2009

Mobility-assisted energy replenishment for sensor networks

Bin Tong
Iowa State University

Mobility-assisted energy replenishment for sensor networks

by

Bin Tong

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Computer Science

Program of Study Committee:
Wensheng Zhang, Major Professor
Johnny Wong
Lu Ruan
Daji Qiao
Yong Guan

Iowa State University

Ames, Iowa

2009

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my sincerest gratitude to my advisor, Dr. Wensheng Zhang, for giving me so much insightful guidance, constant encouragement, constructive criticism, and expertise on my research topics during the past years.

I am very thankful to Dr. Johnny Wong, Dr. Lu Ruan, Dr. Daji Qiao and Dr. Yong Guan for being on my POS committee and for all their help and insightful comments along the way.

I would like to thank Dr. Wallapak Tavanapong for her guidance throughout the early stages of my graduate career, which inspired me to move forward in my PhD program. I would also like to thank Dr. Ying Cai and Dr. Soma Chaudhuri for their helpful advises in my research and education.

I would like to thank my fellow graduate students: Nalin Subramanian, Jin-Sook Kim, Guanqun Yang, Chuang Wang, Zi Li, Santosh Panchapakesan, Hua Qin, and Taiming Feng. I benefited a lot from many inspirational discussions with them.

Finally, I am very indebted to my wife for her love and support during these years. It was her love and optimism that helped me get through the difficult times in my PhD career. I could not have accomplished this without her. I also want to thank my parents for always being there to offer me encouragement and support.

ABSTRACT

Over the past decade, sensor networks have consistently been a focus of the computer research community, and a large number of prototypes have been built for military and civilian applications. One of the fundamental research issues of sensor networks is the energy scarcity problem. Due to the nature of sensor networks, e.g., small-size sensor nodes and large-scale deployment, sensor nodes cannot carry a large amount of energy or be conveniently recharged. As a result, the amount of energy that can be carried by a sensor node fundamentally limits the use of sensor networks. A large number of schemes have been proposed to address this issue. These schemes, however, have one or more the following drawbacks: (i) energy cannot be replenished to the network, and thus the network lifetime is bounded by the amount of energy preloaded to sensor nodes; (ii) the ways of replenishing energy to the network may not be practical and reliable, and thus may not support normal operations of the network; and (iii) sensor nodes drained of energy are left in the deployment field, and thus may cause pollution to the environment.

Fundamentally addressing this problem requires energy to be continually replenished to sensor nodes. This can be achieved in two approaches (i) *The Node Reclamation and Replacement Approach*: Sensor nodes with low or no energy are reclaimed periodically, and are replaced with fully charged ones. (ii) *The Wireless Recharging Approach*: Sensor nodes are periodically recharged with energy transmitted from wireless chargers over radio. Both the approaches exploit mobility in accomplishing energy replenishment. Specifically, one or more mobile agents, which could be human technicians or robots, travel around the network, and perform sensor reclamation and replacement or wireless recharging task.

In this dissertation, we propose an array of new mobility-assisted energy replenishment schemes.

Firstly, for the node reclamation and replacement approach, we propose a *node replacement and*

reclamation (NRR) strategy, with which a mobile robot or human labor periodically traverses the sensor network, reclaims nodes with low or no power supply, replaces them with fully-charged ones, and brings the reclaimed nodes back to an energy station for recharging. To effectively and efficiently realize the NRR strategy for different application scenarios, we present several implementing schemes for NRR under point coverage and area coverage models, respectively. We also present schemes to improve reliability in implementation of NRR.

Secondly, the wireless recharging approach takes advantage of emerging wireless recharging technology to continually transfer energy into the network. To support long network lifetime with the wireless recharging approach, the recharging agents' activities and sensors' activities can be scheduled in a similar way as with the node reclamation and replacement approach. Therefore, in this line of research, we focus on a unique problem with the wireless recharging technology, that is, how wireless recharging affects sensor network deployment and routing arrangement. We prove the problem is NP-complete, and propose heuristic algorithms to solve it.

Extensive analysis and simulations have been conducted to verify the effectiveness and efficiency of the proposed schemes. As the battery technology lags far behind that of MEMS, we believe energy replenishment is necessary to long-lived surveillance sensor networks. To the best of our knowledge, our works of sensor node reclamation and replacement and wireless recharging are among the first efforts on studying how to re-design sensor networks to fully leverage different energy replenishment techniques.

CHAPTER 1. INTRODUCTION

1.1 Background

A wireless sensor network is composed of a large number of small-size sensor nodes which are deployed to a certain target field. Sensor nodes form a wireless network in an ad-hoc fashion. In a typical application, each sensor node monitors its surrounding area, and reports its sensory data via multi-hop communication to designated sinks for further processing. Due to its natures of infrastructure-less, low-cost and easy deployment, sensor networks are attractive for both military and civilian applications, e.g., battle field monitoring, structural health monitoring for bridges and tunnels, border surveillance, road condition monitoring, and so on. Furthermore, sensor networks are different from traditional networks in many aspects. Their attractiveness in application and unique characteristics have raised a number of research problems. Among these problems, the *energy scarcity* problem has been a paramount one since the debut of sensor networks.

1.2 Energy Scarcity Problem

The *energy scarcity* problem is defined as how to support sensor networks to work towards a long period of time given scarce energy supply in the sensor nodes. A sensor node is powered by tiny batteries which has only a small amount of energy reserve. When a sensor node uses up its energy, it dies. When one sensor node or a certain number of sensor nodes die, the network could be partitioned and stop functioning. We call the duration between the time at which the network starts functioning and the time at which the network stops functioning the *lifetime* of the network. In general, energy consumption in the network is imbalanced due to sensor nodes' different task assignment and workload. For instance, data dissemination in sensor networks exhibits a unique *funneling effect* [1] where, as data are forwarded towards the sink, the traffic load intensifies along the forwarding paths. As a result, sensor nodes closer to the sink consume their energy at higher rates. This gives rise to short network lifetime even though there is plenty of energy in other sensor nodes. The short lifetime fundamentally limits the use of sensor networks in many long-term surveillance tasks, such as structural health monitoring for bridges and tunnels, border surveillance, road condition monitoring and so on.

1.3 Existing Solutions and Their Limitations

Over years, sensor network researchers have proposed a large number of schemes addressing the energy scarcity problem. Although these schemes do mitigate energy constraints to a certain level, they have salient limitations. These schemes can be classified into the following categories:

- *Energy Conservation Schemes:* Many energy conservation schemes [2, 3, 4, 5, 6, 7, 8, 9] have been proposed to conserve energy of sensor nodes, based on the fact that sensor node can operate at different levels of power saving modes to save energy when they are not performing critical tasks. These schemes may slow down the rate of energy consumption in sensor nodes, but they cannot replenish energy to the network. Therefore, the lifetime of the network is inherently limited by the amount of energy preloaded to sensor nodes.

- *Energy Consumption Balancing Schemes:* As imbalanced energy consumption among sensor nodes may drastically shorten the lifetime of a sensor network, researchers have proposed a number of energy management schemes [10, 11, 12, 13, 14] aiming to balance energy consumption among sensor nodes. However, these schemes do not always pick the most energy-efficient sensor nodes to perform a certain task, and thus consume more than necessary energy. Furthermore, the lifetime of the network is still limited by the amount of energy preloaded to sensor nodes.

- *Environmental Energy Harvesting Schemes:* In recent years, researchers have started to study the possibility to harvest various types of environmental energy, such as solar energy, acoustic vibrations, and so on [12, 15, 16, 17, 18, 19], to recharge batteries on sensor nodes. However, these schemes are not likely to provide sufficient, steady and reliable power supply since they deeply rely on uncontrollable environmental conditions.

- *Incremental Deployment Schemes:* Incrementally deploying sensor nodes [20, 21] to take the roles of dead sensors seems to be a convenient solution. However, dead sensors are left in the field and cause pollution to the environment.

1.4 Overview of Our Research

To fundamentally address the energy scarcity problem, sensor nodes need to be continually replenished with energy. This can be achieved in two approaches:

- *The Node Reclamation and Replacement Approach*: Sensor nodes with low or no energy are reclaimed periodically, and are replaced with fully charged ones.

- *The Wireless Recharging Approach*: With the cutting-edge wireless charging technology [22], sensor nodes are periodically recharged with energy transmitted from wireless chargers over radio.

In addition, the introduction of mobility into sensor networks has been recognized as an effective solution to a number of research problems in sensor networks and attracted significant attention [13, 14, 23, 24, 25, 26]. Mobility enables physical contact to each sensor node, and thus facilitates many tasks of sensor networks. In our research, we leverage mobility in realizing the node reclamation and replacement approach and the wireless recharging approach. Specifically, we employ one or more human technicians or robots, called *mobile replacemen* in the node reclamation and replacement case or *mobile rechargers* in the wireless recharging case, to move to sensor nodes and perform node reclamation and replacement or recharging task. Fig. 1.1 shows an overview of our work.

For the node reclamation and replacement approach, we propose a *node replacement and reclamation (NRR) strategy*, with which a *mobile repairman* (MR) periodically traverses the sensor network, reclaims nodes with low or no power supply, replaces them with fully-charged ones, and brings the reclaimed nodes back to an energy station for recharging. The objective of the NRR strategy is to minimize the system cost, which is measured by the total travel distance of the MR or the total number of

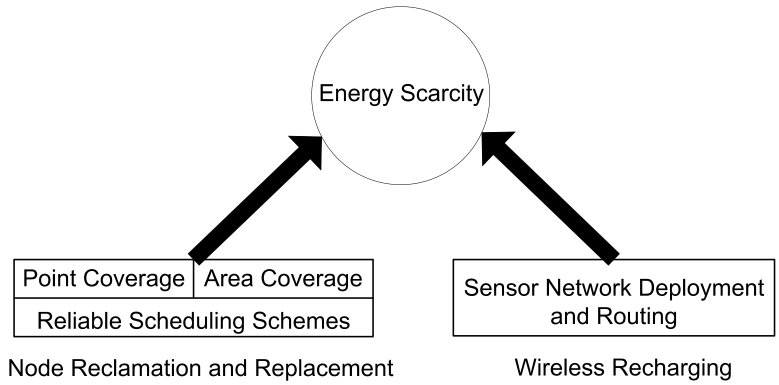


Figure 1.1 Overview of our research

replacement tours traveled by the MR. To effectively and efficiently realize the NRR strategies under different application scenarios, we propose a number of implementing schemes of the NRR strategy: (i) For point sensing coverage model, we propose an *adaptive rendezvous-based two-tier scheduling (ARTS) scheme*. (ii) For area sensing coverage model, we propose a *staircase-based node reclamation and replacement scheme*. (iii) To address reliability issues in realizing the NRR strategy, especially sensor node failures and irregular energy consumption rate, we propose *reliable node reclamation and replacement schemes*.

The wireless recharging approach takes advantage of the emerging wireless recharging technology. Recent advances in the wireless charging technology has enabled us to charge small electronic devices via radio [22] in a distance of several feet. With the wireless recharging technology, a recharger does not need to physically contact sensor nodes; instead, it only needs to move to the vicinity of them.

To support long network lifetime with the wireless recharging approach, the recharging agents' activities and sensors' activities can be scheduled in a similar way as in the node reclamation and replacement approach. Therefore, in this line of research, we focus on a unique problem with the wireless recharging technology, that is, how wireless recharging affects sensor network deployment and routing arrangement. Specifically, due to the broadcast nature of radio transmission, the charging efficiency will be improved when multiple sensor nodes in a neighborhood are recharged simultaneously. Given a number of locations whose surrounding area need to be monitored, how to deploy sensor nodes to these locations, and how to construct a routing tree rooted at the base station, such that the amount of energy transmitted by the recharger per unit of time is minimized. We prove the problem is NP-complete, and propose two heuristic algorithms to solve it.

1.5 Organization

This dissertation is organized as follows: Chapter 2 discusses related works. Chapter 3 to Chapter 5 present our proposed implementing schemes of the NRR under different application scenarios. Specifically, Chapter 3 discusses the ARTS scheme designed for the point coverage model, Chapter 4 discusses the staircase-based scheme designed for the area coverage model, and Chapter 5 discusses reliable implementation of NRR. Chapter 6 discusses the impact of wireless charging technology on sensor network deployment of routing arrangement. Chapter 7 concludes this dissertation.

In pursuit of my PhD, the following conference papers [27, 28, 29, 30, 31, 32] have been published or submitted.

CHAPTER 2. RELATED WORK

In this chapter, we discuss existing works that are relevant to ours. These works can be classified into the following categories.

2.1 Energy Conservation Schemes

Many schemes [2, 3, 4, 5, 6, 7, 8, 9, 33] have been proposed to slow down energy consumption rate at different hardware components and different communication layers in sensor networks based on the fact that sensor node can operate at different levels of power saving modes to save energy whenever possible. The schemes in [3, 4, 5] aim to reduce energy consumption for media access control. In [6], the authors proposed energy efficient routing protocols. The schemes in [7, 8] propose energy conservation schemes in applications of data placement and localization, respectively. In [9], the author designed algorithms to minimize energy cost for single-source multicast.

However, when a network is deployed for long-term tasks, such as structural health monitoring, the energy required to support network activities is much more than what can be carried by the batteries of sensor nodes. With these schemes, the rate of energy consumption is slowed down, but consumed energy cannot be compensated. Therefore, the effectiveness of these schemes is inherently restrained by the amount of energy preloaded to sensor nodes.

2.2 Energy Consumption Balancing Schemes

As imbalanced energy consumption among sensor nodes may drastically shorten the lifetime of a sensor network, researchers have proposed a number of energy management schemes [10, 11, 12, 13, 14, 34, 35] aiming to balance energy consumption among sensor nodes. In [10], the authors proposed a system in which sensor nodes collaboratively schedule their duty-cycle with the aim to support a long term surveillance mission. The works presented in [11, 12] discuss possible improvement to routing protocols in order to balance energy consumption among sensor nodes. The solution proposed in [13] employs resource-rich mobile nodes to share workload of sensor nodes located at critical spots, and the solution proposed in [14] assumes that the base station is mobile, and moves the base station around the periphery of a circular sensing field to balance forwarding workload among sensor nodes.

However, these schemes do not always pick the most energy-efficient sensor nodes to perform a certain task, and thus consume more than necessary energy. Furthermore, the lifetime of the network is still limited by the amount of energy preloaded to sensor nodes.

2.3 Environmental Energy Harvesting Schemes

Researchers have studied the possibility of harvesting various types of environmental energy such as sunlight and acoustic vibrations [12, 15, 17, 18, 19, 16, 36, 37]. In these schemes, part or all sensor nodes are equipped with environmental energy harvesting devices, e.g., solar cells. These devices harvest environmental energy in an opportunistic fashion, and store the harvested energy in the sensor nodes' batteries. When scheduling various types of tasks, the sensor nodes capable of harvesting environmental energy will be scheduled first since their energy can be replenished.

These schemes have the following practical issues: (i) They deeply rely on uncontrollable environment conditions. For instance, cloudy skies may prevent a sensor node from harvesting solar energy. (ii) In most cases, the amount of environmental energy a sensor node can harvest is proportional to the size of the energy harvesting device. For instance, the energy that a solar cell can harvest is proportional to its surface area. It is not feasible to equip a tiny sensor with large energy harvesting devices. Therefore, it is likely that the harvested energy is limited and cannot satisfy the needs of sensor nodes.

2.4 Incremental Deployment Schemes

A number of schemes [20, 21, 38] deploy sensor node incrementally with the aim to extend the network lifetime. When sensor nodes fail or are drained of energy, a controller will deploy new sensor nodes to replace them. This approach seems to be a convenient solution; however, it is not environmental friendly or practical in many scenarios. For example, in the applications of natural environmental monitoring [39], continually deploying sensor nodes without reclaiming the deserted ones may pollute the environment. Therefore, seeking an effective and efficient way to guarantee long-term energy supply has persisted as a big challenge.

2.5 Application of Mobility in Sensor Networks

In recent years, the introduction of mobility into sensor networks has been recognized as an effective solution to data collection [13, 14, 23, 24, 25, 26], sensor network deployment [40, 41, 42], and sensor network self-repair [43, 38, 20].

In [44], the authors classified mobility-based data collection schemes into three categories: *the mobile base station based schemes*, *the mobile data collector based schemes*, and *the rendezvous based schemes*. In the mobile base station based approaches [13, 14], mobile base stations (i.e., sinks) need to move to different locations periodically to balance the network traffic. The frequent movement of base stations may consume large amount of energy for both motion and maintaining the communication paths between sensors and the base stations; further, base stations may not be allowed to move in some scenarios (e.g., they are wiredly connected to the Internet). In the mobile data collector based approaches [23, 24, 25], a set of mobile data collectors traverse the network periodically to collect the data generated and buffered at sensors, while in the rendezvous based approaches [26], sensors send their data to designated rendezvous nodes, which are visited by mobile data collectors periodically for data collection.

In [40, 41, 42], the authors proposed mobility-assisted sensor network deployment schemes. A bidding protocol is proposed in [40] for deployment of a hybrid sensor network, i.e., sensor network composed of both stationary and mobile nodes. In [41], the authors proposed three deployment protocols for a fully mobile sensor network. Both works aim to maximize the sensory coverage of the network by relocation of mobile sensors. In [42], the authors proposed a scan-based deployment protocol for mobile sensor networks. Their solution partitions the network deployment field into grids, and aims to balance the number of sensors in the grids all over the network.

Recent studies [43, 38, 20] have also shown that mobility plays a critical role in sensor network self-healing and self-repair. The work presented in [43] introduces mobile sensors to replace sensors died of energy depletion. Their solution aims to find the most appropriate mobile sensor to move to the location of the dead sensor, and the best movement schedule for the mobile sensor. Schemes proposed in [38, 20] employ unmanned aerial vehicles or robots to repair networks. These works are closely related to our proposed node reclamation and replacement scheme. However, these schemes may not be realistic due to the following salient drawbacks: (i) They assume vehicles or robots carry infinite number of backup sensor nodes. (ii) These schemes incur intensive communication between sensors and base station(s), and between sensors and robots, and hence a large overhead on energy consumption.

2.6 Other Related Works

The Vehicle Routing Problem with Time Windows (VRPTW), which is closely related to the travel scheduling of mobile replacemen in our proposed node reclamation and replacement approach, has been extensively studied in the operational research [45, 46, 47, 48, 49]. VRPTW has several slightly different versions, and the version we used is as follows. A vehicle needs to deliver goods to a number of vertices. The vehicle must start and end at a depot. Each vertex has a demand for goods which must be satisfied upon the vehicle's visit, and is associated with a time window which the vehicle's visiting time to the vertex must fall into. The objective is to minimize the travel distance of the vehicle.

Our proposed wireless charging approach is enabled by the cutting-edge wireless charging [22]. With the new approach, energy can be transferred from the transmitting antenna of a power charger to the receiving antenna of sensor nodes; the charger and sensor nodes could be several feet apart, and they do not have to be aligned with each other [22]. It can be anticipated that, the wireless recharging technology will be applicable to many application scenarios. For example, in structural health monitoring applications where sensor nodes are deployed to the walls or tops of high buildings, surfaces of bridges and so on, climbing robots [50] equipped with wireless chargers can be used to recharge the nodes, avoiding frequent engagement of human workers in the risky job. In factories (e.g., petrochemical plants) where sensors are deployed to monitor hazard material or machines operating in hazard environment, unmanned vehicles equipped with wireless chargers can be used to approach and wirelessly recharge sensor nodes. Even when autonomous recharging is difficult, wireless recharging can also ease the human-controlled energy replenishment. For example, to recharge sensor nodes deployed in the wetland or natural forest (e.g., in the everglades national park of Miami), human labor standing on the trail near the wetland or forest can use long-hand devices which are equipped with wireless chargers to perform the recharging.

CHAPTER 3. NODE RECLAMATION AND REPLACEMENT UNDER POINT COVERAGE MODEL

3.1 Introduction

Mobility enables short-range interaction between the user of the network or his/her delegate and sensor nodes, and thus makes reclamation and replacement of sensor nodes possible. In this work, we propose a new strategy called *node reclamation and replacement (NRR)*. With the NRR strategy, a robot or human labor called *mobile repairman (MR)* periodically reclaims sensor nodes of low or no energy, replaces them with fully-charged ones, and brings the reclaimed sensor nodes back to a place called *energy station (ES)*; in the ES, the reclaimed sensor nodes are recharged, temporarily stored, and can be used to replace other sensor nodes in later time. This approach is applicable to sensor networks that are deployed in environments accessible to robots or human labors, such as roadsides, plants, factories, parks, forests, gardens, and so on.

In this work, we consider a point coverage model where sensor nodes are deployed around a number of points of interest, and monitor the surrounding areas of these points. Sensor nodes deployed around a point are close to each other, and thus they provide approximately the same sensing quality.

Effective and efficient realization of the NRR strategy under the point coverage model is challenging because it should schedule both the travels of the MR and the duty cycles of sensor nodes to achieve the following goals simultaneously.

(i) *Providing a guaranteed quality of service*: The duty cycles of sensor nodes should be properly scheduled to ensure a sufficient number of sensor nodes being alive before the MR's visit. Moreover, the number of sensor nodes needed to be reclaimed/replaced each time should be small such that the MR is able to complete reclamation and replacement in time considering the limited number of sensor nodes that the MR can carry at one time. This implies that sensor nodes should not die around the same time, and therefore, the widely-practiced load balancing philosophy and techniques do not apply.

(ii) *Minimizing the overhead caused by the reclamation and replacement*: The MR's travel should be properly scheduled such that the travel distance of the MR is minimized in the long run.

What is even more challenging is *the travel scheduling of the MR and the duty cycling of sensor nodes are tightly coupled*. How sensor nodes determine their duty cycles locally depends on when the MR comes to replace them; meanwhile, the travel schedule of the MR depends on the energy level of sensor nodes.

We propose an *adaptive rendezvous-based two-tier scheduling (ARTS) scheme* to tackle the above problem and thus realize the NRR strategy effectively and efficiently. In the scheme, sensor node reclamation and replacement are performed round by round. During each round, scheduling is conducted in two tiers: the global tier and the local tier. The global-tier scheduling determines how many sensor nodes should be reclaimed and replaced as well as in what order the MR should reclaim and replace these sensor nodes. Meanwhile, sensor nodes in the network collaborate to conduct local-tier scheduling to determine their duty cycles. The two tiers of scheduling interact with each other through certain visiting appointments (*rendezvous*) that can be changed from round to round adaptively. In each round, the scheduling aims at (i) maintaining required quality of service, (ii) concentrating energy consumption on sensor nodes that are to be reclaimed and replaced (hence, the amount of energy remaining in these sensor nodes is minimized when they are reclaimed), and (iii) reducing the travel cost of the MR. This way, high efficiency of reclamation and replacement is achieved in each round, eventually leading to high efficiency in the long run. Extensive simulations are conducted to evaluate the proposed scheme. The results show that the ARTS scheme meets the objectives of the NRR strategy. The results also provide insights for network designers to choose appropriate system parameters when the ARTS scheme is deployed.

3.2 NRR System Assumptions under Point Coverage Model

We consider a WSN that is deployed for long-term surveillance. The network needs to monitor a number of locations, called *posts*. Surrounding each post a number of sensors are deployed, and these sensors form a *group*. The sensors in a group are close to each other, and thus they can provide approximately the same sensing quality. We assume that the locations of posts are given. In reality, they may be determined based on the shape of the deployment field, the required sensing quality and the sensing range of each sensor.

The system architecture of the NRR strategy under point coverage model is shown in Fig. 3.1. As can be seen, the system consists of a *mobile repairman (MR)*, an *energy station (ES)*, and a WSN composed of groups of sensors surrounding the posts. The MR traverses the network periodically to reclaim sensors having low or no energy, and replace them with fully-charged sensors. The NRR

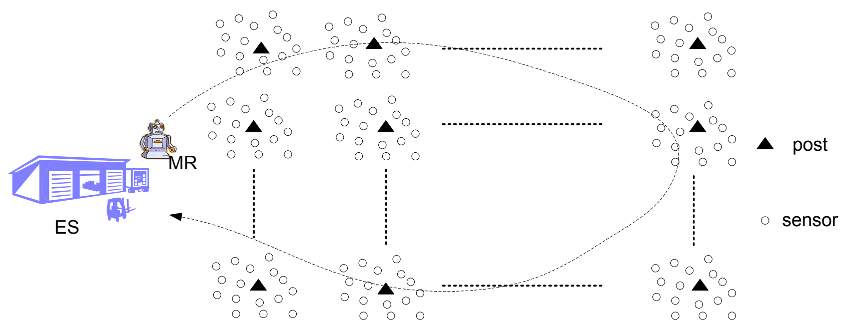


Figure 3.1 System architecture. Sensors surrounding each post are very close to each other.

strategy has the following assumptions:

(i) All sensors are loosely time synchronized. Time is divided into *phases* of a constant length. A certain number of phases compose a *round*, the length of which is denoted as l. The MR visits each post at most once every round.

(ii) A sensor has two modes: *active* and *sleep*. For every phase, if a sensor is in the active mode, its energy is reduced by δ; if it is in the sleep mode, its energy is unchanged. Let the energy of a fully-charged sensor be e, which is a multiple of δ. If a sensor is in the active mode all the time, its lifetime is denoted as τ.

(iii) At the beginning of each phase, all sensors in each group should wake up and participate in the duty-cycle scheduling. A sensor active in the previous phase collects sensory readings from all the other sensors also active in the previous phase, and based on these readings it determines the number of sensors that need to be active in the current phase. We call this number *Surveillance Number*. Surveillance number varies between N_{min} and N_{max}. In reality, the number could be determined based on if there are events taking place at the post monitored by the group and other factors. The sensor announces surveillance number, and all the other sensors listen to the announcement and decide whether they will be in the active mode or in the sleep mode in the current phase based on our proposed local-tier scheduling algorithm.

(iv) The MR has a limited capacity, denoted as C, which is defined as the maximum number of (reclaimed or fully-recharged) sensors it can carry. The MR has orientation and localization ability such that it can travel to designated locales and perform sensor replacement task.

Note that the NRR strategy does not require a WSN be connected. If a WSN is not connected, it often implies that delay in collecting sensory data can be tolerated. In this case, the MR can also serve as the mobile data collector. No matter the network is connected or not, the NRR strategy does not require each group to report its energy status through multi-hop communication. Instead, the MR collects the energy status of each group only when it visits the group. This way, communication overhead can be reduced.

