8-2014

Barrier Coverage in Wireless Sensor Networks

Zhibo Wang
University of Tennessee - Knoxville, zwang32@utk.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Zhibo Wang entitled "Barrier Coverage in Wireless Sensor Networks." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Computer Engineering.

Hairong Qi, Qing Cao, Major Professor

We have read this dissertation and recommend its acceptance:

Husheng Li, Xiaobing Feng

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Barrier Coverage in Wireless Sensor Networks

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Zhibo Wang

August 2014

This dissertation is dedicated to my loving wife Huiyan Pang and my cute son Henry Wang.

Acknowledgements

I would like to thank all the people who have inspired, encouraged and advised me in the preparation of this dissertation.

First, I would like to thank my co-advisors, Dr. Hairong Qi and Dr. Qing Cao. I am indebted to have had advisors that gave me all the freedom, resources, guidance and support during the period that lead up to this dissertation. Their broad knowledge in different areas inspired me and helped me overcome many difficulties in my research. I hope one day I would become as good an advisor as them. I would also like to thank my co-advisor, Dr. Zhi Wang, in Zhejiang University. I am deeply grateful to him for giving me chance to join the University of Tennessee, Knoxville. I would further like to thank Dr. Husheng Li and Dr. Xiaobing Feng. I greatly appreciate their time and input to this dissertation.

As a member of the AICIP group and the Lanterns group, I owe many thanks to my fellow graduate students. I enjoyed the many conversations and discussions that have had a tremendous impact on my research and myself as a person. Yang Bai, Mahmut Karakaya, Sangwoo Moon, Dayu Yang, Li He, Jiajia Luo, Shuangjiang Li, Wei Wang, Rui Guo, Liu Liu, Bryan Bodkin, Dani Martinez Capilla, Jilong Liao, Yanjun Yao, Kefa Lu and Lipeng Wan thank you very much.

Last but not least, I would like to express my greatest gratitude to my parents, my wife and my brother, for their unconditional love, support, understanding and encouragement. I would not be able to achieve anything without their love and support.

Abstract

Barrier coverage is a critical issue in wireless sensor networks (WSNs) for security applications, which aims to detect intruders attempting to penetrate protected areas. However, it is difficult to achieve desired barrier coverage after initial random deployment of sensors because their locations cannot be controlled or predicted. In this dissertation, we explore how to leverage the mobility capacity of mobile sensors to improve the quality of barrier coverage.

We first study the 1-barrier coverage formation problem in heterogeneous sensor networks and explore how to efficiently use different types of mobile sensors to form a barrier with pre-deployed different types of stationary sensors. We introduce a novel directional barrier graph model and prove that the minimum cost of mobile sensors required to form a barrier with stationary sensors is the length of the shortest path from the source node to the destination node on the graph. In addition, we formulate the problem of minimizing the cost of moving mobile sensors to fill in the gaps on the shortest path as a minimum cost bipartite assignment problem and solve it in polynomial time using the Hungarian algorithm.

We further study the k-barrier coverage formation problem in sensor networks. We introduce a novel weighted barrier graph model and prove that determining the minimum number of mobile sensors required to form k-barrier coverage is related with but not equal to finding k vertex-disjoint paths with the minimum total length on the WBG. With this observation, we propose an optimal algorithm and a faster greedy

algorithm to find the minimum number of mobile sensors required to form k-barrier coverage.

Finally, we study the barrier coverage formation problem when sensors have location errors. We derive the minimum number of mobile sensors needed to fill in a gap with a guarantee when location errors exist and propose a progressive method for mobile sensor deployment. Furthermore, we propose a fault tolerant weighted barrier graph to find the minimum number of mobile sensors needed to form barrier coverage with a guarantee.

Both analytical and experimental studies demonstrated the effectiveness of our proposed algorithms.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Wireless Sensor Networks

Recent advances in micro-electro-mechanical systems (MEMS) technology, wireless communications, computing and sensor technology have enabled the rapid development of low-cost, small-size sensor nodes that integrate sensing, data processing and wireless communication (Akyildiz et al., 2002; Yick et al., 2008). Each sensor node has the ability of sensing the environment, processing the sensing data and sending the results out via its wireless channel. As shown in Figure 1.1, wireless sensor networks (WSNs) are composed of a large number of sensors nodes which are densely deployed in the region of interest (ROI). WSNs are able to take environment measurements, perform data processing and route data to users, and also receive and execute commands from the users, which greatly changes the intersection manner between physical world and human beings. WSNs have been considered as one of the 21 most influential technologies at the 21th century (Coy and Gross, 1999) and one of the 10 emerging technologies that will change the world (Roush, 2003).

Compared to traditional wired networks (e.g., Internet), WSNs have a lot of unique features. First, sensor nodes are embodied with lots of sensors, such as GPS, temperature, light, pressure, magnetic, acoustic and camera, which are able to

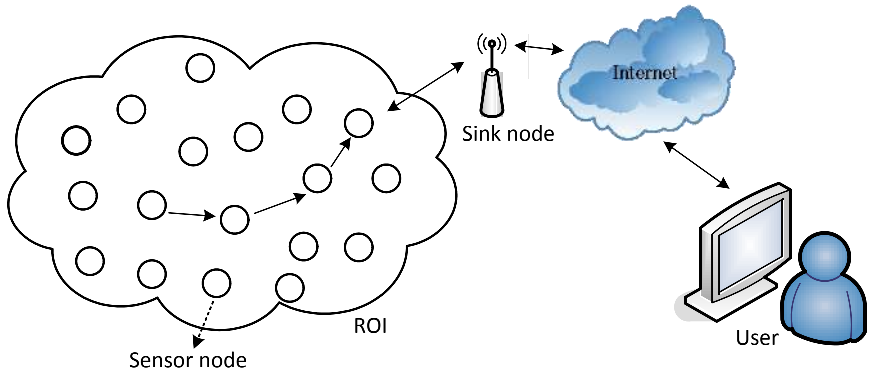


Figure 1.1: A typical architecture of WSNs.

monitor a wide variety of ambient conditions. This is also the fundamental difference between WSNs and other networks since the latter ones do not have the sensing ability. Second, sensor nodes communicate via wireless and often self-organize after being deployed in an ad hoc fashion. Therefore, WSNs are very flexible to be deployed and can be deployed at remote and dangerous terrains to fulfill specific tasks. Also, sensor nodes are usually densely deployed at the ROI, so they can collaborate with each other to fulfill tasks that are difficult to complete by individual sensor nodes. What's more, WSNs are data-centric, which is totally different from IP-based Internet.

The aforementioned unique features ensure a wide range of applications for sensor networks including military applications, environment motoring, homeland security and so on. We summarize some applications of WSNs in the following (Akyildiz et al., 2002; Yick et al., 2008).

- *Environment monitoring:* It is one of the primary applications of WSNs where hundreds of sensors nodes are deployed to monitor the environment for a specific task such as volcano condition monitoring, precision agriculture, forest fire detection, air and water pollution. Sensor nodes continuously sample the environment and report data to the base station.

- *Military applications:* WSNs play a very important role in military applications. For remote battlefield and other dangerous terrains, it is dangerous for soldiers to reach and stay there for too long time. Instead, WSNs can be deployed to watch out activities of enemies. Once an event happens, for example, intruders enter the protected area, WSNs can localize them and guide us to capture them.

- *Homeland security:* WSNs can also be used for homeland security. Instead of building walls on the border among countries, WSNs can be deployed to form virtual barriers that could actively detect any intruder crossing the border.

The unique features of WSNs and its wide applications have attracted extensive attentions from military and academic. In the early 1990s, DARPA first started the research on WSNs, and founded many projects such as SmartDust (Kahn et al., 1999), WINS (Pottie and Kaiser, 2000) and SensIT (Kumar and Shepherd, 2001). Meanwhile, WSNs also received attentions from the academic. Many groups such as CENS at UCLA, CITRIS at UCB and Harvard sensor network lab were founded and have greatly contributed the development of WSNs. It is not unreasonable to expect that the world will be covered with WSNs with access to them via the Internet (Lee et al., 2007).

Although the new technology is exciting and promising, it is challenging to design protocols and build systems for applications of using WSNs, which is because of the following reasons:

- *Resource limited:* Sensor nodes are usually power limited as well as computation capacities and memory limited. Consequently, energy efficiency and lightweight computation become very important factors for protocols design.

- *Wireless communication:* Sensor nodes can communicate and collaborate through wireless communication. However, the channel bandwidth is limited and the data transmission rate is low. Additionally, wireless communication is easy to be affected by wireless interference, attacks and environment conditions. Consequently, resource utilization and security are very important factors for protocols design.

- *Dynamic topology.* Sensor nodes are prone to failures due to lack of power, physical damage and environment interference. Therefore, the topology of a sensor network may changes very frequency. Consequently, robustness and fault tolerance should be considered for protocols design.

1.2 Intruder Detection with Barrier Coverage

Intruder detection is one of the most important problems of WSNs, the purpose of which is to detect any intruder that attempts to penetrate the ROI. In reality, lots of security applications need to detect intruders, such as border protection, critical infrastructure protection, and dangerous substance monitoring. In WSNs, area coverage (Cardei and Wu, 2004) and barrier coverage (Kumar et al., 2005) are proposed to realize the purpose of intruder detection.

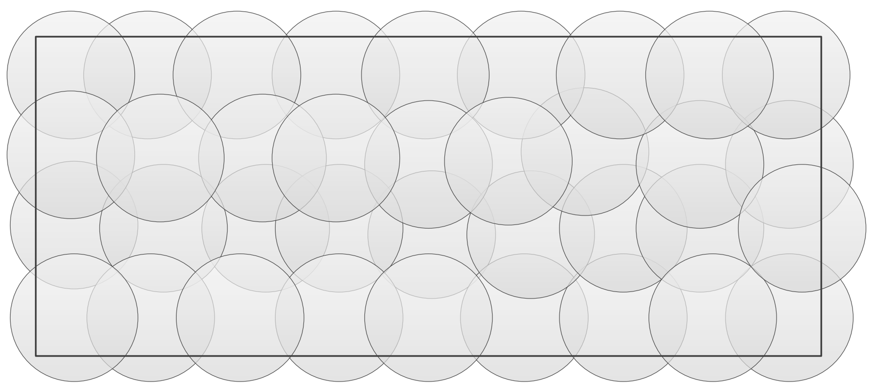


Figure 1.2: Area coverage.

Figure 1.2 shows an example of area coverage. Given a two-dimensional ROI, the sensor network provides area coverage for the ROI if and only if every point in the ROI is covered by at least one sensor*. Here a point is covered by a sensor means that the point is within the sensing region of the sensor. Accordingly, the sensor network provides k-area coverage for the ROI if and only if every point in the ROI is covered by at least k distinct sensors. Area coverage is the earliest coverage model proposed

*Without confusion, we use "sensor node" or "sensor" interchangeably in this dissertation.

in WSNs, which is able to monitor the events happening in the whole ROI and of course can detect the presence of intruders.

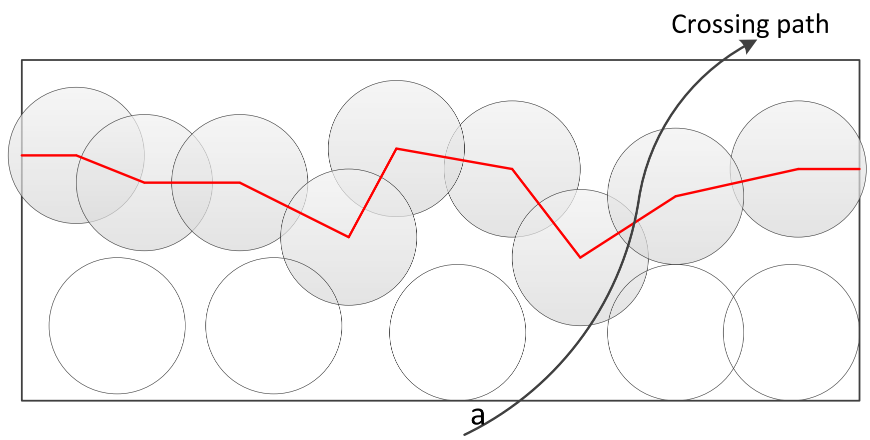


Figure 1.3: Barrier coverage.

Figure 1.3 shows an example of barrier coverage. Given a two-dimensional ROI, the sensor network provides barrier coverage if and only any intruder can be detected by at least one sensor no matter how it penetrates the ROI. As shown in Figure 1.3, the sensing regions of gray sensors form a virtual barrier denoted by a red line spanning from the left boundary to the right boundary of the ROI. We can observe that no matter how an intruder penetrates the ROI, it will be detected by at least one sensor. Therefore, the sensor network provides barrier coverage for the ROI.

Different from area coverage, barrier coverage does not care what event happens in the ROI but instead whether there are intruders penetrating the ROI. Barrier coverage does not even require most parts of the ROI to be covered by sensors, which significantly reduces the cost of sensor deployment. Therefore, barrier coverage in WSNs is an ideal solution for the purpose of intruder detection of security applications. Please refer to Cardei and Wu (2004); Huang and Tseng (2005); Wang and Xiao (2006); Ghosh and Das (2008); Wang (2011) for more information about different coverage problems.

1.2.1 Preliminaries of Barrier Coverage

Barrier coverage in WSNs was firstly defined by Kumar et al. (2005) where the sensors form a barrier for the intruders. The major goal of barrier coverage is to detect intruders as they cross a border or as they penetrate a protected area. Depending on the application scenarios, the ROI can be an open belt region (e.g., border) or closed belt region (e.g., airport). In this paper, we mainly focus on barrier coverage for an open belt region.

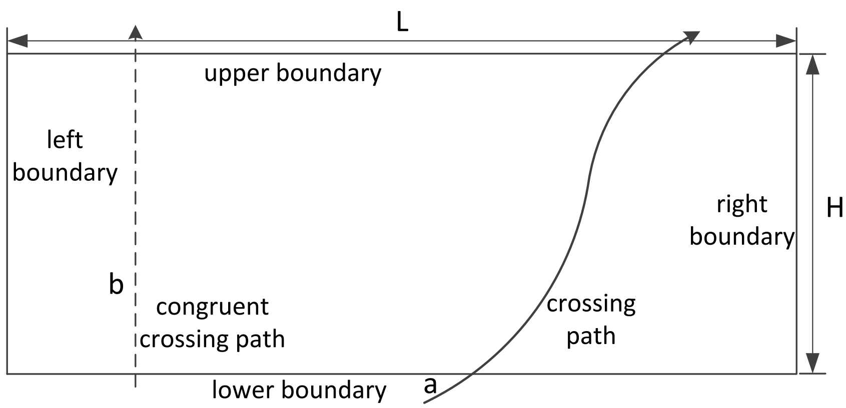


Figure 1.4: An illustration of belt region and crossing paths

As shown in Figure 1.4, the ROI is a two-dimensional rectangular belt region with length of L and width of H. A crossing path is any path that crosses the width of the region completely. A congruent crossing path is special crossing path that is congruent to the width of the belt. The path a and path b shown in Figure 1.4 demonstrate a random crossing path and a congruent crossing path, respectively. A path is said to be k-covered if it intersects with the sensing regions of at least k distinct sensors.

k-barrier coverage: A sensor network provides k-barrier coverage for the ROI (or the ROI is said to be k-barrier covered with a sensor network) if all crossing paths through the region are k-covered.

Two types of barrier coverage: ***weak barrier coverage*** and ***strong barrier coverage***, were introduced in Kumar et al. (2005). Weak barrier coverage requires the union of sensors form a virtual barrier in the horizontal direction from the left boundary to the right boundary, so that every intruder moving along congruent crossing paths can be detected. Figure 1.5 shows an example of weak barrier coverage. We can see that sensors in gray color form a virtual horizontal barrier spanning the whole range of the ROI. However, weak barrier coverage cannot guarantee the detection of intruders following any crossing path (e.g., path a in Figure 1.5). In contrast, strong barrier coverage requires that sensors form a virtual barrier spanning the whole range of the ROI so that any intruder can be detected no matter what crossing path it takes. An example of strong barrier coverage is shown in Figure 1.3.

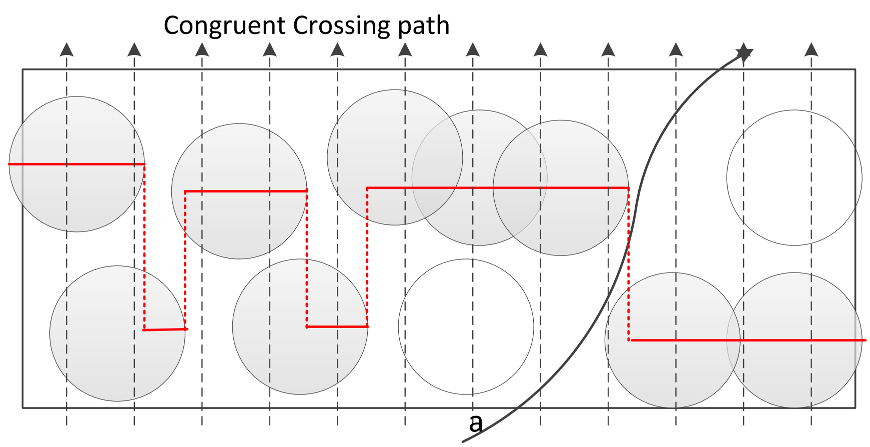


Figure 1.5: Weak barrier coverage

1.2.2 Applications and Challenges

The unique purpose of barrier coverage ensures its wide usage for various security applications. In the following, we summarize several typical applications that barrier coverage can be used for.

- *Border protection:* Border protection is the most important issue for a country. Lots of efforts and money have been spent on it by the department of homeland security to prevent intruders (e.g., terrorists, drug dealers and illegal immigrants) from other countries. The traditional way is to build physical walls on the border, which is very expensive because of the long distance of the border but not effective as the physical walls cannot detect intruders even they are crossing. Instead, sensor networks can be deployed on the border that actively detect any intruder that attempts to cross the border. Once an intruder crosses a barrier formed by sensors, sensors can detect the intruder and report the result to the department of homeland security.

- *Critical infrastructures protection:* Security is very important for critical infrastructures, such as military base and nuclear power plant, since unauthorized intruders may steal secret and destroy these infrastructures. To efficiently and actively detect unauthorized intruders, sensor networks can be deployed at the outside of the protected regions.

- *Dangerous substance monitoring:* Sensor networks can also be deployed at the outside of dangerous and poisonous factories to monitor the leakage of dangerous substance such as nuclear and poisonous gas. A barrier surrounding the factory is formed by sensors. Once any sensor on the barrier detect the leakage of dangerous substance, it sends an alarm to the security department, and then corresponding measures can be taken to save people.

While the technique is promising, there are a lot of challenging issues to provide barrier coverage for security applications.

First, a barrier might not be formed after initial deployment of sensor networks. Due to budget limit and lots of ROIs are dangerous or hard to reach, random deployment of sensors (e.g., dropped by aircraft) is commonly used, which makes the locations of sensors random and unpredictable. Therefore, it is difficult to form a barrier after initial deployment of sensor networks. Consequently, the first challenging issue is how to determine whether a ROI is barrier covered or not. If the ROI is not barrier covered, what measure can be taken to form a barrier?

Second, to fulfill the application requirements, different types of sensors (e.g., camera, radar and ground sensor are deployed in SBInet project (Archibold, 2007)) are deployed to form a barrier to detect intruders. Different types of sensors have different costs, communication and computation capabilities. Due to the budget limit, we cannot use all the types of sensors as much as we want. Therefore, a challenging issue is to design deployment methods that achieve application requirements of barrier coverage by using different types of sensors within the budget limit.

What's more, barrier coverage is highly affected by the location errors of sensors. As we mentioned, sensors are usually randomly deployed in the ROI, which makes the true locations of sensors unpredictable. It is cost-expensive to equip GPS receivers on each sensor considering the fact of large-scale sensor networks. To reduce the cost for localization, a commonly used solution is equipping GPS receivers on a small portion of sensors which are called beacons, and estimating other sensors' locations using the beacons' location information. A lot of localization algorithms have been proposed to estimate the location of sensors, including the range-based and the range-free localization algorithms. However, none of them can provide the accurate locations and therefore inevitably has location errors. When location error of sensors exist, it is difficult to say whether two sensors overlap or not. Therefore, the first question is what's the effect of location errors on barrier coverage? How to guarantee the sensor network provide barrier coverage when location errors exist?

1.3 State-of-the-art

Barrier coverage in WSNs has attracted extensive attentions and a lot of protocols have been proposed in recent years.

The concept of barrier coverage first appeared in Gage (1992) in the context of robotic sensing. Kumar et al. (2005) firstly defined the notion of k-barrier coverage for WSNs and proposed an efficient algorithm to determine whether a belt region is k-barrier covered or not. They also introduced two notions of probabilistic barrier coverage - weak barrier coverage and strong barrier coverage, and derived critical conditions for weak k-barrier coverage in randomly deployed sensor networks. Kumar et al. (2007) further proposed a centralized, optimal sleep-wakeup algorithm to prolong the lifetime of barrier coverage. Chen et al. (2007) introduced the notion of local barrier coverage and devised localized sleep-wakeup algorithms that provide near-optimal solutions. Liu et al. (2008) devised an efficient distributed algorithm to construct multiple disjoint barriers for strong barrier coverage in a randomly deployed sensor network on a long irregular strip region. Saipulla et al. (2009) studied the barrier coverage of the line-based deployment rather than the Poisson distribution model, and a tight lower-bound for the existence of barrier coverage was established. Li et al. (2012) proposed an energy efficient scheduling algorithm for barrier coverage with probabilistic sensing model. Li et al. (2011) studied the weak k-barrier coverage and derived a lower bound for the probability of weak k-barrier coverage with and without considering the border effect, respectively. Beyond line-based sensor deployment, He et al. (2013) designed curve-based sensor deployment algorithms for barrier coverage.

Recently, barrier coverage in directional sensor networks has gradually received more and more attention. Zhang et al. (2009) studied the strong barrier coverage problem for rotationally directional sensors. A novel full-view coverage model was introduced in Wang and Cao (2011b) for camera sensor networks. A full-view coverage verification method was proposed and an estimate of deployment density to achieve full-view coverage for the whole monitored area was given. With the full-view coverage model, Wang and Cao (2011a) further proposed a novel method to select camera sensors from an arbitrary deployment to form a camera barrier. The minimum camera barrier coverage problem was studied in camera sensor networks (Ma et al., 2012). Tao et al. (2012) investigated the problem of finding appropriate orientations of directional sensors such that they can provide strong barrier coverage.

With the development of mobile sensors, sensor mobility is exploited to improve barrier coverage. Shen et al. (2008) studied the energy efficient relocation problem for barrier coverage with mobile sensors. A centralized barrier algorithm was proposed to compute the relocated positions for all sensors to form a barrier. Keung et al. (2010)

focused on providing k-barrier coverage against moving intruders. They demonstrated that the problem is similar to classical kinetic theory of gas molecules in physics, and derived the inherent relationship between barrier coverage and a set of crucial system parameters including sensor density, sensor and intruder density. Ban et al. (2010) studied the problem on how to relocate mobile sensors to construct k grid barriers with minimum energy consumption. They gave an integer linear programming model and devised an approximation algorithm AHGB to construct one grid barrier. Saipulla et al. (2010a) exploited sensor mobility to improve barrier coverage. They proposed a greedy algorithm to find barrier gaps and adopted maximum flow algorithm to relocate mobile sensors to fill the gaps. They also studied how to improve barrier coverage using mobile sensors with limited mobility in Saipulla et al. (2010b).

1.4 Motivations and Contributions

Barrier coverage is a critical issue in WSNs for security applications, which cannot be guaranteed after initial random deployment of sensors, especially for sensors with limited sensing angles (e.g., cameras). Although lots of work have been done on barrier coverage, however, most of them mainly focus on critical condition analysis and barrier construction for stationary sensors, little effort has been made to explore how to efficiently use mobile sensors to form barrier coverage with stationary sensors. In addition, existing studies on barrier coverage only focus on homogeneous sensor network, little effort has been put on barrier coverage with heterogenous sensor networks which is more practical and useful in real-world applications. Moreover, to the best of our knowledge, none of existing work explores the effects of location errors of sensors on barrier coverage and how to guarantee the formation of barrier coverage when location errors exist.

In this dissertation, we study the 1-barrier coverage formation problem in heterogeneous sensor networks and explore how to efficiently use different types of mobile sensors to form a barrier with pre-existing stationary sensors. To the best of our knowledge, we are the first to study barrier coverage formation problem in heterogeneous sensor networks. In specific, we introduce the notion of directional barrier graph to model barrier coverage formation problem. We prove that the minimum cost of mobile sensors required to form a barrier with stationary sensors is the length of the shortest path from the source node to the destination node on the directional barrier graph. We then formulate the problem of minimizing the cost of moving mobile sensors to fill in the gaps on the shortest path as a minimum cost bipartite assignment problem and solve it in polynomial time using the Hungarian algorithm.

We further study the k-barrier coverage formation problem in WSNs. To the best of our knowledge, we are the first to study k-barrier coverage formation problem in WSNs. We introduce a novel weighted barrier graph to model the barrier coverage formation problem and prove that the minimum number of mobile sensors required to form k-barrier coverage with stationary sensors is related with but not equal to finding k vertex-disjoint paths with the minimum total length on the weighted barrier graph. Based on this observation, we propose an efficient optimal algorithm as well as a greedy algorithm to find the minimum number of mobile sensors needed to form k-barrier coverage. Meanwhile, we also propose an efficient optimal algorithm as well as a greedy algorithm to find the maximum number of barriers that can be formed given pre-deployed stationary sensors and available mobile sensors.

Finally, we study the barrier coverage formation problem when sensors have location errors. To the best of our knowledge, we are the first to study this problem. We study the barrier coverage problem when sensors have location errors and deploy mobile sensors to improve barrier coverage if the network is not barrier covered after initial deployment. We analyze the relationship between the true distance and the measured distance of two stationary sensors and derive the minimum number of mobile sensors needed to connect them with a guarantee when location errors exist. Furthermore, we propose a fault tolerant weighted barrier graph, based on which we prove that the minimum number of mobile sensors needed to form barrier coverage with a guarantee is the length of the shortest path on the graph.

1.5 Dissertation Organization

The dissertation is organized as follows: In Chapter 2, we explore how to efficiently form 1-barrier coverage using mobile sensors with stationary sensors in heterogeneous sensor networks. Chapter 3 further studies the k-barrier coverage formation problem in sensor networks. In chapter 4, we analyze the effect of location errors of sensors on barrier coverage and propose a fault tolerant barrier coverage algorithm to form a barrier with a guarantee using the minimum number of sensors. Finally, we conclude this dissertation with a summary of work and directions for future work in Chapter 5.

Chapter 2

Cost-Effective 1-barrier Coverage Formation in Sensor Networks

In this chapter, we study the 1-barrier coverage formation problem in heterogeneous sensor networks and explore how to efficiently use different types of mobile sensors with different costs and sensing models to form a barrier with pre-deployed different types of stationary sensors. In specific, we introduce the notion of directional barrier graph to model barrier coverage formation problem, and prove that the minimum cost of mobile sensors required to form a barrier with stationary sensors is the length of the shortest path from the source node to the destination node on the directional barrier graph. We then formulate the problem of minimizing the cost of moving mobile sensors to fill in the gaps on the shortest path as a minimum cost bipartite assignment problem and solve it in polynomial time using the Hungarian algorithm. Finally, we demonstrate the effectiveness of the proposed algorithms using simulations, where we show that the proposed solutions work for both weak barrier coverage and strong barrier coverage problems in heterogeneous sensor networks.

2.1 Introduction

Wireless sensor networks (WSNs) have been widely used as an effective surveillance tool for security applications, such as battlefield surveillance, border protection, and airport intruder detection. To detect intruders who penetrate the regions of interest (ROI), we need to deploy a set of sensor nodes that can provide coverage of the ROI, a problem that is often referred to as *barrier coverage* (Kumar et al., 2005), where sensors form a *barrier* to prevent intruders from crossing the ROI. When only stationary sensors are used, however, after the initial random deployment, it is possible that sensors could not form a barrier due to gaps in their coverage, which would allow intruders to cross the ROI without being detected. In fact, it is difficult if possible at all to improve barrier coverage for sensor networks consisting of only stationary sensors. Fortunately, with recent technical advances, practical mobile sensors (e.g., Robomote (Dantu et al., 2005), Packbot (Somasundara and Ramamoorthy, 2007)) have been developed, which provides us a way to improve barrier coverage performance after sensor networks have been deployed.

An intruder detection system could consist of only one type of sensors where all sensors have the same sensing range and angle. This kind of sensor network is often refereed as *homogeneous sensor network*. Cameras probably are the most widely used sensors for security applications. For example, the FREEDOM system (Ramirez, 2006), deployed on the border between Mexico and United States, uses cameras to detect illegal intruders. However, in reality, it is more general that a system consists of different types of sensors where they have different sensing ranges, sensing angles and costs. This kind of sensor network is often refereed as *heterogeneous sensor network*. For example, The SBInet project (Archibold, 2007) supported by US government uses cameras, radars and ground sensors to construct a virtual fence to detect illegal intruders (e.g., drug dealers and illegal immigrants). In this chapter, we mainly focus on the more general heterogenous sensor network and consider the homogeneous sensor network as a special case of the heterogeneous sensor network.

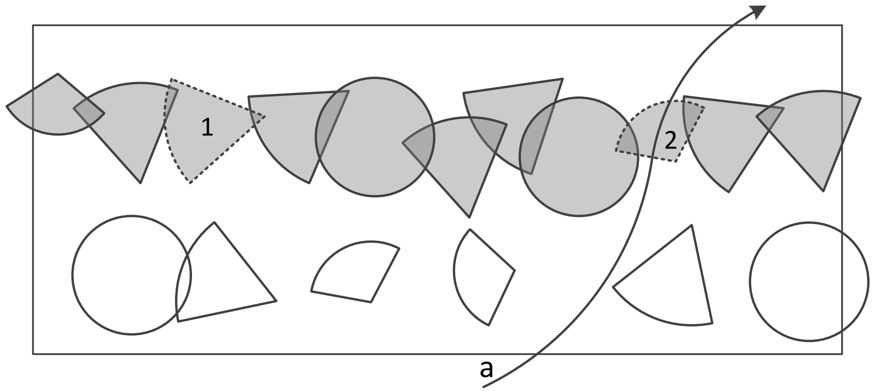


Figure 2.1: Illustration of forming a strong barrier with the help of mobile sensors. Mobile sensors 1 and 2 fill in the gaps between stationary sensors and form a strong barrier for the ROI.

A lot of work has been done on barrier coverage. However, most of existing work mainly focus on critical condition analysis for stationary sensors (Kumar et al., 2005; Chen et al., 2007; Liu et al., 2008; Saipulla et al., 2009), little effort has been made to explore how to efficiently use mobile sensors to form barrier coverage with stationary sensors. In addition, none of existing work explores barrier coverage in heterogenous sensor network. To the best of our knowledge, we are the first to study how to efficiently form barrier coverage using mobile sensors in heterogeneous sensor network. In particular, we consider a two-phase deployment: in the first phase, after stationary sensors are deployed, their barrier gaps are identified and the cost of mobile sensors needed can be calculated; in the second phase, mobile sensors are deployed and move to desired locations to fill in these gaps to form a barrier. Figure 2.1 shows an example of forming a strong barrier using mobile sensors. Mobile sensors 1 and 2 fill in the gaps between stationary sensors and form a strong barrier with pre-existing stationary sensors for the ROI.

