Quality-Aware Scheduling Algorithms in Renewable Sensor Networks

Xiaojiang Ren

A thesis submitted for the degree of

Doctor of Philosophy

The Australian National University

November 2015

Declaration

This thesis is a presentation of the original work except where otherwise started. I completed this work jointly with my supervisor, Associate Professor Weifa Liang. My contribution to the work is around 85%.

Xiaojiang Ren

Acknowledgments

This thesis is the result of three and a half years of work whereby I have been accompanied and supported by many people. It is a pleasure to express my gratitude to all of them.

First and foremost, I would like to thank my supervisor, Associate Professor Weifa Liang, for his constant support and encouragement over the past few years. He guided me to become a qualified researcher through expert supervision to my Ph.D. work with his perceptive insight and valuable experience. I have been impressed by his rigorous academic attitude, which is something I have striven and will strive to emulate in my own research. Now, as I am almost at the end of the Ph.D. program, I feel extremely lucky that I have met him and worked under his guidance. He has patiently nurtured me to develop intellectually and professionally.

Similarly, I am greatly indebted to other members of my supervisory panel, Professor Brendan Mckay and Professor Qian Zhang, for their brilliant ideas, professional guidance, and fruitful discussions. Without their support, this thesis would not be possible.

I also would like to thank the people in the Research School of Computer Science at The Australian National University, for their generous help and assistance in various aspects. Professor Alistair Rendell, Associate Professor Henry Gardner, Janette Garland, Di Wellach-Smith, Jadon Radcliffe, and Shaw Tyi Tan deserve to be especially appreciated.

My gratitude also extends to my friends, Grace Xu, Zichuan Xu, Qiufen Xia, Wenzheng Xu, Mojtaba Rezvani, and Elly Liang for their great friendship and kind help during my years at ANU.

Last but not least, I would like to express my special thankfulness to my family for their patience and constant support throughout my entire Ph.D. study.

Publications

Journal

[1] Weifa Liang, Wenzheng Xu, Xiaojiang Ren, and Xiaohua Jia.
Maintaining rechargeable sensor networks perpetually via multiple mobile charging vehicles. Submitted to *ACM Transactions on Sensor Networks*, under revision, 2015.
[2] Xiaojiang Ren, Weifa Liang, and Wenzheng Xu.
Data collection maximization in renewable sensor networks via time-slot scheduling. *IEEE Transactions on Computers*, Vol. 64, pp. 1870–1883, 2015.
[3] Xiaojiang Ren, Weifa Liang, and Wenzheng Xu.
Quality-Aware Target Coverage in Energy Harvesting Sensor Networks. *IEEE Transactions on Emerging Topics in Computing*, Vol. 3, pp. 8–21, 2015.
[4] Weifa Liang, Xiaojiang Ren, Xiaohua Jia, and Xu Xu.
Monitoring quality maximization through fair rate allocation in harvesting sensor networks. *IEEE Transactions on Parallel and Distributed System*, Vol. 24, pp. 1827–1840, 2013.

Conference

[1] Xiaojiang Ren, Weifa Liang, and Wenzheng Xu.
Maximizing charging throughput in rechargeable sensor networks. *In Proceedings of 23rd International Conference on Computer Communications and Networks (ICCCN)*, IEEE, 2014.
[2] Xiaojiang Ren and Weifa Liang.
Data quality maximization via mobile sinks in renewable sensor networks. *In Proceedings of 25th Annual International Sympo on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, IEEE, 2014.

[3] Wenzheng Xu, Weifa Liang, Xiaojiang Ren, and Xiaona Lin. On-demand energy replenishment for sensor networks via wireless energy transfer. *In Proceedings of 25th Annual International Sympo on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, IEEE, 2014.

[4] Weifa Liang, Wenzheng Xu, Xiaojiang Ren, Xiaohua Jia, and Xiaona Lin. Maintaining sensor networks perpetually via wireless recharging mobile vehicles. *In Proceedings of 39th Annual IEEE Conference on Local Computer Networks (LCN)*, IEEE, 2014.

[5] Wenzheng Xu, Weifa Liang, Xiaona Lin, Guoqiang Mao, and Xiaojiang Ren. Towards perpetual sensor networks via deploying multiple mobile wireless chargers. *In Proceedings of 43rd International Conference on Parallel Processing (ICPP)*, IEEE, 2014.

[6] Xiaojiang Ren, Weifa Liang, and Wenzheng Xu. Use of a mobile sink for maximizing data collection in energy harvesting sensor networks. *In Proceedings of 42nd International Conference on Parallel Processing (ICPP)*, IEEE, 2013.

[7] Xiaojiang Ren and Weifa Liang. The use of a mobile sink for quality data collection in energy harvesting sensor networks. *In Proceedings of 2013 IEEE Wireless Communications and Networking Conference (WCNC)*, IEEE, 2013.

[8] Xiaojiang Ren and Weifa Liang. Delay-tolerant data gathering in energy harvesting sensor networks with a mobile sink. *In Proceedings of 2012 IEEE Global Communications Conference (GLOBECOM)*, IEEE, 2012.

Abstract

Wireless sensor network has emerged as a key technology for various applications such as environmental sensing, structural health monitoring, and area surveillance. Energy is by far one of the most critical design hurdles that hinders the deployment of wireless sensor networks. The lifetime of traditional battery-powered sensor networks is limited by the capacities of batteries. Even many energy conservation schemes were proposed to address this constraint, the network lifetime is still inherently restrained, as the consumed energy cannot be replenished easily. Fully addressing this issue requires energy to be replenished quite often in sensor networks (*renewable sensor networks*). One viable solution to energy shortages is enabling each sensor to harvest renewable energy from its surroundings such as solar energy, wind energy, and so on. In comparison with their conventional counterparts, the network lifetime in renewable sensor networks is no longer a main issue, since sensors can be recharged repeatedly. This results in a research focus shift from the network lifetime maximization in traditional sensor networks to the network performance optimization (e.g., monitoring quality). This thesis focuses on these issues and tackles important problems in renewable sensor networks as follows.

We first study the target coverage optimization in renewable sensor networks via sensor duty cycle scheduling, where a renewable sensor network consisting of a set of heterogeneous sensors and a stationary base station need to be scheduled to monitor a set of targets in a monitoring area (e.g., some critical facilities) for a specified period, by transmitting their sensing data to the base station through multihop relays in a real-time manner. We formulate a coverage maximization problem in a renewable sensor network which is to schedule sensor activities such that the monitoring quality is maximized, subject to that the communication network induced by the activated sensors and the base station at each time moment is connected. We approach the problem for a given monitoring period by adopting a general strategy. That is, we divide the entire monitoring period into equal numbers of time slots

and perform sensor activation or inactivation scheduling in the beginning of each time slot. As the problem is NP-hard, we devise efficient offline centralized and distributed algorithms for it, provided that the amount of harvested energy of each sensor for a given monitoring period can be predicted accurately. Otherwise, we propose an online adaptive framework to handle energy prediction fluctuation for this monitoring period. We conduct extensive experiments, and the experimental results show that the proposed solutions are very promising.

We then investigate the data collection optimization in renewable sensor networks by exploiting sink mobility, where a mobile sink travels around the sensing field to collect data from sensors through one-hop transmission. With one-hop transmission, each sensor could send data directly to the mobile sink without any relay, and thus no energy are consumed on forwarding packets for others which is more energy efficient in comparison with multi-hop relays. Moreover, one-hop transmission particularly is very useful for a disconnected network, which may be due to the error-prone nature of wireless communication or the physical limit (e.g., some sensors are physically isolated), while multi-hop transmission is not applicable. In particular, we investigate two different kinds of mobile sinks, and formulate optimization problems under different scenarios, for which both centralized and distributed solutions are proposed accordingly. We study the performance of the proposed solutions and validate their effectiveness in improving the data quality.

Since the energy harvested often varies over time, we also consider the scenario of renewable sensor networks by utilizing wireless energy transfer technology, where a mobile charging vehicle periodically travels inside the sensing field and charges sensors without any plugs or wires. Specifically, we propose a novel charging paradigm and formulate an optimization problem with an objective of maximizing the number of sensors charged per tour. We devise an offline approximation algorithm which runs in quasi-polynomial time and develop efficient online sensor charging algorithms, by considering the dynamic behaviors of sensors' various sensing and transmission activities. To study the efficiency of the proposed algorithms, we conduct extensive experiments and the experimental results demonstrate that the proposed algorithms are very efficient.

We finally conclude our work and discuss potential research topics which derive from the studies of this thesis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Wireless sensor network (WSN) has emerged as a key technology for many monitoring and surveillance applications. For example, sensor networks can be used to monitor building integrity during earthquakes; sensor networks can be deployed for habitat monitoring and environmental sensing; sensor networks can be applied to monitor temperature or power usage of data centers; sensor networks can be used to support control, communications, surveillance and targeting system functions; sensor networks can be deployed to monitor patients and assist patients with disabilities; sensor networks can be used to monitor the pesticides level in the drinking water, and the level of soil erosion. Some other commercial applications include managing inventory, monitoring product quality, and monitoring disaster areas [1].

However, the lack of easy access to a continuous power source and the limited lifetime of batteries have hindered the wide-scale deployment of sensor networks. Conventional WSNs are typically required to run for a long periods, often several years, only powered by batteries. Finite battery capacity means that sensor nodes operate for a finite duration, which implies limited lifetime of the WSN applications or additional cost and complexity to regularly change batteries. Indeed, batteries cannot easily be replaced and sometimes are very dangerous to replace, since typically there are hundreds to thousands of sensor nodes and sensor nodes may be deployed in unreachable places (e.g., sensor networks used to monitor a nuclear disaster site or volcanic eruption). To make matters worse, depleted batteries constitute environmental problems. A viable solution to this problem is to allow sensor nodes to harvest energy from their surroundings. In addition to being environmentally friendly, harvesting energy could also enable sensor nodes to function indefinitely, allowing the network to operate perpetually and eliminating the cost for battery replacement.

The main sources of energy suitable for sensor networks can be broadly classified into the following two categories, (i) Ambient Energy Sources: sources of energy from the surrounding environment, e.g., solar energy, wind energy, and vibration energy; and (ii) Human Power Sources: sources of energy harvested from body movements of humans, e.g., blood pressure, body heat and breath [85]. Several implementations of renewable sensor nodes exists, e.g., Prometheus, HydroWatch, Heliomote, Ambimax, and Sunflower, which have been discussed in [85]. Moreover, a few example applications have been deployed and tested in the real environment. ZebraNet [115] is a mobile sensor platform with sparse network coverage and high-energy GPS sensors to track zebra movement. The ZebraNet node has a Li-ion rechargeable battery for support at night and bad weather. TurtleNet [93] is similar to the ZebraNet and extends on ZebraNet's design for perpetual wildlife tracking. Trio testbed [25] is an outdoor sensor network deployment that consists of 557 solar-powered motes, seven gateway nodes and a root server. SHiMmer [72] is a wireless sensor platform for structural health monitoring, in which the nodes is powered by solar energy and uses super-capacitor as storage. RiverMote [32] is also a wireless sensor platform for environmental monitoring and consists of low-power motes with energy harvesting system.

Energy harvesting introduces a change to the fundamental principles based on which protocols for wireless sensor networks are designed. Instead of focusing on energy efficient protocols that aim to maximize network lifetime, the main design objective in sensor networks with energy harvesting (*renewable sensor networks*) is to maximize the performance of the network, given that energy is available to be harvested from the surroundings. In other words, the surplus of harvested energy can be used to improve the performance of the network. For example, a sensor node can increase its sampling frequency or its duty-cycle to increase sensing reliability, or increase transmission power to decrease length of routing paths.

1.1 System Model of Renewable Sensor Networks

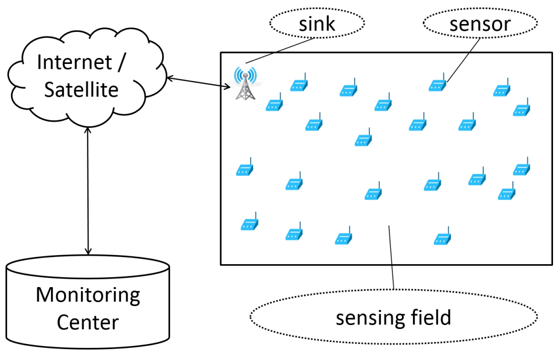


Figure 1.1: A renewable sensor network.

Similar to a conventional sensor network, a renewable sensor network consists of hundreds/thousands of renewable sensor nodes. The positions of sensor nodes need not be engineered or predetermined. This allows the random deployment in inaccessible terrains or disaster relief operations. Figure 1.1 gives an example of a renewable sensor network. The sensor nodes are usually scattered in a *sensing field*. Each of these scattered sensor nodes has the capability to collect data and route data back to the base station (*sink*). The sink may communicate with the *monitoring center* via Internet or Satellite.

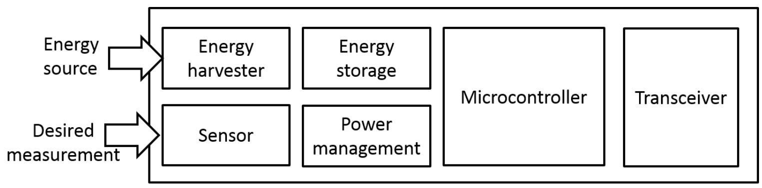


Figure 1.2: An illustration of renewable sensor node components.

Illustrated in Figure 1.2, renewable sensor nodes consist of sensing, data processing, communicating and energy harvesting components, which are able to monitor a wide variety of ambient conditions that include temperature, humidity, vehicular movement, lightning condition, pressure, noise levels, and the current characteristics such as speed, direction, and size of an object. Once the energy is collected by a sensor node, it will be stored either in NiMH batteries, Lithium batteries, ultra-capacitor, or even directly supply this sensor node.

1.2 Challenges in Research of Renewable Sensor Networks

The harvested energy however depends heavily on environmental conditions, and its time-varying, intermittent availability poses significant challenges in algorithm/protocol design for renewable sensor networks. For example, for sensor nodes with energy harvesting abilities, conservative energy expenditures may lead to missed recharging opportunities due to battery capacity limitations, while aggressive usage of energy may cause battery outages that result in reduced coverage or connectivity of the network for certain time periods. Further complications come about, since the heterogeneous spatial harvesting capabilities across different nodes in a sensing space. Take solar power for instance, the harvesting capabilities of different sensors vary, due to under the shade or cloud coverage.

Particularly, the challenges lie in:

- The added dimension of harvested energy makes the energy management in renewable sensor networks substantially different from their conventional sensor networks. Sensor node's internal power system should be realistically modelled (e.g., renewable energy replenishment and battery recharging/discharing process) to provide reliable energy awareness;
- The harvested energy variations among sensor nodes require that their duty-cycles must be carefully allocated, as biased allocations lead to poor representations of sensory data from low energy sensor nodes, thereby compromising the monitoring quality. The capacity of self-management for each individual node is required to ensure sustainable operation while optimizing its long-term power usage;
- The intermittent connectivity between sensor nodes requires that data routing protocols must be robust and intelligently adapted to the changes of network topology. Distributed and adaptive algorithms and protocols should be devised, that enable sensor nodes to change their activities intelligently to respond

to the dynamic changes;

- The large scale deployment of renewable sensor networks requires that the proposed algorithms and protocols must be scalable with the growth of the network size. Moreover, there is no need to find an exact solution as finding an exact solution usually takes a much longer time due to the uncertainty of the convergence speed of the solution. Even if such a solution is found, it may no longer be applicable, since the profiles of energy harvesting sources have been dramatically changed during the solution finding period.

1.3 Research Topics in Renewable Sensor Networks

Essential to our vision is a focus on the design of algorithms/protocols for renewable sensor networks that can optimize resource utilization and network performance. Renewable sensor networks open the door to an unattended, uninterrupted, and virtually unlimited information-gathering paradigm for a wide-range of applications, such as field exploration, environmental monitoring, and security surveillance.

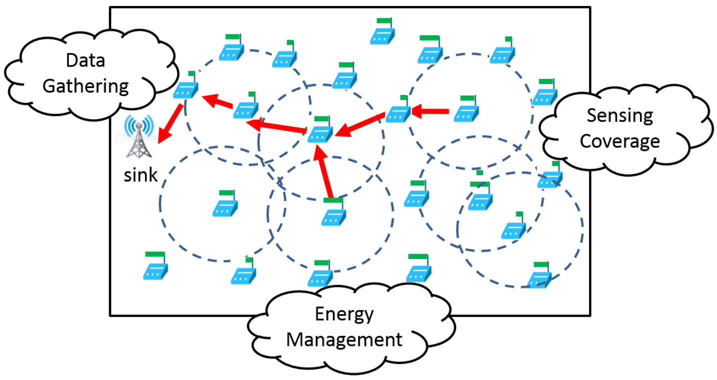


Figure 1.3: An overview.

As illustrated in Figure 1.3, to provide quality-aware service with a renewable sensor network, several important constraints should be addressed such as sensing coverage, data gathering, and energy management. To this end, this thesis specifically focuses on the following research topics:

- Quality-aware target coverage in renewable sensor networks via sensor duty cycle scheduling
- Quality-aware data collection in renewable sensor networks by exploring sink mobility
- Quality-aware energy provisioning in renewable sensor networks by utilizing wireless energy transfer

1.3.1 Quality-Aware Target Coverage in Renewable Sensor Networks via Sensor Duty Cycle Scheduling

Coverage in wireless sensor networks is usually defined as a measure of how well and how long the sensors are able to observe the physical space, which indicates the monitoring quality provided by wireless sensor networks. For example, in an application of forest monitoring, one may ask how well the network can observe a given area and what the chances are that a fire starting in a specific location of forest will be detected in a given time frame. Additionally, coverage formulations can try to find weak points in a sensor field and suggest future deployment or reconfiguration schemes for improving the coverage performance. In addition to coverage, it is important for a sensor network to maintain connectivity. Connectivity can be defined as the ability of the sensor nodes to reach the sink. If there is no available route from a sensor node to the data sink then the data collected by that node can not be processed.

Sensing coverage in conventional sensor networks has been extensively studied in the past decade [14, 18, 58, 97]. Depending on the coverage objectives and applications, they can be roughly classified into three categories: (1) area coverage [97], in which the objective is to cover (monitor) a region (the collection of all space points within the sensor field), and each point of the region need to be monitored (e.g., cover a forest); (2) target coverage [14, 58], in which the objective is to cover a set of target points with known location that need to be monitored (e.g., cover a collection of precious renaissance paintings); (3) barrier coverage [18], in which the objective is to minimize the probability of undetected penetration through the region, (e.g., detect the spread of lethal chemicals around a chemical factory). Among these problems, most of the research focused on the network lifetime prolongation. To maximize the network lifetime, various strategies of sensor activity scheduling have been proposed. Among them, the popular one is the adoption of duty-cycles, that is, each sensor works either in one of two modes: active mode or sleep mode [12, 24, 36, 37, 58, 60, 64, 99].

In comparison with conventional sensor networks, we focus on the target coverage problem in renewable sensor networks. Several studies on target coverage have been conducted with the aim of optimizing the coverage performance [22, 76, 86, 110]. These mentioned studies however did not consider the connectivity of the communication network induced by the activated sensors and the base station. It is well known that both sensing coverage and network connectivity are the fundamental performance metrics in wireless sensor networks, where the coverage quantifies the quality of monitoring while the network connectivity indicates the accessibility from the base station to the sensory data. Unlike existing works, we study how to schedule sensor nodes to maximize coverage for a given monitoring period by adopting a general strategy. That is, we start by dividing the entire monitoring period into number of equal time slots. We then perform sensor activation or inactivation scheduling in the beginning of each time slot. The challenges to solve the problem are as follows: (1) at which time slots, a sensor should be activated or deactivated, as the amount of harvested energy (consumed energy) at a sensor depends on not only different scheduling strategies but also the availabilities of time-varying energy harvesting sources in the entire monitoring period? (2) how to make sure that all activated sensors and the base station form a connected component at each time slot? (3) how to devise an efficient sensor scheduling algorithm whose solution will maximize the target coverage quality for the entire monitoring period?

1.3.2 Quality-Aware Data Collection in Renewable Sensor Networks by Exploring Sink Mobility

Besides the active (via in-situ observation) or passive (via remote-sensing technologies) sensing on the interested real-word phenomena, the paramount task in a wireless sensor network is data collection, which considers how to efficiently gather sensing data from scattered sensor nodes.

Traditionally, data collection in sensor networks assumed that the network is dense, the data produced by a sensor node is sent to a fixed sink via a short-range multi-hop routing path. Although the data collection paradigm based on fixed sinks may be applicable to small to medium size networks, it is definitely not suitable for large-scale networks, due to limited communication bandwidth, etc. Meanwhile, the multi-hop communication makes the sensor nodes near to the sink deplete their energy much faster than the other nodes, which causes high unbalance of energy consumption in the whole network and results in the premature termination of the network lifetime.

To overcome the mentioned drawbacks, several approaches have been proposed for efficient data collection. Based on the focus of the work, we can roughly divide them into three categories. The first category is the enhanced relay routing [19, 80, 88], in which load balance, schedule pattern and data redundancy are jointly considered. The second category introduces a hierarchical infrastructure to improve the scalability [31, 40, 59], in which sensor nodes are organized into clusters and cluster heads take the responsibility of forwarding data to the sink. Clustering can be very effective in local data aggregation since it can dampen collisions and support load balance among sensor nodes. The third category explores the sink mobility and adopts mobile sinks [23, 69], in which one or more mobile sinks move around the sensor nodes deployed over the area of interest. Once a mobile sink is within the communication range of a sensor node, the data accumulated in the sensor node is forwarded to the mobile sink. Clearly, such a mobile sink based strategy alleviates the problem of the multi-hop communication scheme since no node needs to be involved in energy-exhaustive multi-hop message forwarding. However, in enhanced relay routing schemes, some critical sensor nodes on the path may still run out of energy faster than others. In cluster-based schemes, cluster heads will inevitably consume much more energy than other sensor nodes due to the handling of intra-cluster aggregation and inter-cluster data forwarding. In contrast, using mobile sinks can effectively alleviate the non-uniform energy consumption by confining packet relays, and take the burden of data routing away from sensors.

In the past few years, extensive studies on mobile data collection in conventional sensor networks have been conducted and demonstrated that such mobile sink based strategy can significantly improve various network performance including reducing the energy consumption of sensors, balancing the workload among the sensors, and prolonging the network lifetimes [9, 16, 23, 28, 29, 52, 56, 92, 107, 109]. In general, existing studies can be classified into three categories in terms of sink mobility: sinks with random mobility which are often mounted on human being or animals which move randomly in the monitored area to collect data from sensors [41]; sinks with controlled mobility which actively control their trajectories [35, 39]; sinks with deterministic mobility (*path-constrained mobile sinks*) which move along a pre-defined path [16, 29, 45, 84]. Most existing studies focused on minimizing the energy consumption so as to prolong the network lifetime since sensors are powered by energy-limited batteries.

In contrast to conventional sensor networks, very little attention has been paid to data collection in renewable sensor networks with mobile sinks. Most existing studies on data collection in such networks are to adjust sensors' duty-cycle or sampling rate and forward data collected to one or more static sinks through multi-hop relays [26, 55, 61, 63], that is to throttle activity during times of limited energy and increase activity when energy is readily available. Orthogonal to existing works, we consider data collection in a renewable sensor network with a mobile sink. The time-varying characteristics of energy renewable sources poses a great challenge in the design of routing protocols for renewable sensor networks. That is, how to design a routing protocol such that the volume or quality of the collected data is maximized, under the dynamic energy replenishment constraint.

1.3.3 Quality-Aware Energy Provisioning in Renewable Sensor Networks by Utilizing Wireless Energy Transfer

Considering that energy harvesting in renewable sensor networks is not stable and often varies over time, the recent breakthrough in wireless energy transfer technology provides a promising alternative or supplementary solution to power sensors. Particularly, employing two strongly coupled magnetic resonant objects, Kurs *et al.* [49] exploited the resonant magnetic technique to transfer energy from one storage device to another without any plugs or wires. They empirically demonstrated that a wireless illumination of a 60 watts light bulb from 2 meters away achieved a 40% energy transfer efficiency. What makes such wireless energy transfer technology particularly attractive is that it does not require line-of-sight or any alignment (i.e., omnidirectional). This promising technique will provide a controllable and perpetual energy source to recharge sensors if needed.

Armed with the wireless energy transfer technology, several studies on employing mobile vehicles with high volume batteries as mobile chargers to recharge energy for sensors have been conducted [4, 21, 27, 38, 51, 75, 83, 95, 101, 116, 118]. Most of these studies considered sensor energy recharging and data flow routing jointly. These joint consideration of energy replenishment and data flow routing in literature may have limited applications, due to their unrealistic assumptions such as (1) the energy consumption rate and/or data generation rate do not change over time; (2) the flow conservation at each sensor node is maintained; and (3) reliable wireless communications among the sensor nodes are always assumed. However, in reality sensing data rates of sensors are usually closely related to specific applications of the sensor network (e.g. event detection application). The flow conservation prevents data aggregation at intermediate nodes while data aggregation at sensor nodes can not only reduce data traffics but also bring node energy savings [48]. Also, it is well-known that wireless communication is notoriously unreliable [117], and retransmissions at some nodes at some unexpected time intervals may lead to substantial energy consumption of the nodes.

Orthogonal to these studies, we consider a heterogenous sensor network in which sensors have significant variations in sampling and energy consumptions. A typical example is that a sensor network deployed for ecological study consists of sensors of different modalities including humidity, temperature, video, etc. The sensing rates of different sensors vary, depending on their physical phenomena. Under this setting, we investigate an on-demand wireless sensor charging paradigm. That is, sensors send their recharging requests to the base station according to their residual energy status. A wireless mobile charger then will be dispatched to start a charging tour to recharge these requested sensors. We study how to schedule the mobile charger to maximize the number of sensors charged (*charging throughput*) per tour. The challenges to tackle this problem are: (1) which sensors are to be included in this tour? (2) what is the charging order of the sensors in this tour?

1.4 Thesis Contributions

The main contribution of this thesis is to systematically study the use of renewable sensor networks for sustainable monitoring, including proposing new concepts, formulating non-trivial optimization problems, developing novel approaches to solve them, and evaluating the solutions through extensive experiments. The proposed techniques schedule sensors' duty cycle according to their energy status to optimize the coverage quality, route data efficiently through utilizing mobile sinks to maximize the quality of the collected data, and find a close tour for mobile charger to maximize the throughput of the mobile charger. Specifically, the thesis contributions are listed as follows.

- Energy prediction is fundamental to the algorithm design for renewable sensor networks. Existing prediction approaches are investigated and validated in Chapter 2, using real solar data profiles obtained from The National Solar Radiation Data Base [7] in the States, which contains the most comprehensive collection of solar data and is freely available.

- The coverage maximization problem in a renewable sensor network is considered in Chapter 2, where a renewable sensor network is deployed for monitoring a set of targets for a given monitoring period. A new coverage quality metric is proposed to measure the monitoring quality within two different time scales: one is within each time slot, in which the monitoring quality of a target is modelled by a sub-modular function of the number of sensors covering it; another is within the entire monitoring period, the monitoring quality of a target is measured by a sub-modular function of the number of time slots it is covered. As the problem is NP-hard, efficient centralized and distributed algorithms are devised, provided that the amount of harvested energy of each sensor for a given monitoring period can be predicted accurately. Otherwise, an adaptive framework is proposed to handle energy prediction fluctuation for the monitoring period.

