if } k'_{ij} \leq K \\ K & \text{if } k'_{ij} > K \end{array} \right\}$$

$$f_i(k) = \left\{ \begin{array}{ll} \sum_{j=1}^G k'_{ij}/K & \text{if } \sum_{i=1}^N \sum_{j=1}^T 1^{s_{ij}} \leq N * E \\ 0 & \text{if } \sum_{i=1}^N \sum_{j=1}^T 1^{s_{ij}} > N * E \end{array} \right\}$$



○ Test Points

Figure 5.4: Coverage rate example

Take Figure 5.4 as an example. In order to maintain 2-cover, both sensor A and B have to be scheduled in the same time slot. If we assume that the network lasts for 2 minutes, then the fitness result for such a schedule is

$$\text{Fitness} = \frac{5 + 5}{2 * 42} = 5/42$$

Assuming the neighboring sensors of each grids are computed before running GA algorithm. If we consider the function to determine the distance between a vertex and a sensor as a meta function, in the worst case, the complexity of grid based fitness function at each generation is $O(GK_{max})$. Since the size of grid is determined by the accuracy requirement r_{grid} , the complexity is $O(MK_{max}/r_{grid}^2)$. Such simplification could introduce errors into the computation of coverage degree, however, when the size of grid is far

smaller than the size of sensing disk the error could be ignored. However, the complexity will be increased.

5.4.2 Set-Covering Based Fitness Function

The grid based fitness function is easy to be implemented. However, it suffers several drawbacks. The first, the accuracy of coverage estimation relies on the size of grid; The second, for a given estimation accuracy, the complexity of fitness function increases with the size of surveillance area. Because the effect of sensing range and the deployment density are ignored the fitness function for networks with single-sensor and multiple-sensor has the same complexity. The third, grid based coverage estimation will ignore the grids on the edge of sensing disk. It introduces many false-negative coverage degree estimation on the grids as shown in Figure 5.4. In order to overcome the drawbacks of grid based method, we introduce another coverage estimation method based on the set partition.

If we consider the bounded surveillance area, M , and the sensing disks s_1, \dots, s_i , as closed sets containing infinite $2D$ points, then the disk covering problem is converted to a set covering problem. The boundaries of M and s_1, \dots, s_i divided M into many subsets. the union of all the subset is a tessellation of entire monitoring space. Since M and s_1, \dots, s_i are closed sets the subsets M'_j are closed set too and each element in the M'_j has the same coverage degree. Therefore, we can pick up only one element from each M'_j to test their coverage degree and only calculate the size of those M'_j whose coverage degree equals to K . Since all the sets are closed set the determination of sets is equal to find the minimum cycles in the graph. In order to select element from every subset M'_j we are using the intersection point method (IPM) introduced by Azzedine etc. in [46].

Since subset M'_j are closed set it is obvious that each subset M'_j must be composed of the intersection points, intersection curves and boundary edges.

Therefore, we can find test points for every intersection point to discover all breach polygons in P . How we find our test points is shown as Figure 5.5. The intersection

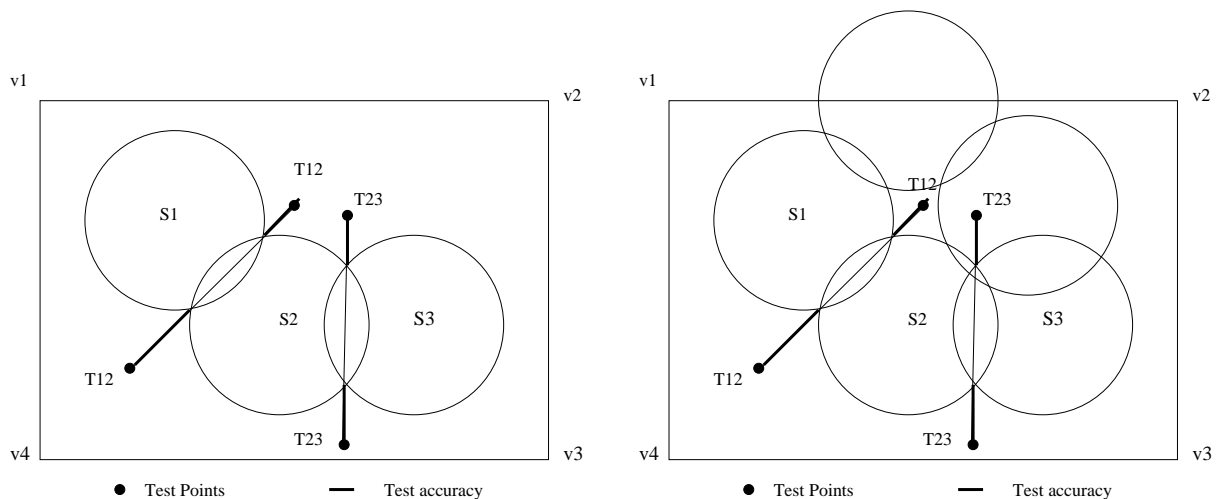


Figure 5.5: Test points

points of two sensing disks are connected by a line segment. Both ends of the segment are extended by a length δ that we referred to the *test accuracy*. The ends of extended line segment become the test points of their corresponding intersection points. For those M'_j without intersection points, the centroid of M'_j will be used as the test points.

5.4.3 Time Complexity of Fitness Functions

assuming the diameter of the smallest M'_j is d . If $\delta < d$, then the test point set T_s contains at least one elements from each subset M'_j . Because any two intersection disks have no more than two intersection points, when we try to cover a rectangle area with disk, it is obvious that, in the worst case, there are $O(m^2)$ intersection points where m is the number of sensors. However, such worst case does not always happen in real scenarios.

Assuming that the distribution of sensors X in the area follows normal distribution, the probability $dist(s, s') \leq r$ is proportional to the size of the area. The probability of two sensors having intersection is

$$p = \frac{4\pi R^2}{M}. \quad (5.2)$$

When n sensors are deployed in an area, the probability of a sensor having x intersection neighbors is p^x . For example, when 100 sensors are deployed in a $160000m^2$ area, when the communication range R is $80m$; the probability p' approaches zero when $x = 9$. If we assuming every sensor has no more than k adjacent neighbors, in the worst case, the number of intersection points is $O(2xn)$ where $x \ll n$. Since for each intersection point there are only four test points, using the same meta function as grid coverage we can determine the coverage of entire area without errors by not more than $O(8xnK_{max})$ tests. Comparing to the complexity of grid coverage, $O(4MK_{max}/r_{grid}^2)$, for the same test accuracy, test point based solution has less complexity than grid based solution.

5.5 Genetic Algorithm for Coverage-aware Scheduling

Based on the fitness function presented in the last section, we detail the setup of our genetic algorithms in this section.

5.5.1 Encoding and Initialization

Our goal is to find the optimal schedules for entire network. We define that a chromosome is a schedule for N sensors; each chromosome contains N genes. The gene of such a chromosome is the schedule of a sensor.

$$\text{chromosome} = g_1g_2g_3g_{N-1}g_N \quad (5.3)$$

$$\text{example : } N = 3, L = 3$$

$$\underbrace{010}_{g_1} \underbrace{111}_{g_2} \underbrace{011}_{g_3}$$

For example, we are looking for the maximum overall coverage for a network which contains N sensors, satisfies k -cover and lasts T minutes. there will be total $L = \lceil T/t_{int} \rceil$ time slots when each working time slot lasts t_{int} minutes. if we use a bit(0/1) to present

the scheduling statue of a sensor in a time slot, then each gene contains L bits and could be present as following:

$$\begin{aligned} \text{gene} &= s_1s_2s_3\dots\dots s_L \\ \text{example} &: 01010111 \end{aligned} \tag{5.4}$$

$$s_i = \left\{ \begin{array}{ll} 0 & \text{scheduled} \\ 1 & \text{unscheduled} \end{array} \right\}$$

The schedule of sensors should not over their energy capability. If the total energy of a sensor is E , then every eligible gene should satisfy the following constrain:

$$\text{constraint} : \sum_{i=1}^L e^{s_i} \leq E \tag{5.5}$$

5.5.2 Selection and Fitness Function

In order to generate a new generation, a proportion of the existing population is selected. Based on the definition of problem a fitness function ranks each solution and preferentially selects the best solutions. It rewards the increase of overall coverage and penalizes the decrease. The higher the rank of an individual, the better chance it gets survived. The selected individuals of the current population crossover with each other to generate new chromosomes which will join the existing population. Tournament selection is used in our work. In tournament selection, X individuals are chosen from population randomly in X rounds with the probability $\{p(1-p)^i | 0 \leq i \leq X-1\}$ where p is a predefined probability. In our simulation we use $X = 100$ and $p = 0.9$.

5.5.3 Reproduction and Termination

The new generation of solutions is created through two genetic operators: crossover (recombination), and/or mutation. Crossover brings new population into original populations. There many crossover techniques exist. In this work the single-point crossover is adapted. In single-point crossover, a point is chosen randomly, and the two parent

chromosomes exchange genes information after that point. If the new generations only inherit the genes from parents it may lead to a local minima problem that the chromosomes in the population become too similar to each other. The problem will slow or even stop the evolution of population. Mutation allows new genetic patterns to be introduced into population. However, there is no guarantee that mutation will produce desirable features in the new chromosome. The mutation of GA can base on genes or chromosome. Genes based mutation is employed by our work where we swap the bits of each genes (0 becomes 1 and vice versa). Generally, the mutation rate used in a GA is very low. In our work, we use 0.4% as our mutation rate.

In order to guarantee the termination the number of generations of the evolution is bounded by 1000. Each time when the new generations were created the fitness function will rate them. If there is a chromosome's fitness is over 95% then such a chromosome will be an optimal schedule and the algorithm will terminate.

5.6 Experimental Results

In order to evaluate the performance of our scheme, a set of simulation experiments is carried out. Sensors are randomly deployed in the 100×100 (meter) area, The number of sensors varies from 10 to 200. Each sensor's sensing range is 20meter and considered to be a disk. The relative positions of sensors will be denoted by their distances and angles. Each on-duty interval lasts 10 minutes and the energy of a sensor is sufficient to support the sensor to work for 50 minutes. The rest parameters of network is shown in the Table I. Our genetic algorithms are implemented using TCL-GAUL. Some parameters of GA are shown in the table II. Initial chromosomes are randomly generated and satisfied the condition of formula 5.5. We use 100 individuals for the initial population of our simulation. The performance of genetic algorithm with different fitness function is evaluated in terms of overall coverage, node distribution, the number of active sensors, estimation accuracy and running time.

Table 5.1: Parameters of Network

size of network	100*100(meter)
sensing range	20 meters
The number of sensors	10 200
length of a time slot	10 mins
energy of a sensor	support 50 mins
k coverage	1-3
lifetime of network	60 150mins

Table 5.2: Parameters of Genetic Algorithm

GA Prameters	
Population Size	100
Number of Generations	150
Cross Over	one point
Cross Over Rate	0.9
Mutations Type	bit flip
Mutation Rate	0.1

Network Coverage Vs. Network Lifetime The scheduling algorithm can improve the overall coverage of wireless network. As shown in Figure, we extend the lifetime of network from 60 to 150 and compare the overall coverage to the original network without scheduling. Since the energy of each sensor only allows the sensor work for 50 minutes, even without the scheduling algorithm the network can maintain 100% coverage for every monitoring time slot, the overall coverage reduces to 33% when the lifetime of network is extend to 150 minutes. However, by using the scheduling algorithm, the network can maintain 96% overall coverage in the same scenario.

In the Figure 5.6 and Figure 5.7, we show the changes of coverage and overall coverage when users request different length of network lifetime. As we define in our parameter table, each sensor can support 25 mins work. Therefore, without our scheduling algorithm the network will last for 25 mins and support 100% 1-cover. When users request 75mins and 100mins network lifetime, sensors are scheduled in different time slots. As shown in Figure 5.6, in each time slot we still have sufficient nodes to cover almost the entire area. Therefore, we extend overall 1-coverage as shown in Figure 5.7. When users request 125 mins lifetime, in order to maximize the overall coverage, the algorithm reduces the coverage in each time slot by scheduling less sensors in the same time slot as shown in Figure 5.8.

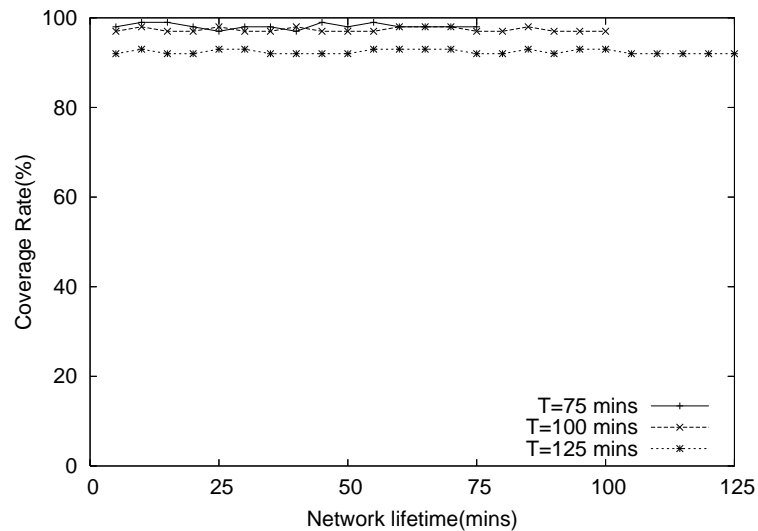


Figure 5.6: Coverage Vs. network lifetime

The Number of Active Sensors Vs. Network Lifetime In Figure 5.8, we show the number of sensors changes by the time. Comparing the results of "75 mins" and "100 mins", we can notice an interesting thing that the algorithm keeps 10 more sensors working in order to get less than 1 percent coverage increasing. When the service time increases to "125 mins", our algorithm reduces around 10 more sensors in each time slot

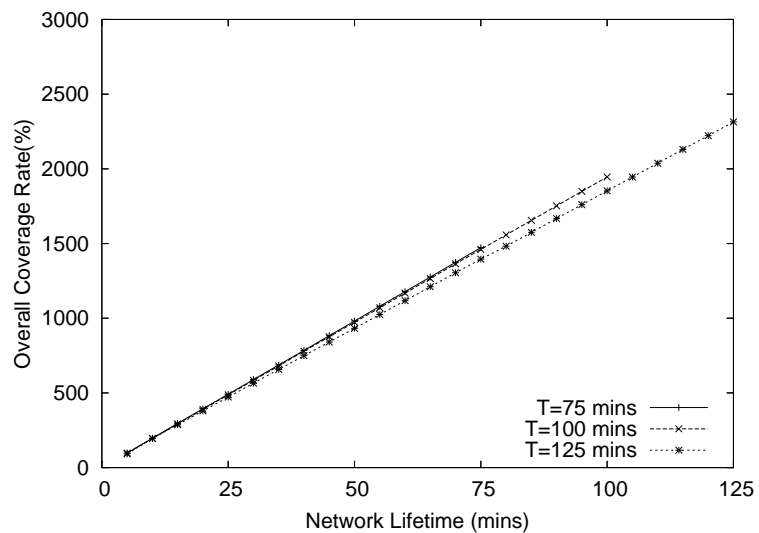


Figure 5.7: Overall coverage Vs. network lifetime

in order to gain a 20% increase on the overall coverage.