There are lots of challenging issues for the barrier coverage formation problem in heterogeneous sensor networks. First, it is challenging to determine whether two sensors overlap with each other or not and calculate the distance between sensors due to the fact that different types of sensors have different sensing ranges, sensing angles and facing directions. Second, since sensors are randomly deployed, it is challenging to determine whether the sensors already form a barrier or not after initial deployment. Third, the manufacturing costs of mobile sensors are usually much higher than that of stationary sensors (Dantu et al., 2005), which motivates us to minimize the cost of mobile sensors needed to form a barrier. Since there are too many ways of deploying different types of mobile sensors to fill in different gaps in the network, it is therefore challenging to find the optimal solution of using mobile sensors to form barrier coverage with the deployed stationary sensors while minimizing the total cost of mobile sensors. Finally, mobile sensors should move to expected locations to fill in the gaps between stationary sensors. However, sensor movement costs a lot of energy and mobile sensors are often power limited. Therefore, another challenging issue is how to schedule and move mobile sensors to the expected locations so that the total moving cost is minimized.

To solve these challenging issues, we introduce the directional barrier graph model and propose the directional barrier coverage algorithm to find the minimum cost of mobile sensors needed to form a barrier. The main contributions of this chapter are summarized as follows:

- To the best of our knowledge, we are the first to study the barrier coverage formation problem using mobile sensors in heterogeneous sensor networks.

- We introduce the directional barrier graph model and prove that determining the minimum cost mobile sensors required to form a barrier is equivalent to finding the shortest path from the source node (left boundary) to the destination node (right boundary) on the directional barrier graph.

- We propose an efficient greedy movement algorithm to efficiently schedule mobile sensors to different gaps while minimizing the total moving cost. In addition, for homogeneous sensor network, we propose a position based optimal movement algorithm that formulates the movement algorithm as a minimum cost bipartite assignment problem, and solve it in polynomial time using the Hungarian algorithm.

- We conduct extensive simulations to evaluate the performance of the proposed algorithms and experimental results validate their effectiveness.

The remainder of this chapter is organized as follows. We present the network model and the sensing model in Section 2.2. The barrier coverage formation problem in heterogeneous sensor networks is formulated in Section 2.3. We analyze the minimum cost mobile sensor problem (MCMS) and the minimum cost barrier formation problem (MCBF) in Section 2.4 and Section 2.5, respectively. Extensive performance evaluation of our algorithm is presented in Section 2.6. Finally, we conclude this chapter in Section 2.7.

2.2 System Model

In this section, we present the system model including the network model and the sensing model for sensors.

We assume that the ROI is a two-dimensional rectangular belt area and different types of stationary sensors are randomly deployed in the belt region. After the minimum cost of mobile sensors needed is calculated, the required types of mobile sensors are deployed further to form a barrier with pre-existing stationary sensors.

We assume there are k types of sensors and different types of sensors have different sensing ranges, sensing angles and costs. However, the same type of sensors have the same sensing range, sensing angle and cost but may have different facing directions. In general, the sensing region of omni-directional sensors is characterized by a 0-1 disk model which however cannot represent directional sensors (e.g., cameras, acoustic sensors and radars). In this chapter, we adopt a more general model to characterize the sensing region of different types of sensors. As shown in Figure 2.2(a), the sensing region of a sensor is characterized by a sector. Let $T(j) = (r(j), \alpha(j), c(j))$ denote the sensing model and cost of type j sensors, where $r(j)$ is the sensing range, $\alpha(j)$ is half of the sensing angles, $c(j)$ is the cost of type j sensors. Let s_i denote the sensor i and suppose it is a jth type sensor, then it can be represented by a 5-tuple $< x_i, y_i, r(j), \alpha(j), \beta_i >$, where $l_i = (x_i, y_i)$ is the two-dimensional location of the sensor i, $r(j)$ is its sensing range and $\alpha(j)$ is half of its sensing angle, and β_i is the facing direction of sensor i. We assume that β_i is uniformly distributed in $[0, 2\pi)$, e.g., $\beta_i \sim U(0, 2\pi)$. Note that omni-directional sensing model is a special case of directional sensing model when $2\alpha = 2\pi$.

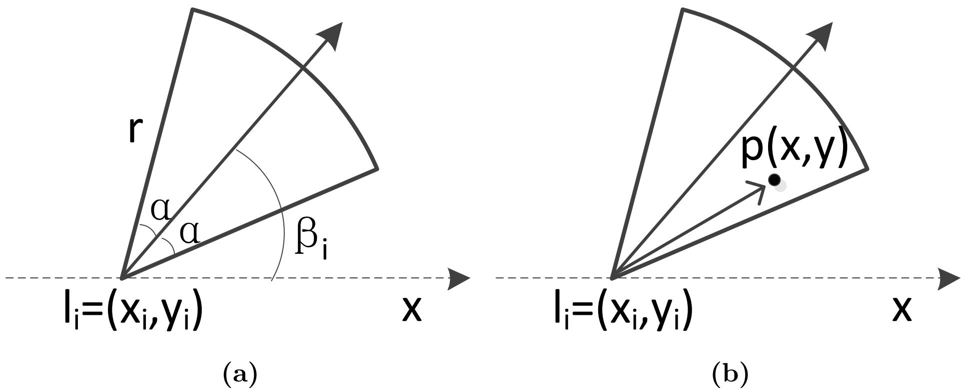


Figure 2.2: (a) The sector sensing model for directional sensors; (b) A point p is covered by the sensor s_i.

Definition 1. *A two-dimensional point* $p = (x, y)$ *is said to be covered by a directional sensor* $s_i = < x_i, y_i, r(j), \alpha(j), \beta_i >$ *if and only if the following two conditions are satisfied.*

- $(x - x_i)^2 + (y - y_i)^2 \le r(j)^2$,
- $ang(\overrightarrow{l_i p}) \in [\beta_i - \alpha(j), \beta_i + \alpha(j)]$, *where* $ang(\cdot)$ *denotes the angle of* $(\cdot)$.

The largest coverage range of the jth type sensors, denoted by $l_r(j)$, is the longest line in its sensing sector, which is either the sensing radius or the longest chord of the sector. Therefore, we have

$$l_r(j) = \begin{cases} \max(r(j), 2r\sin\alpha(j)) & \text{for } 0 \le \alpha(j) < \frac{\pi}{2}, \\ 2r(j) & \text{for } \frac{\pi}{2} \le \alpha(j) \le \pi. \end{cases}$$

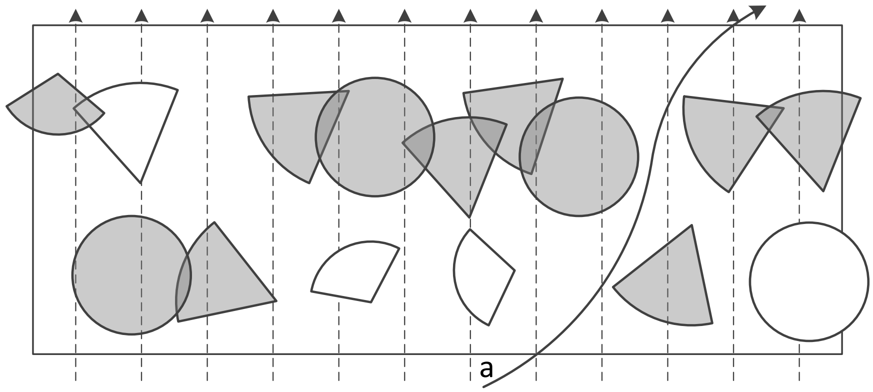


Figure 2.3: An example of weak barrier coverage formed by sensors in gray color. A weak barrier can detect intruders following congruent crossing paths, however, can not guarantee the detection of intruders following any crossing path (e.g., path a).

2.3 Problem Formulation

In this section, we formally describe the barrier coverage formation problem in heterogeneous sensor networks.

As shown in Figure 2.3, the belt region is generally a long and thin strip with length of L and width of H. A crossing path is a path that crosses the complete width of the area from the lower boundary to the upper boundary(e.g., path a). A congruent crossing path (e.g., dashed lines) is a special crossing path that is orthogonal to the upper and lower boundaries. An intruder may attempt to penetrate the area along any crossing path.

Two types of barrier coverage: *weak barrier coverage* and *strong barrier coverage*, were introduced in (Kumar et al., 2005). Weak barrier coverage requires that the union of sensors form a barrier in the horizontal direction from the left boundary to the right boundary, called a *weak barrier*, so that every intruder moving along congruent crossing paths can be detected. As shown in Figure 2.3, sensors in gray color form a weak barrier to guarantee the detection of intruders along any congruent crossing path. However, weak barrier coverage can not guarantee the detection of intruders following random crossing paths (e.g., path a). In contrast, strong barrier coverage requires that the union of sensors forms a barrier from the left boundary to the right boundary, called a *strong barrier*, so that every intruder can be detected no matter what crossing path it takes. An example of strong barrier coverage is shown in Figure 2.1. In this chapter, we address the barrier coverage formation problem for both weak and strong barrier coverage.

By leveraging the mobility feature of mobile sensors, we propose to use mobile sensors to fill in gaps between pre-deployed stationary sensors to form a weak/strong barrier. There are different types of sensors with different sensing ranges, sensing angles and costs. The cost of mobile sensors needed depends on the network topology of stationary sensors and also how many mobile sensors of each types are needed. The first problem we want to solve is to minimize the cost of mobile sensors needed to form barrier coverage, which is called the *minimum cost mobile sensor* problem (MCMS). Let $M(j)$ denote the required number of the jth type mobile sensors. The MCMS problem is formulated as follows:

$$\begin{aligned} \text{Minimize} \quad & \sum_{j=1}^{k} c(j)M(j) \\ \text{subject to} \quad & \text{Stationary and mobile sensors form a barrier.} \\ & M(j) \geq 0 \;\&\; M(j) \in \mathbb{Z}, \forall j = 1, 2, \cdots, k. \end{aligned}$$

When the minimum cost is zero, no mobile sensors are needed and therefore the ROI is barrier covered after initial deployment of stationary sensors, otherwise the ROI is not barrier covered. Thus, whether a network is barrier covered or not after initial deployment of stationary sensors can be easily answered once the MCMS problem is solved.

After the required number for each type of mobile sensors are known, we then can deploy mobile sensors as needed to the network. In order to form a barrier, mobile nodes should move to different gaps, which also consumes a lot of energy. In general,the energy consumed by mobile sensors is proportional to the moving distance. In order to prolong the lifetime of mobile sensors, the total moving distance should be minimized. Therefore, we need to study the problem of scheduling mobile sensors to fill in different gaps while minimizing the total moving distance, which is called the *minimum cost barrier formation* problem (MCBF). Let $\bar{G} = (g_1, g_2, \cdots, g_\varsigma)$ denotes the set of gaps needed to be filled in that uses the minimum cost of mobile sensors to form a barrier. Let $\tau = \sum_{j=1}^{k} M(j)$ denote the required number of mobile sensors to form a barrier, and $M = (m_1, m_2, \cdots, m_\tau)$ denote the set of deployed mobile sensors. Suppose the original location of a mobile sensor m_i is (x_i, y_i) and its target location is $(\bar{x}_i, \bar{y}_i)$. The objective is to schedule mobile sensors to fill in different gaps while minimizing the total moving distance. Therefore, the MCBF problem is formulated as follows:

$$\begin{aligned} \text{Minimize} \quad & \sum_{i=1}^{\tau} \sqrt{(x_i - \bar{x}_i)^2 + (y_i - \bar{y}_i)^2} \\ \text{subject to} \quad & g_i \text{ is filled}, \forall i = 1, 2, \cdots, \varsigma. \end{aligned}$$

It is worth noting that the set of target locations are unknown a prior, which makes the MCBF problem difficult to solve. In the following sections, we present the formulation and analysis of the MCMS problem and the MCBF problem.

2.4 Minimum Cost Mobile Sensor (MCMS) Problem

In this section, we present the formulation and analysis of the MCMS problem. We first describe how to group sensors into clusters and then introduce the directional barrier graph model for the barrier coverage formation problem. With the graph model, we prove that the minimum cost of mobile sensors needed to form a barrier is the length of the shortest path from the source node to the destination node on the graph.

2.4.1 Cluster Formation

The fundamental problem for weak barrier coverage is to decide whether two sensors overlap in the horizontal direction or not. Let x_i^L and x_i^R denote the left and the right coverage boundary of sensor s_i in the horizontal direction, which can be obtained by geometric calculation. We can claim that two sensors s_i and s_j are *directly weakly connected* if they overlap in the horizontal direction, that is, $x_i^L \leq x_j^L \leq x_i^R$ or $x_j^L \leq x_i^L \leq x_j^R$. Based on this claim, two sensors s_i and s_k are said to be *weakly connected through sensor* s_j if s_i and s_k are not directly weakly connected but both of them are directly weakly connected to s_j.

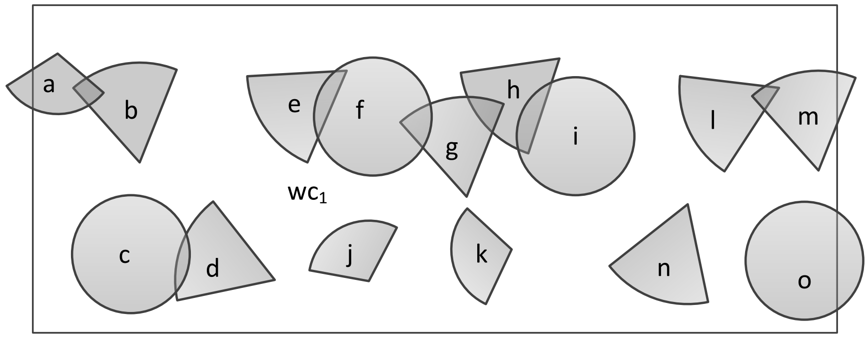
Definition 2. *A* weak cluster *is the union of a set of sensors where each sensor is weakly connected with the rest of sensors in the set either directly or through other sensors.*

As shown in Figure 2.4(a), sensor c is directly weakly connected with sensor b and sensor d. Although sensor b and sensor d are not directly weakly connected, they are weakly connected through sensor c. Similarly, sensor e is weakly connected to sensors a, b, c and d. We can see that all these sensors form one weak cluster, doted by wc_1. We have the following Lemma.

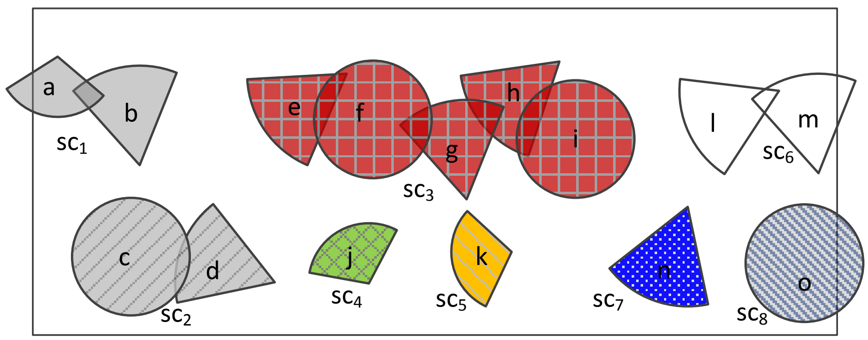
Lemma 2.0.1. *Given a belt region with length L, a weak barrier is formed if there exists a weak cluster whose coverage region in the horizontal direction is* $[0, L]$.

Similar to the definition for weak barrier coverage, we can claim that two sensors s_i and s_j are *directly strongly connected* if they overlap with each other. Two sensors s_i and s_k are said to be *strongly connected through sensor* s_j if s_i and s_k are not directly strongly connected but both of them are directly strongly connected to s_j.

Definition 3. *A* strong cluster *is the union of a set of sensors where each sensor is strongly connected with the rest of sensors in the set either directly or through other sensors.*



(a) Weak clusters



(b) Strong clusters

Figure 2.4: Illustration of weak/strong clusters

As shown in Figure 2.4(b), sensor f are directly strongly connected with sensor e and sensor g. Although sensor e and sensor g are not directly strongly connected, they are strongly connected through sensor f. Therefore, according to the Definition 3, the network has 8 strong clusters after initial sensor deployment.

Lemma 2.0.2. *A strong barrier is formed if there exists a strong cluster overlapping with both the left boundary and the right boundary of the belt region.*

The problem to decide whether two sensors overlap with each other is easy to answer for omni-directional sensors of disk sensing model. However, it is much harder for directional sensors with sector sensing model due to their different facing directions and limited angle of views. For example, we can claim that two omni-directional sensors with sensing range of r overlap with each other if the Euclidean distance between their centers is smaller than or equal to $2r$. However, two directional sensors might not overlap even when they are very close to each other, e.g., two cameras can be side by side but looking at opposite directions. Therefore, using only distance information would not work for directional sensors. Note that the sensing region of a directional sensor is bounded by two line segments and an arc. We have the following Lemma.

Lemma 2.0.3. *Directional sensors s_i and s_j overlap with each other if and only if there exists at least one intersection between the two line segments and the arc of s_i and the two line segments and the arc of s_j.*

Proof. $\Rightarrow$. If there exists an intersection between the two line segments and the arc of s_i and the two line segments and the arc of s_j, there must exist one point covered by both s_i and s_j. Then s_i and s_j overlap with each other.

$\Leftarrow$. If s_i and s_j overlap with each other, there exists at least one point covered by both s_i and s_j. Since the point is bounded by the two line segments and the arc of each sensor, there must exist at least one intersection between the two line segments and the arc of s_i and that of s_j. □

Based on Lemma 2.0.3, the problem of deciding whether s_i and s_j overlap or not can be simplified to check whether there exist intersections between the line segments of s_i and the line segments of s_j, the line segments of s_i (s_j) and the arc of s_j (s_i), and the arc of s_i and the arc of s_j.

2.4.2 Directional Barrier Graph (DBG)

Due to random deployment and directional sensing features, the sensor network may not be able to provide barrier coverage for the ROI, especially for strong barrier coverage. The usage of mobile sensors can potentially fill in coverage holes and help form barrier coverage. The objective of the MCMS problem is to minimize the cost of mobile sensors needed. In this section, we introduce a novel graph model, directional barrier graph (DBG), to formulate the barrier coverage formation problem and solve the MCMS problem.

According to Lemma 2.0.1 and Lemma 2.0.2, forming a weak/strong barrier actually is equal to forming a weak/strong cluster. Denote the set of clusters by $C = \{c_1, c_2, \cdots, c_\pi\}$. Depending on the application requirement, this could be either a set of weak clusters or strong clusters. In the DBG, we consider each cluster as a vertex, and denote the left and right boundary of the belt region by virtual vertices s and t, respectively. s is called the source node and t is called the destination node. The following is a formal definition of the DBG.

Definition 4. *A* ***directional barrier graph*** *of a sensor network is represented by $G = (V, E, W)$. V is the set of vertices corresponding to the left boundary (s), clusters (C) and the right boundary (t), that is, $V = \{v_1, v_2, \cdots, v_{\pi+2}\} = \{s \cup C \cup t\}$. $E = \{e(v_i, v_j)\}$ is the set of edges between any pair of vertices. $W : E \to \mathbb{R}$ is the set of weights of each edge, where the weight $w(v_i, v_j)$ of edge $e(v_i, v_j)$ is the minimum cost of mobile sensors needed to connect v_i and v_j.*

Figure 4.4(a) shows the DBG for weak barrier coverage, which is constructed based on weak clusters shown in Figure 2.4(a). Figure 4.4(b) shows the DBG of strong barrier coverage, which is constructed based on strong clusters shown in Figure 2.4(b). The weight of each edge is the minimum cost of mobile sensors needed to fill in the gap. We will describe how to calculate weights later in detail. Let $T(1) = (18, \pi/3, 9)$, $T(2) = (30, \pi/6, 8)$, and $T(3) = (18, \pi, 15)$, the values of the weights are shown in the figure. Note that $w(s, sc_1) = 0$ because sc_1 overlaps with the left boundary. With the DBG, we can have the following theorems. It is worth noting that all the conclusions work for both weak and strong barrier coverage.

Theorem 2.1. *The minimum cost of mobile sensors needed to form a barrier with pre-deployed stationary sensors is exactly the length of the shortest path from s to t on the directional barrier graph G.*

Proof. According to the definition of directional barrier graph G, if we want to form a barrier, we only need to choose a path from s to t, and put exactly the required

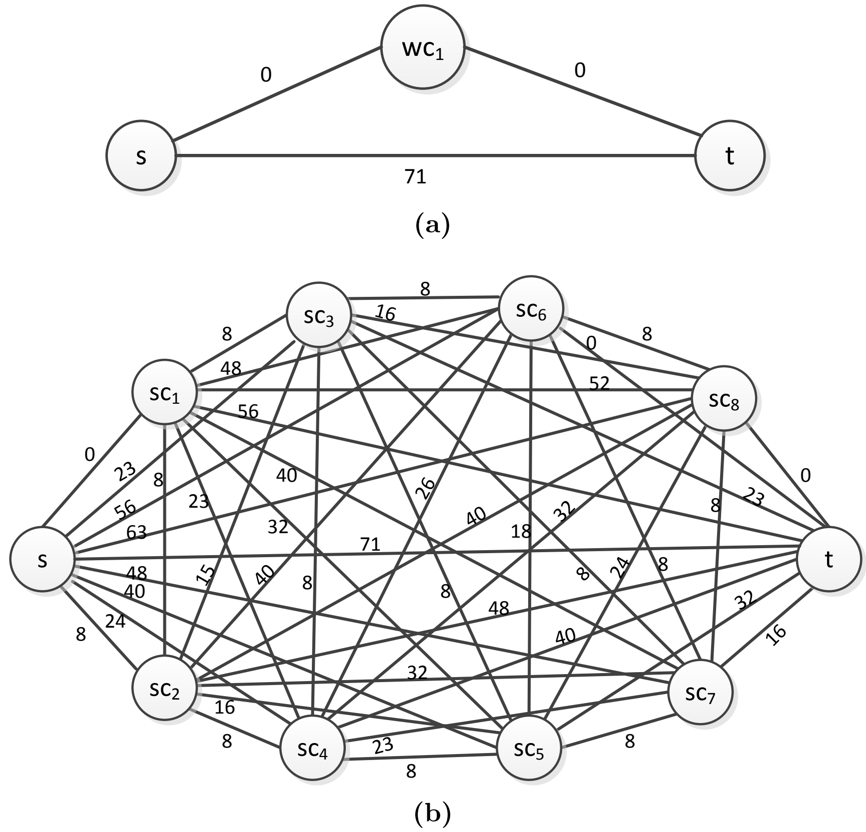


Figure 2.5: Directional barrier graph representation for: (a) weak barrier coverage of Figure 2.4(a); (b) strong barrier coverage of Figure 2.4(b).

cost of mobile sensors needed on each edge of the path. That is, for a chosen path, the cost of mobile sensors required to form a barrier is equal to the sum of weights of all edges on the path, which is the length of the path. Therefore, the minimum cost of mobile sensors needed to form a barrier is the length of the shortest path from s to t on graph G. $\square$

Theorem 2.2. *A region is barrier covered after initial random deployment if and only if the length of the shortest path from s to t on the directional barrier graph is* 0.

Proof. $\Rightarrow$. If the length of the shortest path from s to t on the directional barrier graph is 0, there exists at least one cluster that overlaps with both the left boundary and the right boundary of the belt region. Therefore, the region is barrier covered.

$\Leftarrow$. If the region is barrier covered, there exists one cluster that overlaps with both the left boundary and the right boundary of the belt region. Then weights are 0 from s to the cluster and from the cluster to t. Therefore, there exist one path from s to t with a length of 0, which is the shortest path. $\square$

Theorem 2.3. *The minimum cost of mobile sensors required to form a barrier for a given belt region is upper bounded by $w(s,t)$. For homogeneous sensor network, $w(s,t) = c\lceil \frac{L}{l_r} \rceil$ where c and l_r are the cost and the largest coverage range of a mobile sensor.*

Proof. The edge $e(s,t)$ means to deploy mobile sensors directly from the left boundary to the right boundary of the belt region. The path containing only the edge $e(s,t)$ could either be the shortest or not. If it is not the shortest path, according to Theorem 2.1, the minimum cost of mobile sensors required is smaller than $w(s,t)$; otherwise, the minimum number of mobile sensors required is equal to $w(s,t)$. Therefore, the minimum number of mobile sensors required to form a barrier is always upper bounded by $w(s,t)$.

As for homogeneous sensor network, all the mobile sensors have the same sensing range and cost. The optimal way of connecting the left and right boundary is to deploy mobile sensors continuously in a horizontal straight line. Therefore, the minimum number of mobile sensors required to connect the left and right boundary directly is $\lceil \frac{L}{l_r} \rceil$. Therefore, $w(s,t) = c\lceil \frac{L}{l_r} \rceil$. Hence, the minimum cost of mobile sensors needed to form a barrier in homogeneous sensor network is upper bounded by $c\lceil \frac{L}{l_r} \rceil$ $\square$

Theorem 2.1 proves that the minimum cost of mobile sensors required to form a barrier is exactly the length of the shortest path from s to t on graph G, which can be found by the classical Dijkstra's algorithm (Cormen et al., 2009). The shortest path shown in Figure 4.4(a) for weak barrier coverage is $s \to wc_1 \to t$, the length of which is $0 + 0 = 0$. Therefore, the region is already weak barrier covered after initial deployment, and no mobile sensor is needed to form a weak barrier. As for strong barrier coverage, the shortest path shown in Figure 4.4(b) is $s \to sc_1 \to sc_3 \to sc_6 \to$

t, the length of which is $0+8+8+0=16$. That is, the region is not strong barrier covered after initial deployment, and at least 16$ mobile sensors are needed to form a strong barrier. There are two gaps on the shortest path: $sc_1 \rightarrow sc_3$, and $sc_3 \rightarrow sc_6$, which requires 8$, and 8$ mobile sensors, respectively.

2.4.3 Directional Barrier Coverage Algorithm

According to the analysis on the directional barrier graph, we propose a directional barrier coverage algorithm to find the minimum cost of mobile sensors needed to form a weak/strong barrier. The algorithm is formally presented in Algorithm 1.

Algorithm 1 Directional Barrier Coverage Algorithm (DBC)

Require: $S = \{s_1, s_2, \cdots, s_n\}$, L and H
Ensure: The minimum cost of mobile sensors needed

1: group sensors into weak/strong clusters
2: calculate the distance between any two clusters
3: calculate the minimum cost of mobile sensors required to connect any two clusters
4: construct the directional barrier graph G
5: find the shortest path *sp(G,s,t)* using Dijkstra's algorithm
6: **if** the length of *sp(G,s,t)* equals 0 **then**
7:     ROI is already barrier covered
8: **else**
9:     ROI is not barrier covered and the minimum cost needed is the length of *sp(G,s,t)*
10: **end if**

After sensors are deployed, they report their types, locations and facing directions to the server. The server first groups sensors into weak or strong clusters depending on the application requirement of weak or strong barrier coverage. It then calculates the minimum cost of mobile sensors required to connect any two clusters and constructs the directional barrier graph G. The shortest path can be found using the classical Dijkstra's algorithm, the length of which is the minimum cost of mobile nodes needed to form a weak/strong barrier.

Note that we still need to efficiently group sensors into weak/strong clusters and calculate the minimum cost of mobile sensors needed to connect any two clusters. In the following, we first describe the algorithms to group sensors into weak and strong clusters, respectively, and then present algorithms to find the minimum cost of mobile sensors to connect any two clusters.

Clusters Formation

We first describe how to group sensors into weak clusters when applications require weak barrier coverage, and then describe how to group sensors into strong clusters when applications require strong barrier coverage.

Based on Definition 2, sensors in a cluster are weakly connected either directly or through other sensors. Therefore, the basic idea of our algorithm to form weak clusters is described as follows. First, we calculate the coverage region of each sensor in the horizonal direction; second, we sort all the sensor in the increasing order according to their left coverage boundary; third, check each sensor one by one in the sorted set, put one sensor into a weak cluster if the sensor overlaps with the weak cluster in the horizontal direction and then update the right coverage boundary of the cluster, otherwise, initialize a new weak cluster. The third step repeats until the last sensor in the sorted set is checked. Finally, each sensor must belong to one and only one cluster. Let $C_w = \{wc_1, wc_2, \cdots\}$ denote the set of weak clusters. The pseudocode of the algorithm is presented in Algorithm 2.

Given n stationary sensors, the running time of the sorting operation in step 5 is $O(n \lg n)$. The running time of the comparison process from step 7 to the end is $O(n)$. Therefore, the running time of the weak cluster formation algorithm is $O(n \lg n)$.

We then describe the strong cluster formation algorithm that group sensors into strong clusters. We claim that each sensor belonging to a cluster as *assigned sensor*, otherwise *unassigned sensor*. The basic idea of the algorithm is described as follows: First, initialize a new cluster with an unassigned sensor; second, for each newly added sensor s_j in the cluster, check all the unassigned sensors and put all sensors

Algorithm 2 Weak Cluster Formation

Require: $S = \{s_1, s_2, \cdots, s_n\}$
Ensure: $C_w = \{wc_1, wc_2, \cdots\}$
$C_w \leftarrow \emptyset$
for $i = 1$ to n **do**
    calculate x_i^L and x_i^R for sensor s_i
end for
sort S according to x_i^L in the increasing order
initialize a queue Q, $Q \leftarrow S$, and $k \leftarrow 0$
while $Q! = \emptyset$ **do**
    $k \leftarrow k + 1$
    $s_j \leftarrow Q_{\cdot pullFirst}$
    $wc_k \leftarrow \{s_j\}$, $wc_k^L \leftarrow x_j^L$, and $wc_k^R \leftarrow x_j^R$
    for each sensor s_p in Q **do**
        if $x_p^L \leq wc_k^R$ **then**
            $pop(s_p)$
            $wc_k \leftarrow wc_k \cup \{s_p\}$, and $wc_k^R \leftarrow \max(wc_k^R, x_p^R)$
        else
            break
        end if
    end for
    $C_w \leftarrow C_w \cup \{wc_k\}$
end while

overlapping with s_j into the cluster. We call the checking process in the second step as the neighbor finding process. Repeat the neighbor finding process until no sensor can be added into the cluster, then go to the first step. Finally, the algorithm terminates when no sensor is left.

The pseudocode of the strong cluster formation algorithm is presented in Algorithm 3. Let $C_s = \{sc_1, sc_2, \cdots\}$ denote the set of strong clusters. The algorithm perform neighbor finding process for each node and at each process at most n nodes should be checked. Therefore, the running time of the strong cluster formation algorithm in the worst case is $O(n^2)$.

Algorithm 3 Strong Cluster Formation

Require: $S = \{s_1, s_2, \cdots, s_N\}$
Ensure: $C_s = \{sc_1, sc_2, \cdots\}$.

$SC \leftarrow \emptyset$, $k \leftarrow 1$
initialize a queue Q, $Q \leftarrow S$
while $Q! = \emptyset$ **do**
    $sc_k \leftarrow Q._{pullFirst}$
    while new sensors are added into sc_k **do**
        for each new added sensor s_j of sc_k **do**
            for each sensor $s_p \in Q$ **do**
                if s_j and s_p are strongly connected **then**
                    $pop(s_p)$
                    $sc_k \leftarrow sc_k \cup \{s_p\}$
                end if
            end for
        end for
    end while
    $C_s \leftarrow C_s \cup \{sc_k\}$
    $k \leftarrow k + 1$
end while

Weight Calculation

In this section, we describe how to calculate the weight of each edge on the DBG, which is the minimum cost of mobile nodes needed to connect any pair of clusters.