- Data collection in a renewable sensor network with a mobile sink is explored in Chapters 3 and 4, where two different kinds of sink mobility are considered. For sink with controlled mobility, the problem is to find an optimal trajectory for the mobile sink that consists of sojourn locations in a given location space and the exact sojourn time at each sojourn location, assuming that the mobile sink can only collect data from one-hop sensors. Both centralized heuristic and its distributed implementation are provided. For sink with deterministic mobility, the problem is to allocate time slots to individual sensors under their energy replenishment rate constraints. An offline algorithm with a provable approximation ratio, and a fast, scalable online distributed algorithm are devised when only considering maximizing the quantity of the collected data.

- The charging throughput maximization problem in a renewable sensor network with a mobile charger is studied in Chapter 5, where a mobile charger is employed to replenish energy to sensor nodes in renewable sensor networks by utilizing wireless energy transfer technology. Considering sensors in renewable sensor networks have significant variations in the sampling needs and energy consumptions, an on-demand wireless sensor charging paradigm is advocated. That is, sensors send their recharging requests to the base station according to their residual energy status, and the base station then dispatches the wireless mobile charger to start a charging tour and recharge these requested sensors. For the charging throughput maximization problem, an offline approximation algorithm is proposed, which runs in quasi-polynomial time by reducing the formulated optimization problem to the orienteering problem with time windows. The delivered solution is proven to be fractional of the optimum. Two online heuristics are devised where future charging request knowledge is not available.

- For all proposed algorithms, extensive experiments by simulation are conducted. The impacts of constraint parameters on the algorithm performance are also studied, and the effectiveness of the proposed algorithms is validated in various aspects. The performance of the proposed algorithms is compared with that of comparable existing approaches, and their superiority is demonstrated.

1.5 Thesis Overview

The remainder of the thesis is organized as follows. Chapter 2 investigates the prediction models of harvested energy, and formulates the coverage maximization problem, for which an adaptive framework and heuristics are developed accordingly. Chapters 3 and 4 focus on mobile data collection in renewable sensor networks using two different types of mobile sinks, and proposes solutions for each of them separately. Chapter 5 studies an on-demand energy replenishment in renewable sensor networks by employing a wireless mobile charger, formulates a charging throughput maximization problem, and devises both offline and online algorithms. Chapter 6 summarizes the thesis and proposes future work.

Chapter 2

Target Coverage Maximization in Renewable Sensor Networks

2.1 Introduction

Sensing coverage is a fundamental problem in wireless sensor networks for event detection, environment monitoring and surveillance purposes. In conventional sensor networks, there is a tradeoff between network lifetime and sensor coverage. To achieve a better coverage, more sensors have to be active at the same time, then more energy would be consumed and the network lifetime is reduced. On the other hand, if more sensors are put into sleep to extend the network lifetime, the coverage will be adversely affected. Most studies focused on the network lifetime prolongation. To maximize the network lifetime, various strategies of sensor activity scheduling have been proposed. Among them, a popular one is the adoption of duty-cycles, that is, each sensor works either in active or sleep mode [13, 24, 37, 58, 60, 64, 99].

In comparison with conventional sensor networks, network lifetime in renewable sensor networks is no longer a main issue since sensors can be recharged repeatedly by renewable energy sources. This results in the research focus shift from the network lifetime maximization to scheduling sensor activities to keep them survival through accurate energy harvesting predictions. Several studies on target coverage have been conducted with the aim of optimizing the coverage performance [22, 76, 86, 110]. These mentioned studies however did not consider the connectivity of the communication network induced by the activated sensors and the sink. It is well known that both sensing coverage and network connectivity are the fundamental performance metrics for wireless sensor networks, where the coverage quantifies the quality of monitoring while the network connectivity indicates the accessibility from the sink to sensory data. Orthogonal to the existing work, we study how to schedule sensor activities such that the coverage quality is maximized, subject to that the communication network induced by the activated sensors and the sink at each time moment is connected.

In this chapter, we consider the coverage maximization problem in a renewable sensor network, which can be stated as follows. Given a set of targets (e.g., some critical facilities) in a monitoring region, a sensor network that consists of a set of heterogeneous sensors powered by renewable energy and a sink is deployed to monitor the set of targets for a specified period, where sensors transmit their sensing data to the sink in a real-time manner. The problem is to activate sensors such that the target coverage quality is maximized, subject to that (i) the amount of energy consumed by each sensor is no more than that it has been charged during this monitoring period; and (ii) the communication network induced by the active sensors and the sink at each time point is connected. One such an application scenario is a renewable sensor network deployed for forest fire monitoring. Unlike most existing studies on conventional sensor networks that the energy of each sensor decreases monotonically over time, the energy consumption at each sensor in such networks can be well managed. In contrast, the energy harvesting rate of each sensor in renewable sensor networks varies over time, and the energy of each sensor can be replenished if needed. However, the energy consumption at each sensor must be carefully managed. On one hand, if there is enough amount of harvested energy available in the near future, we must fully make use of the harvested energy for maximizing target coverage; otherwise, the conservative use of the harvested energy may miss the next recharging opportunity. On the other hand, if the energy charging chances of a sensor in the near future is predictably small, its energy should not be used carelessly despite that the sensor may still have plenty of energy. Otherwise, the sensor will expire very soon, and its coverage quality will severely decrease. In summary, time-varying characteristics of renewable energy sources in renewable sensor networks makes sensor activity scheduling become very difficult, not to mention ensuring that all activated sensors and the sink must be connected.

We approach the coverage maximization problem for a given monitoring period by adopting a general strategy. That is, we start by dividing the entire monitoring period into L equal numbers of time slots. We then perform sensor activation or inactivation scheduling in the beginning of each time slot. The challenges to solve the problem are as follows: (1) at which time slots among the L time slots, a sensor should be activated or deactivated, as the amount of harvested energy as well as its consumed energy at a sensor depends on not only different scheduling strategies but also the availabilities of time-varying energy harvesting sources in the entire monitoring period? (2) how to make sure that all activated sensors and the sink form a connected component at each time slot? (3) how to devise an efficient sensor scheduling algorithm whose solution will guarantee that the target coverage quality for the entire monitoring period is maximized?

The novelty of our work lies in two aspects. We are the first to introduce a new coverage metric to accurately measure the target coverage quality. This new metric enables to model the coverage quality of each target by two different time scales: One is within each time slot, in which the coverage quality of the target is modeled by a sub-modular function of the number of sensors covering it, which implies that the margin gain of the coverage quality of the target decreases with the number of sensors it is covered in the time slot. Another is within the entire monitoring period, the coverage quality of a target is measured by the number of time slots it is covered, this metric is also modeled by a sub-modular function that may be different from the one within each time slot, which implies that the more the number of time slots the target is covered, the higher the coverage quality of the target will be. The overall coverage quality of a target for the entire monitoring period then is a weighted linear combination of these two sub-modular functions. Not only do we introduce this new coverage quality metric, but also do we devise novel centralized and distributed algorithms for the coverage maximization problem in a renewable sensor network. Also, we propose an adaptive framework for the problem under both network connectivity and harvesting energy prediction fluctuation constraints. The main contributions of this chapter are as follows. We first consider quality-aware target coverage in a renewable sensor network by introducing a new coverage metric that can measure the coverage quality accurately, and formulating a novel coverage maximization problem that takes both sensing coverage quality and network connectivity into consideration. As the problem is NP-hard, we then devise efficient centralized and distributed algorithms for it, provided that the amount of harvested energy of each sensor for a given monitoring period can be accurately predicted. Otherwise, we propose an adaptive framework to handle energy prediction fluctuations during the monitoring period. We finally conduct extensive experiments by simulations to evaluate the performance of the proposed algorithms. Experimental results show that the solutions delivered by the proposed algorithms are very promising.

The rest of the chapter is organized as follows. Section 2.2 surveys related works. Section 2.3 introduces basic models, defines the coverage maximization problem, and shows its NP-hardness. A centralized heuristic algorithm and its distributed implementation are given in Section 2.4. An adaptive framework dealing with energy prediction fluctuation is proposed in Section 2.5. Section 2.6 presents the simulation results, and Section 2.7 concludes the chapter.

2.2 Related Work

Sensing coverage problems in conventional sensor networks have been extensively investigated in the past [3, 13, 15, 37, 64, 94]. One efficient method is to partition sensors in a sensor network into multiple subsets (*sensor covers*) such that the sensors in each subset can cover all targets. Thus, only one sensor cover at each time slot is activated for a fractional of the entire monitoring period and only the sensors in the active sensor cover are in active mode, while the others are in sleep mode to save their energy [13]. In terms of connected coverage problem, Gupta *et al.* [37] proposed the minimum connected sensor cover problem to find a minimum number of sensors to achieve a full coverage while the communication graph induced by the sensors is connected. They presented a greedy algorithm with a guaranteed performance ratio, assuming that each sensor can adjust its transmission range dynamically. Wu *et al.* [99] recently presented an improved approximation algorithm for it. Liu and

Liang [64] studied the connected coverage problem with a given coverage guarantee. They introduced the partial coverage concept, and presented a centralized heuristic algorithm which takes both partial coverage and sensor connectivity into account simultaneously. They also considered the full coverage and sensor connectivity by partitioning the lifetime of a sensor into several equal intervals and finding a collection of connected sensor covers such that the network lifetime is maximized [65]. Ammari and Das [3] addressed the k-coverage problem that within each scheduling round, every location in a monitoring field is covered by at least k active sensors while keeping all active sensors connected. They proposed several heuristic algorithms for the problem.

Compared with the studies on sensing coverage in conventional sensor networks, few attentions have been paid to the sensing coverage problem in renewable sensor networks. Tang *et al.* [86] studied the problem and proposed an approximation algorithm with an approximation ratio 1/2, by assuming that the coverage quality is characterized by a sub-modular function and the communication graph induced by the active sensors and the sink may be disconnected. They [87] also extended their work by proposing distributed sensing schedule algorithms with provable convergence and performance bound by fixing the duty cycle of each sensor. Dai *et al.* [22] considered a similar problem for stochastic event capture by formulating a coverage optimization problem and presenting an approximation algorithm with an approximation ratio 1/2. Yang and Chin [110] considered the problem of maximizing the network lifetime while ensuring all targets are continuously monitored by at least one sensor. They proposed a linear programming solution to determine the activation schedule of sensors, where one subset of sensors is active while the rest of sensors keep in sleep modes to conserve energy. However, none of these mentioned works takes into consideration of the connectivity of active sensors and the sink. Consequently, the sensing data generated by active sensors may not be relayed to the sink immediately. However, many critical real-time applications do need the sensed data to be collected in a real-time manner. Considering that the transmission energy consumption of each sensor in most real applications is the dominant one among its energy consumptions in sensing, computation and communications, its sensing data must be relayed to the sink through multiple relays to reduce its energy consumption. The connectivity among active sensors and the sink thus is necessitated to ensure such real-time data transfer. This connectivity requirement thus poses great challenges in the design of approximation algorithms for the problem. That is why none of approximation algorithms for the problem under the connectivity constraint with an optimization objective expressed by a sub-modular function has ever been developed.

2.3 Coverage Maximization Problem

We consider a renewable sensor network $G = (V \cup \{s\}, E)$ consisting of $|V|$ heterogeneous stationary sensors and a sink s, which is deployed to monitor m targets $O = \{o_1, o_2, \cdots, o_m\}$ in a 2D region of interest. Each sensor $v \in V$ is powered by renewable energy source such as solar energy, and has a fixed transmission and sensing ranges. There is an undirected edge in E between two sensors or a sensor and the sink if they are within the transmission range of each other. For each sensor $v \in V$, let C_v be the set of targets within its sensing range. For each target $o \in O$, let S_o be the set of maximum number of active sensors covering it.

2.3.1 Energy Budget Model

Following a widely adopted renewable energy replenishment assumption [62], we assume that the energy replenishment rate of each sensor is much slower than its energy consumption rate, and the amount of energy harvested by the sensor in a future time period is uncontrollable but predictable, based on its source type and its historic energy harvesting profile. Assume that time is divided into equal time slots [57]. Let L be the number of time slots after which the next recharging pattern will be repeated, where *a recharging pattern* of solar energy depends on the weather conditions accordingly (e.g., 24 hours on default). Assume that the L time slots are indexed by $1, 2, \cdots, L$. To estimate the amount of energy harvested of each sensor at a recharging pattern, several prediction algorithms are available [11, 44]. Kansal *et al.* [44] proposed the very first algorithm, referred to as the Exponentially

Weighted Moving-Average (EWMA), which applies weighting factors to previously harvested sampling energy values that are constantly decreasing. At the same time, the prediction takes into account every single harvesting energy sample with different relevance. Let $\overline{Q}(t)$ be the prediction of the amount of harvested energy of sensor $v_i \in V$ at time slot t with $1 \leq t \leq L$. The value of $\overline{Q}(t)$ is calculated as follows.

$$\overline{Q}(t) = w \cdot \overline{Q}(t') + (1 - w) \cdot Q(t'), \tag{2.1}$$

where w is a given weight with $0 < w < 1$, t' is the tth time slot in the previous recharging pattern, and $Q(t')$ is the actual amount of energy harvested at time slot t'. A similar prediction strategy has also been adopted by Noh *et al* [74], referred to as the Variance Exponentially Weighted Moving-Average (VEWMA), in which the final prediction on the amount of harvested energy $\hat{Q}(t)$ is calculated by adjusting the base prediction with the current environmental conditions (e.g., a sunny day or a cloudy day), as follows.

$$\hat{Q}(t) = \overline{Q}(t) \cdot \frac{Q(t-1)}{\overline{Q}(t-1)} \tag{2.2}$$

With the knowledge of its harvesting energy prediction, the energy budget $P(v_i)$ of sensor $v_i \in V$ in the next L time slots is defined as

$$P(v_i) = \min\{B(v_i), RE(v_i) + \sum_{t=1}^{L} \overline{Q}(t)\}, \tag{2.3}$$

where $B(v_i)$ and $RE(v_i)$ are the battery capacity and the residual energy of sensor v_i in the beginning of the previous recharging pattern, $1 \leq i \leq |V|$.

2.3.2 Energy Consumption Model

Recall that each sensor $v_i \in V$ at each time slot operates in either active or sleep (or inactive) mode. Let e_i^{active} and e_i^{sleep} be the energy consumptions of sensor v_i in active and sleep modes at each time slot, respectively. Assume that $e_i^{sleep} \ll e_i^{active}$ and the energy consumption of sensor v_i in sleep mode is negligible. The sink will determine the schedule of sensors in the beginning of every L time slots, according to the energy budget of each sensor. By the energy neutral operation theory [44], to support continuous monitoring services, sensors should not consume more energy than that they harvested at any period. The activation of a sensor thus is constrained by the actual amount of energy it harvested. Let $b_i = \lfloor \frac{P(v_i)}{e_i^{active}} \rfloor$ be *the time slot budget* of sensor $v_i \in V$ for a monitoring period of L time slots. Then, sensor v_i cannot be activated more than b_i time slots for a monitoring period of L time slots, where $P(v_i)$ is the energy budget of sensor v_i.

2.3.3 Coverage Quality

In each time slot, a different subset of sensors will be activated, which leads to a different subset of targets to be covered. Also, the more time slots in which a target is covered, the higher the coverage quality of the target will be. To measure the coverage quality of targets, we here consider the target coverage quality within two different time scales, which is illustrated by a simple motivation example in Fig. 2.1, where sensors v_1, v_2, and v_3 are deployed to monitor targets o_1, o_2, and o_3 for a monitoring period of 6 time slots. Assuming that the time slot budgets of sensors v_1, v_2, and v_3 are 2, 4, and 3, respectively. There are two different solutions A and B for sensor activation in a given monitoring period. Targets in solution A are covered by more sensors in each time slot but for less time slots, e.g., target o_1 is covered by both sensors v_1 and v_3 in time slots 1 and 2, but it is only covered by 3 time slots among the monitoring period of 6 time slots. Targets in solution B are covered by more time slots but by less sensors in each time slot, e.g., target o_1 is covered by 4 time slots, but it is only covered by a single sensor at time slots 1, 3, and 4, respectively. From these two different solutions, it can be seen that the coverage quality of each target o is determined by not only the number of time slots it is covered but also the number of sensors it is covered within each time slot.

In the following we first adopt a utility metric similar to the one in [58], where the coverage quality of a target is measured by the number of time slots in which the target is covered. Specifically, for each target $o \in O$ at each time slot t with $1 \leq t \leq L$, let $N_1(o,t) = \{t\}$, which is a set containing the index of time slot t if target o is covered by an active sensor in time slot t; $N_1(o,t) = \emptyset$ otherwise. Let N_c^o

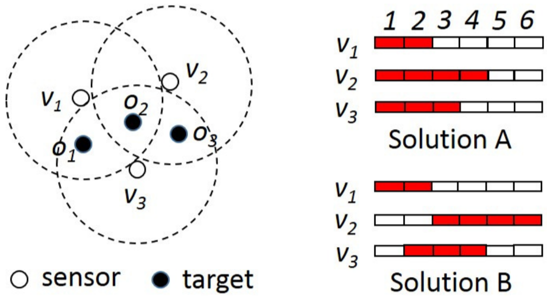


Figure 2.1: A simple motivation example for measuring the coverage quality.

be the set of time slots in which target o is covered, then $N_c^o = \cup_{t=1}^{L} N_1(o,t)$. Clearly, N_c^o is a subset of the set of all time slots $\{1, 2, \ldots, L\}$. Let $U_1(o) = f_1(N_c^o)$ represent the coverage quality of target o, by counting the number of time slots the target being covered during a monitoring period of L time slots, where f_1 is a sub-modular function whose definition is as follows. $f_1 : 2^A \mapsto R^{\geq 0}$ satisfies the following three properties:

$$(1)\ \ f_1(\emptyset) = 0; \tag{2.4}$$

$$(2)\ \ f_1(A_1) \leq f_1(A_2) \quad \text{where } A_1 \subseteq A_2 \subseteq A \tag{2.5}$$

$$\text{and } A \text{ is a finite ground set;} \tag{2.6}$$

$$(3)\ \ f_1(A_1 \cup \{a\}) - f_1(A_1) \geq f_1(A_2 \cup \{a\}) - f_1(A_2) \tag{2.7}$$

$$\text{where } A_1 \subseteq A_2 \subseteq A \text{ and } \exists a \in A \setminus A_1 \cup A_2. \tag{2.8}$$

The rationale behind the adoption of the sub-modular function f_1 (sometimes it is also referred to as a utility function) is that f_1 is a monotonic increasing function, whose marginal utility decreases with the increase of the number of time slots. In other words, for each target $o \in O$, the more time slots it is covered, the higher coverage quality it will have. However, with the further increase on the number of time slots it is covered, the net gain of its coverage quality becomes diminishing.

The use of coverage metric $U_1(\cdot)$ to measure the target coverage quality however is biased. Under this metric, for a given target, it cannot be distinguished whether the target is covered by only a single sensor or by multiple sensors at a given time slot. For example, in event detection applications, the more amount of the sensors an event is detected by, the higher probability the event can be discovered [111, 112]. To capture the coverage quality of each target both in each time slot and for the entire monitoring period, we then introduce a new coverage quality metric within two different time scales that takes into account not only the number of sensors covering a target at each given time slot but also the number of time slots the target is covered for the monitoring period of L time slots, through two non-decreasing sub-modular functions $f_1(\cdot)$ and $f_2(\cdot)$, respectively. Specifically, for each target $o \in O$ at each time slot t, let $U_2(o,t) = f_2(S_o^t)$ represent the coverage quality of target o at time slot t, where $S_o^t \subseteq S_o$ is the set of active sensors covering target o at time slot t. The coverage quality of target o for L consecutive time slots thus is

$$U(o) = \alpha \cdot U_1(o) + (1-\alpha) \cdot \sum_{t=1}^{L} U_2(o,t), \tag{2.9}$$

where α is a given utility weight with $0 \le \alpha \le 1$. When $\alpha = 0$, this means we only consider the coverage quality caused by the number of sensors covering target o, while $\alpha = 1$ means we only consider the coverage quality by the number of time slots target o being covered during the entire monitoring period. Hence, the overall coverage quality achieved for the L time slots is $\sum_{o \in O} U(o)$.

2.3.4 Problem Statement

Given a renewable sensor network $G = (V \cup \{s\}, E)$ deployed for monitoring a set of targets O for a period of L consecutive time slots, and the time slot energy budget b_i of each sensor $v_i \in V$, *the coverage maximization problem* in G is to activate a subset of sensors V_t ($V_t \subseteq V$) at each time slot t with $1 \le t \le L$ such that the overall coverage quality for the monitoring period $\sum_{o \in O} U(o)$ is maximized, where

$$\sum_{o \in O} U(o) = \alpha \sum_{o \in O} U_1(o) + (1-\alpha) \sum_{o \in O} \sum_{t=1}^{L} U_2(o,t) \tag{2.10}$$

$$= \alpha \sum_{o \in O} f_1(\cup_{t=1}^{L} N_1(o,t)) + (1-\alpha) \sum_{o \in O} \sum_{t=1}^{L} f_2(S_o^t), \tag{2.11}$$

$$N_1(o,t) = \begin{cases} \emptyset & \text{if } \nexists v \in V_t \text{ s.t. } o \in C_v \\ \{t\} & \text{if } \exists v \in V_t \text{ s.t. } o \in C_v, \end{cases} \tag{2.12}$$

and

$$S_o^t = \begin{cases} \emptyset & \text{if no sensor node in } V_k \text{ covers target } o \\ \{v \mid v \in V_t,\ o \in C_v\} & \text{otherwise,} \end{cases} \tag{2.13}$$

subject to the following two constraints:

1. the induced communication subgraph by activated sensors in V_t and the sink is connected, i.e., $G(V_t \cup \{s\})$ is a connected graph for each time slot t with $1 \le t \le L$. Thus, the sensing data of activated sensors in V_t can be relayed to the sink in real time.

2. For each sensor $v_i \in V$, the number of time slots in which it is activated is no more than its time slot budget b_i so that none of the sensors will run out of its budgeted energy, i.e., $\sum_{t=1}^{L} I(V_t, v_i) \le b_i$, where $I(V_t, v_i)$ is an indicator function, which is defined as $I(V_t, v_i) = 1$ if $v_i \in V_t$ and $I(V_t, v_i) = 0$ otherwise.

2.3.5 NP-Hardness

The coverage maximization problem defined is NP-hard. It is easy to verify that the dynamic activation schedule problem in [86] is a special case of the problem, where each sensor can communicate with the sink directly, and the utility weight α is 1. Even for this special case, it has been shown to be NP-hard, which implies the NP-hardness of the coverage maximization problem.

2.4 Offline Heuristics

Due to the NP-hardness of the coverage maximization problem, we here propose a greedy heuristic for it, assuming that the energy budget of each sensor for a monitoring period of L time slots is given in advance. In general, for each time slot t with $1 \leq t \leq L$, we assume that there is a corresponding tree rooted at the sink consisting of all activated sensors at time slot t. Initially, there is a forest consisting of L trees with each tree containing only the tree root - the sink. Recall that b_i is the time slot budget of sensor $v_i \in V$ in the beginning of a monitoring period of L time slots. Then, sensor v_i can join no more than b_i trees in the forest; otherwise, its energy budget is not enough to support its operation.

The construction of the forest proceeds iteratively. Within each iteration, a sensor node is added to one of the L trees in the forest if it results in the maximum utility gain in terms of the coverage quality by eq. 2.10. This procedure continues until either no more sensors can be added to the trees, or no more utility gain on the coverage quality can be achieved. Note that none of the sensor nodes is added to a single tree twice.

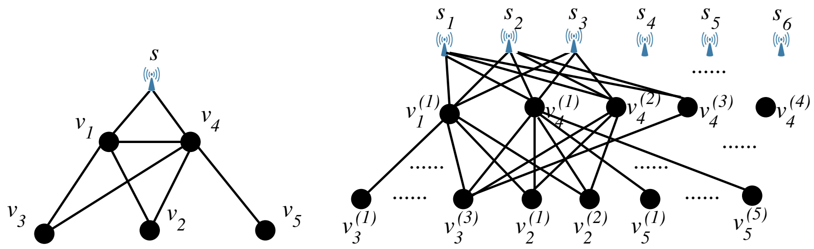
2.4.1 Centralized Algorithm

Given the time slot budget $b_i \geq 0$ of sensor $v_i \in V$ for all i with $1 \leq i \leq |V|$, we first construct an auxiliary graph $G' = (V' \cup \{s_1, s_2, \cdots s_L\}, E')$ from the renewable sensor network $G = (V, E)$ as follows.

For the sink s, there are L corresponding copies $s_1, s_2, \cdots, s_L$ in G' with each being the root of a tree T_j, $1 \leq j \leq L$. These L trees form a forest. For each sensor $v_i \in V$, there are b_i corresponding node copies $v_i^{(1)}, v_i^{(2)}, \cdots, v_i^{(b_i)}$ in V' with each corresponding an activation of sensor v_i in one of up to b_i time slots, assuming that $b_i \ll L$. For each edge $(v_i, s) \in E$ that corresponds that the sink and sensor v_i are within the transmission range of each other, there are $b_i \times L$ corresponding edge copies $(v_i^{(1)}, s_1), \cdots, (v_i^{(b_i)}, s_1), \cdots, (v_i^{(1)}, s_L), \cdots, (v_i^{(b_i)}, s_L)$ in E'. For each edge $(v_i, v_j) \in E$ that corresponds that sensors v_i and v_j are within the transmission range of each other, there are $b_i \times b_j$ corresponding edge copies $(v_i^{(1)}, v_j^{(1)})$, $\cdots$, $(v_i^{(b_i)}, v_j^{(1)})$,

$\cdots, (v_i^{(1)}, v_j^{(b_j)}), \cdots, (v_i^{(b_i)}, v_j^{(b_j)})$ in E'.

Fig. 2.2(b) is an illustrative construction of graph G' of the original renewable sensor network $G = (V \cup \{s\}, E)$ in Fig. 2.2(a), where the time slots are indexed by $1, 2, \cdots, L$ with $L = 6$ and the sensor set $V = \{v_1, v_2, v_3, v_4, v_5\}$. Let $b_i = i$ be the time slot energy budget of each sensor $v_i \in V$ for a given monitoring period of L time slots, $1 \leq i \leq 5$.



(a) $G = (V \cup \{s\}, E)$ (b) $G' = (V' \cup \{s_1, s_2, \cdots, s_L\}, E')$

Figure 2.2: An example: $L = 6$ and a renewable sensor network $G = (V \cup \{s\}, E)$ with the set of sensors $V = \{v_1, v_2, v_3, v_4, v_5\}$ and $b_i = i$ for all i with $1 \leq i \leq 5$.

The forest consists of L trees $T_1, T_2, \cdots, T_L$, which is constructed as follows. Initially, each tree T_j contains only the root node s_j, $1 \leq j \leq L$. We add the other copies of sensor nodes in V' to the trees iteratively. Within each iteration, a node is added to the forest if it leads to the maximum utility gain of the coverage quality. Specifically, for each node $v_i^k \in V'$ with $1 \leq k \leq b_i$, let $v_i \in V$ be its corresponding sensor and $V(v_i^k) = \{v_i^{(1)}, v_i^{(2)}, \cdots, v_i^{(b_i)}\}$ the set of copies of v_i in G'. Recall that C_{v_i} is the set of targets within the sensing range of v_i. We set $C(v_i^k) = C_{v_i}$ for each node v_i^k, which is the set of targets covered by node v_i^k. For each tree T_j rooted at node s_j, let $V(T_j) \subseteq V'$ be the set of nodes in tree T_j and $C(T_j) \subseteq O$ the set of targets covered by the sensor nodes in $V(T_j)$ with $1 \leq j \leq L$. Recall that N_c^o is the subset of time slots in which target o is covered for the monitoring period of L time slots, where $N_c^o = \{j \mid \exists j \ s.t. \ o \in C(T_j), 1 \leq j \leq L\}$. For each node $v_i^k \in V'$ that has not been contained by any tree and one of its adjacent nodes in G' is in tree T_j, we can calculate the potential utility gain of the coverage quality ΔU_{ij} if node v_i^k is added to

T_j by Eq.(2.14),

$$\begin{cases} 0 & V(v_i^k) \cap V(T_j) \neq \emptyset \text{ implies that another copy of } v_i \text{ has been contained by tree } T_j \\ \alpha \cdot \sum\limits_{o \in \{C(v_i^k) - C(T_j)\}} (f_1(N_c^o \cup \{j\}) - f_1(N_c^o)) + (1-\alpha) \cdot \sum\limits_{o \in C(v_i^k)} (f_2(S_o^j \cup \{v_i\}) - f_2(S_o^j)) \\ \quad \text{otherwise} \end{cases} \tag{2.14}$$

where $V(v_i^k) \cap V(T_j) \neq \emptyset$ represents that sensor v_i has already been activated at time slot j.