3.3 Overview of the ARTS Scheme

In the proposed adaptive rendezvous-based two-tier scheduling (ARTS) scheme, node reclamation/replacement goes round by round. In each round, the mobile repairman (MR) visits each group at most once. When the MR visits a group, it reclaims/replaces a number of sensors, collects the information about residual energy in the group, and notifies the group of its next visiting time as well as the number of sensors to be reclaimed/replaced in the next visit. The calculation of the travel schedule and the number of sensors to be reclaimed/replaced is referred to as *global-tier scheduling*. Provided the MR's visiting time, sensors in the group collaborate in scheduling their activities, which is referred to as *local-tier scheduling*.

Specifically, in each round j, when the MR visits a group, it reclaims/replaces sensors, collects information about the residual energy of the group, and notifies its visiting time and the number of sensors to be reclaimed/replaced at round $j+1$. After the MR visits all groups, at the end of round j, the MR knows the residual energy of all the groups. Based on this information, the MR employs our proposed global-tier scheduling algorithm to calculate its visit time and the number of sensors to be reclaimed/replaced for each group in round $j+2$. Then in round $j+1$, the MR notifies the groups its schedule for round $j+2$ and at the same time, collects information for the global-tier scheduling for round $j+3$.

One fundamental objective of both global-tier scheduling and local-tier scheduling is that the quality of service will not be violated, i.e., there are always enough sensors alive and working. For the global-tier scheduling, this means the MR can not visit the groups too late; otherwise, there are not enough alive sensors before the MR comes. For the local-tier scheduling, given the MR's next visiting time, it must schedule the tasks among sensors effectively such that there are enough sensors alive before the MR visits.

With the above fundamental objective as a pre-requisite, there is another objective for the scheduling in both tiers: the amount of remaining energy in sensors to be reclaimed/replaced should be as small as possible. Note that, by overusing the sensors to be reclaimed/replaced, the energy of other sensors would remain high and thus reduce the workload for further reclamation/replacement. With this objective, the local-tier scheduling algorithm becomes fundamentally different from most of the existing scheduling algorithms that are targeted at load balancing; instead, it should overuse some sensors. For the global-tier scheduling, this objective means the MR should not visit the groups too early; otherwise, the sensors to be reclaimed still have a lot of energy.

In addition, especially for the global-tier scheduling, the travel distance of the MR should be minimized to save both time and energy of the MR. This objective together with the above objectives make the travel scheduling for the MR a NP-complete problem. We formally prove its NP-completeness and propose efficient heuristic solutions in Chapter 3.5 after we present the local-tier scheduling algorithm in the following.

3.4 Local-Tier Scheduling

Local-tier scheduling is performed every phase in each group. In our scheme, all sensors in a group have a consistent view regarding the amount of remaining energy in all sensors in the group. This is because (i) each sensor runs the same local-tier scheduling algorithm, thus it knows which sensors shall be in the active mode for any phase; and (ii) when the MR visits the group, it will notify the group the number of sensors to be reclaimed/replaced in the next round, thus all sensors know which sensors are to be reclaimed/replaced based on the agreement that the least energy supplied sensors will be reclaimed/replaced. It is possible node failures cause inconsistency temporarily in the view regarding the amount of remaining energy in all sensors. We provide a solution to it and discuss the solution in Chapter 3.7.

At the beginning of each phase, all sensors should wake up and participate in the scheduling. For each group, one of the sensors active in the previous phase, which is called *leading sensor*[1], collects sensory readings from other sensors active in the previous phase. Based on these readings, the leading sensor determines the surveillance number (the required number of active sensors) in the current phase, and announces the number to all sensors in the group. How the surveillance number is determined depends on application requirements and is out of the scope of this work. Then, each sensor in the group runs our proposed local-tier scheduling algorithm independently to determine which sensors shall be in the active mode. If a sensor should not be active, it will go back to sleep. Our algorithm ensures that, as long as sensors have consistent view regarding the amount of remaining energy in all sensors in the group, they can reach the same scheduling decision.

The local scheduling algorithm has two objectives: (i) Quality of service is guaranteed, i.e., there are always at least N_{max} sensors alive in the group for any phase before the MR's next visit to the group (note that without knowing future energy consumption pattern, we have to assume the worst case, i.e., using N_{max} as surveillance number). (ii) With objective (i) as a prerequisite, the local scheduling should be conducted such that the residual energy in sensors to be reclaimed/replaced is minimized.

The inputs to the local-tier scheduling algorithm include (i) the number of sensors in the group (denoted as N_d) and their residual energy; (ii) required surveillance number for the current phase; (iii)

[1]The leading sensor could be the one with the smallest ID among active sensors or determined by other methods.

the number of remaining phases before the MR's next visit. The output of the algorithm is a set of sensors that should be active for the current phase. The schedule must guarantee that the number of alive sensors is no less than N_{max} for all the subsequent phases before the MR's next visit.

We present our local-tier scheduling algorithm in two steps: We first introduce a *guard inequality* which is a condition that should be satisfied in order to attain objective (i). Then, we present our *controlled-greedy* algorithm which always attempts to schedule first the sensors with the lowest energy supply as long as doing so does not violate the guard inequality. Note that these sensors will be chosen to be reclaimed/replaced when the MR comes.

3.4.1 A Guard Inequality for Guaranteeing Quality of Service

To guarantee quality of service, we discover a guard inequality which has an attractive property that, for any phase, satisfying the inequality guarantees that we can always find a duty-cycle schedule such that at least N_{max} sensors are alive for the current phase and all the subsequent phases before the MR's next visit. Before introducing the inequality, we first introduce the following data structures: For each sensor, its ID is denoted as u_i ($0 \le i \le N_d - 1$), and the amount of its remaining energy is denoted as e_i. The sensors are sorted into a list L according to the *decreasing* order of 2-tuples $\langle e_i, u_i \rangle$.

Let t be the number of remaining phases before the next replacement/reclamation and δ be the amount of energy consumed by an active sensor per phase. We divide L into two sub-lists as follows:

- $L_1 = \langle u_0, \cdots, u_{m-1} \rangle$: Each sensor in L_1 has remaining energy of at least $t\delta$.
- $L_2 = \langle u_m, \cdots, u_{N_d-1} \rangle$: Each Sensor in L_2 has remaining energy of less than $t\delta$.

We call m the *turning* point. The following is the theorem that formally introduces the guard inequality:

Theorem 3.4.1. *If Inequality (3.1) is satisfied at the beginning of a phase, a duty-cycle schedule can be found for the current phase and all the subsequent phases before the MR's next visit, such that the quality of service is guaranteed for these phases.*

$$\sum_{i=m}^{N_d-1} e_i \ge (N_{max} - m)t\delta \tag{3.1}$$

Inequality (3.1) is called the guard inequality.

Proof. (Sketch) We first prove that if the guard inequality is satisfied at the beginning of any phase, the quality of service is satisfied for that phase. We need to show that the number of alive sensors is at least N_{max}. We consider two cases:

- $m >= N_{max}$: Clearly, there are more alive sensors than N_{max}.

- $m < N_{max}$: Suppose at the beginning of a phase, there are N_a (obviously, $N_a \leq N_d$) alive sensors. Note that dead sensors have remaining energy of 0. Hence, $\sum_{i=m}^{N_d-1} e_i = \sum_{i=m}^{N_a-1} e_i$. Therefore, Inequality (3.1) is equivalent to

$$\sum_{i=m}^{N_a-1} e_i \geq (N_{max} - m)t\delta.$$

  Note that each sensor u_i ($m \leq i \leq N_a - 1$) belongs to L_2, and its remaining energy e_i is subject to $e_i < t\delta$. Hence, it follows $N_a - m > N_{max} - m$, i.e., $N_a > N_{max}$. Thus, the number of alive sensors is greater than N_{max}.

Next, we prove that if the guard inequality is satisfied at the beginning of a phase, a duty-cycle schedule can be found such that the guard inequality is satisfied at the beginning of the next phase. Suppose the guard inequality is satisfied at the beginning of the phase, i.e.,

$$\sum_{i=m}^{N_d-1} e_i \geq (N_{max} - m)t\delta$$

We select N_{max} sensors for the phase as follows: i) $min(N_{max}, m)$ sensors with the least energy in list L_1, and ii) $max(N_{max} - m, 0)$ *alive* sensors with the least energy in list L_2. Then, at the beginning of the next phase, we construct a new List L' and new turning point m', which divides L' into two sub-lists L'_1 and L'_2. Assuming the remaining energy in any sensor u_i ($0 \leq i \leq N_d - 1$) in L' is e'_i, we consider two cases (Note that m' cannot be less than m):

- $m' = m$: We have

$$\sum_{i=m'}^{N_d-1} e'_i \geq (N_{max} - m')(t-1)\delta$$

- $m' > m$: We have

$$\sum_{i=m'}^{N_d-1} e'_i = \sum_{i=m}^{N_d-1} e'_i - \sum_{i=m}^{m'-1} e'_i$$

Clearly, for sensor u_i ($m \leq i \leq m' - 1$), $e'_i = (t-1)\delta$. It follows:

$$\begin{aligned}\sum_{i=m'}^{N_d-1} e'_i &= \sum_{i=m}^{N_d-1} e'_i - (m'-m)(t-1)\delta \\ &\geq (N_{max}-m)(t-1)\delta - (m'-m)(t-1)\delta \\ &= (N_{max}-m')(t-1)\delta\end{aligned}$$

Thus, the guard inequality is satisfied at the beginning of the next phase.

The correctness of Theorem 4.1 is straightforward due to the above two results. □

To further explain Theorem 4.1, a counter-example is shown in Fig. 3.2 to illustrate that, if the guard inequality is not satisfied in scheduling, the quality of service could be violated. Here, $N_{max} = 5$, the number of remaining phases before the MR's next visit (i.e., t) is 3, and for all the remaining phases, surveillance number is 5. At the beginning of the first phase, the guard inequality is satisfied. If a greedy scheduling policy is deployed, which always schedules sensors with the least residual energy, then at the beginning of the second phase, the left side of the guard inequality, i.e., the summation of u_3 and u_4's residual energy, is 2, and the right side of the guard inequality is 4. Thus, the guard inequality is violated at the beginning of the second phase, and at the beginning of third phase there are only 3 alive sensors, but surveillance number is 5. As a result, the scheduling fails.

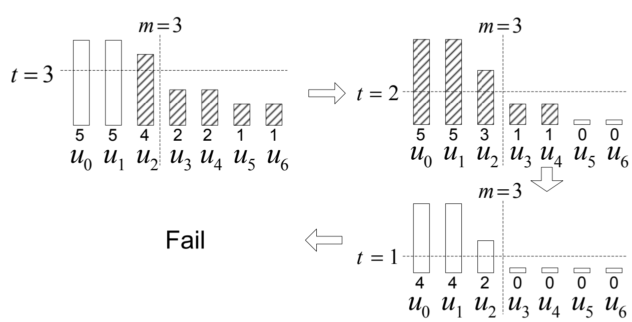


Figure 3.2 An counter example. Each bar represents a sensor, whose residual energy is marked by the value under it. Shaded bars represent sensors that are chosen to be active in the current phase. We assume $\delta = 1$.

3.4.2 Controlled-greedy Algorithm for Local-tier Scheduling

Suppose phase p is t phases before the MR's next visit, the surveillance number for phase p is x, and the guard inequality is satisfied at the beginning of phase p. Our proposed controlled-greedy algorithm will schedule the duty cycles of sensors such that the guard inequality is still satisfied at the beginning of phase $p + 1$. The details of the algorithm are as follows:

The first step is an attempted *Greedy Scheduling*. Among all the alive sensors, x sensors with the least energy are chosen. We simulate that these sensors are scheduled to be active for this phase. Hence, the energy of these sensors is deducted by δ. Then we simulate that, at the beginning of phase $p + 1$, based on the new energy level, lists L_1 and L_2 are constructed, and the guard inequality is tested. If the guard inequality is satisfied, meaning the attempt for greedy scheduling succeeds, the x chosen sensors are *really* scheduled to be active for phase p. Otherwise, the attempted greedy scheduling fails and we go forward to the second step.

The second step is *Semi-greedy Scheduling*. Note that when the semi-greedy scheduling is needed, we must have $m < N_{max}$ and $x > N_{max} - m$ at the beginning of phase p; otherwise, the greedy scheduling will succeed. The semi-greedy scheduling schedules the $x - (N_{max} - m)$ sensors with the least energy in L_1, and $N_{max} - m$ sensors with the least energy in L_2.

We use an example in Fig. 3.3 to illustrate the algorithm. As shown in Fig. 3.3, at the beginning of the first phase, we have $t = 3$, $m = 3$, $N_{max} = 5$, and for all the remaining phases, surveillance number is 5, i.e., $x = 5$. The first step fails, so we go forward to the second step. Since $x = 5 > (N_{max} - m) = 2$, we schedule $x - (N_{max} - m) = 3$ least energy supplied sensors in L_1, i.e., u_0, u_1, and u_2. The other 2 sensors are the 2 sensors with the least energy in L_2, i.e., u_5 and u_6. For the second and third phases, greedy scheduling succeeds, and the quality of service is satisfied until the MR's next visit. Algorithm 3.4.1 formally presents the above local-tier scheduling algorithm.

3.5 Global-Tier Scheduling

At the end of each round j, after the MR completes reclamation, it knows the amount of residual energy of each group, i.e., the summation of residual energy on all sensors in the group. Based on the amount, the MR first calculates the number of sensors to be reclaimed/replaced for the group in

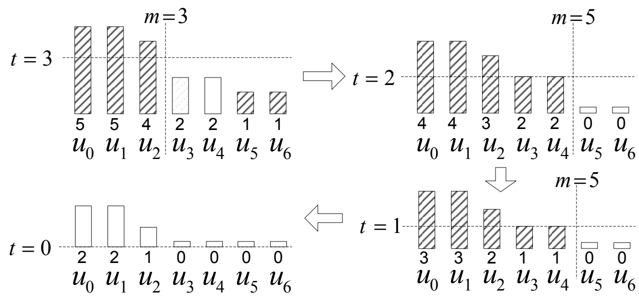


Figure 3.3 Local-tier scheduling. Each bar represents a sensor, whose residual energy is marked by the value under it. Shaded bars are chosen to be active in the previous phase. We assume $\delta = 1$.

Algorithm 3.4.1 Local-Tier Scheduling in a group at phase p

Notations:
x: surveillance number in phase p
t: the number of remaining phases before the MR's next visit
L, L_1, L_2, L': lists of sensors in an energy decreasing sequence
e_i, e'_i: the amount of the remaining energy of the i^{th} sensor in list L, L' (respectively)
δ, N_d: defined previously

1: Sort $\langle e_i, u_i \rangle$ $(0 \le i \le N_d - 1)$ into a list L in decreasing order.
2: Calculate m and partition L into two sublists: L_1, L_2. $|L_1| = m, \forall e_i \in L_1, e_i \ge t\delta; \forall e_i \in L_2, e_i < t\delta$.
3: Tentatively schedule the last x sensors in L to be active, and decrement their energy by δ.
4: Construct a new list L' and calculate the new turning point m'.
5: **if** $\sum_{i=m'}^{N_d-1} e'_i \ge (N_{max} - m')(t-1)\delta$ **then**
6:     Confirm the schedule and return.
7: **else**
8:     Cancel the changes made in line 3.
9:     Schedule the last $(x - (N_{max} - m))$ sensors in L_1.
10:     Schedule the last $(N_{max} - m)$ sensors in L_2.

---

round $j + 2$, which is called the *replacement number*. Based on the replacement numbers and the residual energy information of all groups, the MR runs our proposed MR travel scheduling algorithm to calculate the optimal schedule to visit every group in round $j + 2$. When the MR visits a group in the coming round (round $j + 1$), it notifies the group of its visiting time and replacement number in round $j + 2$. With the prerequisite of satisfying the quality of service, the global-tier scheduling has two objectives: (i) the total travel distance of the MR is minimized, and (ii) the remaining energy of sensors to be replaced is minimized. To satisfy the above objectives makes the global-tier scheduling a NP-hard problem. In the following, we present how to calculate the replacement number, prove the

NP-hard nature of calculating the MR's travel schedule, and present heuristic solutions to this problem.

3.5.1 Calculation of Replacement Numbers

The replacement number at group g_i $(1 \leq i \leq n)$ in round j $(j \geq 1)$ is denoted as $N_r(i, j)$. When the MR comes to group g_i, it chooses $N_r(i, j)$ sensors with the minimum residual energy to reclaim, and replace them with the same number of fully-charged sensors. Let $E(i, j)$ denote the summation of residual energy on all sensors in group g_i after the MR completes replacement/reclamation at group g_i in round j and e be the amount of energy held by a fully-charged sensor. Recall that, l is the length of a round and τ is the lifetime of sensor if being active all the time. $N_r(i, j)$ is conservatively calculated as follows:

$$\begin{cases} N_r(i,1) = N_r(i,2) = max\{\lceil (l/\tau)N_{max} \rceil, N_{max}\} \\ N_r(i,j+2) = max\{\lceil 3(l/\tau)N_{max} \rceil - N_r(i,j+1) - \lfloor \frac{E(i,j)}{e} \rfloor, 0\} \quad (j \geq 1) \end{cases}$$

For the first two rounds, the replacement number is predetermined as $max\{\lceil (l/\tau)N_{max} \rceil, N_{max}\}$ to guard against the worst case scenario when N_{max} nodes are needed to be active all the time. Here, l/τ is the number of sensors needed such that at any time at least one sensor is alive throughout a round. The replacement number of round 3 and later is calculated by the second formula. The part $\lceil 3(l/\tau)N_{max} \rceil - N_r(i, j + 1)$ reflects that in the worst case, group g_i may be required to have the maximum number of active sensors for three rounds in a row, rounds j, $j+1$ and $j+2$, but with only one replenishment in round $j + 1$. This happens when the MR only visits group g_i at the very beginning of round j and the end of round $j+2$. $\lfloor \frac{E(i,j)}{e} \rfloor$ is the number of fully-charged sensors that have equivalent energy to the total residual energy in group g_i. Note that, if the MR does not visit group g_i in round j, it cannot obtain the exact number of $E(i, j)$ and will conservatively estimate it assume that N_{max} sensors are required to be active throughout round j. If $\lceil 3(l/\tau)N_{max} \rceil - N_r(i, j + 1) - \lfloor \frac{E(i,j)}{e} \rfloor < 0$, then $N_r(i, j + 2) = 0$, which implies that group g_i does not need replacement/reclamation in round $j + 2$ and the MR does not visit group g_i in the round.

3.5.2 Calculation of the Travel Schedule for the MR

For each round, we need to (i) minimize the total travel distance of the MR in this round and (ii) minimize the residual energy in sensors to be reclaimed/replaced. Since minimizing the residual energy of sensors to be reclaimed/replaced affects the travel distance of the MR, and vice versa, these two objectives are not likely to be satisfied at the same time. Hence, we associate a weight with each objective and aim to optimize the combined objective. In the following, we introduce related data structures, formally state this optimization problem, prove its NP-completeness, and present heuristic solutions.

3.5.2.1 Data Structures

At the end of round j, the MR calculates its visiting schedule to all the groups in round $j + 2$. A 2-D table R, is used in calculating the best visiting time of the MR to each group. R records the total residual energy in the sensors to be reclaimed/replaced in each group at each phase of round $j+2$ if the MR visits the group at the phase. The energy can be positive, zero, or an invalid number. The invalid number means the quality of service is already violated in the phase because the MR comes too late. Fig. 3.4 shows an example. Each entry $e(i, p)$ in the table represents the total residual energy in the sensors to be reclaimed/replaced in group g_i when the MR visits it at phase p in round $j + 2$. Along a row, the value of $e(i, p)$ gets lower as p increases, i.e.,

$$e(i, 1) > e(i, 2) > \cdots > e(i, c_i) = \cdots = e(i, d_i) = 0$$

The best phases for the MR to visit group i in round $j + 2$ is within $[c_i, d_i]$, for which the residual energy is 0. Phase d_i is called the *deadline* for the MR to visit group g_i in round $j + 2$, because if the MR comes later, the quality of service is violated. It is also possible that even if the MR visits a group at the last phase of round $j + 2$, the residual energy is still greater than 0. Group g_x is such an example. In the case, we call the last phase, i.e., phase m, the deadline for the MR to visit group g_x in round $j + 2$, i.e., $d_x = m$.

Because the future energy consumption pattern is not known beforehand, the table can only be constructed based on prediction. In our scheme, we simulate the local-tier scheduling algorithm phase

group \ phase	1	2	···	m
g_1	$e(1,1)$	$e(1,2)$	··· 0 ··· 0 / ···	/
g_2	$e(2,1)$	$e(2,2)$	··· 0 ··· 0 / ···	/
⋮	⋮	⋮	⋮	⋮
g_x	$e(x,1)$	$e(x,2)$	···	$e(x,m)$
⋮	⋮	⋮	⋮	⋮
g_n	$e(n,1)$	$e(n,2)$	··· 0 ··· 0 / ···	/

Figure 3.4 Table R: Residual energy in sensors to be reclaimed/replaced in all groups at all phases in round $j+2$. "/" represents invalid number.

by phase until phase p in round $j+2$, using the maximum surveillance numbers (N_{max}) conservatively to obtain entry $e(i,p)$. Other methods to predict surveillance numbers will be explored in our future work.

In addition to the 2-D table R, the following data structures are needed in formalizing the problem.

- $G(V, E, W(V), W(E), R)$ denotes a complete undirected graph. $V = \{g_i \mid 0 \leq i \leq n\}$, where g_0 represents the ES and $g_1, g_2, \cdots, g_n$ represent n groups. $W(V) = \{N_r(1, j{+}2), \cdots, N_r(n, j{+}2)\}$, where $N_r(i, j+2)$ is the replacement number for group g_i ($1 \leq i \leq n$) in round $j+2$. For any two groups g_i and g_k ($0 \leq i \neq k \leq n$), there is an edge (g_i, g_k), whose weight represents the cost for the MR's travel between groups g_i and g_k, and $W(E)$ stores the cost of each such edge. R is defined above.

- $\vec{t} = (t_1, t_2, \cdots, t_n)$ is a visiting time vector in which t_i is the phase when the MR visits group g_i in round $j+2$.

- $D = d(\vec{t})$ is the total traveling distance for the MR to fulfill a complete reclamation/replacement tour according to time vector $\vec{t}$. Note that due to the limited capacity of the MR, the MR needs to go back to the ES for reloading multiple times.

3.5.2.2 NP-Completeness of MR travel scheduling Problem

The problem is to find a time vector $\vec{t}$ for the MR which can carry up to C fully-charged sensors to visit the groups in the network and replace $W(V)$ sensors, such that the following metric is minimized.

$$Y = \alpha D + \beta \sum_{i=1}^{n} e(i, t_i) = \alpha d(\vec{t}) + \beta \sum_{i=1}^{n} e(i, t_i), \tag{3.2}$$

where α and β are two system parameters, each representing the weight of the associated item.

The MR travel scheduling problem is a NP-hard problem since the well-known NP-hard Vehicle Routing Problem with Time Windows (VRPTW) [45, 46, 51] can be reduced to this problem. The sketch of a formal proof is presented as follows.

Theorem 3.5.1. *The MR travel scheduling problem is a NP-complete problem.*

Proof. (Sketch) To facilitate the proof, we formulate the decision version of the MR travel scheduling problem as, *given a cost Z, is there a feasible time sequence $\vec{t}$ for the MR travel scheduling with object value Y no greater than Z?*

First of all, the MR travel scheduling problem is a NP problem. Given a time sequence $\vec{t}$, we can check in polynomial-time whether the overall cost is no greater than the given cost Z or not by computing Y in Eq. (3.2).

Next, we show that VRPTW $\leq_p$ MR travel Scheduling. Given an instance of VRPTW

$$G'(V', E', W(V'), W(E'), TW(V')),$$

where G' is a complete undirected graph with vertex set $V' = \{v'_i \mid 0 \leq i \leq n\}$ (v'_0 represents the depot) and edge set E'. $W(V')$ is the set of demands on all vertices in V', and $W(E')$ is the set of distances of all edges in E'. $TW(V') = \{[c'_i, d'_i] \mid 1 \leq i \leq n\}$ is the set of time windows of all vertices in V'. A vertex v'_i ($1 \leq i \leq n$) has to be visited during its time window $[c'_i, d'_i]$.

We construct an instance of the MR travel scheduling problem $G = (V, E, W(V), W(E), R)$ based on the same topology as VRPTW, that is, $V = V', E = E', W(V) = W(V'), W(E) = W(E')$. In terms of R, for each vertex g_i in V, we create a row vector e_i, where $c_i = c'_i, d_i = d'_i, \forall j = 1, \cdots, c_i - 1$, $e(i, j) =$ a positive number, and $\forall j = c_i, \cdots, d_i, e(i, j) = 0$. Note that $[c'_i, d'_i]$ is the time window of vertex v'_i in VRPTW and similarly, $[c_i, d_i]$ is the best visiting time window to group g_i in the MR travel scheduling problem.

We make two claims: (i) If there is a solution to the VRPTW problem such that the vehicle visits all vertices during their time window, and the total travel distance of the vehicle is no greater than D, there must be a solution to the MR travel scheduling problem such that the object value Y in Eq. (3.2) is no greater than αD. (ii) If there is a solution to the MR travel scheduling problem such that the object value Y in Eq. (3.2) is no greater than αD, there must be a solution to the VRPTW problem such that the vehicle visits all vertices during their time window, and the total travel distance of the vehicle is no greater than D.

Claim (i) can be proved as follows: If there exists a time sequence $t'_1, t'_2, \cdots t'_n$ as a feasible solution to the VRPTW problem, where the total travel distance is no greater than D, such that each vertex $v'_i \in V'$ is visited within its time window. By applying the same time sequence to the MR travel scheduling problem, we have the following inequality because of the mapping relationship:

$$Y \leq \alpha D + \beta \sum_{i=1}^{n} e(i, t_i) = \alpha D$$

Claim (ii) can be proved as follows: If there exists a time sequence $t_1, t_2, \ldots t_n$ as a feasible solution to the MR travel scheduling problem, where the total cost is no greater than αD. Clearly, each group g_i $(1 \leq i \leq n)$ is visited at a phase when its residual energy in the sensors to be reclaimed/replaced is 0, i.e., within $[c_i, d_i]$. If we apply the same time sequence to the VRPTW problem, each vertex can be visited during its time window because of the mapping relationship. Thus, the time sequence is the solution to the VRPTW problem with a travel distance no greater than D.

Therefore, the MR travel scheduling problem is NP-complete. $\square$

3.5.2.3 Heuristic Solutions

The object value Y depends on two optimization objectives, the total MR travel distance and the total residual energy in sensors to be reclaimed/replaced, which cannot be optimized at the same time. Moreover, due to the NP-completeness of the problem, we propose heuristic solutions to optimize the two objectives according to their associated weights, α and β.