Suppose the distance between two clusters is $\tilde{d}$. If we want to form a weak barrier, $\tilde{d}$ denotes the *weak distance* between the two clusters. If we want to form a strong barrier, $\tilde{d}$ denotes the *strong distance* between the two clusters. Recall that the largest coverage range and cost of type j sensor are denoted by $l_r(j)$ and $c(j)$, respectively. Let $n(j)$ denote the required number of the jth type mobile nodes. Then the problem of minimizing the cost of mobile nodes needed to connect the two clusters is formulated as follows:

$$
\begin{aligned}
\text{Minimize} \quad & \sum_{j=1}^{k} c(j)n(j) \\
\text{subject to} \quad & \sum_{j=1}^{k} l_r(j)n(j) \geq \tilde{d}. \\
& n(j) \geq 0 \ \& \ n(j) \in \mathbb{Z}.
\end{aligned}
$$

The objective is to minimize the total cost of mobile nodes needed while the first constraint condition restricts that mobile nodes should fill in the gap. The formulated problem is an Integer Linear Programming (ILP) problem. Although the ILP problem is well known as NP-Hard, we can use Branch and Bound algorithm to efficiently to solve it. The weights shown in Figure 4.4 are calculated by solving the ILP problem for each edge using the Branch and Bound algorithm.

It is worth noting that calculating the weight of each edge is much easier for homogeneous sensor network where the same type of mobile sensors with the same sensing range and cost are used to fill in gaps.

Homogeneous Sensor Network: Since mobile sensors have the same sensing range and cost, minimizing the cost of mobile nodes needed means minimizing the number of mobile nodes needed. Let l_r and c denote the largest coverage range and the cost of this type of mobile sensors. The minimum number of mobile nodes needed is therefore

$$
n_m(v_i, v_j) = \begin{cases} \lceil \frac{d_w(v_i,v_j)}{l_r} \rceil & \text{weak barrier covreage} \\ \lceil \frac{d_s(v_i,v_j)}{l_r} \rceil & \text{strong barrier coverage} \end{cases} \tag{2.1}
$$

where $d_w(v_i, v_j)$ denote the *weak distance* between two vertices v_i and v_j for weak barrier coverage, and $d_s(v_i, v_j)$ denote the *strong distance* between two vertices v_i and v_j for strong barrier coverage. The weight of the edge $w(v_i, v_j)$ is therefore $cn_m(v_i, v_j)$.

For *weak distance* $d_w(v_i, v_j)$, when none of v_i and v_j is a boundary, $d_w(v_i, v_j) = v_j^L - v_i^R$ given the assumption that $v_j^L > v_i^R$. While if v_i is a boundary, $d_w(v_i, v_j) = 0$ if v_j intersects the boundary; otherwise, $d_w(v_i, v_j) = v_j^L$ if v_i is the left boundary s and

$d_w(v_i, v_j) = L - v_j^R$ if v_i is the right boundary t. When v_i is the left boundary s and v_j is the right boundary t, $d_w(v_i, v_j) = L$. x_i^L and x_i^R is the left coverage boundary and the right coverage boundary of v_i in the horizontal direction.

For *strong distance* $d_s(v_i, v_j)$, when none of v_i and v_j is a boundary, $d_s(v_i, v_j) = \min(d(p_i, p_j))$ where p_i and p_j are points on v_i and v_j, respectively, and $d(p_i, p_j)$ is the Euclidean distance between p_i and p_j. When any of v_i and v_j is a boundary, $d_s(v_i, v_j) = d_w(v_i, v_j)$.

2.5 Minimum Cost Barrier Formation (MCBF) Problem

By using the DBC algorithm, we can obtain the minimum cost of mobile nodes needed to form a barrier. We then deploy the required number of each type of mobile sensor to the network. Since mobile sensors are also randomly deployed, they should move to fill in the gaps on the shortest path to form a barrier. The objective of the MCBF problem is to minimize the total moving distance of mobile sensors to form a barrier.

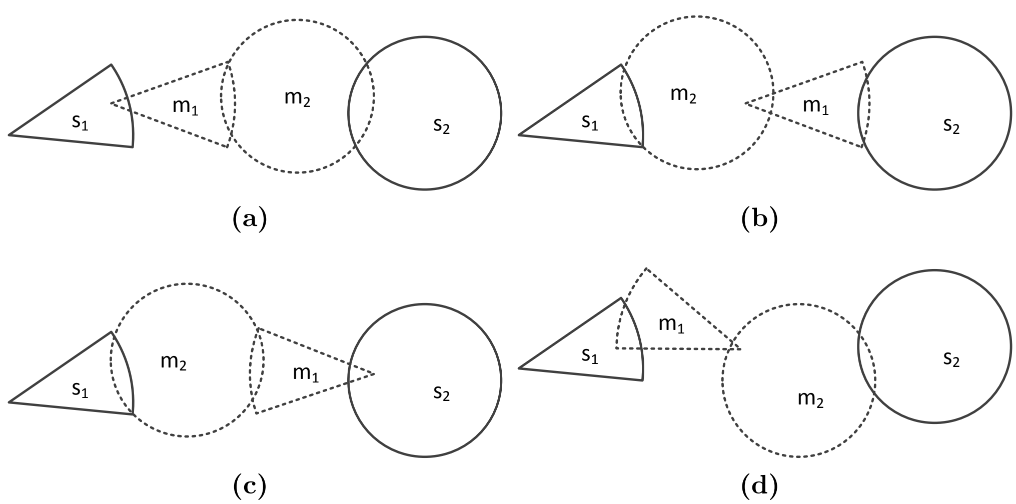


Figure 2.6: Illustration of the complexity of deploying mobile sensors to fill in a gap

It is worth noting that the formulated MCBF problem is difficult to solve. The first reason is that there are too many ways for mobile sensors to fill in a gap. As shown in Figure 2.6, all these four deployments can successfully connect s_1 and s_2. Actually there are numerous ways of deployment and it is impossible to decide which one is the optimal deployment. Second, even we know which position should be deployed with a mobile sensor, it is challenging to schedule which sensor to which position while the total moving distance is minimized. To the best of our knowledge, there is no optimal solution to the MCBF problem. To solve this problem, we propose a greedy movement algorithm that assigns mobile sensors one by one to fill in gaps for heterogeneous sensor networks. As for homogeneous sensor networks, since mobile nodes have the same sensing range, the order of assignment does not matter. Therefore, we propose a position based optimal movement algorithm to schedule mobile sensors. In the following, we first present the greedy movement algorithm and then describe the position based optimal movement algorithm.

2.5.1 Greedy Movement Algorithm

Since weak barrier coverage is a special case of strong barrier coverage, we use the latter as an example to explain how the greedy movement algorithm works. Suppose $e(v_a, v_b)$ is an edge on the shortest path of the directional barrier graph. The gap between v_a and v_b, denoted by $g(v_a, v_b)$, should be filled in by mobile sensors. Let p_a and p_b be the closet pair of points on v_a and v_b, respectively, so the strong distance $d_s(v_a, v_b) = d(p_a, p_b)$. If the line segment $p_a p_b$ is covered by mobile sensors, the gap will be filled in. Note that there are too many ways to deploy mobile sensors to fill in a gap. Our deployment provides one of the easier ways to calculate the target locations.

The basic idea of the greedy movement algorithm is to greedily move the required type of mobile sensor that is closet to the gap to the line segment $p_a p_b$ until it is fully covered. In order to measure the distance from a sensor to the gap, at each time a target location is given for the required type of sensors. For the line segment $p_a p_b$, the starting target location is p_a. Once a mobile sensor move to $p_a p_b$, the target location will be updated to the leftmost point on $p_a p_b$ that has not been covered yet. Hence, we can calculate the distance between required types of sensors and the current target location and find the closet one to move. The algorithm is formally presented in Algorithm 4.

Algorithm 4 Greedy Movement Algorithm

1: **for** each gap on the shortest path **do**
2: move the required type of sensor that is closet to the gap to its line segment until it is fully covered
3: **end for**

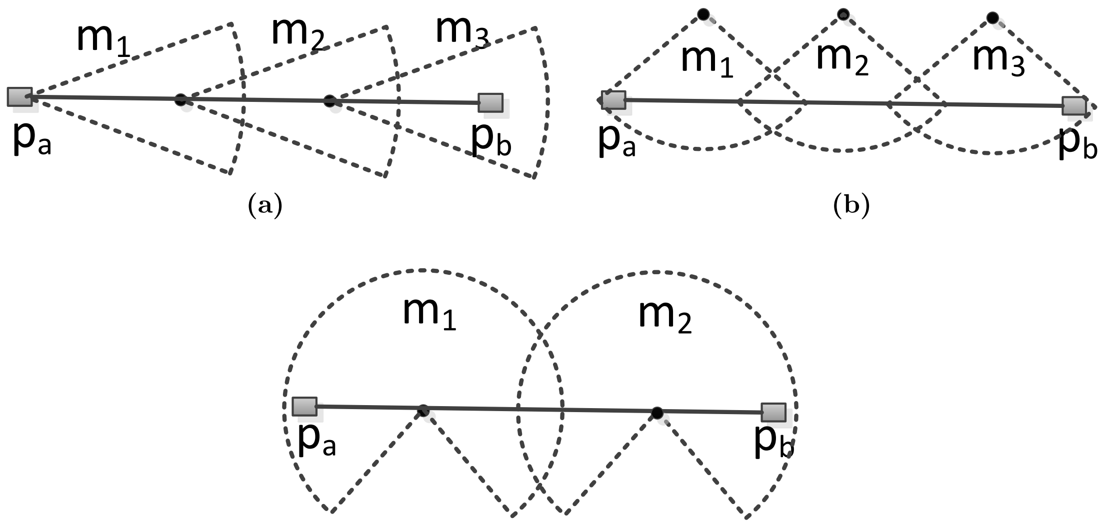
2.5.2 Position based Optimal Movement Algorithm

Although the greedy movement algorithm also works for homogenous sensor networks, since sensors have the same sensing range and angel, we propose a more efficient position based optimal movement algorithm that reduces the total moving distance. We divide the problem into two subproblems. First, we calculate the target locations for mobile sensors to fill in gaps in some kind of deployment. Second, we optimally schedule mobile sensors to the calculated target locations with the minimum total moving distance. In the following, we will describe how to solve these two subproblems to yield a suboptimal solution to the MCBF problem for homogeneous sensor networks.

For two strong clusters v_a and v_b, the closest pair of points are $p_a = (x_a, y_a)$ on v_a and $p_b = (x_b, y_b)$ on p_b. Thus, the minimum distance between v_a and v_b is

$$d_s(v_a, v_b) = d(p_a, p_b) = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2}$$

Then the minimum number of mobile sensors to fill in the gap is $n(v_a, v_b) = \lceil \frac{d(p_a, p_b)}{l_r} \rceil$. We distribute mobile sensors evenly with the longest line of the sensing



(a) (b) (c)

Figure 2.7: Illustration of different deployment methods for: (a) $l_r = r$ when $2\alpha < \pi$; (b) $l_r = 2r \sin\alpha$ when $2\alpha < \pi$; (c) $l_r = 2r$ when $2\alpha \geq \pi$.

sector along the line segment $p_a p_b$. Therefore, the interval between two mobile sensors is $d_v = \frac{d(p_a,p_b)}{n(v_a,v_b)}$. As mentioned in Section **??**, the longest line could either be the radius or the longest chord when $0 \leq 2\alpha < \pi$, or $2r$ when $\pi \leq 2\alpha \leq 2\pi$. Corresponding to these three cases, we have three deployment strategies, as shown in Figure 2.7.

Let φ denote the direction of $\overrightarrow{p_a p_b}$. Let h denote the height from the center to the longest chord of a sector. Suppose the target locations are $t_i = (t_i^x, t_i^y, t_i^o)$ for $i = 1, 2, \cdots, w(sc_1, sc_2)$, where t_i^x and t_i^y are the x-coordinate and y-coordinate of the target location t_i, and t_i^o is the facing direction of the mobile sensor on t_i.

As show in Figure 2.7(a), when $l_r = r$ and the sensing angle $2\alpha < \pi$, mobile sensors are evenly deployed with the radius along the facing direction on the line segment. Therefore, the target locations are calculated as follows:

$$t_i^x = x_a + (i-1)d_v \cos\varphi$$
$$t_i^y = y_a + (i-1)d_v \sin\varphi$$
$$t_i^o = \varphi$$

As show in Figure 2.7(b), when $l_r = 2r\sin\alpha$ and the sensing angle $2\alpha < \pi$, mobile sensors are evenly deployed with the longest chord on the line segment. The target locations are calculated as follows:

$$t_i^x = x_a + (i-1)d_v \cos\varphi + \bar{l}\cos(\varphi + \lambda)$$
$$t_i^y = y_a + (i-1)d_v \sin\varphi + \bar{l}\sin(\varphi + \lambda)$$
$$t_i^o = (\varphi + 3\pi/2) \mod 2\pi$$

where $\bar{l} = \sqrt{h^2 + (d_v/2)^2}$, $\lambda = \arctan(2h/d_v)$.

Finally, as shown in Figure 2.7(c), when $l_r = 2r$ and $2\alpha \geq \pi$, mobile sensors are evenly deployed with the diameter on the line segment. Therefore, the target locations are calculated as follows:

$$t_i^x = x_a + (i-0.5)d_v \cos\varphi$$
$$t_i^y = y_a + (i-0.5)d_v \sin\varphi$$
$$t_i^o = (\varphi + \pi/2) \mod 2\pi$$

Suppose the number of mobile sensors required corresponding to the minimum cost is γ. Then there are γ target locations for γ mobile sensors to move to. Denote the set of target locations by $T = \{t_1, t_2, \cdots, t_\gamma\}$. In order to form a barrier, γ mobile sensors are deployed. Let δ_{ij} denote a decision variable, where $\delta_{ij} = 1$ if mobile sensor

m_i is assigned to target location t_j, $\delta_{ij} = 0$ otherwise. d_{ij} is the distance for mobile sensor m_i to move to target location t_j. Then the MCBF problem can be formulated as how to assign γ mobile sensors to γ target locations while minimizing the total moving distance.

$$
\begin{aligned}
\text{Minimize} \quad & \sum_{i=1}^{\gamma}\sum_{j=1}^{\gamma} d_{ij}\delta_{ij} \\
\text{subject to} \quad & \sum_{i} \delta_{ij} = 1, \forall j = 1, 2, \cdots, \gamma \\
& \sum_{j} \delta_{ij} \leq 1, \forall i = 1, 2, \cdots, \gamma \\
& \delta_{ij} = 0 \text{ or } 1, i = 1, 2, \cdots, \gamma; j = 1, 2, \cdots, \gamma
\end{aligned}
$$

The objective function is to minimize the total moving distance. The first constraint restricts that any target location must be assigned with one and only one mobile sensor. The second constraint restricts that each mobile sensor can be assigned to at most one target location.

Provided with a set of target locations and a set of mobile sensors, the formulated problem indeed is a minimum cost bipartite assignment problem. The Hungarian algorithm (Kuhn, 1955; Lawler, 1976) provides the optimal solution to this problem and its complexity is proved to be $O(\mu^2\tau)$. Note that Ban et al. (2010) studied a similar problem and also used the Hungarian algorithm to solve it. Please refer to (Kuhn, 1955; Lawler, 1976) for the details of the Hungarian algorithm.

2.6 Performance Evaluation

In this section, we conduct simulations to evaluate the performance of the proposed algorithms. The ROI is a belt region of length $L = 1000m$ and width $H = 100m$. Stationary sensors are randomly deployed in the belt region. After the minimum cost of mobile nodes is calculated, mobile sensors are then randomly deployed to form a barrier. The evaluation mainly focuses on the following performance metrics:

- The cost of mobile nodes needed to form a barrier.
- The total moving distance for mobile sensors to form a barrier.
- The total cost of sensors including the cost of stationary and mobile sensors.
- The probability that the sensor network is already barrier covered after initial random deployment, which is calculated by the ratio of the number of experiments that sensor network is barrier covered to the total number of experiments performed.

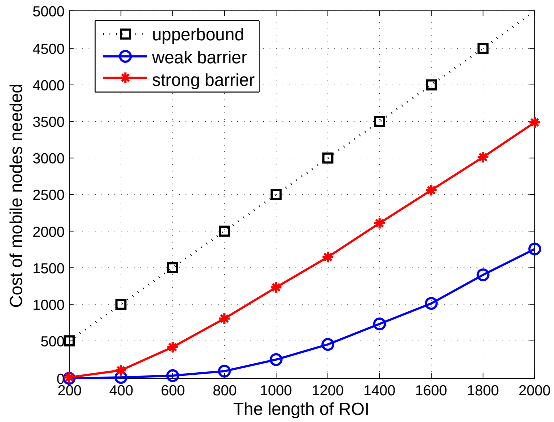
Evaluation of these performance metrics is conducted on different parameters, such as the length of the ROI (L), the number of stationary sensors (n), sensing range (r) and sensing angle (2α). For all the simulation results presented in this chapter, each data point is an average of 100 experiments. Both weak barrier coverage and strong barrier coverage are evaluated. We first present the evaluation results for homogeneous sensor networks and then for heterogenous sensor networks.

2.6.1 Evaluation for Homogeneous Sensor Network

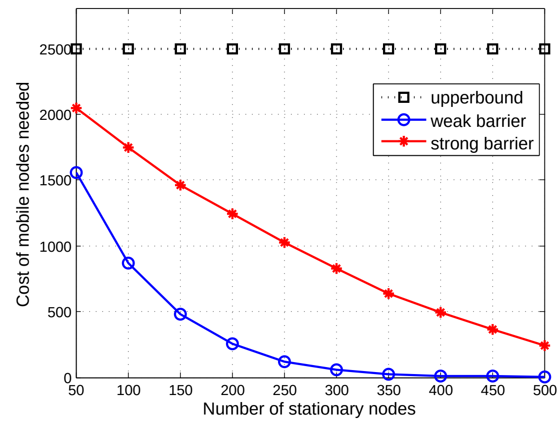
The cost of mobile nodes needed

Figure 2.8 shows the effects of different parameters on the cost of mobile nodes needed. The default setting of parameters are $L = 1000m$, $n = 200$, $r = 20m$ and $2\alpha = \pi/3$. Let c_s and c_m denote the cost of a stationary sensor and a mobile sensor, respectively. We assume $c_s = 10\$$ and $c_m = 50\$$. As shown in Figure 2.8, the blue line with markers of circles represents the cost of mobile nodes needed to form a weak barrier; the red line with markers of stars represents the cost of mobile nodes needed to form a strong barrier; and the black dashed line with markers of squares represent the upper bound of cost of mobile nodes needed to form a weak/strong barrier, which is calculated from Theorem 2.3.

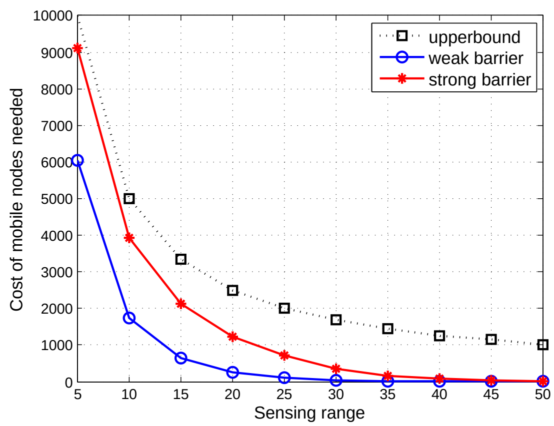
First we can observe that, no matter how the parameters change, the cost of forming a weak barrier is always smaller than that of forming a strong barrier, and



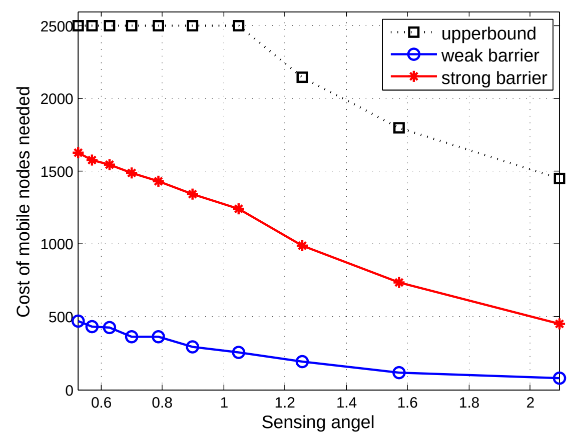
(a) L changes



(b) n changes



(c) r changes

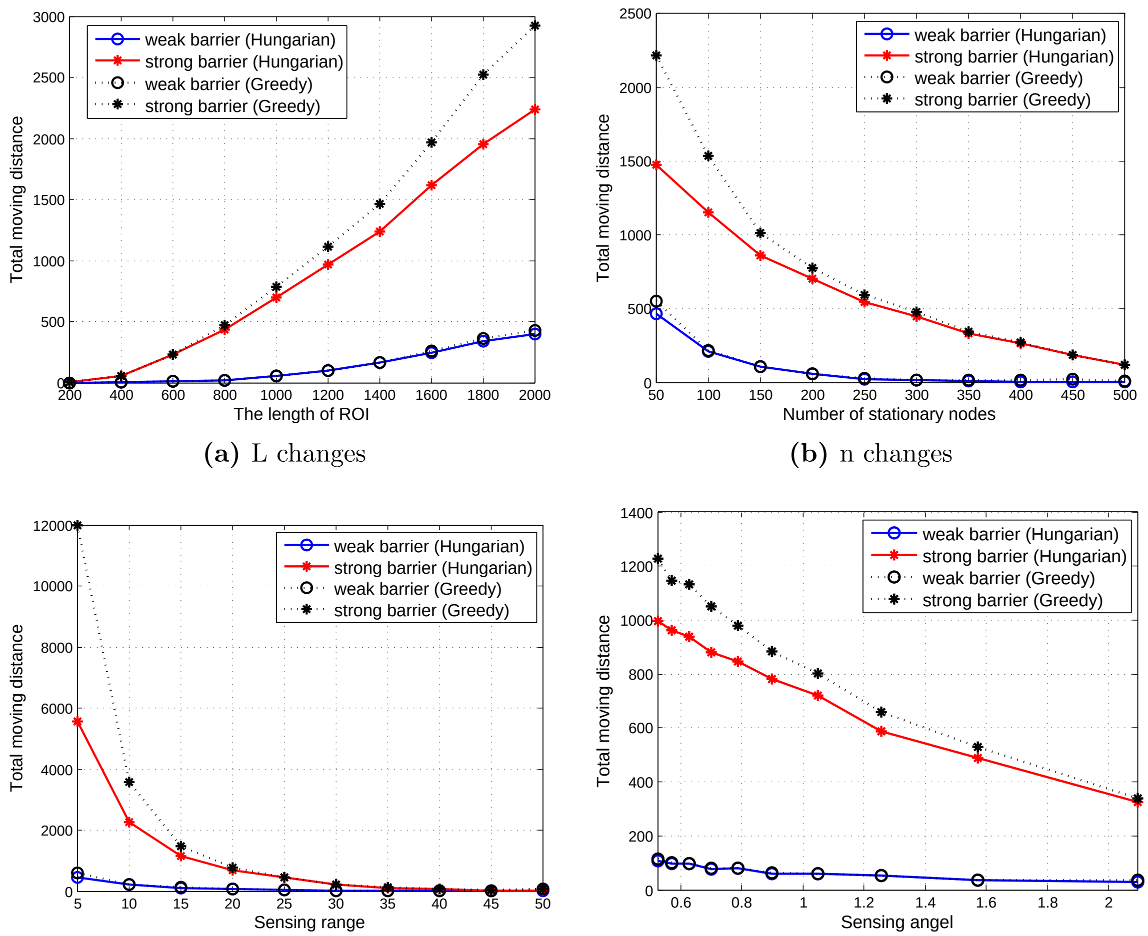


(d) 2α changes

Figure 2.8: The effects of different parameters on the cost of mobile sensors needed to form a barrier. Default setting of parameters are $L = 1000m$, $n = 200$, $r = 20m$ and $2\alpha = \pi/3$.

both of them are smaller than the upper bound of the cost. From Figure 2.8(a), we can observe that the cost of mobile sensors needed increases as the length of the ROI increase. This is because increasing the length increases the number of gaps and their sizes, which requires more number/cost of mobile sensors to form a barrier. As shown in Figure 2.8(b), the cost of mobile sensors needed decreases as the number of stationary sensors increase, which is because more stationary sensors reduces the number of gaps and their sizes. Note that the upper bound of cost does not change in this case. As shown in Figure 2.8(c) and (d), the cost of mobile sensors needed decreases when the sensing range or the sensing angle increases. The reason is that larger sensing range or sensing angle increases the probability of forming larger clusters, which then results in fewer gaps and smaller sizes of gaps on the shortest path. Note that the upper bound of cost does not change when the $2\alpha \leq \pi/3$, which is because the longest coverage range is always r when $2\alpha \leq \pi/3$ and then increases as 2α increases.

The total moving distance



(a) L changes **(b)** n changes

(c) r changes **(d)** 2α changes

Figure 2.9: The effects of different parameters on the total moving distance of mobile sensors.

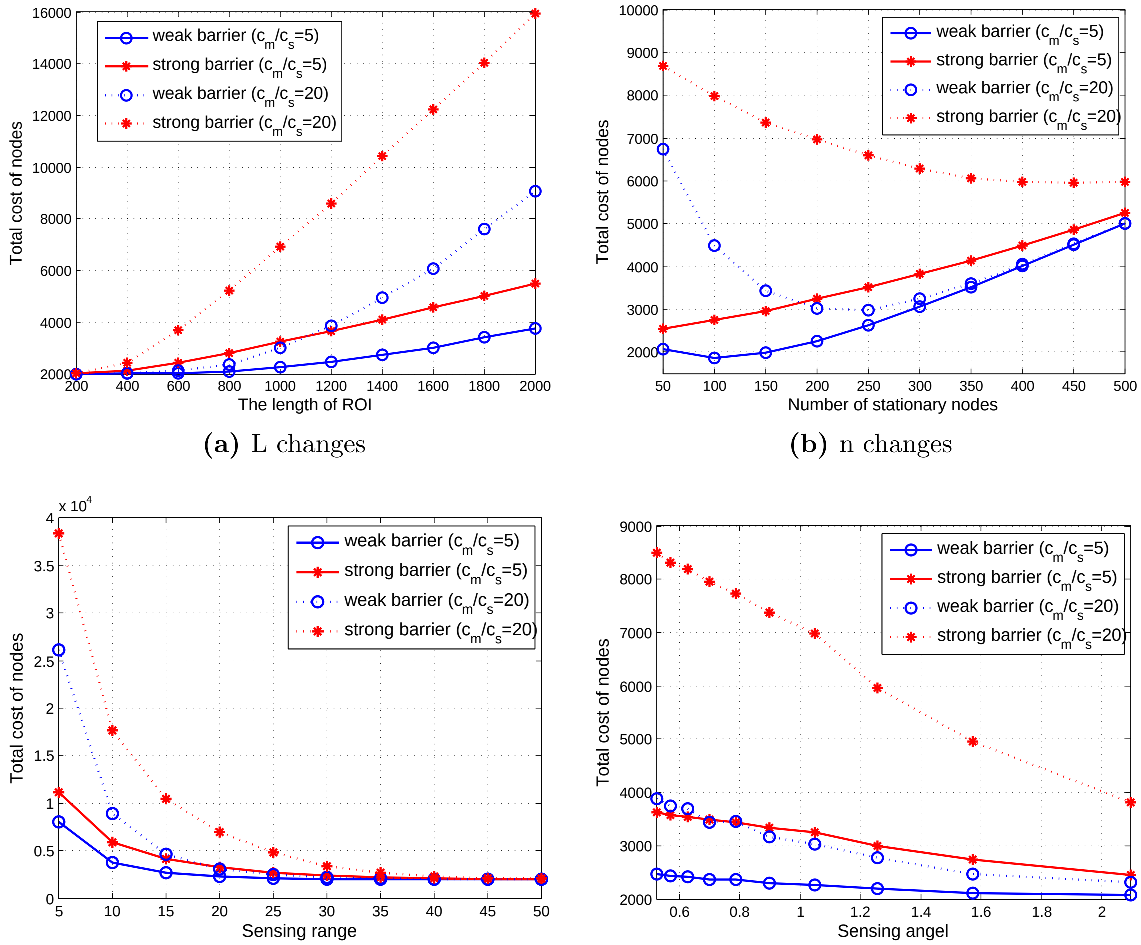
Figure 2.9 shows the effects of different parameters on the total moving distance. We compare the performance of the greedy movement algorithm and the position based optimal movement algorithm. As shown in Figure 2.9, the real lines are drawn from the position based optimal movement algorithm, and the dashed lines are drawn from the greedy movement algorithm. We can see that in all cases the total moving distance by using the position based optimal movement algorithm is always smaller than that by using the greedy movement algorithm. Meanwhile, the total moving distance needed to form a strong barrier is always larger than that to form a weak barrier. The first reason is that, as shown in Figure 2.8, forming a weak barrier requires less mobile sensors. The second reason is that mobile sensors only need to move in the horizontal direction in order to form a weak barrier.

From Figure 2.9(a), we can observe that the total moving distance increases when the length of the ROI increases, which is mainly because more mobile nodes are needed for larger length of the ROI. When the ROI is fixed, increasing the number of stationary sensors, sensing range or sensing angle could decrease the total moving distance, since increasing any of them can increase the probability of forming larger clusters, which then results in fewer gaps and smaller sizes of gaps on the shortest path.

The total cost

The total cost needed to form a barrier is the sum of the cost of deployed stationary sensors and the cost of mobile sensors needed. We still assume $c_s = 10\$$ for each stationary sensor, and compare the total cost when $c_m/c_s = 5$ and $c_m/c_s = 20$.

Figure 2.10 shows the effects of different parameters on the total cost. The solid and dashed red lines with markers of stars represent the total cost to form a strong barrier when $c_m/c_s = 20$ and $c_m/c_s = 5$, respectively. The solid and dashed blue lines with markers of circles represent the total cost to form a weak barrier when $c_m/c_s = 20$ and $c_m/c_s = 5$, respectively. We can see that the total cost to form



(a) L changes

(b) n changes

(c) r changes

(d) 2α changes

Figure 2.10: The effects of different parameters on the total cost.

a strong is larger than that to form a weak barrier, and also the total cost when $c_m/c_s = 20$ is obviously larger than $c_m/c_s = 5$.

From Figure 2.10(a), we can observe that the total cost increases when the length of the ROI increases, which is mainly because more mobile nodes are needed for larger length of the ROI. When the ROI is fixed, as shown in Figure 2.10(c) and (d), increasing the sensing range or sensing angle decreases the total cost since less mobile nodes are needed. It is worth noting that, as shown in Figure 2.10(b), the total cost does not always increase or decrease when the number of stationary sensors increases. Take the total cost to form a weak barrier when $c_m/c_s = 20$ as an example.

The total cost reaches the minimum when the number of stationary sensors reaches 200. Since $c_m/c_s = 20$, the cost of one mobile sensor equals that of 20 stationary sensors. When the number of stationary sensors is small, a lot of mobile sensors are needed, and therefore the cost of mobile sensors dominates the total cost. Hence, the total cost decreases as the needed cost of mobile sensors decreases when the number of stationary sensors increases from 50 to 200. When the number of stationary sensors is larger than 200, the required number/cost of mobile sensors does not change too much. At this time, the total cost increases as the deployed number of stationary sensors increases. Therefore, we can conclude that, given an ROI, the deployed number of stationary sensors to be deployed highly depends on c_m/c_s.