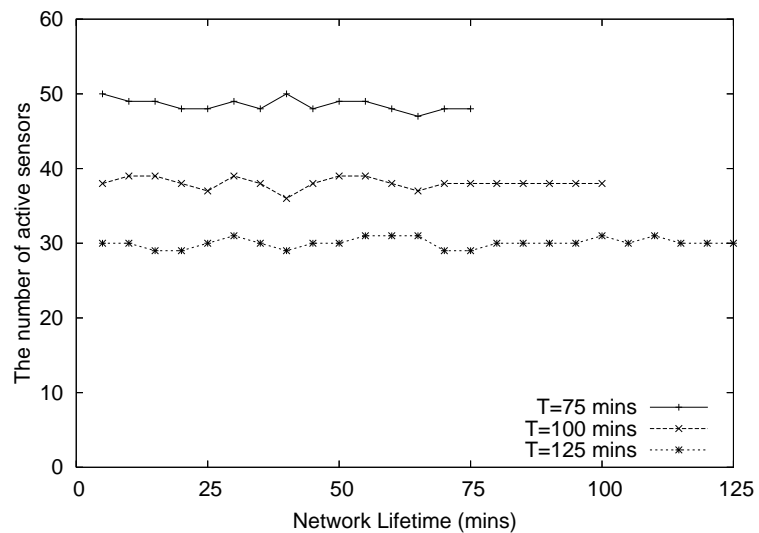


Figure 5.8: The number of active sensors Vs. network lifetime

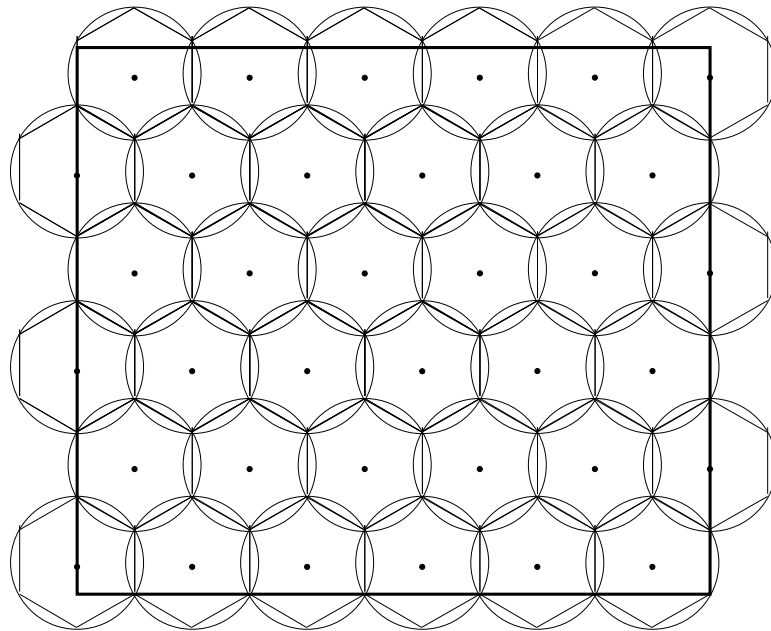


Figure 5.9: Honeycomb distribution, 1-cover

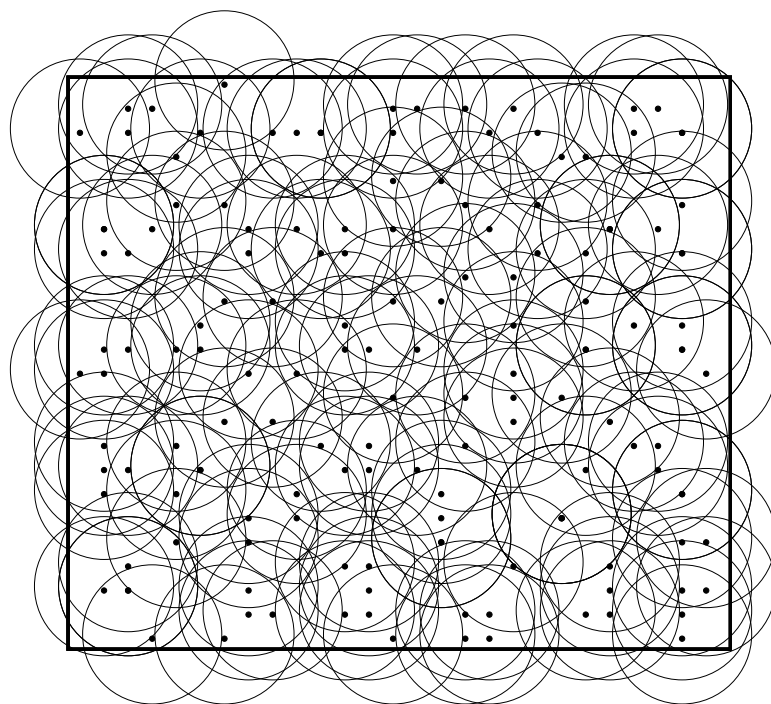


Figure 5.10: Distribution for grid based GA after 20 generations, 100 nodes, 1-cover

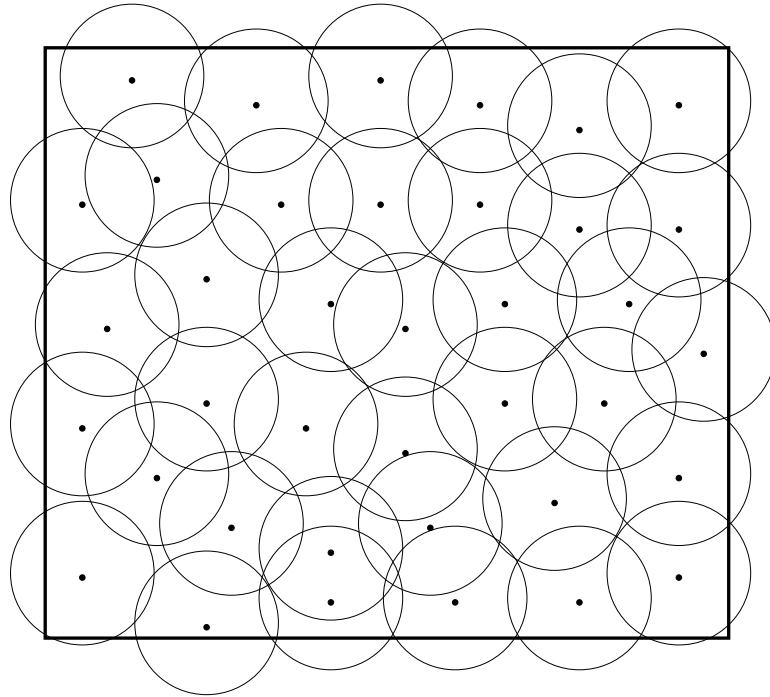


Figure 5.11: Distribution for set-covering based GA after 20 generations, 100 nodes, 1-cover

5.6.1 The Node Distributions

Because honeycomb is the best 1-cover node distribution which can be obtained literally, we use it as best scenario and compare it with the scheduling generated by our algorithm. In order to maximize the overall coverage, the scheduling algorithm also has to reduce the coverage redundancy in each time slot. In this section we compare the node distribution generated by GA for the first time slot with the honeycomb arrangement. Based on the estimation using Honeycomb arrangement for 1-cover, 36 nodes are deployed in a 100m*100m square area as shown in the Figure 5.9 . The initial distribution of 150 sensors is shown as Figure 5.10. For 1-cover, after 200 runs, the node distribution in the first time slot is shown in Figure 5.11, the new node distribution becomes more uniform than the initial one.

5.7 Conclusion

Using the concept of set covering, we introduce the test point based fitness function for coverage estimation in the genetic algorithm. Comparing to the grid based algorithm, the accuracy of testing does not affect the complexity of test point based algorithm which is only sensitive to the change of sensing range and deployment density. The complexity of test point based algorithm is less than the classic grid based fitness function in most scenarios. The two types of function are implemented using tcl-gaul. The performance of two type functions is compared in terms of running time, overall coverage and network lifetime. The simulation results show that the test points based genetic algorithm can improve the overall coverage and use less running time than the grid coverage based genetic algorithm and it is more suitable for the coverage-aware application requiring high test accuracy than the grid based solution.

Chapter 6

MDP Based Coverage-preserving Scheduling Protocol

6.1 Introduction

Two key QoS measurements of wireless sensor network are the lifetime of network and the coverage which presents how well the surveillance areas have been monitored by the application. In order to prolong the lifetime of network while maintaining the coverage quality, many coverage-preserving scheduling protocols have been proposed. In the protocols, sensors are partitioned into sets while each set can maintain a required coverage quality. However, most current protocols use boolean coverage model or probability coverage model in the partition process and assume that the energy of sensors are mostly used by data transmission. They ignore that the coverage quality and energy usage of sensors could be changed by many factors such as temperature, light, obstacles and so on. The classic boolean coverage model and gaussian distribution based probability coverage model are hard to describe such changes. When such changes happen in the network, even though the lifetime of network is prolonged by using less sensors at each second, the coverage quality could not be maintained because the sensors in a set dissipate energy in different speed. By observing the change probability of coverage and energy in the

network, a wireless sensor network (WSN) could be treated as a random process due to the fact that many influencing factors impact on its performance. By model modeling the behaviors of WSN as Markov chain, we can analyze the variety of the current network status, forecast the status changes and get optimized strategies.

In this section, we model the coverage quality and energy states of a sensor as two Markov chains whose states are changed by pre-obtained transition probability matrices. The network is formulated as a restless bandit problem. By resolving a Markov decision process (MDP) we propose our scheduling algorithm which optimizes both coverage quality and lifetime of network.

6.2 Restless Bandit Formulation and Solution

Restless bandit problem is one of the extensions to the multi-armed bandit problem which is a well studied framework where a decision-maker must dynamically select multiple projects to get the maximum reward [47, 48, 49]. Restless bandit problem relaxes the assumption of multi-armed bandit problem that any non activated process remains fixed by allowing idle projects to change states between decision times.

6.2.1 The Restless Bandit Problem

There are N projects, of which M can be active at a time period. A project n is characterized at (discrete) time t by its state s_n^t , which belongs to a finite state space. If project n works at time t , one receives a rewards $r_{s_n^t}$. The state s_n^t then evolves to a new state according to given transition probabilities. The goal is to find a policy which decides at each time period which project should work in order to maximize the expected sum of the discounted rewards over an infinite horizon. In our scheme, we use cost instead of reward, the optimization objective is to minimize the cost.

6.2.2 System Formulation

A cost-based restless bandit problem is a discounted Markov decision problem (MDP) $\{(S_n, P_n^1, P_n^0, R_n^0, \beta), 1 \leq n \leq N, M\}$ with the following elements:

Definition 34 *State Space:* The set of all possible states of system is the Cartesian product $S = \times_n^N S_n$ where S_n is the state space for bandit n , $1 \leq n \leq N$. The state of the process at time t is $X^t = \{x_1^t, \dots, x_N^t\}$ where x_n^t is the state of bandit n at time t .

Definition 35 *Action Set:* At each decision epoch the collection of actions at $X \in S$ is given as following:

$$A = \left\{ a = (a_1, \dots, a_N) \mid \sum_{i=1}^N a_i = M, a_i \in \{0, 1\} \right\} \quad (6.1)$$

Definition 36 *Costs:* If a transition from x to x' occurs in bandit n under action a_n at time t a cost $c_n^{a_n}(x, x')$ is applied. Costs are additive across bandits and over time. We use the notation:

$$C_X^a = \sum_{i=1}^N c_i^{a_i}(x_i) \quad (6.2)$$

for the aggregated reward earned when taking action $a()$ in state X .

Definition 37 *Transition Probabilities:* Suppose action a^t is taken at time t . For $a_n^t = 1$, bandit n evolves according to Markov law P_n^1 .

$$P \left\{ x_n(t+1) = x' \mid x_n(t) = x, 1 \right\} = P_n^1(x, x'), x, x' \in S_n \quad (6.3)$$

For $a_n^t = 0$, bandit n evolves according to Markov law P_n^0 , i.e.

$$P \left\{ x_n(t+1) = x' \mid x_n(t) = x, 0 \right\} = P_n^0(x, x'), x, x' \in S_n \quad (6.4)$$

Definition 38 *Policy:* A policy is a rule for taking actions. We denote by U the class of all admissible policies. The admissible policy $u \in U$ is a $T \times N$ matrix, whose element of the t th row and the n th column is a_n^t , representing the action taken by node n in time

slot t . The goal of policy optimization is to achieve the minimal cost. According (6.2) the optimal policy u^* is

$$u^* = \arg \min_{u \in U} Z(u). \quad (6.5)$$

Definition 39 *Priority Index*: The priority index for potential node n with state s_n^t at time t is represented as δ_{k_n} . The optimal policy has an index rule: The M nodes with the smallest indices in a given time slot t act as the active nodes. That is, assuming $\{\delta_{k_1}, \delta_{k_2}, \dots, \delta_{k_M}\}$ to be the set of indices arranged from the smallest value to the largest value in time slot t , the node n 's action should be

$$a_n^t = \begin{cases} 1, & \text{if } n \in \{k_1, k_2, \dots, k_M\} \\ 0, & \text{otherwise} \end{cases}. \quad (6.6)$$

6.2.3 Solving the Restless Bandit Problem by LP Relaxation

In this subsection, to solve the restless bandit problem, a hierarchy of increasingly stronger LP relaxations [50] is developed based on the result of LP formulations of Markov decision chains (MDCs). The restless bandit problem can be formulated as the following linear program (LP):

$$Z^* = \min_{x \in X} \sum_{n \in N} \sum_{i_n \in S_n} \sum_{a_n \in \{0,1\}} c_{i_n}^{a_n} x_{i_n}^{a_n} \quad (6.7)$$

where $X = \{x = (x_{i_n}^{a_n}(u))_{i_n \in S_n, a_n \in \{0,1\}, n \in N} | u \in U\}$, i_n denotes the state of node n in state space S and $c_{i_n}^{a_n}$ is the cost for node n taking action a in state i . The first-order relaxation can be formulated as the linear program:

$$Z^1 = \min_{x \in X} \sum_{n \in N} \sum_{i_n \in S_n} \sum_{a_n \in \{0,1\}} c_{i_n}^{a_n} x_{i_n}^{a_n}$$

subject to

$$\begin{aligned} x_n &\in Q_n^1, n \in N, \\ \sum_{n \in N} \sum_{i_n \in S_n} x_{i_n}^1 &= \frac{M}{1-\beta} \end{aligned} \quad (6.8)$$

where Q_n^1 is the performance region of the first-order MDC corresponding to project n and $|\varphi_{\max}| = \max_{n \in N} |\varphi_n|$, with the size polynomial in the problem dimensions [50].

Primal-Dual Priority-Index Heuristic

In this subsection, a heuristic for the restless bandit problem that uses the information contained in optimal primal and dual solutions to the first-order relaxation (6.8) is used. The primal-dual heuristic is interpreted as a priority-index heuristic as well. The dual of (6.8) is

$$D^1 = \min \sum_{n \in N} \sum_{j_n \in S_n} a_{j_n} \lambda_{j_n} + \frac{M}{1 - \beta} \lambda$$

subject to

$$\begin{aligned} \lambda_{j_n} - \beta \sum_{j_n \in S_n} p_{i_n j_n}^0 \lambda_{j_n} &\geq c_{i_n}^0, i_n \in S_n, n \in N, \\ \lambda_{j_n} - \beta \sum_{j_n \in S_n} p_{i_n j_n}^1 \lambda_{j_n} &\geq c_{i_n}^1, i_n \in S_n, n \in N, \\ \lambda &\geq 0 \end{aligned} \tag{6.9}$$

We denote by $\{\overline{x_{i_n}^{a_n}}\}$ and $\{\overline{\lambda_{i_n}}, \overline{\lambda}\}$ the optimal primal and dual solution pair to the first-order relaxation (6.8) and its dual (6.9). Let $\overline{\gamma_{i_n}^{a_n}}$ represent the corresponding optimal reduced cost coefficients:

$$\begin{aligned} \overline{\gamma_{i_n}^0} &= \overline{\lambda_{j_n}} - \beta \sum_{j_n \in S_i} p_{i_n j_n}^0 \lambda_{j_n} - c_{i_n}^0 \\ \overline{\gamma_{i_n}^1} &= \overline{\lambda_{j_n}} - \beta \sum_{j_n \in S_i} p_{i_n j_n}^1 \lambda_{j_n} - c_{i_n}^1 \end{aligned} \tag{6.10}$$

which must be non-negative. Furthermore, $\overline{\gamma_{i_n}^0}$ and $\overline{\gamma_{i_n}^1}$ could be interpreted as the rates of decrease in the objective-value of linear program (6.8) per unit increase in the value of the variable $x_{i_n}^0$ and $x_{i_n}^1$, respectively. Based on the cost coefficients computed in (6.10), the index of the sensor n in state i_n is defined as:

$$\delta_{i_n} = \overline{\gamma_{i_n}^1} - \overline{\gamma_{i_n}^0} \tag{6.11}$$

The priority-index rule is to select the M nodes that have the smallest indices to be active. In case of ties, set active node with $\overline{x_{i_n}^1} > 0$.

6.3 Problem Statement and Modeling

The solution of restless bandit problem give us an opportunity to predict an optimal schedule for entire sensor network. Let us consider such a scenario that, at each time slot, we select M sensors which remains high energy level and can provide high coverage as well. If we assume that the coverage and energy of sensors are independent Markov chains, then based on the presented solution, we can predict the schedule for entire network. In order to achieve this goal, we model the scheduling problem by restless bandit problem and present our algorithm in the rest of this section.