For one extreme case, α is much larger than β and hence distance D is optimized with priority. Because of the limited capacity of the MR, multiple tours to visit all the groups are calculated to minimize D, where each tour is a sequence of groups that the MR should visit in order. Since changing the ordering of different tours does not affect D, based on table R, we find the order of these tours to minimize the residual energy in sensors to be reclaimed/replaced.

For the other extreme case, β is much larger than α and hence the residual energy in sensors to be reclaimed/replaced is minimized first. Intuitively, groups with similar deadlines should be visited at similar times. This can done by distributing groups into tours based on similarity of their deadlines. The MR will conduct these tours in the increasing order of these deadlines; that is, tours with early deadlines will be done first. Within each tour, the visiting order of the groups is determined to minimize the MR travel distance.

For more general cases, we propose a *super-tour heuristic solution* and use a parameter M to reflect the relative weight of α and β. The basic idea of the super-tour heuristic solution is as follows: given n groups, we distribute them into a number of *super tours* based on similarity of their deadlines. A super tour is composed of M *physical* tours. That is, in a super tour, the MR needs to go back to the ES for $M - 1$ times for reloading. We order super tours according to the deadlines aiming to minimize the total amount of residual energy in sensors to be reclaimed/replaced. Inside each super tour, we first calculate multiple physical tours to minimize the total travel distance; then we determine the sequence of these physical tours according to the deadlines to minimize the residual energy.

When M gets larger, the total travel distance tends to be smaller, but the total amount of residual energy in sensors to be reclaimed/replaced tends to be larger, and vice versa. Note that when $M = 1$, the solution is reduced to the extreme case that the residual energy is minimized first. M is tuned according to the weights α and β such that Y is minimized.

Algorithm 3.5.1 formally presents the super-tour heuristic solution. One step in the algorithm is to divide a super tour into M physical tours such that the travel distance is minimized, i.e., the Capacitated Vehicle Routing Problem (CVRP) problem, which is NP-hard. We employ a well-known and effective heuristic algorithm proposed in [52] to solve it.

Algorithm 3.5.1 Super-Tour Heuristic Solution (for round $j+2$)

1: Sort all groups g_i into a sequence L in an increasing order based on pair $\langle d_i, e(i, d_i)\rangle$ $(1 \le i \le n)$.
2: **while** L is *NOT* empty **do**
3: Construct a super tour ST by reversely traversing L such that the sum of their demands is less than or equal to $M * C$ while adding one more group will make the sum greater than $M * C$.
4: Divide ST into M physical tours such that the overall travel distance is minimized.
5: Decide the visiting times to all groups in ST.
6: Remove all groups in ST from L.

3.6 Performance Evaluation

3.6.1 Experimental Settings, Metrics and Methodology

We built a custom simulator to evaluate the performance of the proposed scheme. We consider a sensor network with a number of groups randomly deployed in a square field. In all experiments, we normalize the full energy level of a sensor to 400 units and the energy consumption rate is 0.1 unit/minute. Thus, each sensor's lifetime τ is 4000 minutes. The length of a phase is set to 10 minutes. The length of a round l is also set to 4000 minutes unless otherwise mentioned. we set $N_{min} = 3$ and $N_{max} = 8$ for all groups. The number of sensors deployed in each group is 16. Note that N_{max} is 8 and a sensor's lifetime is equal to a round length. 16 sensors means initially deployed sensors are able to guarantee the quality of service for two rounds in the worst case scenario[2]. Unless otherwise stated, 36 groups of sensors are deployed in a $1000m * 1000m$ square field at random. The MR has a capacity of 40 sensors, and its speed is 20 meters/minute.

In reality, surveillance number is determined by the application, as well as the real-time frequency and distribution of events. In our simulation, we consider two types of distributions of the number: the linear decrease distribution and the Gaussian distribution.

- *Linear decrease*: The Probability Density Function *(pdf)* of a surveillance number is inversely proportional to the surveillance number. Specifically, the *pdf* for surveillance number $i \in [N_{min}, N_{max}]$ is:

$$f(i) = \frac{N_{max} - i + N_{min}}{\sum_{k=N_{min}}^{N_{max}} k}$$

[2]In the simulation, we let the all groups operate for one round before the first round in which the MR starts to perform reclamation/replacement.

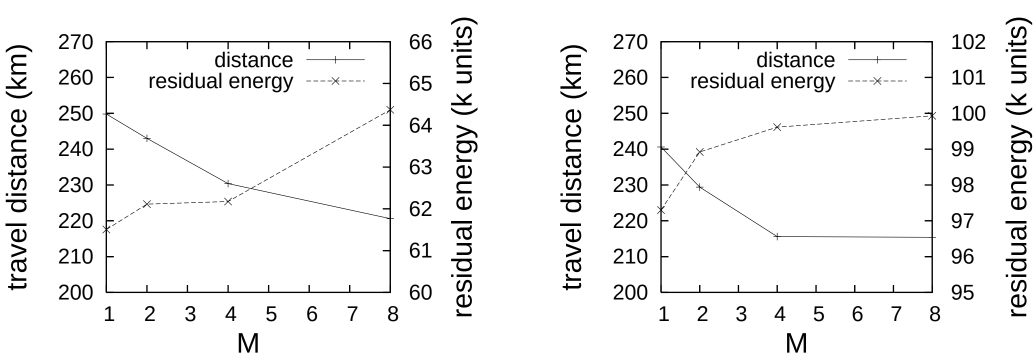
- *Gaussian*: We deploy Gaussian distribution $N(\mu = N_{min}, \sigma^2 = 4)$ and truncate it to the range $[N_{min}, N_{max}]$.

We conducted the following sets of experiments: (i) Tradeoff between the residual energy in sensors to be reclaimed/replaced and the MR's travel distance; (ii) Comparison between the super-tour heuristic solution and the optimal solution; (iii) Impact of MR's capacity on the performance; (iv) Impact of round length on the performance.

For each experiment, our algorithm is executed for a long time period, starting at 0 and and ending at a *cutoff* time. The cutoff time is set to 40,000 minutes for all experiments. Furthermore, we run each simulation for 20 times and take the average for each evaluated metric.

3.6.2 Tradeoff between Residual Energy in Sensors to Be Reclaimed/Replaced and MR's Travel Distance

In this section, we study the tradeoff between the amount of residual energy in sensors to be reclaimed/replaced and the MR's travel distance, by measuring and showing the energy amount and the travel distance as functions of the number of physical tours in a tours (i.e., system parameter M). Fig. 3.5(a) and 3.5(b) show how the amount of residual energy in sensors to be reclaimed/replaced and the MR's travel distance change as M varies.



(a) Linear decrease (b) Gaussian

Figure 3.5 Impact of M

As can be seen, when M increases, the total residual energy in sensors to be reclaimed/replaced increases, while the total travel distance decreases. The value of M leverages the two optimization factors in Eq. (3.2). When $M = 1$, it corresponds to the energy-first heuristic, since there is only one physical tour in a super tour; while $M = 8$, it corresponds to the distance-first heuristic since the MR can visit all the sensors in one super tour.

3.6.3 Comparison between Our Solution and Optimal Solution

Due to the NP-Completeness nature of the problem, we cannot work out the optimal solution for a network with a practical size. However, for a network small enough, we can work out the optimal solution by enumerating all the possible travel schedules and get the best one. In the experiment, we randomly deploy 9 groups into a $500m * 500m$ square field. The MR has a capacity of 10 sensors, and its speed is set to 10 meters per minute. In addition, we set $\alpha = 5$ and $\beta = 1$.

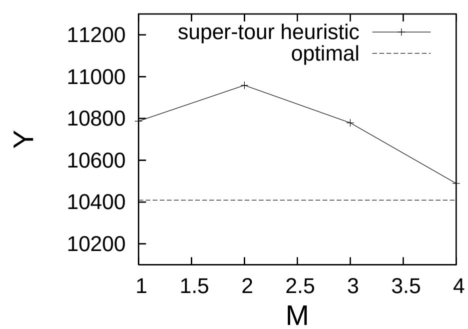
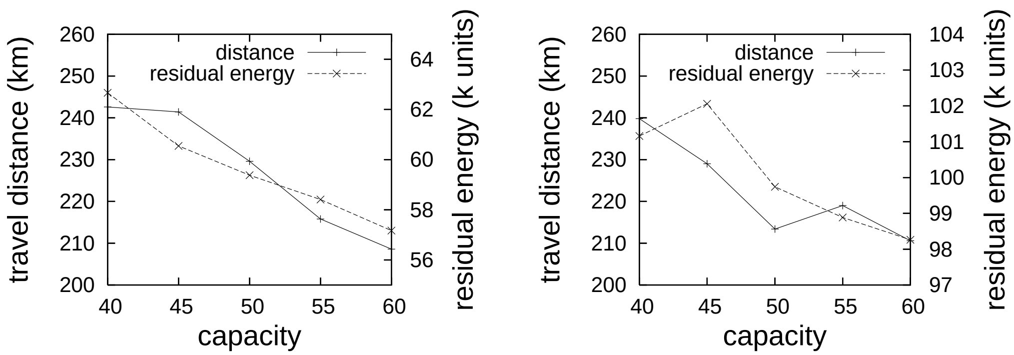


Figure 3.6 Comparison between super-tour heuristic and optimal solution

Fig. 3.6 shows the performance difference between the super-tour heuristic solution and the optimal solution. The y-axis of Fig. 3.6 is object value Y in Eq. (3.2). From the figure we can see, when M = 4, the object value of the super-tour heuristic solution is only 0.8% higher than the optimal value. Even though the result is from a small size problem, we still can see the effectiveness of our proposed heuristic. Note that even though the objective value is worse when M is smaller than 4, through simulation, the M which makes the best result can be found out and used.

3.6.4 Impact of MR's Capacity

As can be seen from Fig. 3.7, both performance metrics decrease as the MR carries more sensors. With higher capacity, the MR can pay less number of trips to finish the replenishment process, and thus travel shorter in total. On the other hand, since the MR is able to finish the replenishment process in a shorter time, groups tend to be visited later by the MR and thus have more chance to use the energy in the sensors to be reclaimed/replaced. In general, larger capacity is beneficial.

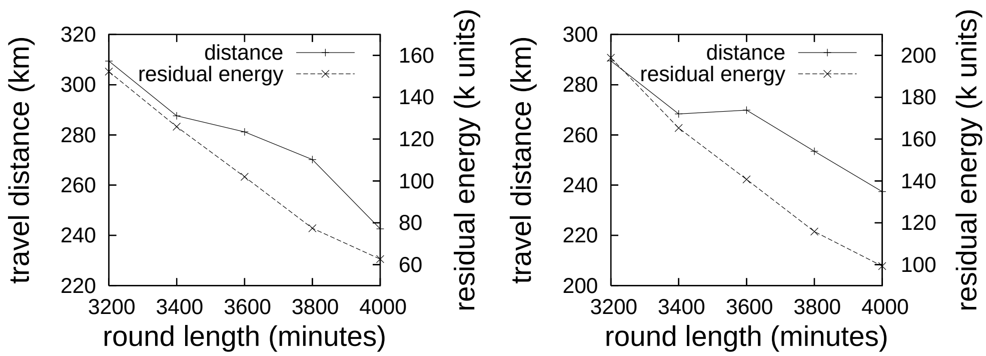


(a) Linear decrease   (b) Gaussian

Figure 3.7 Impact of capacity

3.6.5 Impact of Round Length

We set $M = 2$. Round length l is varied among $\{3200, 3400, 3600, 3800, 4000\}$ minutes, and the initial deployment number per group is calculated as $\lceil 2(l/\tau)N_{max} \rceil$. In Fig. 3.8, both residual energy and travel distance decrease as round length increases. The decrease of the former metric is because longer round length gives groups more chances to use up the energy in the sensors before they are reclaimed/replaced. The decrease in total travel distance can be explained as follows: Let us first consider the average length of a tour. When round length increases, each group has more sensors to be replaced in one round, which implies a less number of visiting destinations in each tour and hence shorter distance for each tour. Next consider the number of tours during the entire simulation time (40,000 minutes). As round length increases, the number of tours needed in a round increases; meanwhile the total number of rounds decreases. The total number of tours is the result of joint effects of these two conflicting factors. By checking simulation traces, we find that the number of tours generally decreases as round length increases.



(a) Linear decrease

(b) Gaussian

Figure 3.8 Impact of round length

3.7 Discussions

In this section, we discuss some practical issues in implementing the ARTS scheme. First, the ARTS scheme treats failed sensors the same as sensors drained of energy, i.e., failed sensors will be replaced by the MR. We employ the following method to detect sensor failures. At the time for scheduling (i.e., at the beginning of a phase), a sensor u that is chosen to be active in the phase, should broadcast a message to all sensors in the group. Other sensors also know which sensors shall be active for the phase. If they do not receive the message from sensor u, they assume sensor u has failed. Hence, they re-run the local-tier scheduling algorithm to select another active sensor to replace sensor u.

In the local-tier schedule of the ARTS scheme, sensors in a group may have different views regarding the amount of remaining energy in all sensors due to sensor failure of other reasons. To address this issue, we let each sensor broadcast its amount of remaining energy at the beginning of a phase every certain number of phases. When other sensors receive this information, they update their record accordingly.

Sensors of different types may be deployed to monitor a post. These sensors may have different amount of initial energy and different energy consumption rate. Furthermore, sensors of the same type may provide different sensing qualities dependent on their location or other factors. We plan to study the methods for adapting the ARTS schemes to these scenarios in our future work.

CHAPTER 4. NODE RECLAMATION AND REPLACEMENT UNDER AREA COVERAGE MODEL

4.1 Introduction

The ARTS scheme only considers point coverage while in many application scenarios such as border surveillance, guaranteeing area coverage [53, 54, 55] is desired.

In this work, we propose another implementing scheme of the NRR strategy under the area coverage model. The objective of the scheme is to minimize the system cost, which is mainly reflected by the frequency that the MR should be dispatched to perform reclamation and replacement. The number of backup sensor nodes is usually limited and recharging nodes that have been replaced takes nontrivial time. Given these constraints, how to minimize the maintenance frequency of the MR poses as a difficult problem. Conventionally, duty cycles of sensor nodes are scheduled in a balanced manner such that all nodes die at the similar time. If this philosophy is still applied, the MR is required to use a limited number of backup nodes to replace nearly all nodes within a short time period, which is an impossible mission. Hence, new protocols for scheduling duty cycles are demanded.

It is ideal that the duty cycles of sensor nodes are scheduled appropriately such that, every certain time interval, only a subset of sensor nodes with the same number as backup nodes are to deplete energy and need replacement, and the time interval should be longer than the time needed to recharge all sensor nodes that have been replaced. This way, as every time sensor nodes needing to be replaced are no more than backup nodes, they can all be replaced and hence the lifetime of the network can be maintained. As the time interval for replacing two sets of nodes is longer than the time needed to recharge sensor nodes that have been replaced, it is guaranteed that there is always enough number of fully-charged backup nodes when replacement is needed.

We propose the *staircase-based scheme* to realize the above idea, under the assumption that any sensor node, if active, consumes energy at the same rate, and there is no sensor node failure. The scheme schedules the duty cycles of sensor nodes in a well-planned manner. Ideally, at any moment, all sensor nodes in the network form a staircase according to the amount of their residual energy. Sensor nodes with the lowest level of residual energy is in the lowest layer of the staircase, those with the second to the least level of residual energy is in the second to the lowest layer, and so on and so forth. The difference between any two adjacent levels of residual energy is constant, and the time for a node to consume it is larger than the time to recharge nodes replaced. This staircase-based scheduling of sensor nodes according to their residual energy guarantees that during a fixed time interval, a fixed number, same to the number of the backup nodes, of sensor nodes deplete their energy.

The staircase-based scheme consists of three tightly coupled components: (i) the protocol for sensors to coordinate their duty-cycle scheduling locally, (ii) the protocol for sensors and the ES to communicate with each other, and (iii) the algorithm for the ES to determine how to perform node reclamation and replacement on demand. These three components work together to achieve the following objectives: (a) required area coverage is guaranteed without disruption in the field monitored by the sensor network; (b) the total number of replacement tours traveled by the MR is minimized.

4.2 NRR System Assumptions under Area Coverage Model

We consider a network of n sensors, denoted as $s_1, s_2, s_3, \cdots, s_n$, is deployed to a continuous field for long-term monitoring. The monitored field is divided into m small areas, denoted as $a_1, a_2, a_3, \cdots, a_m$, such that, within any area a_i, the required sensing coverage level is the same at any point of the area.

As shown in Fig. 4.1, the whole NRR system is composed of an *energy station (ES)*, a *mobile repairman (MR)*, and a sensor network. The ES stores a certain number (denoted as x) of backup sensors, and can recharge energy to sensors. The MR can be a human technician or a mobile robot. The MR can traverse the sensor network, reclaiming sensors of no or low energy, replacing them with fully-charged ones, and bringing the reclaimed ones back to the ES for recharging. Other assumptions of the system are as follows:

- All sensors are time synchronized. Time is divided into phases. A phase is a basic scheduling

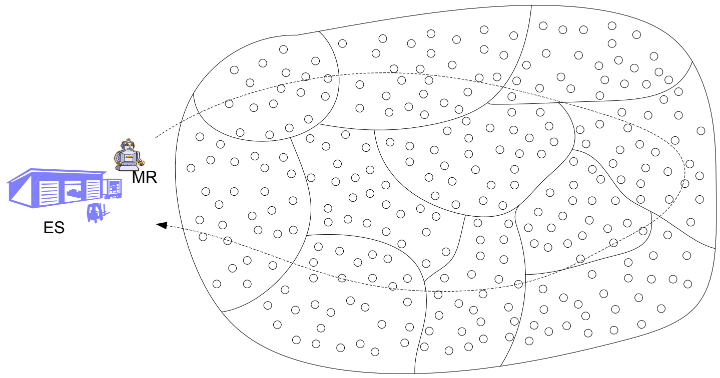


Figure 4.1 System architecture of NRR for area coverage

unit for duty-cycle scheduling; i.e., a sensor will not change its mode (active or sleeping) during a phase.

- The network is connected, and there is a communication path from every sensor to the base station.

- A sensor has two modes: *active* and *sleeping*. For every phase, if a sensor is in the active mode, its energy is reduced by a fixed amount; if it is in the sleeping mode, its energy is unchanged. Let the energy of a fully-charged sensor be e. If a sensor is in the active mode all the time, its lifetime is denoted as T.

- For each area, the required sensing coverage level varies from N_{min} to N_{max}, subject to certain (e.g., Gaussian) distribution.

- Each area is deployed with $N_{max} + N_{back}$ (N_{back} is an integer greater than or equal to 1) *disjoint* sets of sensors, where each set of sensors can completely cover the area. That is, every point in the area can be covered by at least one sensor in each of the sets. We call these sets *coverage sets*. The reason for having more than N_{max} sets of sensors is to avoid service disruption at the time of node reclamation and replacement (Note: node reclamation and replacement cannot be completed in non-negligible time; hence, reclamation and replacement will inevitably disrupt the working of nodes that are reclaimed or newly placed).

- The MR has orientation and localization ability such that it can travel to designated locales and perform sensor replacement task. In this work, we assume that the MR is able to carry x sensors a time. This can be relaxed to the case that the capacity of MR is smaller, and the trip scheduling algorithm studied in [29] may be applied to address this problem.

- Charging a sensor at the ES takes *non-negligible* time, which is denoted as τ. Note that, sensors can be recharged in parallel, we assume that it is possible to recharge all x backup sensors managed by the ES at the same time.

Design Goal. In this work, we aim to design a collaborative scheduling scheme for sensors and the reclamation and replacement scheduling algorithm for the ES/MR, such that (i) the sensor network can maintain the required area coverage for an infinite period of time, and (ii) the number of travels the MR should take is as small as possible (i.e., the average interval between two consecutive replacement trips is as large as possible).

4.3 Overview of the Staircase-Based Scheme

4.3.1 Key Ideas

To achieve guaranteed area coverage for an infinite period of time, two necessary tasks should be performed: firstly, sensors should collaboratively schedule their duty-cycles to achieve required area coverage; secondly, sensors and the ES/MR should coordinate to replenish energy into the network through node reclamation and replacement.

If the ES have unlimited number of backup sensors to use and the reclamation/replacement can be finished instantly, the above two tasks can be achieved easily. For example, any existing collaboratively duty-cycle scheduling schemes [10] can be applied for the first task; as for the second task, whenever an area is short of alive sensors, a request is sent to the ES, which then dispatches the MR to reclaim and replace sensors for the area. In reality, however, the backup sensors owned by the ES are limited and should be not too large for economic reasons, and the reclamation/replacement and recharing take non-negligible time. Using the above naive approach, it may happen that, at some time instance, 1000 sensors should be replaced while the ES has only 500 backup sensors.

To address the above problem, the duty-cycle scheduling of sensors and the node reclamation/replacement activities should be carefully planned. In our design, we propose a *staircase scheduling model* for this purpose. The key ideas are as follows:

Coverage Set-level Scheduling. In each area, sensors are grouped into disjoint coverage sets, where nodes in each single coverage set can together cover any points in the area. Sensors are scheduled in the unit of coverage sets.

Intra-group Staircase. In each area, coverage sets are scheduled in a thoughtful way that, the required area coverage is guaranteed and meanwhile, the remaining energy levels of different sets are kept different, which form a *staircase* among the sets. Hence, different sets can be reclaimed and replaced at different time instances. As to be elaborated later, this facilitates the ES/MR to temporally reuse limited number of backup nodes to maintain lifetime.

Inter-group Staircase. Intra-group staircase may not be sufficient. It is likely that each of multiple areas needs to replace one of their coverage sets at the same time instance, and the demanded number of backup sensors could exceed what can be offered by the ES. To avoid this inter-group congestion of demands, our delicately designed scheduling strategy ensures that different areas issue demands at different time instances. This way, inter-group staircase is formed to further scatter demands and thus provide more flexibility to the ES/MR to plan the reclamation/replacement activities.

Redundancy for Flexibility. If the replacement requests issued by every area should be satisfied immediately by the ES/MR, the flexibility for performing reclamation/replacement activities will be strictly limited. At least, the number of trips taken by the MR may be too large, which may incur high system maintenance overhead. To address this issue, redundant nodes are deployed to areas to form backup coverage sets. With these backup sets, replacement requests can be satisfied with some delay, which allows the ES/MR to use one trip to satisfy multiple requests to reduce the maintenance cost.

4.3.2 Framework

Based on the above key ideas, the framework of our scheme is summarized as follows:

Duty-Cycle Scheduling. In our scheme, sensors in each area a_i are grouped into $N_{max} + N_{back}$ disjoint coverage sets, denoted as $cs_1, cs_2, \cdots, cs_{(N_{max}+N_{back})}$, where nodes in each coverage set can together sense every point in the area. The sensors in the same coverage set are scheduled together as an integral entity. Hence, sensors in the same coverage set have similar remaining energy levels at any time; to simplify scheduling, we assume all sensors in the same coverage set have the same remaining energy level. All coverage sets fall into two categories: N_{max} *primary* sets and N_{back} *backup* sets. At any phase, only primary sets can be scheduled, and a coverage set can change its role from primary to backup and vice versa. Each sensor knows which coverage set it belongs to, and also maintains the information of the estimated remaining energy levels of sensors in other coverage sets. Therefore, every sensor in each area has a consistent view regarding the remaining energy levels of sensors in the same area.

In each area, a *head* is elected among all sensors through a certain collaborative selection algorithm [56], and the role is rotated among the nodes to balance energy consumption. At the beginning of each phase, the head broadcasts the coverage requirement for the current phase, i.e., the number of coverage sets (called *coverage number*) that shall be active. How to determine the coverage number is application-dependent and out of the scope of this work. A possible approach is, the coverage number is determined based on the observations by active sensors in the last phase; if some event was detected in the last phase, the coverage number may be increased and vice versa. At the beginning of a phase, all sensors will wake up and listen to the broadcast of the coverage number. Upon receipt of the coverage number, each sensor runs our proposed duty-cycle scheduling algorithm independently to determine whether it should be active or not. Since all sensors in an area have the consistent view about the remaining energy level of all nodes in the same area, they will arrive at the same scheduling decision.

Interactions between Area Heads and the ES. Our duty-cycle scheduling algorithm ensures that, different primary sets will use up their energy at different time instances. Shortly before a primary set (say, cs_i) of sensors uses up its energy, it hands over its duty to a backup set (say, cs_j), which has full energy. After the handoff, cs_i becomes a backup set waiting to be reclaimed and replaced, while cs_j becomes a primary set. Meanwhile, the head of the area sends a *ready* message to the ES with the number of sensors in cs_i, which is the number of sensors that need to be reclaimed and replaced. Specifically, the ready message has the following format:

$$ready\langle a, cs_i, c\rangle,$$

where a is the ID of the area, cs_i is the ID of the coverage set needing to be reclaimed and replaced, and c is the total number of sensors in the cs_i.

If a primary set is about to use up its energy, and there is no backup set with fully-charged nodes to which the primary set can hand over its duty to, the head of the area sends out a *deadline* message to the ES. Specifically, the deadline message has the following format:

$$deadline\langle a \rangle,$$

where a is the ID of the area.

Node Reclamation and Replacement. Alg. 4.3.1 formally describes how the ES responds to the above ready and deadline messages. Specifically, when the ES receives a ready message, it accumulates the total number of sensors that are ready to be replaced. The ES will dispatch the MR when either of the following conditions is true: (i) It receives a deadline message; or (ii) the total number of sensors that are ready to be replaced exceeds x.

Algorithm 4.3.1 Reclamation and Replacement Scheduling: for the ES

Notations:
- x: number of backup sensors
- R: set of ready messages that have not been served
- t: total number of sensors that are ready to be replaced

Initialization:

$R \leftarrow \phi$
$t \leftarrow 0$

Upon receipt of a ready message: $ready\langle a, cs, c \rangle$

$R \leftarrow R \cup ready$
$t \leftarrow t + c$
if $t >= x$ **then**
    Dispatch the MR to serve the earliest x replacement requests.
    $t \leftarrow t - x$
    $R \leftarrow R - \{\text{served requests}\}$

Upon receipt of a deadline message: $deadline\langle a \rangle$

Dispatch the MR to serve all pending replacement reqeusts
$R \leftarrow \phi$
$t \leftarrow 0$

4.4 Detailed Description of the Staircase-Based Scheme

The duty-cycle scheduling scheme is performed at each sensor in each area at the beginning of each phase. The input to the duty-cycle scheduling scheme is (i) the estimated remaining energy level of every sensor in all coverage sets and (ii) the coverage number for the current phase. The output of the scheme is the coverage sets that should be active in the current phase. To ease understanding, we first describe how the scheduling scheme works when the coverage number of every area is fixed (i.e., N_{max}), which is followed by the general case where the coverage number of every area is variable ranging from N_{min} to N_{max}.