Probability of barrier coverage

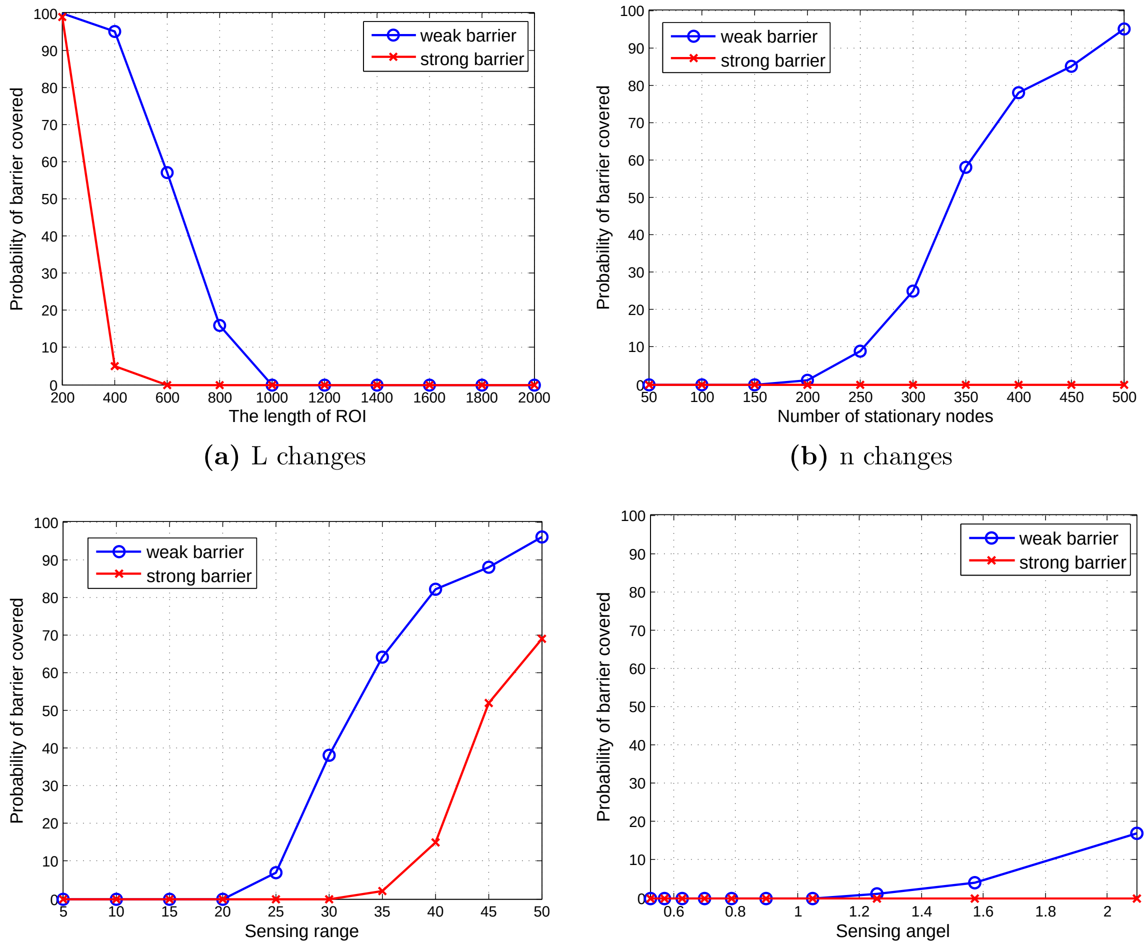
Figure 2.11 shows the effects of different parameters on the probability of barrier covered after initial random deployment. We can that it is always easier to form a weak barrier than to form a strong barrier. The probability decreases when the length of the ROI increases. When the region is fixed, the probability increases when the number of stationary sensors, the sensing range, or the sensing angle increase.

2.6.2 Evaluation for Heterogeneous Sensor Networks

In this subsection, we evaluate the performance of proposed algorithm for heterogeneous sensor networks. We consider three types of sensors where their sensing regions are denoted by $T(1) = (10, \pi, 10\$)$, $T(2) = (25, \pi/6, 12\$)$ and $T(3) = (22, \pi/4, 15\$)$. We assume $c_m/c_s = 5$ for each type of sensors. Evaluation is conducted on the length of the ROI and the number of stationary sensors.

Performance vs. the length of the ROI

Figure 2.12 shows the performance results when the length of the ROI changes. As shown in Figure 2.12(a)-(d), these four performance metrics for heterogenous sensor



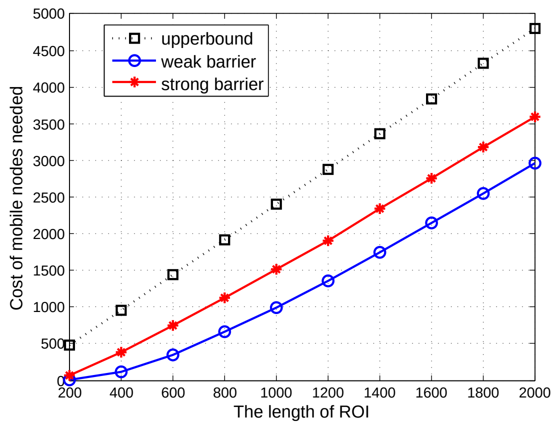
(a) L changes **(b)** n changes

(c) r changes **(d)** 2α changes

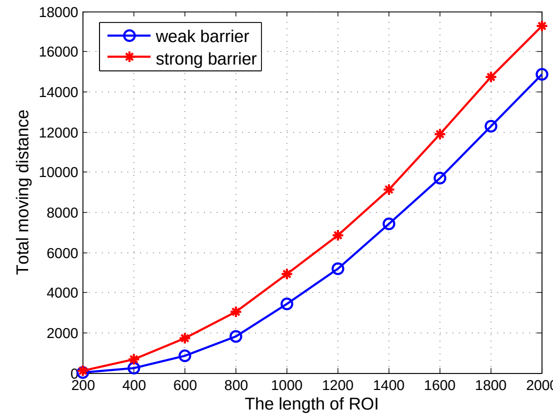
Figure 2.11: The effects of different parameters on the probability of barrier covered after initial random deployment.

networks show similar trend as that for homogeneous sensor networks. As the length of the ROI increases, the cost of mobile sensors needed, the total moving distance, and the total cost are all increasing, but the probability of barrier covered after initial random deployment is decreasing.

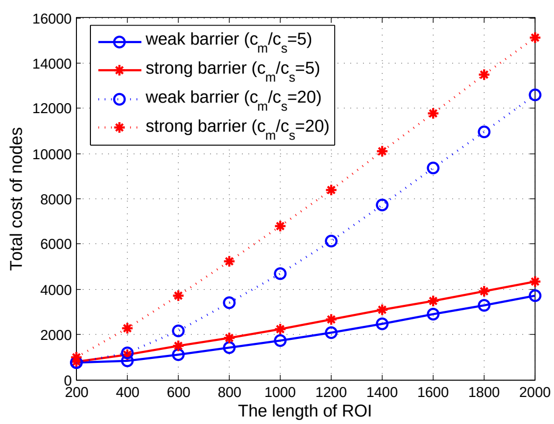
Figure 2.12(e) and (f) show the comparison of cost using only one type of mobile sensors and using multiple types of mobile sensors. Each dashed line represent the cost using only one type of mobile sensors, and the real line represent the cost using all the three types of mobile sensors. Figure 2.12(e) and (f) show the costs needed to



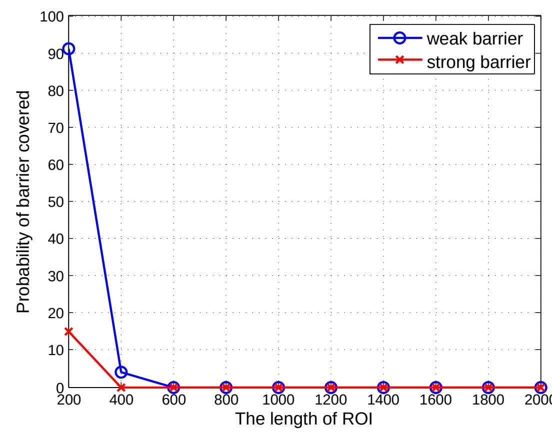
(a) Cost of mobile sensors



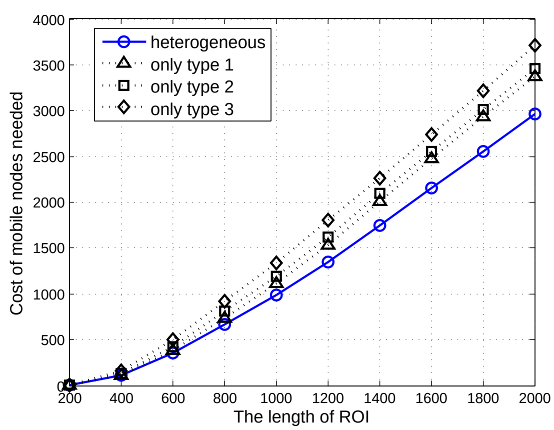
(b) Total moving distance



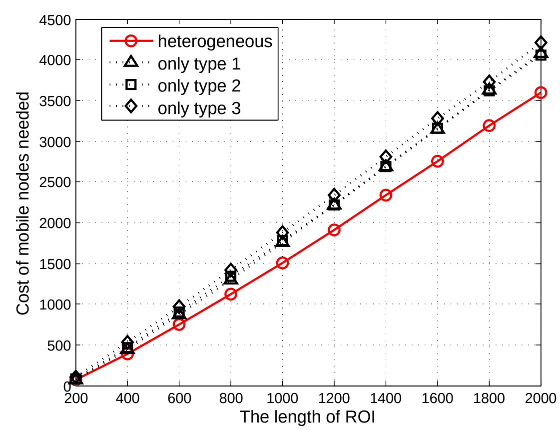
(c) Total cost



(d) Probability



(e) Cost comparison (weak)



(f) Cost comparison (strong)

Figure 2.12: Performance results vs. the length of the ROI

form a weak barrier and a strong barrier, respectively. We can see that a combination of sensors can reduce the cost needed to form a weak/strong barrier compared only one type of sensors, which motivates the deployment of multiple types of sensors from the cost efficiency perspective.

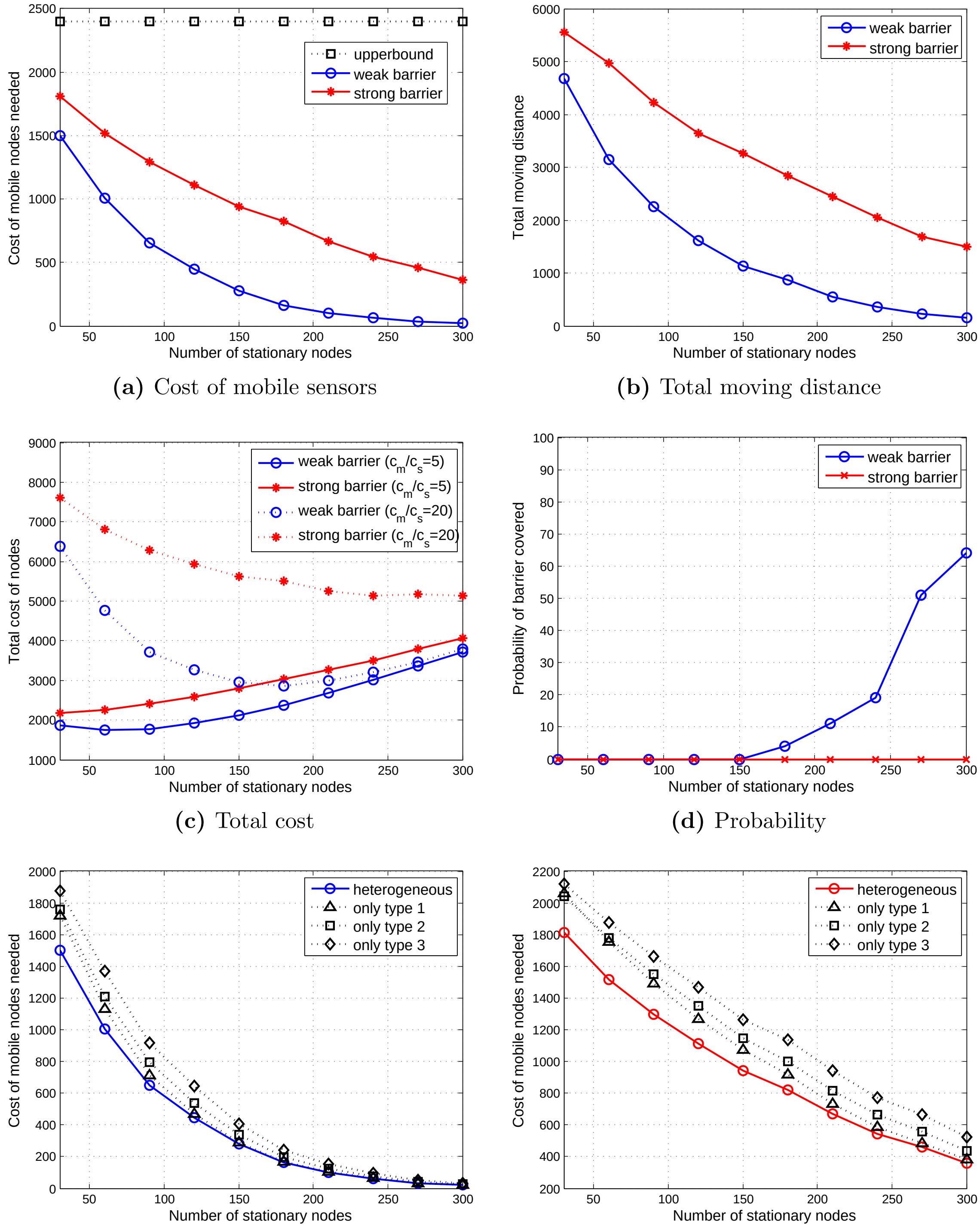
Performance vs. the number of stationary sensors

Figure 2.13 shows the performance results when the number of stationary sensors changes. As shown in Figure 2.12(a)-(d), these four performance metrics for heterogenous sensor networks show similar trend as that for homogeneous sensor networks. As the number of stationary sensors increases, the cost of mobile sensors needed and the total moving distance are decreasing, but the probability of barrier covered after initial random deployment is increasing. The variation of the total cost of sensors depends on the value of c_m/c_s.

Figure 2.13(e) and (f) show the comparison of cost using only one type of mobile sensors and using multiple types of mobile sensors. Each dashed line represent the cost using only one type of mobile sensors, and the real line represent the cost using all the three types of mobile sensors. Figure 2.13(e) and (f) show the costs needed to form a weak barrier and a strong barrier, respectively. We can observe the similar trend as that in Figure 2.12 that a combination of sensors can reduce the cost needed to form a weak/strong barrier compared only one type of sensors, which shows the advantage of using multiple types of sensors.

2.7 Summary

In this chapter, we studied the barrier coverage formation problem for heterogenous sensor networks and explored how to efficiently form a barrier by using mobile sensors to fill in the gaps between stationary sensors. We introduced the directional barrier graph model, and proved that the minimum cost of mobile sensors required to form a barrier is the length of the shortest path from the source node to the destination



(a) Cost of mobile sensors

(b) Total moving distance

(c) Total cost

(d) Probability

(e) Cost comparison (weak)

(f) Cost comparison (strong)

Figure 2.13: Performance results vs. the number of stationary sensors

node on the graph. To efficiently assign mobile sensors to different gaps while minimizing the total moving distance, we proposed a greedy movement algorithm for heterogenous sensor network, and also a position based optimal movement algorithm for homogeneous sensor network which formulated the MCBF problem as the minimum cost bipartite assignment problem and solved it using the Hungarian algorithm. Extensive evaluation results on both homogeneous and heterogeneous sensor networks validate the effectiveness of our proposed algorithms.

Chapter 3

Cost-Effective k-barrier Coverage Formation in Sensor Networks

In this chapter, we further study how to efficiently use mobile sensors to form k-barrier coverage in hybrid sensor networks. In particular, two problems are studied under two scenarios. First, when only the stationary sensors have been deployed, what is the minimum number of mobile sensors required to form k-barrier coverage? Second, when both the stationary and mobile sensors have been pre-deployed, what is the maximum number of barriers that could be formed? To solve these problems, we introduce a novel concept of weighted barrier graph (WBG) and prove that determining the minimum number of mobile sensors required to form k-barrier coverage is related with but not equal to finding k vertex-disjoint paths with the minimum total length on the WBG. With this observation, we propose an optimal solution and a greedy solution for each of the two problems. Both analytical and experimental studies demonstrate the effectiveness of the proposed algorithms.

3.1 Introduction

k-barrier coverage was first introduced in (Kumar et al., 2005). A sensor network provides k-barrier coverage for a ROI if all crossing paths through the region is k-covered and a crossing path is said to be k-covered if it can be covered by at least k distinct sensors. Kumar et al. (2005) proved that a network provides k-barrier coverage iff there exists k sensor-disjoint barriers in the ROI. The term of sensor-disjoint barriers means that none of any two barriers have sensors in common.

$k(k \geq 1)$-barrier coverage is needed due to the following reasons. First, binary sensing model is ideal for sensors. However, in practice, false alarms and detection failures happens frequently for intruder detection. Consequently, multiple barriers are needed to increase the intruder detection probability. Second, sensors are prone to failures due to lack of power, physical damage and environment interference. Providing multiple barriers could significant increase the fault tolerance capacity of the sensor network. Finally, if the number of barriers formed is larger than the required number of barriers, we can apply sleep-wakeup protocols to prolong the network lifetime.

In order to form k-barrier coverage for the ROI, mobile sensors should move to fill in gaps between stationary sensors. Compared with 1-barrier coverage formation, k-barrier coverage formation is more challenging because stationary sensors need to contribute to distinct barriers. Due to the expensive cost of mobile sensors, we should use as few mobile sensors as possible to form k barriers. In particular, we focus on the following problems:

1. ***Min-Num-Mobile(k) problem***: Given a ROI and a deployed sensor network with only stationary sensors, does the network provides k-barrier coverage for the ROI? If not, what is the minimum number of mobile sensors required to form k-barrier coverage with the pre-deployed stationary sensors?

2. ***Max-Num-Barrier problem***: Given a ROI and a deployed sensor network with both stationary and mobile sensors, what is the maximum number of barriers that could be formed?

In this chapter, we systematically address the *Min-Num-Mobile(k) problem* and the *Max-Num-Barrier problem*, and the main contributions are summarized as follows:

- To the best of our knowledge, we are the first to study how to efficiently use mobile sensors to form k-barrier coverage with pre-deployed stationary sensors.
- We introduce a weighted barrier graph model for barrier coverage formation problem. We prove that determining the minimum number of mobile sensors required to form k-barrier coverage is related with finding k vertex(sensor)-disjoint * paths with the minimum total length on the weighted barrier graph. We also prove that the ROI is k-barrier covered after initial deployment if and only if there exists k vertex-disjoint paths with length of 0 on the weighted barrier graph.
- We propose an efficient optimal solution and an efficient greedy solution for the *Min-Num-Mobile(k) problem*. We also propose an efficient optimal solution and an efficient greedy solution for the *Max-Num-Barrier problem*.
- We conduct extensive simulations to evaluate the performance of the proposed algorithms. Experimental results validate the effectiveness of our algorithms.

The remainder of this chapter is organized as follows. The system model is presented in Section 3.2. We introduce the weighted barrier graph and present theoretical analysis of the barrier coverage problem for directional sensor networks in Section 3.3. We present the optimal solution and the greedy solution for the *Min-Num-Mobile(k) problem* in Section 3.4. We present the optimal solution and the

*Without confusion, we interchangeably use *vertex-disjoint* and *sensor-disjoint* throughout this chapter.

greedy solution for the *Max-Num-Barrier problem* in Section 3.5. The performance evaluation of our algorithms is presented in Section 3.6. Finally, we conclude this chapter in Section 3.7.

3.2 System Model and Preliminaries

We assume that the ROI is a two-dimensional rectangular belt area with the length of L and the width of H. For the *Min-Num-Mobile(k) problem*, n stationary sensors are randomly deployed in the belt region. For the *Max-Num-Barrier problem*, n stationary sensors and τ mobile sensors are randomly deployed in the belt region. We assume that they are the same type of sensors except that mobile sensors have the ability to move. Let $S = \{s_1, s_2, \cdots, s_n\}$ denote the set of stationary sensors where s_i denote the directional sensor i.

We still adopt the more general sector model shown in Figure 2.2 instead of 0-1 disk model for sensors' sensing model. The sensor s_i can be represented by a 5-tuple $< x_i, y_i, r, \alpha, \beta_i >$, where $l_i = (x_i, y_i)$ is the two-dimensional location of the center of sensor i, r is the sensing range and α is half of the sensing angle of a sensor. We assume that each sensor has the identical sensing range and sensing angle. According to the ground truth data in (Guvensan and Yavuz, 2011), the sensing angle of directional sensors, 2α, is usually less than π. β_i is the orientation or the facing direction of sensor i. We assume that β_i is uniformly distributed in $[0, 2\pi)$, e.g., $\beta_i \sim U(0, 2\pi)$. Note that the omni-directional sensing model is a special case of the directional sensing model when $2\alpha = 2\pi$.

The largest coverage range of a directional sensor, denoted by l_r, is the length of the longest line in its sensing sector. Since the longest line is either the sensing radius or the longest chord of the sector, we have

$$l_r = \begin{cases} \max\{r, 2r\sin\alpha\} & 0 \leq \alpha < \frac{\pi}{2}, \\ 2r & \frac{\pi}{2} \leq \alpha \leq \pi. \end{cases} \tag{3.1}$$

Table 3.1 summarizes the notations used in the paper.

Table 3.1: Summary of notations

Symbol	Description
L	the length of the belt region
H	the width of the belt region
n	the number of stationary sensors deployed
τ	the number of mobile sensors deployed for the *Max-Num-Barrier problem*
s_i	the ith stationary sensor
l_i	$l_i = (x_i, y_i)$ the location of s_i
r	the sensing range
α	half of the sensing angle
β_i	the facing direction of s_i
l_r	the largest coverage range of each sensor
G	the weighted barrier graph $G = (V, E, W)$
P_q^*	the set of q vertex-disjoint paths with the minimum total length on G
P_q^k	the k-auxiliary set of P_q^*, which is composed of P_q^* and $k-q$ direct paths
$\hat{P}_k$	the optimal set of k sensor-disjoint barriers to the *Min-Num-Mobile(k) problem*
N_m	the minimum number of mobile sensors required for the *Min-Num-Mobile(k) problem*
N_b	the maximum number of barriers for the *Max-Num-Barrier problem*

3.3 Problem Formulation and Analysis

In this section, we introduce a novel graph model, weighted barrier graph, for the barrier coverage formation problem, and present theoretical analysis to find the minimum number of mobile sensors needed to form k-barrier coverage based on the weighted barrier graph.

Definition 5. *A* **weighted barrier graph** $G = (V, E, W)$ *of a sensor network is constructed as follows. The set V consists of vertices corresponding to the left boundary (s), all the stationary sensors (S) and the right boundary (t) of the belt region, that is, $V = \{v_1, v_2, \cdots, v_{n+2}\} = \{s \cup S \cup t\}$. $E = \{e(v_i, v_j)\}$ is the set of edges between any pair of vertices. $W : E \to \mathbb{R}$ is the set of weights of each edge,* *where the weight $w(v_i, v_j)$ of edge $e(v_i, v_j)$ is the minimum number of mobile sensors needed to connect v_i and v_j.*

To calculate the minimum number of mobile sensors needed to connect vertices v_i and v_j, the distance between two vertices must be calculated first. Therefore, we further give the following definitions.

Definition 6. *Let $d_w(v_i, v_j)$ denote the* **weak distance** *between vertices v_i and v_j. $d_w(v_i, v_j) = 0$ if v_i and v_j overlap in the horizontal direction; otherwise, $d_w(v_i, v_j) = x_j^L - x_i^R$ given the assumption that $x_j^L > x_i^R$, where x_j^L is the left coverage boundary of v_j and x_i^R is the right coverage boundary of v_i in the horizontal direction.*

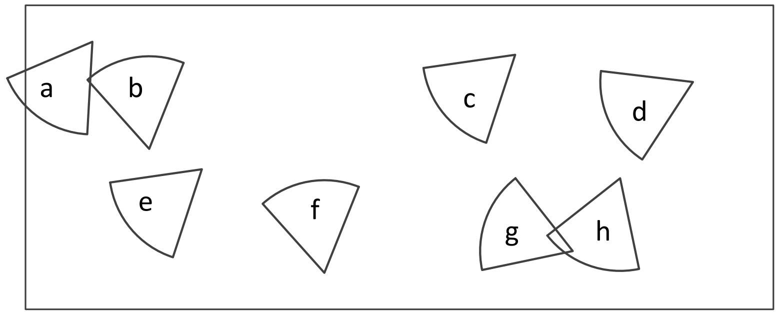
Definition 7. *Let $d_s(v_i, v_j)$ denote the* **strong distance** *between vertices v_i and v_j. $d_s(v_i, v_j) = 0$ if v_i and v_j overlap; otherwise, $d_s(v_i, v_j) = \min\{d(p_i, p_j)\}$ where p_i and p_j are points on v_i and v_j, respectively, and $d(p_i, p_j)$ is the Euclidean distance between p_i and p_j.*

The minimum number of mobile sensors needed to connect vertices v_i and v_j is, therefore, calculated as follows:

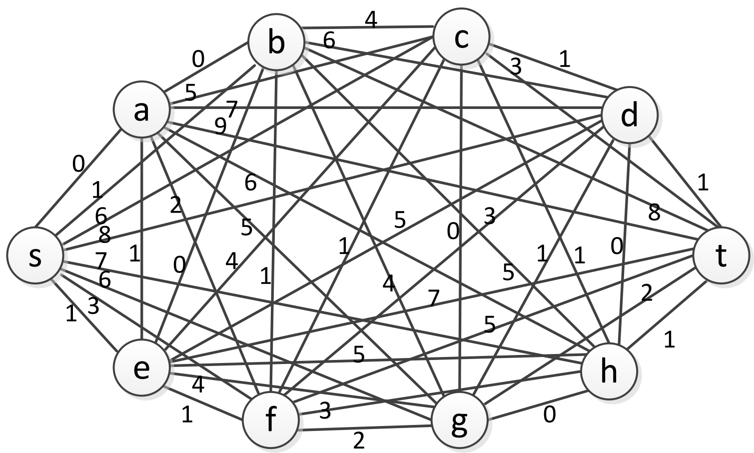
$$w(v_i, v_j) = \begin{cases} \lceil \dfrac{d_w(v_i, v_j)}{l_r} \rceil, & \text{weak barrier coverage,} \\ \lceil \dfrac{d_s(v_i, v_j)}{l_r} \rceil, & \text{strong barrier coverage.} \end{cases}$$

where l_r is the largest coverage distance of a sensor. Note that weak distance $d_w(v_i, v_j)$ and strong distance $d_s(v_i, v_j)$ are used for weighted barrier graph of weak barrier coverage and weighted barrier graph of strong barrier coverage, respectively.

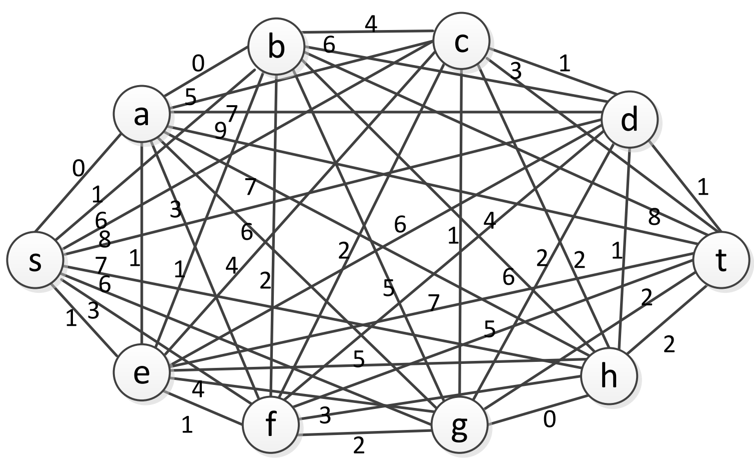
Figure 4.4(b) and 4.4(c) demonstrate the weighted barrier graph of weak barrier coverage and weighted barrier graph of strong barrier coverage for the sensor network shown in Figure 4.4(a). Any pair of vertices is connected by an edge except s and t. The weight of edge (s, a) is 0 because sensor a intersects with the left boundary.



(a)



(b)



(c)

Figure 3.1: (a) The deployed sensor network; (b) weighted barrier graph for weak barrier coverage; (c) weighted barrier coverage for strong barrier coverage.

The two weighted barrier graphs have the same set of vertices and edges. The only difference between them is the set of weights. For example, $w(b, f) = 1$ for weak barrier coverage while $w(b, f) = 2$ for strong barrier coverage, which means that 1 and 2 mobile sensors can connect sensors b and f for weak and strong barrier coverage, respectively.

3.3.1 Theoretical Analysis

In the following, we present theoretical analysis of the barrier coverage formation problem based on the WBG. Note that all the conclusions work for both weak and strong barrier coverage.

Lemma 3.0.1. *Any path from s to t on the weighted barrier graph is a barrier composed of pre-existing stationary sensors and virtual mobile sensors. The length of the path is the minimum number of mobile sensors required to form the barrier.*

Proof. According to the definition of weighted barrier graph, if we choose a path from s to t, and put exactly the number of mobile sensors on each edge of path, then the stationary sensors on the path are connected by mobile sensors, therefore, a barrier is formed. The minimum number of mobile sensors required to form the barrier is equivalent to the sum of weights of all edges on the path, which is the length of the path. $\square$

To better explain Lemma 3.0.1, take the path $s \to a \to b \to c \to d \to t$ in Figure 4.4(b) for example. The path length is 6, which means 6 mobile sensors are required to form the barrier along the path. There are three gaps on the path: $b \to c$, $c \to d$, and $d \to t$, which requires 4, 1 and 1 mobile sensors to fill them, respectively.

Recall that we want to use the minimum number of mobile sensors to form k sensor-disjoint barriers with stationary sensors. Based on the definition of weighted barrier graph, we have the following theorem.