6.3.1 The Coverage Model

In order to evaluate the coverage of area, we divide the entire space into unit grids. If a grid is k -covered and in the range of sensor i , then the coverage rate of sensor i increase one. The coverage state d_i^t of sensor i at the time instant t is evaluated through dividing the coverage rate of sensor i by the number of grids in its range. The coverage states evolve according to an L-state Markov chain with one-step transition probability matrix A_n^a :

$$A_i^a = (\phi_{pq})_{p,q \in I_i} = \Pr(d_i^{t+1} = q | d_i^t = p) \quad (6.12)$$

where a stands for an action. In our system, we consider two actions $\{0, 1\}$ which refer to OFF and ON action respectively. So A_i^1 is the transition probability matrix for turning on sensor i and A_i^0 is the transition probability matrix for turning off sensor i .

6.3.2 The Energy Model

Most mobile devices are powered by batteries with limited energy, the energy should be used carefully to avoid over-utilizing of certain nodes and maximize the network life. The residual battery energy can be detected locally as e_i^t . For simplification, the continuous battery residual energy can be divided into discrete levels, denoted by $E = (e_1, e_2, \dots, e_h)$, where h is the number of available energy state levels. Inspired by [51], we model

the transition of energy levels of sensors as a Markov chain with one-step transition probability matrix:

$$B_i^a = (\phi_{pq})_{p,q \in I_i} = \Pr(e_i^{t+1} = q | e_i^t = p) \quad (6.13)$$

6.3.3 The Cost Model

The costs associated with node selection are composed by coverage cost $c_d(d_i^t, a_i^t)$ and energy cost $c_e(e_i^t, a_i^t)$. At time t , the instantaneous cost incurred due to the selection of sensor i is:

$$c_i^t = (1 - \gamma)c_d(d_i^t, a_i^t) + \gamma c_e(e_i^t, a_i^t) \quad (6.14)$$

where γ is the weight factor for the two kinds of costs.

We define the cost of action in coverage model as the coverage contribution of the sensors (for simplification, we only consider the 1-cover scenario). By dividing area into grids, we determine the coverage contribution (CC) of sensor i as following:

$$\begin{aligned} CC_{grid} &= 0 \quad \text{or} \quad \left(1 + \sum_{i=1}^k \frac{1}{\omega^{*k}}\right) \\ CC_i &= \sum CC_{grid} \\ c_d(d_i^t, OFF) &= -CC_i \\ c_d(d_i^t, ON) &= CC_i \end{aligned} \quad (6.15)$$

where k is the coverage degree of a grid, ω is the parameter to adjust the contribution of grid whose coverage degree is higher than 1.

The cost of energy model $c_e(e_i^t, a_i^t)$ is the average energy usage in percentage for a sensor in its working interval. We define $c_e(e_i^t, a_i^t)$ as following:

$$\begin{aligned} c_e(e_i^t, ON) &= Energy/interval \\ c_e(e_i^t, OFF) &= 0 \end{aligned} \quad (6.16)$$

Because we select M active nodes at each time, the cost of system at time t is $l(t) = \sum_{i=1}^M c_i^t$, where $i \in [1, \dots, M]$. The total expected discounted cost of over infinite time

horizon is given by:

$$Z(u) = E \left[\sum_{t=0}^{\infty} \beta^t l(t) \right] \quad (6.17)$$

where u denotes policy which is the all actions taken in the life span of network; E denotes mathematical expectation; $\beta \in (0, 1)$ is the discount factor to ensure the expectation is bounded. The optimization objective is to find the optimal policy u to minimize the cost $Z(u)$.

6.4 The Scheduling Algorithm

One issue of our model is how to determine the M . In our work, we try to maintain the 1-cover coverage quality for entire area. Therefore, we select the M by choosing the minimum number of sensors that can maintain 1-cover for an area. It has been proved that hexagon arrangements is the most economical arrangement in lattice arrangements. Base on such arrangements, The lower bound of coverage density of circle covering for a rectangle plane has been given by Kershner as $2\pi/\sqrt{27}$. For a given area AR and sensing range r , the M can be determined by following equation:

$$AR * 2\pi / \sqrt{27} = M * 2\pi r^2 \quad (6.18)$$

After M is determined, the priority index of entire network is computed off-line. When the priority index is ready, the scheduling process can start at any time by collecting the current state of sensors. Based on priority index and current state of sensors, M sensors with the lowest priority index will be selected as active nodes. The final decision is broadcasted to sensors. When the M selection can not be satisfied, we consider the network is dead.

The pseudo code of the scheduling algorithm is shown as algorithm 8.

Algorithm 8 Node Selection Algorithm

Determine M by sensing range and monitoring area;
 Off-line computing priority index;
 State collection;
 Node selection;
if M nodes are selected **then**
 Distributing the schedule;
else
 Algorithm terminating;
end if

6.5 Simulation Experiments

In this section, we illustrate the performance of the proposed scheme. The simulation results of our protocols are compared to the random node selection scheme, boolean model based coverage-preserving node selection scheme and range-detectable coverage-preserving node selection scheme. The performances of schemes are evaluated in terms of coverage quality and lifetime of network.

6.5.1 The Simulation Environment

In order to simplify the simulation experiment scenarios, we consider a single cluster network with 80 sensors which are randomly deployed in a $40m * 40m$ area. Each sensor can communicate with sink directly. The sensing range of sensor is 11m. Based on Kershner's lower bound of coverage density, in each round we keep 5 sensors active ($M = 5$). The evolution of coverage and energy states of sensors are considered as two independent Markov chains. For the reason of computation complexity, in the simulation, we use only four coverage states $\{[0], [1, 30], [31, 60], [61, 100]\}$ (in percentage of coverage) and three energy states. The transmit probability matrix and cost matrix for each model are defined as Table 5.4 to 5.5.

Table 6.1: Transition Matrices for Coverage Model

Transition matrix for Action ON (1)				
	s4	s3	s2	s1
s4	0.7	0.3	0	0
s3	0.15	0.7	0.15	0
s2	0	0.15	0.7	0.15
s1	0	0	0.3	0.7
Transition matrix for Action OFF (0)				
s4	0.99	0.01	0	0
s3	0.005	0.99	0.005	0
s2	0	0.005	0.99	0.005
s1	0	0	0.01	0.99
Cost matrix for Action ON				
	3	9	27	84

Table 6.2: Transition Matrices for Energy Model

Transition matrix for Action ON (1)			
	e3	e2	e1
e3	0.98	0.02	0
e2	0	0.98	0.02
e1	0	0	1
Transition matrix for Action OFF (0)			
e3	0.999	0.001	0
e2	0	0.999	0.001
e1	0	0	1
Cost matrix			
	3	50	500

6.5.2 Experimental Results

At each simulation unit, the random node selection scheme and coverage-preserving node selection scheme run after the coverage states and energy states of sensors are decided. Random selection scheme randomly active M ($M=5$) nodes; The boolean model based coverage-preserving scheme select as less as possible of sensors to maintain the coverage rate; The range-detectable coverage-preserving scheme change the coverage of each sensor by their coverage state and find the minimum set of active sensors. The simulation terminates when there is less than M alive nodes in the network (the network is considered dead).

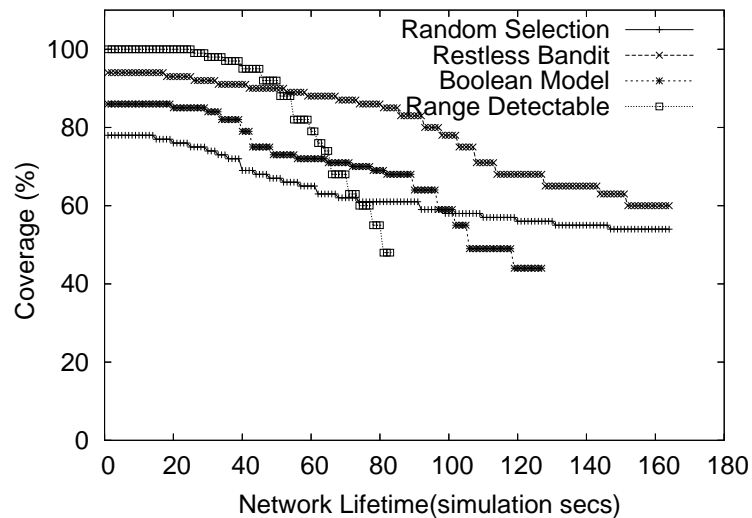


Figure 6.1: Coverage of network

The coverage of entire network is recorded at each simulation second. The result is shown in Figure 6.1. In the figure, range-detectable scheme have the best coverage quality; however, it dies after 84 simulation seconds. the proposed restless bandit based scheme have the second best coverage quality and longest the lifetime, 164 seconds. The boolean model based scheme ranks three in terms of coverage quality because it does not consider the changes of coverage states. The boolean model based scheme suffer a short lifetime for the same reason as range-detectable scheme. They active more sensors

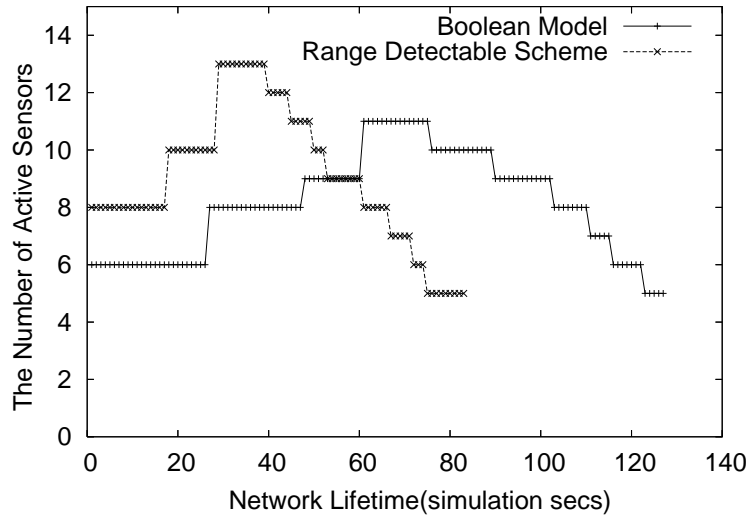


Figure 6.2: The number of active nodes

than other schemes as shown in Figure 6.2. The random selection scheme gives the worst performance of coverage in average of twenty runs because of its highly random coverage rate. However, the random selection scheme and our proposed scheme dies at the same time for using the same number of active nodes.

6.6 Conclusion

We have presented a centralized coverage-preserving scheduling protocol which schedules sensors by their coverage quality and energy states. Instead of using boolean coverage model and probability coverage model, the evolutions of coverage and energy states of sensors are modeled as Markov chain. The sensor selection process is formulated as a restless bandit problem and solved by a primal dual algorithm. We compared the performance of proposed scheme with random selection algorithm and full-cover algorithm. The simulation results show that the proposed scheme can improve both coverage quality and network lifetime.

Chapter 7

Mobile Coverage and Mobile Data Gathering Protocols for WSN

With the recent developments in inexpensive radio communication, data processing, storage, and battery power, wireless sensor networks (WSNs) have been employed by many applications to monitor environments inaccessible to humans, such as the battlefield, desert, and forest. In these applications, sensors detect events and forward the sensed data to a sink (base station) through multi-hop communication. As a result, a significant amount of energy is consumed especially by the nodes in the area near the sink. This causes unbalanced energy depletion and eventually leads to the disconnection of a network. Considering the price of battery recharging and replacement in a large WSN, researchers are looking for an alternative way to perform the same data gathering job in an energy efficient manner. With the advances of wireless networks and the widespread use of thin mobile devices such as cellular phones and personal digital assistants (PDAs), researchers are investigating the use of mobile entities to move within the monitored area and collect information from sensors at a very low cost, usually involving only single-hop communication. A mobile collection approach not only decreases energy consumption and traffic load, but also removes the burden on the nodes closer to a sink. In addition, the delivery success rate is increased too because multi-hop data propagation is replaced

by single-hop communication.

In a mobile collection approach, the behavior of the mobile elements greatly affects the performance of the system. In order to gather data from all the random deployed data sources, mobile collectors have to discover and visit the data sources as much as possible in a given time. It posts a great challenge to find an optimal way for dispatching mobile collectors to the targets while minimizing the moving and messaging cost. We refer such a challenge as a “mobile coverage” problem.

In this chapter, we consider the single-hop mobile data gathering with the energy efficiency and data freshness concerns. We first investigate the upper/lower bound of the covering time for a single collector to cover the monitoring area. Through our investigation we show that for a bounded rectangle area a hexagon walk could explore the area more efficiently than a random walk when the edges of area is known. We then propose a virtual force mobile model (VFM) in which the energy consumption for data transmission is modeled as a virtual elastic force and used to guide of mobile collectors to move to optimal positions for energy saving. Based on the VFM with hexagon walk exploration, we develop an efficient localized mobile data gathering protocol (VFMDG). It supports the collector to independently make moving decisions based only on the sensors in the single-hop neighborhood.

The remainder of this chapter is organized as follows. Section I presents the problem of data gathering and introduces some related works. Section II describes our virtual force model and Section III describes the virtual force mobile data gathering (VFMDG) protocol. Section VI discusses the experimental results obtained in the simulations. We conclude this study in Section V with future work.

7.1 Related Works of Mobile Data Gathering

Due to tremendous practical interests, recent research has been devoted to efficient data collection and many approaches have been proposed. Based on the sink’s capability in

the network, these approaches can be classified into two categories, static data gathering and mobile collection. Since the algorithm in the two categories share some common interesting on the performance, we first present some static data gathering protocols and then show the state of arts in the mobile data collection.

7.1.1 Static Data Gathering

In the category of static data gathering, sensors are organized into routing trees or hierarchical clusters, and packets propagate to a static sink through multi-hop relay. Considering the nature of tree-like network structures, the network capability, load balance and data redundancy are consistent concerns. England *et al.* use a spanning tree topology for data gathering in [52], which lowers the energy usage and is resilient to data loss. Jain *et al.* [53] consider the effect of interference on data gathering in a WSN. They model the interference using a conflict graph and determine an optimal schedule for maximizing the flow of information towards the destination. Duarte-Melo and Liu [54] study the capacity of data gathering under the protocol model. El Gamal [55] study a similar problem and investigate whether collaborative transmission schemes could improve system throughput. Other works such as [56] and [57] have tried to optimize the locations of nodes using a genetic algorithm and integer linear programming (ILP).

7.1.2 Mobile Data Gathering

In the mobile data gathering, instead of waiting for the arrival of messages, sinks are attached to a vehicle or aircraft to collect data from sensors directly. There are three advantages that make mobile data collectors (MDCs) suitable for data gathering applications in WSNs. First, this approach radically eliminates the non-uniformity of energy consumption among sensors. As each sensor sends data directly to its associated MDC, it will no longer consume energy on forwarding packets from other sensors as in traditional relay routing. Second, the approach works well not only in a connected network,

but also in a disconnected network. The moving path of each MDC acts as a virtual link among separated sub networks, thus eliminate the problems of the network coverage and connectivity. Third, when all possible locations for the MDCs are known, the tour of each MDC becomes predictable, which provides an opportunity to dynamically find an optimal tour to achieve efficient data gathering. With these three advantages, many protocols in this category have been proposed recently.

Considering the data propagation methods mobile data gathering protocols can be divided into two categories, single-hop data gathering and multi-hop data gathering. In each category, we further classify the protocols into two subcategories, controllable moving scheme and uncontrollable moving scheme, or reactive moving scheme and proactive moving scheme, based on their mobility management. In the controllable moving scheme the mobile element adjust their mobility to meet the changes of network and environment. Oppositely, in the uncontrollable moving scheme the mobile elements move randomly or following predefined optimal moving patterns based on a priori knowledge of network such as geometry map, the data generation probability and the location of data source. Comparing to the controllable moving scheme uncontrollable scheme is resistant to the communication error and detection errors.

Single-hop Data Gathering

The single-hop data gathering problem could be defined as following. Given a set of data-producing nodes where all the data must be gathered, and taking into account their locations, the goal of single-hop data gathering is to design a set of walks for MDCs, such that most nodes are visited by MDCs and the performance criteria including energy efficiency, through output, and time constraints are met.