4.4.1 A Special Case: Fixed Coverage Requirement

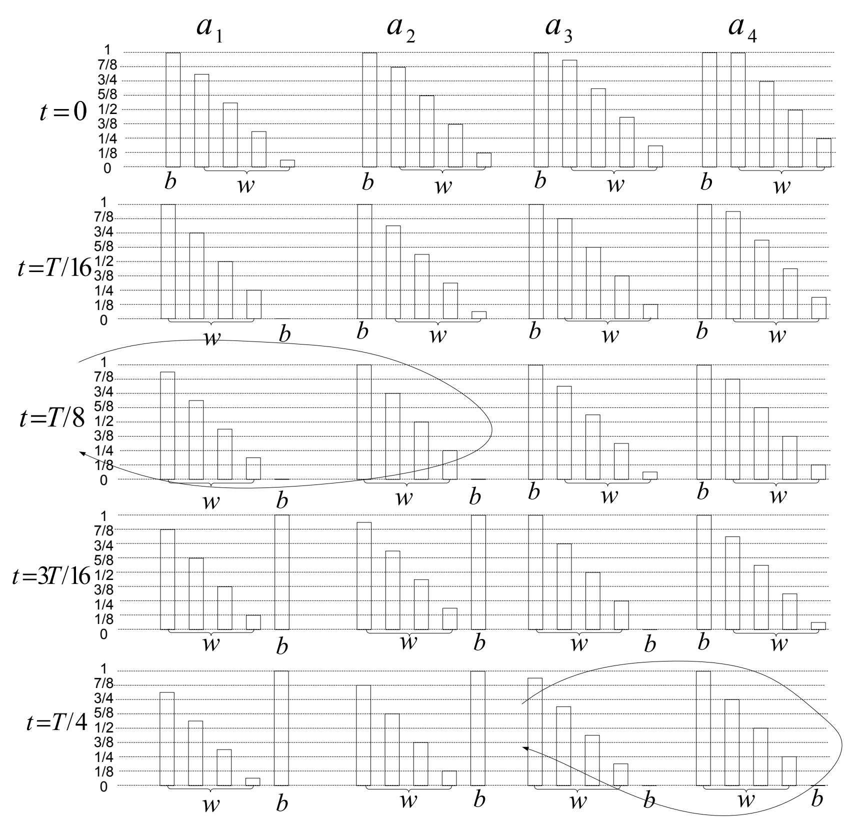


Figure 4.2 Example 1: duty-cycle scheduling. Each bar represents a coverage set. $N_{max} = 4$, $N_{back} = 1$, $m = 4$, and $x = 32$. Each coverage set in every area has 16 sensors. "w" means primary set, and "b" means backup set.

Suppose for each area a_i, the number of sensors in each coverage set of area $a_i, 1 \leq i \leq m$, is denoted as c_i. Since areas are divided based on coverage requirement, c_i could be different for different areas. For each area $a_i, 1 \leq i \leq m$, we need to schedule all N_{max} primary coverage sets at any phase.

For all the N_{max} primary coverage sets, we let their remaining energy per node form a "staircase", and the height of each stair is

$$\frac{e}{N_{max}},$$

where e is the amount of full energy of a sensor. The formation procedure of this staircase is discussed later.

Fig. 4.2 shows an example where the monitored field consists of four areas. Each row in Fig. 4.2 shows the snapshot of remaining energy of each coverage set in each area at different time points. As can be seen, out of five coverage sets in each area, one is in the backup role, and the other four are in the primary role. The remaining energy per node of the four primary coverage sets forms a staircase with a stair height of $e/4$.

In our scheme, we define an order in which areas are visited by the MR to reclaim and replace sensors in these areas. For any two areas that are to be visited consecutively, their staircases have a phase difference δ, where δ and the height of a stair have the following relation:

$$\frac{e}{N_{max}} = m\delta, \tag{4.1}$$

where m is the number of areas. In Fig. 4.2, areas are sorted as a_1, a_2, a_3, a_4. As can be seen, at time point 0, the staircase of primary coverage sets in a_2 is $e/16$ higher than that of the primary coverage sets in a_1, the staircase of the primary coverage sets in a_3 is also $e/16$ higher than that of the primary coverage sets in a_2, and so on. This phase difference remains as time evolves.

Since the coverage requirement is always N_{max}, all the four primary coverage sets will be active at any time. When the primary coverage set with the minimum energy drains of its energy, it will (i) shift its duty to a backup coverage set, which has full energy; (ii) becomes a backup set. Meanwhile, the head of the area will send a ready message to the ES, and the full energy backup coverage set will become a primary coverage set.

In Fig. 4.2, at time $t = T/16$, the primary coverage set with the minimum energy in a_1 drains of its energy, and shifts its duty to the only backup coverage set. A ready message is also sent to the ES.

Since at this time, the total number of nodes that are ready to be replaced is 16, which is less than $x = 32$, the MR will wait. At time $t = T/8$, the primary coverage set with the minimum energy in a_2 drains of its energy, and shifts its duty to the backup coverage set. A ready message is also sent to the ES. At this time, the total number of nodes that are ready to be replaced equals to x. Thus, the MR makes a replacement tour, replacing nodes in the backup sets of a_1 and a_2. Similarly, the MR makes another replacement tour at $t = T/4$, replacing nodes in the backup coverage sets of a_3 and a_4.

One noteworthy fact is that, in this example, recharging x sensors should be completed in $T/8$. We have derived a relation between recharging time and the minimum number of backup sensors needed, which is to be discussed later.

Staircase Formation In the above, we assume that the staircase structure is already formed. However, when a sensor network starts operating, all sensors in the sensing field have full energy. To form the staircase structure, we propose the following method. Without loss of generality, we assume the pre-defined visiting order to the areas is $\langle a_1, a_2, \cdots, a_m \rangle$. When a primary coverage set in a_1 consumes δ energy [1], it shifts its duty to a backup coverage set, and becomes a backup coverage set itself. The head of area a_1 also sends a ready message to the ES. Similarly, when a primary coverage set in a_2 consumes 2δ energy, it shifts its duty to a backup coverage set, and becomes a backup coverage set itself. Besides, the head of area a_2 sends a ready message to the ES. In general, a primary coverage set in a_i will make the role transition and trigger ready message reporting after it consumes $i\delta$ energy.

The next time for role transition and ready message reporting in a_1 is after a primary coverage set with the minimum energy has consumed $m\delta$ energy after the first role transition. The third time for role transition and ready message reporting in a_1 is after a primary coverage set with the minimum energy has consumed $m\delta$ energy after the second role transition; and so on. Other areas will follow the same rule to conduct their role transitions and ready message reporting. After time T, the staircase structure will be naturally formed. Fig. 4.3 shows an example of staircase formation of area a_1 in Fig. 4.2. Note that, the staircase shown at $(t = T)$ is the same as that at $(t = 0)$ in Fig. 4.2.

[1] Since all primary coverage sets will have the same remaining energy at that time, we randomly pick one.

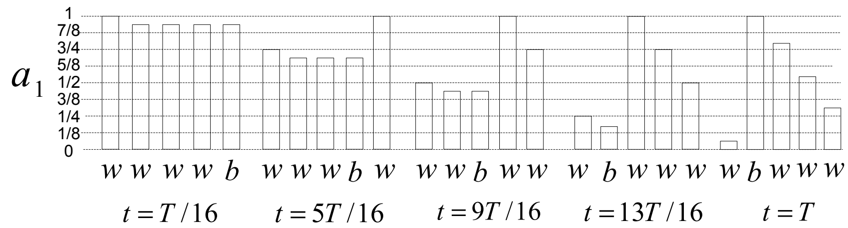


Figure 4.3 Example 2: initial staircase formation of area a_1 in Fig. 4.2. $\delta = T/16$.

4.4.2 General Case: Variable Coverage Requirement

In this section, we consider the general case that the required coverage number is not always N_{max}, but varies in range $[N_{min}, N_{max}]$.

Given an area $a_i, 1 \leq i \leq m$, we let the remaining energy per node of its N_{max} primary coverage sets, denoted as $w_1, w_2, \cdots, w_{N_{max}}$, form a staircase as described above. Assume $e_i, 1 \leq i \leq N_{max}$ represents the remaining energy of coverage set w_i. Without loss of generality, we have $e_1 < e_2 < e_3 < \cdots < e_{N_{max}}$, where the difference between any two consecutive terms is $m\delta$. The duty-cycle scheduling is performed phase by phase.

Assuming the coverage number for the first phase is $q_0, N_{min} \leq q_0 \leq N_{max}$, we will need to schedule q_0 primary coverage sets. In our scheme, we schedule primary coverage sets $\{w_1, w_2, w_3, \cdots, w_{q_0}\}$ for the first phase. If the coverage number for the next phase is $q_1, N_{min} \leq q_1 \leq N_{max}$, we will need to schedule q_1 primary coverage sets. In this case, we will schedule coverage sets

$$w_{(q_0+1) \mod N_{max}}, w_{(q_0+2) \mod N_{max}}, \cdots, w_{(q_0+q_1) \mod N_{max}}$$

In other words, we adopt a round-robin scheduling policy while maintaining the staircase structure.

In this case, whenever each area a_i uses up its primary coverage set with the minimum energy, its head sends a ready message to the ES if there are backup coverage sets with full energy. If all the backup coverage sets have empty energy before the primary coverage set with the minimum energy is about to use up its energy, the head will send a deadline message to the ES.

The formal duty-cycle scheduling algorithm for the variable coverage number case is described in Alg. 4.4.1.

Fig. 4.4 shows an example. In this example, the length of a phase is $T/16$, and the coverage number

Algorithm 4.4.1 Duty-Cycle Scheduling for the variable coverage requirement case: for sensors in primary coverage set $w_i, 1 \leq i \leq N_{max}$

Notations:
q: coverage number in phase p
$b_j, 1 \leq j \leq N_{back}$: N_{back} backup coverage sets
var $start$; // start position of primary coverage sets for phase p.

```
if w_i ∈ {w_start, w_((start+1) mod N_max), ···, w_((start+q−1) mod N_max)} then
    Schedule coverage set w_i.
    if w_i drains of its energy then
        Randomly choose a backup coverage set with full energy, b_j.
        Coverage set w_i changes its role to backup.
        Coverage set b_j changes its role to primary.
start ← (start + q) mod N_max
```

is two for the first four phases for all areas. As can be seen, in the first phase, the two primary coverage sets with the minimum remaining energy in all areas are scheduled. In the second phase, the next two primary coverage sets in all areas are scheduled. This process is carried on.

At time $t = T/16$, the head of area a_1 sends out a ready message to the ES. The MR will not make a replacement tour since the number of sensors that are ready to be replaced, 16, is less than $x = 32$. At time $t = 3T/16$, the head of area a_2 sends another ready message. At this time, the number of sensors that are ready to be replaced reaches x, and thus the MR conducts a replacement.

Staircase Formation. The staircase formation procedure for the variable coverage number case is the same as that for the fixed coverage requirement case.

4.5 Discussions

4.5.1 Lower Bound of Required Number of Backup Nodes

Since charging batteries takes non-negligible time, the energy replenishment rate is affected by the number of backup nodes owned by the ES. Assuming the number of backup nodes is x, the time to recharge a sensor is τ, and full energy of a sensor is e, the energy replenishment rate is

$$xe/\tau.$$

This rate should be large enough to compensate energy consumption of the network even in the worst case scenario. Specifically, the worst case energy consumption rate occurs when the coverage number

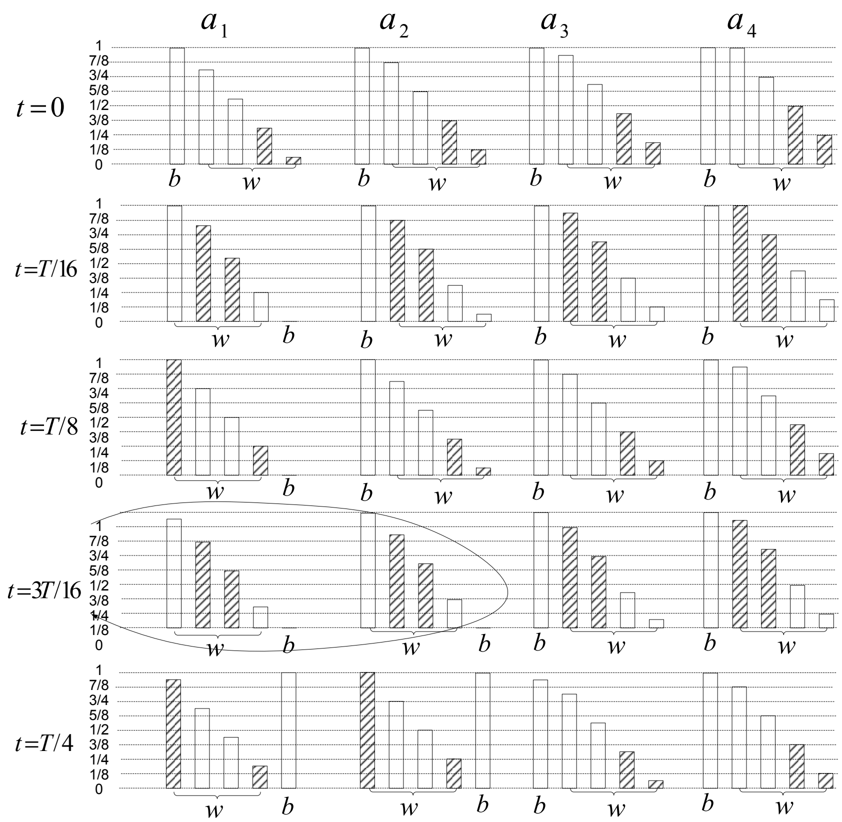


Figure 4.4 Example 3: duty-cycle scheduling. Each bar represents a coverage set. Shaded bars are scheduled in the current phase. $N_{max} = 4$, $N_{back} = 1$, $m = 4$, and $x = 32$. Each coverage set in every area has 16 sensors. Phase length is $T/16$.

in each area is N_{max}.

Consider area $a_i, 1 \le i \le m$, in which each coverage set has c_i sensors. N_{max} coverage sets will each consume e/N_{max} energy in T/N_{max} time, where T is a sensor's lifetime. Thus the total energy consumption of area a_i in T/N_{max} time is

$$c_i N_{max} \frac{e}{N_{max}} = c_i e$$

It follows that the energy consumption rate in area a_i is $c_i e N_{max}/T$.

The total energy consumption rate over all areas is

$$\frac{e}{T} \sum_{i=1}^{m} c_i N_{max}$$

We have

$$
\begin{aligned}
\frac{xe}{\tau} &\geq \frac{e}{T} N_{max} \sum_{i=1}^{m} c_i \\
x &\geq \frac{\tau}{T} N_{max} \sum_{i=1}^{m} c_i
\end{aligned} \tag{4.2}
$$

4.5.2 Upper Bound of Number of Backup Nodes

In the proposed scheme, the MR only replaces sensors in backup coverage sets for each area. The reason is that replacement will disrupt sensor nodes' operation. By not replacing the N_{max} primary coverage sets, service disruption is avoided.

As a result, at one time, the maximum number of sensors that are ready to be replaced in area a_i is $N_{back} c_i$, and the total number of sensors that are ready to be replaced over all areas is

$$
N_{back} \sum_{i=1}^{m} c_i \tag{4.3}
$$

In general, this is the upper bound for x in the sense that if $x > N_{back} \sum_{i=1}^{m} c_i$, the surplus backup sensors will never be used.

However, there is an exception when the lower bound calculated by Eq. (4.2) is greater than the upper bound calculated by Eq. (4.3). This case is discussed in the following.

4.5.3 Impact of Node Recharging Time

If sensor recharging time at the ES is very long, it is possible that the lower bound of x calculated by Eq. (4.2) is greater than the upper bound calculated by Eq. (4.3). Here we face a dilemma: On one hand, x should be greater than the calculated lower bound in order to guarantee the coverage requirement over an infinite period of time; on the other hand, if x is greater than the calculated upper bound, the surplus sensors will not be used. We propose the following method to address this issue.

Assume the lower bound of x calculated by Eq. (4.2) is denoted as l, and the upper bound calculated by Eq. (4.3) is denoted as h. Given sensor recharging time τ, we list its divisors by natural numbers $2, 3 \cdots$, and for each divisor, we calculate a lower bound l' using Eq. (4.2). This process stops $l' < h$. Assume at this time the divisor of τ is $\tau/k, k \geq 2$.

If we have $kh \geq x \geq kl'$, then the x backup sensors can be divided into k batches. All sensors in a batch will start being recharged at the ES at the same time. Further, we order the k batches into a sequence, and the start times for any two consecutive batches in the sequence being recharged differ by τ/k. In other words, the system generates x/k, $h \geq x/k \geq l'$, fully charged sensors every τ/k. This way, the proposed scheme works as the regular case.

4.5.4 Some Practical Issues

Next, we discuss some practical issues in implementing the proposed scheme.

First, sensor nodes may fail at any time. Our scheme can tolerate sensor failures, i.e., failed sensors will be replaced by the MR. We employ the following method to detect sensor failures. At the time for scheduling (i.e., at the beginning of a phase), if a primary coverage set w is chosen to be active in the phase, all sensors in the coverage set will send a message to the head of the area. If the head does not receive the message from a sensor u for more than a threshold of times, it considers u has failed, and then sends a *failure* message to the ES. The MR will replace the failed sensor in its next replacement trip.

Second, our scheme requires communication between active sensors and the head of each area in every phase. Since the size of an area is typically small, the imbalance in energy consumption among sensor nodes for forwarding data packets is limited. Further, we factor the maximum energy consumption for packet forwarding into total energy consumption at each sensor.

Third, the head of each area will report ready and deadline messages, which may travel a long route. However, reporting of these messages is infrequent since they are only sent out when the area have consumed considerable amount of energy, which is on the magnitude of sensor batteries's lifetime.

4.6 Performance Evaluation

We built a custom simulator using C++ to evaluate the performance of the staircase-based scheme.

Table 4.1 General experimental settings of the staircase-based scheme

field size	$500m * 500m$
# of areas	80
sensing range	$20m$
transmission range	$40m$
N_{min}	1
N_{max}	4
N_{back}	$\{1, 2, 3\}$
recharging time	6 hours
sensor's lifetime time	240 hours (5 days)
# of sensors per coverage set	$Gau(16, 3)$
sensor's full energy	1440 units
phase length	10 minutes
energy consumption rate	0.1 unit/minute
cut-off time	4800 hours (200 days)

4.6.1 Experimental Settings, Metrics and Methodology

Table 4.1 shows system parameters we used in the simulation. We consider a sensor network composed of 80 areas. Each area has $(N_{max} + N_{back})$ disjoint coverage sets, and each of which is able to cover the whole area. The number of sensors in each coverage set is a random number, which complies to a Gaussian distribution, $Gau(16, 3)$, with mean of 16.

In the experiments, we normalize the full energy level of a sensor to 1440 units and the energy consumption rate is 0.1 unit/minute if the sensor is active. Thus, each sensor's lifetime T is 240 hours, i.e., 5 days. The length of a phase is set to 10 minutes. Coverage numbers for each area vary between N_{min} and N_{max}. N_{min} is set to 1, and N_{max} is set to 4 in all experiments.

In reality, coverage number is determined by the application, as well as the real-time frequency and distribution of events. In our simulation, coverage number complies to a truncated Gaussian distribution, which is $Gau(\mu = N_{min}, \sigma = 2)$ truncated to the range $[N_{min}, N_{max}]$.

The performance metrics include:

- *Average replacement interval*: Average time between two consecutive replacement tours made by the MR.

- *Average utilization of the MR*: The MR may not carry x sensors in each replacement tour due to the replacement deadlines set by each area. Average utilization of the MR is the average ratio of the number of backup sensors actually carried by the MR to x.

- *Distribution of replacement intervals*: To ease reclamation/replacement planning, a distribution of replacement intervals with smaller variance is preferred in practice.

We consider the following sets of scenarios: (i) All areas have the same coverage number at any time, and (ii) All areas subject to the same distribution of coverage numbers, but coverage numbers in all areas are independent of each other. For each experiment, our proposed scheme is executed for a long time period, starting at 0 and ending at a *cut-off* time. The cutoff time is set to 4800 hours, i.e, 200 days, for all experiments. Furthermore, we run each simulation for 50 times for the metrics of average replacement interval and average utilization of the MR, and 500 times for the metric of distribution of replacement intervals, and take average for each of the metrics.

4.6.2 Scenario I: Same Coverage Number for All Areas

In this experiment, coverage number is the same for all areas at any phase. The number of backup coverage sets, N_{back}, varies among $\{1, 2, 3\}$. The results are shown in Fig. 4.5. Fig. 4.5(a) and Fig. 4.5(b) show the trend of average replacement interval and utilization of the MR when coverage number complies to the truncated Gaussian distribution. As can be seen, given the number of backup coverage sets, average replacement interval increases as the number of backup sensors, x, increases in an approximately linear fashion. At the same time, the utilization of the MR keeps at 1. However, when x reaches a certain value, the average replacement interval levels off, and at the same time, the utilization of the MR starts to drop.

For example, given one backup coverage set for each area, when x exceeds 1300, the average replacement interval stops increasing, and the utility of the MR drops to 0.95.

The reason for this phenomenon is explained as follows. Since all areas have the same coverage number, their primary coverage sets consume their energy at the same rate. Further, in our scheme, the remaining energy of primary coverage sets in any two consecutive areas according to the pre-defined visiting order has a phase difference δ. Therefore, the time instances for the heads in all areas to send ready messages are evenly distributed as time evolves.

when x is small, the time instances when the number of sensors that are ready to be replaced exceeds x are always ahead of arrival of any deadline message. Thus, the MR will replace x sensors in each replacement tour, which results in a full MR utilization. Further, given the total amount of energy consumption of the network until the cutoff time, the total amount of energy that is needed to be replenished into the network is fixed. As a result, average replenish interval increases with x in a linear fashion.

On the other hand, when x exceeds the upper bound of x calculated by Eq. (4.3), which is between 1200 and 1300 in this experiment, a deadline message will arrive before the number of sensors that are ready to be replaced reaches x. Therefore, replacement interval stops to increase at this point. Furthermore, since the number of backup sensors that are actually used stays at the upper bound value, as x increases, the utilization of the MR decreases in a reciprocal fashion.

The results show that given a fixed number, N_{back}, of backup coverage sets, we cannot raise average replacement interval over a certain value by simple increasing x. Instead, N_{back} will need to be increased.

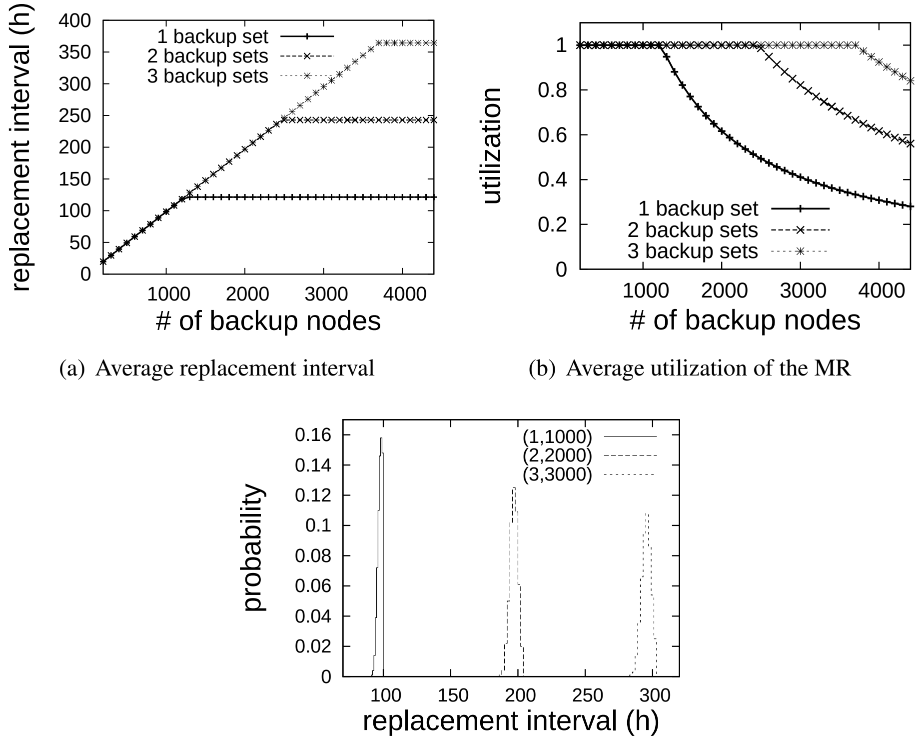
Fig. 4.5(c) shows histograms of replacement intervals for three different parameter sets. In Fig. 4.5(c), the first number in a pair of parentheses is the number of backup coverage sets, and the second number is x. For example, "(1,1000)" means one backup coverage set and 1000 backup sensors. Note that for all the three parameter sets, the utilization of the MR is 1. As can be seen in Fig. 4.5(c), replacement intervals cluster in a small range. For parameter set (1,1000), the mean of replacement intervals is 98.53, and the standard deviation is 2.35.

4.6.3 Scenario II: Same Coverage Number Distribution for All Areas

In this experiment, all areas have the same value of $N_{max} = 4$, and their coverage numbers comply to the same probability distribution. However, coverage numbers of different areas are independent of each other. In addition, we assume in each area, coverage numbers at different phases are independent of each other.

Fig. 4.6(a) and Fig. 4.6(b) show very similar patterns as in Fig. 4.5(a) and Fig. 4.5(b), respectively. This can be explained as follows.

Since coverage number is a random variable between N_{min} and N_{max}, and coverage numbers at different phases are independent of each other, the summation of coverage numbers over a large num-



(a) Average replacement interval

(b) Average utilization of the MR

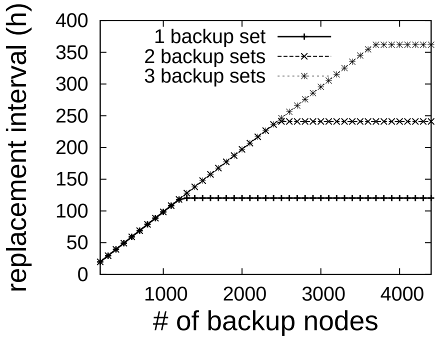
(c) Histogram of replacement intervals

Figure 4.5 Scenario I: same coverage number for all areas

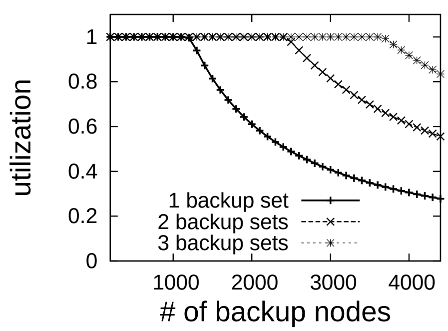
ber phases can be approximated with a Gaussian distribution by the Central Limit Theorem. According to our experiment settings in this experiment, it takes 360 phases for a sensor to consume energy to the amount of the stair height (i.e., $\frac{e}{N_{max}}$). Since all area's coverage number complies to the same distribution, their summation of coverage numbers over a large number of phases can be approximated with the same Gaussian distribution with the same mean. Thus, all areas consume energy at approximately the same average rate.