Theorem 3.1. *If each of the k barriers to be formed must contain at least one stationary sensor, determining the minimum number of mobile sensors required to form k-barrier coverage with pre-existing stationary sensors is equivalent to finding k vertex-disjoint paths on the weighted barrier graph with the minimum total length.*

Proof. Based on Lemma 3.0.1, each barrier containing at least one stationary sensor must be a path from s to t on the WBG. Therefore, finding k sensor-disjoint barriers is equivalent to finding k vertex-disjoint paths on the WBG. Since we want to use the minimum number of mobile sensors to form k sensor-disjoint barriers, we should find the set of k vertex-disjoint paths on the WBG that has the minimum total length. □

Corollary 3.1.1. *The sensor network provides k-barrier coverage for the ROI after initial deployment iff there exist at least k vertex-disjoint paths with length of* 0 *on the WBG.*

Proof. A path with length 0 on the WBG means the stationary sensors on the path can form a barrier after initial deployment. When no mobile sensors is needed, finding k-barrier coverage is equivalent to finding k vertex-disjoint paths on the WBG. Therefore, a region is k-barrier covered after initial deployment is equivalent to the existence of k vertex-disjoint paths with length of 0 on the WBG. □

Besides all the paths from s to t on the WBG, there is a kind of special paths using only mobile sensors to form barriers. That is, s and t are directly connected by using only mobile sensors. For this kind of barriers, the optimal way of using the minimum number of mobile sensors, obviously, is to deploy them continuously along the horizontal direction. We call this kind of barrier as *direct barrier* and the corresponding path (s, t) as *direct path*. Given the length of belt region is L, the minimum number of mobile sensors needed to form a direct barrier is $\lceil \frac{L}{l_r} \rceil$. We can observe that a direct barrier is always sensor-disjoint from other paths on the WBG, and different direct barriers are always sensor-disjoint from each other. With this observation, we have the following lemma.

Lemma 3.1.1. *Given a belt region with length L, the minimum number of mobile sensors required for each barrier in the optimal set to the* Min-Num-Mobile(k) problem *is upper bounded by $\lceil \frac{L}{l_r} \rceil$.*

Proof. Suppose $\hat{P}_k$ is the optimal set of k sensor-disjoint barriers with the minimum number of mobile sensors needed to form k-barrier coverage. If any barrier in $\hat{P}_k$ needs more mobile sensors than a direct barrier, we can always replace it with a direct barrier for less number of mobile sensors needed. Therefore, the previous $\hat{P}_k$ is not the optimal set, which contradicts to our assumption. Hence, no barrier in the optimal set needs more than $\lceil \frac{L}{l_r} \rceil$ mobile sensors. □

Direct barriers are also needed when the vertex-disjoint paths found on the WBG are not enough. Suppose the application requires 5-barrier coverage, but the maximum number of vertex-disjoint paths found on the WBG is 3, then we can add two direct barriers to reach 5-barrier coverage.

Suppose there exist k vertex-disjoint paths on the WBG, and P_k^* denote the set of k vertex-disjoint paths with the minimum total length on the WBG. Note that P_k^* may not the optimal set to the *Min-Num-Mobile(k) problem* that has the minimum total length after considering direct paths, even no path in P_k^* is longer than $\lceil \frac{L}{l_r} \rceil$. We will present the algorithm to find the optimal set in Section 3.4.

3.4 The Min-Num-mobile(k) problem

In this section, we present an efficient optimal algorithm and a greedy algorithm to solve the *Min-Num-Mobile(k) problem.*

Before introducing our optimal algorithm, we first introduce the *vertex-disjoint path algorithm* (Bhandari, 1998) which can find a set of vertex-disjoint paths with the minimum total length, P_q^*, on a graph.

Given P_i^*, the *vertex-disjoint path algorithm* performs the following steps to find P_{i+1}^*.

Step 1: Graph transformation. Transform the graph G into a new graph NG based on P_i^* by using the following procedures. First, replace the edges of the disjoint paths in P_i^* by arcs directed towards the source, and make the length of the arcs negative; Second, split each vertex (except for endpoint vertices) on the disjoint paths into two co-located subvertices joined by an arc of length zero. Direct the arc of length zero towards the source. Replace each external edge connected to a vertex on the shortest paths by its two arcs of the same length, where one arc is directed to the first subvertex and the other one is directed from the second subvertex.

Step 2: Shortest path finder. Find the shortest path np on the new graph NG using the modified Dijkstra algorithmBhandari (1998).

Step 3: Path update. Update P_i^* and np to get P_{i+1}^*: transform to the original graph G and erase any edge of this shortest path interlacing with the previous set of vertex-disjoint paths P_i^*. Find the new set of vertex-disjoint paths P_{i+1}^* after removing the interlacing edges.

The initialization of the algorithm is P_1^* which is the shortest path on the graph. Once P_1^* is obtained, we can perform these steps iteratively to find P_2^*, P_3^* and so on. Note that for $i < j$, P_i^* may not be a subset of P_j^*. Take Figure 4.4(c) as an example, $P_2^* = \{\{s,a,b,c,d,t\}, \{s,e,f,g,h,t\}\}$, and $P_3^* = \{\{s,a,b,c,d,t\}, \{s,e,t\}, \{s,f,g,h,t\}\}$. More details of the algorithm can be found in (Bhandari, 1998).

3.4.1 Optimal Algorithm

Let $\hat{P}_k$ denote the optimal set of k sensor-disjoint barriers requiring the minimum number of mobile sensors, and $N_m = |\hat{P}_k|$ denote the minimum number of mobile sensors needed. Note that $|\cdot|$ denotes the total length of paths in $\cdot$. We first define the k-auxiliary set to help us find the optimal set $\hat{P}_k$.

Definition 8. k-*auxiliary set*: *P_q^k is called the k-auxiliary set of P_q^* ($0 \leq q \leq k$), which is composed of P_q^* and $k-q$ direct barriers (s,t).*

We leverage the *vertex-disjoint path algorithm* to help the design of our optimal algorithm. The basic idea is to first find all the sets of vertex-disjoint paths with the minimum total length on the WBG, and then extend each set P_q^* $(0 \leq q \leq \eta)$ to its k-auxiliary set P_q^k, where $\eta = \min(k, \zeta)$ and ζ is the maximum number of vertex-disjoint paths on the WBG. The optimal set $\hat{P}_k$ is the k-auxiliary set that has the minimum total length among all k-auxiliary sets.

Algorithm 5 Min-Num-Mobile(k)-Optimal algorithm

Require: Weighted barrier graph G, L, k and l_r
Ensure: $\hat{P}_k$ and N_m
1: Let $P_0^* \leftarrow \emptyset$ and P_0^k denote a set of k direct barriers
2: $P_1^* \leftarrow$ Dijkstra(G)
3: $\eta \leftarrow 1$
4: **while** $\eta < k$ **do**
5: $\quad NG \leftarrow$ graph-transform(G, P_η^*)
6: $\quad$ **if** there exist paths from s to t on NG **then**
7: $\quad\quad np \leftarrow$ modified-Dijkstra(NG)
8: $\quad\quad \eta \leftarrow \eta + 1$
9: $\quad\quad P_\eta^* \leftarrow$ path-update$(P_{\eta-1}^*, np)$
10: $\quad$ **else**
11: $\quad\quad$ break
12: $\quad$ **end if**
13: **end while**
14: $\hat{P}_k \leftarrow P_\eta^k$, and $N_m \leftarrow |P_\eta^*| + (k-\eta)\lceil \frac{L}{l_r} \rceil$
15: **for** $q = 0$ to $\eta - 1$ **do**
16: $\quad$ **if** $|P_q^*| + (k-q)\lceil \frac{L}{l_r} \rceil < N_m$ **then**
17: $\quad\quad \hat{P}_k \leftarrow P_q^k$, and $N_m \leftarrow |P_q^*| + (k-q)\lceil \frac{L}{l_r} \rceil$
18: $\quad$ **end if**
19: **end for**

Algorithm 5 describes the details of the optimal algorithm where Step 2 finds the first shortest path on the WBG, i.e., P_1^*, Step 4 through 11 perform the *vertex-disjoint path algorithm* iteratively to find all P_q^* for $1 < q \leq \eta$, and Step 13 through 15 find the k-auxiliary set with the minimum total length among all k-auxiliary sets and claim it as the optimal set.

Theorem 3.2. *The optimal set of k sensor-disjoint barriers requiring the minimum number of mobile sensors, $\hat{P}_k$, is the k-auxiliary set with the minimum total length among all k-auxiliary sets.*

Proof. We first prove that $\hat{P}_k$ must be a k-auxiliary set P_q^k composed of P_q^* ($q \in [0, \eta]$) and $k-q$ direct barriers, where $\eta = \min(k, \zeta)$ and ζ is the maximum number of vertex-disjoint paths on the WBG.

Each barrier either contains at least one stationary sensor or no stationary sensor. Therefore, each barrier in $\hat{P}_k$ is either a path on the WBG or a direct barrier.

Suppose no barrier in $\hat{P}_k$ is a direct barrier, then all barriers in $\hat{P}_k$ are paths on the WBG. We know that P_k^* is the set of k vertex-disjoint paths with the minimum total length on the WBG. Therefore, $\hat{P}_k = P_k^*$, which is composed of P_k^* and $k-k=0$ direct barriers.

When $\hat{P}_k$ contains direct barriers, suppose there are $k-q$ ($0 \leq q \leq \eta$) direct barriers in $\hat{P}_k$. We prove that the rest q sensor-disjoint barriers in the optimal set must be P_q^*. We prove it by contradiction. Suppose the rest q sensor-disjoint barriers (vertex-disjoint paths) in $\hat{P}_k$ is not P_q^*, we can always use P_q^* to replace these q sensor-disjoint barriers to get a new set of k sensor-disjoint barriers with smaller total length, which means that $\hat{P}_k$ is not the optimal set. This contradicts to our assumption. Therefore, the rest q sensor-disjoint barriers in $\hat{P}_k$ must be P_q^*.

Therefore, the optimal set of k sensor-disjoint barriers must be composed of P_q^* ($q \in [0, \eta]$) and $k-q$ direct barriers, which is a k-auxiliary set. The total length of a k-auxiliary set is $|P_q^*| + (k-q)\lceil \frac{L}{l_r} \rceil$. Since q ranges from 0 to η, the k-auxiliary set with the minimum total length is the optimal set of k sensor-disjoint barriers and the minimum number of mobile sensors needed is:

$$N_m = \min\{|P_q^*| + (k-q)\lceil \frac{L}{l_r} \rceil\}_{q=0}^{\eta}$$

The optimality of Algorithm 5 is proved. $\square$

Theorem 3.3. *Given a sensor network with n stationary sensors, the optimal algorithm can solve the* Min-Num-mobile(k) problem *in $O(kn^2)$.*

Proof. The number of vertices on the WBG is $n+2$, which is on the order of n. The number of edges on the graph is $n(n-1)/2-1$, which is on the order of n^2. The *vertex-disjoint path algorithm* consists of graph transformation, modified Dijkstra algorithm and path update. The running time of graph transformation and path update is $O(n)$ and the running time of the modified Dijkstra algorithm is $O(n \log n + n^2)$. Thus, the running time of the vertex-disjoint path algorithm is $O(n^2)$. Since the *vertex-disjoint path algorithm* is performed at most k times, the optimal algorithm can solve the *Min-Num-Mobile(k) problem* in $O(kn^2)$. □

3.4.2 Greedy Algorithm

The *vertex-disjoint path algorithm* involves a lot of operations, such as graph transformation (node-split and node-merge), which are complicated especially for large-scale networks. In this section, we propose a greedy algorithm which is faster than the optimal algorithm.

The basic idea of the greedy algorithm is to repeatedly find the shortest path on the WBG until k paths are found or the latest found path is longer than $\lceil \frac{L}{l_r} \rceil$ or no path can be found. If, in the end, the number of found paths is smaller than k, additional direct barriers are added to form the k barriers. The procedures of the greedy algorithm are described as follows:

1. Initialize $\hat{P}_k$ as an empty set.

2. If there exist paths from s to t on the WBG, find the shortest path using Dijkstra's algorithm; otherwise, go to 5).

3. If the found shortest path is longer than $\lceil \frac{L}{l_r} \rceil$, discard the path, go to 5); otherwise, go to 4).

4. Add the path into $\hat{P}_k$. If the path is the k-th found path, stop; otherwise, remove all the vertices (except s and t) on the found path from the WBG, go to 2).

5. Suppose the number of paths in $\hat{P}_k$ is q, add $k - q$ direct barriers into $\hat{P}_k$.

The pseudocode of the greedy algorithm is presented in Algorithm 6.

Algorithm 6 Min-Num-Mobile(k)-Greedy algorithm

Require: Weighted barrier graph G, L, k and l_r
Ensure: $\hat{P}_k$ and N_m

$\hat{P}_k \leftarrow \emptyset$, $q \leftarrow 0$
while $q < k$ **do**
  if there exist paths from s to t on G **then**
    $p \leftarrow$ Dijkstra(G)
    if $|p| \leq \lceil \frac{L}{l_r} \rceil$ **then**
      $\hat{P}_k \leftarrow \hat{P}_k \cup p$
      $q \leftarrow q + 1$
      update G by removing all the vertices (except s and t) on p
    else
      break;
    end if
  end if
end while
$N_m \leftarrow |\hat{P}_k|$
if $q < k$ **then**
  $\hat{P}_k \leftarrow \hat{P}_k \cup \underbrace{\{(s,t), \cdots, (s,t)\}}_{k-q}$, and $N_m \leftarrow |\hat{P}_k|$
end if

Theorem 3.4. *Given a sensor network with n stationary sensors, the greedy algorithm can solve the* Min-Num-Mobile(k) problem *in $O(kn^2)$.*

Proof. We have shown that the number of vertices and edges on the WBG are on the order of n and n^2, respectively. Therefore, the running time of Dijkstra's algorithm is $O(n^2)$. Since the greedy algorithm runs Dijkstra's algorithm at most k rounds, the greedy algorithm can solve the *Min-Num-Mobile(k) problem* in $O(kn^2)$. □

Although the running times of the optimal algorithm and the greedy algorithm are both $O(kn^2)$ in the worst case, the greedy algorithm is usually much faster than the optimal algorithm, especially for large scale networks, since it does not need to perform graph transformation and path update. We will show the comparison of computation time between two algorithms in the performance evaluation section.

3.5 The Max-Num-Barrier problem

Once the minimum number of mobile sensors required to form k-barrier coverage is solved, the *Max-Num-Barrier problem* can be solved accordingly. Notice that the *Max-Num-Barrier problem* is studied under a different scenario where both the stationary and mobile sensors have been pre-deployed.

Given an ROI and a deployed hybrid sensor network with n stationary and τ mobile sensors, the maximum number of barriers that could be formed, denoted by N_b, is k if the minimum number of mobile sensors required to form k-barrier coverage is less than or equal to τ, but the minimum number of mobile sensors required to form $(k+1)$-barrier coverage is larger than τ. Therefore, the optimal solution to the *Max-Num-Barrier problem* is based on the optimal solution to the *Min-Num-Mobile(k) problem*. In the following, we propose an optimal algorithm as well as a faster greedy algorithm to solve the *Max-Num-Barrier problem*.

3.5.1 Optimal Algorithm

The optimal algorithm is described as follows:

1. Perform Algorithm 5 (the *Min-Num-Mobile(k)-Optimal Algorithm*) with k increasing until $|\hat{P}_{k+1}| > \tau$.

2. The maximal number of barriers is k.

According to Theorem 3.2, the set of N_b barriers is composed of a set of vertex-disjoint paths P_q^* on the WBG and direct barriers. Therefore, we have

$$N_b = q + \lfloor (\tau - |P_q^*|)/\lceil \frac{L}{l_r} \rceil \rfloor \tag{3.3}$$

Theorem 3.5. *The maximum number of barriers N_b is lower bounded by $\lfloor \tau/\lceil \frac{L}{l_r} \rceil \rfloor$ and upper bounded by $n + \lfloor \tau/\lceil \frac{L}{l_r} \rceil \rfloor$.*

Proof. When all barriers are direct barriers, the maximum number of barriers reaches its lower bound $\lfloor \tau/\lceil \frac{L}{l_r} \rceil \rfloor$. In Eq. 3.3, $N_b = q + \lfloor (\tau - |P_q^*|)/\lceil \frac{L}{l_r} \rceil \rfloor$. For a WBG, $q \leq n$ because the maximum number of vertex-disjoint paths on it cannot be larger than the number of stationary sensors n. When q reaches n, and the total length $|P_n^*|$ is 0, the maximum number of barriers reaches its upper bound $n + \lfloor \tau/\lceil \frac{L}{l_r} \rceil \rfloor$. □

Theorem 3.6. *For a deployed sensor network with n stationary sensors and τ mobile sensors, the optimal algorithm can solve the* Max-Num-Barrier problem *in $O(n^3)$.*

Proof. The basis of the optimal algorithm is the *Min-Num-Mobile(k)-Optimal Algorithm*, the running time of which is $O(kn^2)$ for k barriers. According to Theorem 3.5, in the worst case, the maximum number of barriers could be $n + \lfloor \tau/\lceil \frac{L}{l_r} \rceil \rfloor$. The running time of the *Min-Num-Mobile(k)-Optimal Algorithm* is $O(n^3 + n^2 \lfloor \tau/\lceil \frac{L}{l_r} \rceil \rfloor)$ for $n + \lfloor \tau/\lceil \frac{L}{l_r} \rceil \rfloor$ barriers. Since $\lfloor \tau/\lceil \frac{L}{l_r} \rceil \rfloor$ is a constant, the optimal algorithm can solve the *Max-Num-Barrier problem* in $O(n^3)$. □

3.5.2 Greedy Algorithm

We also propose a faster greedy algorithm for the *Max-Num-Barrier problem*. The basic idea is to repeatedly find the shortest path on the WBG until the deployed number of mobile sensors is reached or no path can be found or the latest found path is longer than $\lceil \frac{L}{l_r} \rceil$. In the end, if some mobile sensors are left, we use them to construct direct barriers. The greedy algorithm is described as follows:

1. Initialize q with 0, and P_q as an empty set.

2. If there exist paths from s to t on the WBG, find the shortest path p using Dijkstra's algorithm; otherwise, go to 5).

3. If the found shortest path is longer than $\lceil \frac{L}{l_r} \rceil$, discard the path, go to 5); otherwise, go to 4).

4. If $|P_q| + |p| < \tau$, remove all the vertices (except for s and t) on the path p from the WBG, put p into P_q and increase q by 1, go to 2). If $|P_q| + |p| = \tau$, $k = q+1$, stop; otherwise, $k = q$, stop.

5. The maximum number of barriers is $q + \lfloor (\tau - |P_q|)/\lceil \frac{L}{l_r} \rceil \rfloor$.

The pseudocode of the greedy algorithm is presented in Algorithm 7.

Algorithm 7 Max-Num-Barrier-Greedy algorithm

Require: Weighted barrier graph G, L, l_r and τ
Ensure: N_b

```
q ← 0 and P_q ← ∅
while true do
    if there exist paths from s to t on G then
        p ← Dijkstra(G)
        if |p| ≤ ⌈L/l_r⌉ then
            if |P_q| + |p| ≤ τ then
                q ← q + 1
                P_q ← P_{q−1} ∪ p
                update G by removing all the edges incident to the vertices (except s and t)
                on p
            else
                if |P_q| + |p| = τ then
                    N_b ← q + 1, break.
                else
                    N_b ← q, break.
                end if
            end if
        else
            N_b ← q + ⌊(τ − |P_q|)/⌈L/l_r⌉⌋, break.
        end if
    else
        N_b ← q + ⌊(τ − |P_q|)/⌈L/l_r⌉⌋, break.
    end if
end while
```

Theorem 3.7. *The maximum number of barriers found by the greedy algorithm is lower bounded by* $\lfloor \tau / \lceil \frac{L}{l_r} \rceil \rfloor$ *and upper bounded by* $n + \lfloor \tau / \lceil \frac{L}{l_r} \rceil \rfloor$.

Proof. The proof is similar to that of Theorem 3.5. □

Theorem 3.8. *For a deployed sensor network with n stationary sensors and τ mobile sensors, the greedy algorithm can solve the* Max-Num-Barrier problem *in $O(n^3)$.*

Proof. The running time of Dijkstra's algorithm is $O(n^2)$. In the worst case, the greedy algorithm would perform Dijkstra's algorithm $n + \lfloor \tau / \lceil \frac{L}{l_r} \rceil \rfloor$ times. Therefore, the running time for the greedy algorithm is $O(n^3 + n^2 \lfloor \tau / \lceil \frac{L}{l_r} \rceil \rfloor)$. Since $\lfloor \tau / \lceil \frac{L}{l_r} \rceil \rfloor$ is a constant, the greedy algorithm can solve the *Max-Num-barrier problem* in $O(n^3)$. □

3.6 Performance Evaluation

In this section, we conduct simulations using Matlab to evaluate the performance of our proposed algorithms.

3.6.1 Performance Evaluation on the Min-Num-Mobile(k) Problem

The ROI is a belt region of length $L = 500m$ and width $H = 100m$. Initially, stationary sensors are uniformly deployed in the belt region. After the minimum number of mobile sensors is calculated, mobile sensors are deployed uniformly in the belt region and then assigned to different target locations using the Hungarian algorithm to form k-barrier coverage.

The evaluation mainly focuses on four performance metrics:

- The minimum number of mobile sensors required to form k-barrier coverage
- The total moving distance for mobile sensors to form k-barrier coverage
- The average moving distance for mobile sensors to form k-barrier coverage
- The number of direct barriers needed in k barriers

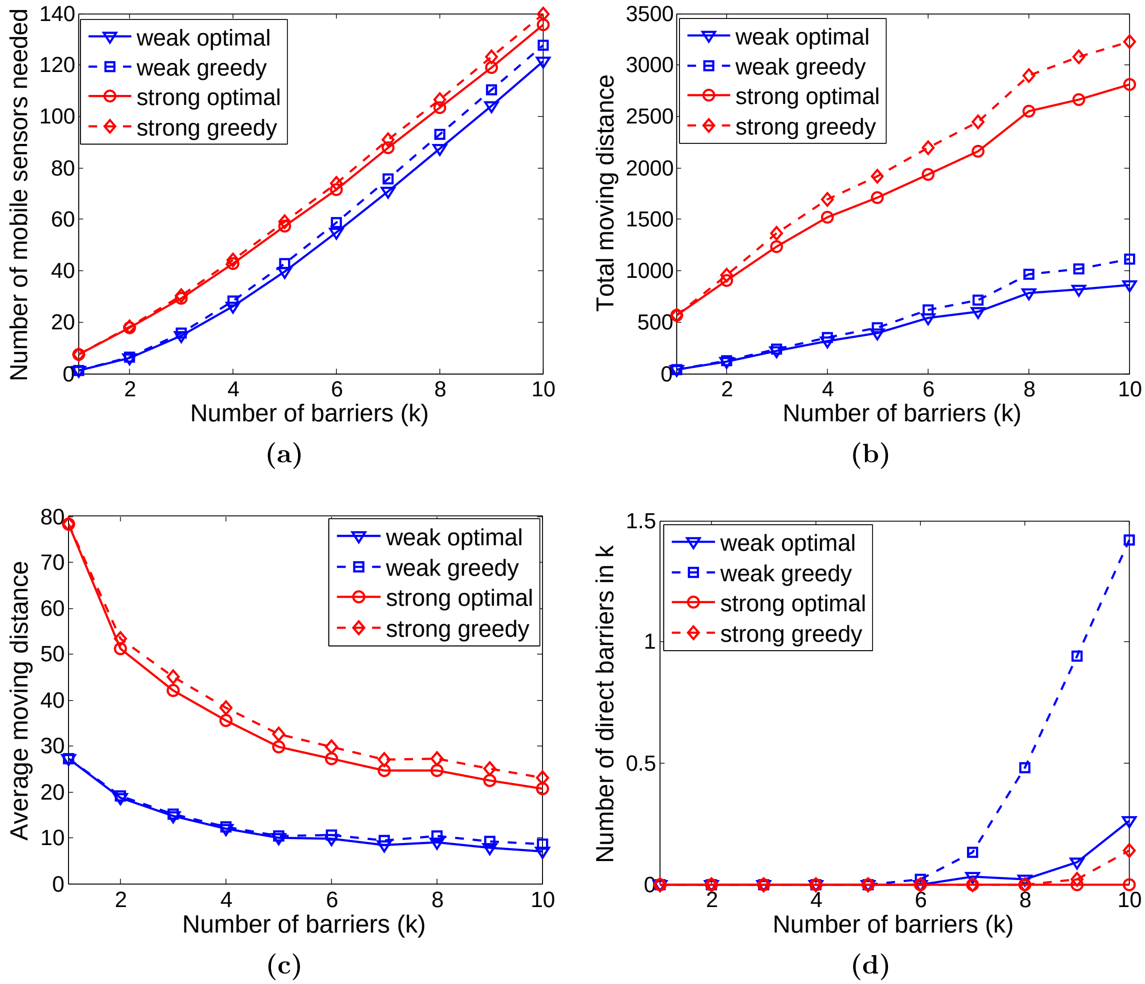


Figure 3.2: Performance evaluation on different number of barriers (k), and fixed $n = 100$, $r = 20$ and $\alpha = \pi/4$

Evaluation of these performance metrics is conducted on different parameters, such as the number of barriers, the number of stationary sensors, the sensing range and the sensing angle (or field of view). For all the simulation results presented in this chapter, each data point is an average of 100 experiments. Both weak and strong barrier coverage are studied.

Effects of the Number of Barriers

We first evaluate the performance of the algorithms on the number of barriers. Figure 3.2 shows the performance results. The number of mobile sensors required to form k-barrier coverage increases as k increases for all the algorithms, as shown in Figure 3.2(a). The optimal algorithms always use less number of mobile sensors to realize k weak/strong barrier coverage than the greedy algorithms. When $k \leq 3$, the two algorithms give the same result. Therefore, when smaller number of barriers is needed to be formed, greedy algorithms are more suitable because they are faster than the optimal algorithms. We can also observe that forming strong barrier coverage always requires more number of mobile sensors than forming weak barrier coverage.

The total moving distance, shown in Figure 3.2(b), increases as k increases because more mobile sensors are needed to fill in more gaps when k becomes larger. The total moving distance for strong barrier coverage is always longer than that for weak barrier coverage. This is due to two reasons. First, forming strong barrier coverage requires more number of mobile sensors than forming weak barrier coverage. Second, mobile sensors only need to move in the horizontal direction for weak barrier coverage while they need to move in two dimension for strong barrier coverage. Although the total moving distance is increasing, the average moving distance for mobile sensors, as shown in Figure 3.2(c), decreases when k becomes larger. This is because that less number of mobile sensors is required for smaller number of barrier coverage, which results in less number of mobile sensors deployed. Then mobile sensors are much more dense for larger number of barrier coverage. Therefore, each mobile sensor under larger k moves less on average to reach a target location as compared to the mobile sensor under smaller k.

As shown in Figure 3.2(d), no direct barriers are needed when $k \leq 5$, and then the number of direct barriers increases linearly as k increases. This is because stationary sensors can work with mobile sensors to construct barriers when k is small. When

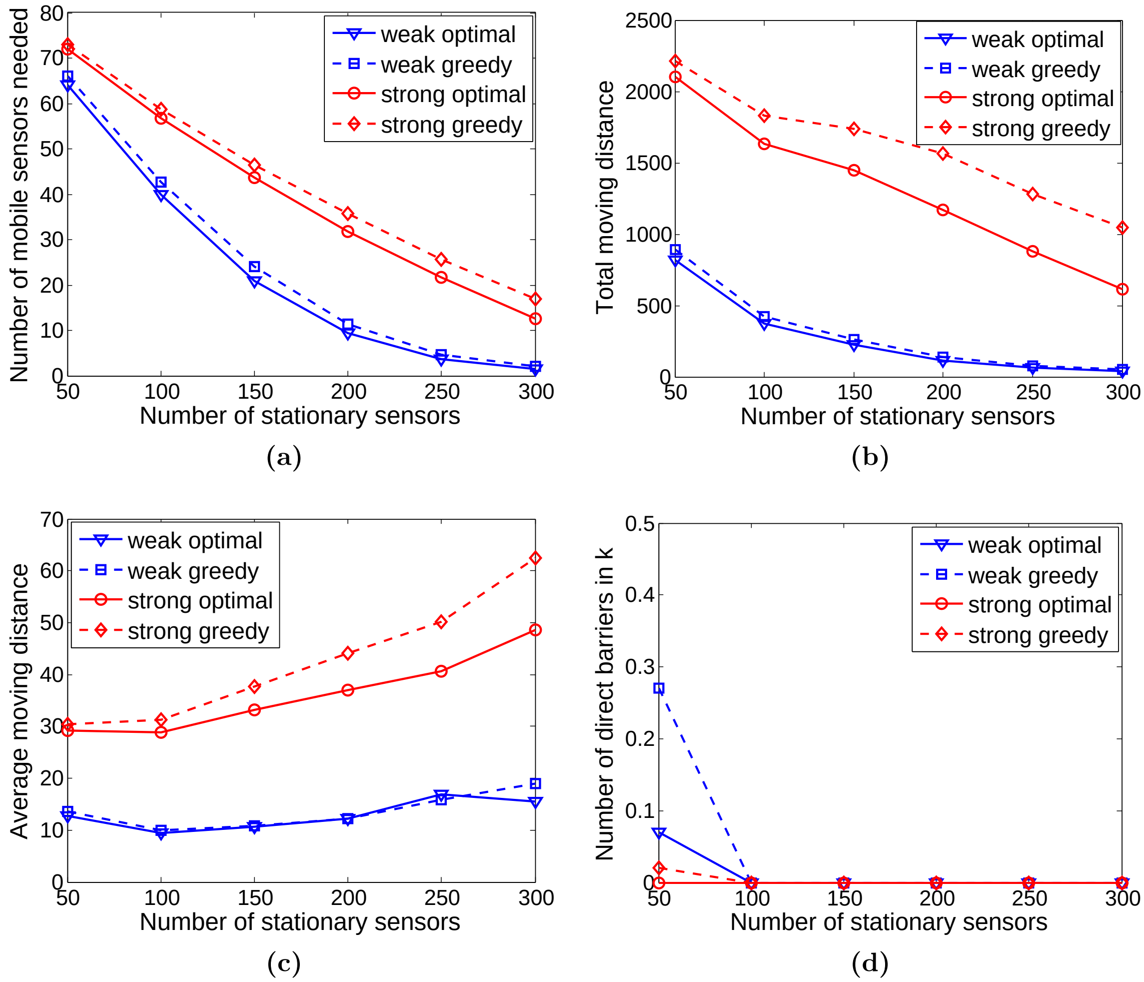


Figure 3.3: Performance evaluation on different number of stationary sensors (n), and fixed $k = 5$, $r = 20$ and $\alpha = \pi/4$

most of stationary sensors are used, if we want to form more barriers, the direct barrier is obviously the best choice.

Effects of the Number of Stationary Sensors

We then evaluate the effects of the number of stationary sensors on the performance metrics. Figure 3.3 shows the performance results. Given a fixed number of barriers needed to form, the number of mobile sensors required decreases when the number of stationary sensors increases. This is because that, when the number of stationary sensors is larger, more stationary sensors can be used to form barriers, which reduces the number of mobile sensors needed. The optimal algorithms always require less number of mobile sensors than that of the greedy algorithms. Achieving strong barrier coverage requires more number of mobile sensors than achieving weak barrier coverage.

The total moving distance, as shown in Figure 3.3(b), decreases as the number of stationary sensors increases. This is because that less number of mobile sensors is required to form barriers when the number of stationary sensors is larger. The total moving distance for strong barrier coverage is always larger than that of weak barrier coverage. Figure 3.3(c) shows that the average moving distance of mobile sensors increases as the number of stationary sensors increases. As shown in Figure 3.3(d), when the number of stationary sensors is small, direct barriers might be needed to achieve the specified number of barriers.

Effects of the Sensing Range

We also evaluate the performance on different sensing range of sensors. Figure 3.4 shows the performance results. The number of mobile sensors required decreases when the sensing range increases. The reason is that, larger sensing range results in smaller gaps between sensors and less number of mobile sensors to fill in the gaps. The optimal algorithm always requires less number of mobile sensors than the greedy algorithm. Achieving strong barrier coverage requires more number of mobile sensors than achieving weak barrier coverage.