Algorithm 1 `Greedy_Heuristic`

Input: A renewable sensor network $G = (V \cup \{s\}, E)$, a set of targets O, and time slots that are indexed by $1, 2, \cdots, L$. For each sensor $v_i \in V$, its energy budget $P(v_i)$ in L time slots is given.

Output: For each time slot j, a set of sensors $V_j \subseteq V$ which will be activated at time slot j with $1 \leq j \leq L$.

```
Calculate its time slot budget b_i by its energy budget P(v_i) for each sensor v_i ∈ V;
Construct an auxiliary graph G' = (V' ∪ {s_1, s_2, ···, s_L}, E');
Construct a forest in G' consisting of L trees T_1, T_2, ···, T_L rooted at nodes
  s_1, s_2, ···, s_L, respectively;
T_j ← ({s_j}, ∅) initially, 1 ≤ j ≤ L;
W ← V'; /* The nodes in W have not been examined */
/* Add the nodes in W to the L trees one by one */
zero_gain ←' true';
while (there is a node in W that has not been contained by any tree) and zero_gain
  do
    Calculate the gain of the coverage quality ΔU_ij for each node v_i^k ∈ W and one
    of its adjacent nodes in a tree T_j rooted at s_j for each of these adjacent nodes in
    the adjacent list of v_i^k;
    Identify a node v_{i'}^{k'} with the maximum ΔU_{i'j'} among the nodes in W;
    if ΔU_{i'j'} == 0 then
      zero_gain ←' false'; /* No further improvement in the coverage quality is
      achieved */
    else
      V(T_{j'}) ← V(T_{j'}) ∪ {v_{i'}^{k'}}; /* Add node v_{i'}^{k'} to tree T_{j'} */
      W ← W \ {v_{i'}^{k'}};
    end if
end while
Construct V_j from V(T_j) by adding the corresponding sensor of a copy of a sensor
  in V(T_j);
return The set of active sensors at time slot j is V_j for all j with 1 ≤ j ≤ L.
```

We then choose a node $v_{i'} \in V'$ with the maximum utility gain of the coverage quality $\Delta U_{i'j'}$, and add $v_{i'}$ to tree $T_{j'}$ if this results in the maximum gain of the coverage quality. This procedure continues until all nodes are added to the forest or no further improvement in the coverage quality can be achieved. That is, either all nodes in G' have been added to the trees in the forest, or no node addition results in a positive utility gain of the coverage quality. As a result, trees $T_1, T_2, \cdots, T_L$ rooted at nodes $s_1, s_2, \cdots, s_L$ are obtained, where the nodes in tree T_j rooted at s_j represent that their corresponding sensors in G will be activated at time slot j, and these sensors and the sink will be connected, $1 \le j \le L$. Notice that it is very likely there are some trees in the forest containing the root node only. If this is the case, it implies that none of the sensors in the network at the corresponding time slot of this tree is active. The detailed description of the proposed algorithm is given in `Algorithm` 1.

Theorem 1 *Given a renewable sensor network $G = (V \cup \{s\}, E)$ deployed for monitoring a set of targets in the region for a period of L time slots, there is an algorithm* `Greedy_Heuristic` *for the coverage maximization problem, which takes $O(b_{max}^3 \cdot |V|^2 \cdot |E| + b_{max} \cdot d_{max} \cdot L)$ time, where $|V|$ is the number of sensor nodes, $b_{max} = \max_{v_i \in V}\{b_i\}$, $d_{max} = |N(v)|$, and $N(v)$ is the set of neighbors of node v in G. Notice that d_{max} usually is a constant, while b_{max} is a constant and even if it is not, then $b_{max} \ll L$.*

Proof We first show that the algorithm is correct. That is, each sensor node will not run out of its energy budget. As there are b_i nodes for sensor v_i in G' with each corresponding its energy consumption at one time slot. Thus, v_i will not run out of its energy budget as it can only join at most b_i trees. Following the construction of the trees, each of the b_i copies of v_i can appear in a tree only once. Also, within the time slot to which a tree corresponds, all sensors in the tree will be activated, and the activated sensors and the sink are in the same connected component. Thus, the solution delivered by algorithm `Greedy_Heuristic` is a feasible solution to the coverage maximization problem.

We then analyze the time complexity of the proposed algorithm `Greedy_Heuristic` in the following. The auxiliary graph G' contains at most $|V| \cdot b_{max}$ nodes since there are at most b_{max} copies in G' of each node in G. The number of edges in G', $|E'|$, is no more than $d_{max} \cdot b_{max} \cdot L + \sum_{e \in E} b_{max}^2 = b_{max} \cdot d_{max} \cdot L + b_{max}^2 \cdot |E|$ edges. Thus, the construction of G' takes $O(b_{max} \cdot d_{max} \cdot L + |V| \cdot b_{max} + b_{max}^2 |E|)$ time. Within each iteration, for each unscheduled node $v_i^k \in V'$, let $N_{G'}(v_i^k)$ be its neighbor set in G', we need to calculate the incremental coverage quality ΔU_{ij} for each $v' \in V(T_j) \cap N_{G'}(v_i^k)$ with tree root s_j, and choose a node $v_{i'}^{k'}$ with the maximum incremental coverage quality among the unscheduled nodes in V', this takes $O(\sum_{v_i^k \in V'} |N_{G'}(v_i^k)| \cdot |V'| \cdot C_{max}) = O(b_{max}^2 \cdot |V| \cdot |E| \cdot C_{max}) = O(b_{max}^2 |V||E|)$ time, where C_{max} is the maximum number of targets covered by a sensor, which usually is a constant in practice. It is easy to verify that the number of iterations of the proposed algorithm is bounded by $|V'|$. The algorithm thus takes $O(b_{max} \cdot |V| \cdot b_{max}^2 \cdot |V| \cdot |E| + b_{max} \cdot d_{max} \cdot L) = O(b_{max}^3 |V|^2 |E| + b_{max} \cdot d_{max} \cdot L)$ time. $\square$

2.4.2 Distributed Algorithm

As real sensor networks are distributive, it is desirable that algorithms for sensor networks are distributed algorithms, whereas the solution obtained by the centralized algorithm usually serves as the benchmark of the solutions obtained by distributed algorithms. In this section, we propose a distributed implementation of the proposed centralized algorithm `Greedy_Heuristic`. Following most common assumptions in the design of distributed algorithms, we assume that the amount of energy consumed for finding a distributed solution can be neglected, in comparison with the amount of energy consumed for sensing coverage, local computation and sensing data transmission.

The idea behind the distributed implementation is that we treat the original network G as a *host graph*, and the constructed auxiliary graph G' as a *guest graph*. We 'embed' the guest graph into the host graph. Each node v_i in the host graph G simulates its b_i copies in the guest graph G'. Each link (v_i, v_j) in the host graph G simulates its corresponding $b_i \cdot b_j$ links in the guest graph G' between the copies of nodes v_i and v_j. In the host graph G, there is a broadcast tree which is dynamically constructed. The broadcast tree will be used for tree information broadcasting of the L trees constructed from G', it also serves as collecting 'joining-tree request' messages from non-tree nodes in G'. In the guest graph G', there is a forest consisting of the L trees with the sensors in each tree corresponding to the activated sensors at one time slot among the L time slots in the monitoring period. The sink contains the L trees of the forest with each tree T_j having a tree root at s_j and spanning all activated sensors at time slot j, $1 \leq j \leq L$. Assume that the broadcast tree in G contains the sink only initially.

The construction of the forest $\mathcal{F}$ which consists of the L trees $T_1, T_2, \ldots, T_L$ proceeds iteratively. Within each iteration, some nodes in V' join some of the L trees in the forest, and their 'joining-tree request' messages will be propagated to the sink along the links of the broadcast tree. The sink then calculates the coverage quality and broadcasts the L tree messages to those unjoined nodes which are close to the tree nodes, i.e., there is an edge in G' between a tree node and an unjoined node. This procedure continues until either all the nodes in V' have joined the trees in the forest, or there is no improvement on the utility gain of the coverage quality. In the following, we detail the distributed implementation of the proposed algorithm at iteration t.

Within iteration t, let $V_t(\mathcal{F})$ be the set of nodes in the forest and $W_t = V' \setminus V_t(\mathcal{F})$ the set of nodes that are not in the forest yet. Assume that each node in $V_t(\mathcal{F})$ is labelled as a *tree node* which contains the following information: *its tree root, the set of members in the tree, and the value of the coverage quality*. Let $E_t = E' \cap (V_t(\mathcal{F}) \times W_t)$ be the set of edges in G' across the two sets $V_t(\mathcal{F})$ and $V' \setminus V_t(\mathcal{F})$. For each unlabeled node in $v \in W_t$, let $(v, u_1), (v, u_2), \cdots, (v, u_l)$ be its incident edges in E_t. These l nodes $u_1, u_2, \ldots, u_l$ form a set, which is then partitioned into l' subsets, where all the nodes in the same tree in $\mathcal{F}$ belong to the same subset. Discard these subsets in which the trees contain a copy of v already. Denote by l'' the remaining subsets (or trees). Clearly $l'' \leq l' \leq l$. Compute the utility gain of the unlabeled node v if it is added to one of the l'' trees, identify a tree with the maximum gain of the utility, and v then sends a 'joining-tree request' to the tree node and puts it as a candidate of joining that tree. All tree nodes send their received 'joining-tree request' messages to the sink. The sink then updates the members of the trees in the forest $\mathcal{F}$, by adding the new members to the trees and updating their utility values. For a given tree (e.g.,

T_j), there may have multiple joining-tree requests such as (v, u) and (v', w) where $u, w \in V(T_j)$. If both v and v' are different copies of the same sensor, only one of them will join the tree. Or, if there is no positive gain for all trees or all the nodes in V' have been included in forest $\mathcal{F}$, the procedure terminates. Otherwise, the sink broadcasts the updated information of the L trees along the links of the broadcast tree. Each unlabeled node in G' that has sent a 'joining-tree request' message will check whether it becomes a member in its requested tree. If yes, label itself as a tree node, and check whether its host node is included in the broadcast tree already. If not, set the host node as a tree node in the broadcast tree, and send a message to its parent host node. The parent host node then sets the host node as one of its children in the broadcast tree.

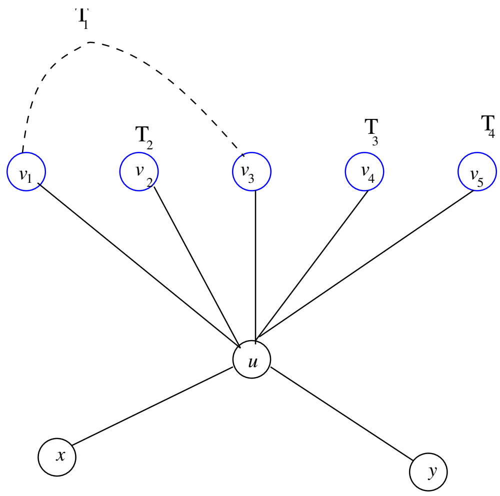


Figure 2.3: An illustration of an unlabeled node v joining one of the L trees.

We here use an example to illustrate the procedure of node joining the trees (see Fig. 2.3). Assume that an unlabeled node u has 5 tree neighboring nodes $v_1, v_2, \ldots, v_5$, and two unlabeled neighboring nodes x and y. We further assume that v_1 and v_3 are in the same tree in the forest and denote by this tree as T_1. Nodes v_2, v_4 and v_5 are in trees T_2, T_3 and T_4, respectively. We further assume that tree T_3 contains a copy of sensor u already. Thus, in this case $l = 5$, $l' = 4$ and $l'' = 3$. Node v can join either of trees T_1, T_2, and T_4. Assume that u joining T_2 will result in the maximum utility gain of coverage quality utility, then node u sends a 'joining-tree request' to the tree node v_2 for joining T_2. The sink then updates each of the L tree information once it receives all 'joining tree request' messages from its tree nodes. Assume that it updates tree T_2, if there is no other messages from the other unlabeled nodes that are the copies of the same sensor as node u, then u is added to T_2 as a new member. Otherwise, the sink chooses one of different copies of the same sensor to admit, and broadcasts all updated tree information to each tree node through the broadcast tree. When u received the updated message, it checks whether it has been admitted. If yes, set itself as a tree node, and also check whether its host node is in the broadcast tree. Otherwise, set the host node as a tree node in the broadcast tree, and send its parent in the tree a message that it will be its child, and its parent node sets it as one of its children.

Now, we estimate the utility gain delivered by the proposed distributed algorithm. Consider a tree T_j at iteration t, assume that the member set of T_j is $V_t(T_j)$ prior to iteration t. Let $v_1, v_2, \ldots, v_k$ be the nodes added to T_j after iteration t, then the estimated gain of the utility in T_j is $\sum_{i=1}^{k} U(V_t(T_j) \cup \{v_i\})$ when these nodes joined it. The actual increase on the utility gain in tree T_j however is $U(V_t(T_j) \cup \{v_i \mid 1 \le i \le k\}) \le \sum_{i=1}^{k} U(V_t(T_j) \cup \{v_i\})$. The detailed implementation of Algorithm `Distributed_Implement` consists of two subroutines `Distributed_Implement_Base_Station` as Algorithm 2 and `Distributed_Implement_Sensor` as Algorithm 3.

Algorithm 2 `Distributed_Implement_Base_Station`

Broadcast an initial message which contains the following information: L trees with each having root at it, its coverage quality utility value, and its members;
while Receive 'joining-tree request' messages from its broadcast tree nodes **do**
  if No 'joining-tree request' messages are received or all nodes are included in the forest **then**
    Terminate; /*The sensor schedules are finalize*/
  else
    Process received requests by removing redundancies. That is, for a given tree T_j, there may have multiple joining requests originated from the same sensor, then only one of them will join;
    Broadcast the updated broadcast message which contains the updated tree nodes and the value of coverage quality along the broadcast tree edges to each tree node; /* Start next iteration */
  end if
end while

Algorithm 3 `Distributed_Implement_Sensor`

```
while Receive a broadcast message from its neighbor nodes or the sink do
    if It is already a tree node then
        Broadcast this message to its children nodes or other neighbor nodes;
    else if Its 'joining-tree request' in the previous round has been admitted then
        Label itself as a tree node;
        Broadcast this message to its neighbor nodes;
    else
        Identify which tree that it should join through computing the utility gain
        of the coverage quality if it is added to the tree, and choose a tree with the
        maximum gain of the utility;
        Send a 'joining-tree request' message to its parent node;
    end if
end while
while Receive 'joining-tree request' messages from other neighbor nodes or its
children nodes do
    Forward the received messages along its tree paths towards its parent nodes;
end while
```

Lemma 1 *Algorithm* `Distributed_Implement` *delivers a feasible solution to the coverage maximization problem.*

Proof Since algorithm `Distributed_Implement` consists of a number of iterations, we show that the final L trees in the forest is a feasible solution to the problem by induction on the number of iterations. At iteration $t = 0$, there are L trees with each containing a root node only. It is a feasible solution. Let $\mathcal{F}_t$ be the forest of the L trees constructed so far by iteration $t - 1$, in which each tree meets the following conditions: (1) there is no more than one copy of each sensor in each tree; (2) the communication subgraph induced by the sensor nodes in each tree and the sink (the tree root) is connected. We now deal with iteration t. Within iteration t, some unlabeled nodes (or non-tree nodes) join the trees in $\mathcal{F}_t$. Clearly, if another copy of a joining node is already in a tree, it will not be added to the tree. Or, if there are multiple copies of a sensor seeking to join a tree, only one of them will succeed. Also, there must have an edge in G' between a tree node and the joining node. Thus, the resulting forest $\mathcal{F}_{t+1}$ is still feasible. When no positive utility gain of the coverage quality can be obtained at iteration t, this implies that the trees containing the neighbors of each node $v \in W_t$ have already contained another copy of the sensor that node v is one of its copies. The lemma then follows. □

Theorem 2 *Given a renewable sensor network $G = (V \cup \{s\}, E)$ deployed to monitor a set of targets for a period of L time slots, there is a distributed algorithm* `Distributed_Implement` *for the coverage maximization problem, which takes $O(L|V| + |V|^2)$ time and $O(L|V|^2 + |E|)$ messages, where $|V|$ is the number of sensors and $|E|$ is the number of links in G.*

Proof Following Lemma 1, it can be seen that algorithm `Distributed_Implement` will deliver a feasible solution to the coverage maximization problem. Assume that there are l iterations of the entire algorithm. Within iteration i, the amount of time spent for the message broadcasting of the L trees is $\max\{L, t_i\}$ by broadcasting the L tree messages along the tree edges of the broadcast tree in a pipeline manner, where t_i is the longest one among the shortest distances between the sink and a node in W_t at iteration i, clearly $t_i \leq |V|$, $1 \leq i \leq l$. The time for collecting the 'joining-tree request' messages from joining nodes in W_t through the tree edges is t_i, The number of messages needed for iteration i thus is $m_i = O(L(n_i - 1) + |E_i|) = O(L|V| + |E_i|)$, where n_i is the number of nodes in the broadcast tree of the host graph at iteration i. There are l iterations of the distributed implementation of the proposed algorithm, thus, the time complexity of the distributed implementation of the proposed algorithm is $O(\sum_{i=1}^{l} \max\{L, t_i\}) = O(\sum_{i=1}^{l} \max\{L, |V|\}) = O(\max\{L|V|, |V|^2\}) = O(L|V| + |V|^2)$ since $l \leq |V|$. Similarly, the number of messages needed by the distributed implementation of the proposed algorithm is $O(\sum_{i=1}^{l} m_i) = O(\sum_{i=1}^{l}(L|V| + |E_i|)) = O(L|V|^2 + \sum_{i=1}^{l} |E_i|) = O(L|V|^2 + |E|)$ since $\sum_{i=1}^{l} |E_i| = |E|$. The theorem then follows. □

2.5 Online Dynamic Optimization Framework

The proposed centralized and distributed algorithms so far for the coverage maximization problem are based an assumption. That is, the energy budget of each sensor for the entire monitoring period of L time slots can be accurately predicted. In reality, the accuracy of energy prediction however depends heavily on weather conditions and the prediction duration. Particularly, a longer period prediction usually is less accurate. The assumption thus is problematic in realistic applications, and especially for sensors whose actual amounts of harvested energy are significantly less than their predicted amounts, they may not have enough energy to maintain their scheduled activities for the monitoring period. Moreover, other active sensors with sufficient energy may also be inversely affected by these sensors when they serve as relay nodes between the sink and the sensors with sufficient energy. Consequently, the overall coverage quality of the network will drastically degrade. To remove or eliminate this realistic assumption, in this section we propose an adaptive framework to deal with harvesting energy prediction fluctuations, and show that under this adaptive framework, the proposed centralized and distributed algorithms are still applicable.

The basic idea is that we schedule sensor activities by a 'dynamic interval' concept, where an interval consists of the number of consecutive time slots that is significantly less than L, while the length of an interval is adaptively determined by the energy prediction accuracy so far. Thus, the entire monitoring period of L time slots consists of a number of intervals, and the proposed algorithm `Greedy_Heuristic` or `Distributed_Implement` is applied within each of these intervals. The only modification to these algorithms is that we cannot fully make use of all predicted energy budget for this interval, as the sensors in future intervals may not be recharged again. Instead, we only use a fraction γ of the energy budget for the current interval, e.g., $0.4 \leq \gamma < 1$. Specifically, let $|I_i|$ be the number of time slots in an interval I_i.

In the beginning of interval I_i, we first compute the amount of predicted energy of each sensor in this interval, by applying a given prediction algorithm EWMA in [44]. We then schedule sensor activities within the interval by applying algorithm `Greedy_Heuristic` (or algorithm `Distributed_Implement`). Given an interval I_i, let $V(I_i)$ be the set of active sensors in I_i. The energy prediction accuracy of a sensor $v \in V(I_i)$ $\theta_i(v)$ is defined as $\theta_i(v) = \frac{|Q_v - \bar{Q}_v|}{Q_v}$, where Q_v and $\bar{Q}_v$ are the actual and predicted amounts of harvested energy of sensor v in I_i. Denote by $\theta_i = \sum_{v \in V(I_i)} \theta_i(v)/|V(I_i)|$ *the energy prediction accuracy* of interval I_i, which is the average energy prediction accuracy among active sensors in this interval. We adaptively adjust the number of time slots $|I_{i+1}|$ for the next interval I_{i+1} by the energy prediction accuracy θ_i in I_i. The number of time slots $|I_{i+1}|$ for the next interval I_{i+1} is defined as follows.

$$|I_{i+1}| = \begin{cases} \max\{1, \lfloor |I_i| \cdot \beta \rfloor\} & \theta_i \geq \epsilon \\ \min\{L_{ini}, \lfloor \frac{|I_i|}{\beta} \rfloor, L'\} & \text{otherwise,} \end{cases} \tag{2.15}$$

where β is a tuning rate with the default value of 0.5 in the rest of chapter with $0 < \beta \leq 1$, L_{ini} is a given initial value with the default value of $\lceil 0.2 \cdot L \rceil$, and $L' \leq L$ is the remaining available number of time slots for a monitoring period of L time slots, i.e., $L' \leq L$. That is, when the energy prediction in interval I_i is quite accurate (i.e., the value of θ is less than a given threshold ϵ), the number of time slots $|I_{i+1}|$ is increased for the next interval I_{i+1} by setting $|I_{i+1}| = \frac{|I_i|}{\beta}$ until it is either L_{ini} or L'; otherwise, the number of time slots is decreased by setting $|I_{i+1}| = |I_i| \cdot \beta$ until it decreases to 1. Thus, the entire monitoring period of L time slots consists of a number of variable-length intervals. This procedure continues until all the L time slots have been scheduled. The detailed adaptive optimization framework for the quality coverage maximization problem is described in `Algorithm` 4.

Notice that in terms of the energy budget allocation to the current interval I_k in Algorithm `Adaptive_Framework`, only a fraction of the energy budget $P^k(v_i)$ of each sensor $v_i \in V$ is allocated to interval I_k. The rationale behind is that we need to keep some residual energy of the sensor for later intervals if no further energy can be harvested in future intervals (such as obtaining the solar energy in the middle of night).

Theorem 3 *Given a renewable sensor network $G = (V \cup \{s\}, E)$ deployed to monitor a set of targets in the region for a period of L time slots, there is an algorithm* `Adaptive_Framework` *for the coverage maximization problem, which takes $O(b_{max}^3 |V|^2 |E| + d_{max} b_{max} L)$ time, where $|V|$ is the number of sensors, where $b_{max,i} = \max_{v_j \in V}\{b_j\}$ at interval I_i, $b_{max} = \sum_{i=1}^{l} b_{max,i}$, $d_{max} = |N(v)|$, and $N(v)$ is the set of neighbors of node v in G, assuming that there are l intervals to cover the entire monitoring period of L time slots. Notice that d_{max} usually is a constant while b_{max} is a constant and even if it is not, then $b_{max} \ll L$.*

Algorithm 4 `Adaptive_Framework`

Input: A renewable sensor network $G = (V \cup \{s\}, E)$, a set of targets O, and time slots that are indexed by $1, 2, \ldots, L$.
Output: Schedule sensor activities in entire L time slots.

/* These settings can be changed according to specific requirements */
$\beta \leftarrow 0.5$; $L_{ini} \leftarrow \lceil 0.2 \cdot L \rceil$;
$|I_1| \leftarrow L_{ini}$; /* Initial the first interval */
$L' \leftarrow L$; /* The remaining number of time slots for the entire of L time slots */
/* Schedule sensors' activities interval by interval */
/*Assume that the current interval is I_k with $k \geq 1$*/
while $L' > 0$ **do**
  for each sensor $v_i \in V$ **do**
    Predict the amount of energy harvested of v_i in the current interval I_k;
    Compute its energy budget $P^k(v_i)$ by Eq. (2.3);
    The amount energy budget allocated for the current interval I_k is $\gamma B^k(v_i)$ where γ is a constant with $0.4 \leq \gamma < 1$, e.g., $\gamma = 0.5$
  end for;
  Schedule sensor activities within the current interval I_k by invoking algorithm `Greedy_Heuristic` (or algorithm `Distributed_Implement`). Notice that in the construction of the auxiliary graph, instead of L trees rooted at s_j with $1 \leq j \leq L$, there are $|I_k|$ trees rooted at s_j^k with $1 \leq j \leq |I_k|$, the budget of each sensor v_i now is b_i^k in the current interval I_k.
  $L' \leftarrow L' - |I_k|$; /* Update the remaining available number of time slots */
  /* In the end of the current interval, examine the energy prediction accuracy θ in the current interval; adjust the number of time slots in the next interval according to the energy prediction accuracy by Eq. (2.15) */
  if $\theta_k \geq \epsilon$ **then**
    /* decrease the number of time slots in the next interval */
    $|I_{k+1}| \leftarrow \max\{1, \ \lfloor |I_k| \cdot \beta \rfloor\}$;
  else
    /* increase the number of time slots in the next interval */
    $|I_{k+1}| \leftarrow \min\{L_{ini}, \ L', \ \lfloor \frac{|I_k|}{\beta} \rfloor\}$;
  end if
end while

Proof Following Theorem 1, it can be seen that algorithm `Adaptive_Framework` will deliver a feasible solution to the coverage maximization problem. Assume that there are l intervals of the entire monitoring period of L time slots, denoted by $I_1, I_2, \ldots, I_l$, respectively. Let I_i be the ith interval with $1 \leq i \leq l$, i.e., $\sum_{i=1}^{l} |I_i| = L$. Let $b_{max,i}$ be the maximum number of energy budget among sensors at interval i. Thus, algorithm `Greedy_Heuristic` will be invoked l times, and the amount of time taken by each of its invoking is $O(b^3_{max,i}|V|^2|E| + d_{max}b_{max,i}|I_i|)$ for interval I_i. The algorithm `Adaptive_Framework` consists of l intervals, thus, its time complexity is

$$\begin{aligned}
&O(\sum_{i=1}^{l}(b^3_{max,i}|V|^2|E| + d_{max}b_{max,i}L)) \\
&= O(|V|^2|E|(\sum_{i=1}^{l} b_{max,i})^3 + d_{max}L(\sum_{i=1}^{l} b_{max,i}))) \\
&= O(b^3_{max}|V|^2|E| + d_{max}b_{max}L), \qquad (2.16)
\end{aligned}$$

where $b_{max} = \sum_{i=1}^{l} b_{max,i}$. $\square$

Notice that the time complexity of each interval is affected by both the network size and the interval length. The interval length should be carefully chosen based on the network size, such that the algorithm can run fast in time less than the interval length.

The distributed implementation of algorithm `Distributed_Implement` is similar to the one in the previous section, omitted.

2.6 Performance Study

In this section, we study the performance of the proposed algorithms through experimental simulation. We also investigate the impact of related parameters: network size, number of targets, tuning rate β, threshold ϵ, and parameter γ on the coverage quality.