Furthermore, our scheme maintains a phase difference δ among the staircases in different areas. Thus, the time instances for all areas to send out ready message are *approximately* evenly distributed as time evolves. Therefore, both average replacement interval and utility of the MR follow the similar pattern as in Fig. 4.5.

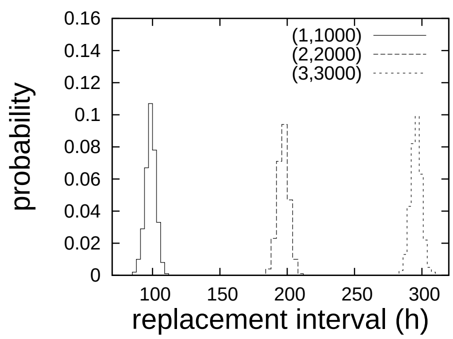
One notable difference between Fig. 4.6 and Fig. 4.5 is in the histograms of replacement intervals. The histograms in Fig. 4.5(c) are taller and narrower than the corresponding ones in Fig. 4.6(c), which implies smaller standard deviations. This is because the independence of coverage numbers of the areas brings more variance in terms of the interval between two consecutive time instances when the number of sensors that are ready to be replaced reaches x.



(a) Average replacement interval



(b) Average utilization of the MR



(c) Histogram of replacement intervals

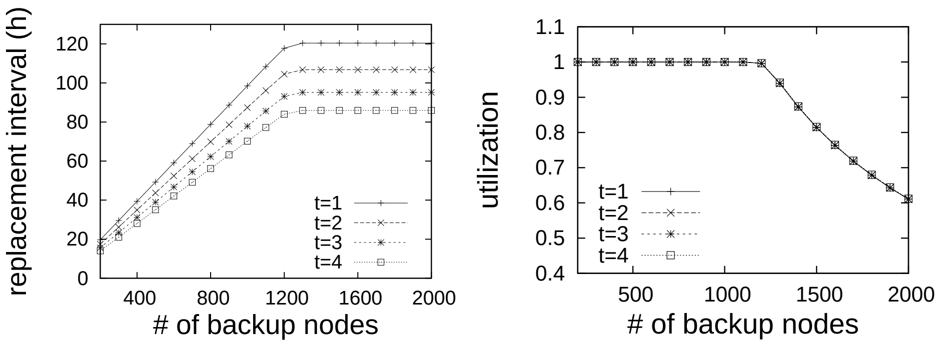
Figure 4.6 Scenario II: same coverage number distribution for all areas

4.6.4 Variable Distribution of Coverage Numbers

In this experiment, all areas have the same values of parameters N_{min} and N_{max}, and their coverage numbers comply to the same truncated Gaussian distribution and are independent of each other. In the prior experiments, we always truncate $Gau(\mu = N_{min}, \sigma = 2)$ to the range $[N_{min}, N_{max}]$ to get truncated Gaussian coverage numbers. In this experiment, we set $N_{min} = 1$, $N_{max} = 4$, and truncate $Gau(\mu = t, \sigma = 2)$ to the range $[N_{min}, N_{max}]$, where t varies in $\{N_{min}, N_{min} + 1, \cdots, N_{max}\}$, i.e., $\{1, 2, 3, 4\}$. We only consider one backup coverage set in this experiment.

Fig. 4.7 shows the trend of average replacement interval and utilization of the MR when t varies.

As can be seen, when t is larger, average replacement interval is smaller. This is because larger t implies higher energy consumption rate of the network, and thus the MR needs to replace sensors more frequently. On the other hand, the value of x where average replacement interval levels off and the utilization of the MR starts to drop is the same for all the values of t. This is because the distribution of coverage numbers does not affect the upper bound of x according to Eq. (4.3).



(a) Average replacement interval

(b) Average utilization of the MR

Figure 4.7 Variable Gaussian distribution

CHAPTER 5. RELIABLE NODE RECLAMATION AND REPLACEMENT

5.1 Introduction

In reality, sensor nodes are susceptible to failures and irregular energy consumption rate. By irregular energy consumption rate we mean when performing the same task, (i) different sensor nodes may consume their energy at different rate, and (ii) the same sensor node may consume its energy at different rate at different time. Neither the basic ARTS scheme nor the staircase-based scheme provides a complete solution to address these issues.

The ARTS scheme can take sensor failures and irregular energy consumption rate into account in the following way.

- To deal with sensor node failures, sensor nodes that are supposed to be active need to broadcast a message to other sensor nodes deployed to the same post. Other sensor nodes can detect failures by listening to this message. Once a failure is detected, the failed node is excluded from the local duty-cycle scheduling.

- To deal with irregular energy consumption rate, sensor nodes deployed to the same post can periodically broadcast the information of their remaining energy. Since these sensor nodes are close to each other, this approach does not cause significant communication overhead.

On the other hand, solving these issues under the area coverage model is a challenging task. This is because (i) sensor nodes are not interchangeable under the area coverage model, and thus a failed sensor node could leave a coverage hole in the field; and (ii) sensor nodes are deployed in a large area, which prohibits each sensor node to broadcast it remaining energy information to other sensor nodes.

Although the staircase-based scheme has been shown to achieve a good performance on minimizing the frequency of maintenance service performed by the MR and meanwhile maintaining area coverage, the presence of sensor node failures and irregular energy consumption rate may destroy the staircase structures, and largely degrade the performance of the scheme.

Therefore, we focus on the reliable staircase-based schemes in this work. To address the two issues and improve the reliability of scheduling, we propose the following three schemes.

- *The staircase repairing scheme*: This scheme is primarily designed to handle sensor failures. When node failure occurs, the staircase structures are repaired through *virtually* reducing the remaining energy of some operational sensor nodes.

- *The debit/credit scheme*: This scheme is primarily designed to handle sensor failures. When node failure occurs, the staircase structure is repaired through energy "borrowing" and "returning".

- *The energy consumption balancing scheme*: This scheme is primarily designed to handle irregular energy consumption rate. In this scheme, sensor nodes with higher energy consumption rate are scheduled less frequently, while sensor nodes with low energy consumption rate are scheduled more frequently.

5.2 System Assumptions of Reliable NRR under Area Coverage Model

The system architecture of *Reliable* NRR under the area coverage model is the same as the one of NRR under the area coverage model, which is shown in Fig. 4.1 in Chapter 4.

The system assumptions are similar to the ones of NRR under the area coverage model in Chapter 4 with the following differences. If a sensor is in the active mode all the time and the energy consumption per phase is always α_{mean}, its lifetime is denoted as T.

- Each sensor node knows its location.

- For every phase, if a sensor is in the active mode, its amount of energy consumption complies to certain distribution, with the mean denoted as α_{mean}.

Design Goal. In this work, we aim to design a collaborative scheduling scheme for sensors and the reclamation and replacement scheduling algorithm for the ES/MR, such that (i) the sensor network can maintain the required area coverage for an infinite period of time, and (ii) the number of travels the MR should take is as small as possible (i.e., the average interval between two consecutive replacement trips is as large as possible).

5.3 Proposed Schemes

We propose three schemes besides a naive scheme to cope with sensor failure and irregular energy consumption rates.

5.3.1 Sensor Failure Detection

In our scheme, when a sensor fails, its 1-hop neighbors are responsible for detecting it. Specifically, at the beginning of a phase, if a node u is supposed to be active in a phase, it broadcasts a *on-duty* message to its 1-hop neighbors. This message can be broadcast several times to make sure all u's neighbors receive it. If u's neighbors do not receive this message, they consider that u has failed. In this case, a *failure* message is broadcast to the whole area, and all sensors in the area know the information of the failed sensor, including its ID, and the ID of its coverage set.

5.3.2 Naive Scheme

The original staircase-based scheme does not consider irregular energy consumption rate and sensor failures. The scheme can be extended as follows to consider reliability issues:

(i) Whenever a sensor node in a coverage set is drained of energy, a *ready* or *deadline* message is sent to the ES. Note that what message to be sent depends on whether there are full-energy backup coverage sets. The entire coverage set then becomes a backup coverage set, and is to be replaced as a whole later.

(ii) Whenever a sensor node fails, we treat it as being drained of all energy. The coverage set that the failed sensor node belongs to becomes a backup set, and is to be replaced later.

The problem with this naive solution is that when the first sensor in a coverage set dies or fails, ready and deadline messages are sent out with irregular interval, which may corrupt the staircase structure.

Fig. 5.1 shows an example when using the naive scheme to deal with failures. Fig. 5.1(a) shows a normal case when there is no failure in an area. As can be seen, the area sends out a ready message every $T/4$. If we order the primary coverage sets according to their remaining energy, any two adjacent primary coverage sets have their remaining energy differing by $e/4$.

Fig. 5.1(b) shows a case when a sensor node in coverage set 3 fails at $t = T/8$. In the naive scheme, this coverage set becomes a backup coverage set, and coverage set 0, which is a backup coverage set with full energy, becomes a primary coverage set. Note that coverage set 0 starts to work $T/8$ earlier than in Fig. 5.1(a). Furthermore, the difference in remaining energy between coverage sets 0 and 1 is $T/8$, and the difference in remaining energy between coverage sets 2 and 4 is $T/4$. The staircase structure is deformed at this time, i.e., ready messages are sent out at intervals $\{T/2, T/4, T/8, T/8\}$, not at the fixed interval of $T/4$ as in Fig. 5.1(a). At time instance $t = T/4$, since coverage set 3 has not been replaced, a deadline message is sent out, and both coverage sets 3 and 4 will be replaced. As can be seen, the pattern of intervals for sending out ready messages remains, which causes deadline messages to be sent out again and again in future.

Since when the MR performs a replacement in response to a deadline message, it may not carry all x batteries, the MR has to performs more replacements to avoid service disruption of the network.

5.3.3 Staircase Repairing Scheme

The basic idea for staircase repairing scheme is that when the staircase structure is deformed due to failures, we repair the structure, such that each area still sends out ready messages at fixed intervals as before the failure. Specifically, when failure happens, we "reduce" the remaining energy of some stairs, such that if we sort the primary coverage sets according to their remaining energy level, any two adjacent coverage sets still keep their remaining energy differing by e/N_{max}. Note that we do not physically reduce remaining energy of a coverage set, instead we replace it earlier.

Fig. 5.1(c) shows an example. When a sensor node in coverage set 3 fails at $t = T/8$, coverage set 3 becomes a backup coverage set, and coverage set 0 becomes a primary coverage set. Then we "reduce" coverage set 0's remaining energy by $T/8$. This is done by sending out ready message when coverage set 0's remaining energy drops to $e/8$, instead of 0. As a result, the $e/8$ energy in coverage set 3 will not be used, and this part of remaining energy is shaded in Fig. 5.1(c). We further reduce coverage sets 1 and 2's energy by $e/4$, not $e/8$. Now if we sort the primary coverage sets 0,1,2, and 4 according to their remaining energy, any two adjacent coverage sets will keep their remaining energy differing by $T/4$.

At time instance $t = T/4$, a deadline message will be sent out, since coverage 3 has not been replaced. Both coverage sets 3 and 4 will be replaced at the time. After that, the area will send out a ready message every $T/4$, which is same to the normal case in Fig. 5.1(a). The staircase repairing algorithm is described in Alg. 5.3.1.

Algorithm 5.3.1 Staircase repairing algorithm: for sensor node u

Notations:
  $h[N_{max}]$: array that records the height of each stair $i, 0 \leq i \leq N_{max}$
  $stair(u)$: node u's stair
  $bottom(u)$: if u's remaining energy is lower than $bottom(u)$, then it is ready to be replaced
  $e(u)$: u's remaining energy

Initialization:
  $bottom(u) \leftarrow 0$

Upon receipt of a failure message: $failure\langle v, stair(v)\rangle$
  if $stair(u) > stair(v)$ **then**
    if $stair(u) = N_{max}$ **then**
      $bottom(u) \leftarrow bottom(u) + h[0]$
    else
      $bottom(u) \leftarrow bottom(u) + e/N_{max}$

At the beginning of each phase
  if $e(u) < bottom(u) + \alpha_{mean}$ **then**
    if there is a backup set with full energy **then**
      Shift its duty to the backup set
      Send out a ready message
    else
      Send out a deadline message
    u changes its role to backup

5.3.4 Debit/Credit Scheme

In the staircase repairing scheme, when a coverage set has a failed node, the whole coverage set are replaced. On the other hand, the debit/credit scheme only requests for replacing the failed node at the time of failure. The scheme adopts the notions of debit and credit from banking systems. Specifically, when a coverage set has a failed sensor, it becomes a backup coverage set, and another backup coverage set with full energy becomes a primary coverage set. This new primary coverage set starts to work earlier than expected because of the failure, and its remaining energy will be lower than its expected level. We can view this as that the failed coverage set "borrows" a certain amount of energy from the new primary coverage set. As long as the failed coverage set has its failed sensor replaced, it starts to "return" the energy it owes to the new primary set, until the energy level of the new primary set goes back to the expected level.

Fig. 5.1(d) shows an example. When a sensor in coverage set 3 fails at time instance $t = T/8$, it becomes a backup coverage set, and coverage set 0 becomes a primary coverage set. Meanwhile, a ready message is sent to the ES. Note that this message only asks for replacing the failed sensor node, not the entire coverage set. Since coverage set 0 starts to work $T/8$ earlier, we treat this as that coverage 0 debits $e/8$ energy to coverage set 3.

At time instance $t = T/4$, a deadline message is sent to the ES since both coverage sets 3 and 4 are not able to work normally. As a result, the failed sensor node in coverage set 3 and the entire coverage set 4 are replaced. Coverage set 3 will start to credit back $e/8$ energy to coverage set 0.

Since both coverage set 0 and 3 are primary coverage sets, the return of energy is done opportunistically. Specifically, if in a phase coverage 0 is supposed to be active, and coverage set 3 is not, coverage set 3 will be active in place of coverage set 0[1].

In the example shown in Fig. 5.1 (d), at time instance $t = 3T/16$, coverage set 3 has returned $e/16$ to coverage set 0, and at time instance $t = T/2$, coverage set has returned all $e/8$ energy to coverage set 0. At this time, the staircase structure goes back to the same shape as in Fig. 5.1 (a), and a ready message will be sent out at $t = T/2$.

5.3.5 Energy Consumption Balancing Scheme

In the case of irregular energy consumption rate, we still adopt the notion of disjoint coverage sets and staircase structure. However, the formation of staircase structure is based on the mean of energy consumption rate α_{mean}. In other words, the staircase structure is formed as if all the sensors consumes their energy at the rate α_{mean}.

[1]For phases with coverage number 4, the return of energy will not occur.

The basic idea to deal with irregular energy consumption is to balance energy consumption among sensor nodes. In other words, if a sensor node consumes energy at a high rate, we can schedule the sensor node less frequently. On the other hand, if a sensor node consumes energy at a low rate, we can schedule the sensor node more frequently.

This can be done in two ways:

(i) If a sensor with energy consumption rate higher than α_{mean} is supposed to be active, neighboring sensors with relatively lower energy consumption rate (thus relatively higher remaining energy) can take its role.

(ii) If sensors in one geographical area consume energy faster than α_{mean} on average, we need to schedule these sensors less frequently, since if these sensors die, all coverage sets need to be replaced. In this case, we will need to find other geographical areas in which sensors consume energy slower than α_{mean} on average. Then we form a chain or tree for energy transfer. Note if one replacing sensor node is sufficient to cover the area of replaced sensor node, a chain is formed. Other wise, a tree is needed. Fig. 5.2(a) shows an example of energy transfer chain. In Fig. 5.2(a), node 0 has a faster energy consumption rate, and thus its remaining energy level is lower than expected. Node 4 has a slower energy consumption rate, and thus its remaining energy level is higher than expected. When node 0 is supposed to be active, while node 1 is not, node 1 will be active in place of node 0. When node 1 is supposed to be active, while node 2 is not, node 2 will be active in place of node 1, and so on. As a result, node 0 will save its energy for one phase, while node 4 will lose its energy for one phase. Fig. 5.2(b) shows an example of energy transfer tree where each node needs two of its neighbors to cover it when it is not active.

We call the two methods *energy transfer* methods, since sensor nodes with lower energy consumption rate offer its energy to help sensors with high energy consumption rate. Our scheme fulfills this object with low communication overhead.

5.3.5.1 Terms and Notations

In our scheme, each node u maintains the information about its coverage set, and the corresponding stair. Each node also maintains the location information of its neighbors. Our scheme is composed of two algorithms: energy providing algorithm and energy requesting algorithm. Before presenting the algorithms, we give the key terms and notations.

- For each sensor u, we define *remedy coverage requirement*, $rc(u)$, which is the percentage of the area that needs to be covered by its replacing sensors if u is not active. The maximum remedy coverage requirement is 100%.

- For each sensor u, we define a coverage combination of u as a set of u's neighboring sensors that can cover at least $rc(u)$ of sensing area of u if u is not active. Since u knows the locations of all its neighbors, it can derive combinations of its neighbors that satisfy $rc(u)$. Each combination may have different number of sensors in it. Each node u maintains a set $C(u)$ which stores all the combinations that satisfy $rc(u)$.

- Each sensor node u has a status, $st(u)$, which could be *provider*, *savable*, and *non-savable*. The status *provider* means the sensor node has surplus remaining energy to offer to other nodes. The status *savable* means the node can receive energy from a providing sensor node through an energy transfer chain or a tree. The status *non-savable* means the node cannot receive energy from providing nodes.

- For each sensor u, we define a valid coverage combination of u, $VC(u)$, which is a subset of $C(u)$. For each sensor in each combination belonging to $VC(u)$, it has surplus energy itself (its status is *provider* or it can obtain energy from other nodes (its status is *savable*).

- Each node u's remaining energy is denoted as $e(u)$, while its expected remaining energy, i.e., the height of its stair, is denoted as $s(u)$.

5.3.5.2 Providing Energy

The energy providing algorithm is run every certain number I of phases, which I is system parameter. At the beginning of a phase in which the energy providing algorithm is scheduled to run, if a node u finds its remaining energy level is higher than its expected energy level, i.e., the height of its corresponding stair, by at least a given threshold t_p, it marks its status as *provider*, and broadcasts a

Algorithm 5.3.2 Providing energy: for sensor node u at the beginning of designated phases

Notations:

$C(u)$: u's coverage combination set

$VC(u)$: u's valid coverage combination set, which is a subset of $C(u)$. For all sensors in each combination in $P(u)$, their status is provider or savable

$pro(u)$: u's set for received provide messages

$st(u)$: u's current status, s:savable, n:non-savable, p:provider

$e(u)$: sensor u's remaining energy

$s(u)$: sensor u's expected remaining energy, i.e., the height of its stair

$t_p = a\alpha_{mean}$: threshold for providing energy

Initialization:

$VC(u) = \phi$
if $e(u) - s(u) > t_p$ **then**
  mark its status as $provider$: $st(u) = p$
  broadcast $provide\langle u \rangle$
else
  $st(u) = n$

Upon receipt of a provide message: $provide\langle v \rangle$

Add the message to $pro(u)$: $pro(u) \leftarrow pro(u) \cup provide$
Check whether a combination C can be found to cover u using v
if C can be found **then**
  Add C to $VC(u)$: $VC(u) \leftarrow VC(u) \cup C$
  if $st(u) = n$ **then**
    $st(u) = s$
    // u can an intermediate node to provide v's energy to other nodes
    broadcast $provide\langle u \rangle$

provide message to its neighbors. Given the average consumption rate α_{mean}, t_p can be calculated as $t_p = a\alpha_{mean}$, where a is a system parameter. The provide message has the following format:

$$provide\langle u \rangle$$

If node u's remaining energy level is not higher than its expected energy level by least t_p, it set its status to *non-savable*.

When a node v receives multiple provide messages, it checks whether the senders of these messages can form a coverage combination that satisfy $rc(v)$. If such a combination C can be found, node v adds C to its valid coverage combination $VC(v)$. If node v's status is not *savable* or *provider*, it changes its status to *savable*, and broadcasts a new provide message:

$$provide\langle v \rangle$$

If node v's status is already *savable* or *provider*, node v does not send out provide message.

This process is carried on. When there is no provide message being transmitted in the area, all sensor nodes that can be provided with energy by other nodes, no matter these nodes are its neighbors or not, will be marked as *savable* or *provider*.

Alg. 5.3.2 describes the procedure of providing energy.

5.3.5.3 Requesting Energy

The energy requesting algorithm is run at every phase. At the beginning of a phase, if a node u is supposed to be active according to the duty-cycle schedule, and finds its remaining energy is lower than its corresponding stair by at least a given threshold t_r, it requests help from other sensors. Given the average consumption rate α_{mean}, t_r can be calculated as $t_r = b\alpha_{mean}$, where b is a system parameter.

Node u checks its current status, and if the status is *non-savable*, then u cannot get help from other nodes. On the other hand, if u's status is *savable*, it checks its valid coverage combination set $VC(u)$, and randomly picks one combination. Node u then sends a request message to each sensor node in the selected combination. The format of a request message is:

$$request\langle u, s(u) - e(u)\rangle$$

, where $s(u) - e(u)$ is its energy deficit.

When a node v receives a request message, it first checks its status. If its status is *savable*, it reserves α_{mean} energy for sensor u. Then it randomly selects a combination in $VC(v)$, and forwards the request message to each of the sensor nodes in $VC(v)$.

If v's status is *provider*, v waits for a give time-out period τ, in which all requests should have been propagated to v. Then node v reserves energy for these requests. Assuming the number of requests is n, if v's energy surplus $e(v) - s(v)$ is sufficient to serve all the requests, i.e., $s(v) - e(v) > n\alpha_{mean}$, then v reserves $n\alpha_{mean}$ energy. Otherwise, v will not serve all the requests, instead it chooses $\lfloor \frac{s(v)-e(v)}{n\alpha_{mean}} \rfloor$ requests with highest energy deficit, and sends out a *reject* message to other non-selected requesting sensors.

After reserving appropriate amount of energy, if v finds its remaining energy has dropped below $s(v) + b\alpha_{mean}$, it checks its valid coverage combination set $VC(v)$. If $VC(v)$ is not empty, it changes its status to *savable*. Otherwise, it changes its status to *non-savable*, and broadcasts a *cancel* message:

$$cancel\langle v\rangle$$

When a node w receives this message, it checks its valid coverage combination set $VC(w)$, and removes the combinations that include node v. If $VC(w)$ becomes empty after the removal and its status is not *provider*, node w marks itself as *non-savable*, and broadcasts a *cancel* message:

$$cancel\langle w\rangle$$

On the other hand, if node $VC(w)$ is not empty after the removing the combinations that includes v, node w will broadcast a *provide* message. This step is necessary since there are nodes reachable from w whose status has been changed to *non-savable* due to the cancel message $cancel\langle v\rangle$.

After this process terminates, sensor v, who initiates the cancel message will not receive any request message any more.

Handling of Reject Messages: Reject messages will be forwarded back to the requesting sensor. During the forwarding, each intermediate node v will cancel the reservation that is made when receiving the corresponding request message. Further, if v has sent the corresponding request message to other nodes, it will send these nodes a *withdraw* message, and any node that receives this message will cancel the reservation that is made when receiving the corresponding request message.

Note that a providing node only sends reject messages when its remaining energy drop to below its expected energy level. Once this happens, it will not provide any energy to other nodes (its status is not *provider*) until the beginning of the next cycle for broadcasting provide messages.

Alg. 5.3.3 describes the procedure of requesting energy. For purpose of clarity, it does not include handling of reject messages.

5.4 Performance Evaluation

We built a custom simulator using C++ to evaluate the performance of the proposed scheme.

Table 5.1 General experimental settings of the reliable staircase-based schemes

field size	$1000m * 1000m$
# of areas	80
sensing range	$20m$
transmission range	$40m$
N_{min}	1
N_{max}	4
N_{back}	1
recharging time	6 hours
sensor's lifetime time	240 hours (5 days)
# of sensors per coverage set	16 (by default)
sensor's full energy	1440 units
phase length	10 minutes
α_{mean}	0.1 unit/minute
provide broadcasting interval I	$\{25, 50, 75, 100, 125\}$ phases
cut-off time	4800 hours (200 days)

5.4.1 Experimental Settings, Metrics and Methodology

Table 5.1 shows system parameters we used in the simulation. We consider a sensor network composed of 80 areas. The network field size is $1000m * 1000m$. Each area has $(N_{max} + N_{back})$ disjoint coverage sets, and each of which is able to cover the whole area. The number of sensors in each coverage set is 16 unless otherwise mentioned. The method for deploying sensor nodes is as follows: For each coverage set, we randomly deploy its first sensor, and then deploy other sensor in a way such that every two adjacent sensors are $25m$ apart.

In the experiments, we normalize the full energy level of a sensor to 1440 units and the mean value of energy consumption rate, α_{mean}, is 0.1 unit/minute if the sensor is active. Thus, each sensor's lifetime T is 240 hours, i.e., 5 days. The length of a phase is set to 10 minutes. Coverage numbers for each area vary between N_{min} and N_{max}. N_{min} is set to 1, and N_{max} is set to 4 in all experiments.

In reality, coverage number is determined by the application, as well as the real-time frequency and distribution of events. In our simulation, coverage number complies to a truncated Gaussian distribution, which is $Gau(\mu = N_{min}, \sigma = 2)$ truncated to the range $[N_{min}, N_{max}]$.

The performance metrics include:

- *Average replacement interval*: Average time between two consecutive replacement tours made by the MR.

- *Average utilization of the MR*: The MR may not carry x sensors in each replacement tour due to the replacement deadlines set by each area. Average utilization of the MR is the average ratio of the number of backup sensors actually carried by the MR to x.

- *Communication overhead* The total number of control messages, including provide, request, cancel, reject, withdraw messages per area per phase.

We consider the following sets of scenarios:

(i) The system is error-free, but sensors consume energy at different rates. We model the energy consumption rate in the following way: For each sensor u, its have a mean value $\alpha_{mean}(u)$ of its energy consumption rate, which is determined by manufacture reasons. $\alpha_{mean}(u)$ is a random variable which complies to $Gau(\alpha_{mean}, \sigma_1)$. Sensor u's energy consumption rate $\alpha(u)$ at a certain phase is another variable which complies to $Gau(\alpha_{mean}(u), \sigma_2)$. σ_1 and σ_2 are system parameters.