The total moving distance, as shown in Figure 3.4(b), decreases as the sensing range increases. This is because that less number of mobile sensors is required to form barriers when the sensing range is larger. The total moving distance for strong barrier coverage is always larger than that of weak barrier coverage. Figure 3.4(c) shows that the average moving distance of mobile sensors increases as the sensing range increases. As shown in Figure 3.4(d), when the sensing range is small, before reaching the specified number of barriers, most of stationary sensors have been used

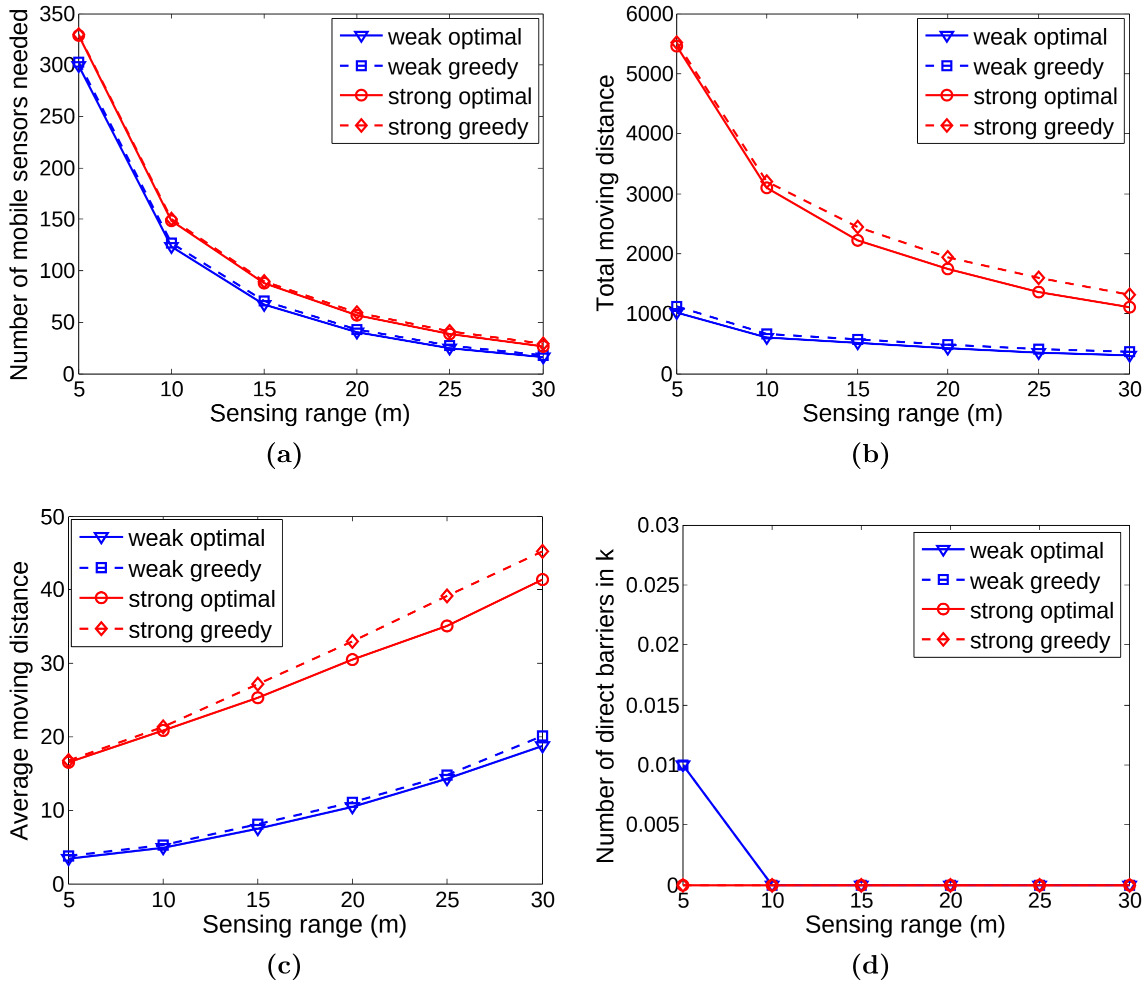


Figure 3.4: Performance evaluation on different sensing range (r), and fixed $k = 5$, $n = 100$ and $\alpha = \pi/4$

for constructing barriers, therefore, direct barriers might be needed to achieve the specified number of barriers.

Effects of the Sensing Angle

Finally, we evaluate the performance of the proposed algorithms on different sensing angle of sensors. Figure 3.5 shows the performance results. The number of mobile sensors required decreases when the sensing angle increases. The reason is that, larger sensing angle results in smaller gaps between sensors and less number of mobile sensors

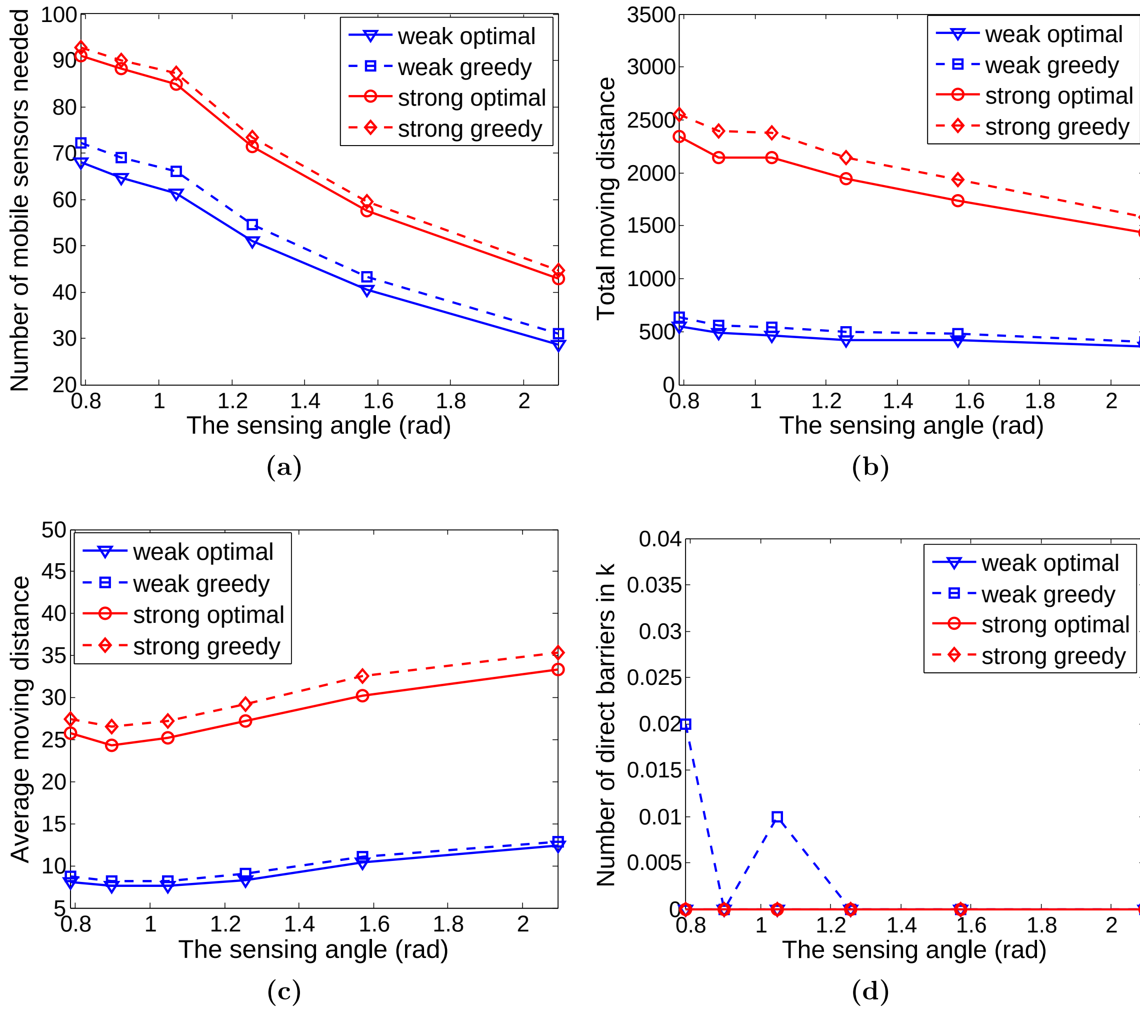


Figure 3.5: Performance evaluation on different sensing angle (2α), and fixed $k = 5$, $n = 100$ and $r = 20$

to fill the gaps. The optimal algorithms always require less number of mobile sensors than that of the greedy algorithms. Achieving strong barrier coverage requires more number of mobile sensors than achieving weak barrier coverage.

The total moving distance, as shown in Figure 3.5(b), decreases as the sensing angle increases. This is because that less number of mobile sensors is required to form barriers when the sensing angle is larger. The total moving distance for strong barrier coverage is always larger than that of weak strong barrier coverage. Figure 3.5(c) shows that the average moving distance of mobile sensors increases as the sensing angle increases. As shown in Figure 3.3(d), when the sensing angle is small, before reaching the specified number of barriers, most of stationary sensors have been used for constructing barriers, therefore, direct barriers might be needed to achieve the specified number of barriers.

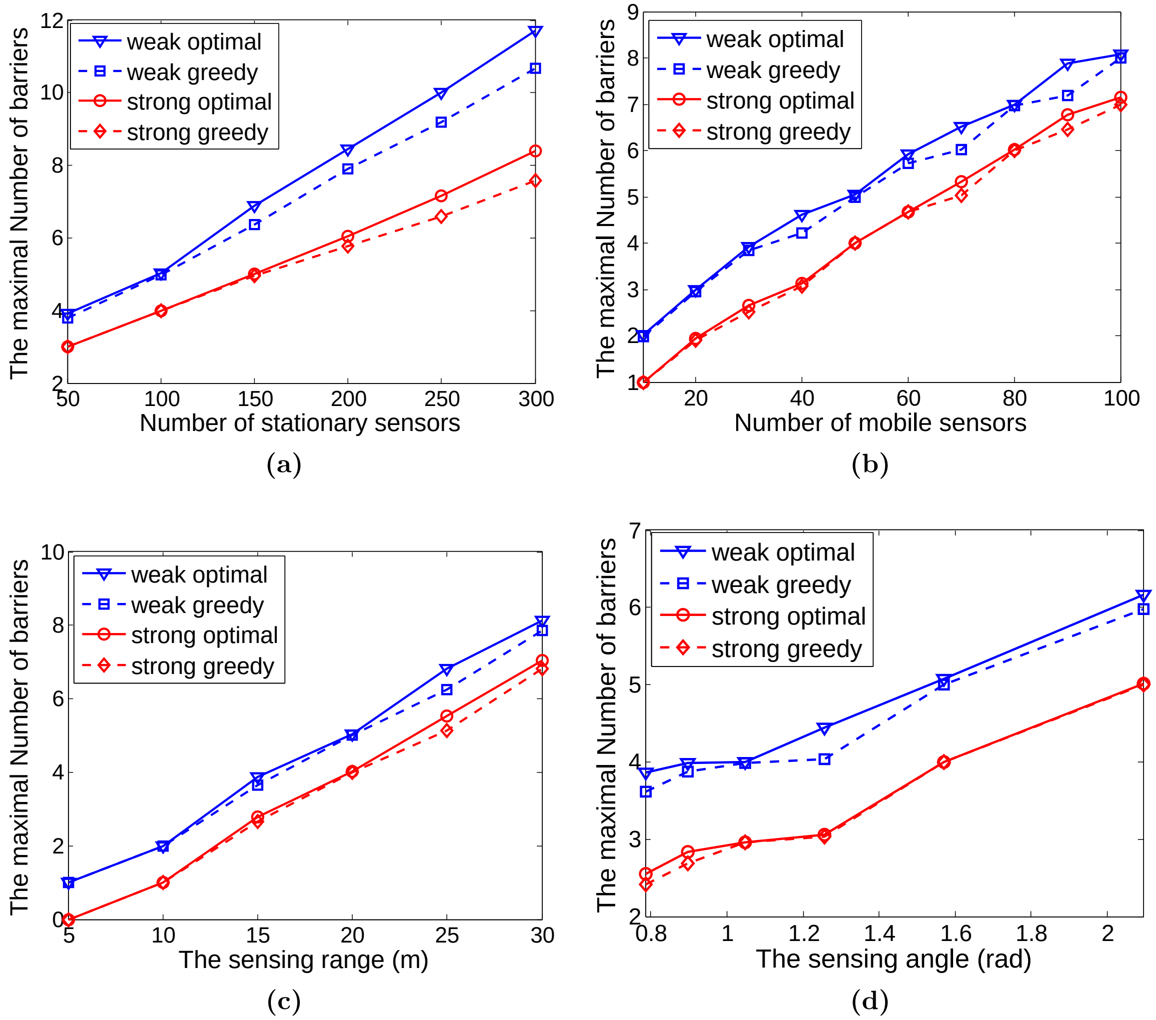


Figure 3.6: Performance evaluation on: (a) different number of stationary sensors, fixed $\tau = 50$, $r = 20m$ and $\alpha = \pi/4$; (b) different number of mobile sensors, fixed $n = 100$, $r = 20m$ and $\alpha = \pi/4$; (c) different sensing range, fixed $n = 100$, $\tau = 50$ and $\alpha = \pi/4$; (d) different sensing angle, fixed $n = 100$, $\tau = 50$ and $r = 20m$

3.6.2 Performance Evaluation on the Max-Num-Barrier Problem

In this section, we evaluate the performance of the proposed algorithms for the *Max-Num-Barrier problem*.

The ROI is a belt region of length $L = 500m$ and width $H = 100m$. Initially, both n stationary sensors and τ mobile sensors are uniformly deployed in the ROI. After the maximum number of barriers and the set of barriers are found, mobile sensors can move to target locations to form multiple barriers. The maximum number of barriers is the performance metric of the evaluation. The evaluation is conducted on different parameters including the number of stationary sensors, the number of mobile sensors, the sensing range and the sensing angle.

Figure 3.6 shows the performance evaluation results. We can observe that the increasing of any one of the four factors could result in the increasing of the maximum number of barriers. The proposed optimal algorithms always perform better than the greedy algorithms. However, the difference between them is not very obvious. For example, the maximum number of barriers for greedy and optimal algorithm are almost the same in Figure 3.6(b). We also observe that the maximum number of weak barriers is always larger than that of strong barriers, given the same network deployment. That is, it is much easier to form weak barriers than to form strong barriers.

3.6.3 Computation Time Comparison

Figure 3.7 demonstrate the comparison of computation time between the optimal algorithm and the greedy algorithm. The algorithms run on Thinkpad T420 with CPU of 2.80GHz and 4GB RAM. We can see that the computation time of optimal algorithms increases significantly with the increase of the number of barriers or the number of mobile sensors deployed. The computation time of greedy algorithm, however, does not increase significantly. Therefore, although two algorithms for the

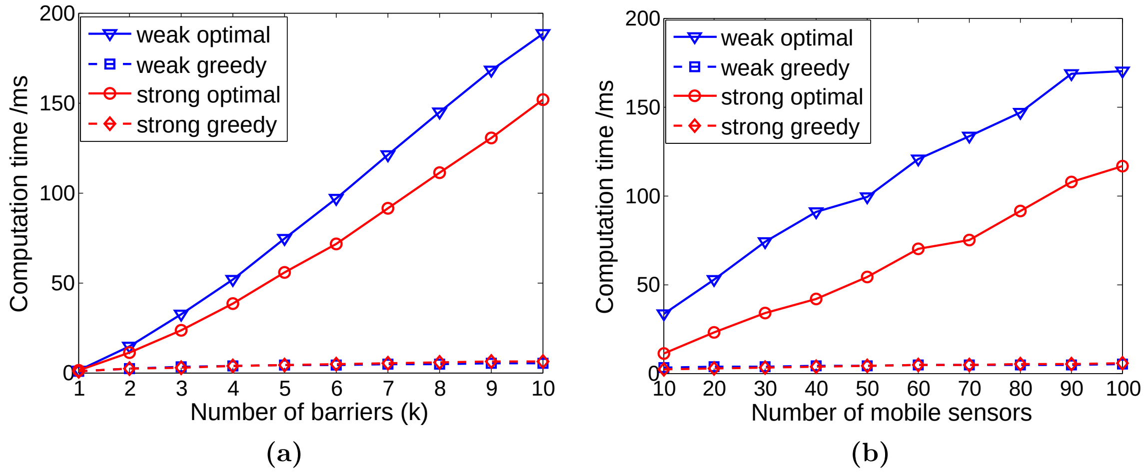


Figure 3.7: Computation time comparison between the optimal and greedy algorithm for: (a) Min-Num-Mobile(k) problem; (b) Max-Num-Barrier problem with fixed $n = 100$, $r = 20$ and $\alpha = \pi/4$.

same problem have the same running time in the worst case, the greedy algorithm is usually faster and more scalable to large-scale networks as compared to the optimal algorithm.

3.7 Summary

In this chapter, we studied the k-barrier coverage formation problem in sensor networks. We introduced a novel weighted barrier graph (WBG) model for the barrier coverage formation problem, and proved that determining the minimum number of mobile sensors required to form k-barrier coverage is related with but not equal to finding k vertex-disjoint paths with the minimum total length on the WBG. With this observation, we proposed an optimal algorithm and a faster greedy algorithm to find the minimum number of mobile sensors required to form k-barrier coverage with pre-deployed stationary sensors. We also proposed an optimal algorithm and a faster greedy algorithm to determine the maximum number of barriers when both the stationary and mobile sensors have been pre-deployed. Both analytical and experimental studies demonstrated the effectiveness of our proposed algorithms.

Chapter 4

Fault Tolerant Barrier Coverage for Sensor Networks

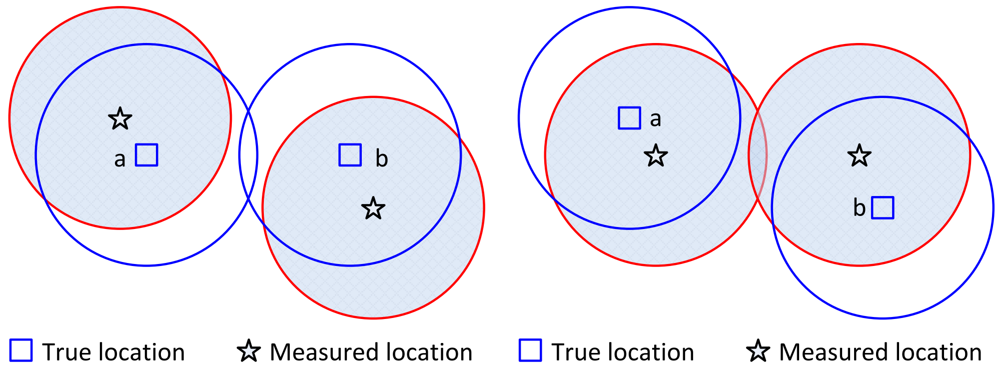
Barrier coverage is highly related with locations of sensors. Existing work on barrier coverage mainly assume that sensors have accurate location information, however, little work explores the effects of location errors on barrier coverage. In this chapter, we study the barrier coverage problem when sensors have location errors and deploy mobile sensors to improve barrier coverage if the network is not barrier covered after initial deployment. We analyze the relationship between the true distance and the measured distance of two stationary sensors and derive the minimum number of mobile sensors needed to connect them with a guarantee when sensors location errors exist. Furthermore, we propose a fault tolerant weighted barrier graph, based on which we prove that the minimum number of mobile sensors needed to form barrier coverage with a guarantee is the length of the shortest path on the graph. Simulation results validate the correctness of our analysis.

4.1 Introduction

Deterministic and random deployment are the two most popular ways of deploying sensors to the ROIs. For the ROIs with friendly environment, deterministic deployment can be used to deploy sensors to the exact locations as we expect. However, in general, the ROIs are in harsh environment and difficult for human being to reach, which makes random deployment (e.g., dropped by aircraft) the only way to deploy sensors. When only stationary sensors are used, after the initial random deployment, it is highly possible that sensors could not form a barrier due to the gaps in their coverage, which would allow intruders to cross the ROIs without being detected. Therefore, it is necessary to deploy more sensors to form a barrier. In fact, it is difficult if possible at all to improve barrier coverage for sensor networks consisting of only stationary sensors. Fortunately, with recent technological advances, practical mobile sensors (e.g., Robomote (Dantu et al., 2005), Packbot (Somasundara and Ramamoorthy, 2007)) have been developed, which provides us a way to improve barrier coverage performance after sensor networks have been deployed.

Location information of sensors serves the basis of lots of applications, such as navigation and target tracking. However, it is cost-expensive to equip GPS receivers on each node. Therefore, the location information of sensors are unknown when they are randomly deployed. To obtain the location information of each node, a lot of localization algorithms have been proposed including the range-based (e.g., TOA (Hofmann-Wellenhof et al., 1993), TDOA (Pirzadeh, 1999) and RSSI (Bahl and Padmanabhan, 2000)) and the range-free (e.g., DV-HOP (Niculescu and Nath, 2003) and APIT (He et al., 2003)) localization algorithms. However, none of them can provide the accurate locations and therefore inevitably has location errors.

The existence of location errors can significantly affect the quality of barrier coverage provided by sensor networks. In reality, we can only know the measured locations instead of true locations of sensors. As shown in Figure 4.1(a), although node a and node b actually overlap with each other, due to the location errors, we



(a) overlap → no overlap **(b)** no overlap → overlap

Figure 4.1: The effects of location errors. According to the measured locations, (a): two overlapping sensors are considered as no overlapping; (b): two no overlapping sensors are considered as overlapping

think they do not overlap and need to deploy more mobile sensors between them to prevent intruder from crossing without being detected, which increases the cost of deployment. In contrast, as shown in Figure 4.1(b), based on the measured locations, we think node a and node b overlap with each other and all intruders crossing the line segment ab can be detected. However, since they actually do not overlap, intruders can cross the line segment ab without being detected. Therefore, location errors cannot only increase the cost of node deployment but also increase the miss rate of intruders.

A lot of work has been done on barrier coverage, however, little considers the effects of location errors of sensors. In this chapter, we study the barrier coverage problem when sensors have location errors.

First, how can we know whether the network provide barrier coverage or not after initial random deployment when sensors have location errors? The problem is challenging because the true locations of sensors are unknown. Even the network with measured locations provide barrier coverage for the ROI, it does not mean the network really can. Therefore, it is necessary to find an efficient way to decide whether the network provides barrier coverage or not with a guarantee. When the ROI is not barrier covered, mobile sensors can be deployed to form barrier coverage. However, the manufacturing cost of mobile sensors are usually more expensive than stationary sensors, which demands the usage of as few mobile sensors as possible. Therefore, the second problem is to find the minimum number of mobile sensors needed to form barrier coverage when sensors have location errors. To solve this problem, we need to first find the minimum number of mobile sensors needed to connect two stationary sensors with a guarantee when sensors have location errors, which is challenging because the number of mobile sensors calculated from the measured locations may not be enough in reality. Moreover, there are too many ways of deploying mobile sensors to form barrier coverage and how to find the optimal way using the minimum number of mobile sensors is also challenging.

In this chapter, we systematically address these problems and the main contributions are summarized as follows:

- To the best of our knowledge, our work is the first to explore the effects of location errors on barrier coverage.
- We theoretically analyze the relationship between the true distance and the measured distance of two stationary sensors, and derive the minimum number of mobile sensors needed to connect two stationary sensors with a guarantee when sensors have location errors.
- We propose a fault tolerant weighted barrier graph to model the barrier coverage formation problem, based on which we prove that the minimum number of mobile sensors needed to form barrier coverage with a guarantee is the length of the shortest path on the graph.

The remainder of this chapter is organized as follows. We present the system model in Section 4.2. We study the barrier coverage problem when only stationary sensors have location errors in Section 4.3 and the barrier coverage problem when both stationary and mobile sensors have location errors in Section 4.4. The performance evaluation of our work is presented in Section 4.5. Finally, we conclude this chapter in Section 4.6.

4.2 System Model

We assume that the ROI is a two-dimensional rectangular belt area and n stationary sensors are randomly deployed in the ROI. The belt region with the length of L and the width of H is generally a long and thin strip. A crossing path is a path that crosses the complete width of the area (e.g. path a in Figure 2.1). A congruent crossing path is a special crossing path that is orthogonal to the upper and lower boundaries of the belt region (e.g., dashed lines in Figure 2.1). An intruder may attempt to penetrate the area along any crossing path.

We assume that stationary and mobile sensors have the same type of sensors, but mobile sensors have the ability to move. We adopt the most commonly used disk model for the sensing ability of sensors, and assume that all sensors have the same sensing range, denoted by r_s. That is, when an intruder is within the distance of r_s of a sensor node, the sensor node can detect the intruder; otherwise, the sensor node cannot detect the intruder. Let $s_i = (x_i, y_i, r_s)$ denote the sensor node i whose true location is $l_i = (x_i, y_i)$. Each node can obtain its location by using suitable localization algorithms, which is called the measured location, denoted by $\tilde{l}_i = (\tilde{x}_i, \tilde{y}_i)$ for s_i. Thus, $d(l_i, \tilde{l}_i)$ is called the location error for s_i, where $d(\cdot)$ represents the Euclidean distance. We assume that the location error is upper bounded by δ where $\delta < r_s$.

The notations used throughout this chapter are summarized in Table 4.1.

4.3 Barrier Coverage When Stationary Sensors have Location Errors

In this section, we consider that only stationary sensors have location errors. We assume that mobile sensors are equipped with GPS receivers, so that they can accurately know their locations without errors. For this case, we first analyze the effects of location errors on the minimum number of mobile sensors needed to connect a pair of stationary sensors, and then propose a progressive method that uses exactly

Table 4.1: Summary of notations

Symbol	Description
L	the length of the belt region
H	the width of the belt region
n	the number of deployed stationary sensors
r_s	the sensing range of each sensor node
δ	the upper bound of location errors and $\delta < r_s$
s_i	the ith stationary sensor node
l_i	$l_i = (x_i, y_i)$ the true location of s_i
$\tilde{l}_i$	$\tilde{l}_i = (\tilde{x}_i, \tilde{y}_i)$ the measured location of s_i
R_i	Location region of s_i
$d(l_i, l_j)$	the true distance between s_i and s_j
$d(\tilde{l}_i, \tilde{l}_j)$	the measured distance between s_i and s_j
$N(s_i, s_j)$	the true minimum number of mobile sensors needed to connect s_i and s_j
$N_s^u(s_i, s_j)$	the upper bound of $N(s_i, s_j)$ when only stationary sensors have location errors
$N_s^l(s_i, s_j)$	the lower bound of $N(s_i, s_j)$ when only stationary sensors have location errors
$N_{sm}^u(s_i, s_j)$	the upper bound of $N(s_i, s_j)$ when both stationary and mobile sensors have location errors
$N_{sm}^l(s_i, s_j)$	the upper bound of $N(s_i, s_j)$ when both stationary and mobile sensors have location errors

the upper bound of the true minimum number of mobile sensors needed to connect a pair of stationary sensors with a guarantee. Finally, we model the barrier coverage problem as a fault tolerant weighted barrier graph and prove that the minimum number of mobile sensors needed to form barrier coverage with a guarantee is the length of the shortest path on the graph.

4.3.1 Minimum Number of Mobile Sensors Needed to Connect Two Stationary Sensors

The basis of barrier coverage is to decide whether two sensors overlap or not and how many mobile sensors are needed when they do not overlap. The problem is easy to answer if each node knows its true location. For example, given two sensors s_i and s_j and their true locations l_i and l_j, they overlap with each other if $d(l_i, l_j) \leq 2r_s$. When

$d(l_i, l_j) > 2r_s$, s_1 and s_2 do not overlap with each other and the minimum number of mobile sensors needed to connect them, denoted by $N(s_i, s_j)$, is $\lceil \frac{d(l_i,l_j)-2r_s}{2r_s} \rceil$.

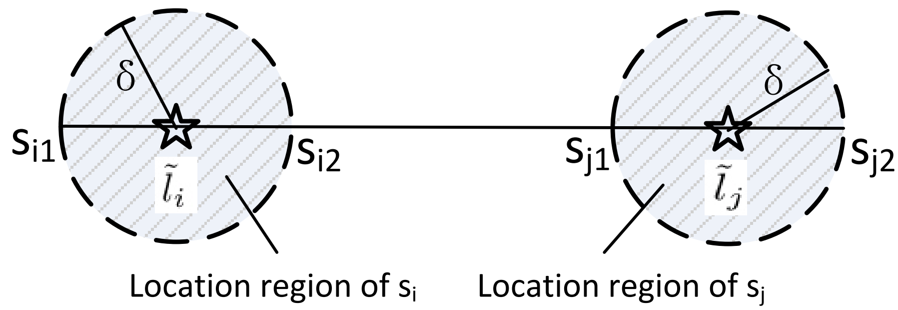


Figure 4.2: The location region of a sensor node given its measured location

However, each node does not know its true location but instead the measured location. Suppose the measured locations for s_i and s_j are $\tilde{l}_i$ and $\tilde{l}_j$, respectively. As shown in Figure 4.2, given a measured location, the true location is within the shaded circle centered at the measured location with the radius of δ, where δ is the upper bound of location errors. We call the shaded circle centered at $\tilde{l}_i$ as the location region of s_i, denoted by R_i. Given the measured location $\tilde{l}_i$, we know that the true location of s_i is in the location region R_i. Therefore,

$$\max(0, d(\tilde{l}_i, \tilde{l}_j) - 2\delta) \leq d(l_i, l_j) \leq d(\tilde{l}_i, \tilde{l}_j) + 2\delta \tag{4.1}$$

Lemma 4.0.1. *Given two stationary sensors s_i and s_j and their measured locations $\tilde{l}_i$ and $\tilde{l}_j$, s_i and s_j overlap with each other with a guarantee when $d(\tilde{l}_i, \tilde{l}_j) + 2\delta \leq 2r_s$.*

Proof. According to Equation (4.1), when $d(\tilde{l}_i, \tilde{l}_j) + 2\delta \leq 2r_s$, the true distance $d(l_i, l_j) \leq 2r_s$, so s_i and s_j overlap with each with a guarantee. □

Lemma 4.0.2. *Given two stationary sensors s_i and s_j and their measured locations $\tilde{l}_i$ and $\tilde{l}_j$, the minimum number of mobile sensors needed to guarantee the connection of s_i and s_j is $\lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil - 1$.*

Proof. Recall that $N(s_i, s_j) = \lceil \frac{d(l_i,l_j)-2r_s}{2r_s} \rceil$ denotes the true minimum number of mobile sensors needed to connect s_i and s_j. According to Equation (4.1), we have

$$\max(0, \lceil \frac{d(\tilde{l}_i, \tilde{l}_j) - 2\delta}{2r_s} \rceil - 1) \leq N(s_i, s_j) \leq \lceil \frac{d(\tilde{l}_i, \tilde{l}_j) + 2\delta}{2r_s} \rceil - 1$$

In order to guarantee the connection of s_i and s_j, at least $\lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil - 1$ mobile sensors are needed. $\square$

Let $N_s^u(s_i, s_j)$ and $N_s^l(s_i, s_j)$ denote the upper and lower bound of $N(s_i, s_j)$, respectively. That is, $N_s^u(s_i, s_j) = \lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil - 1$ and $N_s^l(s_i, s_j) = \max(0, \lceil \frac{d(\tilde{l}_i,\tilde{l}_j)-2\delta}{2r_s} \rceil - 1)$. Thus, $\triangle N_s(s_i, s_j) = N_s^u(s_i, s_j) - N_s^l(s_i, s_j)$ represents the influence of location error on the minimum number of mobile sensors needed. When $\triangle N_s(s_i, s_j) = 0$, $N(s_i, s_j) = N_s^u(s_i, s_j) = N_s^l(s_i, s_j)$ and therefore the location error would not affect the minimum number of mobile sensors needed to connect s_i and s_j.