2.6.1 Experimental Environment Setting

We consider a renewable sensor network consisting of 100 to 500 sensors randomly deployed in a $100m \times 100m$ square region, where a sink is randomly located. The targets in O are also randomly deployed in this square region. We consider a monitoring period of 24 hours with each time slot of 30 minutes, i.e., the monitoring period consists of $L = 48$ time slots. We adopt the energy consumption parameters of real radio CC2420 [89], which consumes $56.4mW$ and $0.06mW$ when it is in active and sleep modes, respectively. Each sensor is powered by a solar panel with a dimension $10mm \times 10mm$. The solar power harvesting profile is derived from the solar data profiles in The National Solar Radiation Data Base (NSRDB) in the States [7], which contains the most comprehensive collection of solar data. Specifically, for each different network topology for a one day monitoring period, each sensor node is assigned a solar data sequence of one day. Each data item in the sequence is the amount of energy harvested in that 30-minute time slot of that day. For the sake of convenience, we assume that both the sink and sensor nodes have identical transmission ranges of 20 and sensing ranges of 25 meters. We further assume that the given coverage quality weight α is 0.5 in the default setting. Denote by `LOG` a utility function which is the sum of two sub-modular functions: $f_1(N_c^o) = \log(|N_c^o| + 1)$ and $f_2(S_o^t) = \log(|S_o^t| + 1)$. Similarly, denote by `SQR` another utility function which is the sum of two sub-modular functions: $f_1(N_c^o) = \sqrt{|N_c^o|}$ and $f_2(S_o^t) = \sqrt{|S_o^t|}$. We will adopt these two different utility functions to measure the target coverage quality. Each value in figures is the mean of the results by applying each mentioned algorithm to 30 different network topologies with the same network size.

2.6.2 Harvested Energy Prediction

We first investigate the accuracy of the harvested energy prediction approach `VEWMA` in comparison with the one of a basic prediction approach `EWMA`, using the real solar data profiles obtained from The National Solar Radiation Data Base (NSRDB) in the States [7] which contains the most comprehensive collection of solar data and is freely available.

Fig. 2.4 shows the actual solar data measurements within 10 consecutive days under different weather conditions and the predicted values by algorithms `EWMA` and `VEWMA`, respectively, from which it can be seen that the accumulative error between the estimated ones and the real ones is given by the following equation.

$$Error = \frac{1}{M}\sum_{i=1}^{M}|1 - \frac{Real}{Estimated}| \tag{2.17}$$

Where M is the number of predictions made in the past. By setting the weight w to be 0.5, both algorithms `EWMA` and `VEWMA` will deliver small accumulative errors. Specifically, the error by algorithm `VEWMA` is 9.1%, compared with 12.6% by algorithm `EWMA`. Given 3 different independent datasets, Fig. 2.5 implies that with the increase of w from 0.1 to 0.9, the errors by both algorithms `VEWMA` and `EWMA` decrease slightly. However, when the value of w is greater than 0.9, the errors by both algorithms `VEWMA` and `EWMA` increase significantly and can reach upto from 66% to 300%. In order to obtain better harvesting energy prediction performance, the value of w should be adjusted, based on the historical harvesting energy profiles.

2.6.3 Performance Evaluation of Centralized and Distributed Algorithms on the Coverage Quality

We then investigate the proposed centralized algorithm `Greedy_Heuristic` and the distributed implementation `Distributed_Implement`, against a variant of an existing centralized algorithm in [37] `CPS_Cover` which finds such a connected sensor cover that maximizes the number of targets covered at each time slot. The number of sensors varies from 100 to 500, and the number of targets $|O|$ is set as 25 and 50, respectively.

Fig. 2.6(a) clearly shows that in terms of the coverage quality function `SQR`, the centralized algorithm `Greedy_Heuristic` significantly outperforms algorithms `Distributed_Implement` and `CPS_Cover`, and algorithm `CPS_Cover` is the worst among all three mentioned algorithms. The coverage quality of algorithm `Greedy_Heuristic` is around 30% higher than that of algorithm `Distributed_Implement`, regardless of the number of targets $|O|$ is either 25 or 50. With the growth of network size,

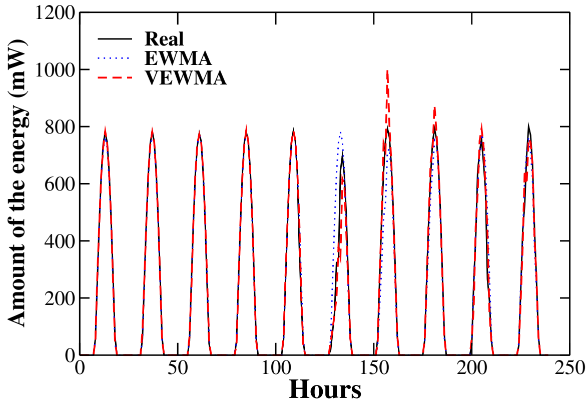


Figure 2.4: The accuracy performance of prediction algorithms VEWMA and EWMA with weight $w = 0.5$.

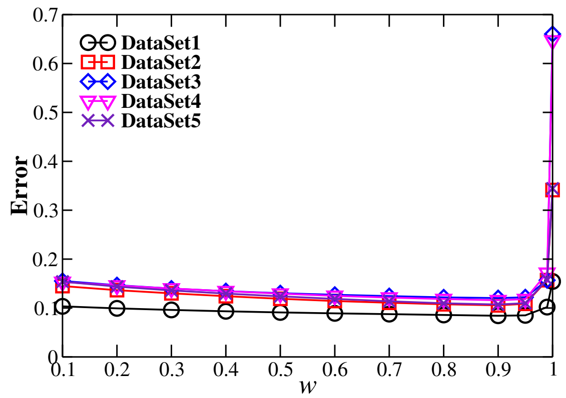
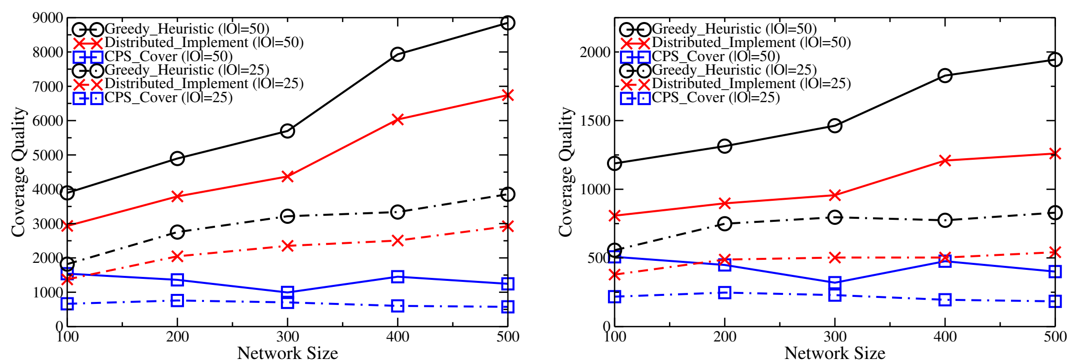


Figure 2.5: Accumulative errors of prediction accuracy by algorithms VEWMA and EWMA with different weights w.



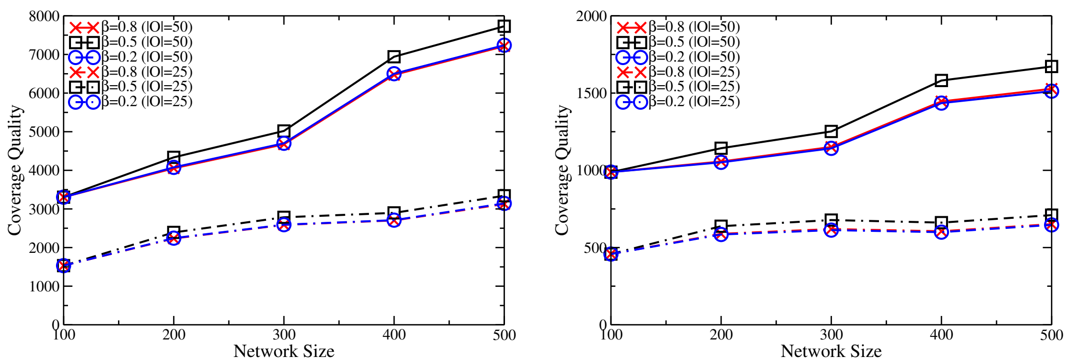
(a) SQR-metric (b) LOG-metric

Figure 2.6: Performance of centralized algorithm Greedy_Heuristic and distributed algorithm Distributed_Implement under different quality measure functions SQR and LOG.

this performance gap is still stable. The coverage quality delivered by algorithms `Greedy_Heuristic` and `Distributed_Implement` is at least 100% more than that of algorithm `CPS_Cover`. For the coverage quality function `LOG`, Fig. 2.6(b) exhibits similar performance behaviors, and the coverage quality delivered by algorithm `Greedy_Heuristic` is about 50% higher than that by algorithm `Distributed_Implement`. With the increase of network size, it can be also seen from Fig. 2.6 that the coverage quality delivered by algorithms `Greedy_Heuristic` and `Distributed_Implement` increases accordingly. The coverage quality delivered by both algorithms increase too when the number of targets increases, while keeping the network size fixed.

2.6.4 Impact of Tuning Rate β on the Performance of Dynamic Framework

We also study the efficiency of the proposed dynamic optimization framework `Adaptive_Framework`, where algorithm `Greedy_Heuristic` is employed as its subroutine. We fix the threshold ϵ at 0.2 and the parameter γ at 0.5 while putting the tuning rate β as 0.2, 0.5, and 0.8, respectively.



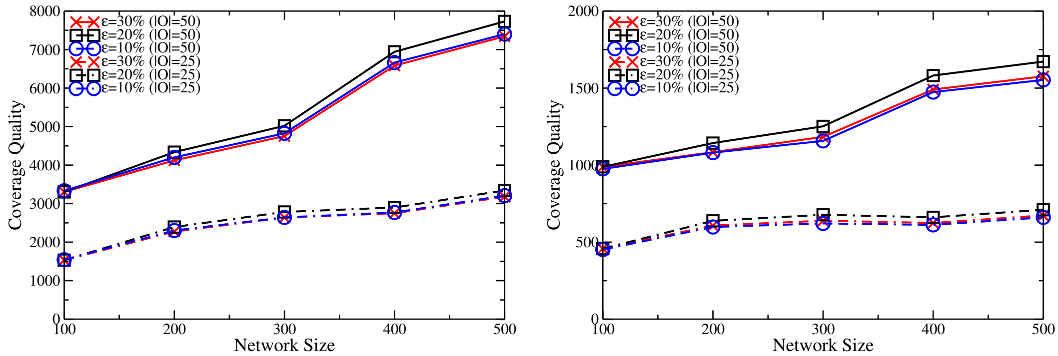
(a) `SQR-metric` (b) `LOG-metric`

Figure 2.7: Impact of tuning rate β on the performance of `Adaptive_Framework` under different quality measure functions *SQR* and *LOG*.

Fig. 2.7 demonstrates that the coverage quality delivered by `Adaptive_Framework` is the highest in comparison with the other settings when the tuning rate $\beta = 0.5$. For example, when the number of targets is fixed at 25, for the coverage quality function `SQR` in Fig. 2.7(a), the coverage quality delivered by the algorithm when $\beta = 0.5$ is about 5% and 6% higher than that by the algorithm when $\beta = 0.2$ and $\beta = 0.8$, respectively. For the coverage quality function LQG in Fig. 2.7(b), the coverage quality delivered by the algorithm when $\beta = 0.5$ is about 9% and 8% higher than that by it when $\beta = 0.2$ and $\beta = 0.8$, respectively.

2.6.5 Impact of Threshold ϵ on the Performance of Dynamic Framework

We also evaluate the impact of threshold ϵ on the coverage quality delivered by the proposed framework `Adaptive_Framework`, in which the subroutine `Greedy_Heuristic` is employed. We set the threshold ϵ as 0.1, 0.2, and 0.3 while fixing the tuning rate β at 0.5 and parameter γ at 0.5.



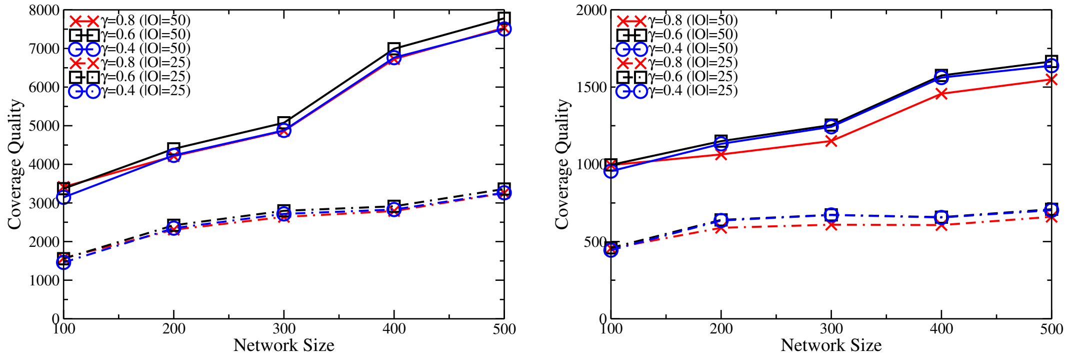
(a) `SQR-metric` (b) `LOG-metric`

Figure 2.8: Impact of threshold ϵ on the performance of `Adaptive_Framework` under different quality measure functions *SQR* and *LOG*.

Fig. 2.8(a) indicates that for the coverage quality function `SQR`, the coverage quality achieved by `Adaptive_Framework` is the highest compared with those of other settings when $\epsilon = 0.2$. Specifically, when the number of targets is fixed at 50, the coverage quality delivered by the algorithm with $\epsilon = 0.2$ is about 4% and 5% higher than those by it with $\epsilon = 0.1$ and $\epsilon = 0.3$, respectively. When the number of targets is fixed at 25, the coverage quality delivered by the algorithm with $\epsilon = 0.2$ is about 5% higher than that by it with $\epsilon = 0.1$ or $\epsilon = 0.3$. Fig. 2.8(b) exhibits the similar performance behaviors for the coverage quality function `LOG`, omitted.

2.6.6 Impact of Parameter γ on the Performance of Dynamic Framework

We finally evaluate the impact of parameter γ on the coverage quality delivered by the proposed framework Adaptive_Framework, in which the subroutine Greedy_Heuristic is employed. We set parameter γ as 0.4, 0.6, and 0.8 while fixing the tuning rate β at 0.5 and the threshold ϵ at 0.2, respectively.



(a) SQR-metric (b) LOG-metric

Figure 2.9: Impact of parameter γ on the performance of Adaptive_Framework under different quality measure functions SQR and LOG.

Fig. 2.9(a) implies that for the coverage quality function SQR, the coverage quality delivered by Adaptive_Framework with $\gamma = 0.6$ is higher than that by it with $\gamma = 0.4$ or $\gamma = 0.8$. Specifically, when the number of targets is fixed at 50, the coverage quality delivered by Adaptive_Framework with $\gamma = 0.6$ is about 3.5% higher than that by it with $\gamma = 0.4$ or $\gamma = 0.8$. When the number of targets is fixed at 25, the coverage quality delivered by the algorithm with $\gamma = 0.6$ is about 3% higher than that by it with $\gamma = 0.4$ or $\gamma = 0.8$. Fig. 2.9(b) exploits the performance behavior curves of Adaptive_Framework for the coverage quality function LOG. The coverage quality delivered by it with $\gamma = 0.4$ and $\gamma = 0.6$ is higher than or at the same level as that by the algorithm with $\gamma = 0.8$.

2.7 Conclusions

In this chapter we studied the quality-aware target coverage problem in a renewable sensor network deployed for monitoring a set of targets for a given monitoring period, where sensors are powered by renewable energy sources and operate in duty-cycle mode, for which we first introduced a new coverage quality metric that is a weighted linear combination of two utility sub-modular functions to measure the coverage quality within two different time scales. We then formulated a novel coverage maximization problem that takes both sensing coverage quality and network connectivity into consideration. Due to the NP-hardness of the problem, we instead devised efficient centralized and distributed algorithms, provided that the harvesting energy prediction of each sensor for the monitoring period is accurate. Otherwise, we proposed an adaptive framework to deal with energy prediction fluctuations. We finally evaluated the performance of the proposed algorithms through experimental simulations. Experimental results demonstrate that the proposed solutions are promising.

Chapter 3

Data Collection Maximization in Renewable Sensor Networks via Time-Slot Scheduling

3.1 Introduction

Wireless sensor network has emerged as a key technology for various applications such as environmental sensing, structural health monitoring, and area surveillance. In most applications, hundreds or even thousands of sensors normally are dispersed over the monitoring area, and these sensor nodes self-organize into a wireless network, in which each sensor node periodically reports its sensed data to the sink(s). Thus, how to efficiently collect the sensed data from scattered sensor nodes is one of the most critical challenges.

The conventional sensor network architectures are based on the assumption that the network is dense, so that the sensing data generated by sensors is transmitted to the sink(s) through multi-hop relays for further processing. As a consequence, in most cases the sink(s) are assumed to be static, and mobility is not considered as an option. Moreover, the sensor nodes near to the sink(s) usually bear disproportionate amounts of traffic and deplete their energy much faster than others. Thus, in case of failure or malfunctioning of sensors around the sink(s), network connectivity and coverage may not be guaranteed. To mitigate this uneven energy consumption among sensor nodes, the concept of mobile sinks has been exploited, and extensive studies have shown that mobile sinks can significantly improve various network performance including network lifetime, connectivity, data delivery reliability, throughput, etc [6, 52, 53, 67, 82, 98, 106, 108, 109]. Most of these studies focused on the network lifetime maximization when performing data gathering, and the proposed routing algorithms are not applicable for renewable sensor networks due to the fact that the network can virtually operate perpetually as long as the sensors are recharged with sufficient harvested energy. That is, the network lifetime in renewable sensor networks is no longer a major optimization objective, which creates a shift in research focus from energy efficient to energy neutral approaches [44]. In contrast, little attention has been paid to data collection in renewable sensor networks with mobile sinks.

In this chapter, we consider data collection in a renewable sensor network with a path-constrained mobile sink, where will employ a mobile sink (e.g., a vehicle) to periodically travel along the pre-defined path at a constant speed to collect data from its one-hop sensors. Such an application can be a highway traffic surveillance, where sensors are deployed along both sides of a highway for traffic monitoring to get traffic related information such as the number of vehicles, types of vehicles, and individual vehicle speeds, which can later be used for road usage and maintenance, and driver behavior analysis. Another potential application scenario is the ecosystem monitoring in a forest, e.g., such a network can be deployed for monitoring exotic plant growths and/or endangered animals (e.g., giant panda) existence and behavior observations, where humans or vehicles can only access the limited roads rather than everywhere in the forest. Also, a vehicle can receive sensing data from a sensor if the vehicle is within the transmission range of the sensor. The sensors that can communicate with the vehicle usually serve as gateways where the other sensors will forward their sensing data to them through multi-hop relays. There are many other applications that are also fitted in this application scenarios such as oil/gas/water pipeline monitoring [42], structural health monitoring for bridges [47], etc.

Specifically, the following issues must be addressed: (1) Due to the time-varying characteristics of energy renewable sources, the energy replenishment rate of each sensor is unknown in advance, the sensor thus must have its time-varying energy budget (amount of available energy) for transmitting data to avoid its energy expiration. (2) For a given sensor, it requires using different data transmission rates to transmit its data when the mobile sink is at different locations, while different transmission rates will consume different amounts of its transmission energy. (3) During each tour of the mobile sink, it is very likely that multiple sensors can communicate to the mobile sink at the same time. Simultaneous transmissions of these sensors will result in a collision at the mobile sink and none of the transmissions will succeed. In this chapter we will address these issues by scheduling sensors at which time slots to transmit their data to the mobile sink so that the accumulative volume of the data collected by the mobile sink per tour is maximized. We achieve this through incorporating the time-varying sensor energy budget and employing multi-rate wireless communications.

Our main contributions in this chapter are as follows. We consider data collection in a renewable sensor network, using a path-constrained mobile sink. We first formulate a novel data collection maximization problem by incorporating multi-rate transmissions and transmission time slot scheduling, and show that the NP-hardness of the problem. We then devise an offline algorithm with a provable approximation ratio for the problem, assuming that the global knowledge of the network and sensor profiles (their locations and available energy) are given. We also extend the proposed algorithm by minor modifications to solve a generalized case of the problem where the harvested energy at each sensor is not given and link communications are unreliable. We thirdly develop a fast, scalable online distributed algorithm for the problem without the global knowledge of the network and sensor profiles, which is more suitable for real distributed sensor networks. For a special case of the problem where each sensor has a fixed transmission power, we propose an exact solution for it. We finally conduct extensive experiments by simulations to evaluate the performance of the proposed algorithms. Experimental results demonstrate that the proposed algorithms are very promising and the solutions obtained are fractional of the optimum.

The remainder of the chapter is organized as follows. Section 3.2 reviews related work. Section 3.3 introduces the system model, notions, problem definition, and shows the NP-completeness of the problem. Section 3.4 devises an offline approximation algorithm with a provable approximation ratio for the problem. Sections 3.5 develops a fast, scalable online distributed algorithm, and Section 3.6 devises an exact solution to the problem when each sensor has only one fixed transmission power. Section 3.7 evaluates the performance of the proposed algorithms through experimental simulations, and Section 3.8 concludes the chapter.

3.2 Related Work

Most existing solutions in renewable sensor networks assumed that the collected data is routed to a fixed sink through multi-hop relays [26, 55, 61, 63, 91, 114]. For example, Liu *et al.* [61, 63] formulated the problem as a lexicographic maximin rate allocation problem, and provided a centralized algorithm for the problem by solving an integer linear program. Liang *et al.* [55] developed a fair rate allocation algorithm by incorporating temporal-spatial sensing data correlations. Zhang *et al.* [114] studied the problem as a utility maximization problem by representing the utility gain at each sensor node as a concave utility function. They proposed an efficient algorithm for finding the accumulative sum of utility gains in a tree network. Although the data collection paradigm based on fixed sinks may be applicable to small to mediate size networks, it is definitely not suitable for large-scale networks due to long delays on data delivery by multihop relay, limited communication bandwidth, etc. To mitigate the deficiencies brought by fixed sinks, a feasible solution is to introduce mobile sinks.

Sink mobility in conventional sensor networks has been extensively studied in the past few year and demonstrated that it can significantly improve various network performance including reducing the energy consumption of sensors, balancing the workload among the sensors, reducing the data delivery delays, and prolonging the network lifetimes [10, 16, 23, 29, 52, 56, 92, 105, 107, 109]. Most existing studies focused on minimizing the energy consumption so as to prolong the network lifetime since sensors are powered by energy-limited batteries. The use of a path-constrained mobile sink for data collection in conventional sensor networks has been well studied. For example, Kansal *et al.* [45, 84] addressed a network infrastructure based on the use of a path-constrained mobile sink for data collection, where a sensor sends its data to the sink along a minimum number of hops routing path. They proposed a speed control algorithm to maximize the volume of data collected. Assuming that the mobile sink moves at a constant speed, Gao *et al.* [29] addressed the energy minimization problem by proposing a novel data collection scheme, where sensors close to the trajectory of the mobile sink are chosen as 'subsinks' and other sensors make use of different subsinks for their data relays. They formulated the subsink choice problem as a problem of minimizing the number of hops from each sensor to its subsink by providing a heuristic solution. They also studied time slot allocations for subsinks when the mobile sink collect data from the subsinks. Chakrabarti *et al.* [16] considered the dependence of transmission setting and packet loss rate of the mobile data collection problem by modeling the process of data collection as an M/D/1 queue. They then proposed an algorithm that ensures adequate data collection and minimizes the energy consumption. Liang *et al.* [56] considered another data collection problem by assuming that the subsinks (the gateways) are given in advance. They devised several approximation algorithms for the problem, by formulating the problem as a minimum cost capacitated forest problem that finds a minimum cost capacitated forest consisting of routing trees rooted at gateways and spanning all sensors. Unlike the mentioned work in conventional sensor networks that focused on energy conservation to prolong the network lifetime, maximizing network lifetime is no longer a main issue for renewable sensor networks as the sensors can be continuously recharged by renewable energies. Thus, in principle, such networks can be operational perpetually. Unfortunately, very little attention has been paid to data collection in renewable sensor networks, by using mobile sinks.

3.3 Data Collection Maximization Problem

We consider a renewable sensor network $G = (V \cup \{s\}, E)$ where V is a set of n stationary sensors that are densely deployed along a pre-defined path, and a mobile sink s periodically travels along the path at a constant speed r_s without stops to collect data from one-hop sensors. Each sensor is powered by renewable energy (e.g., solar energy) and has stored enough sensing data for collection. There is a link in E between a sensor $v \in V$ and the mobile sink s when they are within the transmission range of each other. Assume that the maximum transmission range of each sensor is R, and the length of the pre-defined path is L. The duration per tour by the mobile sink is determined by its traveling speed r_s, which is referred to as the *data latency*. That is, the faster the mobile sink travels, the shorter the duration per tour is, resulting in a shorter delay on data delivery from its generation to its collection by the mobile sink.

We here adopt a discrete-time system where the duration per tour is slotted into equal time slots with each lasting τ time units [57]. Given the mobile sink speed r_s, the number of time slots per tour can be determined, which is $T = \lceil \frac{L}{r_s \cdot \tau} \rceil$, where L is the length of the pre-defined path. We index the T time slots by $1, 2, \cdots, T$. Let $A(v)$ represent the set of consecutive time slots in which the data transmitted by sensor $v \in V$ can be collected by the mobile sink. Then, $A(v)$ will be determined by the maximum transmission range R of v and its distance from the pre-defined path. Fig. 3.1 uses an example to illustrate this concept. Given two sensors v_i and v_j, then $A(v_i) = \{i_s, i_s + 1, \cdots, i_e - 1, i_e\}$ and $A(v_j) = \{j_s, j_s + 1, \cdots, j_e - 1, j_e\}$ are the sets of time slots in which they can transmit their data to the mobile sink, $1 \leq i_s \leq i_e \leq T$ and $1 \leq j_s \leq j_e \leq T$. Notice that if $A(v_i) \cap A(v_j) \neq \emptyset$, they share some time slots at which they both can transmit their data to the mobile sink. However, following wireless communication interference model [96], the mobile sink at any given time slot can receive the data from one sensor only; otherwise, none of the transmitted data can be received by the mobile sink due to the channel interference. Thus, we need to allocate these time slots to the sensors such that each time slot is allocated to one sensor only with an objective to maximize the amount of data collected by the mobile sink.