In this scenario, we study the performance of the energy balance scheme under different the system parameters, including the number of backup sensors x, σ_1 and interval for broadcasting provide messages I.

(ii) The system has failures, and also sensors consume their energy at different rates. We model failure events in the following way: at any phase, a failure event occurs with a certain probability f_p. If there is a failure event in a phase, the failure could happen at any sensor node with equal probability.

In this scenario, we fix system parameters σ_1, σ_2 and interval for broadcasting provide messages I, and compare the performance of the staircase repairing and the debit/credit scheme under different system parameters, including the number of backup sensors x and f_p.

For each experiment, our proposed scheme is executed for a long time period, starting at 0 and ending at a *cut-off* time. The cutoff time is set to 4800 hours, i.e, 200 days, for all experiments. Furthermore, we run each simulation for 50 times for each of the performance metrics.

5.4.2 Scenario I: Irregular Energy Consumption Rate

In this section, we study the performance of the proposed energy balancing scheme comparing with the naive scheme, and the impact of different parameters on the performance of the energy balancing scheme. In all experiments in this scenario, all sensor nodes have the same *remedy coverage requirement*, and the value can be either 50% or 70%. Further, the threshold for providing energy, and the threshold for requesting energy are $4\alpha_{mean}$, i.e., $t_p = t_r = 4\alpha_{mean}$.

5.4.2.1 Impact of Number of Backup Sensors

In this experiment, we vary the number of backup sensors from 200 to 1500, and fix the provide broadcasting interval to 50 phases. We compare the energy balancing scheme when the remedy coverage requirement is 50% and 70% for all sensors, with the naive scheme in terms of the average replacement interval and the average MR utilization. As can be seen from Fig. 5.3, the energy balancing scheme has much better performance when the coverage requirement is 50%. The reason is that sensor nodes can find many providing sensors to help them. Further, all curves in 5.3(a) level off when x exceeds a certain value. This is because irregular energy consumption deforms the staircase structure, which incurs deadline messages being sent. In this case, the MR does not fully utilize the x backup sensors. In other words, more backup sensors will not help increase the replacement interval. The energy balancing scheme postpones the point when the replacement interval stops to increase.

5.4.2.2 Impact of σ_1

In this study, we fix the number of backup sensors to 500, and provide broadcast interval to 50 phases. σ_1 is varied among $\{0.06, 0.08, 0.1, 0.12, 0.14\}$. σ_2 is fixed at 0.2. Fig. 5.4 shows the average replacement interval and average MR utilization for the energy balancing scheme when the remedy coverage requirement is 50% and 70%, and the naive scheme. As shown in the figure, both performance metrics decrease when σ_1 increases. Larger σ_1 causes more irrgularity on the time interval for a sensor to use up a coverage set, and thus causes more deadline messages being sent.

5.4.2.3 Impact of Provide Broadcast Interval I

In this study, we investigate the impact of provide broadcast interval I on the performance of the energy balancing scheme when the remedy coverage requirement is 50% and 70%, respectively. Fig. 5.5(b) shows the average number of control messages, including provide, request, cancel, reject, withdraw messages per area per phase. As can be seen, the communication overhead of the energy balancing scheme is fairly low. For instance, when the remedy coverage requirement is 50% and I is 50 phases, each area sees 4.12 messages on average.

5.4.3 Scenario II: Failures and Irregular Energy Consumption Rate

We set $\sigma_1 = 0.05$, $\sigma_2 = 0.2$, the provide broadcast interval $I = 50$ phases, the remedy coverage requirement for all sensors to be 50%, and the threshold for providing energy and the threshold for requesting energy to be $4\alpha_{mean}$. We vary the number of backup sensors x and failure probability f_p, and compare the performance of the staircase repairing scheme and the debit/credit scheme.

5.4.3.1 Impact of Number of Backup Sensors x

In Fig. 5.6, we compare three schemes in which "no action" means we only run the energy balancing scheme without other schemes to deal with failure. The other two schemes work together with the energy balancing scheme. As can be seen from Fig. 5.6(a), the staircase repairing scheme and the debit/credit schemes perform much better than "no action". Furthermore, the debit/credit scheme performs better than the staircase repairing scheme when x is small, but performs worse when the x exceeds a certain value. This can be explained as follows.

When x is small, the demand for replacement exceeds x quickly, and thus the MR's replacing activities is mainly driven by the demand for replacement exceeding x. In the staircase repairing scheme, when a sensor node dies, the whole coverage set is replaced. This increases the demand for replacement, and hence the MR performs more replacements. However, when x is large enough, deadline messages will arrive before the demand for replacement exceeding x, and the MR's activities become driven by deadline messages. Since when the MR sets out for replacement in response to a deadline message, the demand is often less than x, a little more demands do not have a significant impact. This can also be seen from Fig. 5.6(b), which shows that the staircase repairing scheme has higher MR utilization.

There is also another reason that explains why the staircase repairing scheme performs better than the debit/credit scheme when x is large. In the staircase repairing scheme, when a sensor in a coverage set fails, the sensors in coverage sets with a higher stair will "reduce" their energy to one stair down. However, sensors in such a coverage set will not reduce the same amount of energy. Sensors with higher remaining energy could reduce more amount of energy, while sensors with lower remaining energy could reduce less. In other words, a sensor failure gives sensors in some coverage sets an opportunity to narrow down their difference in remaining energy level, thus helps keep the staircase structure.

5.4.3.2 Impact of Failure Probability f_p

In this study, we fix $x = 500$, and vary failure probability f_p among $\{0.001, 0.005.0.01, 0.015, 0.02\}$.

Fig.5.7 shows that as f_p increases, both average replacement interval and average utilization of the MR decreases. This is because more failures cause more deadline messages being sent, which forces the MR to perform replacement without making full use of available backup sensors.

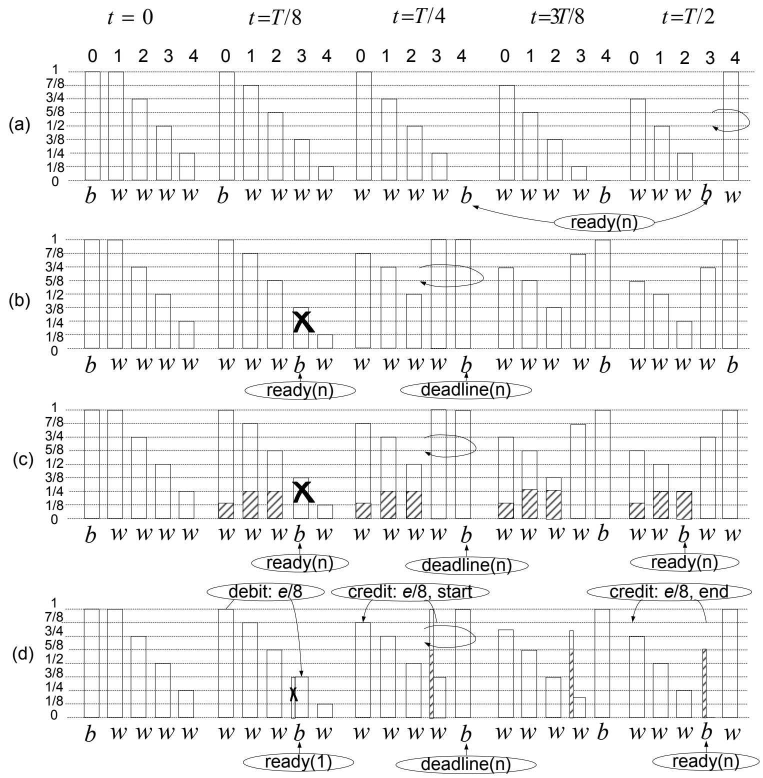


Figure 5.1 Failure example for an area. Each bar represents a coverage set. $N_{max} = 4$, $N_{backup} = 1$. Each coverage set has n sensors. When there is no deadline message in the system, the area will be visited by the MR every $T/4$. Each bar under a U-turn arrow represents a sensor which is just replaced. The parameter in ready/deadline messages is the number of sensors that need to be replaced concerning this message. (a) Failure-free case: the area sends out an $ready(n)$ message every $T/4$. (b) Naive scheme for failure. (c) Staircase repairing scheme. Shaded part of remaining energy will not be used. (d) Debit/credit scheme. Shaded part of remaining energy will not be used.

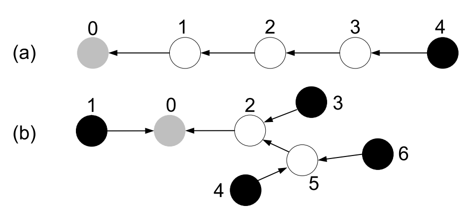


Figure 5.2 Energy transfer methods. Gray nodes are recipients of energy, dark nodes are providers of energy, and other nodes serve as intermediate nodes to help transfer energy. (a) Energy transfer chain. (b) Energy transfer tree.

Algorithm 5.3.3 Requesting energy: for sensor node u at the beginning of every phase

Notations:

$VC(u)$: u's valid coverage combination set
$req(u)$: u's set for received request messages
$st(u)$: u's current status, s:savable, n:non-savable, p:provider
$e(u)$: sensor u's remaining energy
$s(u)$: sensor u's expected remaining energy, i.e., the height its stair
$t_r = b\alpha_{mean}$: threshold for requesting energy
$t_p = b\alpha_{mean}$: threshold for providing energy

Initialization:

if $s(u) - e(u) > t_r$ **then**
  if $u_s = s$ **then**
    Randomly select a combination C in $VC(u)$, and send a request $request\langle u, s(u)-e(u)\rangle$ to each sensor in C

Upon receipt of a request message from a neighbor w: $request\langle v, s(v) - e(v)\rangle$

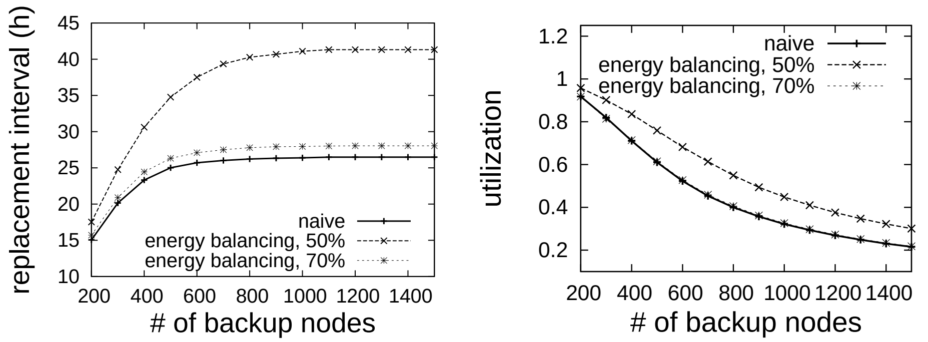
Reserve α_{mean} energy for w
if $st(u) = p$ **then**
  Add the message to $req(u)$: $req(u) \leftarrow req(u) \cup \langle w, request\rangle$
  if timer T is not started **then**
    Start timer T
else
  // u's status must be savable, i.e., $st(u) = s$
  Randomly select a combination C from $VC(u)$, and send $request\langle v, s(v) - e(v)\rangle$ to each sensor in C.

Upon timer T fired

// u must initiate a provide message, i.e., $st(u) = p$
Assume u receives n request messages
if $e(u) - s(u) > n\alpha_{mean}$ **then**
  Reserve energy $n\alpha_{mean}$
else
  Choose $\lfloor \frac{e_u - s_u}{\alpha_{mean}} \rfloor$ request messages with the highest energy deficit, and send a reject message to other requesting sensors
Reset the timer
Assume the reserved amount of energy is e_r
if $e(u) - e_r < s(u) + t_p$ **then**
  if $VC(u) = \phi$ **then**
    $st(u) = n$
    Broadcast cancel message $cancel\langle u\rangle$
  else
    $st(u) = s$

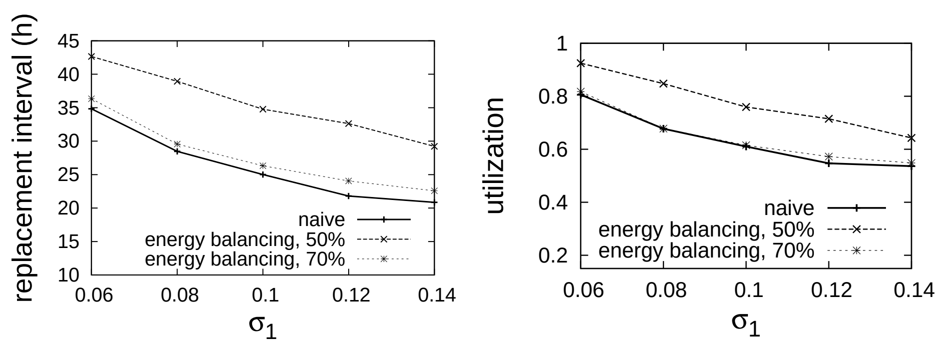
Upon receipt of a cancel message: $cancel\langle v\rangle$

Remove all combinations in $VC(u)$ which includes v
if $st(u) = s$ **then**
  if $VC(u) = \phi$ **then**
    Broadcast cancel message $cancel\langle u\rangle$
    $st(u) = n$
  else
    Broadcast provide message $provide\langle u\rangle$



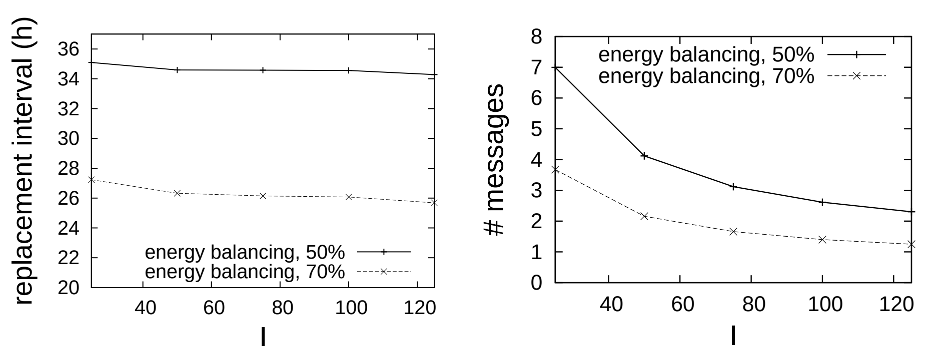
(a) Average replacement interval (b) Average utilization of the MR

Figure 5.3 Impact of x



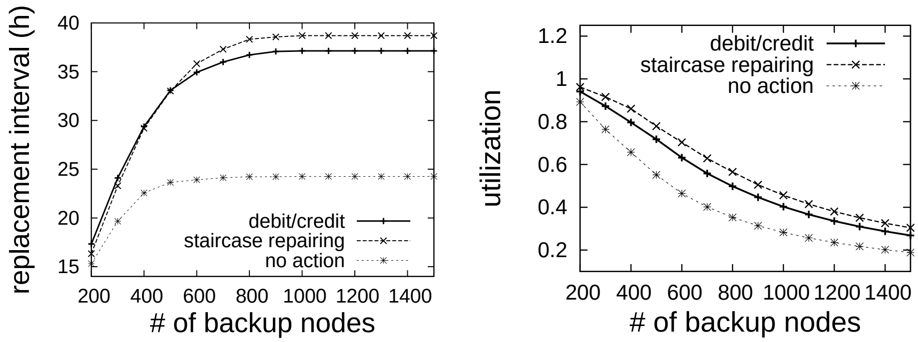
(a) Average replacement interval (b) Average utilization of the MR

Figure 5.4 Impact of σ_1



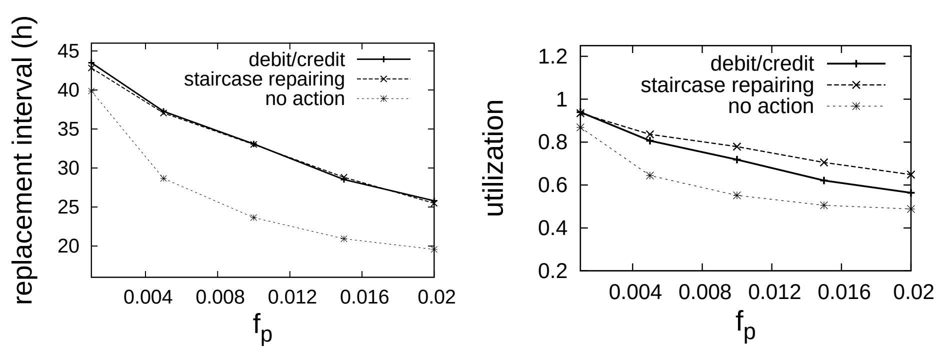
(a) Average replacement interval (b) Average utilization of the MR

Figure 5.5 Impact of provide broadcast interval I



(a) Average replacement interval (b) Average utilization of the MR

Figure 5.6 Impact of x



(a) Average replacement interval (b) Average utilization of the MR

Figure 5.7 Impact of failure probability f_p

CHAPTER 6. WIRELESS RECHARGING

6.1 Introduction

Many sensor networks are deployed to a sensing field which is dangerous, costly or technically infeasible for human or its delegates to access. For example, in structure health monitoring and factory monitoring applications [57, 58], sensor nodes are often embedded in or tightly attached to the wall, the surface of bridge, the container of hazard materials, etc. In this case, physical reclamation or replacement of sensor nodes is not a viable solution.

The new advance of wireless charging technology [22] casts a light on this problem. The technology enables an energy charger to transfer energy over radio to an energy receiver which is several feet away. Therefore, a mobile recharger can be used to move to the vicinity of sensor nodes and recharge them.

To support long network lifetime with the wireless recharging approach, the recharging agents' activities and sensors' activities can be scheduled in a similar way as with the node reclamation and replacement approach. Therefore, in this work, we focus on a unique problem with the wireless recharging technology, that is, how wireless recharging affects sensor network deployment and routing arrangement.

6.1.1 Feasibility of Wireless Rechargeable Sensor Networks

We have conducted field experiments with equipments from Powercast [22], where a charger continuously sends out RF radio in frequency 903-927 MHz to rechargeable sensor nodes. The preliminary experiments, as detailed in Chapter 6.2, demonstrate the feasibility of applying the wireless charging technology in sensor networks. The charger and sensor nodes could be several feet apart without alignment. It can be anticipated that robots, vehicles or even human operators carrying wireless chargers can move around and recharge sensor nodes deployed on the ground, and that climbing robots [50] can recharge sensor nodes deployed to the walls or tops of high buildings.

6.1.2 Research Problem

The cost of long-term energy recharging is fundamentally determined by two factors, namely, long-term energy consumption rate in the sensor network, and long-term recharging efficiency to the network (i.e., energy recharged to the network vs. energy consumed by the recharger). To minimize the energy recharging cost, the energy consumption rate of the network should be reduced and the recharging efficiency should be improved. As discussed below, these two goals are difficult to accomplish simultaneously.

To improve recharging efficiency, we propose a new deployment strategy motivated by our field experiment result. As detailed in Chapter 6.2, our experiments show that when there is single sensor receiver $20cm$ away from a charger, the typical charging efficiency is less than 1% and more than 99% energy is wasted in the air. However, as the number of sensors being charged simultaneously increases, the total energy obtained by all the sensors increases approximately linearly. The experimental results motivate us to propose a new deployment strategy, which deploys multiple nodes together in each post and let them work in a rotation manner. Considering the low cost of sensor nodes and generally employed redundant deployment methodology, deploying multiple nodes in one post can increase the recharging efficiency and fault tolerance while decrease long-time recharging maintenance cost (i.e., recharging cost). Thus it is a choice of high performance/cost ratio. How many nodes should be deployed in each post is affected by the energy consumption rate in the post. The higher the rate, the more nodes should be deployed, such that the recharger does not need to come frequently to the post to recharge nodes and meanwhile the recharging efficiency is high. On the other hand, if a post has multiple nodes and thus has a high recharging efficiency, more workload should be allocated to these nodes, such that nodes with low recharging efficiency (in other posts) can be allocated with low workload to reduce their energy consumption.

To increase energy efficiency, i.e., reduce the energy consumption rate of the network, an optimal communication topology and routing arrangement should be found such that the overall data reporting activities can follow the most energy efficient routes from sensors to the sink. This is especially important by considering communication is usually the biggest source of energy consumption. By adjusting energy level, nodes can have different communication range, and thus there exist a large number of possible topologies and routes to choose from. The optimal one depends on the locations of posts and workload at sensor nodes deployed to each post.

The energy efficiency-targeted routing arrangement and the recharging efficiency-targeted node deployment cannot be determined independently and simply merged together to achieve the minimum energy recharging cost. Instead, they are entangled together. On one hand, the routing arrangement affects the energy consumption rate at every post; specifically, a post passed through by more packets has higher energy consumption rate than that passed through by less packets. This in turn affects node deployment decision because more nodes should be deployed in where energy consumption rates are high. On the other hand, node deployment also affects the routing decision. If a post has more nodes deployed and hence a higher charging efficiency, it should be assigned more forwarding tasks. Due to the above reasons, *the optimal decisions on routing arrangement and node deployment should be made at the same time* to minimize the total recharging cost of the system, which is the problem studied in this work.

In this work, we prove the problem is NP-complete. To address the problem efficiently and effectively, we also propose a set of heuristic algorithms: the routing-first heuristic (RFH), the iterative version of RFH, and the incremental deployment-based heuristic (IDB).

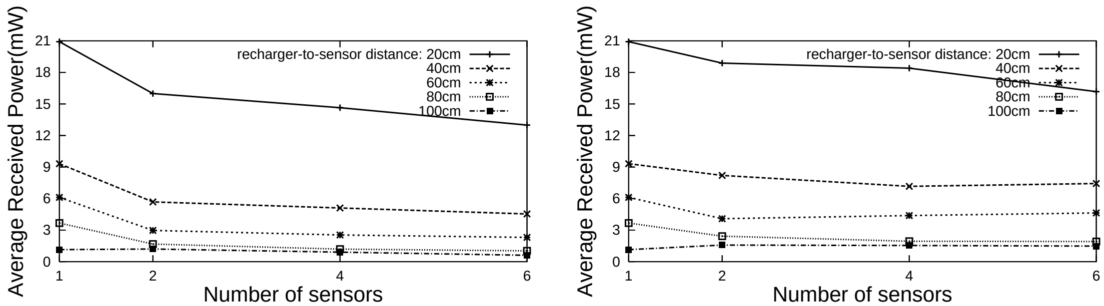
6.2 Preliminary: Field Experiments and Observations

We have conducted field experiments to study the feasibility of recharging sensor nodes in a wireless fashion with equipments provided by Powercast [22], and collected associated data. The results show that the efficiency to recharge a single node is low and most of the energy is wasted when propagated in the air. Particularly, when a sensor is $20cm$ away from the charger, on average the node can obtain less than 1% of the energy transmitted by the charger. As the distance increases, the efficiency decreases exponentially.

To study how recharging efficiency can be improved, we conduct experiments on recharging multiple sensor nodes simultaneously. We vary three parameters, the number of nodes being recharged

Table 6.1 Field experiment on wireless charging

Parameter	Value
Number of sensors	1, 2, 4, 6
Charger-to-sensor distance	20cm, 40cm, 60cm, 80cm, 100cm
Sensor-to-sensor distance	5cm, 10cm



(a) sensor-sensor distance = 5cm (b) sensor-sensor distance = 10cm

Figure 6.1 Field experiment result

simultaneously, the distance between sensor nodes, and the distance between the sensor nodes and the charger. Table 6.1 summarizes the values of these parameters used in the experiment. For each value of the three parameters, we conduct 40 experiments and plot the average of the received power in Fig. 6.1.

Both figures show that, when the number of sensor nodes charged simultaneously increases from 2 to 6, the average power received at each node remains approximately the same, i.e., the efficiency for charging energy to the network (note: not the charging efficiency for a single node) has a linear relationship with the number of sensors being charged simultaneously. When the number of nodes changes from 1 to 2, a noticeable decrease in the average power received by each node is observed when sensor-sensor distance is 5cm, the difference decreases when the sensor-sensor distance increases to 10cm.

In addition, comparing Fig. 6.1(a) and (b), we can see that when sensor-sensor distance becomes larger, the charging efficiency increases more when multiple sensors are charged together. This is because the RF power sent out by the charger at a certain direction is finite. When sensors are more spread out, they can better capture the energy in the air. Considering $10cm$ is a relatively short distance, the linear relationship between charging efficiency and the number of sensors being charged simultaneously can be more obvious when sensor-sensor distance increases.

6.3 System Model

Focusing on how wireless charging technology affects network deployment and routing arrangement, we consider the following simplified system model.

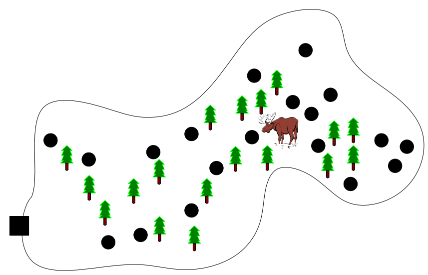


Figure 6.2 Example of post configuration in an island. The solid square represents the base station, and the solid circles represent post.

As shown in Fig. 6.2, a sensor network is deployed in a field for long-term, continuous monitoring. The field has N posts of interest and each post must have at least one sensor node deployed. The locations of the posts are determined by applications based on the shape of the terrain, the required sensing quality, etc., and are given. The network has M sensor nodes ($N \leq M$). Sensor nodes monitor their nearby environment and every certain time interval, at least one node at each post generates a report. The report will be forwarded hop by hop to the base station, which is located at a corner of the deployment field. If a post has multiple nodes deployed, these nodes rotate in performing the sensing/reporting tasks such that they maintain nearly the same level of residual energy level.

Each node is assumed to have k transmission levels (denoted as $l_1, \cdots, l_k$), which enables it to transmit a message to the distances of $d_1(d_{min})$, d_2, $\cdots$, d_{k-1} and $d_k(d_{max})$, respectively. Assume the energy consumed for transmitting one bit to distance d_t is denoted as e_t, and the energy consumed for receiving one bit is denoted as e_r. e_t and e_r can be calculated as follows:

$$\begin{cases} e_t &= \alpha + \beta d^\gamma, \\ e_r &= \alpha \end{cases} \tag{6.1}$$

where α is the energy needed to run the transceiver circuitry, β is the energy consumed in the amplifier circuitry to transmit the data, and γ is the loss factor, which varies from 2 to 4, depending on the quality of channel. We assume $\gamma = 4$ in the work. Based on Eq.(6.1), the amount of energy for transmitting one bit when using each of the k power levels can be computed, and the value is denoted as e_i ($i = 1, \cdots, k$). Note that, in this work, we only consider the energy consumption for packet transmission and reception, the biggest source of energy consumption. However, the results can be extended to other sources of energy consumption such as sensing and computation.