Theorem 4.1. *Given two stationary sensors s_i and s_j and their measured locations $\tilde{l}_i$ and $\tilde{l}_j$, the location error does not affect the minimum number of mobile sensors needed to connect s_i and s_j when $d(\tilde{l}_i, \tilde{l}_j) + 2\delta \leq 2r_s$ or $\lceil \frac{d(\tilde{l}_i,\tilde{l}_j)-2\delta}{2r_s} \rceil = \lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil \geq 2$.*

Proof. When $N_s^u(s_i, s_j) = N_s^l(s_i, s_j) = 0$, $\lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil - 1$ should be 0. Therefore, $d(\tilde{l}_i, \tilde{l}_j) + 2\delta \leq 2r_s$ is required.

When $N_s^u(s_i, s_j) = \lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil - 1 = N_s^l(s_i, s_j) = \lceil \frac{d(\tilde{l}_i,\tilde{l}_j)-2\delta}{2r_s} \rceil - 1 > 0$, $\lceil \frac{d(\tilde{l}_i,\tilde{l}_j)-2\delta}{2r_s} \rceil = \lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil = k$ is required where k is an integer and $k \geq 2$.

Therefore, when any one of them is satisfied, the location error does not affect the minimum number of mobile sensors needed to connect s_i and s_j. $\square$

Theorem 4.2. *Given a sensor network where only stationary sensors have location errors upper bounded by $\delta < r_s$, at most 2 more mobile sensors are needed to connect any pair of stationary sensors compared to the true minimum number of mobile sensors needed. That is, $\triangle N_s(s_i, s_j) \leq 2$ for any pair of s_i and s_j when $\delta < r_s$.*

Proof. $\triangle N_s(s_i, s_j)$ represents the influence of location error on the minimum number of mobile sensors needed. We prove the theorem from the following cases.

Case 1: When $d(\tilde{l}_i, \tilde{l}_j) + 2\delta \leq 2r_s$, according to Lemma 4.0.1, $\triangle N_s(s_i, s_j) = 0$.

Case 2: When $d(\tilde{l}_i, \tilde{l}_j)+2\delta > 2r_s$ and $d(\tilde{l}_i, \tilde{l}_j)-2\delta \le 2r_s$, $N_s^u(s_i, s_j) = \lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil - 1$ and $N_s^l(s_i, s_j) = 0$. Therefore $\triangle N_s(s_i, s_j) = \lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil - 1$. Since $d(\tilde{l}_i, \tilde{l}_j) + 2\delta > 2r_s$, $\lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil \ge 2$. Since $d(\tilde{l}_i, \tilde{l}_j) - 2\delta \le 2r_s$, $d(\tilde{l}_i, \tilde{l}_j) + 2\delta \le 2r_s + 4\delta < 6r_s$ and then $\lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil \le 3$. Therefore, $1 \le \triangle N_s(s_i, s_j) \le 2$.

Case 3: When $d(\tilde{l}_i, \tilde{l}_j) + 2\delta > 2r_s$ and $d(\tilde{l}_i, \tilde{l}_j) - 2\delta > 2r_s$, we have

$$\begin{aligned}\triangle N_s(s_i, s_j) &= \lceil \frac{d(\tilde{l}_i, \tilde{l}_j) + 2\delta}{2r_s} \rceil - \lceil \frac{d(\tilde{l}_i, \tilde{l}_j) - 2\delta}{2r_s} \rceil \\ &< \lceil \frac{d(\tilde{l}_i, \tilde{l}_j) + 2r_s}{2r_s} \rceil - \lceil \frac{d(\tilde{l}_i, \tilde{l}_j) - 2r_s}{2r_s} \rceil = 2\end{aligned}$$

In all cases, $\triangle N_s(s_i, s_j) \le 2$ which means that at most 2 more mobile sensors are needed compared to $N(s_i, s_j)$ when only stationary sensors have location errors. □

4.3.2 Progressive Mobile Node Deployment

For any two known true locations of s_i and s_j within R_i and R_j respectively, $N_s^u(s_i, s_j)$ is enough to connect them with a guarantee. However, the difficulty is that the true locations are unknown in reality, so deploying $N_s^u(s_i, s_j)$ mobile sensors derived from the largest distance of two known true locations may not be able to connect s_i and s_j with a guarantee. To solve this problem, we propose a progressive method to use as few mobile sensors as possible to connect two stationary sensors with a guarantee.

The basic idea of the progressive method is to deploy mobile sensors progressively from the left stationary node to the right stationary node. Given two stationary sensors s_i and s_j and their measured locations $\tilde{l}_i$ and $\tilde{l}_j$, the progressive method is described as follows:

- Step 1: Deploy a mobile node on the line segment $\tilde{l}_i\tilde{l}_j$ to make it overlap with all sensors located within the location region of s_i and the distance between the mobile node and $\tilde{l}_i$ maximized.

- Step 2: Check whether the new deployed mobile node overlap with all sensors located within the location region of s_j or not. If yes, stop; otherwise, go to step 3.

- Step 3: Deploy a new mobile node on the line segment $\tilde{l}_i\tilde{l}_j$ that is $2r_s$ away from the previously deployed mobile node, go to step 2.

Suppose the first deployed mobile node is denoted by m_k and its expected location is $l_k = (x_k, y_k)$. According to Step 1, we have

$$\begin{aligned} \text{Maximize} \quad & d(\tilde{l}_i, l_k) = \sqrt{(\tilde{x}_i - x_k)^2 + (\tilde{y}_i - y_k)^2} \\ \text{subject to} \quad & \sqrt{(\tilde{x}_i - x_i)^2 + (\tilde{y}_i - y_i)^2} \leq \delta \\ & \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2} \leq 2r_s \\ & (y_k - \tilde{y}_i)(\tilde{x}_j - x_k) = (\tilde{y}_j - y_k)(x_k - \tilde{x}_i) \end{aligned}$$

The objective is to maximize the distance between the $\tilde{l}_i$ and l_k. The first constraint indicates that the location error between the true and measured location is no larger than δ, and the second constraint indicates that m_k should overlap with s_i no matter where the true location of s_i is, and finally the third one restricts m_k on the line segment $\tilde{l}_i\tilde{l}_j$.

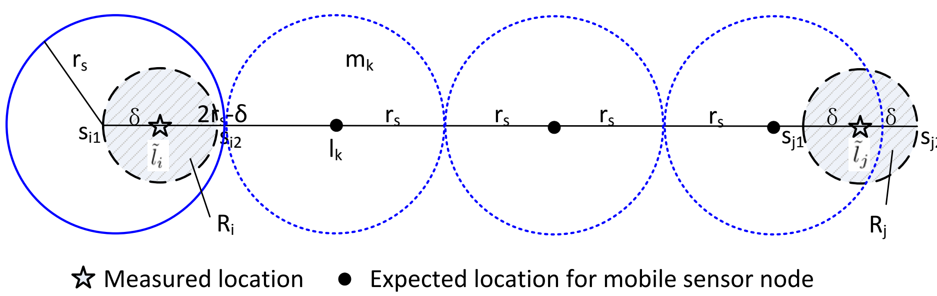


Figure 4.3: An illustration of the progressive method. The blue solid circle with radius of r_s denotes the sensing region of s_i located at s_{i1}. The blue dashed circles with radius of r_s denote the sensing regions of mobile sensors.

As shown in Figure 4.3, s_{i1} and s_{i2} are the two intersections of line $\tilde{l}_i\tilde{l}_j$ and the location region of s_i, R_i. According to geometry, for any point p on line segment $\tilde{l}_i\tilde{l}_j$, the largest distance from the point p to any point within R_i is the distance from the point p to the point s_{i1}. In other words, if the mobile node at point l_k overlaps with a node at point s_{i1}, it overlaps all the sensors located within R_i. As m_k moves along $\tilde{l}_i\tilde{l}_j$, both $d(s_{i1}, l_k)$ and $d(\tilde{l}_i, l_k)$ increase accordingly. When $d(\tilde{l}_i, l_k) = 2r_s - \delta$, m_k cannot move further since any further movement would not guarantee the overlap of m_k and s_i. Therefore, the maximum of $d(\tilde{l}_i, l_k)$ is $2r_s - \delta$. When more mobile sensors are required, they will be added one by one with the interval of $2r_s$ until a mobile node overlaps with all sensors located within R_j.

Theorem 4.3. *The progressive method is an optimal way that connects s_i and s_j with a guarantee by using $\lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil - 1$ mobile sensors.*

Proof. We first prove that the progressive method uses $\lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil - 1$ mobile sensors to guarantee the connection of s_i and s_j, and then prove that it is optimal.

When $d(\tilde{l}_i, \tilde{l}_j) + 2\delta \leq 2r_s$, no mobile node is needed. When $d(\tilde{l}_i, \tilde{l}_j) + 2\delta > 2r_s$, mobile sensors should be deployed. In the progressive method, for the first mobile node m_k, $d(\tilde{l}_i, l_k) = 2r_s - \delta$ and therefore $d(s_{i1}, l_k) = 2r_s$. Thus, we deploy the first mobile node $2r_s$ away from s_{i1} on $\tilde{l}_i\tilde{l}_j$, and then deploy other mobile sensors one by one with the interval of $2r_s$ until the distance between a mobile node and s_{j2} is not larger than $2r_s$. Therefore, the number of mobile sensors needed in the progressive method is $\lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil - 1$.

In Lemma 4.0.2, we proved that at least $\lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil - 1$ mobile sensors are needed to connect s_i and s_j with a guarantee. Since the progressive method uses exactly $\lceil \frac{d(\tilde{l}_i,\tilde{l}_j)+2\delta}{2r_s} \rceil - 1$ mobile sensors, it is an optimal way of deploying mobile sensors. $\square$

4.3.3 Minimum Number of Mobile Sensors Needed to Form Barrier Coverage

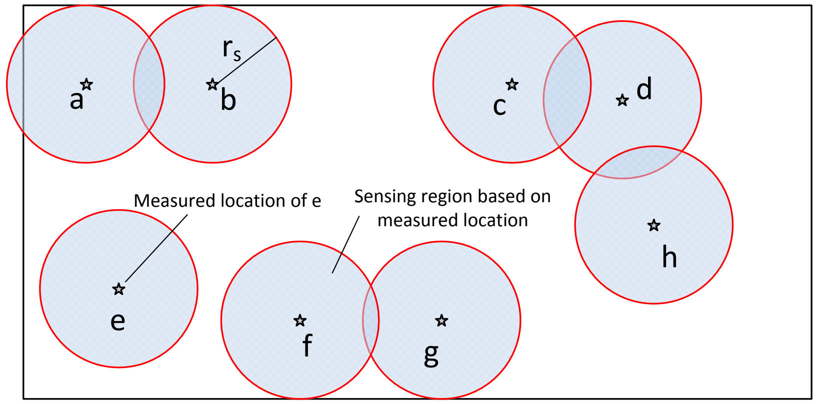
Mobile sensors can be deployed between stationary sensors to fill in gaps to form a barrier. However, there are too many ways to deploy mobile sensors and how to find the optimal way using the minimum number of mobile sensors is challenging. In this subsection, we will model the barrier coverage formation problem with location errors as a fault tolerant weighted barrier graph and use it to find the minimum number of mobile sensors needed to form barrier coverage with a guarantee.

Definition 9. *A* **Fault tolerant weighted barrier graph** $G = (V, E, W)$ *of a sensor network is constructed as follows. The set V consists of vertices corresponding to the left boundary (s), all the stationary sensors (S) and the right boundary (t) of the belt region, that is, $V = \{v_1, v_2, \cdots, v_{n+2}\} = \{s \cup S \cup t\}$. $E = \{e(v_i, v_j)\}$ is the set of edges between any pair of vertices. $W : E \to \mathbb{R}$ is the set of weights of each edge, where the weight $w(v_i, v_j)$ of edge $e(v_i, v_j)$ is the minimum number of mobile sensors needed to guarantee the connection of v_i and v_j.*

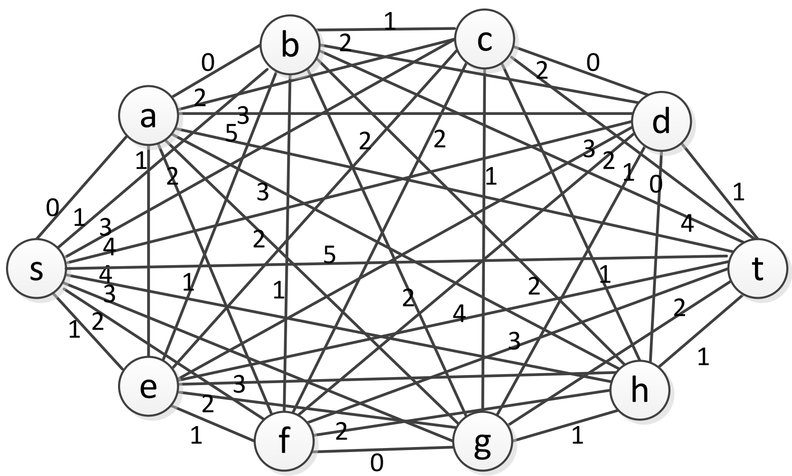
According to Theorem 4.3, in order to guarantee the connection of s_i and s_j, $\lceil \frac{d(\tilde{l}_i, \tilde{l}_j)+2\delta}{2r_s} \rceil - 1$ mobile sensors should be deployed and therefore $w(s_i, s_j) = \lceil \frac{d(\tilde{l}_i, \tilde{l}_j)+2\delta}{2r_s} \rceil - 1$. For a node s_j, the maximum distance between it and the left boundary s is $\tilde{x}_j + \delta$, where $\tilde{x}_j$ is x-coordinate of the measured location of s_j. In order to guarantee the connection of them, $w(s, s_j) = \lceil \frac{\tilde{x}_j+\delta-r_s}{2r_s} \rceil$ mobile sensors are needed. Also, the maximum distance between s_j and the right boundary t is $L - (\tilde{x}_j - \delta)$. In order to guarantee the connection of them, $w(t, s_j) = \lceil \frac{L-(\tilde{x}_j-\delta+r_s)}{2r_s} \rceil$ mobile sensors are needed. We can also deploy mobile sensors directly from the left boundary to the right boundary, and the minimum number of mobile sensors needed to connect s and t is $w(s, t) = \lceil \frac{L}{2r_s} \rceil$. In summary, we have

$$w(v_i, v_j) = \begin{cases} \lceil \frac{d(\tilde{l}_i, \tilde{l}_j)+2\delta}{2r_s} \rceil - 1 & \text{if } v_i = s_i \text{ and } v_j = s_j \\ \lceil \frac{\tilde{x}_j+\delta-r_s}{2r_s} \rceil & \text{if } v_i = s \text{ and } v_j = s_j \\ \lceil \frac{L-(\tilde{x}_j-\delta+r_s)}{2r_s} \rceil & \text{if } v_i = t \text{ and } v_j = s_j \\ \lceil \frac{L}{2r_s} \rceil & \text{if } v_i = s \text{ and } v_j = t \end{cases} \tag{4.2}$$

Figure 4.4 shows a deployed sensor network and its corresponding fault tolerant weighted barrier graph. s and t are the virtual vertices corresponding to the left and



(a) Deployed sensor network



(b) Fault tolerant weighted barrier graph

Figure 4.4: Sensor network and its corresponding fault tolerant weighted barrier graph when only stationary sensors have location errors ($r_s = 10m$ and $\delta = 1m$)

right boundary of the belt region. The weight of each edge is the minimum number of mobile sensors needed to guarantee the connection of the pair of vertices.

Theorem 4.4. *The minimum number of mobile sensors needed to form a barrier with a guarantee with stationary sensors is exactly the length of the shortest path from s to t on the fault tolerant weighted barrier graph G and upper bounded by $\lceil \frac{L}{2r_s} \rceil$.*

Proof. According to the definition of the fault tolerant weighted barrier graph G, if we want to form a barrier, we only need to choose a path from s to t, and put exactly the number of mobile sensors needed on each edge of the path. That is, for a chosen path, the number of mobile sensors required to form a barrier with a guarantee is equal to the sum of weights of all edges on the path, which is the length of the path. Therefore, the minimum number of mobile sensors required to form a barrier with a guarantee is the length of the shortest path from s to t on graph G.

The path containing only the edge $e(s,t)$ could either be the shortest or not. If it is not the shortest path, the minimum number of mobile sensors required is smaller than $w(s,t)$; otherwise, the minimum number of mobile sensors required is equal to $w(s,t)$. Therefore, the minimum number of mobile sensors required to form a barrier with a guarantee is always upper bounded by $w(s,t) = \lceil \frac{L}{2r_s} \rceil$. □

Theorem 4.5. *The ROI is guaranteed to be barrier covered after initial deployment of sensors if the length of the shortest path from s to t on the fault tolerant weighted barrier graph G equals zero.*

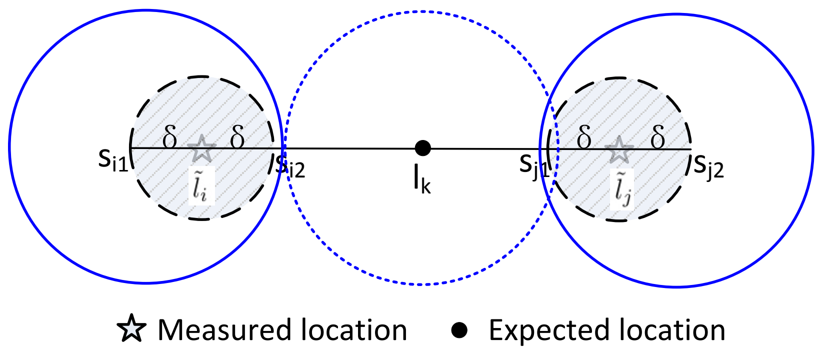
Proof. $\Rightarrow$. If the length of the shortest path from s to t on G equals zero, the shortest path is a barrier that does not need any mobile node. Therefore, the ROI is guaranteed to be barrier covered.

$\Leftarrow$. If the ROI is barrier covered with a guarantee by the sensor network, there exists a barrier (path) on the graph G and no mobile sensor node is needed between any two adjacent vertices on the path. Therefore, the length of the shortest path from s to t on G equals zero. □

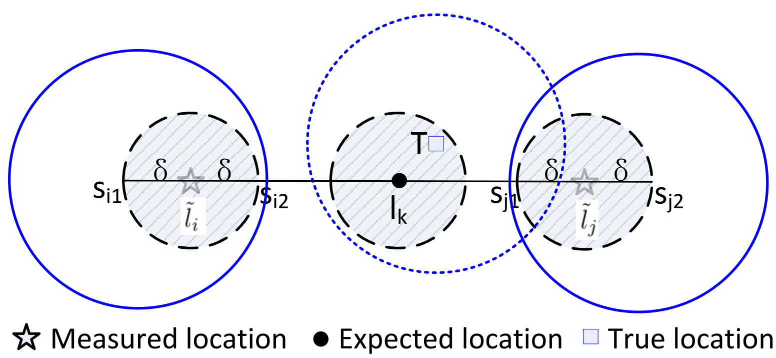
According to Theorem 4.4, we can use the classical Dijkstra's algorithm (Cormen et al., 2009) to find the minimum number of mobile sensors needed to form barrier coverage with a guarantee and check whether the ROI is guaranteed to be barrier covered or not after initial deployment. As shown in Figure 4.4, the shortest path is $s \to a \to b \to c \to d \to t$, the length of which is $0+0+1+0+1=2$. Therefore, the ROI is not guaranteed to be barrier covered after initial random deployment and 2 mobile sensors are needed to deploy between b and c, and d and the right boundary to guarantee the formation of barrier coverage.

4.4 Barrier Coverage When both Stationary and Mobile Sensors have Location Errors

In this section, we consider that not only stationary sensors but also mobile sensors have location errors. The location error of mobile node is also assumed to be upper bounded by $\delta < r_s$.



(a) The mobile node does not have error



(b) The mobile node has error

Figure 4.5: The effect of location error for mobile sensors

The barrier coverage problem is more complicated when mobile sensors also have location errors. This is because although the measured location of a mobile node shows to be the expected location, due to the location error, the true location of the node may not be the expected location. As shown in Figure 4.5(a), when the mobile node m_k does not have location error, it can move to the expected location l_k and connect s_i and s_j with a guarantee. However, when the node has a location error, as shown in Figure 4.5(b), although the measured location is l_k, the true location is actually at point T (denoted by the blue square) which cannot guarantee the connection of s_i and s_j.

Lemma 4.5.1. *Given two stationary sensors s_i and s_j and their measured locations $\tilde{l}_i$ and $\tilde{l}_j$, the minimum number of mobile sensors needed to guarantee the connection of s_i and s_j is $\lceil \frac{d(\tilde{l}_i,\tilde{l}_j)}{2r_s-2\delta} \rceil - 1$ when both stationary and mobile sensors have location errors.*

Proof. Since both the stationary and mobile sensors have location errors upper bounded by $\delta < r_s$, according to Lemma 4.0.1, two sensors (either be stationary or mobile sensors) overlap with each other with a guarantee only if their measured distance is no larger than $2r_s - 2\delta$. Therefore, the distance between two expected locations of two mobile sensors should not be larger than $2r_s - 2\delta$, otherwise they may not overlap with each other. Thus, in order to use as few mobile sensors as possible, the expected locations should be on the line segment $\tilde{l}_i\tilde{l}_j$ with an interval of $2r_s - 2\delta$. Therefore, the minimum number of mobile sensors needed to guarantee the connection of s_i and s_j is $\lceil \frac{d(\tilde{l}_i,\tilde{l}_j)}{2r_s-2\delta} \rceil - 1$ when both stationary and mobile sensors have location errors. $\square$

Recall that $N(s_i, s_j)$ is the true minimum number of mobile sensors needed to connect s_i and s_j. Let $N^u_{sm}(s_i, s_j)$ and $N^l_{sm}(s_i, s_j)$ denote the upper and lower bound of $N(s_i, s_j)$ when both stationary and mobile sensors have location errors. According to Lemma 4.5.1, $N^u_{sm}(s_i, s_j) = \lceil \frac{d(\tilde{l}_i,\tilde{l}_j)}{2r_s-2\delta} \rceil - 1$. According to Equation (4.1), the lower bound is $N^l_{sm}(s_i, s_j) = \max(0, \lceil \frac{d(\tilde{l}_i,\tilde{l}_j)-2\delta}{2r_s} \rceil - 1)$. Thus, $\triangle N_{sm}(s_i, s_j) = N^u_{sm}(s_i, s_j) - N^l_{sm}(s_i, s_j)$ represents the influence on $N(s_i, s_j)$ when both stationary and mobile sensors have location errors. When $\triangle N_{sm}(s_i, s_j) = 0$, the location error does not affect the minimum number of mobile sensors needed.

Theorem 4.6. *Considering a sensor network where both stationary and mobile sensors have location errors upper bounded by $\delta < r_s$, at most* $\max(\lceil \frac{4\delta}{2r_s-2\delta} \rceil, \lceil \frac{\delta d(\tilde{l}_i,\tilde{l}_j)+2\delta r_s-2\delta^2}{r_s(2r_s-2\delta)} \rceil)$

more mobile sensors are needed to connect s_i and s_j compared to the true minimum number of mobile sensors needed. That is, $\triangle N_{sm}(s_i, s_j) \leq \max(\lceil \frac{4\delta}{2r_s - 2\delta} \rceil, \lceil \frac{\delta d(\tilde{l}_i, \tilde{l}_j) + 2\delta r_s - 2\delta^2}{r_s(2r_s - 2\delta)} \rceil)$.

Proof. When $d(\tilde{l}_i, \tilde{l}_j) - 2\delta \leq 2r_s$, $\triangle N_{sm}(s_i, s_j) = \lceil \frac{d(\tilde{l}_i, \tilde{l}_j)}{2r_s - 2\delta} \rceil - 1 \leq \lceil \frac{2r_s + 2\delta}{2r_s - 2\delta} \rceil - 1 = \lceil \frac{4\delta}{2r_s - 2\delta} \rceil$.

When $d(\tilde{l}_i, \tilde{l}_j) - 2\delta > 2r_s$, $\triangle N_{sm}(s_i, s_j) = \lceil \frac{d(\tilde{l}_i, \tilde{l}_j)}{2r_s - 2\delta} \rceil - \lceil \frac{d(\tilde{l}_i, \tilde{l}_j) - 2\delta}{2r_s} \rceil \leq \lceil \frac{\delta d(\tilde{l}_i, \tilde{l}_j) + 2\delta r_s - 2\delta^2}{r_s(2r_s - 2\delta)} \rceil$.

Therefore, at most $\max(\lceil \frac{4\delta}{2r_s - 2\delta} \rceil, \lceil \frac{\delta d(\tilde{l}_i, \tilde{l}_j) + 2\delta r_s - 2\delta^2}{r_s(2r_s - 2\delta)} \rceil)$ more mobile sensors are needed when both stationary and mobile sensors have location errors compared to the true minimum number of mobile sensors needed to connect any pair of stationary sensors with a guarantee. □

According to Theorem 4.2, at most 2 more mobile sensors are needed when only stationary sensors have location errors. However, according to Theorem 4.6, $\triangle N_{sm}(s_i, s_j)$ is related with the measured distance and δ when both stationary and mobile sensors have location errors. As δ or the measured distance increases, more mobile sensors will be needed. Therefore, the existence of location error for mobile sensors could significantly influence the minimum number of mobile sensors needed to form barrier coverage.

In order to find the minimum number of mobile sensors needed to form barrier coverage with a guarantee, we can also build a corresponding fault tolerant barrier graph for the sensor network. Similar to the graph in Section 4.3.3, the left and right boundary are considered as virtual vertices s and t, respectively. Each stationary node is modeled as a vertex. There is an edge between any pair of vertices and a weight is assigned for each edge which represents the minimum number of mobile sensors needed to connect any pair of vertices with a guarantee.

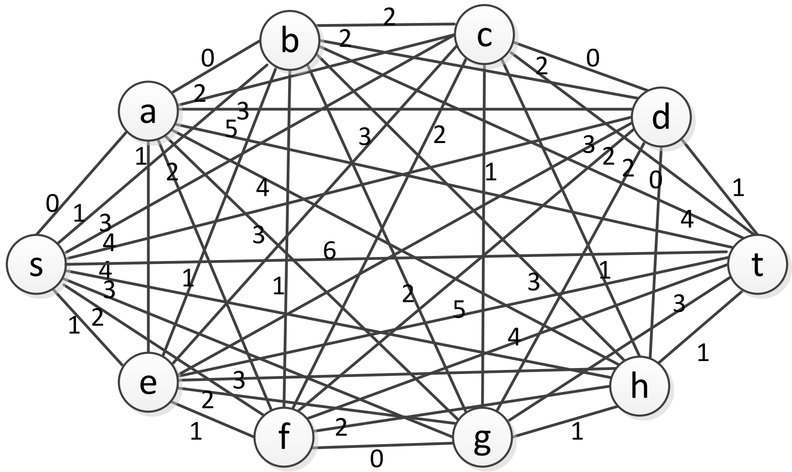


Figure 4.6: The fault tolerant weighted barrier graph corresponding to Figure 4.4(a) when both stationary and mobile sensors have location errors

Since mobile sensors also have location errors, the weight of each edge is not the same as that in Equation (4.2). Similar to the derivation for Equation (4.2), we have

$$w(v_i, v_j) = \begin{cases} \lceil \frac{d(\tilde{l}_i, \tilde{l}_j)}{2r_s - 2\delta} \rceil - 1 & \text{if } v_i = s_i \text{ and } v_j = s_j \\ \lceil \frac{\tilde{x}_j - (r_s - \delta)}{2r_s - 2\delta} \rceil & \text{if } v_i = s \text{ and } v_j = s_j \\ \lceil \frac{L - \tilde{x}_j - (r_s - \delta)}{2r_s - 2\delta} \rceil & \text{if } v_i = t \text{ and } v_j = s_j \\ \lceil \frac{L}{2r_s - 2\delta} \rceil & \text{if } v_i = s \text{ and } v_j = t \end{cases} \tag{4.3}$$

Theorem 4.7. *The minimum number of mobile sensors needed to form a barrier with a guarantee with stationary sensors when both stationary and mobile sensors have location errors is upper bounded by* $\lceil \frac{L}{2r_s - 2\delta} \rceil$.

Proof. The proof is omitted because it is similar to the proof of Theorem 4.4. □

Figure 4.6 shows the fault tolerant weighted barrier graph when both stationary and mobile sensors have location errors. Note that the only difference between this figure and Figure 4.4(b) is the weight of each edge representing the minimum number of mobile sensors needed to connect any pair of vertices with a guarantee. As shown in Figure 4.6, the shortest path is $s \to a \to b \to d \to t$, the length of which is

$0 + 0 + 2 + 1 = 3$. Therefore, the ROI is not guaranteed to be barrier covered after initial random deployment and 3 mobile sensors are needed to guarantee the formation of barrier coverage.

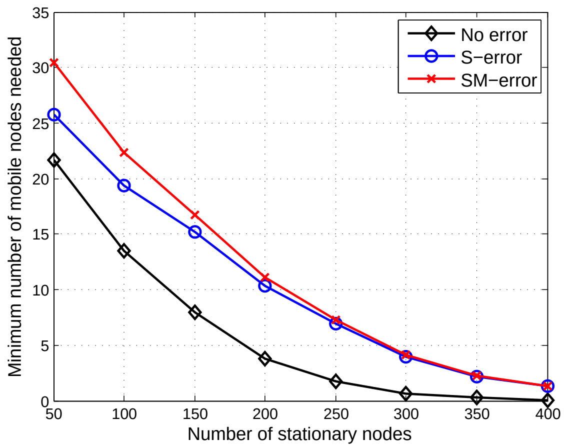
4.5 Performance Evaluation

In this section, we conduct simulations to evaluate the effects of location errors on barrier coverage. The ROI is a belt region of length $L = 1000m$ and width $W = 100m$. Initially, stationary sensors are randomly deployed in the ROI. After the minimum number of mobile sensors is calculated, mobile sensors are deployed to form barrier coverage. The evaluation mainly focuses on three metrics: the minimum number of mobile sensors needed to form barrier coverage, the total cost needed to form barrier coverage, and the influence of location error on the minimum number of mobile sensors to connect any pair of stationary sensors.

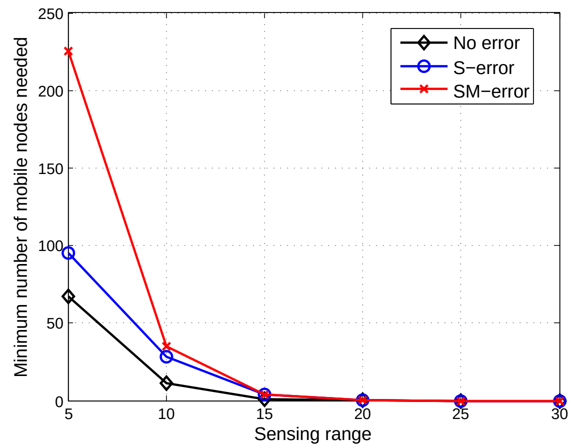
We evaluate the number of stationary sensors, the sensing range and δ for these metrics. For all the simulation results in Figure 4.7 and 4.8, each data point is an average of 100 experiments. For all the simulation results in Figure 4.9, each data point is the maximum value of 100 experiments.