In the uncontrollable moving schemes of single-hop data gathering, the mobile elements move in the area without reaction to the observation in the data gathering process. Researchers explore the mobility of mobile collectors to maintain the performance of protocols. The most widely used uncontrollable moving strategy is random walk because

of its autonomous characteristics, low topology requirement and low overhead. Random walk based protocols can work with any other data relay protocols without knowledge of network topology. The only information requested for the random walk is the boundary of the monitoring area. Sensor Networks with Mobile Access (SENMA) [58] is one of the protocols based on random walk. Instead of relaying messages by a routing protocol, data are sent to the MDCs flying above the sensor field. In SENMA, authors investigate the effects of altitude and trajectory, and show the improvement on channel capability and energy usage. They mainly focus on the channel capability and ignore other aspects of performance, such as latency. MULE [59] is another interesting single-hop mobile data gathering protocol. In MULE, the authors consider the MDC as a large mobile queue that gathers data from sensors and carries the data to the sink. The performance of MULE is evaluated in terms of the data success ratio and transmission delay under four mobility models: random way-point; random walk; deterministic arrivals (fixed route and velocity), and poisson arrivals. Simulation results show that while MULE improves the energy efficiency, the latency and delivery rate of the system are sacrificed.

As the most light-weight mobile data gathering protocol, random walk can guarantee the visit of every sensor in a bounded area even in a sparse deployment. In a long run, it offers acceptable performance when enough collectors are deployed. By giving different weight to the movement of mobile device, the performance of random walk could be further improved by scarifying their resistance to the errors. Since a biased random walk changes the probability of movement based on the detection and communication, it falls into the category of controllable moving scheme and will be described in the later of this section.

Instead of let the mobile collector move randomly in the area, some researchers try to determine an optimal trajectory of collectors after the sensors are deployed. By moving along the trajectory the optimal goals are reached. Since all the data is uploaded through single-hop communication, the energy efficiency is maintained in the predefined path strategy. In [60], Chakrabarti *et al.* consider a collector that repeatedly moves

along a predefined path and collects data from its one-hop neighboring sensors. Their analysis shows that based on the predictable movement, high energy efficiency can be reached by adjusting the communication range and buffer size of sensors, as well as the velocity of the collector. The big problem of this kind of strategy is that it is hard to adapt to changes in the network such as the location of sensors or the data generation rate. Any changes will cause the protocol to reset and find the new optimal trajectory, which leads to an interruption of services. Using a mobile element which follows a long tour through the network to collect the data from all sensors may introduce unacceptable delays in the delivery of the data to the sink. The length of report latency is dependent on the tour path and the speed of the MDCs. If the wait time is too long, data will be dropped from the sensors' buffer in order to accommodate the new data.

In the controllable mobility strategy, collectors deal with the dynamic of the network and environment by periodically detecting the system states and determining their moving velocity to meet the performance requirements. One of the performance metrics is the covering time which reflects the visiting fairness and visiting efficiency for the data sources. Since random walk can guarantee the visit of each sensor in a bounded area, it has been adopted in many localized mobile data gathering protocols. By introducing controllability into random walk researchers propose the biased random walk which has the advantages of random walk and without its shortcomings. In [61], authors propose three biased random walks: the random walk with inertia, the explore-and-go random walk and the curly random walk to further balance the memory and performance. By comparing to the blind random walk and biased walk with memory, their simulation results show that the performance of proposed random walk is close to the biased walk with memory in terms of covering time while for the partial covering time and proximity variation the proposed random walk even outperforms the classic biased walk.

Another performance metric used to determine the movement of MDC is the energy usage. In [62], Gandham *et al.* determine the movement of a sink using the energy usage. In their work, multiple mobile sinks periodically change their locations to optimal posi-

tions obtained by an ILP model. The advantage of such a solution is that collectors can adapt to the event distribution, however, resolving the ILP problem requires a powerful CPU and memory resources. Besides, the periodic path planning process involves high overhead and requires the knowledge of the entire network and monitoring area.

Network lifetime is another performance metric close to the energy usage. In paper [63], Basagni etc. define a Mixed Integer Linear Programming (MILP) model to find the optimal moving routes for maximizing the network lifetime. Two distributed and localized algorithms are proposed based on the energy saving concern and random movement. Their simulation result shows that comparing to uncontrolled mobility model controllable mobility model leads to remarkable improvements on performance of network.

Multi-hop Data Gathering

In the single-hop data gathering, the sensors have to wait for the arrival of collectors even though the collector is only 2 or 3 hops away. In order to avoid this problem, instead of using single-hop communication to save energy, the multi-hop data gathering maintains the routing trees rooted at the MDCs. Since MDCs roam around the monitoring area, the balance of energy consumption is maintained. Sensors can deliver their data at any time without waiting for the arrival of collectors because the network always keeps its connectivity. However, this strategy is not suitable for scenarios where the connectivity of network cannot be maintained. The periodic routing rebuilding process leads to a heavy overhead, which consumes the precious energy of the sensor. The mobility of MDCs is limited to avoid dropping packets.

Some researchers attack the multi-hop data gathering problem by investigating the routing tree maintenances. In [64], the sink moves around the periphery of the covered circular region to ensure the balance of energy consumption. In [65], an adaptive reversal tree (ART) protocol is presented to update routing tables when the sink roams in the area. Based on the adaptive reversal algorithm and dynamic root change mechanism, data disseminate from sensors to a mobile sink. The partitioned tree is quickly recovered

by ART in order to maintain a robust routing tree. In [66], Yang etc. aim at discovering the shortest path from data source to the mobile sink. They propose a SIMPLE protocol using swarm agents to build the shortest path with high residual energy. The SIMPLE is resilience to the node failures by the multiple path nature. Authors analytically investigate the scalability, cost, robustness of their algorithm and validate the correctness of SIMPLE. In paper [67], authors build a framework to theoretically analyze the joint problem of sink mobility and routing construction by constraining the sink to a finite number of locations. A primal-dual algorithm is proposed to solve the induced subproblem with single-sink and then is generalized to approximate the problem with multiple sinks. The algorithm is applied to different types of graphs to prove their advantages.

Finding a set of optimal gathering points or nodes to visit is another interesting research in the mobile data gathering. In paper [68], authors investigate a rendezvous-based data collection problem where a set of node act as rendezvous node which cache the data from data sources and forward to the BS via short-range transmissions when mobile collector arrives. All the data in the network should be delivered to the mobile collector in time T while the energy consumption for data propagation is minimized. Targeting such a problem authors proposed two protocols. In the first algorithm, a subset of an approximate sterin tree is found. Therefore, the mobile collector can visit all the rendezvous nodes while satisfying the constraints of travel length. In the second algorithm, an approximate minimum spanning tree is built to maintain the rendezvous points on the fixed moving tracks of mobile collectors. The performance of protocols is theoretically analyze and the performance bound are derived. In paper [69], Wang etc. study the lifetime upper bounds of a densely delayed network with sensors uniformly deployed in the circular area with radius R and prove that the mobile sinks can maximize the lifetime of network by only staying in the 2-hops area around the static sink. In such a circumstance, they construct two algorithms ARA and ARALN which joint mobility and routing and yield a network lifetime close to the upper bound. In the algorithms the monitoring space is divided into aggregation rings. The data are first aggregated in the

ring and then forward to the sink. Their simulation results show that one mobile relay can at least double the network lifetime in a randomly deployed WSN. In paper [70], in order to maximize the network lifetime with a guaranteed event collection rate, authors exploit event's spatial temporal correlation and chose only a portion of static sensors to communicate with mobile sink. The sensor selection problem is proved to be solved in polynomial time, if global knowledge of events is available and there is no velocity constraint on mobile sink. The constraint of global knowledge of events is remove the lately in their online scheme.

Instead of explore the mobility of mobile element, some protocols focus on the data dissemination without the knowledge of mobile capability. In paper [71], authors investigates proactive data dissemination problem to allow the mobile sink gather the enough data by only visiting a subset of nodes. Authors introduce a DEEP protocol which is based on probabilistic flooding. The protocol is compared with RaWMS, a mobile data gathering protocol based on random walk, in terms of data gathering efficiency, communication overhead and data distribution quality. The result shows that DEEP has the similar performance as SaWMS in terms of data gathering efficiency as well as use less messages than SaWMS.

7.2 Problem Statement and Analysis

In this work, we consider a heterogeneous monitoring application where different type of sensor is deployed in the area. Since the sensors did not share the same routing protocol, the connectivity of data sources in the area is not guaranteed even the density of deployment is high. Therefore, a relay network could not be established. In order to gather the sensed data while maintain the freshness of data, mobile collectors are sent to the area and work in a “walk-stop” model. Since in the applications such as emergency preparedness and battle field monitoring the geometry info of data sources and monitored area are not known a priori, a localized mobile data gathering protocol is more preferred

than a centralized or distributed solution. In such a circumstance, we investigate a joint problem of saving the energy of sensors and maintaining the data freshness in the single collector mobile data gathering. Before we start the investigation of problem, we first define some terms using in our work.

Definition 40 *Data freshness is the time interval from the time that data is sensed to the time that data is gathered by a collector.*

Definition 41 *Gathering rate, $Gr = \sum Dg/T$, is the average volume of gathered data per time unit in a given time span T .*

Definition 42 *Covering time is the time of all sensors in the area being visited.*

7.2.1 Energy Efficiency of Single-hop Data Gathering

In order to maximizing the data gathering rate, usually the MDC collects data from all the sensors in its communication range. Since each sensor may carry different quantities of data, some sensors may have to transfer data to the MDC using extra energy. If we can determine the most energy efficient location and let the MDC only collect data at such place the energy of data transmission could be saved. In the rest of this section, we will first review the radio energy model of sensor and then propose our virtual force model.

Radio Energy Model

The energy consumption of sensors mainly occurs during four tasks: transmitting; receiving; sensing; and processing. Compared to the energy usage for data transmission, the energy cost for sensing and processing E_{elec} is relatively low, which is usually consider as $E_{elec} = \alpha E_{ampl}$ where α is a number between zero and 0.5. In our work, we use a simplified radio model in which the transmitter dissipates energy to run the radio electronics and power amplifier, while the receiver only dissipates energy to run the radio electronics.

Based on this radio model, we can formulate the energy usage for transmitting m bits as follows:

$$E_T = mE_{elec} + mE_{ampl}, \quad (7.1)$$

where E_{elec} is the energy cost of modules, such as the coding and modulation modules that process data before transmission. Assuming that E_{elec} is a constant small value for every sensor, the energy usage of a transmission E_T is only related to the energy consumption of the amplifier E_{ampl} .

Different assumptions about the radio characteristics will change the performance of different protocols. In this work, we choose the channel propagation model used in NS2 and [72], where both the free space model and the multi-path fading model are used for different distances between the transmitter and receiver. A crossover distance is defined as $d_{crossover} = \frac{4\pi\sqrt{L}h_r h_t}{\lambda}$, where L is the system loss factor, h_t and h_r are the height of the transmitting and receiving antennas above ground, respectively, and λ is the wavelength of the carrier signal (for consistency we adopt the notations of NS2). When the distance between two sensors is lower than $d_{crossover}$, the Friss free space model is used; otherwise, the two-ray ground propagation model will be adapted [72].

Since the power threshold P_{r-th} for successfully receiving a signal is determined by the sensitivity and noise level of the receiver, when P_{r-th} is given, we can compute E_{ampl} and the transmission power P_t for different propagation models as follows.

$$\begin{aligned} & \text{when } d \leq d_{cross-over}, (\text{Friss free space equation}) \\ P_t &= \frac{P_{r-th}(4\pi d)^2}{LG_t G_r \lambda^2} = \omega_{friss} d^2, \quad \omega_{friss} = \frac{P_{r-th}(4\pi)^2}{LG_t G_r \lambda^2} \\ & \text{when } d \geq d_{cross-over}, (\text{Two-ray ground propagation equation}) \\ P_t &= \frac{P_{r-th} d^4}{G_t G_r h_r^2 h_t^2} = \omega_{two-ray} d^4, \quad \omega_{two-ray} = \frac{P_{r-th}}{G_t G_r h_r^2 h_t^2} \\ E_{ampl} &= \begin{cases} m\omega_{friss} d^2 \\ m\omega_{two-ray} d^4 \end{cases} \end{aligned} \quad (7.2)$$

Based on the formulation in (7.1) and (7.2), the total energy consumption E_{total} for

transmitting data to the sink through multi-hop propagation can be formalized as:

$$E_{total} = \begin{cases} \sum_{i=1}^n \sum_{j=1}^{hops_i} m_i \omega_{friss} d_{ij}^2 \\ \sum_{i=1}^n \sum_{j=1}^{hops_i} m_i \omega_{two-ray} d_{ij}^4 \end{cases} \quad (7.3)$$

where n is the number of sensors, Therefore, the mobile data gathering algorithm can save the energy of network only when the following condition is satisfied (using Friss free space propagation model as an example):

$$E_{gain} = \sum_{i=1}^n \left(m_i \omega_{friss} \left(\sum_{j=1}^{hops_i} d_{ij}^2 - \sum_{j=1}^{colhops_i} d'_{ij}{}^2 \right) \right) \quad (7.4)$$

where E_{sink} is the energy consumption for propagating all data to the sink through the sensors; $E_{collector}$ is the energy usage for gathering data through MDCs, $hops_i$ and $colhops_i$ are the number of hops from node i to the sink and mobile collector respectively; d'_{ij} is the distance of the j hop from node i to mobile sink and E_{gain} is the energy difference between the two methods. Compared to the energy usage of amplifier E_{ampl} , E_{elec} is very small. If we ignore the energy usage of other modules E_{elec} , and only consider the transmission under the free space model, the energy usage for transmitting m bits is $mE_{ampl}d^2$. From (7.1), we can see that E_{ampl} can be considered a constant when p_{r-th} , G_t , G_r and λ are given. In order to minimize $\sum_{j=1}^{colhops_i} d'_{ij}{}^2$ in (7.4), MDCs need to move toward a position where they can gather more data (m_i) than the others and reduce the hops ($hops_{col}$) and distance (d_j) for data transmission.

Virtual Force Model

Our virtual force model targets at an energy-efficient mobile gathering problem where the gathering protocol maximizes the gathering rate while minimizing the energy usage by moving the collector in a controllable manner [73]. Considering a spring system where multiple springs are attached to a single object, as shown in Figure 7.1. When the object is displaced by an external force, based on the principle of minimum total potential

energy, the spring system will always try to reform to a stable state where the potential energy of the entire system is at a minimum. For a set of points in a 2D plane, the point with the minimum total potential energy is their weighted centroid or mass center. It can be computed using the following equation:

$$C = \frac{x_1 + x_2 + \dots + x_k}{K}. \quad (7.5)$$

Based on Hooke's law, the restore force F for a string is $F = -kx$, where x is the displacement of the spring's end from its equilibrium position, and k is the spring constant. The potential energy U of the displaced string is

$$U = Fx = -kx^2. \quad (7.6)$$

Consider the transmission model where $d < d_{crossover}$, we notice that the energy usage for transmitting m bits over distance d is very similar to the potential energy of a stretched string where the spring constant K is determined by $m\omega_{friss}$ and $d = x$.

$$U = \int Fx = \frac{1}{2}kx^2 = P_t = m\omega_{friss}d^2 \quad (7.7)$$

If we simulate the energy usage for transmitting data as the potential energy of a displaced string system, the problem of moving a collector to an optimal point for minimizing $\sum m\omega_{friss}d^2$ is equal to finding the equilibrium state of the system where the potential energy of the entire system is at a minimum. As shown in the Figure 7.1, when data is sensed at a sensor, the sensor generates a virtual pulling force $F = \omega_{friss}x \sum m_i$ on the collector to drag it close to the sensor to reduce the energy used for data propagation. The collector is displaced from the stable state by many virtual forces and must try to restore the stable state. The net force \vec{F} defines the velocity of movement, which determines the report latency of data gathering and the energy usage for driving the mobile element, $E = \frac{1}{2}mv^2$. The energy saved by moving the collector from the original location to the centroid is

$$\begin{aligned} & \sum_{i=1}^k m_i \omega_{friss} (d_i^2 - d_{ic}^2) \\ &= \sum_{i=1}^k m_i \omega_{friss} \left((x_i - x_c)^2 + (y_i - y_c)^2 - \left(x_i - \frac{\sum (m_i x_i)}{\sum m_i} \right)^2 + \left(y_i - \frac{\sum (m_i y_i)}{\sum m_i} \right)^2 \right). \end{aligned} \quad (7.8)$$

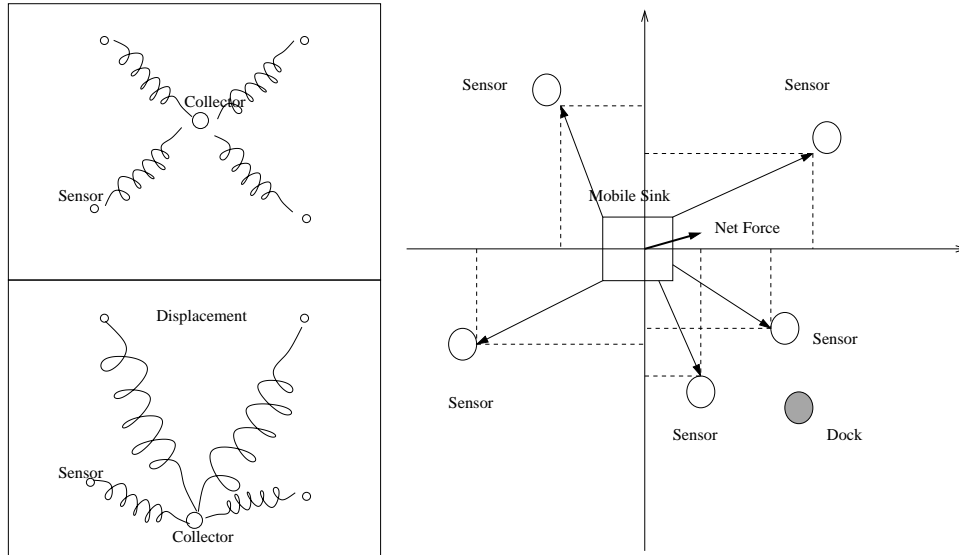


Figure 7.1: Potential energy.