3.3.1 Energy Budget Model

As sensors are powered by renewable energy, the amount of energy harvested by a sensor at each different time slot is different. This implies that a sensor cannot transmit its data to the mobile sink without any restriction. In principle, a given sensor

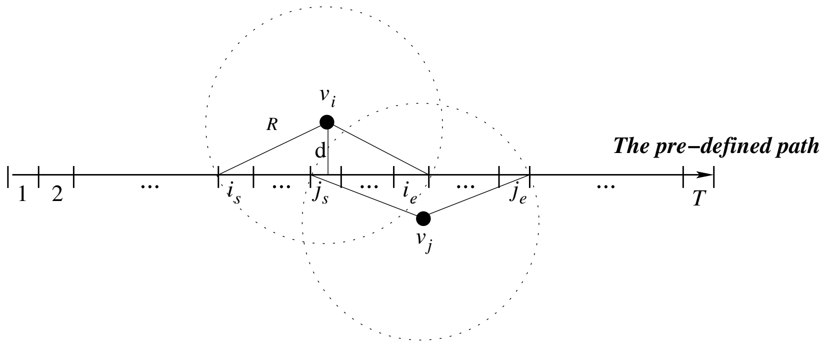


Figure 3.1: An illustration of time slots covered by sensors v_i and v_j.

v can transmit its data to the mobile sink in all time slots in $A(v)$ if it has sufficient energy to support it doing so. However, it may not have enough energy at this moment to achieve that. We here adopt a similar energy budget model mentioned in chapter 2.3. Denote by $B(v)$ the battery capacity of each sensor v, and denote by $P_j(v)$ and $RE_j(v)$ the amounts of available energy of node v prior to and after tour j, respectively. Thus, sensor v consumes the amount of energy $P_j(v) - RE_j(v)$ for transmitting its data to the mobile sink in tour j. Let $Q_j(v)$ be the amount of harvested energy of sensor v between the $(j-1)$th tour and the jth tour, $P_j(v)$ thus can be expressed as $\min\{RE_{j-1}(v) + Q_j(v), B(v)\}$, where $0 \leq P_j(v) \leq B(v)$. Furthermore, to support long-term, continuous monitoring service, we assume that sensors should not consume more energy than they can collect in order to achieve 'perpetual' operations [44]. Hence, without loss of generality, we refer to $P_j(v)$ as the *energy budget of sensor v* at tour j. We also refer to $P(v)$ as the energy budget of sensor v per tour.

3.3.2 Energy Consumption Model

It is known that wireless signals suffer from path loss, fading, shadowing, interference and other impairments, and the communication performance is determined by the received Signal to Noise Ratio (SNR). Hence reliability and efficiency are often at odds with each other. Reliability can be improved by transmitting packets at the maximum transmission power. However, this introduces unnecessarily high energy consumption. Motivated by the fact that popular radio hardware such as CC2500 RF

Transceiver [90] not only allows setting multiple transmission power levels, but also allows setting multiple transmission rates, a novel communication model between sensors and the mobile sink is adopted [2, 46]. That is, each sensor has multiple transmission power levels and transmission rate levels. Specifically, for each sensor v_i, a transmission rate out of $\mathcal{R} = \{R_1, R_2, \cdots\}$ pre-defined rate levels needs to be adopted for transmission. Let $\mathcal{P} = \{P_1, P_2, \cdots\}$ be the set of transmission power levels which sensor v_i can select when transmitting. Given a time slot (i.e., the location of the mobile sink is given), for each transmission power level and rate level, this 'power-rate' pair could be adopted by sensor v_i at this time slot if the Shannon channel capacity [81] is larger than the transmission rate. Thus, there may exist multiple transmission 'power-rate' pairs which could be adopted by sensor v_i at this time slot. Sensor v_i will adopt one 'power-rate' pair by its PHY/MAC layer protocols, which is out of the focus of our work. In the rest of discussions, we thus assume that the relevant transmission power $P_{i,j}$ and transmission rate $r_{i,j}$ are given when sensor v_i transmits data to the mobile sink at time slot j.

3.3.3 Approximation Algorithm

We say an algorithm for a maximization optimization problem is an α-approximation algorithm if the ratio of the approximate solution to the optimal solution is no less than α, where α is a constant with $0 < \alpha < 1$.

3.3.4 Problem Statement

Given a renewable sensor network G and T time slots per tour in which the mobile sink travels along with a pre-defined path to collect data from one-hop sensors, the *data collection maximization problem* is to maximize the volume of the data collected by the mobile sink through allocating the T time slots to individual sensors, under the constraints on both the energy replenishment rate and multi-rate data transmission rate at each time slot.

Intuitively, each sensor should transmit its data to the mobile sink at all available time slots to it in order to maximize its share on the collected data, thereby maximizing the volume of the data collected from the entire network. However, since the energy replenishment rate of each sensor is much slower than its energy consumption rate, each sensor may only make use of some of all available time slots to transmit its data subject to its energy budget. What followed is which time slots it should choose for its data transmission. Since the sensor at different time slots will have different data transmission rates, this results in different amounts of its transmission energy consumption. Furthermore, it is very likely that multiple sensors sharing the same time slot will compete with each other for the time slot to transmit their own data, as sensors in the network are densely deployed. Thus, allocating each shared time slot to one of the competing sensors so as to maximize the accumulative data volume is a challenging task.

In other words, the data collection maximization problem in G can be described as follows. Given T time slots and a pre-defined path, the mobile sink travels along the path at a given constant speed to collect data from one-hop sensors. Associated with each sensor $v_i \in V$, there are $|A(v_i)|$ potentially available time slots for sensor v_i to transfer its data to the mobile sink, where $r_{i,j}$ is the average data transmission rate of v_i if it transmits data at time slot $j \in A(v_i)$ with the amount of energy consumption $P_{i,j} \cdot \tau$. We further assume that the number of different transmission rates of each sensor v_i, $r'_{i,1}, r'_{i,2}, \ldots, r'_{i,k_i}$, is given and $r'_{i,x} < r'_{i,y}$ if $1 \leq x < y \leq k_i$. Usually, k_i is a fixed integer. To ensure that the transmitted data can be received by the receiver successfully, the use of a different transmission rate for data transmission will consume a different amount of power of sensor v_i. For the sake of convenience, in the rest of the chapter we assume that all sensors have the same number of transmission power levels k, i.e., $k_i = k$ for all i with $1 \leq i \leq n$. Also, it is well known that wireless communication is unreliable. In this chapter we thus assume that the *link reliability* of the link between sensor v_i and the mobile sink at time slot j is $\rho_{i,j}$ with $0 \leq \rho_{i,j} \leq 1$ and $1 \leq j \leq T$. The data collection maximization problem in G thus is to allocate a subset of time slots $A'(v_i)$ ($\subseteq A(v_i)$) to the sensors such that the volume of data transmitted from all sensors, $\sum_{v_i \in V} \sum_{j \in A'(v_i)} (\rho_{i,j} \cdot r_{i,j} \cdot \tau)$ is maximized, subject to (i) each time slot is allocated to only one sensor if there are multiple sensors sharing the time slot; (ii) the total energy consumption of each sensor v_i per tour is no more than its energy budget $P(v_i)$, i.e., $\sum_{j \in A'(v_i)} P_{i,j} \cdot \tau \leq P(v_i)$, where τ is the duration of each time slot and $A'(v_i) \subseteq A(v_i)$ for all i with $1 \leq i \leq n$.

3.3.5 NP-Hardness

We show that the data collection maximization problem is NP-hard by the following theorem.

Theorem 4 *The data collection maximization problem in a renewable sensor network is NP-hard.*

Proof We show the claim by a reduction from a well known NP-complete problem - the generalized assignment problem (GAP), which is defined as follows. Given a set of bins and a set of items that have a different size and profit for each bin, pack a maximum profit subset of items into the bins. In other words, let $A = \{a_1, a_2, \cdots a_m\}$ be a set of m items and $B = \{B_1, B_2, \cdots B_n\}$ a set of bins, where each B_i has a capacity b_i for all i with $1 \leq i \leq n$. Assigning item a_j to bin B_i will consume the amount of resource $b_{i,j}$ of B_i, and the benefit brought by this assignment is $c_{i,j}$. The objective is to allocate the items in A to the bins in B such that the total profit is maximized, subject to the total amount of resources consumed of each bin B_i being no more than its capacity b_i for all i with $1 \leq i \leq n$.

We now show that a special case of the data collection maximization problem can be reduced from the defined GAP problem. The data collection maximization problem is given as follows: we assume that the maximum transmission range of each sensor R is large enough to cover the entire tour path, and wireless communication is reliable, i.e., the link reliability of each link is 1. We proceed with the following reduction.

Each item in A corresponds a time slot, thus the set of items corresponds to the set of time slots. Each bin B_i in B corresponds to a sensor $v_i \in V$, the capacity b_i of B_i corresponds to the energy budget of sensor v_i, $P(v_i)$, to perform its data transmission for a certain number of time slots in $A(v_i)$, and $P_{i,j} \cdot \tau$ is the amount of transmission energy consumed by v_i if it sends its data to the mobile sink at time slot a_j, i.e., the amount of its resource consumed. The profit brought by allocating time slot a_j to sensor v_i is $c_{i,j}$ ($= r_{i,j} \cdot \tau$), which is the amount of data transmitted, where $r_{i,j}$ is the average data transmission rate of v_i at time slot a_j, which usually is determined by the Euclidean distance $d_{i,j}$ between v_i and the mobile sink at time slot a_j and the transmission power adopted by v_i. This implies that at different time slots, different data transmission rates will be adopted, thereby leading to different amounts of data collected by the mobile sink. Allocating the T time slots to the n sensors such that the amount of data collected by the mobile sink is maximized is equivalent to maximizing the profit in the GAP. Hence, the data collection maximization problem is NP-hard. □

3.4 Offline Approximation Algorithm

Since the data collection maximization problem is NP-hard, in this section we devise an approximation algorithm with a provable approximation ratio for the problem, by exploiting the combinatorial property of the problem, provided that the mobile sink has the global knowledge of the network topology and the profile of each sensor (e.g., the energy budget of each sensor at the current tour, the location of the sensor, the starting and ending time slots of the sensor, etc).

For the sake of convenience, in the following we first deal with the data collection maximization problem under the assumptions that the amount of available energy at each sensor for the current mobile sink tour is given and all links are reliable. We then show how to extend the proposed solution with minor modifications to the problem without the specified assumptions.

3.4.1 Algorithm

Cohen et al. [20] proposed a local search algorithm for the generalized assignment problem (GAP). We show how to adopt their algorithm to the data collection maximization problem by necessary modifications, as we have already shown that the data collection maximization problem is equivalent to GAP.

The technique they adopted is based on a novel combinatorial translation of any (exact or approximation) algorithm for the knapsack problem into an approximation algorithm for GAP. Thus, any β-approximation algorithm for the knapsack problem can be transformed into a $\frac{\beta}{1+\beta}$-approximation algorithm for GAP, where β is a constant with $0 < \beta < 1$. The theoretical foundation of their technique is based a local-ratio theorem [5]. Specifically, the Cohen et al. [20] algorithm proceeds iteratively. It essentially decomposes the profit function into two profit functions: one is used for the current bin packing; and another is used for the rest of bin packing. The initial profit matrix is defined as follows.

$$D_{i,j}^{(0)} = \begin{cases} r_{i,j} \cdot \tau & \text{if time slot } j \in A(v_i) \\ 0 & \text{otherwise.} \end{cases} \tag{3.1}$$

Within iteration l, it packs items in $A(v_l)$ into bin B_l, using the profit function $D_{i,j}^{(l)}$, i.e., it packs time slots $j \in A(v_l)$ to sensor v_l, based on the profit entries of row l in $D_{i,j}^{(l)}$, subject to the capacity constraint $P(v_l)$ of sensor v_l.

Let $\overline{S_l}$ be the set of time slots allocated to sensor v_l by a β-approximation algorithm for the knapsack problem, clearly $\overline{S_l} \subseteq A(v_l)$. Then, the profit function $D_{i,j}^{(l)}$ is decomposed into two profit functions $D_{i,j}^{(l+1)}$ and $T_{i,j}^{(l+1)}$ as follows.

$$D_{i,j}^{(l+1)} = \begin{cases} D_{l,j}^{(l)} & \text{if time slot } j \in \overline{S_l} \text{ or } i = l \\ 0 & \text{otherwise.} \end{cases} \tag{3.2}$$

and

$$T_{i,j}^{(l+1)} = D_{i,j}^{(l)} - D_{i,j}^{(l+1)}. \tag{3.3}$$

The decomposition of the profit function implies that $D_{i,j}^{(l+1)}$ is identical to $D_{i,j}^{(l)}$ with regard to bin B_l. In addition, if time slot $j \in \overline{S_l}$, then it is allocated in $D_{i,j}^{(l+1)}$ the same profit as that in $D_{i,j}^{(l)}$ for all bins l' if $j \in A(v_{l'})$. All other entries are zeros. The new profit function for bin B_{l+1}, $D_{i,j}^{(l+1)}$ then is $T_{i,j}^{(l+1)}$, i.e.,

$$D_{i,j}^{(l+1)} = T_{i,j}^{(l+1)}. \tag{3.4}$$

The procedure continues until the last bin B_n is packed. Then, an approximate solution to the data collection maximization problem finally is derived. That is, let S_l be the set of time slots allocated to sensor v_l. If $l = n$, then $S_n = \overline{S_n}$; otherwise, the set of time slots allocated to sensor v_l is $S_l = \overline{S_l} \setminus \cup_{j=l+1}^{n} S_j$ for all l with $1 \leq l \leq n-1$.

The offline approximation algorithm for the data collection maximization problem is thus as follows.

Algorithm 5 `Offline_Appro`

Input: The number of time slots T, the set of sensors V, the energy budget $P(v_i)$ and the set of available time slots $A(v_i)$, the transmission rate $r_{i,j}$ and the corresponding energy consumption $P_{i,j}$ of each sensor $v_i \in V$, and the profit matrix $D_{i,j}^{(0)}$ for all i and j with $1 \leq i \leq n$ and $1 \leq j \leq T$.

Output: Allocate T time slots to the n sensors.

Sort all sensors in increasing order of the indices of their starting time slots, followed by their ending time slots. Let $v_1, v_2, \ldots, v_n$ be the sorted sensor sequence;

Profit matrix's initialization: $D_{i,j}^{(1)} \leftarrow D_{i,j}^{(0)}$ for all i and j with $1 \leq i \leq n$ and $1 \leq j \leq T$;

for $l \leftarrow 1$ to n **do**

  /* Assume that $A(v_l) = \{l_s, \cdots l_e\}$ */

  Apply a β-approximation algorithm for a single bin packing (knapsack problem) to allocate time slots in $A(v_l)$ to sensor v_l, subject to the capacity of v_l, $P(v_l)$, using the profit function $D_{i,j}^{(l)}$, i.e., the entries in row l of the matrix. Let $\overline{S_l}$ be the result delivered by the approximation algorithm to sensor v_l, where $\overline{S_l} \subseteq A(v_l)$;

  /* Decompose the profit function into two profit functions $D_{i,j}^{(l+1)}$ and $T_{i,j}^{(l+1)}$ */

  $D_{i,j}^{(l+1)} \leftarrow T_{i,j}^{(l+1)}$;

end for

/* construct a solution to the time slot allocation */

$S_n \leftarrow \overline{S_n}$;

for $l \leftarrow n-1$ downto 1 **do**

  $S_l \leftarrow \overline{S_l} \setminus \cup_{j=l+1}^{n} S_j$;

end for

return S_l for all l with $1 \leq l \leq n$.

Initially, we sort the sensors in increasing order of indices of their starting time slots, (i.e., the index of the first time slot in $A(v_i)$ for sensor v_i). If there are multiple sensors with the same starting time slot, then sort them in increasing order of indices of their ending time slots. In case the indices of these ending time slots are also identical, the tie between the sensors will be broken arbitrarily. Without loss of generalization, assume that $v_1, v_2, \ldots v_n$ is the sorted sensor sequence starting from time slot indexed by 1, and the mobile sink starts its data collection tour from the first time slot. The detailed offline approximation algorithm `Offline_Appro` is presented in `Algorithm` 5.

3.4.2 Complexity Analysis

Theorem 5 *Given a renewable sensor network $G = (V \cup \{s\}, E)$ with $n = |V|$, there is an approximation algorithm for the data collection maximization problem with an approximation ratio of $\frac{1}{2+\epsilon}$. The time complexity of the proposed approximation algorithm is $O(n^2)$.*

Proof Cohen *et al.* [20] have showed that algorithm `Offline_Appro` is a $\frac{\beta}{1+\beta}$-approximation algorithm, where β is the approximation ratio of an approximation algorithm for the single knapsack problem with $0 < \beta < 1$. Obviously, the approximation ratio of the approximation algorithm is $\beta = \frac{1}{1+\epsilon}$ [50], where ϵ is a constant with $0 < \epsilon < 1$, and it takes $O(|A(v_l)| \log \frac{1}{\epsilon} + \frac{1}{\epsilon^4}) = O(t_{max})$ time to find the subset $\overline{S_l}$ ($\subseteq A(v_l)$), where $t_{max} = \max\{|A(v)| \mid v \in V\}$. The updating of profit matrices $D_{i,j}^{(l)}$ and $T_{i,j}^{(l)}$ also takes time. However, it is noticed that there is no need to update all entries. We only need to update the entries in row l and the related columns $j \in \overline{S}_l$, thus, it takes $O(|A(v_l)| + \sum_{j \in \overline{S_l}} O(n)) = O(|A(v_l)| + O(n \cdot |\overline{S_l}|)) = O(nt_{max})$ time. The running time of allocating all time slots into the n sensors therefore is $\sum_{v_l \in V} O(t_{max} + nt_{max}) = O(nt_{max} + n^2 t_{max}) = O(n\Gamma + n^2\Gamma) = O(n^2)$ since $t_{max} \leq 2\Gamma$ and $\Gamma = \lfloor \frac{R}{r_s \cdot \tau} \rfloor$ usually is a constant in practice, where R is the maximum transmission range of sensors and r_s is the travelling speed of the mobile sink. The approximation ratio of the approximation algorithm for the data collection maximization problem thus is $\frac{\beta}{1+\beta} = \frac{1}{2+\epsilon}$. $\square$

3.4.3 Harvested Energy Estimation and Unreliable Link Reliability

The proposed approximation algorithm, `Algorithm` 5, is proposed, under the assumptions that the energy budget $P(v_i)$ of each sensor $v_i \in V$ is given and the link reliability of each link $\rho_{i,j}$ between sensor v_i and the mobile sink at each time slot j is reliable (i.e., $\rho_{i,j} = 1$) for all $v_i \in V$ and all $j \in A(v_i)$. In reality, the battery energy information $P(v_i)$ at each sensor v_i is not known, and the wireless communication between a sensor and the mobile sink is error-prone and not always noise free, thus interferences are not avoidable. Therefore, both harvested energy predictions and unreliable link reliability must be taken into account when dealing with the design of real protocols for renewable sensor networks. In this subsection we show how to extend the proposed algorithm for this general case.

Assume that the mobile sink starts its tour t. We take the predicted energy budget $\hat{P}_t(v_i)$ and link reliability $\rho_{i,j}$ of each sensor into consideration when the mobile sink performs its next tour t, i.e., when the mobile sink performs packing time slots in $A(v_i)$ to bin v_i with the estimated energy budget constraint $\hat{P}_t(v_i)$ and the link reliability $\rho_{i,j}$ for all $j \in A(v_i)$, the β-approximation algorithm for the knapsack problem with reliable link reliability can still be applied to this general setting through a minor modification. That is, the profit brought by allocating time slot j to sensor v_i now is $D^{(0)}_{i,j} = \rho_{i,j} \cdot r_{i,j} \cdot \tau$, not the original $r_{i,j} \cdot \tau$, when sensor v_i consumes the amount of energy $P_{i,j}$ to transmit data at time slot j with link reliability $\rho_{i,j}$. The rest is almost identical to the proposed algorithm, `Algorithm` 5, omitted.

3.5 Online Distributed Algorithm

In the previous section we provided an offline approximation algorithm with a provable approximation ratio for the data collection maximization problem. However, the solution obtained is based the assumptions that the global knowledge of the network topology and the profiles of sensors including their physical locations, energy budgets, starting and ending time slots are available. In reality, there is no way for the mobile sink to know the profile of each sensor unless it is within the transmission range of the sensor. Also, even if the mobile sink is able to collect the topological information of the entire network and the profiles of sensors at its previous tours, using the piggybacking strategy or linear regression prediction, it then performs time slot scheduling based on the collected information, the solution obtained however may not be applicable due to the fact that both the energy harvesting and the link reliability profiles of some sensors may have experienced drastic changes over the period of the mobile sink tour. In this section we will develop a fast, scalable online distributed algorithm for the problem without the mentioned assumptions. For the sake of discussion convenience, we first assume that all links are reliable, i.e., the link reliability of each link is one. We then extend the distributed solution to the unreliable link case through minor modifications.

3.5.1 Overview of the Distributed Algorithm

The overview of the proposed online distributed algorithm proceeds as follows. The mobile sink periodically broadcasts a 'Probe' message with a 'Registration' timer, announcing its presence once per time interval when it travels along the pre-defined path, where each *time interval* consists of $\Gamma = \lfloor \frac{R}{\tau \cdot r_s} \rfloor$ time slots. The 'Probe' message is broadcast in the beginning of each interval, which will be used to detect whether the mobile sink and the sensors are within the transmission range of each other. Each sensor receiving the 'Probe' message will send the mobile sink back an 'Ack' message which contains its current power level, the indices of its starting and ending time slots, its location coordinate, its link reliability, etc. The sensor then enters the waiting status to get the reply from the mobile sink when performing its next action. Once the 'Registration' timer expires, the mobile sink starts scheduling the Γ time slots to the registered sensors, using a time-slot scheduling algorithm $\mathcal{A}$ which will be detailed later. It finally broadcasts the time-slot allocation results to the registered sensors. Each registered sensor (in the waiting status) then sets its own scheduling, i.e., in which time slots it will transmits its data to the mobile sink.

Within the rest of the current time interval, each registered sensor transmits its data to the mobile sink at its allocated time slots. For the sake of simplicity, we here assume that the time spent by the mobile sink in probing and time slot scheduling is negligible in comparison with the time at each time slot for data transmission.

When the mobile sink receives the data from the sensor at the last time slot in the current time interval, it sends a 'Finish' message to all the registered sensors. The registered sensors then update their own energy profiles after having received the 'Finish' message and wait for their scheduling in the next time interval. This procedure continues until there is no response from any sensor to the 'Probe' message sent by the mobile sink in some time interval, which means that the mobile sink finishes the tour already, as we assumed that the sensors are densely deployed along the pre-defined path and there is at least one sensor at each time interval. The detailed online distributed algorithm is given in `Algorithm` 6 and `Algorithm` 7.

Algorithm 6 `Distributed_Algorithm (the mobile sink)`

continue $\leftarrow$ '*true*';/*the current tour finishes or not*/
$j \leftarrow 0$; /* the number of time intervals per tour */
while *continue* **do**
    $j \leftarrow j+1$; /* The current time interval j*/
    Mobile sink broadcasts a 'Probe' message with a 'Registration' timer to one-hop sensors;
    if the timer expires **then**
        if the mobile sink received 'Ack' messages from sensors **then**
            Call a time-slot scheduling algorithm, `Algorithm` $\mathcal{A}$, in the mobile sink to allocate the time slots in time interval t to the registered sensors, subject to the power constraint on each registered sensor;
            The mobile sink broadcasts the scheduled results to sensors in the network;
            /*Each registered sensor performs data transmissions in its allocated time-slots; */
            The mobile sink broadcasts a 'Finish' message to sensors when it finished the data collection from the last time slot in time interval j;
            /* The registered sensors update their energy profiles when they received the 'Finish' messages. That is, each registered sensor v_i updates its power: $P_j(v_i) \leftarrow P_j(v_i) - \sum_{j' \in S_i} \cdot P_{i,j'} \cdot \tau$, where S_i is the set of time slots assigned to v_i by algorithm $\mathcal{A}$ in the current time interval j and $S_i \subseteq A(v_i)$; */
        else
            continue $\leftarrow$ '*false*'; /* finish the tour */
        end if
    else
        Waiting for the replies from one-hop sensors;
    end if
end while

3.5.2 GAP-based Time Slot Scheduling

In the rest we devise a *GAP-based time-slot scheduling algorithm* as `Algorithm` $\mathcal{A}$ in algorithm 6. Recall that the starting and ending time slots of sensor $v_i \in V$ are the i_sth and the i_eth time slots, denote by $[i_s, i_e]$ the time slot interval in which sensor

Algorithm 7 `Distributed_Algorithm (sensor node` v_i`)`

1: At each time slot, sensor node $v_i \in V$ performs its data collection based on its duty-cycling;
2: When it receives a 'Probe' message from the mobile sink, it responds by sending back of an 'Ack' message that includes its current energy $P(v_i)$ and link reliability in the last tour $\rho_{i,j}$, and waiting for the reply from the mobile sink;
3: When it receives the time-slot allocation information from the mobile sink, set its time-slot scheduling, and perform data transmission in its allocated time slots;
4: When it receives a 'Finish' message from the mobile sink, it updates its energy budget for next time interval.

v_i can transmit its data to the mobile sink. Given the current time interval j, $[a_j, b_j]$ where a_j and b_j are the starting and ending time slots in the current time interval, then $|b_j - a_j| = \lfloor \frac{R}{r_s \cdot \tau} \rfloor$. If $[i_s, i_e] \cap [a_j, b_j] \neq \emptyset$, then sensor v_i can transmit its data to the mobile sink in time interval j within time slot interval $[i'_s, i'_e] = [i_s, i_e] \cap [a_j, b_j]$ with $i_s \leq i'_s$ and $i'_e \leq i_e$. Let $P_j(v_i)$ be the amount of power of sensor v_i in the beginning of time interval j, then it consumes the amount of energy $P_{i,j'} \cdot \tau$ when sensor v_i transmits its data in a time slot $j' \in [i'_s, i'_e]$. It may transmit its data within multiple time slots as long as its residual energy enables itself to do so. The mobile sink schedules the current Γ time slots to these registered sensors in the current time interval, using the offline approximation algorithm. This GAP-based algorithm is described in `Algorithm` 8.

Algorithm 8 `GAP-based_Time-slot_Scheduling` at time interval j

1: Let a_j and b_j be the starting and ending indices of time slots in time interval j, let TS_j be the set of sensor nodes responded to the 'Probe' request message issued by the mobile sink;
2: Let $v_i \in TS_j$ and i_s and i_e be the starting and ending indices of time slots of v_i in $[a_j, b_j]$. That is, let $A'(v_i) = \{i_s, i_s + 1, \ldots, i_e\} \subseteq A(v_i)$ be the subset of time slots of v_i in time interval j;
3: **for** each $v_i \in TS_j$ **do**
4: Apply an β-approximation algorithm for bin packing to pack time slots in $A'(v_i)$ to sensor v_i with the bin capacity $P_j(v_i)$;
5: Let S_i be the set of allocated time slots to sensor v_i, i.e., $S_i \subseteq A'(v_i) \subseteq A(v_i)$;
6: **end for**

We then have the following lemma and theorem.

Lemma 2 *Given the sensor network $G = (V \cup \{s\}, E)$, following the proposed online dis-*

tributed algorithm, `Algorithm` *6 and* `Algorithm` *7, we claim that each sensor is within at most two consecutive broadcasting regions (or two consecutive time intervals).*

Proof We show the claim by contradiction. Considering Fig. 3.2, assume that a

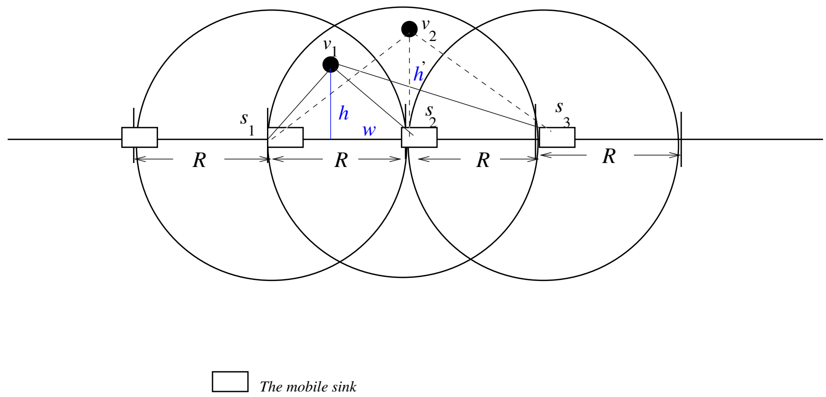


Figure 3.2: A sensor v_1 (or v_2) cannot be in three consecutive time intervals.

sensor v_1 is within three consecutive 'Probe' message broadcasting regions, i.e., when the mobile sink broadcasts its probing messages at s_1, s_2, and s_3 locations, sensor v_1 is able to receive the message three times. Following this assumption, we have $d(v_1, s_1) \leq R$, $d(v_1, s_2) \leq R$, and $d(v_1, s_3) \leq R$. We now show that this is impossible by the following three cases:

Case one: sensor v_1 is in the left side of s_2, then $d(v_1, s_3) = \sqrt{h^2 + (w+R)^2} > \sqrt{R^2} = R$, which contradicts the fact that $d(v_1, s_3) \leq R$.