4.5.1 Minimum Number of Mobile Sensors Needed

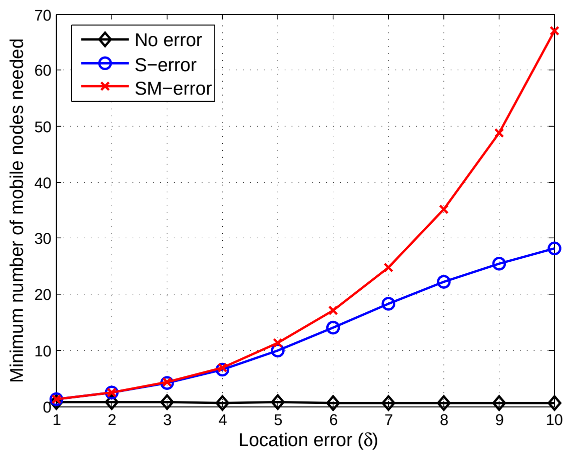
Figure 4.7 shows the effects of different parameters on the minimum number of mobile sensors needed to form barrier coverage with a guarantee. As shown in Figure 4.7(a) and (b), we can see that the minimum number of mobile sensors needed decreases as the number or the sensing range of sensors increases. This is because more number of stationary sensors deployed or larger sensing range can reduce the number of gaps between stationary sensors as well as the sizes of gaps. From Figure 4.7(c), we also observe that the minimum number of mobile sensors needed increases when the location error increases. This is because larger location error results in larger



(a) $r_s = 15m$ and $\delta = 3m$



(b) $n = 300$ and $\delta = 3m$



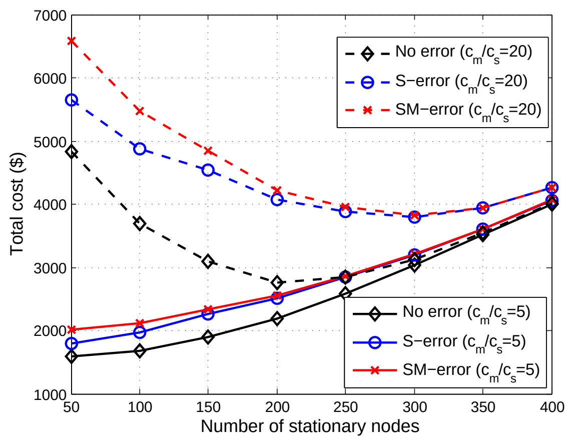
(c) $n = 300$ and $r_s = 15m$

Figure 4.7: The effects of different parameters on the minimum number of mobile sensors needed. "No error" means that sensors do not have location error, "S-error" means that only stationary sensors have location error, and "SM-error" means that both stationary and mobile sensors have location error

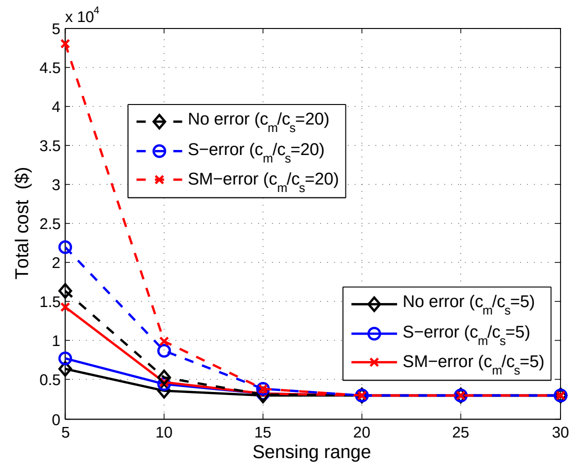
instability of a location and therefore requires more mobile sensors. We can also observe that the required number of mobile sensors when both stationary and mobile sensors have location errors is usually larger than that when only stationary node have location errors.

4.5.2 Total Cost Needed

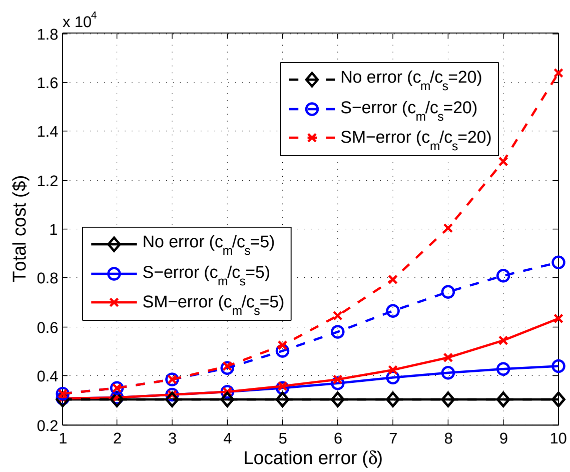
The total cost needed to form a barrier is the sum of the cost of deployed stationary sensors and the cost of mobile sensors needed. Let c_s and c_m denote the cost of a stationary node and a mobile node, respectively. For simplicity, we assume $c_s = 10\$$ for a stationary node.



(a) $r_s = 15m$ and $\delta = 3m$



(b) $n = 300$ and $\delta = 3m$



(c) $n = 300$ and $r_s = 15m$

Figure 4.8: The effects of different parameters on the total cost to form a barrier

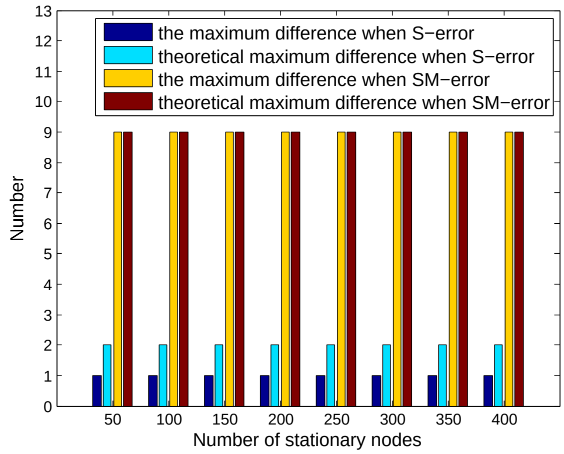
As shown in Figure 4.8(a), when mobile sensors are not very expensive (e.g., $c_m/c_s = 5$), the total cost mainly depends on the number of deployed stationary sensors. Therefore, the total cost increases as the number of deployed stationary sensors increase when $c_m/c_s = 5$. However, when mobile sensors are much more expensive than stationary sensors (e.g., $c_m/c_s = 20$), the number of mobile sensors needed can significantly affect the total cost needed. For example, the total cost for $n = 50$ is much larger than that for $n = 200$ because the former one needs much more mobile sensors to form a barrier. For the simulated belt region, the total cost reaches the minimum when 200 stationary sensors are deployed. Therefore, we can conclude that, given an ROI, the number of stationary sensors to be deployed highly depends on c_m/c_s.

We can see from Figure 4.8(b) that the total cost needed decreases when the sensing range of sensors increases, which is because the number of mobile sensors needed decreases. As shown in Figure 4.8(c), the total cost needed increases when the location error increases, which is because more mobile sensors are needed for a larger location error.

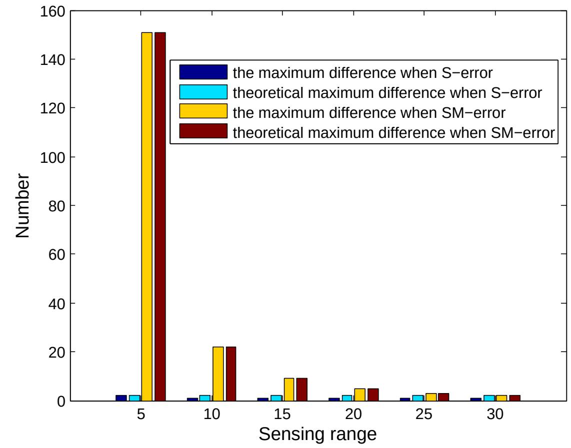
4.5.3 $\triangle N_s(s_i, s_j)$ and $\triangle N_{sm}(s_i, s_j)$

$\triangle N_s(s_i, s_j)$ and $\triangle N_{sm}(s_i, s_j)$ represents the influence of location errors on the minimum number of mobile sensors needed when only stationary sensors have location errors and when both stationary and mobile sensors have location errors, respectively. Figure 4.9 shows the effects of different parameters on $\triangle N_s(s_i, s_j)$ and $\triangle N_{sm}(s_i, s_j)$ and also their theoretical upper bound. First we can observe that the maximum of $\triangle N_s(s_i, s_j)$ when only stationary sensors have location errors is always no larger than 2, which validates the correctness of Theorem 4.2. We then observe that the maximum of $\triangle N_{sm}(s_i, s_j)$ when both stationary and mobile sensors have location errors is always no larger than its theoretical upper bound, which validates the correctness of Theorem 4.6.

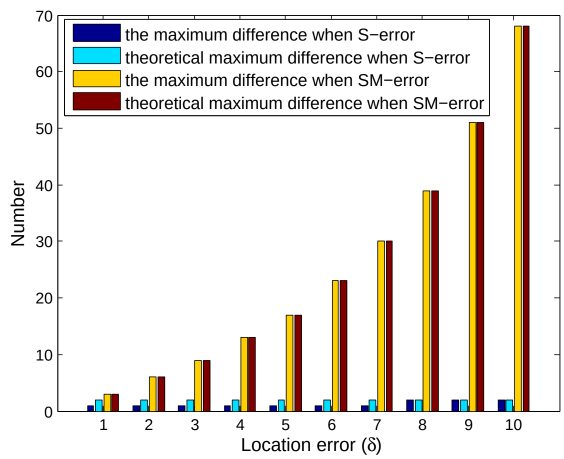
As shown in Figure 4.9(a), the maximum of $\triangle N_{sm}(s_i, s_j)$ does not change when the number of stationary sensors increases. This is because the largest distance of two stationary sensors is almost always the length of the area. Figure 4.9(b) shows



(a) $r_s = 15m$ and $\delta = 3m$



(b) $n = 300$ and $\delta = 3m$



(c) $n = 300$ and $r_s = 15m$

Figure 4.9: The influence of location errors on the minimum number of mobile sensors needed

that the maximum of $\triangle N_{sm}(s_i, s_j)$ decreases when the sensing range increases, which implies that the influence of location error is smaller for larger sensing range. Figure 4.9(c) shows that the maximum $\triangle N_{sm}(s_i, s_j)$ increases as the location error increases, which implies that the influence of location error is more and more serious when the location error become larger and larger.

4.6 Summary

In this chapter, we studied the barrier coverage problem when sensors have location errors. When only stationary sensors have location errors, we proved that at most 2 more mobile sensors are needed compared to the true minimum number of sensors needed to connect any pair of stationary sensors with a guarantee. When both stationary and mobile sensors have location errors, the difference between the minimum number of mobile sensors needed and the true minimum number of mobile sensors needed is related with the length of the belt region and the location error.

We proposed a progressive method that uses exactly the same minimum number of mobile sensors derived in theory to connect any pair of sensors with a guarantee. Furthermore, we proposed a fault tolerant weighted barrier graph and proved that the minimum number of mobile sensors needed to form barrier coverage with a guarantee is the length of the shortest path on the graph. Extensive simulation results validated the correctness of our analysis.

Chapter 5

Conclusions and Future Work

5.1 Summary

In this dissertation, we explored how to efficiently use mobile sensors to form k-barrier coverage with pre-deployed stationary sensors. The primary goal of barrier coverage is to detect intruders as they cross a border or penetrate protected areas, which makes it a critical issue for a lot of security applications, such as border protection, critical infrastructures protection and dangerous substance monitoring. Although lots of work have been done on barrier coverage, most of them mainly focus on critical condition analysis and barrier construction for stationary sensors and little effort has been made to explore how to improve barrier coverage after initial deployment. In addition, existing studies on barrier coverage only focus on homogeneous sensor network, little effort has been put on barrier coverage with heterogenous sensor networks which is more practical and useful in real-world applications. Moreover, to the best of our knowledge, none of existing work explores the effects of location errors of sensors on barrier coverage and how to guarantee the formation of barrier coverage when location errors exist.

In order to address these problems, we first presented a *cost-effective* 1-barrier coverage formation algorithm in heterogeneous sensor networks where sensor networks consist of different types of mobile sensors with different sensing ranges and costs. To the best of our knowledge, we are the first to study barrier coverage formation problem in heterogeneous sensor networks. We introduced the directional barrier graph model, and proved that the minimum cost of mobile sensors required to form a barrier is the length of the shortest path from the source node to the destination node on the graph. To efficiently assign mobile sensors to different gaps while minimizing the total moving distance, we proposed a greedy movement algorithm for heterogenous sensor network, and also a position based optimal movement algorithm for homogeneous sensor network which formulated the MCBF problem as the minimum cost bipartite assignment problem and solved it using the Hungarian algorithm. Extensive evaluation results on both homogeneous and heterogeneous sensor networks validate the effectiveness of our proposed algorithms.

Secondly, we studied the k-barrier coverage formation problem in hybrid sensor networks. To the best of our knowledge, we are the first to study how to efficiently use mobile sensors to form k-barrier coverage with pre-deployed stationary sensors. We introduced a novel weighted barrier graph (WBG) model for the barrier coverage formation problem, and proved that determining the minimum number of mobile sensors required to form k-barrier coverage is related with but not equal to finding k vertex-disjoint paths with the minimum total length on the WBG. With this observation, we proposed an optimal algorithm and a faster greedy algorithm to find the minimum number of mobile sensors required to form k-barrier coverage with pre-deployed stationary sensors. We also proposed an optimal algorithm and a faster greedy algorithm to determine the maximum number of barriers when both the stationary and mobile sensors have been pre-deployed. Both analytical and experimental studies demonstrated the effectiveness of our proposed algorithms.

Finally, we studied the barrier coverage formation problem when sensors have location errors. To the best of our knowledge, we are the first to explore the effects of location errors on barrier coverage and how to guarantee the formation of barrier coverage in this case. We analyzed the relationship between the true distance and the measured distance of two stationary sensors and proved that at most 2 more mobile sensors are needed compared to the true minimum number of sensors needed to connect any pair of stationary sensors with a guarantee when only stationary sensors have location errors. When both stationary and mobile sensors have location errors, the difference between the minimum number of mobile sensors needed and the true minimum number of mobile sensors needed is related with the length of the belt region and the location errors. We proposed a progressive method that uses exactly the same minimum number of mobile sensors derived in theory to connect any pair of sensors with a guarantee. Furthermore, we proposed a fault tolerant weighted barrier graph and proved that the minimum number of mobile sensors needed to form barrier coverage with a guarantee is the length of the shortest path on the graph. Extensive simulation results validated the correctness of our analysis.

5.2 Directions for Future Research

Although barrier coverage has been well studied in the last decade, there is still a long way to go before it can be used into real systems. In my future work, I would like to focus on applying barrier coverage into real systems. First, I will consider a more realistic sensing model instead of commonly used boolean sensing model. Second, I will consider the rotational capacity of sensors and explore how it affects barrier coverage.

5.2.1 Probabilistic Barrier Coverage

Boolean sensing model might be the most widely used sensor coverage model in the literature due to its simplicity for analysis. In boolean sensing model, an intruder is 100% detected by a sensor if the intruder is within the sensors' sensing region, otherwise, cannot be detected by the sensor. This model, however, may not be able to reflect the true sensing characteristic of a sensor. Some researchers argue that the sensing quality of a sensor reduces with the increase of the distance away from the sensor (Megerian et al., 2002; Veltri et al., 2003) and therefore the intruder may not be 100% detected even it is within the sensors' sensing region. A probabilistic sensing model were proposed to describe this characteristic (Zou and Chakrabarty, 2004).

$$f(d) = \begin{cases} 1 & \text{if } d \leq r_s - r_u \\ e^{-\alpha(d-(r_s-r_u))^\beta} & \text{if } r_s - r_u < d \leq r_s \\ 0 & \text{if } d > r_s. \end{cases}$$

where d is the distance between the intruder and the sensor, $f(d)$ is the detection probability for the intruder, r_s is the sensing range, r_u is the uncertain range, and α and β are constants. We can see that the probability of detecting an intruder decreases when the distance between the intruder and the sensor increases.

The probabilistic sensing model is more realistic than the boolean sensing model, but it also introduces more uncertainty of intruder detection. Given a deployed sensor network with lots of sensor nodes, a challenging issue is how to schedule sensors so that a barrier with the required detection probability can be found? In case the network cannot provide the required detection probability of barrier coverage, what's the minimum cost of mobile sensors needed to realize the objective? Probabilistic barrier coverage has not received too much attention in the last decade and I would like to solve these two aforementioned problems.

5.2.2 Barrier Coverage with Rotational Sensors

In our dissertation, we assume that stationary sensors cannot rotate their sensing regions once they are deployed. In this case, two sensors may not overlap with each other even when are very close to each other (e.g., facing opposite directions), and mobile sensors are needed to fill in this hole. However, this would be not be an issue if sensors can rotate themselves so that they face each other and fill in the hole. Therefore, the rotation capacity of sensors could help improve the quality of barrier coverage after initial deployment even sensors cannot move and may reduce the cost of mobile sensors needed to form k-barrier coverage.

Some researchers have used rotational sensors to improve the quality of area coverage and point coverage. However, little work has been done on barrier coverage. In fact, there are a lot of challenging issues to rotational sensors to improve barrier coverage. First, when sensors rotate themselves to fill in a hole, new holes may be produced and the quality of barrier coverage may be worse than no rotation. Therefore, it is challenging to determine which sensor should be rotated. Second, sensors are power limited and we need to balance the energy consumption of rotation for each sensor so that the network life can be maximized. Therefore, it is challenging to determine how much degree for each sensor to rotate to form k-barrier coverage while maximizing the network lifetime. As part of my future work, I would like to solve these challenging problems.

Publication

1. **Zhibo Wang**, Jilong Liao, Qing Cao, Hairong Qi, and Zhi Wang. Cost-Effective Barrier Coverage Formation in Hybrid Wireless Sensor Networks. *ACM Transactions on Sensor Networks*, (under review).

2. **Zhibo Wang**, Jilong Liao, Qing Cao, Hairong Qi, and Zhi Wang. Friendbook: A Semantic-based Friend Recommendation System for Social Networks. *IEEE Transactions on Mobile Computing*, 2014, DOI: 10.1109/TMC.2014.2322373.

3. **Zhibo Wang**, Jilong Liao, Qing Cao, Hairong Qi, and Zhi Wang. Achieving k-barrier Coverage in Hybrid Directional Sensor Networks. *IEEE Transactions on Mobile Computing*, 2013, DOI: 10.1109/TMC.2013.118.

4. **Zhibo Wang**, Wei Lou, Zhi Wang, Junchao Ma, and Honglong Chen. A Hybrid Cluster-based Target Tracking Protocol for Wireless Sensor Networks. *International Journal of Distributed Sensor Networks*, vol 2013, DOI: 10.1155/2013/494863.

5. **Zhibo Wang**, Zhi Wang, Honglong Chen, Jianfeng Li, Hongbin Li and Jie Shen. HierTrack: An Efficient Cluster-based Target Tracking System for Wireless Sensor Networks. *Journal of Zhejiang University - Science C*, vol 2013, 14(6), pp. $395-406$.

6. **Zhibo Wang**, Honglong Chen, Qing Cao, Hairong Qi, and Zhi Wang. Fault Tolerant Barrier Coverage in Wireless Sensor Networks. *IEEE International Conference on Computer Communications (INFOCOM)*, 2014.

7. **Zhibo Wang**, Jilong Liao, Qing Cao, Hairong Qi and Zhi Wang. Barrier Coverage in Hybrid Directional Sensor Networks. *IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS)*, 2013.

8. **Zhibo Wang**, Clayton Edward Taylor, Qing Cao, Hairong Qi and Zhi Wang. Demo: Friendbook - Privacy Preserving Friend Matching based on Shared

Interests. *ACM Conference on Embedded Networked Sensor Systems (SenSys)*, 2011.

9. Jilong Liao, **Zhibo Wang**, Qing Cao, and Hairong Qi. Smart Diary: A Smartphone-based Framework for Inference and Prediction of Users, *IEEE Sensors Journal*, 2014, accepted.

10. Honglong Chen, Wei Lou, Zhi Wang, Junfeng Wu, **Zhibo Wang**, and Aihua Xia. Securing DV-Hop Localization Against Wormhole Attacks in Wireless Sensor Networks. *Pervasive and Mobile Computing*, 2014, DOI: 10.1016/j.pmcj.2014.01.007.

11. Honglong Chen, Wendong Chen, **Zhibo Wang**, Zhi Wang, and Yanjun Li. Mobile Beacon Based Wormhole Attackers Detection and Positioning in Wireless Sensor Networks. *International Journal of Distributed Sensor Networks*, 2014, DOI: 10.1155/2014/910242.

12. Huajie Shao, Lei Rao, Zhi Wang, **Zhibo Wang**, Xue Liu, and Kui Ren. Optimal Load Balancing and Energy Cost Management for Internet Data Centers in Deregulated Electricity Markets. *IEEE Transactions on Parallel and Distributed Systems*, 2013, DOI: 10.1109/TPDS.2013.227.

13. Honglong Chen, **Zhibo Wang**, Zhi Wang, Jiangming Xu. A Secure Localization Scheme Against Wormhole Attack for Wireless Sensor Networks. *Journal on Communications*, accepted, 2014.

14. Jie Shen, Wenbo He, Xue Liu, **Zhibo Wang**, Zhi Wang, and Jianguo Yao. Poster: Communication Reliability Analysis from Frequency Domain. *IEEE International Symposium on Quality of Service (IWQoS)*, 2013.

15. Xiang Liu, **Zhibo Wang**, Zhi Wang, Shugang Lv, and Tao Guan. A Novel Real-Time Traffic Information Collection System based on Smartphone. *Communications in Computer and Information Science*, 2013.

Bibliography

Akyildiz, I. F., Weilian, S., Sankarasubramaniam, Y., and Cayirci, E. (2002). A Survey on Sensor Networks. *IEEE Communications Magazine*, 40(8):102–114. 1, 2

Archibold, R. C. (2007). 28-Mile Virtual Fence is Rising Along the Border. *New York Times*. 8, 15

Bahl, P. and Padmanabhan, V. N. (2000). RADAR: An In-Building RF-based User Location and Tracking System. In *Proc. of IEEE INFOCOM.*, volume 2, pages 775–784. 80

Ban, D., Yang, W., Jiang, J., Wen, J., and Dou, W. (2010). Energy-Efficient Algorithms for k-Barrier Coverage in Mobile Sensor Networks. *International Journal of Computers, Communication & Control*, V(5):616–624. 11, 39

Bhandari, R. (1998). *Survivable Networks: Algorithms for Diverse Routing*. Kluwer Academic. 60, 61

Cardei, M. and Wu, J. (2004). Coverage in wireless sensor networks. *Handbook of Sensor Networks*, pages 422–433. 4, 5

Chen, A., Kumar, S., and Lai, T. H. (2007). Designing Localized Algorithms for Barrier Coverage. In *Proc. of ACM MobiCom*, pages 63–74. 10, 16

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Clifford Stein (2009). *Introduction to Algorithms*. 28, 92

Coy, P. and Gross, N. (1999). *21 Ideas for the 21st Century*. Business Week. 1

Dantu, K., Rahimi, M. H., Shah, H., Babel, S., Dhariwal, A., and Sukhatme, G. S. (2005). Robomote: Enabling Mobility in Sensor Networks. In *Proc. of IEEE IPSN*, pages 404–409. 15, 17, 80

Gage, D. W. (1992). Command Control for Many-Robot Systems. 9

Ghosh, A. and Das, S. K. (2008). Coverage and Connectivity Issues in Wireless Sensor Networks: A survey. *Pervasive and Mobile Computing*, 4(3):303–334. 5

Guvensan, M. A. and Yavuz, A. G. (2011). On Coverage Issues in Directional Sensor Networks: A Survey. *Ad Hoc Networks*, 9(7):1238–1255. 54

He, S., Gong, X., Zhang, J., Chen, J., and Sun, Y. (2013). Barrier Coverage in Wireless Sensor Networks: From Lined-based to Curve-based Deployment. In *Proc. of IEEE INFOCOM*, pages 470–474. 10

He, T., Huang, C., Blum, B. M., Stankovic, J. A., and Abdelzaher, T. (2003). Range-free Localization Schemes for Large Scale Sensor Networks. In *Proc. of ACM MobiCom*, pages 81–95. 80

Hofmann-Wellenhof, B., Lichtenegger, H., and Collins, J. (1993). Global Positioning System. *Theory and Practice.*, 1. 80

Huang, C.-F. and Tseng, Y.-C. (2005). A Survey of Solutions to The Coverage Problems in Wireless Sensor Networks. *Journal of Internet Technology*, 6(1):1–8. 5

Kahn, J. M., Katz, R. H., and Pister, K. (1999). Next Century Challenges: Mobile Networking for Smart Dust. In *Proc. of IEEE MobiCom*, pages 271–278. 3

Keung, Y., Li, B., and Zhang, Q. (2010). The Intrusion Detection in Mobile Sensor Network. In *Proc. of ACM MobiHoc*, pages 11–20. 10

Kuhn, H. W. (1955). The Hungarian Method for the Assignment Problem. *Naval Research Logistics Quarterly*, (2):83–97. 39

Kumar, S., Lai, T. H., and Arora, A. (2005). Barrier Coverage with Wireless Sensors. In *Proc. of ACM MobiCom*, pages 284–298. 4, 6, 9, 15, 16, 20, 52

Kumar, S., Lai, T. H., Posner, M. E., and Sinha, P. (2007). Optimal Sleep-Wakeup Algorithms for Barriers of Wireless Sensors. In *Proc. of the BROADNETS*, pages 327–336. 9

Kumar, S. and Shepherd, D. (2001). SensIT: Sensor Information Technology for the Warfighter. In *Proc. of IEEE FUSION*. 3

Lawler, E. L. (1976). *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York. 39

Lee, I., Leung, J. Y., and Son, S. H. (2007). *Handbook of Real-Time and Embedded Systems*. CRC Press. 3

Li, J., Chen, J., and Lai, T. H. (2012). Energy-efficient Intrusion Detection with a Barrier of Probabilistic Sensors. In *Proc. of IEEE INFOCOM*, pages 118–126. 10

Li, L., Zhang, B., Shen, X., Zheng, J., and Yao, Z. (2011). A Study on the Weak Barrier Coverage Problem in Wireless Sensor Networks. *Computer Networks*, 55(3):711–721. 10

Liu, B., Dousse, O., Wang, J., and Saipulla, A. (2008). Strong Barrier Coverage of Wireless Sensor Networks. In *Proc. of ACM MobiHoc*, pages 411–420. 10, 16

Ma, H., Yang, M., Li, D., Hong, Y., and Chen, W. (2012). Minimum Camera Barrier Coverage in Wireless Camera Sensor Networks. In *Proc. of IEEE INFOCOM*, pages 217–225. 10

Megerian, S., Koushanfar, F., Qu, G., Veltri, G., and Potkonjak, M. (2002). Exposure in Wireless Sensor Networks: Theory and Practical Solutions. *Wireless Networks*, 8(5):443–454. 106

Niculescu, D. and Nath, B. (2003). DV Based Positioning in Ad Hoc Networks. *Telecommunication Systems*, 22(1-4):267–280. 80

Pirzadeh, H. (1999). Computational Geometry with the Rotating Calipers. *Master Thesis*. 80

Pottie, G. J. and Kaiser, W. J. (2000). Wireless Integrated Network Sensors. *Communications of the ACM*, 43(5):51–58. 3

Ramirez, A. (2006). Freedom cameras. *http://friendsoftheborderpatrol.andyramirez.com/Freecameras.htm*. 15

Roush, W. (2003). 10 Emerging Technologies That Will Change the World. 1

Saipulla, A., Liu, B., Xing, G., Fu, X., and Wang, J. (2010a). Barrier Coverage with Sensors of Limited Mobility. In *Proc. of ACM MobiHoc*, pages 201–210. 11

Saipulla, A., Westphal, C., Liu, B., and Wang, J. (2009). Barrier Coverage of Line-Based Deployed Wireless Sensor Networks. In *Proc. of IEEE INFOCOM*, pages 127–135. 10, 16

Saipulla, A., Westphal, C., Liu, B., and Wang, J. (2010b). Barrier Coverage with Line-based Deployed Mobile Sensors. *Ad Hoc Networks*. 11

Shen, C., Cheng, W., Liao, X., and Peng, S. (2008). Barrier Coverage with Mobile Sensors. In *Proc. of I-SPAN*, number 2006, pages 99–104. 10

Somasundara, A. A. and Ramamoorthy, A. (2007). Mobile Element Scheduling with Dynamic Deadlines. *IEEE Transactions on Mobile Computing*, 6(4):1142–1157. 15, 80

Tao, D., Tang, S., Zhang, H., Mao, X., and Ma, H. (2012). Strong Barrier Coverage in Directional Sensor Networks. *Computer Communications*, 35(8):895–905. 10

Veltri, G., Huang, Q., Qu, G., and Potkonjak, M. (2003). Minimal and Maximal Exposure Path Algorithms for Wireless Embedded Sensor Networks. In *Proc. of ACM SenSys*, pages 40–50. 106

Wang, B. (2011). Coverage Problems in Sensor Networks: A Survey. *ACM Computing Surveys*, 43(4):32. 5

Wang, L. and Xiao, Y. (2006). A Survey of Energy-Efficient Scheduling Mechanisms in Sensor Networks. *Mobile Networks and Applications*, 11(5):723–740. 5

Wang, Y. and Cao, G. (2011a). Barrier Coverage in Camera Sensor Networks. In *Proc. of ACM MobiHoc*. 10

Wang, Y. and Cao, G. (2011b). On Full-View Coverage in Camera Sensor Networks. In *Proc. of IEEE INFOCOM*. 10

Yick, J., Mukherjee, B., and Ghosal, D. (2008). Wireless Sensor Network Survey. *Computer networks*, 52(12):2292–2330. 1, 2

Zhang, L., Tang, J., and Zhang, W. (2009). Strong Barrier Coverage with Directional Sensors. In *Proc. of IEEE GlobeCom*, pages 1–6. 10

Zou, Y. and Chakrabarty, K. (2004). Uncertainty-Aware and Coverage-Oriented Deployment for Sensor Networks. *Journal of Parallel and Distributed Computing*, 64(7):788–798. 106

Vita

Zhibo Wang received his B.E. degree in Automation from Zhejinag University, China in 2007, and was a Ph.D student in the Department of Control Science and Engineering at Zhejiang University since 2007. To pursue Ph.D., He enrolled the graduate program in the Department of Electrical Engineering and Computer Science. He is a graduate research assistant with the Advanced Imaging and Collaborative Information Processing (AICIP) Lab and the Laboratory for Autonomous, Interconnected, and Embedded Systems (Lanterns). He was the recipient of Min Kao Fellowship from 2011 to 2014 and Chancellor's Extraordinary Professional Promise Award in 2013. During his Ph.D. study, he focused on wireless sensor networks, visual sensor networks and mobile sensing systems. He completed his Doctor of Philosophy degree in Computer Engineering in Summer 2014.