Since the collector only communicates with single-hop sensors and the centroid of a set of points is always located in the middle, the moving distance d_c from the original location to the centroid is a number between $[0, R]$, where R is the communication range of the sensors.

Even we derive the virtual force model from the radio energy model, the virtual force model is flexible to combine with other performance factors. Beside the data gathering rate and energy efficiency it allows user to simulate other factors such as data freshness. A virtual force with data freshness concern can be express as $F = \omega_{friss} x \sum m_i * \eta * t_i$ where t_i is the waiting time of data m_i and η is a parameter to balance the weight of data gathering and data freshness.

7.2.2 Data Freshness and Area Exploration

The data freshness reflects the fairness and efficiency of node exploration in the field. Let us consider a bounded 2D plane with n randomly deployed static sensors. Each sensor periodically generates data and tries to upload the data to the mobile collector. We can model the application as that a graph $G = (V)$ consisting of a set of nodes $V \subseteq \mathbb{R}^2$ with

weight $w \in W$; A disk with radius r sweeps the area to cover sensors V and minimize the covering time. We refer such a problem as a minimized disk covering problem or TSPN (travel salesman problem with neighborhoods).

$$\min \{T\} = \min \{(L/V) + jt_g\} \quad (7.9)$$

If we consider the \vec{V} (moving vector of mobile collector) and t_g (data gathering time interval) as constants, we can minimize the time T by finding a shortest travel path (TSP problem) with minimum number of covering set. By dividing the gathering problem into two sub problems we can easily prove that the minimized disk covering problem is a NP-hard problem since one of its sub problems is a classic TSP problem which is NP-hard. When the locations of sensor are able to be obtained, the problem is a TSPN problem. If the locations are not available, the problem is a type of walking problem. For the TSPN problem, there are many works have been done to give an approximate optimized solution. Readers could refer to the works in [74] for more detail. By dividing the TSPN problem into two sub problems we give our approximate solution for the TSPN problem as following and analyze its performance boundary in the rest of this section.

Algorithm 9 Disk Covering Algorithm

1: **procedure** DISK COVERING ALGORITHM ▷

2: Determine the K

3: K -means clustering

4: TSP among the centroid of K clusters

5: **end procedure**

Performance Bounds of Minimum Covering Problem

From the equation 7.9 we know that the covering time of a walk is based on two parameters, the length of travel and the stops of gathering. Assuming that we can group all nodes into at least k clusters where $0 < K < n$. in order to gather the data in each

clusters we visit each cluster at their centroid points. For a K steps TSP in a unit square the upper bound gained by [75] is $(\sqrt{K/2} + 0.72) * 1.015$ where 0.72 is the boundary effect. Kershner [76] has proven that a honeycomb arrangement is the most efficient disk coverage configuration for the 1-cover and the lower bound of coverage density to 1-cover an square area is $2\pi/\sqrt{27}$. Since a hexagon arrangement could intuitively cover a square by place the disk along the perimeters. A loose upper bound for the number of disks to cover the entire space is $k = \frac{\bar{M}}{3\pi r^2}$.

Based on the lower bound of hexagon 1-covering, for a given M and radius r we have an lower bound of optimized number of cluster headers is $|C| < \frac{2\pi}{\sqrt{27}r^2}$. In other words, no matter how many sensors deployed in the area, we can use $\frac{2\pi M}{\sqrt{27}r^2}$ disks to cover it with a hexagon arrangement.

$$\frac{2\bar{M}}{\sqrt{27}r^2} \leq k \leq \frac{\bar{M}}{3\pi r^2} + \frac{x+y}{\sqrt{3}r} + 1 \quad (7.10)$$

In a hexagon arrangement, the distance between the center of two clusters is $1.73r$. Therefore, the upper bound for the minimum covering problem is

$$\frac{3.46\bar{M}}{\sqrt{27}\pi r v} = \frac{0.212\bar{M}}{r v} \leq t = \frac{k * 1.73r}{v} \leq \left(\frac{\bar{M}}{3\pi r^2} + \frac{x+y}{\sqrt{3}r} + 1 \right) \frac{1.73r}{v}. \quad (7.11)$$

For a hexagon arrangement with k hexagons, the lower-bound of a tour through all the center of cluster is a TSP of k points. Suppose the optimal TSP tour is known in a unit square with k points. Given a random point on the tour, we can divide the tour into two parts: the forward sub-path and the backward sub-path, both of which contains approximately $k/2$ points and we know the lower bound of a TSP is $\sqrt{k/2}$ and the low-bound of time to cover all the nodes is $t = \frac{\sqrt{k\bar{M}/2}}{v} + kt_g$.

Performance Bounds of Covering Time in Random Walk Problem

Random walk is the most intuitive and robust mobile data gathering solution since it need the least a priori knowledge of environment and resist to errors. We consider the upper bound of covering time of a random walk in a lattice. Assuming the communication

range of sensors is r . We divide the space into grids whose side length is $r/\sqrt{2}$. Therefore, any sensors in a grid could be reached by a MDC when a MDC enter the grid. Assuming the speed of MDC is $r/\sqrt{2}$. Therefore, each times the MDC will move to a neighboring cell. From the work of [77], we have the expecting time to visit all the cells in M is $t = \frac{N \ln^2 N}{\pi}$ where N is the number of grids $N \approx \frac{\overline{M}}{v^2} = \frac{2\overline{M}}{r^2}$.

Practically, the moving speed of an auto mobile in the city is limited around $1\text{meter}/s$ to $8\text{meter}/s$. Under such a circumstance, Comparing to upper bound of minimum covering problem, we have can see when $M < 6^{10}$ the covering time of TSPN will lower than the random walk.

7.3 A Localized Mobile Data Gathering Protocol Based on VFM

Based on the virtual force model, we present our mobile data gathering protocol (VFMDG). We consider an energy efficient data gathering application where sensors are deployed into the field to detect events and report the event to the sink periodically. In order to save the energy of sensors, some MDCs move around in the area to gather data from sensors and forward data to the sink. We assume that the energy of collectors is rechargeable. Therefore, the energy usage of collectors is not our concern.

In the VFMDG, each MDC plays a role of a cluster header. It requests sensors in the range to join its cluster and reports the size of their data. Based on the report, MDCs arrange a TDMA schedule and broadcast the schedule to each sensor in its cluster. The MDC then moves to the centroid of cluster and starts to collect data. As shown in the flow charts in Figure 7.2, a MDC starts a cluster configuration phase by sending a REQ message. The node that receives a REQ message will first check whether it has been claimed by other MDCs or if the sender is too far away to be reached. If the node is available to join the cluster, it will join the cluster and report to the sender the size of data in its queue. The cluster will be dismissed and rebuilt until the data gathering

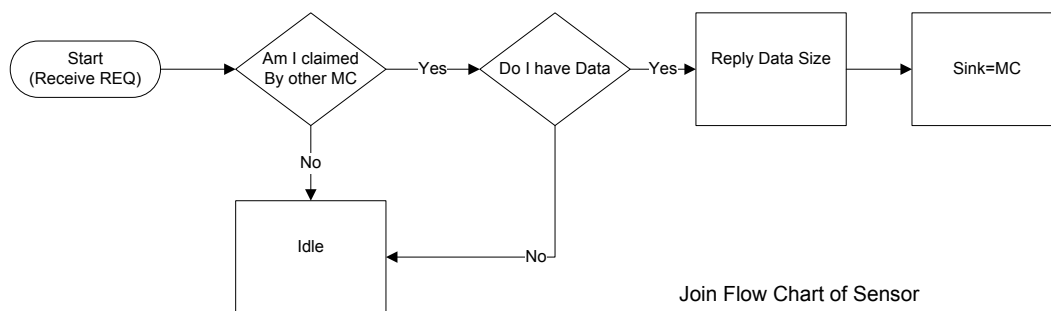
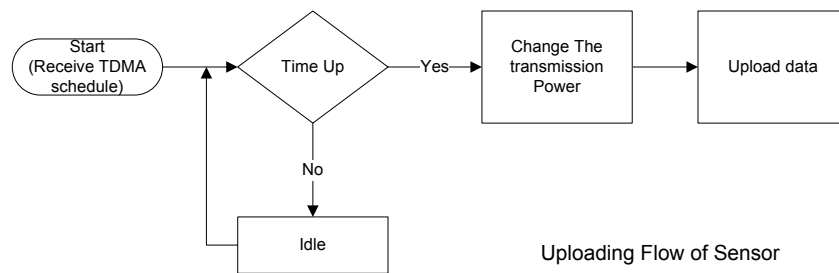
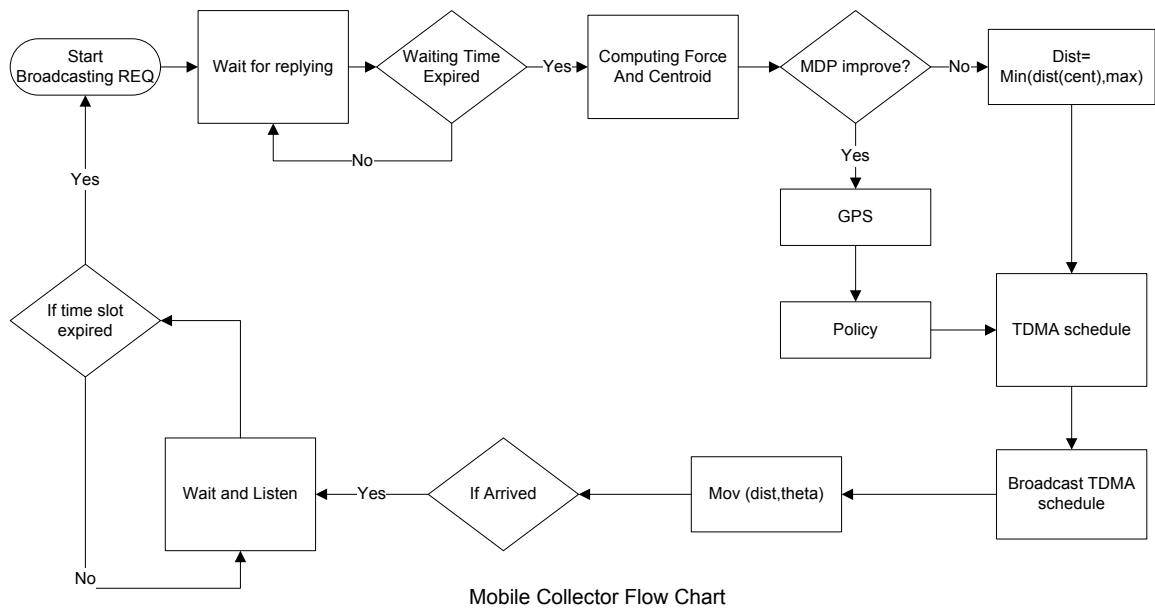


Figure 7.2: The flow charts of VFMDG.

phase is finished. After sensors reply to the REQ message with their size of data and average data freshness, the MDCs determine their moving direction and moving distance by computing the virtual net force. The centroid is computed as shown in equation (7.5). Assuming the moving speed is v and the moving time slot is t , the MDC moves to the

centroid if $dist(mdc, centroid) < vt$, otherwise it will move toward the centroid at speed v for t seconds.

7.3.1 Time Arrangement for Data Transmission

In order to avoid channel congestion, the entire data gathering process is divided into many time windows. There are three kinds of time windows: the control time window, the moving time window, and the transmission time window. In the moving time window, the MDC moves to the centroid. Since the distance from an MDC to the centroid is less than the communication range R , the size of a moving time window T_m is lower than R/v where v is the moving speed of the MDC.

Collectors gather data in the transmission time windows and exchange the control messages with sensors in the control time windows. Each transmission window lasts T_{tran} seconds following by a control time window which lasts $T_{control}$ seconds. The size of transmission and control windows is estimated by the size of packets, number of neighboring sensors, and the bitrate of the channel as follows,

$$\begin{aligned} T_{tran} &= \kappa_{tran} n_{neighbor} m_{data} / BW, \\ T_{control} &= \kappa_{control} n_{neighbor} m_{control} / BW, \end{aligned} \quad (7.12)$$

where κ_{tran} and $\kappa_{control}$ are parameters for size adjustment, m_{data} is the size of the data packet, $m_{control}$ is the size of the control packet, and BW is the bitrate of the channel. For example, when 150 sensors with an 80meter sensing range are deployed in a $160000m^2$ area, the transmission time window T_{tran} is 0.0367sec and the control time window $T_{control}$ is 0.00086sec since the $n_{neighbor}$ can be estimated by $150 * \pi R^2 / 160000m^2 = 18.8$. The transmission time window T_{tran} is divided into many time slots which are assigned to the sensors willing to exchange data with collectors. The size of the uploading time slot for each sensor is determined by its data as follows:

$$t_i = T_{tran} \frac{m_i}{\sum m_i}, \quad (7.13)$$

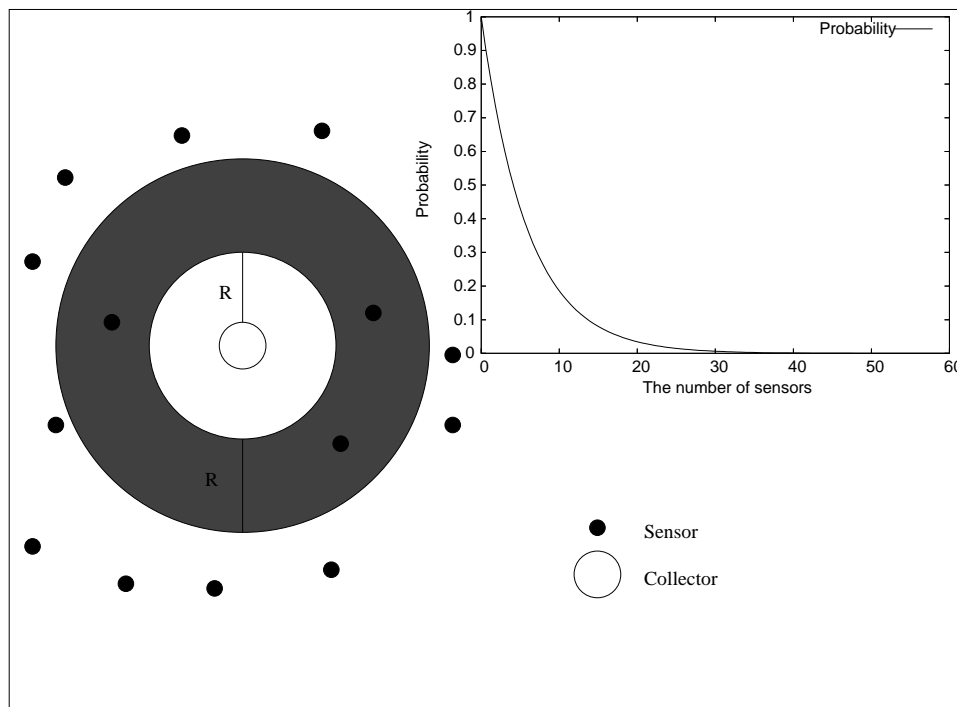


Figure 7.3: Trapped collector.

After the time arrangement is determined, the MDC will broadcast the arrangement before it moves to the centroid.