Case two: sensor v_1 is in the right side of s_2, the proof is similar to Case one, omitted.

Case three: sensor v_1 (i.e., sensor v_2) is just above s_2, then $d(v_2, s_1) = \sqrt{h'^2 + R^2} > \sqrt{R^2} = R$ and $d(v_2, s_3) = \sqrt{h'^2 + R^2} > \sqrt{R^2} = R$. This contradicts that v_2 is in the transmission ranges of s_1 and s_3. $\square$

Theorem 6 *Given a renewable sensor network $G = (V \cup \{s\}, E)$ with $|V| = n$, there is an online distributed algorithm for the data collection maximization problem in G, which takes $O(n)$ time and $O(n)$ messages.*

Proof Following Lemma 2, we notice that each sensor can receive the probing message and the finish message from the mobile sink at most twice per tour, and these messages are issued in two consecutive time intervals. Thus, the total number of probing and finish messages and the time slot allocation messages received by each sensor are four, respectively per tour of the mobile sink, while the number of acknowledgement messages by each sensor is two as well. Thus, the total number of messages transmitted per tour is $O(\sum_{v \in V} d_v) = O(n)$ as each sensor v has $O(d_v) = O(1)$ messages to be received and/or sent out. Clearly, the time for time-slot scheduling by the mobile sink in each interval j is $\sum_{l=1}^{N_j} O(t_{max} \log t_{max}) = O(N_j \cdot t_{max} \log t_{max})$ as sorting by the mobile sink for bin packing at each sensor in this interval takes $O(t_{max} \log t_{max})$ time, and the rest operations takes constant time, where N_j is the number of registered sensors in interval j and $t_{max} = \max_{v \in V}\{|A(v)|\}$. Thus, the time complexity of the online distributed algorithm is proportional to the number of time intervals per tour. As we assume that sensors are densely deployed, this implies that there is at least one sensor responded to each probing request in the beginning of each time interval, while each sensor is included in at most two consecutive time intervals by Lemma 2. Assume that there are K intervals of each tour, then $\sum_{j=1}^{K} N_j \leq 2n$. Thus, the time complexity of the online distributed algorithm is $\sum_{j=1}^{K} O(N_j \cdot t_{max} \log t_{max}) = O(n t_{max} \log t_{max}) = O(n\Gamma \log \Gamma) = O(n)$ as $t_{max} \leq 2\Gamma$ and $\Gamma = \lfloor \frac{R}{r_s \cdot \tau} \rfloor$ usually is a constant in practice, where R is the maximum transmission range of sensors and r_s is the travelling speed of the mobile sink. □

3.5.3 Unreliable Wireless Communication

The proposed online distributed algorithm is built upon the assumption that communications between sensors and the mobile sink are reliable. We now remove this assumption by dealing with a general case where wireless communications are not reliable, for which we will adopt the similar strategy as we did for the offline approximation algorithm. That is, within each time interval, when the mobile sink broadcasts a 'Probe' message, a responding sensor v_i receiving the 'Probe' message will respond by sending an 'Ack' message back to the mobile sink. The Ack message contains not only the current harvested energy $P(v_i)$ of v_i but also its link reliability $\rho_{i,j'}$ in the previous tour for each time slot $j' \in A(v_i)$. The mobile sink then proceeds a time-slot scheduling in this time interval, based on sensor energy budget and the estimation of link reliability. In terms of time slot allocation to a responded sensor, the energy consumption of the sensor by transmitting its data at any given time slot should incorporate its re-transmission energy consumptions (the link reliability). The rest operations are identical to the case for the perfect channel condition, omitted.

3.6 A Special Data Collection Maximization Problem

In this section we deal with a special case of the data collection maximization problem where each sensor $v_i \in V$ has only one fixed transmission power level with power P'_i. For this special case, we devise a fast, scalable online distributed algorithm for the problem as follows.

We reduce this special data collection maximization problem to the maximum weight matching problem in another auxiliary, bipartite graph $G = (X \cup Y, E_{XY})$, where X is the set of sensors which acknowledged the probing message by the mobile sink in the beginning of time interval j, Y is the set of Γ time slots to be allocated to the registered sensors in X. There is an edge between a sensor node v_i that corresponds to a node $x_i \in X$ and a time slot node $y_j \in Y$ if $y_j \in [i'_s, i'_e]$, i.e., y_j is a time slot in interval $[i'_s, i'_e]$. There are $m_i = |i'_s - i'_e| + 1$ edges incident to node x_i in G. The weight associated with edge $(x_i, y_j) \in E_{XY}$ is the average amount of data received by the mobile sink from sensor v_i at time slot y_j, $D^{(0)}_{i,j} = r_{i,j} \cdot \rho_{i,j} \cdot \tau$, where the average data transmission rate $r_{i,j}$ of sensor v_i at time slot y_j is determined by the distance between sensor v_i and the mobile sink at time slot y_j. Our objective thus is to maximize the data collected by the mobile sink in the current time interval through the time slot allocation. In terms of time slot allocation, we notice that each registered sensor v_i in the current time interval can make use of upto $n_i = |A(v_i)|$ time slots to transmit its data. Meanwhile, it is very likely that there are multiple sensors to compete with each other for each shared time slot to transmit their own data. The challenge thus is how to allocate these time slots to the registered sensors such that the sum of amounts of data transmitted is maximized. In the following we propose a solution to this special data collection maximization problem by reducing it to a maximum weight matching problem in another bipartite graph $G' = (\{x_i^{(k)} \mid x_i \in X, 1 \leq k \leq n'_i\} \cup Y, E')$, where G' is derived from the bipartite graph G as follows.

For each node $x_i \in X$ in G, there are n'_i corresponding node copies, $x_i^{(1)}, x_i^{(2)}, \ldots, x_i^{(n'_i)}$ in G', where $n'_i = \min\{\lfloor \frac{R}{r_s \cdot \tau} \rfloor, |i'_s - i'_e| + 1, \lfloor P(v_i)/(P'_i \cdot \tau) \rfloor\}$, where P'_i is the fixed transmission power of sensor v_i, and $P'_i \cdot \tau$ is the amount of energy needed by sensor v_i to transmit a message in a time slot. For each an edge $(x_i, y_j) \in E_{XY}$ in G, there are n'_i corresponding edge copies $(x_i^{(1)}, y_j), (x_i^{(2)}, y_j), \ldots, (x_i^{(n'_i)}, y_j)$ in E', and each of them has a weight $D_{i,j}^{(0)}$. Then, finding a solution of allocating the Γ time slots to the registered sensors such that the amount of data collected by the mobile sink in this time interval is maximized is equivalent to finding a maximum weight matching in G' such that the weighted sum of matched edges is maximized.

Let M be such a maximum weight matching in G'. Then, M corresponds to a time-slot allocation. That is, each edge $(x_i^{(k)}, y_j)$ in M implies that time slot y_j is allocated to sensor v_i, and sensor v_i will successfully transmit its data with the data transmission rate $r_{i,j}$ to the mobile sink. We refer to this online distributed algorithm as `Online_MaxMatch`, and have the following theorem.

Theorem 7 *Given a renewable unreliable sensor network $G = (V \cup \{s\}, E)$ with $|V| = n$, there is an online maximum weight matching-based distributed algorithm for a special data collection maximization problem in G where there is only one fixed transmission power for each sensor. The proposed distributed algorithm takes $O(n^{1.5})$ time and $O(n)$ messages.*

Proof The analysis of time complexity and message complexity of the proposed online distributed algorithm are almost identical to the ones in Theorem 6. The rest will focus on the analysis of time complexity of the operations in each time interval. Let N_j be the number of registered sensors in time interval j. Then, the bipartite graph G' contains $O(N_j \cdot t_{max} + \Gamma)$ nodes and $O((N_j \cdot t_{max}) \cdot \Gamma)$ edges, while it takes $O(\sqrt{|V|} \cdot |E|)$ time to find a maximum weight matching in a bipartite graph $G = (V, E)$ [71]. Thus, it takes $O(N_j^{1.5} \cdot \Gamma^{2.5}) = O(N_j^{1.5})$ time in G' to find the maximum weight matching M, since $t_{max} \leq 2\Gamma$ and $\Gamma = \lfloor \frac{R}{r_s \cdot \tau} \rfloor$ usually is a constant in practice, where R is the maximum transmission range of sensors and r_s is the travelling speed of the mobile sink. Notice that this maximum weight matching-based time-slot scheduling algorithm is performed by the mobile sink. Assuming that there are K intervals, following Lemma 2, each sensor appears at most twice in two consecutive time intervals, thus, $\sum_{j=1}^{K} N_j \leq 2n$. The total amount of time spent for finding maximum weight matchings in all intervals therefore is $\sum_{j=1}^{K} O(N_j^{1.5}) = O(n^{1.5})$. Considering the fact that N_j usually is bounded by a constant in practice, then the proposed online distributed algorithm takes only $O(n)$ time, and the message complexity is still $O(n)$. $\square$

Notice that if the global knowledge of the entire network and the residual energy and location profiles of all sensors are given, an offline algorithm for the special data collection maximization problem based on the maximum weight matching can also be obtained, and delivers an exact solution in polynomial time. We refer to this offline algorithm as algorithm `Offline_MaxMatch`.

3.7 Performance Study

In this section we first evaluate the accuracy of the energy prediction model. We then study the performance of the proposed algorithms through experimental simulation. We finally investigate the impact of parameters: the network size n, the mobile sink speed r_s, and the duration τ of each time slot on the performance of proposed algorithms.

3.7.1 Experimental Environment Setting

We consider a renewable sensor network consisting of 100 to 400 sensor nodes randomly deployed along two sides of a pre-defined path, and a mobile sink s travels along the path at constant speed r_s. We further assume that the length of the pre-defined path is $10,000m$ and the path is a straight line, and the maximum distance between the location of any sensor and the path is $180m$. Each sensor has an identical maximum transmission range of 200 meters and is powered by a $10mm \times 10mm$

square solar panel with the battery capacity of $10,000 Joules$. The solar power harvesting profile is built upon real solar radiation measurements [63], in which the total amount of energy collected from a $37mm \times 37mm$ solar panel over a 48-hour period is $655.15mWh$ in a sunny day and $313.70mWh$ in a partly cloudy day, respectively. We here adopt a 4-pairwise communication parameters setting where its transmission and corresponding distance parameters are listed in Table 3.1. In the default setting

Table 3.1: List of experimental setting parameters

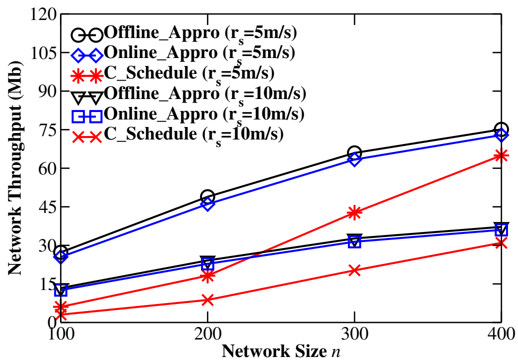
Parameter	**Values**
Number of sensors	100 – 400
Sensor transmission rates $r_{i,j}$ and energy consumption $P_{i,j}$	250 Kbps between 0m–20m at 170 mW 19.2 Kbps between 20m–50m at 220 mW 9.6 Kbps between 50m–120m at 300 mW 4.8 Kbps between 120m–200m at 330 mW
Link reliability $\rho_{i,j}$	[0, 1]
Sensor energy capacity $B(v)$	10,000 Joules
Sink travelling speed r_s	5m/s – 30m/s
Duration of time slot τ	1 sec.–10 sec.

the duration of each time slot τ is 1 second. Each value in figures is the mean of the results obtained by applying each mentioned algorithm to 50 different network topologies of the same network size. Since the deviations of the 50 replication results are minor, for the sake of clarity, we do not provide error bars to indicate their standard deviations. We will adopt an existing offline algorithm `C_Schedule` [78] for a similar data gathering problem as the benchmark, which proceeds to allocate time slots iteratively, starting from time slot 1 and ending at time slot T. Within iteration j, time slot j will be allocated to the sensor with the maximum amount of its data to be transmitted.

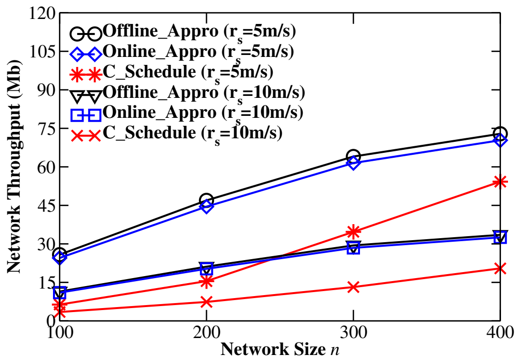
3.7.2 Performance Evaluation of Different Algorithms

We then evaluate the performance of algorithms `Offline_Appro` and `Online_Appro` by varying the network size n from 100 to 400 and setting the mobile sink speed r_s at $5m/s$, and $10m/s$, while the duration of time slot τ is fixed at $1s$, $4s$, and $8s$, respectively.

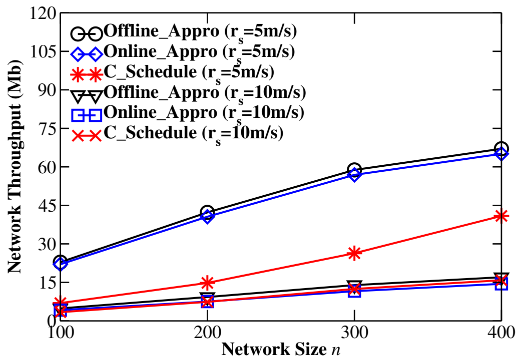
Fig. 3.3 demonstrates that algorithm `Offline_Appro` always outperforms algorithm `Online_Appro` slightly. However, they both outperform the benchmark algorithm `C_Schedule` significantly. For example, when $r_s = 5m/s$ and $\tau = 1s$, the network throughput of algorithm `Online_Appro` is no less than 93% of that of algorithm `Offline_Appro`, while their throughput are no less than from 115% to 400% that of algorithm `C_Schedule`. The reason behind is that algorithm `Online_Appro` only has the local, rather global knowledge of the entire network. It can be also noticed that when the network size is fixed, the longer duration of time slot and the higher mobile sink speed will lead to a lower network throughput of each mentioned algorithm. In other words, to maximize the network throughput, a shorter duration of time slot should be chosen when the mobile sink travels at a faster speed.



(a) $\tau = 1s$



(b) $\tau = 4s$



(c) $\tau = 8s$

Figure 3.3: Network throughput delivered by algorithms `Offline_Appro`, `Online_Appro`, and `C_Schedule` through varying the sink speed r_s and the network size n when all links are reliable.

Fig. 3.4 shows that the network throughput of the three mentioned algorithms

Offline_Appro, Online_Appro and C_Schedule drop down significantly when varying the link reliability between 0 and 1 randomly in comparison with their counterparts in the link reliability case in Fig. 3.3.

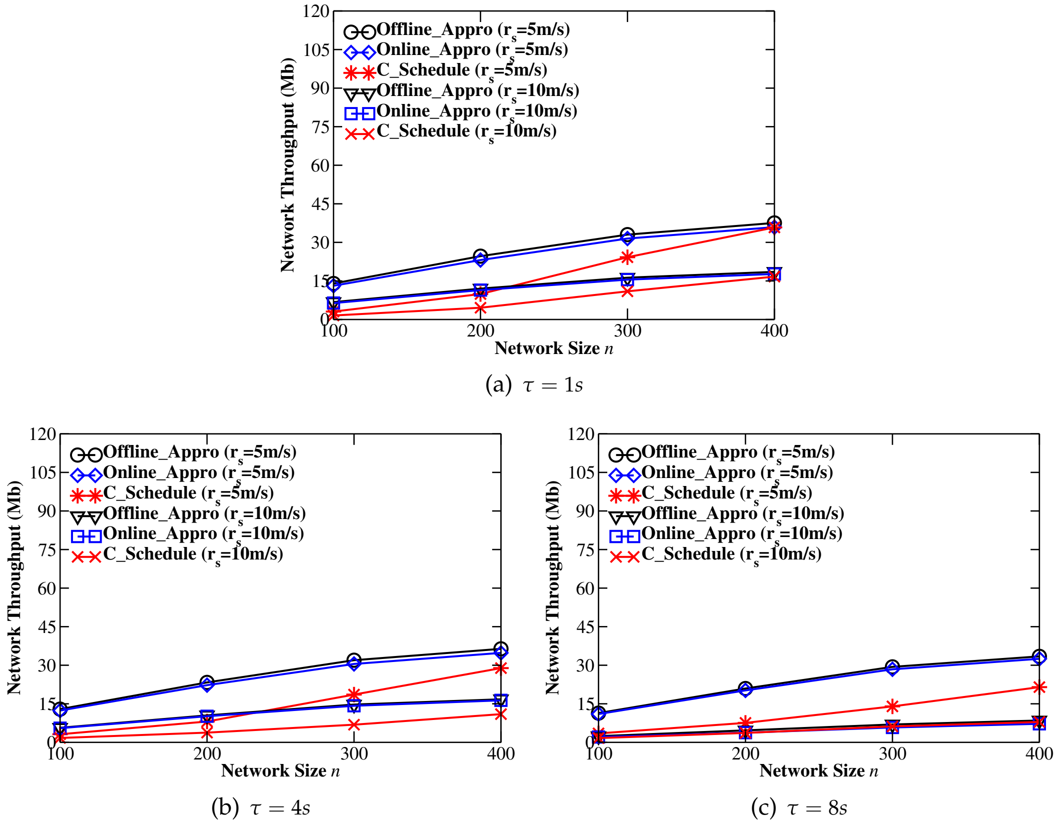
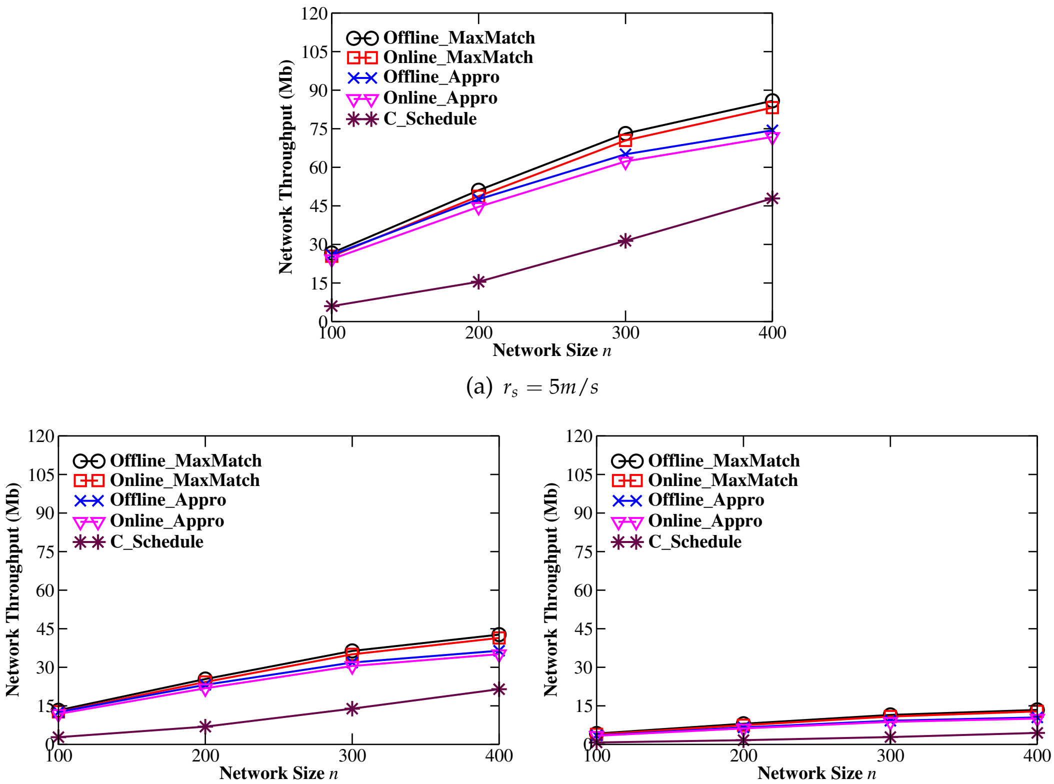


Figure 3.4: Network throughput delivered by algorithms Offline_Appro, Online_Appro, and C_Schedule through varying the sink speed r_s and the network size n when the link reliability is between 0 and 1.

3.7.3 Performance Evaluation of Different Algorithms for the Special Data Collection Maximization Problem

When the transmission power of each sensor is fixed at $300mW$, we now investigate both the performance of algorithms Offline_MaxMatch, Online_MaxMatch, Offline_Appro, and Online_Appro against algorithm C_Schedule and the impacts of the network size n and the mobile sink speed r_s on the performance, by varying n from 100 to 400 and setting r_s at $5m/s$, $10m/s$, and $30m/s$ while the duration of time slot τ is fixed at $1s$.

When the mobile sink speed is fixed at $5m/s$, Fig. 3.5(a) clearly indicates that algorithm `Offline_MaxMatch` outperforms the other four algorithms, and algorithm `C_Schedule` is the worst one among them. Moreover, it is observed that algorithm `Online_MaxMatch` is inferior to algorithm `Offline_MaxMatch`, as algorithm `Online_MaxMatch` only has the local knowledge of the network. However, the performance gap between them is only marginal. It is also noticed that algorithm `Online_MaxMatch` outperforms the other three algorithms, and the performance gaps among them increase with the growth of network size. Specifically, when $n = 100$, the performance of algorithms `Online_MaxMatch`, `Offline_Appro`, and `Online_Appro` are almost the same. When $n = 400$, the performance of algorithm `Online_MaxMatch` is 18% and 23% better than that of algorithms `Offline_Appro` and `Online_Appro`. When the mobile sink speed is fixed at $10m/s$ or $30m/s$ respectively,



(a) $r_s = 5m/s$

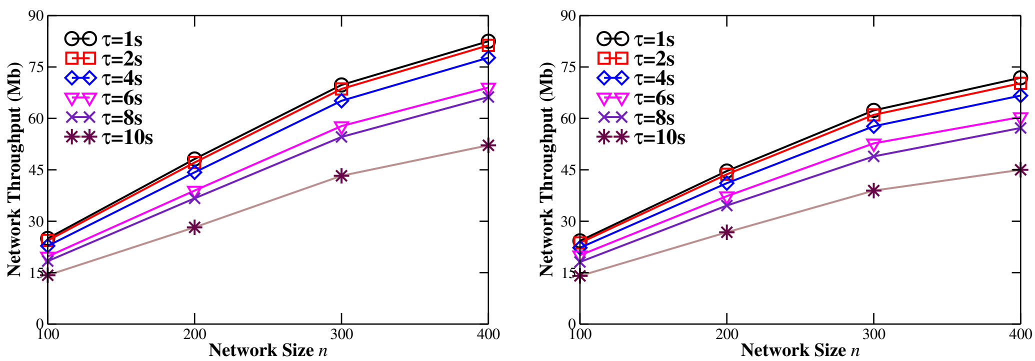
(b) $r_s = 10m/s$

(c) $r_s = 30m/s$

Figure 3.5: Network throughput delivered by different algorithms for a special case through varying the mobile sink speed r_s and the network size n when all links are reliable.

Fig. 3.5(b) and 3.5(c) exhibit the similar performance behaviors as Fig. 3.5(a), omitted. In summary, Fig. 3.5 implies that when the network size is fixed, the network throughput delivered by each mentioned algorithm decreases, with the increase of the mobile sink speed. Specifically, the network throughput delivered by algorithm `Offline_MaxMatch` when $r_s = 5m/s$ is at least 105% and 617% higher than that by itself when $r_s = 10m/s$ and $30m/s$, respectively. This is because when the mobile sink travels at a faster speed, the duration of the mobile sink travels the entire path will be shortened, while the data transmission rate is still keeping unchanged, thus, the amount of uploading data from sensors will be reduced. Although a faster speed leads to a shorter delay on data delivery, it will result in a less amount of data collected per tour.

We finally study the impact of the duration of time slot τ and the network size n on the performance of algorithms `Online_MaxMatch` and `Online_Appro`, by varying n from 100 to 400 and setting τ as $1s$, $2s$, $4s$, $6s$, $8s$, and $10s$ respectively, while keeping the mobile sink speed r_s at $5m/s$. Fig. 3.6(a) and 3.6(b) illustrate that for both



(a) Algorithm `Online_MaxMatch` (b) Algorithm `Online_Appro`

Figure 3.6: Impact of network size n and the time slot duration τ on the network throughput delivered by algorithms `Online_MaxMatch` and `Online_Appro`when all links are reliable.

algorithms `Online_MaxMatch` and `Online_Appro`, the network throughput decreases with the increase of the duration of each time slot. Their performance gap becomes larger and larger, with the growth of the network size. Specifically, in Fig. 3.6(a), the network throughput of algorithm `Online_MaxMatch` with $\tau = 1s$ is at least 3%, 9%, 21%, 28%, and 61% higher than that by itself when $\tau = 2s$, $4s$, $6s$, $8s$, and $10s$, respectively. In Fig. 3.6(b), the network throughput of algorithm `Online_Appro` with $\tau = 1s$ is at least 2%, 7%, 18%, 24% and 56% higher than that by itself when $\tau = 2s$, $4s$, $6s$, $8s$, and $10s$, respectively. The reason behind is that with a shorter time slot, the registered sensors can utilize their energy more efficiently.

3.8 Conclusions

In this chapter we studied data collection in a renewable sensor network using a mobile sink that travels along a pre-defined path, by adopting multi-rate data transmission mechanisms and time-slot scheduling. We first formulated a novel data collection maximization problem and showed its NP-hardness. We then provided an offline approximation algorithm with a provable approximation ratio, by exploiting the combinatorial property of the problem, assuming that the global knowledge of the network is available. We also proposed a fast, scalable online distributed algorithm for realistic sensor networks without the global knowledge assumption. In addition, for a special case of the data collection maximization problem where each sensor has only one fixed transmission power, we proposed an exact solution to the problem. Finally, we conducted experiments by simulations to evaluate the performance of the proposed algorithms. Experimental results demonstrate that the proposed algorithms are efficient and scalable, and the solutions delivered are fractional of the optimum.

Chapter 4

Data Quality Maximization in Renewable Sensor Networks with A Mobile Sink

4.1 Introduction

Extensive studies on mobile data collection in traditional sensor networks have been conducted, which have demonstrated that mobile sinks can significantly improve various aspects of network performance including network lifetime, data delivery reliability, throughput, etc [52, 82, 106, 109]. Most of these existing studies focused on the trade-off between maximizing data quantity and prolonging network lifetime. However, network lifetime maximization is no longer a main issue in renewable sensor networks as sensors can get recharged by renewable energy. This creates a shift in research focus from network lifetime maximization to network utility (e.g., the quality of the collected data) maximization.

Among a variety of different mobile data gathering schemes, a typical scheme is data gathering via a sink with controlled mobility [69, 73]. In particular, a set of locations in the sensing field is chosen as sojourn locations. The mobile sink periodically carries out a data gathering tour by visiting sojourn locations such that it can traverse the transmission range of sensors in the network. When the mobile sink arrives at an sojourn location, it would collect data from sensors in the neighborhood. Thanks to the direct data transmissions between sensors and the mobile sink, uniform energy consumption can be achieved as each sensor no longer relays data for other sensors.