We assume sensor nodes can always be recharged in time before they run out of energy. How to schedule the wireless charger to guarantee this is not the focus of this work. We denote the charging efficiency when a charger recharges a single sensor node to be η ($0 < \eta < 1$). If the recharger disseminate y units of energy and the sensor receives x, $\eta = \frac{x}{y}$. The charging efficiency increases if the charger simultaneously recharges multiple sensors. When charging m sensor nodes simultaneously, the charging efficiency becomes a function of m: $\eta(m) = k(m) * \eta$. Our field experiment shows that $k(m)$ is a linear or sub-linear function of m. To get a quantitative result of sensor deployment, we assume $k(m) = m$ in this work. Since simultaneous charging increases charging efficiency, it is beneficial to deploy multiple sensor nodes together to a post whenever possible.

6.4 Problem Definition and Its Nature

6.4.1 Problem Definition

The problem of determining the optimal node deployment and routing arrangement can be formulated as follows. Given:

- M sensor nodes are in the network and a base station is connected to some of the nodes.
- Each node has k levels of transmission power ($l_1, \cdots, l_k$). At level l_i ($i \in \{1, \cdots, k\}$), the transmission range is d_i and the energy to transmit one bit is e_i.
- There are N deployment posts ($p_1, \cdots, p_N$). Each post needs at least one node deployed.
- If post p_i ($i \in \{1, \cdots, N\}$) has been deployed with m_i ($m_i \geq 1$) nodes, the charging efficiency at p_i is $m_i * \eta$. That is, for every unit of energy consumed by the charger, each of the m_i nodes at p_i can receive η units of energy.

The problem is to

(a) determine how to deploy M sensor nodes to N posts;

(b) for each post p_i ($i \in \{1, \cdots, N\}$), determine the transmission power level that should be used and which post should be chosen as its parent,

such that:

- Based on the chosen transmission power level and parent for each post, packets generated by each sensor node can be transmitted to the base station.

- To maintain infinite network lifetime, the average amount of energy that the charger should consume per time unit is minimized.

6.4.2 Nature of the Problem

Next, we prove that the afore-defined problem is NP-complete. To ease the proof, we restrict the problem a bit, and show that even the restricted problem is NP-complete. The original, more general problem is therefore also NP-complete. Our restrictions are as follows:

- Each node has 2 transmission power levels l_1 and l_2 and $4e_1 = e_2$. The amount of energy for each node to receive one bit is denoted as $e_0 (e_0 < e_1)$.

- Each post can have at most two sensor nodes. Note that, posts with two sensor nodes have twice charging efficiency than posts with one sensor node.

The proof is as follows.

Proof. First of all, we show that the problem is in NP. Clearly, if how m sensor nodes are deployed in n posts is given, and the transmission levels and the parent choices of n posts are also given, the total energy cost at the charger can be calculated. It is determinable if the cost is no greater than a given value W. Therefore, the problem is in NP.

Next, we prove the problem is NP-hard by reducing the 3-CNF SAT problem to this problem.

Suppose there is an instance of the 3-CNF SAT problem which consists of n Boolean variables $x_1, x_2, \cdots, x_n$, and m conjunctive normal forms (CNFs) $C_1, C_2, \cdots, C_m$, where for each $j \in \{1, \cdots, m\}$,

$C_j = y_{j,1} \vee y_{j,2} \vee y_{j,3}$ and the three literals $y_{j,1}, y_{j,2}, y_{j,3} \in \{x_1, \bar{x}_1, x_2, \bar{x}_2, \cdots, x_n, \bar{x}_n\}$. We can construct an instance of our problem as follows.

- Let a network have $M = 3n + 3m$ sensor nodes and $N = 2n + 2m$ posts. That is, $n + m$ posts should have two sensor nodes each, and the rest $n + m$ posts should have only one sensor node each.

- The posts are constructed as follows: (a) for each CNF clause, there are two corresponding posts U_j and V_j, $1 \leq j \leq m$; (b) for each Boolean variable x_i, $1 \leq i \leq n$, there are two corresponding posts $S_{i,1}$ and $S_{i,2}$.

- The base station can be directly reached by any post U_j, $1 \leq j \leq m$, only if they set their transmission power to l_2, but it cannot be reached directly by other posts.

- Assuming the three literals of CNF clause C_j ($1 \leq j \leq m$) are $y_{j,1}$, $y_{j,2}$ and $y_{j,3}$, if x_i is one of these literals, post $S_{i,1}$ can reach U_j only when using transmission power l_2; if $\bar{x}_i$ is one of these literals, post $S_{i,2}$ can reach U_j only when using transmission power l_2.

- Each pair of posts $S_{i,1}$ and $S_{i,2}$ ($1 \leq i \leq n$) can reach each other when using transmission power l_1.

- Each V_j ($1 \leq j \leq m$) can reach the same set of posts as U_i does except the base station, when using transmission power l_1.

Fig. 6.3 shows an example of the constructed instance. Let $W = 7m\frac{e_1}{\eta} + 9n\frac{e_1}{\eta} + m\frac{e_0}{\eta} + n\frac{3e_0}{2\eta}$. We claim that

(i) if there exists an assignment of Boolean values to $x_1, x_2, \cdots, x_n$ such that the instance of 3-CNF SAT is evaluated to be true, then there is a solution to the afore-constructed instance of our problem in which the total energy cost for recharging the afore-constructed network is no greater than W; and

(ii) the reverse of Claim (i).

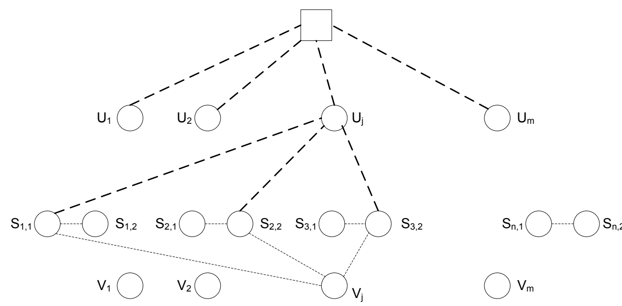


Figure 6.3 NP-Completeness proof. The square represents the base station, and the circles represent posts. Thick dotted lines indicate two end posts can reach each other using transmission power l_2, and thin dotted lines means two end posts can reach each other using transmission power l_1. This example assumes $C_j = x_1 \vee \bar{x}_2 \vee \bar{x}_3$.

Firstly, we prove Claim (i). Suppose there is an assignment of Boolean values to $x_1, x_2, \cdots, x_n$, which satisfies the instance of 3-CNF SAT, we construct a solution to our problem as follows. For each post U_j, $1 \le j \le m$, we deploy two sensor nodes, and they use transmission power l_2 to send data to the base station. For a 3-CNF clause $C_j = y_{j,1} \vee y_{j,2} \vee y_{j,3}$, without losing arbitrariness, let us assume literal $y_{j,k}$ ($1 \le k \le 3$) is true. So there will be two cases: $y_{j,k} = x_i$ or $y_{j,k} = \bar{x}_i$. If $y_{j,k} = x_i$, we do the following:

- Two sensor nodes are deployed to post $S_{i,1}$, and one sensor node is deployed to post $S_{i,2}$.
- $S_{i,1}$ uses transmission power l_2 to send data to U_j, and $S_{i,2}$ uses transmission power l_1 to send data to $S_{i,1}$.
- One sensor node is deployed to each V_j, $1 \le j \le m$, which uses transmission power l_1 to send data to $S_{i,1}$.

On the other hand, if $y_{j,k} = \bar{x}_i$, we do the following:

- Two sensor nodes are deployed to post $S_{i,2}$, and one sensor node is deployed to post $S_{i,1}$.
- $S_{i,2}$ uses power level l_2 to send data to U_j, and $S_{i,1}$ uses transmission power l_1 to send data to $S_{i,2}$.

- One sensor node is deployed to each V_j, $1 \le j \le m$, which uses transmission power l_1 to send data to $S_{i,2}$.

In this way, we have distributed all $3m + 3n$ sensor nodes to the $2m + 2n$ posts, and have chosen transmission power levels and parents for all posts. Next we show the total energy cost for recharging this network is no greater than W.

- To compensate the energy consumed for reporting one bit information at each post U_j ($1 \le j \le m$), the amount of energy consumed at the charger is $\frac{4e_1}{2\eta}$. Therefore, the total for all the m posts is $2m\frac{e_1}{\eta}$.

- For each pair of posts $S_{i,1}$ and $S_{i,2}$ ($1 \le i \le n$), one of them (with two sensor nodes deployed) incurs an energy cost of $\frac{4e_1}{2\eta} + \frac{4e_1}{2\eta} + \frac{e_0}{2\eta}$ for every bit information it has reported to the base station ($\frac{4e_1}{2\eta}$ incurred at itself, another $\frac{4e_1}{2\eta}$ and $\frac{e_0}{2\eta}$ incurred at post U_j for forwarding and receiving this data, respectively), and the other (with one sensor node deployed) incurs an energy cost of $\frac{4e_1}{2\eta} + \frac{4e_1}{2\eta} + \frac{e_1}{\eta} + 2\frac{e_0}{2\eta}$. Therefore, the total for all the $2n$ posts is $n\frac{e_1}{\eta} * (2 + 2 + 2 + 2 + 1) + n\frac{e_0}{2\eta}(1 + 2) = 9n\frac{e_1}{\eta} + 3n\frac{e_0}{2\eta}$.

- For each post V_j ($1 \le j \le m$), the energy cost is $\frac{4e_1}{2\eta} + \frac{4e_1}{2\eta} + \frac{e_1}{\eta} + 2\frac{e_0}{2\eta}$ for every bit information it has reported. Therefore, the total energy cost for all the m posts is $m\frac{e_1}{\eta} * (2 + 2 + 1) = 5m\frac{e_1}{\eta} + m\frac{e_0}{2\eta}$.

Summing up the above amounts of different types of posts, we obtain the total energy cost for recharging the network for each bit that every post has reported, which is $7m\frac{e_1}{\eta} + 9n\frac{e_1}{\eta} + m\frac{e_0}{\eta} + n\frac{3e_0}{2\eta} = W$.

Secondly, we prove Claim (ii). To prove this claim, we first show that if there is a solution to the afore-constructed instance of our problem, the network must satisfy the following two properties:

(ii-A) Each post U_j ($1 \le j \le m$) has two sensor nodes; Each post V_j ($1 \le j \le m$) has one sensor node; and for each pair of posts $S_{i,1}$ and $S_{i,2}$ ($1 \le i \le n$), exactly one of them has two sensor nodes, and the other has only one sensor node.

(ii-B) Given the distribution method of sensor nodes stated in Property (ii-A), there is only one way to choose the transmission power level and the parent for each post, such that the total energy cost for recharging the network is no greater than W. The way to choose the transmission power level and the parent post for each post is as follows: (a) each post U_j ($1 \le j \le m$) uses transmission power level l_2 to send data to the base station; (b) for each pair of posts $S_{i,1}$ and $S_{i,2}$, the post with two sensor nodes uses transmission power level l_2 to send data to a post U_j ($1 \le j \le m$), and the other uses transmission power level l_1 to send data to the former, and (c) each post V_j ($1 \le j \le m$) uses transmission power level l_1 to send data to a post $S_{i,k}$ ($1 \le i \le n, 1 \le k \le 2$) which has two sensor nodes.

We now prove Property (ii-B). Firstly, it is clear that the afore-described way for choosing the transmission power level and the parent post of each post results in a total energy cost of W. Secondly, we want to prove that, if there is another way for choosing the transmission power and the parent post, there exists a sequence of transformations which results in another set of choices of the transmission power level and the parent post with less amount of total energy cost. In other words, any way for choosing the transmission power and the parent post that is different from the one described in (ii-B) will incur a total cost that is greater than W. The transformations are as follows:

- For pairs of posts $S_{i,1}$ and $S_{i,2}$ ($1 \le i \le n$): Without the loss of generality, assume $S_{i,1}$ ($1 \le i \le n$) has only one sensor node, and it uses power level l_2 to send data to a post U_j ($1 \le j \le m$). We can reset $S_{i,1}$'s power level to l_1, and let it send data to $S_{i,2}$, which has two sensor nodes. Clearly, the energy cost for recharging $S_{i,1}$ and $S_{i,2}$ is reduced without affecting other posts.

- For posts V_j ($1 \le j \le m$): Without the loss of generality, assume V_j uses transmission power l_1 to send data to post $S_{i,1}$ ($1 \le i \le n$) which has only one sensor node. We can let V_j to send data to $S_{i,2}$, which has two sensor nodes, with a reduced energy cost for recharging $S_{i,1}$ and $S_{i,2}$ without affecting other posts.

We next prove Property (ii-A). Suppose there is a way to distribute $3m+3n$ sensor nodes into $2m+2n$ posts which is different from the way described in (ii-A), there exists a series of transformations to re-distribute sensor nodes such that with the resulting deployment, less amount of total energy cost can be obtained. The transformations are as follows:

- For each post U_j that has only one sensor node, there must be either a post V_j having two sensor nodes, or a pair of posts $S_{i,1}$ and $S_{i,2}$ both having two sensors. If we move a sensor node from V_j or either of $S_{i,1}$ and $S_{i,2}$ to U_j, it is clear that the total energy cost is reduced.

- For each pair of posts $S_{i,1}$ and $S_{i,2}$ that both have only one sensor node, there must be either a post V_j having two sensor nodes, or another pair of posts $S_{i',1}$ and $S_{i',2}$ both having two sensor nodes. If we move a sensor node from V_j or either of $S_{i',1}$ and $S_{i',2}$ to one of $S_{i,1}$ and $S_{i,2}$, it is clear that the total energy cost is also reduced.

Therefore, both Properties (ii-A) and (ii-B) hold when there is a solution to an instance of our problem.

Based on Properties (ii-A) and (ii-B), we can assign Boolean values to the corresponding instance of the 3-CNF problem as follows: For each pair of post $S_{i,1}$ and $S_{i,2}$, if $S_{i,1}$ has two sensor nodes, then we let $x_i = true$; on the other hand, if $S_{i,2}$ has two sensor nodes, then we let $\bar{x}_i = true$. Due to the way we construct the network, each post U_j ($1 \leq j \leq m$) must have at least one post $S_{i,k}$ ($1 \leq i \leq n, 1 \leq k \leq 2$) as its child. Furthermore, if $k = 1$, then x_i is a literal in 3-CNF clause C_j; and if $k = 2$, then $\bar{x}_i$ is a literal in 3-CNF clause C_j. Due to Property (ii-B), $S_{i,k}$ must have two sensor nodes, and thus $x_i = true$ if $k = 1$, or $\bar{x}_i = true$ if $k = 2$. In either case, C_j is true.

So far, our problem is proven to be NP-hard. Since the problem is also in NP, it is NP-complete. □

6.5 Proposed Heuristic Algorithms

6.5.1 Routing-First Heuristic (RFH) Algorithms

6.5.1.1 Basic Ideas

The objective for co-designing the network deployment and the routing arrangement is to minimize the total energy cost for recharging the network for infinite network lifetime. As discussed in Section I, the total energy cost is affected by two factors: the amount of energy consumed by sensor nodes and the efficiency for recharging sensor nodes. The routing-first heuristic algorithms attempt to first minimize the amount of energy consumed by sensor nodes, which is achieved through finding the most-energy-efficient routing paths for the nodes at every post. Then, based on the found paths, a routing tree is constructed to facilitate every sensor node to send/forward their data to the base station. The routing tree should satisfy the dual conditions: Firstly, the tree contains only the edges of the most-energy-efficient routing paths, which keeps the minimum of energy consumption in sensor nodes. Secondly, the routing workload is concentrated to as few posts as possible, which is motivated by the idea that, letting these nodes consume the most energy and meanwhile deploying a large number of nodes to these posts to improve the efficiency for charging energy to these posts may collectively minimize the total energy cost. After the tree is constructed, the routing workload at each post can be computed, and then sensor nodes are deployed to all posts in the way that the number of nodes deployed to each post is proportional to the workload of the post. In the following, we first describe the basic version of the algorithm, which is followed by an advanced version which iteratively adjusts the routing arrangement and the deployment to reduce the total energy cost as much as possible.

6.5.1.2 The Basic Routing-First Algorithm

The basic Routing-First algorithm runs in the following four phases.

Phase I: Finding the minimum-energy paths from every post to the base station

This phase is conducted as follows:

- A graph $G = (V, E, w)$ is constructed, where V is the set of posts plus the base station. For any pair of nodes v_i and v_j in V, if the distance between them is less than the maximum transmission range (i.e., $dist(v_i, v_j) < d_{max}$), then there is an edge between v_i and v_j (i.e., $(v_i, v_j) \in E$). $w : E \longmapsto \mathbb{R}$ is the weight function for edges. For each edge (v_i, v_j), $w(v_i, v_j)$ is the amount of energy consumed for sending one bit between v_i and v_j, and as described in Chapter 6.3, $w(v_i, v_j)$ can be computed as $w(v_i, v_j) = \alpha + \beta \cdot d_x^{\gamma}$, where $x \in \{1, 2, \cdots, k\}$, and d_x is the smallest transmission range which is larger than the distance between v_i and v_j.

- For each post in V, the Dijkstra algorithm can be run to find the shortest path to the base station. Note that, with the above definition of edge weight, the found shortest path is actually the minimum energy path to the base station. The traditional Dijkstra algorithm returns only one shortest path. If multiple shortest paths exist, we need to find them out to enable the optimization in the next steps. Several methods can be applied to find all the shortest paths. For example, the Dijkstra algorithm can be modified such that it can record multiple shortest paths.

Phase II: Building the minimum-energy and workload-concentrated routing tree

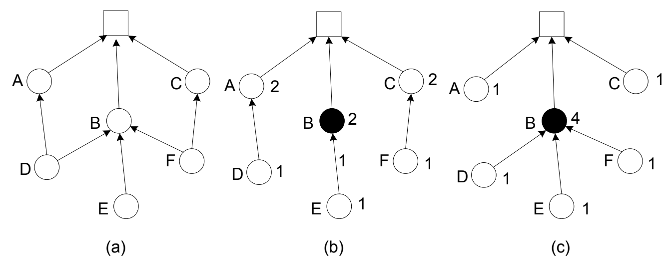


Figure 6.4 The benefit of concentrating routing workload. The square represents the base station, and the circles represent posts. The number to the right of each post is its routing workload. Each post uses e units of energy to send one bit information to its next hop post. The total number of sensor nodes is 7.

Phase I returns a number of minimum-energy paths for each post. We can form a shortest path "fat tree" by combining these paths of all the posts. Note that the final structure is not a tree but a "fat tree", since a node (post) may have multiple parents. We need to trim this fat tree into a tree. As discussed in the subsection of *Basic Ideas*, we adopt the heuristic of concentrating routing workload to a few number of nodes when trimming the tree. The example in Fig. 6.4 further explains why we adopt the heuristic. Here, Fig. 6.4 (a) shows a fat tree composed of shortest paths from every post to the base station. Fig. 6.4 (b) and (c) show two different routing tree structures that can be derived from the fat tree in (a): In Fig. 6.4 (b), routing workload is evenly distributed to three intermediate nodes, while in Fig. 6.4 (c), the workload is concentrated to node B. Suppose we have 7 sensor nodes to deploy to 6 posts. Obviously the extra one nodes should be deployed to one of posts A, B and C in Fig. 6.4 (b) and post B in Fig. 6.4 (c), since leaf posts have less routing workload. In Fig. 6.4 (b), the total energy cost for recharging this network (for every bit information reported by every post) is $3e+2\cdot 2e+2e/2 = 8e$, while the total energy cost is reduced to $5e + 4e/2 = 7e$ in Fig. 6.4 (c). We find that, in a larger scale network with limited number of sensor nodes to deploy, the benefit of routing workload concentration is even more significant. Specifically, the fat tree is trimmed as follows:

- *Step 1*. The routing workload at each post on the fat tree is computed. In RFH, the routing workload of a post on the fat tree is defined as *the number of descendants* of the post.

- *Step 2*. Posts are sorted based on the decreasing order of their routing workload. Then, the sorted posts are stored into a queue L based on the order; specifically, the post with the largest workload is at the head of the queue.

- *Step 3*. Let the current head element of queue L be post p. The following operations are conducted: For each descendant of p, denoted as d_p, its edge to any parent p' (where p' is not p's descendant or p) is deleted. This triggers p' and some of its upstream nodes to update their routing workload because reports from d_p may not pass through them. Consequently, their positions on the queue may have to be changed to maintain that all posts in L are stored in the decreasing order of their routing workload. After the operations are finished, post q is removed from the queue, and this step is repeated on the new head element if the queue is not empty.

After the above steps, a minimum-energy workload-concentrated routing tree is formed. Fig. 6.5 demonstrates a complete example to further illustrate the execution of Phase II.

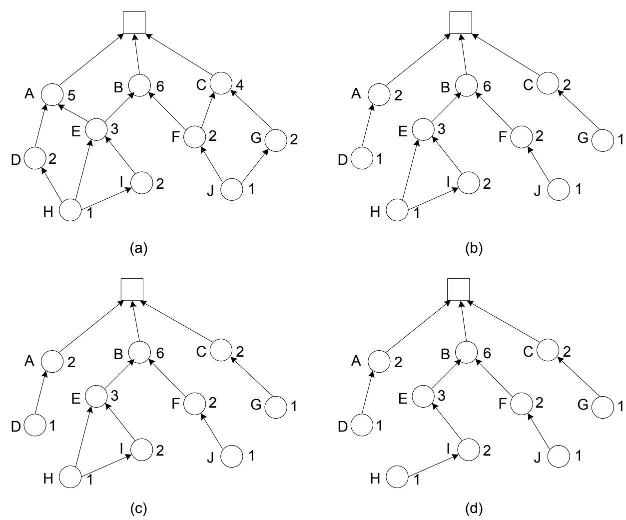


Figure 6.5 Trimming a fat tree into a minimum-energy workload-concentrated routing tree. The square represents the base station. The circles represent posts. The number to the right of a post is its routing workload (the number of its descendants) (a) is a fat tree of all shortest paths. In (b), the post with the highest routing workload (post B) is examined, and all the edges from its sub-tree to the other part of the tree, including (E, A),(F, C),(H, D),(J, G), are deleted, and the workload on affected posts is adjusted. In (c), post E is examined, and no edge is deleted. In (d), post I is examined, and edge (H, E) is deleted.

<u>*Phase III:*</u> Opportunistic merging of sibling posts

In the routing tree constructed so far, there may be multiple sibling posts that are close to each other and can reach each other using low transmission power but need to use high transmission power to reach their common parent. If this is the case, we can ask these sibling posts to send their data to one of them, and the latter is responsible for forwarding the data to their common parent post. This way, routing workload can be further concentrated. Concretely, this phase can be conducted as follows: for each post p in the tree, it is checked whether there are some of its children that can reach each other with smaller transmission range than what they need to reach itself. If there exists such children, they are organized into groups in which each member post can send its data to a designated post (the *head* of the group), and then the head forwards the all the data to p.

Phase IV: Workload-based deployment of sensor nodes

According to the routing tree constructed so far, sensor nodes can be deployed. The basic idea for deployment is, the number of nodes deployed to each post is proportional to the routing workload of that post. Assuming the workload is α_i for post i ($1 \leq i \leq N$), the problem for distributing M sensor nodes to N posts can be formulated as the following minimization problem:

$$\text{Minimize :}$$

$$\sum_{i=1}^{N} \alpha_i / m_i$$

$$\text{Subject to :}$$

$$\sum_{i=1}^{N} m_i = M$$

Where m_i is the number of sensor nodes to be deployed to post i.

Although the classical Lagrange multipliers method [59] can be run to find out m_i ($i = 1 \leq i \leq N$), the resulting m_i may not be integers. Hence, we address the problem in the following way:

- The Lagrange multipliers method is used to obtain first round of the values for m_i ($1 \leq i \leq N$). For the smallest m_j among $m_1, \cdots, m_N$, we round it to the nearest integer, which is the number of sensor nodes to be deployed to post j. Note that if the resulting number is 0, we set the number to 1 since every post should have at least one sensor node.

- Excluding post j and the number of sensor nodes that have been deployed to post j, the Lagrange multipliers method is reused to obtain another round of values for m_i ($i \in \{1, \cdots, N\}/\{j\}$). Similar to the previous step, the smallest m_k among all m_i is rounded to the nearest integer to get the actual number of sensor nodes deployed to post k. Then, this step is repeated until the deployments to all posts have been determined.

When heap data structure is utilized to maintain the list in Phase II, the time complexity of RFH is $O(n^2 \log n)$ which equals that of the most time consuming part of the algorithm, Phase II.

6.5.1.3 The Iterative Routing-First Algorithm

The basic version of the routing-tree first heuristic algorithm is composed of two macro-steps: a minimum-energy and workload-concentrated routing tree is first constructed, and then distributes sensor nodes based on this tree. The routing tree obtained from the first macro-step is of critical importance to the quality of final deployment and routing decisions. The tree is regarded as a *minimum-energy* tree based on the implicit assumption that every post has only one sensor node deployed, which however is not right. The iterative version of the routing-tree first heuristic algorithm is aimed to address this problem.

Our design of the iterative algorithm is motivated by the following observation. After one complete execution of the basic routing-tree first algorithm, the deployment of sensor nodes to posts is decided. From the deployment decision, we can find out the efficiency for charging every post. Taking this into account, we can now compute a more accurate *minimum-energy* tree, and then based on the tree to refine the deployment decision. This way, better routing and deployment decisions can be found. Furthermore, if the above steps are performed for multiple times, decisions can be continuously improved. The above idea is confirmed by the simulation results to be reported in Chapter 6.6: If we run our algorithm iteratively, the total energy cost for recharging the network decreases monotonically, and it converges at a certain value after a small number of iterations.

6.5.2 Incremental Deployment-Based (IDB) Heuristic Algorithm

The naive method to compute the exact optimal solution of the routing arrangement and deployment problem is as follows: For each of the possible ways to deploying M sensor nodes to N posts, a minimum-energy routing tree can be computed (as in Phase III of the DFH algorithm), and the total energy cost is recorded; then, the deployment strategy and the minimum-energy routing tree structure that result in the least total energy cost is the solution. However, the method incurs a runtime complexity of $O(\begin{pmatrix} M-1 \\ N-1 \end{pmatrix})$, which is not affordable when the system scale is large. To reduce the time complexity, we propose an incremental deployment heuristic as follows:

- Initially, each post is deployed with one sensor node.
- The rest $M - N$ sensor nodes are deployed in multiple rounds. In each round, we deploy δ number of sensors, and the total number of rounds is $\frac{M-N}{\delta}$ rounds, where δ is a system parameter.