7.3.2 Trap and Hexagon Exploration

Since the VFMDG relies on the virtual forces of single-hop sensors, in a sparse deployment it has a “Trap” problem, which refers to the situation where MDC is kept in a spot for a long time by a set of sensors. “Trap” happens when there are no new nodes involved after sensors move to the new position. Since the moving distance d is always less than the communication range R , in other words, “Trap” happens when there are no sensors in the belt between πR^2 and $\pi(R + d)^2$ as shown in Figure 7.3. Assuming that the distribution of sensors X in the area follows normal distribution, the probability of $dist(X, collector) \leq r$ is proportional to the size of the area. The probability of sensors

falling in the gray belt of Figure 7.3 is

$$p = \left(1 - \frac{\pi R^2}{Area}\right) \frac{\pi(R+d)^2}{Area}. \quad (7.14)$$

The probability of disconnection $(1-p)^n$ is small enough to be ignored when the number of deployed sensors is high. For example, in a $160000m^2$ area, p is 0.1328 when the communication range R is $80m$ and the moving distance is $8m$; the probability approaches zero after 50 sensors are deployed into the area.

In order to move the MDC out of trap, we introduce the hexagon exploration. In each round, MDC computes the data freshness of gathered data. If the data freshness is lower than a predefined threshold and there is no new nodes being discovered, the collector is considered “trapped”. The MDC starts the hexagon exploration to get out of the “trap” by moving to the nearest hexagon coverage point on its original moving direction. If new sensors are discovered at the hexagon coverage point MDC stops the exploration and goes back to the virtual force model. Otherwise, the MDC keeps a hexagon walk on the space until meet new sensors. The reason we chose hexagon walk instead of random walk for an exploration is it has a lower covering time than the random walk.

7.3.3 Analysis of Multi-collector

In order to reduce the end-end latency and accelerate the data gathering process, multiple MDCs can be deployed to cover the entire area. The performance of a multi-mobile-collector scheme depends on the number of collectors in the field and their cooperation. Since multi-collectors will lead to a high cost for the mobile devices, in order to keep the cost in an acceptable level, we need to figure out the relationships among moving speed, waiting time, and the number of collectors for a given area.

We first consider the scenario when a sensor wants to send data there is at least one collector which could arrive at the sensor in time t . If the speed of the collector is V and the acceptable waiting time for data delivery is t , then we have an extended coverage range $r' = V * t + R$, where R is the transmission range of collectors. Assuming that the

distribution of the sensors X follows normal distribution, the probability that a sensor x falls in a collector's coverage range is proportional to the size of the sensing area. The probability function of $dist(x, c) < r'$ is:

$$F(r') = \begin{cases} 0, & r' < 0; \\ \frac{r'^2}{R_M^2}, & 0 \leq r' < R_M; \\ 1, & r' \geq R_M \end{cases}$$

where R_M is the radius of the area.

When m collectors are applied to the environment the probability that the data of the sensor could be gathered in time t is $p = 1 - (1 - f(r'))^m$. When $R_M = 100$ and $r' = 30$ we need at least 20 collectors to keep a high probability for collecting data efficiently. However, the question as to whether we need so many collectors to maintain the service still need to be answered.

A possible energy and time efficient solution is that for each sensor, there is a collector in its range at all times. In other words, no matter what the deployment and the number of sensors are, an efficient $1 - cover$ collector deployment will give the best performance for both energy and latency concerns. Based on the result of Kershner [76], a honeycomb arrangement is the most efficient $1 - cover$ disk arrangement whose density can achieve $2\pi/\sqrt{27}$. We can derive the number of collectors for 1-covering a given square area by $Limit(n\pi r^2) \rightarrow \frac{2\pi}{\sqrt{27}}\bar{M}$ where \bar{M} is the size of monitored area. Based on the above function, the upper bound of m for a sparse distribution of sensors could be derived through a search for the polygons enclosing all the sensors.

7.4 Experiment and Discussions

In order to evaluate the performance of VFMDG, we carried out a set of simulation experiments. We first introduce our simulation parameters and then discuss the experiment results.

7.4.1 The Simulation Environment

We consider the sensors with $914MHz$ Lucent WaveLAN DSSS radio interface and adapt the radio parameters used in the NS2 as shown in Table 7.1. We assume that the reception threshold of the radio r_{thresh} is $-64dbm$ or $0.36nw$, and the maximum communication range is 40 meters. We can therefore derive that the transmitting power P_t should be less than $0.0034J$ for a $1Mbps$ channel. Since we assume sensors have the ability to adjust the transmitting power, sensors recalculate P_t for different transmission targets. In order to maintain the connectivity and coverage of the network, 150 sensors are randomly deployed in a $400m \times 400m$ square area. Sensors have 50% probability to generate a data packet every second. The sizes of data packets and control packet are $1K$ bits and $32bits$, respectively. Considering the mobility limitation of a vehicle, the speed of our MDC is set at $8meters/sec$.

Table 7.1: Radio Parameters

Parameter	Value	Description
$d_{crossover}$	$87m$	Cross-over Distance
Range	$80m$	Communication Range
Bitrate	$1Mbps$	Radio Bitrate
G_t, G_r	1	Antenna gain factor
h_t, h_r	$1.5m$	Antenna height above the ground
λ	$0.325m$	signal wavelength
$E_{frissamp}$	$0.51pJ/bit/m^2$	$\frac{r_{thresh}(4\pi)^2}{R_b G_t G_r \lambda^2}$
P_t	$\leq 0.0034j$	Transmission Energy, $e_{friss} R_b d^2$
r_{thresh}	$0.36nw(-64dbm)$	The receiving power threshold
Simulation Parameters		
M	$400m*400m$	The square monitoring space
N	150	Number of sensors
V	8 m/sec	Speed of mobile collector
DGR	$1packet/sec$	Data generation rate
CPS	$32 - 1kbits$	Control packet size
DPS	$1kbits$	Control packet size
Queue	1M bytes	The size of sensor's cache

7.4.2 Experimental Results

We first compared the performance of VFMDG with single mobile collector to the biased random walk solution (B-RW)[61] and the hexagon walk. The biased random walk solution refers to the protocols where MDCs move randomly in the fields to gather data from the sensors in the single-hop neighborhood. In the B-RW, MDC increase the probability of directions when it meets a new node. The length of each moving step of B-RW is $r/\sqrt{2}$. Since hexagon walk gives an upper bound of covering time in a bounded rectangle area, we use HW solution as a base for the comparison. The factors considered in the performance comparison of protocols are data freshness, coverage, energy efficiency, gathering rate, and travel length.

We then compared the performance of VFMDG with multiple mobile collectors with MULE type random waypoint walking solution (RW), shortest path tree (SPT) based network relaying solution. SPT is a well adapted structure for routing protocols and is used in our comparison as an example of data relay solutions. Through cooperating with SPT, we implement two hybrid solutions for RW and VFMDG and compare their performance in terms of report latency. The factors considered in the performance comparison of protocols are energy efficiency, energy distribution, gathering rate, latency, coverage and cost.

Data Freshness

Data freshness reflects the fairness and efficiency of data gathering. By covering more sensors in each second and using hexagon walk exploration to discover new data sources, VFMDG not only gather more data than the RW and B-RW solutions but could also has better data freshness.

We compared the average data freshness of all data sources in the Figure 7.4 and 7.5. For the single MDC scenario, after 200 simulation seconds, data in random walk have to wait for 78 simulation seconds until they are gathered by MDC. Both VFMDG and HW show a better performance than RW. Since VFMDG can reach the connected nodes

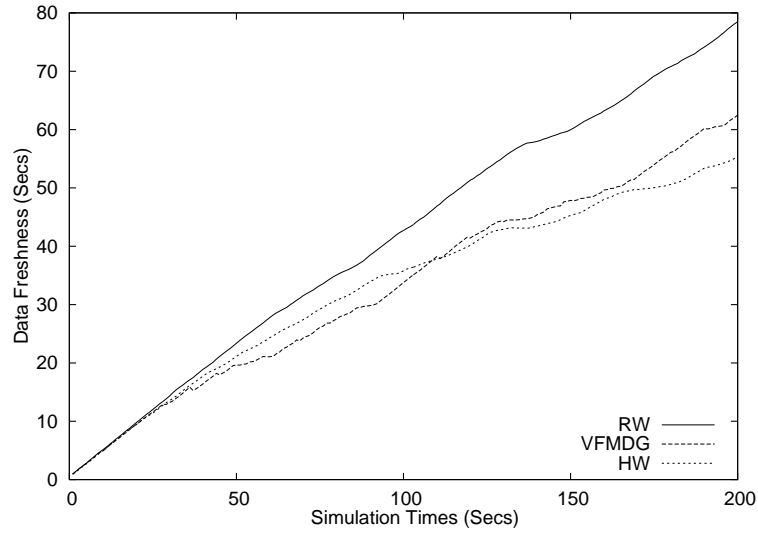


Figure 7.4: Data Freshness with Single MDC.

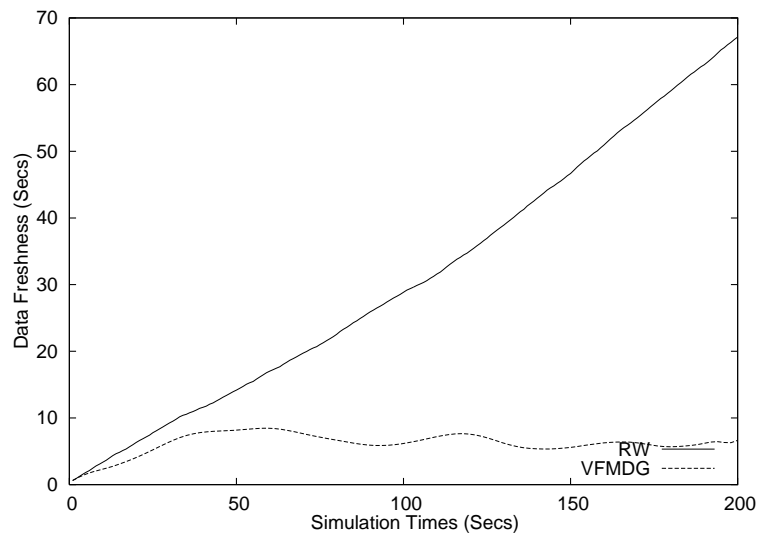


Figure 7.5: Data Freshness with 4 MDCs.

without exploration, in the initial phase VFMDG even show a better performance than HW. After that, the HW becomes the best because of its lowest covering time for the space.

In order to check the performance of algorithms with multiple MDC, in Figure 7.5, we increase the number of collectors from 1 to 4. The performance of RW improves from

78 seconds to 67 seconds. However, the performance of VFMDG highly improves from 68 seconds to 8 seconds. The interesting discovery is that the data freshness of VFMDG becomes stable at the level of 6 – 7 seconds only 36 seconds after the algorithm started.

Mobile Coverage

Coverage refers to how well the sensors can be accessed by the MDCs. It is one of the key reasons that the performance of VFMDG is better than the other mobile data gathering algorithms. In Figure 7.6, we evaluated the coverage of VFMDG, RW and

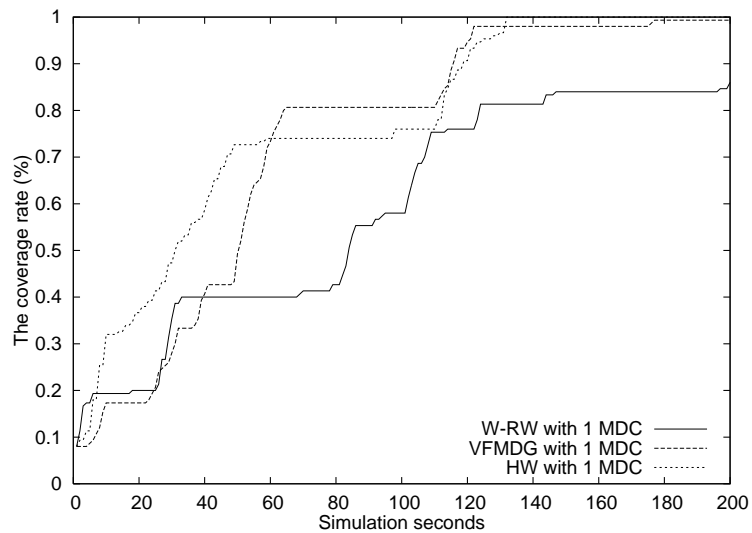


Figure 7.6: Coverage of single MDC.

HW with single MDC. Based on the Equation 7.11, for the hexagon walk a MDC can cover the $400 * 400m^2$ space by traveling not more than 1662 meter in 208 steps. In the simulation, it takes 127 simulation seconds to cover all the nodes in the area. By exploring space using hexagon walk, VFMDG has the second best performance in terms of covering time. Since B-RW moving back and forward in the space, it has a worst covering performance. When the deployment density becomes high, the covering time of VFMDG became longer than the low density scenario. There are two reasons. The first is that the virtual force in the VFMDG is a trade-off between gathering rate and

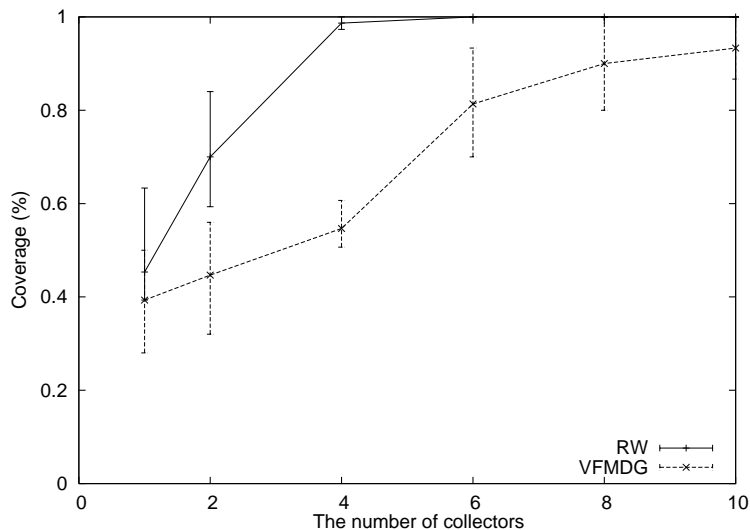


Figure 7.7: Area coverage of multiple-MDC after 200 simulation seconds.

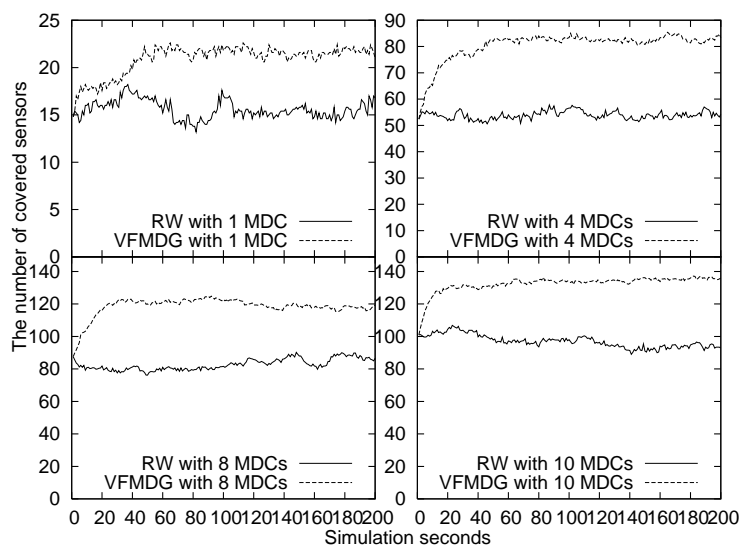


Figure 7.8: Node coverage.

data freshness. With an energy efficiency first setup (small θ), MDC moves less when there is enough data supplement around it. The second reason is that, the distance from current location of MDC to the centroid become short. It makes MDC move less at each step. By increasing the weight of data freshness in the virtual force we can resolve such a problem.

Deploying more MDCs is another way to reduce the covering time. Using different numbers of collectors we show the changes of coverage in the Figure 7.7. Since in the VFMDG the MDCs keep distance from each other by claiming their territory, the covering time of VFMDG has a major improvement. However, since in B-RW, MDCs have more overlap in their territory, the improvement of B-RW is lower than VFMDG. Figure 7.9 shows how the coverage changes in RW and VFMDG with 6 collectors. The disks in the figure represent the communication range of collectors and the lines of different colors represent the moving trajectory of each collectors after 100 moves.