Since the mobile sink's move speed is much slower than that of electromagnetic or acoustic waves, data collection may suffer much larger latency.

In this chapter we consider data collection in a renewable sensor network via a sink with controlled mobility, by formulating a novel constrained optimization problem consisting of finding an optimal close trajectory for the mobile sink and scheduling the sojourn time at each sojourn location such that the network data quality (*the quality of data collected*) is maximized, subject to a specified tolerant delay constraint. Specifically, we assume that the mobile sink traverses along a close trajectory and stops at each sojourn location in the trajectory for a certain amount of time to collect data from one-hop sensors, and each sensor that has only sufficient energy during the sojourn time can perform its data transmission. The mobile sink finally will return to its starting point within the given time bound.

The main contribution of this chapter are as follows. We first formulate this problem as a data quality maximization problem consisting of finding a trajectory and sojourn time scheduling. Since this is a NP-hard problem, we then devise a heuristic algorithm which exhibits low computational complexity and high scalability. A distributed implementation of the proposed algorithm is also developed. Finally, we conduct experimental simulations to evaluate the performance of the proposed algorithms. Experimental results demonstrate that the proposed algorithms are efficient in terms of the quality of data collected.

The rest of the chapter is organized as follows. Section 4.2 provides the literature survey on the sink with controlled mobility for network lifetime prolongation in traditional sensor network. Section 4.3 introduces the network model and problem definition. Sections 4.4 and 4.5 are devoted to devising algorithms for the data quality maximization problem. Section 4.6 evaluates the performance of the proposed algorithms through experimental simulations, while section 4.7 concludes the chapter.

4.2 Related Work

Extensive studies on data gathering in traditional sensor networks have been conducted in the past several years. Via a mobile sink with controlled mobility, the related work on data gathering in such networks is briefly described as follows. Wang *et al.* [98] considered a joint optimization problem of determining the sink movement and its sojourn time at certain network nodes in a grid network so that the network lifetime is maximized. They proposed an integer linear programming based solution for the problem by finding the sojourn time of the mobile sink at each node, assuming that a half of the workload (the number of messages generated and received) of each node flow along its horizontal and another half flow along its vertical links towards the current location of the sink. Since the network is a grid network, the load at each sensor toward the current sink location can be calculated easily. Thus, they are able to calculate the exact energy consumption of each sensor at each possible location of the sink. Luo *et al.* [68] later considered a joint optimization problem for data gathering by proposing a two-stage scheduling. First, the mobile sink visits the 'anchor' locations one by one and sojourns at each of them for a short sampling period. During this stage, the sink collects the power consumption of all nodes and builds the sojourn time profile at each anchor point. The sink then solves an ILP formula, using the given sojourn time profiles. An anchor point would be dropped from the visiting list if the sojourn time at it is below a given threshold. Otherwise it would not be worthwhile to keep such a point as the sojourn time cannot amortize the overhead on building a routing tree rooted at it. What follows is to solve the ILP to find the exact sojourn time at each chosen anchor point. Basagni *et al.* [6] considered a more realistic model for network lifetime maximization by imposing two realistic constraints on mobile sinks, the maximum distance at its each movement and the minimum sojourn time at each sojourn location. To reduce the data loss due to the sink motion from one location to another, it is assumed that the moving distance of the sink from its current location to its next location is bounded by a given value, and the sink sojourns at each chosen location for at least a certain amount of time. Then the problem is to find a route that maximizes the network lifetime. They first formulated the problem into a mixed integer linear program and then proposed a simple, distributed heuristic by utilizing the routing tree structure. Xing *et al.* [104] proposed a rendezvous-based data collection approach that explores the controlled sink mobility and the capacity of in-network data caching by bounding the total travel distance of the mobile sink. They developed two approximation algorithms to minimize the sum of the consumed energy of all involved sensors. Note that the approximation ratios obtained are based on a simplified assumption of data gathering. That is, all data at each relay node is aggregated into a single packet prior to its transmission. Guney *et al.* [34] formulated the sink trajectory problem as a joint optimization problem that aims to identify the optimal sink locations and information flow path between sensors and sinks. They solved the problem through formulating it as a mixed integer linear program and developed several heuristics under the assumption that the mobile sink traverses all potential locations. Yun and Xia [113] considered the network lifetime maximization problem using a mobile sink for the underlying applications that tolerate delayed information delivery to the sink. With the assumption that each sensor is not required to send its data immediately when they are generated, instead the data can be stored at the sensor temporarily and be transmitted when the mobile sink is at the most favorable location to achieve the maximum network lifetime. They formulated the problem as an optimization problem subject to the delay bound constraint, and proposed a flow-based optimization framework for it. Xu *et al.* [106] addressed a delay-tolerant data collection problem for event-detection with a guaranteed collection rate. They formulated the problem as a sensor selection problem and solved the problem by incorporating the spatial-temporal correlation of the event so that the network lifetime can be significantly extended. Liang *et al.* [53, 54] incorporated the travel distance of the sink into the problem formulation and proposed heuristics to find a trajectory for a mobile sink so that the network lifetime can be maximized. Liang *et al.* [52] further extended their work on a single mobile sink to multiple sinks. Xu *et al.* [109] considered a delay-tolerant data collection problem subject to the tolerant delay constraint by proposing a heuristic so that the network lifetime can be significantly prolonged.

4.3 Data Quality Maximization Problem

We consider a renewable sensor network $G = (V \cup S, E)$ deployed for monitoring purpose with a set V of stationary sensor nodes, a set E of links, a mobile sink, and a set S of potential sojourn locations at which the mobile sink can sojourn which usually is determined in advance. For the sake of simplicity, we assume that all sensors are homogeneous, where our solutions can be extended to heterogeneous scenario easily. Each sensor $v \in V$ is powered by a rechargeable battery whose energy is harvested from its surrounding environment, and senses its vicinity with a data generation rate g_v. Given a tolerant data delivery delay T, the sensing data is stored at the sensor temporarily and will be uploaded to the mobile sink with a data transmission rate r_v if possible. Assuming that the maximum transmission range of each sensor and the mobile sink is R, there is a link $l_{i,j} \in E$ between a sensor $v_i \in V$ and the mobile sink located at $s_j \in S$ if the sink is within the maximum transmission range of v_i, i.e., the Euclidean distance $|l_{i,j}|$ between v_i and s_j is no greater than R. Denote by $N(s_j)$ the neighboring set of sensors within the maximum transmission range of s_j.

We consider a communication model similar as the one used in [119]. Specially, the mobile sink is equipped with multiple antennas and has unlimited energy supply in comparison with energy-constrained sensor nodes. Being a receiver with multiple receiving antennas, the mobile sink makes it possible for multiple sensors to concurrently transmit their data to it. By processing the received signals from different sensors with filtering based on the channel state information, the mobile sink can successfully separate and decode the information from different sensors.

We assume that the set of potential sojourn locations S of the mobile sink is known *a priori*. There are several ways to decide potential sojourn locations. One way is to use the positions of a subset of sensors as sojourn locations [104]. An example of this data gathering scenario is illustrated in Fig. 4.1, where the potential sojourn locations are known in advance and the mobile sink starts its tour from a fixed depot and sequentially chooses sojourn location to visit for data collection. The mobile sink will return to the depot within a given tolerant data delivery delay. Note that a sojourn location can be visited for many times. The tolerant delay imposed is based on two different reasons. First, applications often require data to be delivered within a certain amount time deadline. Second, the mobile sink must return to the depot to be serviced (e.g., petrol replenishment or energy recharging) [104].

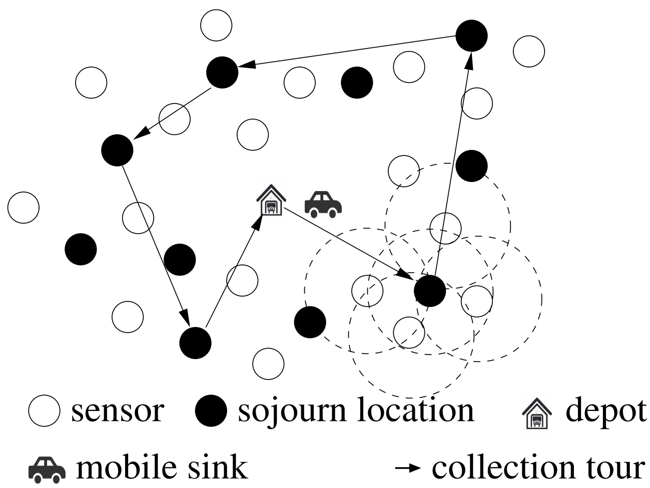


Figure 4.1: An example of the data gathering scheme.

4.3.1 Energy Budget and Consumption Models

We here adopt a similar energy budget model mentioned in chapter 2.3. Denote by B_v the energy storage capacity and $P_v(t)$ the amount of residual energy at time t of each sensor $v \in V$. Then, $P_v(t)$ can be estimated as follows:

$$P_v(t) = \min\{P_v(t') + \int_{t'}^{t} H_v(t)dt, B_v\} \tag{4.1}$$

where t' is the latest time when sensor v finished a data uploading to the mobile sink, $0 \leq t' \leq t$, and $H_v(t)$ represents the pattern of energy acquisition of sensor v. For the sake of discussion, we use P_v to represent the residual energy of sensor v at the moment in the rest of the chapter.

We assume that each sensor node only consumes energy on wireless communication, while its other energy consumptions including sensing and computation are ignored. We also adopt a similar energy consumption model mentioned in chapter 3.3. For each link $l_{i,j}$, let $e_{i,j} \propto |l_{i,j}|^\gamma$ be the energy consumption per second of node v_i by transmitting data to the mobile sink located at s_j, which is determined by the distance. The exponent γ is the path loss factor, which is typically a constant between 2 and 4.

Given a sojourn location $s_j \in S$ of the mobile sink, if it is within the maximum transmission range of a sensor $v_i \in V$, *the survival time* $t_{i,j}$ of a sensor v_i can be determined by its residual energy P_{v_i} and the amount of stored data D_{v_i}, which is calculated as follows.

$$t_{i,j} = \min\{\frac{P_{v_i}}{e_{i,j} \cdot r_{v_i}}, \frac{D_{v_i}}{r_{v_i} - g_{v_i}}\} \tag{4.2}$$

where $\frac{P_{v_i}}{e_{i,j} \cdot r_{v_i}}$ represents the time allowed for data transmission prior to its energy depletion, and $\frac{D_{v_i}}{r_{v_i} - g_{v_i}}$ represents the time required to transfer all the stored data.

4.3.2 Data Quality

To measure the contribution of each sensor node $v \in V$ to the data quality of the entire network in a time period of T, a utility function $u(v)$ is used to represent such a contribution, where $u(v)$ is a monotonic increasing function, whose marginal utility decreases with the growth of its cumulative data collection during of the period of T. When $u(v) = 1$, all sensed data by sensor v during the period of T has been collected by the mobile sink; otherwise, we assume that at any time point during the period, the data generated by node v is treated equally in terms of its contribution to the data quality of the entire network. In other words, we do not distinguish whether the sensing data is obtained at time point t_a or at time point t_b with $t_a \neq t_b$, and treat them equally. We then define the utility of sensor v in a tour of the mobile sink as $u(v) = f(x_v)$ where x_v is the ratio of the data collected to the data generated by sensor v, and the function $f(x)$ is a monotonic increasing function. If there is not any data sent from a sensor v to the mobile sink during the time period of T, then $u(v) = 0$. Clearly, $0 \leq u(v) \leq 1$. Thus, maximizing the data quality of network-wide (or *network data quality*) is equivalent to maximizing the sum of utilities of all nodes, $\sum_{v \in V} u(v)$.

4.3.3 Problem Statement

Given a renewable sensor network $G(V \cup S, E)$ with a sensor set V and a potential sojourn location set S for a mobile sink, and a tolerant data delivery delay T, *the data quality maximization problem* in G is to find an optimal close trajectory for the mobile sink that consists of sojourn locations in S and time scheduling at each sojourn location such that the network data quality is maximized, subject to the tolerant delay T, assuming that the mobile sink only collects data from one-hop sensor nodes, where the tolerant delay T is the total amount of time spent by the mobile sink per tour. Notice that this one-hop data transmission can be easily extended to multi-hop data transmission. In this latter case, we can treat one-hop sensors as gateways that all other sensors' data will be relayed to them. In other words, let $S' = \langle s_0, s_1, s_2, \cdots, s_m \rangle$ be the sequence of sojourn locations in the trajectory of the mobile sink, where for all $s_j \in S'$ t'_j is the travel time from s_{j-1} to s_j, t_j is the sojourn time at location s_j, $1 \leq j \leq m$, and s_0 is the depot of the mobile sink. The utility of sensor $v \in V$ in a tour of the mobile sink is $u(v) = \sum_{j=1}^{m} \delta_j(v) \cdot u_j(v)$, where $\delta_j(v)$ is either 1 or 0, depending on whether v sends its data to the mobile sink when the mobile sink is located at s_j, and $u_j(v)$ is the utility gain of v if it does send its data to the mobile sink. Specifically, assume that sensor v sent its data to the mobile sink previously when the mobile sink was located at $s_{j_1}, s_{j_2}, \ldots$, and s_{j_l}, respectively. When the sink is located at s_j with sojourn duration of t_j, the utility gain $u_j(v)$ of v is calculated as follows.

$$u_j(v) = f(\frac{(\sum_{i=1}^{l} t_{j_i} + t_j) \cdot r_v}{T \cdot g_v}) - f(\frac{(\sum_{i=1}^{l} t_{j_i}) \cdot r_v}{T \cdot g_v}) \tag{4.3}$$

where t_{j_i} is the sojourn duration of the mobile sink at location s_{j_i} with $1 \leq i \leq l$. The data quality maximization problem in G therefore is to find a close trajectory for the mobile sink and the sojourn time for each chosen sojourn location in the trajectory such that the value of $\sum_{v \in V} u(v)$ is maximized, subject to the time constraint T.

4.3.4 NP-Hardness

Assuming the data quality function $f(x)$ is linear, we then formulate the data quantity maximization problem, a special case of the data quality maximization problem.

We now show the data quantity maximization problem is NP-hard by the following theorem.

Theorem 8 *The decision version of the data quantity maximization problem in renewable sensor networks with a mobile sink is NP-hard.*

Proof We show the claim by a polynomial reduction from a well known NP-complete problem – the set cover problem [30], as follows. An instance of the set cover problem is: given a set of n elements $U = \{a_1, a_2, \ldots, a_n\}$ and a family of m subsets $\mathcal{F} = \{S_1, S_2, \ldots, S_m\}$ with $S_j \subseteq U$ and $\bigcup_{j=1}^{m} S_j = U$. Now, given a positive integer K ($K \leq m$), the decision version of the instance is to determine whether there is a collection $\mathcal{C}$ of K sets from $\mathcal{F}$ such that $\bigcup_{S_i \in \mathcal{C}} S_i = U$, where $\mathcal{C} \subseteq \mathcal{F}$.

We now reduce this instance to an instance of the data quantity maximization problem in a renewable sensor network $G(V \cup S, E)$ with a mobile sink as follows. Each element $a_i \in U$ corresponds to a sensor $v_i \in V$ with initial energy capacity of $c_v = 1$, assume that the data generation rate of each sensor is 1. Each subset $S_j \in \mathcal{F}$ corresponds to a potential sojourn location $s_j \in S$, and its elements correspond to the sensors that are within the transmission range of the mobile sink located at s_j, i.e. $N(s_j) = S_j$ if there is no distinction between a sensor v_i and its corresponding element $e_i \in S_j$, there is an edge in E between v_i and s_j. When a location $s_j \in S$ is chosen as the sojourn location of the mobile sink, then, all sensor nodes in $N(s_j)$ that have not yet sent their data to the mobile sink will have their initial energy capacity of 1 following our energy consumption assumption (we ignore the energy consumption in data sensing). They all will consume the same amount of energy $e = 1$ by transmitting a packet from each of them to the mobile sink because they have identical transmission ranges, assuming that the energy consumption per packet transfer is 1. We further assume that the energy replenishment rate of each sensor is very slow during the given time period of T, and the total amount of energy harvested during this period is no greater than $\frac{1}{n^2}$. In other words, during the tolerant delay period of T, each sensor $v \in V$ can send a packet to the mobile sink at most once, since its survival time is $\frac{C_v}{e} = \frac{1}{1} = 1$. Once its data has been sent to the mobile sink, the accumulative harvested energy of sensor v is not enough for it to send its other sensing data to the mobile sink again. Furthermore, we assume that the time spent on the traveling by the mobile sink from one sojourn location to another is a small fraction of 1 , e.g. the time for each movement of the mobile sink is no more than $\frac{1}{n+1}$, thus, the total amount of time used for the traveling of the mobile sink $\zeta \leq \frac{n}{n+1}$ (< 1) is strictly less than 1.

Having constructed an instance of the data quantity maximization problem in G with a given tolerant delay $T = K + 1$, the decision version of the throughput maximization instance is to decide whether there is a close trajectory for the mobile sink and sojourn time scheduling such that the data quantity ratio is $\frac{1}{K}$ (i.e., one packet from each sensor will be collected), subject to the tolerant delay T. If there is such a solution to this instance, there is a corresponding solution to the set cover instance as follows.

It is easy to verify that the found trajectory contains exactly K sojourn locations; otherwise, the total time spent per tour will be larger than $K + 1 > T$, since the sojourn time at each sojourn location is exactly 1. Each sojourn location s_j in the trajectory corresponds to a set in $\mathcal{F}$, and all sensors within the maximum transmission range of the mobile sink located at s_j are the elements in the corresponding set $S_j \in \mathcal{F}$. Given a sojourn location s_j, only these sensors within the transmission range of the mobile sink located at s_j that have the initial energy capacities can transmit their data to the mobile sink, and the volume of data collected by the mobile sink is equal to the number of such sensors. During the time period of T, it is known that each sensor has sent exactly one packet to the mobile sink when the mobile sink is at one of the K sojourn locations, i.e., the corresponding element in U of each sensor will be in a corresponding set of the sojourn location. This means that the union of all elements in U will be covered by the K sets. As the amount of time spent on travelling ζ is less than 1, the amount of time spent by the mobile sink per tour is $K < K + \zeta < T$. Thus, given the tolerant delay $T = K + 1$, there is a solution for the instance of the data quantity maximization problem with the data quantity ratio of $\frac{n}{(T-1)\cdot n \cdot r_g} = 1/K$ if and only if there is a solution to the instance of the set cover problem with the cardinality of K. Note that it takes a fraction of unit-time on mobile sink traveling and we assume that a single packet per time unit will be generated. The total volume of data generated by all sensors per tour is $(T-1) \cdot r_g \cdot |V| = (T-1) \cdot 1 \cdot n$. However, it is well known that the set cover problem is NP-complete [30]. Thus, the data quantity maximization problem is NP-hard. □

To this end, the data quality maximization problem is NP-hard, as the data quantity maximization problem is a special case of this general setting.

4.4 Centralized Heuristic

In this section we deal with the data quality maximization problem by devising a scalable heuristic. The proposed algorithm proceeds a number of iterations. Within each iteration a new sojourn location as well as the sojourn time at the location is added to the constructed trajectory. This procedure continues until the specified tolerant delay of the trajectory does not hold any more.

4.4.1 Algorithm

Suppose that s_i is the current location of the mobile sink in the found trajectory so far, we consider the next sojourn location s_j of the mobile sink. Notice that a visited sojourn location can be revisited multiple times. A location $s_j \in S$ is a *feasible sojourn location* if the time spent on all previous sojourn locations and traveling plus the time t'_j from s_i to s_j, the sojourn time t_j at s_j, and the time $t'_{j,0}$ from s_j to the depot s_0 is no more than T, i.e., $\sum_{l=0}^{i}(t'_l + t_l) + t'_j + t_j + t'_{j,0} \leq T$.

Consider a potential next sojourn location s_j. The data collected by the mobile sink at s_j is determined jointly by its sojourn time t_j and its neighboring sensors. To maximize the quality of data collected at each sojourn location, ideally the mobile sink should move to a location a bit far away from its current location s_i, thus, all neighboring sensors of the new location will have enough energy to transmit their data to the mobile sink and the expected quality of data collected will be maximized because it is very likely that the sensing data from these sensors has not been collected yet. On the other hand, the travel distance between the current location and the next sojourn one should not be too far away from each other, otherwise no data will be collected during the traveling of the mobile sink. To mitigate the data loss due to mobile sink traveling, its travel distance should be shortened. Thus, there is a nontrivial trade-off between the travel distance and the amount of sojourn time at the next sojourn location when the mobile sink chooses its next sojourn location. In the following we show how to choose the next sojourn location $s_j \in S$.

Recall that $N(s_j)$ is the set of sensors that the mobile sink located at s_j is within their transmission ranges and $t_{i,j}$ is the survival time of sensor $v_i \in N(s_j)$ if it sends its data to the mobile sink. Let $v_{i_1}, v_{i_2}, \ldots, v_{i_{|N(s_j)|}}$ be the sensor sequence sorted by their survival time in decreasing order. Denote by $\Delta u(s_j, v_{i_{l'}}) = f(\frac{(T_i(v_{i_{l'}})+t_{i_l,j})\cdot r_{v_{i_{l'}}}}{T\cdot g_{v_{i_{l'}}}}) - f(\frac{T_i(v_{i_{l'}})\cdot r_{v_{i_{l'}}}}{T\cdot g_{v_{i_{l'}}}})$ the utility gain of sensor $v_{i_{l'}}$ by sending its data to the mobile sink at location s_j with the time duration of $t_{i_l,j}$ if $t_{i_l,j} \le t_{i_{l'},j}$, where $T_i(v)$ is the accumulative sojourn time of sensor v prior to the sojourn location s_j. Otherwise, $\Delta u(s_j, v_{i_{l'}}) = f(\frac{(T_i(v_{i_{l'}})+t_{i_{l'},j})\cdot r_{v_{i_{l'}}}}{T\cdot g_{v_{i_{l'}}}}) - f(\frac{T_i(v_{i_{l'}})\cdot r_{v_{i_{l'}}}}{T\cdot g_{v_{i_{l'}}}})$ if $t_{i_l,j} > t_{i_{l'},j}$. Then, the utility gain $u(s_j, i_l)$ when the mobile sink at location s_j with a sojourn time $t_{i_l,j}$ is

$$u(s_j, i_l) = \frac{\sum_{v \in N(s_j)} \Delta u(s_j, v)}{\Delta t(s_j, i_l)}, \tag{4.4}$$

where $\Delta t(s_j, i_l) = t'_j + t_{i_l,j} + t'_{j,0} - t'_{i,0}$ is the time cost associated with this utility gain, assuming that the speed of the mobile sink r_m is fixed. Define the utility gain sequence at location s_j with different sojourn times as follows.

$$u(s_j, i_1), u(s_j, i_2), \ldots, u(s_j, i_{|N(s_j)|}). \tag{4.5}$$

Since we aim to maximize the aggregate utility of all sensors, we can choose a sojourn time $t_{i_k,j}$ for the mobile sink at each sojourn location s_j such that the utility gain $u(s_j, i_k)$ is maximized, i.e., we identify an index i_k of a maximum term in sequence (4.5) as the utility gain of the mobile sink at location s_j, $1 \le k \le |N(s_j)|$ and $1 \le j \le |S|$. If we choose location s_j as the next sojourn location of the mobile sink, the sojourn time of the mobile sink at s_j is $t_j = t_{i_k,j}$, and the utility gain is $U(s_j) = u(s_j, i_k)$. Thus, given all feasible sojourn locations, to maximize the network data quality, a location s_j with the maximum value of $U(s_j)$ will be chosen as the next sojourn location of the mobile sink.

Algorithm 9 `Last_Location`

Input: A found trajectory S' in which s_i is the last sojourn location, sink speed r_m
Output: The last sojourn location *last_soj_location* and its sojourn time *last_soj_time*

$max\_gain \leftarrow 0$;
for each location $s_j \in S$ **do**
  Compute Δt_j;
  if $\Delta t_j > 0$ **then**
    Identify the terms whose survival times are no greater than Δt_j and choose a term with the maximum value from the sequence $u(s_j, i_1), u(s_j, i_2), \ldots, u(s_j, i_{|N(s_j)|})$;
    Let i_k be the index of the term;
    $t_j \leftarrow t_{i_k,j}$; $U(s_j) \leftarrow u(s_j, i_k)$;
    if $max\_gain < U(s_j)$ **then**
      $last\_soj\_location \leftarrow s_j$; $max\_gain \leftarrow U(s_j)$; $last\_soj\_time \leftarrow t_j$;
    end if
  end if
end for
return Last location *last_soj_location* and the sojourn time *last_soj_time*.

With more and more sojourn locations added to the trajectory, we then reach a point where no any location in S will be a feasible sojourn location for the trajectory. Consider a location s_j which has $T_i + t_j + t'_j + t'_{j,0} > T$ while $T_i + t'_j + t'_{j,0} < T$, where T_i is the amount of time spent by the mobile sink prior to location s_j and $t_j = t_{i_k,j}$ defined as the above. For this case, if s_j is chosen as a sojourn location of the mobile sink, it must be the last sojourn location in the trajectory, and at which the sojourn time of the mobile sink should be no more than $\Delta t_j = T - (T_i + t'_j + t'_{j,0})$. To find an appropriate sojourn time at location s_j, we re-examine sequence (4.5) by identifying these terms whose survival times are no greater than Δt_j, choose a term among them with the maximum value (the maximum utility gain), and put its relevant time as the sojourn time of the mobile sink at s_j. If there are multiple such locations, we choose the one that results in the maximum utility gain. The detailed description of the choice of the last sojourn location of the trajectory is given by **Algorithm** 9 `Last_Location`.

The detailed algorithm for the data quality maximization problem, `Max_Utility`, is described in **Algorithm** 10.

Algorithm 10 `Max_Utility`

Input: The set of potential sojourn locations $S \cup \{s_0\}$, the tolerant delay T, and the sink speed r_m

Output: The trajectory of the sink and the sojourn time t_j at each sojourn location s_j

$S' \leftarrow \langle s_0 \rangle$; /* the location sequence in trajectory */;
$max\_gain \leftarrow 0$;
while there is a feasible sojourn location **do**
  for each feasible sojourn location $s_j \in S$ **do**
    for each sensor $v_i \in N(s_j)$ **do**
      Compute the survival time $t_{i,j}$;
    end for
    Sort the survival time sequence in non-increasing order. Let $t_{i_1,j}, t_{i_2,j}, \ldots, t_{i_{|N(s_j)|},j}$ be the sorted sequence and $v_{i_1}, v_{i_2}, \ldots, v_{i_{|N(s_j)|}}$ the corresponding sensor sequence;
    Find the maximum term in sequence $u(s_j, i_1), u(s_j, i_2), \ldots, u(s_j, i_{|N(s_j)|})$. Let i_k be the index of the maximum term;
    $t_j \leftarrow t_{i_k,j}$; $U(s_j) \leftarrow u(s_j, i_k)$;
    if $max\_gain < U(s_j)$ **then**
      $next\_soj\_location \leftarrow s_j$; $next\_soj\_time \leftarrow t_j$; $max\_gain \leftarrow U(s_j)$;
    end if
  end for
  $S' \leftarrow S' \cup \{next\_soj\_location\}$;
  Update the residual energy B_v for every sensor v;
end while
Add the last sojourn location by calling routine `Last_Location` (S');
return S' and the sojourn time t_j for each sojourn location $s_j \in S'$.

4.4.2 Complexity Analysis

We now analyze the properties of algorithm `Max_Utility` as follows.