In each round of the deployment, we examine each possible way to deploy the δ sensor nodes to posts. Thus, each round has a time complexity

$$O(\begin{pmatrix} N+\delta-1 \\ N-1 \end{pmatrix}).$$

Then, for each of the deployment strategies, the corresponding minimum-energy tree and the associated total energy cost are found. Note that, when computing the minimum-energy tree, all sensor nodes that have been deployed in previous rounds are assumed to exist in their deployment posts. After all possible ways have been examined, the one with the minimum-energy tree is chosen; i.e., δ sensor nodes are incrementally deployed to posts according to the chosen deployment strategy. After $\frac{M-N}{\delta}$ rounds of incremental deployment, we obtain the final strategy for deploying all M sensor nodes to N posts. The total time complexity for the algorithm is

$$O(\frac{M-N}{\delta}\begin{pmatrix} N+\delta-1 \\ N-1 \end{pmatrix}).$$

6.6 Performance Evaluation

Our performance evaluation has two objectives: (i) comparing the proposed heuristics with the optimal solution for small-scale networks; (ii) evaluating the proposed heuristic schemes in large scale networks under different system parameter settings to provide insights on choosing these parameters for network designers.

6.6.1 Simulation Setup

In the simulation, we assume the sensor network is deployed within a two-dimensional square field. The base station is located at its lower left corner. Posts are randomly selected within the field. The evaluation metric is the *total energy cost*, which is defined as the total energy disseminated by the wireless charger.

The following are the system parameters we used: In the equation regarding the energy consumption model (Eq. (6.1)), we set $\alpha = 50nJ/bit$, $\beta = 0.0013pJ/bit/m^4$, and $\gamma = 4$, as suggested in [60] . All the experiments choose three transmission ranges, i.e., $(d_1, d_2, d_3) = (25, 50, 75)$ meters except in the experiment studying the effect of number of transmission ranges, in which we used six transmission ranges, i.e., $(d_1, d_2, d_3, d_4, d_5, d_6) = (25, 50, 75, 100, 125, 150)$ meters.

6.6.2 Performance of Iterative RFH Algorithm

We first study the performance of iterative RFH algorithm under different iteration steps to determine the best iteration number. The deployment field is a $500m * 500m$ square, the number of posts is 100, and the number of sensor nodes varies in $\{400, 600, 800, 1000\}$. The results are the average of 20 simulations on different post distributions.

As shown in Fig. 6.6, the total energy cost decreases with more iterations, and it converges quickly after a small number of rounds. The figure shows that all the instances converage after 7 rounds either to a single value or to a very small narrow range. In some instances, the total energy cost does not converage at a single value, but oscillates among two or more values that are very close to each other. For instance, when the number of nodes is 600, the total energy cost for the RFH algorithm oscillates among $\{8.2592, 8.2581\}\mu J$ after the fifth round. We conjecture that the reason is, when we

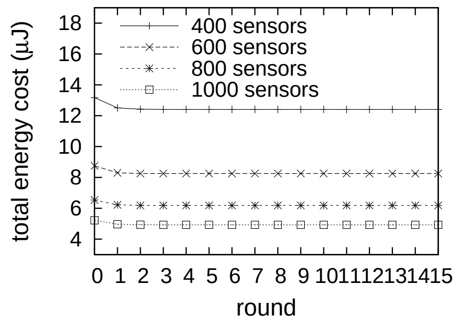


Figure 6.6   The benefit of running RFH iteratively

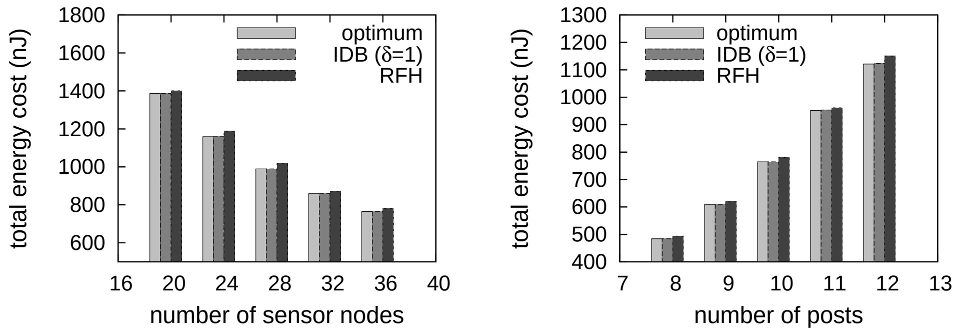
assign sensor nodes to posts, we round the values returned by the Lagrange multipliers method, and the rounding may have different effects in different rounds.

In the following sections, we always use the iterative RFH algorithm with seven iterations as a representative.

6.6.3 Comparing the Performance of Heuristic Algorithms with Optimal Solution

Due to the NP-hardness of the network deployment and routing arrangement problem, it is infeasible to compute the optimal solution for a large scale sensor network. Therefore, we only compute the optimal solution for small-size networks, and compare the optimal solutions with the results obtained from our proposed heuristic schemes under the same network settings. The comparison is to find out the difference between the optimal solutions and the solutions obtained by the heuristic algorithms. The results are the average of five simulations on different post distribution.

In this study, the network field is a $200m * 200m$ square. We conduct two experiments. Firstly, we fix the number of posts to 10, vary the number of nodes among $\{20, 24, 28, 32, 36\}$, and measure the total energy cost. As can be seen from Fig. 6.7(a), the total energy cost for all the algorithms decreases when there are more sensor nodes, since the energy recharging efficiency increases as more sensors are deployed to the same post. We can also see that, both heuristic algorithms achieve a performance close to the optimal solutions under these network settings. Between them, the IDB scheme with $\delta = 1$ has better performance. Specifically, the IDB algorithm delivers the same solutions as the optimal one for all the numbers of the sensor nodes in $\{20, 24, 28, 32, 36\}$. Furthermore, the total energy cost of the solutions found by RFH is up to 3% higher the optimal solutions.



(a) Impact of number of sensor nodes

(b) Impact of the number of posts

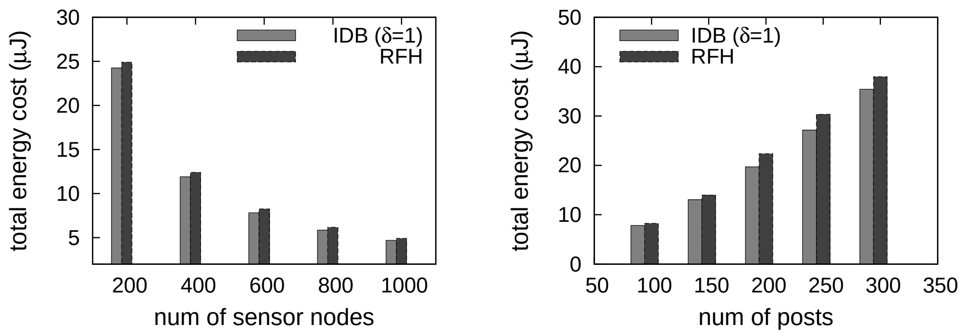
Figure 6.7 Comparison between the heuristics and the optimal solution

Secondly, we fix the number of nodes to 36, vary the number of posts among $\{8, 9, 10, 11, 12\}$, and measure the total energy cost of the solutions produced by different schemes. As shown in Fig. 6.7(b), the total energy cost decreases as the number of posts increases. This is because more data should be sent to the base station as the number of posts increases. Similar to the previous comparison in Fig. 6.7(a), we can see that the performance of heuristic algorithms are also close to that of the optimal solution. When the number of posts is 11 and 12, the total energy cost given by the optimum solution is slightly lower than that given by the RFH algorithm.

6.6.4 Performance of Heuristic Algorithms in Large-Scale Networks

In this section, we show the performance of our heuristic algorithms in large-scale networks. Assuming the sensor network is deployed to a $500m * 500m$ square field, we evaluate the impact of the number of sensors, the number of posts, and the number of transmission ranges on the performance of the heuristics. The results are the average of 20 simulations on different post distributions.

*Impact of number of sensor nodes.*We fix the number of posts at 100, and vary the number of nodes among $\{200, 400, 600, 800, 1000\}$. Fig. 6.8(a) shows that IDB leads with a margin over RFH, which indicates IDB is a better heuristic in terms of performance. For instance, when the number of posts is



(a) Impact of the number of sensor nodes

(b) Impact of the number of posts

Figure 6.8 Heuristic algorithms in large-scale networks

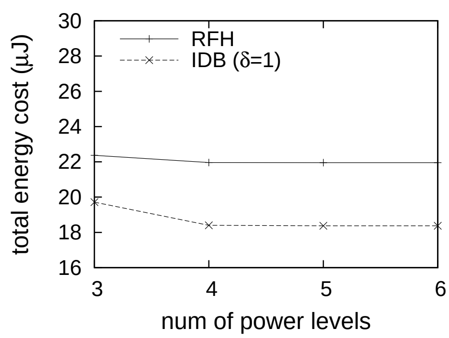


Figure 6.9 Impact of the number of power levels

1000, IDB with $\delta = 1$ computes a solution with total energy cost of $4.6914\ \mu J$, and RFH computes one with total energy cost of $4.9283\ \mu J$, i.e., 5% higher than IDB with $\delta = 1$. On the other hand, our simulation also indicates IDB runs much slower than RFH. Therefore, for large-scale networks, the RFH scheme may be a good choice considering its much shorter running time and a little bit worse performance.

Impact of number of posts: We fix the number of nodes at 600, and vary the number of posts among $\{100, 150, 200, 250, 300\}$. Fig. 6.8(b) shows a similar trend as in Fig. 6.8(a).

Impact of number of transmission ranges: We fix the number of nodes at 600, the number of posts at 200, and vary the number of transmissions among $\{3, 4, 5, 6\}$. When the number of transmission ranges is i, the set of transmission ranges is $\{25, 50, \cdots, 25 * i\}$ accordingly. Fig. 6.9 shows that, when more transmission ranges are available, the total energy cost almost keeps at the same value for IDB and RFH. The reason is that, under the constraint of keeping the network connected, shorter transmission ranges are preferable to larger ones since the power consumption increases much faster than transmission range does as shown by Eq. (6.1). As a result, larger transmission ranges do not have a significant impact on the heuristic algorithms.

CHAPTER 7. CONCLUDING REMARKS AND FUTURE WORK

7.1 Concluding Remarks

The energy scarcity problem is of paramount importance in sensor networks. Fully addressing this problem requires energy to be continually replenished into sensor nodes. In this dissertation, we investigated two approaches for energy replenishment of sensor networks: (i) *The Node Reclamation and Replacement Approach* and (ii) *The Wireless Recharging Approach*. We have proposed a number of schemes to tackle the energy scarcity problem and address different fundamental issues in realizing these approaches.

Firstly, for the node reclamation and replacement approach, we proposed a *node replacement and reclamation (NRR) strategy*, with which a mobile robot or human labor periodically traverses the sensor network, reclaims nodes with low or no power supply, replaces them with fully-charged ones, and brings the reclaimed nodes back to an energy station for recharging. To effectively and efficiently realize the NRR strategies under different application scenarios, we proposed a number of implementing schemes of the NRR strategy. (i) For the point sensing coverage model, we proposed an *adaptive rendezvous-based two-tier scheduling (ARTS) scheme*. (ii) For the area sensing coverage model, we proposed a *staircase-based scheme*. (iii) To address reliability issues in realizing the NRR strategy under the area coverage model, including sensor node failures and irregular energy consumption rate, we proposed three *reliable node reclamation and replacement schemes*, namely, *the staircase repairing scheme*, *the debit/credit scheme*, and *the energy consumption balancing scheme*.

Secondly, the wireless recharging approach takes advantage of emerging wireless recharging technology to continually transfer energy into the network. In this work, we focused on a unique problem with the wireless recharging technology; that is, how wireless recharging affects sensor network deployment and routing arrangement. We conducted field experiments using the cutting-edge wireless charging devices. The experiment results show that the wireless charging technology fundamentally changes the sensor network deployment and routing arrangement. We proved the problem of finding the optimal solutions on network deployment and routing arrangement is NP-complete. To address the problem efficiently and effectively, we also proposed a set of heuristic algorithms: the routing-first heuristic (RFH), the iterative version of RFH, and the incremental deployment-based heuristic (IDB).

To the best of our knowledge, our node reclamation and replacement works are among the pioneering ones to systematically study the node reclamation and replacement approach for energy replenishment of sensor networks; and our wireless recharging work is the first to study the impact of wireless charging technology on sensor network design.

Extensive simulations are conducted to evaluate all the proposed schemes, and the results show that the proposed schemes are effective and efficient.

7.2 Future Work

Future work can be conducted in the following directions:

- More detailed simulations can be conducted and a real testbed can be set up to evaluate the proposed NRR schemes. In the evaluation, the impact of packet collisions and communication delay on the performance of our proposed NRR schemes can be studied.

- Other realistic issues that may arise in specific NRR applications can be studied. These issues may include:

  (i) The cost for replacing different sensors may be different. For instance, in a coal mine monitoring environment, replacing sensors in a narrow and collapse-prone space is more dangerous than replacing ones in an open space.

  (ii) Sensors can be displaced due to environmental reasons, such as flood and collapse. In particular, in the staircase-based scheme, sensors may need to update their location information with the ES when they are relocated. Furthermore, coverage sets may need to be reformed.

(iii) The length of phase may be dynamically adjusted. If the phase length is fixed and too small, the total communication overhead would be high. While if the phase length is fixed and too large, a duty-cycle scheduling scheme may not be responsive to the dynamic environment. Therefore, dynamically adjusting phase length according to application requirement or other information is desirable.

(iv) The delay in processing ready and deadline messages and the travel time of the MR may need to be taken into account for deciding when ready and deadline messages should be sent out. Furthermore, the timing restrictions on performing replacement tasks may also be considered. For example, the MR may only be sent out during daylight times.

- Other protocols for sensor networks may be redesigned to fit in with our proposed energy replenishment schemes. With energy replenishment, the widely-believed load balancing philosophy does not apply, and thus the ways to schedule sensors' activities would be fundamentally different.

- The dynamic nature of sensor networks may be considered to improve wireless recharging efficiency. In this dissertation, we studied an off-line problem where the decisions on the deployment method and routing arrangement are made before the network deployment. How to adaptively adjust the distribution of sensors to posts and the routing arrangement when we need to deploy more sensors or when sensor failures occur may be studied.

BIBLIOGRAPHY

[1] C. Wan, S. Eisenman, A. Campbell, and J. Crowcroft, "Siphon: overload traffic management using multi-radio virtual sinks in sensor networks," in *Proc. of SenSys'05*, San Diego, CA, 2005, pp. 116–129.

[2] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[3] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *Proc. of INFOCOM '02*, New York, NY, 2002, pp. 1567–1576.

[4] T. Dam and K. Langendoen, "An adaptive energy-efficient mac protocol for wireless sensor networks," in *Proc. of SenSys '03*, Los Angeles, CA, 2003, pp. 171–180.

[5] M. Younis, M. Youssef, and K. Arisha, "Energy-aware management for cluster-based sensor networks," *Computer Networks*, vol. 43, no. 5, pp. 649–668, 2003.

[6] M. Younis, M. Youssef, and K. Arisha, "Energy-aware routing in cluster-based sensor networks," in *Proc. of MASCOTS '02*, Washington, DC, USA, 2002, pp. 129 – 136.

[7] Y. Zou and K. Chakrabarty, "Energy-aware target localization in wireless sensor networks," in *Proc. of IEEE PerCom '03*, Washington, DC, USA, 2003, pp. 60 – 69.

[8] S. Bhattacharya, H. Kim, S. Prabh, and T. Abdelzaher, "Energy-conserving data placement and asynchronous multicast in wireless sensor networks," in *Proc. of MobiSys '03*, San Francisco, CA, 2003, pp. 173–185.

[9] W. Liang, "Constructing minimum-energy broadcast trees in wireless ad hoc networks," in *Proc. of MobiHoc '02*, 2002, pp. 112–122.

[10] T. He, S. Krishnamurthy, J. A. Stankovic, T. F. Abdelzaher, L. Luo, R. Stoleru, T. Yan, L. Gu, J. Hui, and B. H. Krogh, “Energy-efficient surveillance system using wireless sensor networks,” in *Proc. of MobiSys ’04*, Boston, MA, 2004, pp. 270–283.

[11] Q. Li, J. Aslam, and D. Rus, “Online power-aware routing in wireless ad-hoc networks,” in *Proc. of MobiCom ’01*, 2001, pp. 97–107.

[12] K. Zeng, K. Ren, W. Lou, and P. J. Moran, “Energy-aware geographic routing in lossy wireless sensor networks with environmental energy supply,” in *Proc. of QShine ’06*, Waterloo, Ontario, Canada, 2006, pp. 8 – 15.

[13] W. Wang, V. Srinivasan, and K. Chua, “Using mobile relays to prolong the lifetime of wireless sensor networks,” in *Proc. of ACM MobiCom’05*, Cologne, Germany, 2005, pp. 270–283.

[14] J. Luo and J.-P. Hubaux, “Joint mobility and routing for lifetime elongation in wireless sensor networks,” in *Proc. of IEEE INFOCOM ’05*, Miami, FL, 2005, pp. 1735–1746.

[15] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. Srivastava, “Design considerations for solar energy harvesting wireless embedded systems,” in *Proc. of IPSN ’05*, Los Angeles, CA, 2005, pp. 457–462.

[16] A. Kansal, J. Hsu, M. B. Srivastava, and V. Raghunathan, “Harvesting aware power management for sensor networks,” in *Proc. of DAC ’06*, San Francisco, CA, 2006, pp. 651–656.

[17] A. Kansal and M. B. Srivastava, “An environmental energy harvesting framework for sensor networks,” in *Proc. of ISLPED ’03*, Seoul, Korea, 2003, pp. 481–486.

[18] A. Kansal, D. Potter, and M. B. Srivastava, “Performance aware tasking for environmentally powered sensor networks,” in *Proc. of ACM SIGMETRICS ’04*, New York, NY, 2004, pp. 223–234.

[19] M. Rahimi, H. Shah, G. S. Sukhatme, J. Heidemann, and D. Estrin, “Studying the feasibility of energy harvesting in a mobile sensor network,” in *Proc. of ICRA ’03*, Taipei, Taiwan, 2003, pp. 19 – 24.

[20] P. Corke, S. Hrabar, R. Peterson, D. Rus, S. Saripalli, and G. Sukhatme, "Autonomous deployment and repair of a sensor network using an unmanned aerial vehicle," in *Proc. of ICRA '04*, New Orleans, LA, 2004, pp. 1143–1151.

[21] L. Filipe, M. Augusto, L. Ruiz, A. Alfredo, D. Ceclio, and A. Fernandes, "Efficient incremental sensor network deployment algorithm," in *Proc. of Brazilian Symposium on Computer Networks '04*, Gramado/RS, Brazil, 2004, pp. 3 – 14.

[22] PowerCast, "Powercast," *http://www.powercastco.com/*, 2008.

[23] R. C. Shah, S. Roy, S. Jain, and W. Brunette, "Data mules: modeling and analysis of a three-tier architecture for sparse sensor networks," *Ad Hoc Networks*, vol. 1, no. 2-3, pp. 215–233, 2003.

[24] A. A. Somasundara, A. Ramamoorthy, and M. B. Srivastava, "Mobile element scheduling for efficient data collection in wireless sensor networks with dynamic deadlines," in *Proc. of RTSS '04*, Lisbon, Portugal, 2004, pp. 296–305.

[25] Y. Gu, D. Bozdag, E. Ekici, F. Ozguner, and C. Lee, "Partitioning based mobile element scheduling in wireless sensor networks," in *Proc. of IEEE SECON '05*, Santa Clara, CA, 2005, pp. 386–395.

[26] D. Jea, A. A. Somasundara, and M. B. Srivastava, "Multiple controlled mobile elements (data mules) for data collection in sensor networks," in *Proc. of DCOSS '05*, Marina del Rey, CA, 2005, pp. 244–257.

[27] G. Yang, B.Tong, D. Qiao, and W. Zhang, "Sensor-aided overlay deployment and relocation for vast-scale sensor networks," in *Proc. of IEEE INFOCOM '08 Mini-conference*, 2008, pp. 2216 – 2224.

[28] B. Tong, S. Panchapakesan, and W. Zhang, "A three-tier framework for intruder information sharing in sensor networks," in *Proc. of IEEE SECON '08*, San Francisco, CA, 2008, pp. 451–459.

[29] B. Tong, G. Wang, W. Zhang, and C. Wang, "Node reclamation and replacement for long-lived sensor networks," in *Proc. of IEEE SECON '09*, Rome, Italy, 2009, pp. 1 – 9.

[30] B. Tong, Z. Li, G. Wang, and W. Zhang, "On-demand node reclamation and replacement for guaranteed area coverage in long-lived sensor networks," in *Proc. of QShine '09 (accepted)*, Las Palmas de Gran Canaria, Spain, 2009.

[31] B. Tong, Z. Li, G. Wang, and W. Zhang, "Towards reliable scheduling schemes for long-lived replaceable sensor networks," in *conference submission*, 2009.

[32] B. Tong, Z. Li, G. Wang, and W. Zhang, "How wireless power charging technology affects sensor network deployment and routing," in *conference submission*, 2009.

[33] C. Alippi, G. Anastasi, C. Galperti, F. Mancini, and M. Roveri, "Adaptive sampling for energy conservation in wireless sensor networks for snow monitoring applications," in *Proc. of IEEE MASS '07*, Pisa, Italy, 2007, pp. 1 – 6.

[34] S. Olariu and I. Stojmenovic, "Design guidelines for maximizing lifetime and avoiding energy holes in sensor networks with uniform distribution and uniform reporting," in *Proc. of INFOCOM '06*, Barcelona, Spain, 2006, pp. 1 – 12.

[35] R. Madan, S. Cui, S. Lall, and A. J. Goldsmith, "Cross-layer design for lifetime maximization in interference-limited wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 5, no. 11, pp. 3142–3152, 2006.

[36] C. M. Vigorito, D. Ganesan, and A. G. Barto, "Adaptive control of duty cycling in energy-harvesting wireless sensor networks," in *Proc. of SECON '07*, San Diego, CA, 2007, pp. 21–30.

[37] C. Alippi and C. Galperti, "An adaptive system for optimal solar energy harvesting in wireless sensor network nodes," *IEEE-Transactions on Circuits and Systems I*, vol. 55, no. 6, pp. 1742 – 1750, 2008.

[38] Y. Mei, C. Xian, S. Das, Y. C. Hu, and Y.-H. Lu, "Sensor replacement using mobile robots," *Comput. Commun.*, vol. 30, no. 13, pp. 2615–2626, 2007.

[39] W. Hu, V. Tran, N. Bulusu, C. Chou, S. Jha, and A. Taylor, "The design and evaluation of a hybrid sensor network for cane-toad monitoring," in *Proc. of International Symposium on Information Porcessing in Sensor Networks '05*, Los Angeles, CA, 2005, pp. 503–508.

[40] G. Wang, G. Cao, and T. La Porta, "A bidding protocol for sensor deployment," in *Proc. of IEEE ICNP'03*, Atlanta, GA, 2003, pp. 315–324.

[41] G. Wang, G. Cao, and T. La Porta, "Movement-assisted sensor deployment," in *Proc. of IEEE INFOCOM '04*, Hong Kong, China, March 2004, pp. 2469–2479.

[42] J. Wu and S. Yang, "Smart: A scan-based movement assisted sensor deployment method in wireless sensor networks," in *Proc. of IEEE INFOCOM '05*, Miami, FL, March 2005, pp. 2323–2324.

[43] G. Wang, G. Cao, T. L. Porta, and W. Zhang, "Sensor relocation in mobile sensor networks," in *Proc. of IEEE INFOCOM '05*, Miami, FL, 2005, pp. 2302–2312.

[44] E. Ekici, Y. Gu, and D. Bozdag, "Mobility-based communication in wireless sensor networks," *IEEE Communications Magazine*, vol. 44, pp. 56–62, 2006.

[45] M. Desrochers, J. Lenstra, M. Savelsbergh, and F. Soumis, "Vehicle routing with time windows: optimization and approximation," in *Vehicle Routing: Methods and Studies*, Amsterdam, Holland, 1988, pp. 65–84.

[46] Auren, "The vrp web," *http://neo.lcc.uma.es/radi-aeb/WebVRP/*, 2007.

[47] N. Bansal, A. Blum, S. Chawla, and A. Meyerson, "Approximation algorithms for deadline-tsp and vehicle routing with time-windows," in *Proc. of STOC '04*, Chicago, IL, USA, 2004, pp. 166–174.

[48] K. C. Tan, L. H. Lee, and K. Q. Zhu, "Heuristic methods for vehicle routing problem with time windows," in *Proceedings of the 6th AI and Math*, Fort Lauderdale, FL, 2000, pp. 281–295.

[49] H. Lau, M. Sim, and K. Teo, "Vehicle routing problem with time windows and a limited number of vehicles," *European Journal of Operational Research*, vol. 148, no. 3, pp. 559–569, 2003.

[50] J. Xiao, H. Dulimarta, N. Xi, and R. Tummala, "Controller design for an autonomous wall climbing micro-robot based on ti320lf2407 dsp chip," *DSPS Fest*, 2000.

[51] J. K. Lenstra and A. H. G. Rinnooy Kan, "Complexity of vehicle routing and scheduling problems," *Networks*, vol. 11, no. 2, pp. 221–227, 1981.

[52] T. Liebling, D. Naddef, and L. A. Wolsey, "On the capacitated vehicle routing problem," *Mathematical Programming*, vol. 94, no. 2-3, pp. 343–359, 2003.

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

[54] H. Zhang and J. C. Hou, "Maintaining sensing coverage and connectivity in large sensor networks," *Wireless Ad Hoc and Sensor Networks*, vol. 1, no. 1-2, pp. 89–123, 2005.

[55] X. Bai, S. Kuma, D. Xuan, Z. Yun, and T. H. Lai, "Deploying wireless sensors to achieve both coverage and connectivity," in *Proc. of MobiHoc '06*, Florence, Italy, 2006, pp. 131–142.

[56] D. Estrin, R. Govindan, J. S. Heidemann, and S. Kumar, "Next century challenges: Scalable coordination in sensor networks," in *Proc. of MobiCom '99*, Seattle, WA, 1999, pp. 263–270.

[57] D. Mascarenas, M. Todd, G. Park, and C. Farrar, "Development of an impedance-based wireless sensor node for structural health monitoring," *Smart Materials and Structures*, vol. 16, no. 6, pp. 2137 – 2145, 2007.

[58] H. Schaap and M. Bijl, "Wireless sensor networks - key to unlocking the smart factory," *Suframa Minapim News*, Nov. 2007.

[59] G. Arfken, *Mathematical Methods for Physicists*, pp. 945–950, Academic Press, 3rd edition, 1985.

[60] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.