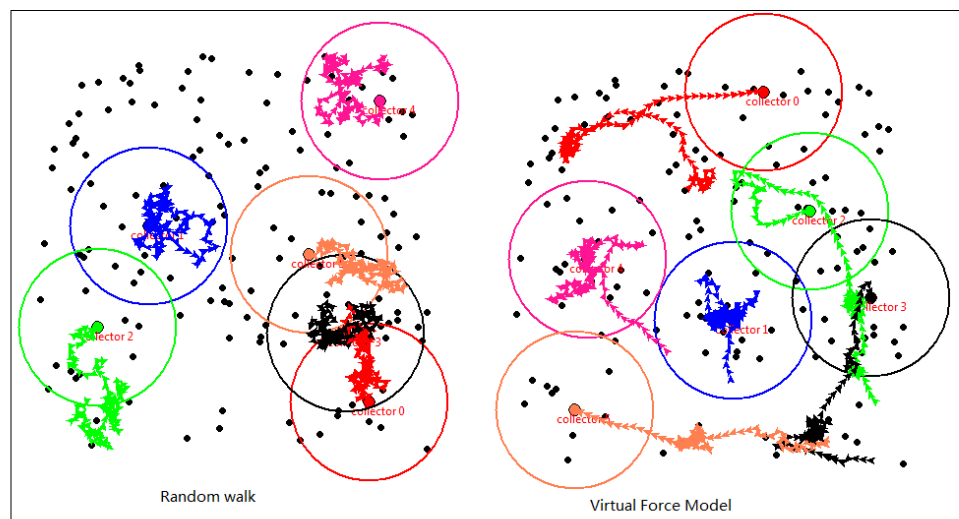


Figure 7.9: The movement trajectory.

We varied the number of collectors from 1 to 10 and evaluated the coverage per second in Figure 7.8. In the single collector variation, after 50 seconds, the collector of the VFMDG covers about 21 sensors out of 150 sensors for each second, while the collector of RW covers about 15 sensors. With an increase in the number of collectors, the coverage per second of both RW and VFMDG increases. However, in the 200 second simulation time, the RW solution simply keeps the initial coverage rate while VFMDG covers 30 to 40 more sensors than the initial stage.

In the Figure 7.10, we investigate the number of visited sensors at different simulation time. We notice that when a single collector is applied, the collector in VFMDG visits

more sensors than the collectors in RW before 100 seconds. However, after 100 seconds, RW regains the advantage. The reason for this phenomenon is that collectors in VFMDG move to other areas until they have larger forces than in the current area. When the current area still supplies sufficient forces, the collector in the VFMDG protocol will stay in the same place to gather the data.

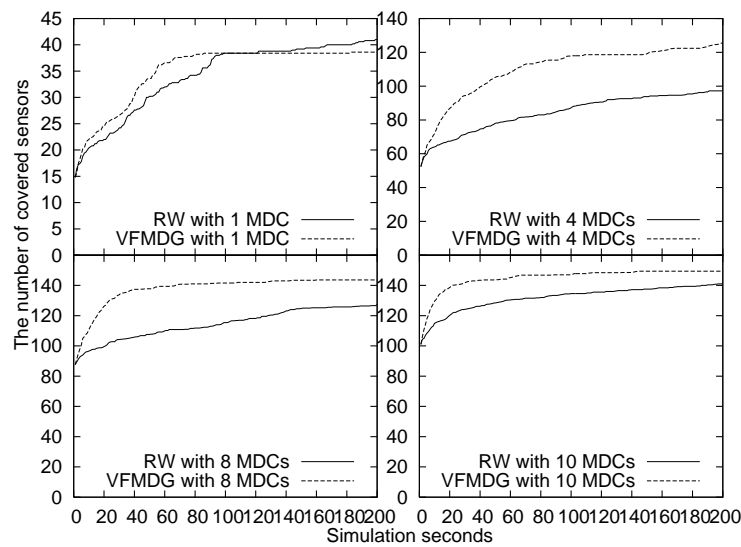


Figure 7.10: The number of visited sensors.

Energy Distribution and Energy Consumption

In a traditional data relay approach, the hot-spot is unavoidable because of the static sink. One benefit of VFMDG is that the transmission hot-spot is released since the MDCs roam in the field. In a network with 150 sensors and 10 MDCs, we measured the energy consumption of each sensor after 200 simulation seconds. The 3d energy dissipation graphs are shown in Figure 7.11 where the average energy dissipation of a SPT solution is compared to the VFMDG protocol with 10 collectors. It can be observed that in the SPT approach the sensors close to the sink dissipate energy faster than others due to message relaying. Both VFMDG and the RW models show a better ability than SPT to balance the energy consumption. The reason behind this result is that after 200

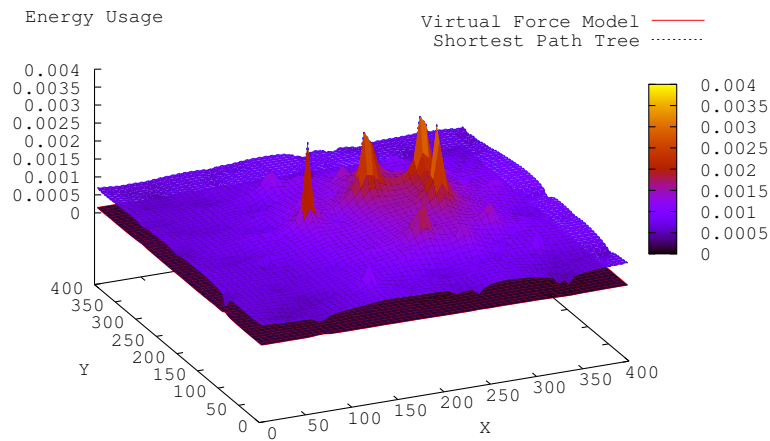


Figure 7.11: Energy distribution of VFMDG and SPT.

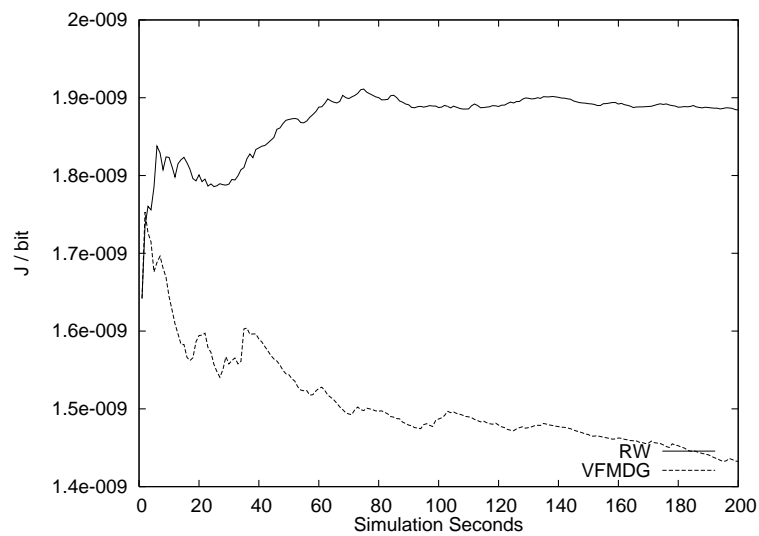


Figure 7.12: The average energy usage for single MDC using gathering-rate first VFM.

seconds MDCs can cover almost 96% of the entire space.

By approaching the centroid of a set of nodes, VFMDG save the energy of data gathering. In Figure 7.12, we compare the average energy usage for gathering a bit in both VFMDG and RW with single-MDC and 150 sensors. Initially, both VFMDG and B-RW consume energy in the same level. However, since MDC always try to minimize

the energy usage, after 200 seconds, VFMDG only uses $1.432e - 9J$ for gathering a bit of data while the RW solution uses $1.908e - 9J$.

Travel Length

Since we need power to drive the mobile vehicles, we consider the energy usage for moving the vehicles as the cost of the proposed protocol. If we assume the weight of the mobile vehicle is a constant, we can measure the energy usage by the moving distance of MDCs. In the Figure 7.13, we show the total travel distance of VFMDG, RW and HW with single-MDC. After 200 simulation seconds, the moving distance of VFMDG and B-RW are 2610 meters and 2476 meters respectively. Since VFMDG does hexagon exploration to get out of trap, it travels 5% percent more than B-RW. However, VFMDG covered over 98% of nodes and RW only cover 86%.

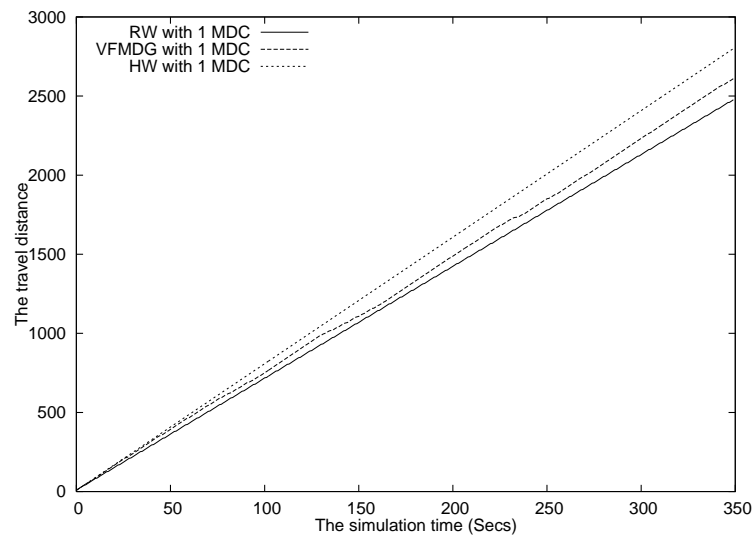


Figure 7.13: Travel length with single MDC.

7.5 Conclusion

In this work, we presented a virtual force model for energy efficient mobile data gathering in sensor networks. The data in the queue of sensors were simulated as elastic forces to attract collectors moving towards the optimal positions. Based on the minimum total potential energy theory we proved that our virtual force model can always minimize the energy usage of data gathering by moving the collector to the centroid. We investigate the covering time of mobile data gathering and provide an upper bound of covering time using hexagon walk. We then presented an autonomous self-adaptive mobile data gathering protocol (VFMDG) using our virtual force model and hexagon walk exploration. A multiple MDC management scheme was also proposed to maintain the quality of data gathering by using a claim mechanism to keep the collectors away from each other. We compared the VFMDG with other data gathering protocols and showed that in addition to the energy efficiency, VFMDG is a lightweight, autonomous, localized, and easy to implement mobile data gathering protocol. The performance of VFMDG was evaluated and compared with the classic shortest path tree based network relaying data gathering, random walk, biased random walk and hexagon walk in terms of throughput, energy consumption, energy distribution, data freshness, and cost. The simulation results showed that compared to the random waypoint, VFMDG saves 30% in energy usage, improves 10% throughput, and increases the coverage rate by 25%. In the hybrid solution, the VFMDG protocol is 70% more efficient than the random waypoint solution in terms of average report delay. Considering the low requirements of network topology and geometry information, and high autonomous characteristics, the VFMDG is a novel localized protocol for single-hop mobile data gathering to prove the performance in terms of data freshness, energy, through output, and cost.

Chapter 8

Conclusion and Future Work

In this dissertation, the problems associated with the coverage of a wireless sensor network and their impacts on the energy consumption are addressed. In the following, I highlight the contributions of the research presented in this thesis and present our future work.

8.1 Our Contributions

Our contributions include three parts.

1. Irregular sensing model

The irregular sensing model of coverage problem in a two dimensional space is proposed and compared with the classic boolean model and probability model. In order to determine the coverage quality of irregular sensing model, revised weighted-alpha-shape algorithms are developed. The numerical results in the thesis show that the irregular sensing model can describe the sensing range more accurate than the original models.

2. Optimal Cover Scheduling for Random Deployments of WSNs

Many practical applications of WSNs do not allow the manual deployment of the

sensor nodes at optimum locations. The sensor nodes in these applications are randomly distributed, for example, dropped from an airplane. In such cases, the problem of determining the coverage and selecting a minimum subset of sensor nodes for complete coverage is important for the proper functioning of the network. In this thesis, we first adapt the genetic algorithm and Markov decision process to determine the achievable optimal results for static and dynamic wireless sensor networks respectively. We then proposed a localized coverage-preserving scheduling algorithm based on our irregular sensing model.

3. Mobile Coverage and Mobile Data Gathering

We extend the research of coverage to the mobile network where the mobile element move in the 2D plane to efficiently cover the entire space or some specific spots. A virtual force model based on the energy consumption of data gathering is proposed to locally guide the mobile elements to move in the field. We investigate the coverage efficiency of random walk, weighted random walk and hexagon walk in a square area and show that the hexagon walk is more efficient than the random walk and could also be used to improve the performance of random walk. Combining the virtual force model and hexagon walk we developed our single-hop mobile data gathering algorithm (VFMDG). Considering the low requirements of VFMDG in the network topology and geometry information, and its high autonomous characteristics, the VFMDG could replace the random walk model in most scenarios to improve the performance of data gathering application in terms of energy, throughput, report delay, and cost.

8.2 Future Work

In our future work, we plan to extend our research from the following four aspects.

1. Topology Control in Sensor Networks

Topology control is the process of controlling the topology of a wireless network by

adjusting the coverage ranges of its wireless nodes. Using rigorous combinatorial and probabilistic analysis, a future goal is to study various topological properties (e.g., connectivity, routing path, coverage degree, local minimum for geographical routing, etc.), and present several variations of the topology control method. The findings should show balance between the various aspects of network performance.

2. Coverage Problem in Vehicular Network

In recent years, vehicular network has attracted research attention. The arrangement of access point or roadside unit is a key challenge of Vehicular Network. However, the mobility of the network elements makes the vehicular network different from the classic wireless sensor network. The protocols to maintain the connectivity of a mobile network pose challenges from the view of mobile coverage.

3. Object Tracking with Dynamic Sensor Nodes

Most current researches assume that the sensor nodes are static and are not mobile. The idea of a mobile sensor network for tracking has not matured yet but applications such as ZebraNet indicate that such a network might exist or be needed in the future. A mobile sensor network for tracking will require new algorithms for communication, collaborative signal processing, and tracking.

4. Data Mining in Smart Sensor Networks

Sensors streaming their data online are turning sensor network as a huge data source. Software platforms that integrate and mine these data streams may create a world in which sensors become pixels and we browse reality as easily as we browse Web pages today. On the one hand, the coverage could be an important performance metric of data Mining in wireless sensor networks. On the other hand, by mining the retrieved data we can determine and model the coverage changes of entire network and further improve the performance of network.

List of Publications

The following listed are the publications related to this thesis.

Submitted Journal Papers

1. Xin Fei, Azzedine Boukerche, Richard Yu: Self-Adaptive Data Gathering Protocol for Wireless Sensor Networks with Mobile Collectors (second round revision)

Published Journal Papers

1. bibliographical record in XML Azzedine Boukerche, Xin Fei: A coverage-preserving scheme for wireless sensor network with irregular sensing range. *Ad Hoc Networks* 5(8): 1303-1316 (2007)

Published Conference Papers

1. Xin Fei, Azzedine Boukerche, Richard Yu: An efficient Markov decision process based mobile data gathering protocol for wireless sensor networks. *IEEE Wireless Communications and Networking Conference 2011*: 1032-1037
2. Xin Fei, Azzedine Boukerche, Richard Yu: A POMDP Based K-Coverage Dynamic Scheduling Protocol for Wireless Sensor Networks. *IEEE Global Communications Conference 2010*: 1-5
3. Xin Fei, Samer Samarah, Azzedine Boukerche: A bio-inspired coverage-aware scheduling scheme for wireless sensor networks. *IEEE International Parallel & Distributed Processing Symposium Workshops 2010*: 1-8
4. Xin Fei, Azzedine Boukerche, Regina Borges de Araujo: Irregular Sensing Range Detection Model for Coverage Based Protocols in Wireless Sensor Networks. *IEEE Global Communications Conference 2009*: 1-6
5. Azzedine Boukerche, Jing Feng, Xin Fei: V-Square: An Accurate Time Synchronization Protocol for Wireless Video Sensor Networks. *IEEE Global Communications Conference 2008*: 5096-5100

6. Xin Fei, Azzedine Boukerche, Jing Feng: Low Information Redundancy Based Node Partition Protocols for Wireless Sensor Networks. IEEE Global Communications Conference 2008: 5480-5484
7. Xin Fei, Azzedine Boukerche: A performance evaluation of a coverage compensation based algorithm for wireless sensor networks. ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems 2008: 109-116
8. Azzedine Boukerche, Xin Fei: Adaptive Data-Gathering Protocols with Mobile Collectors for Vehicular Ad Hoc and Sensor Networks. IEEE International Conference on Wireless and Mobile Computing, Networking and Communications 2008: 7-12
9. Azzedine Boukerche, Xin Fei: A Voronoi Approach for Coverage Protocols in Wireless Sensor Networks. IEEE Global Communications Conference 2007: 5190-5194
10. Azzedine Boukerche, Xin Fei, Regina Borges de Araujo: An Energy Saving and Coverage Preserving Protocol based on Function Layers For Wireless Sensor Networks. IEEE Conference on Local Computer Networks 2007: 711-718

Bibliography

- [1] Chi-Fu Huang and Yu-Chee Tseng. The coverage problem in a wireless sensor network. In *WSNA '03: Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications*, pages 115–121, New York, NY, USA, 2003. ACM.
- [2] Joseph O'Rourke. *Art gallery theorems and algorithms*. Oxford University Press, Inc., New York, NY, USA, 1987.
- [3] Yuh-Ren Tsai, Tien-Yu Lin, and Kai-Jie Yang. Sensing coverage for randomly distributed wireless sensor networks in shadowed environments. In *SUTC (1)*, pages 288–291, 2006.
- [4] D W Gage. Command control for many-robot systems. *Control*, 10(June):28–34, 1992.
- [5] Steve and Fisk. A short proof of chvtal's watchman theorem. *Journal of Combinatorial Theory, Series B*, 24(3):374 –, 1978.
- [6] K. Kar and S. Banerjee. Node placement for connected coverage in sensor networks. 2003.
- [7] K. Chakrabarty, S.S. Iyengar, Hairong Qi, and Eungchun Cho. Grid coverage for surveillance and target location in distributed sensor networks. *Computers, IEEE Transactions on*, 51(12):1448 – 1453, dec 2002.

- [8] Santosh Kumar, Ten H. Lai, and Anish Arora. Barrier coverage with wireless sensors. In *Proceedings of the 11th annual international conference on Mobile computing and networking*, MobiCom '05, pages 284–298, New York, NY, USA, 2005. ACM.
- [9] Bang Wang, Wei Wang, V. Srinivasan, and Kee Chaing Chua. Information coverage for wireless sensor networks. *Communications Letters, IEEE*, 9(11):967 – 969, nov. 2005.
- [10] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M.B. Srivastava. Coverage problems in wireless ad-hoc sensor networks. In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1380 –1387 vol.3, 2001.
- [11] Xiang-Yang Li, Peng-Jun Wan, and O. Frieder. Coverage in wireless ad hoc sensor networks. *Computers, IEEE Transactions on*, 52(6):753 – 763, june 2003.
- [12] B. Liu and D. Towsley. On the coverage and detectability of wireless sensor networks. In *Proc. of WiOpt*, 2003.
- [13] Seapahn Meguerdichian, Farinaz Koushanfar, Gang Qu, and Miodrag Potkonjak. Exposure in wireless ad-hoc sensor networks. In *MOBICOM*, pages 139–150, 2001.
- [14] S. Adlakha and M. Srivastava. Critical density thresholds for coverage in wireless sensor networks. In *Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE*, volume 3, pages 1615 –1620 vol.3, march 2003.
- [15] Yong Gao, Kui Wu, and Fulu Li. Analysis on the redundancy of wireless sensor networks. In *Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications*, WSNA '03, pages 108–114, New York, NY, USA, 2003. ACM.

- [16] Azzedine Boukerche, Xin Fei, and Regina Borges de Araujo. An energy aware coverage-preserving scheme for wireless sensor networks. In *PE-WASUN*, pages 205–213, 2005.
- [17] Santosh Kumar, Ten H. Lai, and József Balogh. On k-coverage in a mostly sleeping sensor network. In *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 144–158, New York, NY, USA, 2004. ACM.
- [18] A. Ghosh. Estimating coverage holes and enhancing coverage in mixed sensor networks. In *Local Computer Networks, 2004. 29th Annual IEEE International Conference on*, pages 68 – 76, nov. 2004.
- [19] Andrew Howard, Maja J. Mataric, and Gaurav S. Sukhatme. An incremental self-deployment algorithm for mobile sensor networks. *Auton. Robots*, 13(2):113–126, 2002.
- [20] Yi Zou and Krishnendu. Chakrabarty. Sensor deployment and target localization in distributed sensor networks. 3(1):61–91, 2004.
- [21] S. Poduri and G.S. Sukhatme. Constrained coverage for mobile sensor networks. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 1, pages 165 – 171, apr. 2004.
- [22] Andrew Howard, Maja J Mataric, and Gaurav S Sukhatme. Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem. In *Distrib Auton Robot Syst 5*, pages 299–308, 2002.
- [23] Di Tian and Nicolas D. Georganas. A coverage-preserving node scheduling scheme for large wireless sensor networks. In *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 32–41, New York, NY, USA, 2002. ACM.

- [24] S. Lu L. Zhang F. Ye, G. Zhong. Energy efficient robust sensing coverage in large sensor networks. *UCLA Technical Report 2002*.
- [25] Mihaela Cardei and Ding-Zhu Du. Improving wireless sensor network lifetime through power aware organization. 11(3):333–340, 2005.
- [26] S. Slijepcevic and M. Potkonjak. Power efficient organization of wireless sensor networks. In *Communications, 2001. ICC 2001. IEEE International Conference on*, volume 2, pages 472 –476 vol.2, 2001.
- [27] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *HICSS*, 2000.
- [28] Azzedine Boukerche, Xin Fei, and Regina Borges de Araujo. An optimal coverage-preserving scheme for wireless sensor networks based on local information exchange. *Computer Communications*, 30(14-15):2708–2720, 2007.
- [29] Jonathan Hui, Zhiyuan Ren, and Bruce H. Krogh. Sentry-based power management in wireless sensor networks. In *IPSN*, pages 458–472, 2003.
- [30] J.M. Rabaey, J. Ammer, T. Karalar, Suetfei Li, B. Otis, M. Sheets, and T. Tuan. Picoradios for wireless sensor networks: the next challenge in ultra-low power design. In *Solid-State Circuits Conference, 2002. Digest of Technical Papers. ISSCC. 2002 IEEE International*, volume 1, pages 200 –201 vol.1, 2002.
- [31] Lin Gu and J.A. Stankovic. Radio-triggered wake-up capability for sensor networks. In *Real-Time and Embedded Technology and Applications Symposium, 2004. Proceedings. RTAS 2004. 10th IEEE*, pages 27 – 36, may 2004.
- [32] A. Cerpa and D. Estrin. Ascent: adaptive self-configuring sensor networks topologies. *Mobile Computing, IEEE Transactions on*, 3(3):272 – 285, july-aug. 2004.

- [33] Benjie Chen, Kyle Jamieson, Hari Balakrishnan, and Robert Morris. Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. *Wirel. Netw.*, 8:481–494, September 2002.
- [34] Ya Xu, John Heidemann, and Deborah Estrin. Geography-informed energy conservation for ad hoc routing. In *Proceedings of the 7th annual international conference on Mobile computing and networking*, MobiCom '01, pages 70–84, New York, NY, USA, 2001. ACM.
- [35] Honghai Zhang and Jennifer C. Hou. Maintaining sensing coverage and connectivity in large sensor networks. *Ad Hoc & Sensor Wireless Networks*, 1(1-2), 2005.
- [36] Xiaorui Wang, Guoliang Xing, Yuanfang Zhang, Chenyang Lu, Robert Pless, and Christopher Gill. Integrated coverage and connectivity configuration in wireless sensor networks. In *Proceedings of the 1st international conference on Embedded networked sensor systems*, SenSys '03, pages 28–39, New York, NY, USA, 2003. ACM.
- [37] Fan Ye, G. Zhong, J. Cheng, Songwu Lu, and Lixia Zhang. Peas: a robust energy conserving protocol for long-lived sensor networks. In *Distributed Computing Systems, 2003. Proceedings. 23rd International Conference on*, pages 28 – 37, may 2003.
- [38] Ting Yan, Tian He, and John A. Stankovic. Differentiated surveillance for sensor networks. In *Proceedings of the 1st international conference on Embedded networked sensor systems*, SenSys '03, pages 51–62, New York, NY, USA, 2003. ACM.
- [39] Jean Carle and David Simplot. Energy-efficient area monitoring for sensor networks. *IEEE Computer*, 37(2):40–46, 2004.

- [40] Martin David Adams. *Sensor Modelling, Design and Data Processing for Autonomous Navigation in Confined Environments*. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 1998.
- [41] H. Edelsbrunner, D. Kirkpatrick, and R. Seidel. On the shape of a set of points in the plane. 29(4):551 – 559, jul 1983.
- [42] D.B. Jourdan and O.L. de Weck. Layout optimization for a wireless sensor network using a multi-objective genetic algorithm. In *Vehicular Technology Conference, 2004. VTC 2004-Spring. 2004 IEEE 59th*, volume 5, pages 2466 – 2470 Vol.5, 2004.
- [43] Mansoor Davoodi, Ali Mohades, and Jafar Rezaei. A genetic algorithm for the constrained coverage problem. In Jörn Mehnen, Mario Kappen, Ashraf Saad, and Ashutosh Tiwari, editors, *Applications of Soft Computing*, volume 58 of *Advances in Intelligent and Soft Computing*, pages 347–356. Springer Berlin / Heidelberg, 2009.
- [44] Jie Jia, Jian Chen, Guiran Chang, and Jie Li. Find the maximum k-disjoint coverage sets in wsn using genetic algorithm. *International Journal of Modelling, Identification and Control*, 9(1-2):43–52, 2010.
- [45] Nejla Essaddi, Mohamed Hamdi, Sami Habib, and Nouredine Boudriga. Optimizing area coverage in wireless sensor networks. In *AICCSA*, pages 1–6, 2010.
- [46] Azzedine Boukerche and Xin Fei. A coverage-preserving scheme for wireless sensor network with irregular sensing range. *Ad Hoc Networks*, 5(8):1303–1316, 2007.
- [47] J. C. Gittins. Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society. Series B (Methodological)*, 41(2):pp. 148–177, 1979.
- [48] P. Whittle. Restless bandits: Activity allocation in a changing world. *Journal of Applied Probability*, 25:pp. 287–298, 1988.
- [49] P. Whittle. Multi-armed bandits and the gittins index. *Journal of the Royal Statistical Society. Series B (Methodological)*, 42(2):pp. 143–149, 1980.

- [50] Dimitris Bertsimas and José Niño Mora. Restless bandits, linear programming relaxations, and a primal-dual index heuristic. *Oper. Res.*, 48:80–90, January 2000.
- [51] Peng Hu, Zude Zhou, Quan Liu, and Fangmin Li. The hmm-based modeling for the energy level prediction in wireless sensor networks. In *Industrial Electronics and Applications, 2007. ICIEA 2007. 2nd IEEE Conference on*, pages 2253–2258, may 2007.
- [52] D. England, Bharadwaj Veeravalli, and J.B. Weissman. A robust spanning tree topology for data collection and dissemination in distributed environments. *Parallel and Distributed Systems, IEEE Transactions on*, 18(5):608–620, may 2007.
- [53] Kamal Jain, Jitendra Padhye, Venkata N. Padmanabhan, and Lili Qiu. Impact of interference on multi-hop wireless network performance. *Wireless Networks*, 11(4):471–487, 2005.
- [54] Enrique J. Duarte-Melo and Mingyan Liu. Data-gathering wireless sensor networks: organization and capacity. *Comput. Netw.*, 43:519–537, November 2003.
- [55] Hesham El Gamal. On the scaling laws of dense wireless sensor networks: the data gathering channel. *IEEE Transactions on Information Theory*, 51(3):1229–1234, 2005.
- [56] Haeyong Kim, Yongho Seok, Nakjung Choi, Yanghee Choi, and Taekyoung Kwon. Optimal multi-sink positioning and energy-efficient routing in wireless sensor networks. In *ICOIN*, pages 264–274, 2005.
- [57] Javid Taheri and Albert Y. Zomaya. A genetic algorithm for finding optimal location area configurations for mobility management. In *LCN*, pages 568–577, 2005.
- [58] Lang Tong, Qing Zhao, and Srihari Adireddy. Sensor networks with mobile agents. In *Proceedings of the 2003 IEEE conference on Military communications - Volume I, MILCOM'03*, pages 688–693, Washington, DC, USA, 2003. IEEE Computer Society.

- [59] Sushant Jain, Rahul C. Shah, Waylon Brunette, Gaetano Borriello, and Sumit Roy. Exploiting mobility for energy efficient data collection in wireless sensor networks. *Mob. Netw. Appl.*, 11:327–339, June 2006.
- [60] Arnab Chakrabarti, Ashutosh Sabharwal, and Behnaam Aazhang. Using predictable observer mobility for power efficient design of sensor networks. In *Proceedings of the 2nd international conference on Information processing in sensor networks*, IPSN'03, pages 129–145, Berlin, Heidelberg, 2003. Springer-Verlag.
- [61] Constantinos-Marios Angelopoulos, Sotiris Nikolettseas, Dimitra Patroumpa, and Jose Rolim. Coverage-adaptive random walks for fast sensory data collection. In *Proceedings of the 9th international conference on Ad-hoc, mobile and wireless networks*, ADHOC-NOW'10, pages 81–94, Berlin, Heidelberg, 2010. Springer-Verlag.
- [62] S.R. Gandham, M. Dawande, R. Prakash, and S. Venkatesan. Energy efficient schemes for wireless sensor networks with multiple mobile base stations. In *Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE*, volume 1, pages 377 – 381 Vol.1, dec. 2003.
- [63] Stefano Basagni, Alessio Carosi, Chiara Petrioli, and Cynthia A. Phillips. Moving multiple sinks through wireless sensor networks for lifetime maximization. In *MASS*, pages 523–526, 2008.
- [64] Jun Luo and Jean-Pierre Hubaux. Joint mobility and routing for lifetime elongation in wireless sensor networks. In *INFOCOM*, pages 1735–1746, 2005.
- [65] Kwang il Hwang and Doo Seop Eom. Adaptive sink mobility management scheme for wireless sensor networks. In *UIC*, pages 478–487, 2006.
- [66] Hua Yang, Fengji Ye, and Biplab Sikdar. Simple: Using swarm intelligence methodology to design data acquisition protocol in sensor networks with mobile sinks. In *INFOCOM*, 2006.

- [67] Jun Luo and Jean-Pierre Hubaux. Joint sink mobility and routing to maximize the lifetime of wireless sensor networks: The case of constrained mobility. *IEEE/ACM Trans. Netw.*, 18(3):871–884, 2010.
- [68] Guoliang Xing, Tian Wang, Weijia Jia, and Minming Li. Rendezvous design algorithms for wireless sensor networks with a mobile base station. In *MobiHoc*, pages 231–240, 2008.
- [69] Wei Wang 0002, Vikram Srinivasan, and Kee Chaing Chua. Extending the lifetime of wireless sensor networks through mobile relays. *IEEE/ACM Trans. Netw.*, 16(5):1108–1120, 2008.
- [70] Xing Xu, Ji Luo, and Qian Zhang. Delay tolerant event collection in sensor networks with mobile sink. In *INFOCOM*, pages 2471–2479, 2010.
- [71] Massimo Vecchio, Aline Carneiro Viana, Artur Ziviani, and Roy Friedman. Deep: Density-based proactive data dissemination protocol for wireless sensor networks with uncontrolled sink mobility. *Computer Communications*, 33(8):929–939, 2010.
- [72] Wendi Beth Heinzelman. *Application-specific protocol architectures for wireless networks*. PhD thesis, 2000. Supervisor-Chandrakasan, Anantha P. and Supervisor-Balakrishnan, Hari.
- [73] Azzedine Boukerche and Xin Fei. Adaptive data-gathering protocols with mobile collectors for vehicular ad hoc and sensor networks. In *WiMob*, pages 7–12, 2008.
- [74] J.T. Isaacs, D.J. Klein, and J.P. Hespanha. Algorithms for the traveling salesman problem with neighborhoods involving a dubins vehicle. Jan 2011.
- [75] David S. Johnson and Lyle A. Mcgeoch. *The Traveling Salesman Problem: A Case Study in Local Optimization*. 1997.
- [76] Richard Kershner. The number of circles covering a set. volume 61, pages pp. 665–671. The Johns Hopkins University Press, 1939.

- [77] M.J.A.M. Brummelhuis and H.J. Hilhorst. Covering of a finite lattice by a random walk. *Physica A: Statistical Mechanics and its Applications*, 176(3):387 – 408, 1991.