Lemma 3 *In algorithm* `Max_Utility`*, given a feasible sojourn location s_j, assume that the sojourn time at that location is $t_{i_k,j}$, then, the time cost by adding location s_j to the trajectory must be greater than zero, i.e., $\Delta t(s_j, i_k) = t'_j + t_{i_k,j} + t'_{j,0} - t'_{i,0} \geq 0$ for some k with $1 \leq k \leq |N(s_j)|$.*

Proof It is known that $t'_{j,0} = \frac{|l_{j,0}|}{r_m}$ and $t'_{i,0} = \frac{|l_{i,0}|}{r_m}$. If $|l_{j,0}| \geq |l_{i,0}|$, then $t'_{j,0} \geq t'_{i,0}$, and $\Delta t(s_j, i_k) = t'_j + t_{i_k,j} + (t'_{j,0} - t'_{i,0}) \geq 0$; otherwise, $|l_{i,0}| \leq |l_{i,j}| + |l_{j,0}|$ by the triangle inequality, i.e., $t'_{i,0} \leq t'_j + t'_{j,0}$. Then, $\Delta t(s_j, i_k) = t'_j + t_{i_k,j} + t'_{j,0} - t'_{i,0} = t_{i_k,j} + (t'_j + t'_{j,0} - t'_{i,0}) \geq 0$. □

Lemma 4 *In the construction of the trajectory in algorithm* `Max_Utility`*, for a given location $s_j \in S$, the following may happen: the sum of time $T_i + t_j + t'_j + t'_{j,0} > T$ but $T_i + t'_j + t'_{j,0} < T$, the location s_j if added to the trajectory must be the last sojourn location in the trajectory, where T_i is the total amount of time (sojourn time and the travel time) spent on the trajectory prior to adding location s_j to the trajectory.*

Proof We first show that the trajectory always meets the tolerant delay constraint T. It is trivial when the trajectory contains the depot s_0 only. Assume that this claim holds for the found trajectory and s_i is the last sojourn location in the found trajectory. We now extend the trajectory by adding the next sojourn location s_j if possible. If s_j is a feasible sojourn location, then, the time constraint T on the tour is still met; otherwise, no feasible sojourn location in S can be found. However, we notice that there is still a time gap between the given time bound T and the actual time spent on the found trajectory so far. We can make use of this time gap to collect more sensing data from the sensors in the network. To do so, one more sojourn location can be added to the trajectory. Let location s_j be such one candidate of the sojourn location, then $T_i + t_j + t'_j + t'_{j,0} > T$ and $T_i + t'_j + t'_{j,0} < T$; otherwise, s_j is a feasible sojourn location that has already been added to the trajectory by our assumption.

To ensure that the resulting trajectory still meets the tolerant delay constraint, a subset of sensors in $N(s_j)$ with a sojourn time of a chosen location will be determined (by algorithm 9), through which the sojourn time should not be greater than the time gap, and such a location s_j must be the last location in the trajectory because no more time gap will be left. $\square$

Theorem 9 *Given a renewable sensor network $G(V \cup S, E)$, a mobile sink with potential sojourn location space S and a specified tolerant delay T, there is an algorithm for the data quality maximization problem in G, which takes $O(n \log n \cdot |S| \cdot T)$ time, where $n = |V|$.*

Proof We analyze the time complexity of the proposed algorithm `Max_Utility` as follows. Within each iteration, a new sojourn location s will be added to the found trajectory, while finding such a new location takes $O(|N_{max}(S)| \log |N_{max}(S)| \cdot |S|) = O(n \log n \cdot |S|)$ time due to survival time sorting of all nodes in $N(s)$ and $|S|$ feasible sojourn locations to be examined, where $N_{max}(S) = \{N(s) \mid |N(s)| \geq |N(s')|$ for all $s, s' \in S\}$ and $|N_{max}(S)| \leq |V| = n$. The number of iterations is determined by T, thus, the algorithm takes $O(n \log n \cdot |S| \cdot n')$ where n' is the number of sojourn locations in the trajectory. If the sojourn time spent at each sojourn location is at least one-unit time, then the algorithm takes $O(n^2 \log n \cdot T)$ time since $n' \leq T$ and $|S| << n$. $\square$

4.5 Distributed Implementation

So far the proposed algorithm is a centralized algorithm. As the network we are dealing with is a distributive sensor network, we now provide a distributed implementation of the proposed algorithm in such a distributed environment, where each node has only the knowledge of its neighbors not the entire network and harvesting energy prediction at each sensor is not required, assuming that the mobile sink knows the network topology and the location information of potential sojourn locations.

The distributed algorithm constructs the trajectory iteratively too. Within each iteration, one sojourn location is added to the trajectory. Assume that the mobile sink is currently located at s_i and tries to find its next sojourn location and the sojourn time at that location. To choose its next sojourn location s_j, the following metrics must be considered. (i) Prior to arriving its next location s_j, the mobile sink does not know the residual energy of sensors in $N(s_j)$. We here use a 'time-stamp' to approximately represent such information associated with each potential location of the mobile sink, which is the most recent visited time by the mobile sink. Intuitively, when a location has a larger time stamp value, it implies that the location should be avoided to be revisited soon because its neighboring sensors just sent their data recently and they have not been yet fully recharged. (ii) The ideal distance of the next sojourn location from the current one should be twice the maximum transmission range of the mobile sink. Otherwise, if the next sojourn location is within the transmission range of the current one, it is very likely that the data generated by most of its neighboring sensors has been collected in the previously visited sojourn locations. On the other hand, if the next sojourn location is chosen far from the current one, although a large volume of data can be collected from that location, the time overhead on traveling is relatively large, as the total amount of time per tour is bounded by T. Thus, each potential sojourn location $s_j \in S$ will be ranked based on its *priority weight* consists of its time stamp, the distance to the current sojourn location, and the amount of traveling time $t'_j + t'_{j,0}$ of s_j. The mobile sink then chooses a location with the highest priority as its next sojourn location. Fig. 4.2 is an example to illustrate the choice of next sojourn location of the mobile sink. If the location s_w near to s_i is chosen as the next sojourn location, the expected volume of data collected from this location would not be big, because most its neighboring sensors have just sent their data to the mobile sink when the sink was at location s_i. The next three locations are s_l, $s_{j'}$ and s_j, where the distances from both $s_{j'}$ and s_j to s_i are at least twice the maximum transmission range R. The priority weight of s_j is higher than that of $s_{j'}$ because the traveling time at s_j is less than that at $s_{j'}$. Although location s_l has the similar property as location s_j and the less amount of traveling time, its time-stamp value is larger than that of location s_j, which means that some of its neighboring sensors have just sent their data to the mobile sink at a previous location s_{i-1}. Thus, its priority weight is lower than that of s_j. For location s_k, although its time stamp value is small, it has a longer distance from the current location and the traveling time at it is far larger than to any other location. Thus, it would not have a higher priority weight. Priority weights of all locations are labeled in the figure.

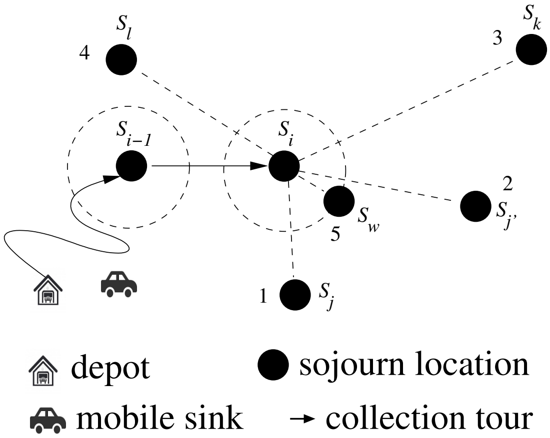


Figure 4.2: The choice of the next sojourn location s_j of the mobile sink, assuming that the current sojourn location is s_i.

The distributed algorithm for the data quality maximization problem, `Dis_Max_Utility`, is described as follows. For the mobile sink located at the current sojourn location s_i (initially located at s_0), it repeats the following procedure. It finds its next sojourn location s_j and the sojourn time based on the priority weight metric. If no such a location exists, this implies that adding any new location will violate the specified tolerant delay constraint, the algorithm terminates. Otherwise, a location s_j with the highest priority will be chosen as its next sojourn location, and the mobile sink travels from s_i to s_j. The mobile sink then broadcasts a 'Hello' message at location s_j. Upon receiving 'ACK' messages from all responded sensors $v_{i'} \in N(s_j)$, the mobile sink computes the survival time $t_{i',j}$ of each sensor $v_{i'}$. Let $t_{i_1,1}, t_{i_2,j}, \ldots, t_{i_{|N(s_j)|},j}$ be the sorted survival time sequence in non-increasing order and let $v_{i_1}, v_{i_2}, \ldots, v_{i_{|N(s_j)|}}$ be the corresponding sensor sequence. The mobile sink then finds the maximum term from sequence (4.5) and let i_k be the index of the maximum term. The mobile sink then updates the time-stamp of location s_j by a new value $T_i + t'_j + t_j$ and broadcasts the sojourn time t_j to all sensors in $N(s_j)$, where $t_j = t_{i_k,j}$. It finally receives the sensed data from the responded sensors.

For each sensor $v_{i'} \in N(s_j)$, upon receiving the 'Hello' message from the mobile sink, it responds by sending an 'ACK' message. The 'ACK' message contains the information of node $v_{i'}$: its current residual energy and its distance to location s_j. Upon receiving the sojourn time t_j from the mobile sink located at s_j, sensor $v_{i'}$ sends its data to the mobile sink for a time duration of t_j if $t_{i',j} \geq t_j$. Otherwise, it sends its data to the mobile sink for a time duration of $t_{i',j}$ if $t_{i',j} \leq t_j$.

Theorem 10 *Given a renewable sensor network $G(V \cup S, E)$, a mobile sink with potential sojourn location space S and a specified tolerant delay T, there is a distributed algorithm for the data quality maximization problem in G, which takes $O(|S| \cdot T)$ time.*

Proof We analyze the time complexity of the proposed algorithm `Dis_Max_Utility` as follows. Within each iteration, a new sojourn location s will be added to the found trajectory, while finding such a new location takes $O(|S|)$ time due to $|S|$ feasible sojourn locations to be examined. Thus, the algorithm takes $O(|S| \cdot n')$ where n' is the number of sojourn locations in the trajectory. If the sojourn time spent at each sojourn location is at least one-unit time, then the algorithm takes $O(|S| \cdot T)$ time since $n' \leq T$. $\square$

4.6 Performance Study

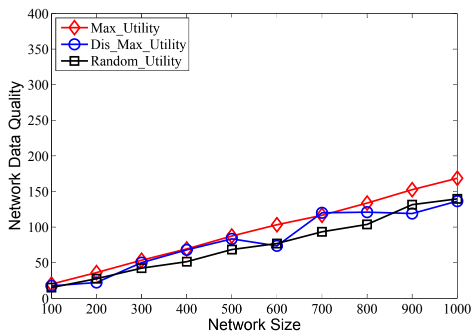
In this section we study the performance of the proposed algorithms through experimental simulation.

4.6.1 Experimental Environment Setting

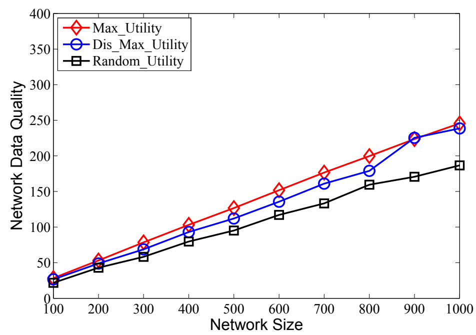
We consider a renewable sensor network consisting of 100 to $1,000$ sensors randomly deployed in a $100m \times 100m$ square region. The depot of the mobile sink is located at one corner of the square. For the sake of convenience, we set $r_m = 2m/s$ for the speed of the mobile sink, and adopt similar communication parameters setting mentioned in Chapter 3, where the unit energy consumption varies according to the corresponding distance to some extent. The potential sojourn locations in S

are also randomly generated with the default value $|S| = 50$. Each sensor v has a data generation rate $g_v = 1Kbps$ and a data transmission rate $r_v = 5Kbps$. Each sensor is powered by a solar panel with a dimension $(10mm \times 10mm)$ and its battery capacity is $10,000 Jules$. The solar power harvesting profile is built upon the real solar radiation measurements [61], in which the total amount of energy collected from a $37mm \times 33mm$ solar cell over a 48-hour period is $655.15mWh$ in a sunny day and $313.70mWh$ in a partly cloudy day. We here adopt $f(x) = \sqrt{x}$ as the utility function, which can be easily extended to other utility functions. Each value in figures is the mean of the results by applying each mentioned algorithm to 30 different network topologies of the same network size.

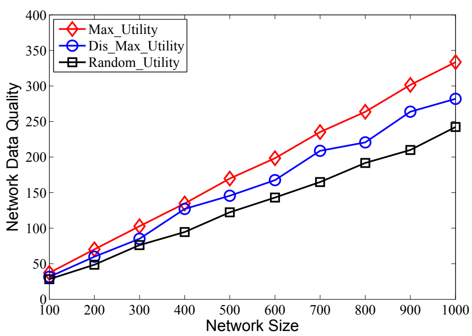
4.6.2 Performance Evaluation of Different Algorithms



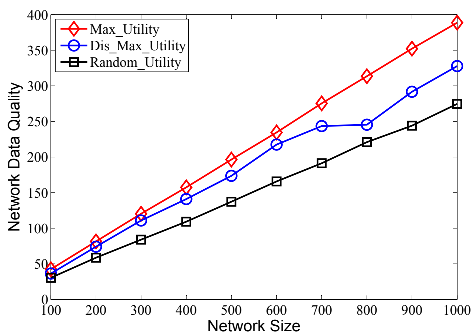
(a) $T = 200s$



(b) $T = 400s$



(c) $T = 800s$

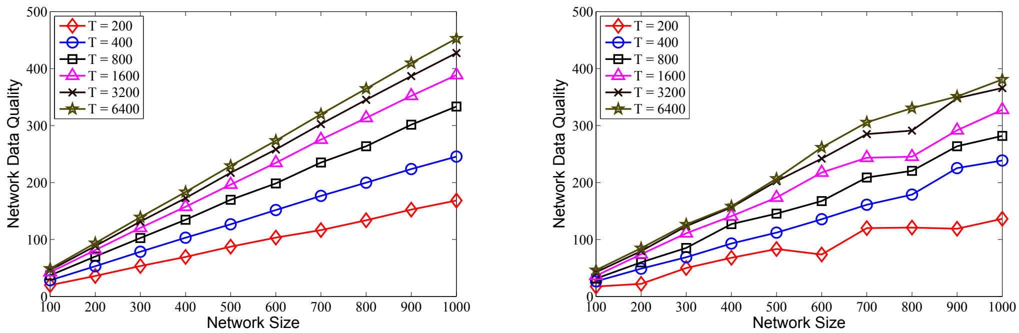


(d) $T = 1,600s$

Figure 4.3: The network data quality performance of different algorithms by varying the network size n and setting the tolerant delay $T = 200s$, $400s$, $800s$ and $1,600s$.

We first investigate the performance of algorithms `Max_Utility` and `Dis_Max_Utility` against that of another heuristic `Random_Utility` - a variant of algorithm `Max_Utility` by randomly selecting a feasible sojourn location in S within each iteration. We vary network size n from 100 to $1,000$ and set tolerant delay T as $200s$, $400s$, $800s$ and $1,600s$, respectively. Fig. 4.3 clearly shows that both algorithms `Max_Utility` and `Dis_Max_Utility` outperform algorithm `Random_Utility` significantly. For instance, when $T = 400s$ or $T = 1,600s$, the network data quality of algorithms `Max_Utility` and `Dis_Max_Utility` is around 32% and 21%, or 40% and 25% higher in comparison with that of algorithm `Random_Utility`. Note that when $T = 200s$, the network data quality of algorithms `Dis_Max_Utility` and `Random_Utility` sometimes are at the same level, because $200s$ is too tight so that the mobile sink can not have enough opportunities to visit sojourn locations.

4.6.3 Impact of Tolerant Delay T and Network Size N on the Performance of Algorithms



(a) Algorithm `Max_Utility` (b) Algorithm `Dis_Max_Utility`

Figure 4.4: Impact of parameters tolerant delay T and network size n on the network data quality performance of different algorithms.

We then study the impact of tolerant delay T and network size n on the performance of algorithms by varying n from 100 to $1,000$ while keeping tolerant delay T at $200s$, $400s$, $800s$, $1,600s$, $3,200s$, and $6,400s$, respectively. Fig. 4.4 shows that with the growth of T, the performance of algorithms `Max_Utility` and `Dis_Max_Utility` significantly improves. However, the performance gap among the mentioned algorithms becomes marginal when $T \geq 3,200s$, because all potential sojourn locations in S have almost been visited during this long time period. With the increase of network size, we observe that the network data quality of all proposed algorithms increases, too. For example, when T is fixed at 800s and n varies from 100 to $1,000$, the network data quality of algorithm `Max_Utility` is 37.1, 70.2, 102.7, 134.7, 169.7, 198.4, 235.1, 263.7, 301.5, and 333.5, respectively. The network data quality of algorithm `Dis_Max_Utility` is 31.5, 59.9, 85.1, 127.1, 145.5, 167.5, 208.8, 220.6, 263.8, and 281.8, respectively. Notice that algorithm `Max_Utility` outperforms algorithm `Dis_Max_Utility`, as `Dis_Max_Utility` only has the local knowledge of the network and does not need harvesting energy predictions of sensors.

4.7 Conclusions

In this chapter we have studied mobile data collection in a renewable sensor network via a mobile sink with controlled mobility, subject to the tolerant delay constraint. We first formulated a data quality maximization problem. We then devised a heuristic algorithm, and provided an efficient distributed implementation of the proposed algorithm. Finally, we evaluated the performance of the proposed algorithms through experimental simulation, and experimental results demonstrate that the proposed algorithms are very promising.

Chapter 5

Charging Maximization in Renewable Sensor Networks with A Mobile Charger

5.1 Introduction

With the advance of energy harvesting technology, renewable sensor networks are a key step in paving the way for truly green systems that can operate 'perpetually' and do not adversely impact on the environment. An ideal solution is to enable sensors to harvest energy from their surroundings [26, 44, 55, 77, 78, 79]. However, energy harvesting unfortunately is not stable and the amount of energy harvested is hardly predictable. For example, the solar energy harvested is usually affected by many factors including time (whether exposed under the sun), weather, and season. This poses a great challenge in the design of energy-efficient protocols for wireless sensor networks to maintain them operational.

The recent breakthrough in wireless energy transfer technology provides a promising alternative or supplementary solution to power sensors. Particularly, employing two strongly coupled magnetic resonant objects, Kurs *et al.* [49] exploit the resonant magnetic technique to transfer energy from one storage device to another without any plugs or wires. The reported experiment demonstrated that a wireless illumination of a 60 watts light bulb from 2 meters away achieved a 40% energy transfer efficiency. What makes such wireless energy transfer technology particularly attractive is that it does not require line-of-sight or any alignment (i.e., omnidirectional), and is insensitive to environments. Armed with this advanced technology, sensor can be charged at steady and high charging rates. On the other hand, another breakthrough in the ultra-fast charging battery materials further fuels the feasibility of the wireless power transfer technique. Scientists from MIT implemented a ultra-fast charging in material $LiFePO_4$, which can be charged at a rate as high as 400 *Coulombs* per second [43]. The time of fully-charging a battery thus can be shortened into a few seconds. Therefore, wireless power charging is a promising technique to prolong the lifetime of sensor networks. This promising technique will provide a controllable and perpetual energy source to recharge sensors if needed.

In this chapter, we employ mobile wireless chargers to replenish sensors in a large-scale sensor network with wireless power transfer. We consider a heterogenous sensor network in which sensors have significant variations in the sampling needs and energy consumptions. A typical example is a sensor network deployed for ecological study that consists of sensors of different modalities like humidity, temperature, video, etc. The sensing rates of different sensors vary, depending on their physical phenomenon. Under this setting we here investigate an on-demand wireless sensor charging paradigm. That is, sensors send their recharging requests to the base station according to their residual energy status, and the base station then dispatches the wireless mobile charger to start a charging tour and recharge these requested sensors. Specifically, the following issues must be addressed: (1) which sensors are to be included in each charging task? (2) given a set of to-be-charged sensors, which sensors should be charged first? We tackle these challenges by formulating a novel optimization problem and devising efficient scheduling algorithms for it.

The contributions of this chapter are summarized as follows.

- We first study an on-demand energy replenishment in renewable sensor networks by employing a wireless mobile charger and formulate an optimization problem with an objective of maximizing the number of sensors charged (charging throughput) per tour.

- We then devise an offline approximation algorithm which runs in quasi-polynomial time by reducing the formulated optimization problem to the orienteering problem with time windows. We also provide online heuristics where recharging requests arrive one by one without the future arrival knowledge.

- We finally conduct extensive simulations to study the efficiency of the proposed algorithms in both small-scale and large-scale networks. Experimental results demonstrate that the proposed algorithms are very efficient in terms of charging throughput.

The rest of the chapter is organized as follows. Section 5.2 summaries related works. Section 5.3 introduces the network model and problem definition. Sections 5.4 and 5.5 propose an offline approximation algorithm and two online heuristics, respectively. Section 5.6 presents the simulation results, and Section 5.7 concludes the chapter.

5.2 Related Work

Armed with the wireless energy transfer technology, several studies on employing mobile vehicles with high volume batteries as mobile chargers to recharge energy for sensors have been conducted [4, 21, 27, 38, 51, 75, 83, 95, 101, 116, 118]. For example, Shi *et al.* [83, 100] applied this technology for a wireless sensor network, where the sensing rates of sensors are fixed and known in advance, and sensing data is forwarded to a stationary base station through multi-hop relays. They formulated a joint optimization problem for flow routing and energy recharging, and showed that each sensor will not run out of energy by having a mobile charger periodically visits and charges it. Xie *et al.* [102] extended this solution by allowing charging multiple sensors simultaneously. Li *et al.* [51] analyzed the possibility of practical and efficient joint routing and charging schemes where each sensor sends data hop-by-hop to the sink periodically using the Collection Tree Protocol. They showed that the network lifetime is prolonged by a mobile charger which mostly moves along energy-minimum paths. Xie *et al.* [101, 103] applied this technology for a wireless sensor network where a mobile station is employed for both data collection and energy charging. They formulated an optimization problem that involves joint optimization of traveling path, stopping points, charging schedule and data flow routing, and developed a provably near-optimal solution. Zhao *et al.* [118] considered a joint optimization of mobile data collection and energy charging, and devised an adaptive solution that jointly selects the sensors to be charged and finds the optimal data gathering scheme. Wang *et al.* [95] studied wireless energy charging in event detection scenarios and proposed a joint solution including stochastic charging and adaptive sensor activation. Most of these mentioned studies assumed that both the sensing rate and the energy consumption rate of each sensor are fixed and given in advance. However, in terms of different application scenarios (e.g. event detection), both the sensing and energy consumption rates of each sensor vary over time. Thus, most of these existing solutions are not applicable in such time-varying application scenarios.

In terms of on-demand sensor energy replenishment, He *et. al.* [38] also studied an on-demand mobile charging problem. An essential difference between their work with ours is that they didn't put any constraint on the mobile charger in consideration, while we consider tour time constraint on the mobile charger.

5.3 Charging Throughput Maximization Problem

We consider a sensor network consisting a set V of heterogenous sensors and a stationary base station v_0 deployed over a rectangle region. Each sensor $v_i \in V$ is equipped with a rechargeable battery of capacity B_i and consumes energy on sensing and data transmission activities. Each sensor v_i will send its recharging request $c_i = (v_i, RE_i, r_i)$ once its residual energy RE_i falls below a pre-defined threshold $M_i = \alpha \cdot B_i$, where RE_i is the residual energy of v_i, r_i is the release time and α is a constant with $0 < \alpha < 1$.

5.3.1 Charging Model

A mobile charger is a moving vehicle equipped with a powerful wireless charger and it can keep information synchronized with the base station via a long range radio [51]. It starts from the base station and recharges sensors based on the recharging requests received. Since the mobile charger consumes petrol either on moving or charging, we then assume that each charging tour of the mobile charger is bounded by a pre-defined time period T. That is, the mobile charger must finally return to the base station within time period T to be serviced (e.g., refueling, maintenance service). For simplicity, we assume that a mobile charger per tour has enough energy to charge all sensors [38, 83]. In our model the charging is performed from points to points, i.e., only one sensor can be fully charged at each time by the mobile charger when the sensor is in the vicinity of the mobile charger so that the charging process has the maximum efficiency. Given battery material breakthroughs for ultra-fast charging [43], we further assume that the charging time at each sensor is a constant C [38]. We also assume that the mobile charger travels at a constant speed S. An example of this charging paradigm is illustrated in Fig. 5.1, where sensors will send their requests to either the base station or the mobile charger anytime if their residual energy levels are below the given thresholds. The mobile charger then starts a charging tour from the base station and travels around the deployment field to charge sensors. When the mobile charger is traveling, it may still receive new charging requests from sensors as well. Finally it returns to the base station within time period T so that it can be maintained and prepared for next charging tour.

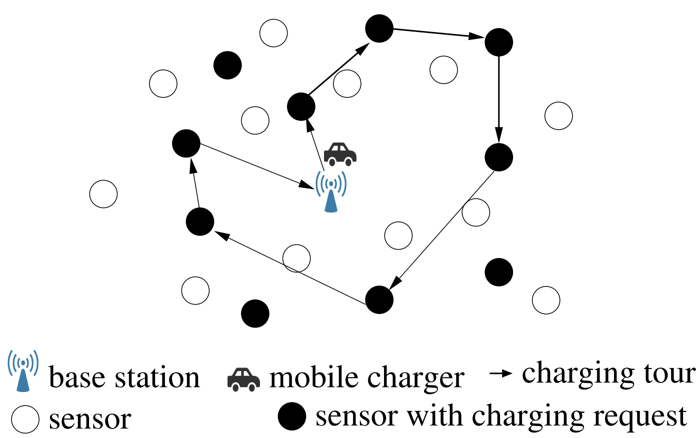


Figure 5.1: An example of charging paradigm.

5.3.2 Charging Throughput

In order to measure the contribution of the mobile charger, we introduce *the charging throughput* concept. If a sensor runs out of energy, it will stop functioning. We thus expect that each sensor never runs out of its energy, or it will be recharged prior to its energy expiration. Ideally, we define the charging throughput of the mobile charger to be the average functioning time of sensors during a charging tour. However, due to the dynamic nature of sensor activities, it is hard to get or predict the sensors' functioning time. To be practical, we here use the total number of sensors getting charged during a charging tour to represent the charging throughput of this charging tour. For an instance, in Fig. 5.1, there are total 10 sensors waiting for charging, and the mobile charger finally charges 8 of them before it returns to the base station. Thus, the charging throughput of this charging tour is 8. Note that the rest 2 sensors uncharged will keep stay in the waiting list, and the mobile charger will take them into consideration during next charging tours until they finally get charged.

5.3.3 Problem Statement

Given a time period T per tour by the mobile charger, the base station may receive many recharging requests, depending on the network scale and energy status of sensors. Let Q_c be the queue of recharging requests and V_c be the set of sensors to be charged, which are updated dynamically as recharging requests arrive one by one. Since the mobile charger takes time when it travels among the sensors, sometimes it may not be possible to charge all requested sensors per tour within time period T. The *charging throughput maximization problem* thus is to find a close tour for the mobile charger, such that the charging throughput is maximized, subject to the amount of time per tour being bounded by T. Specifically, assuming that all recharging requests from sensors $Q_c = \{(v_j, RE_j, r_j) | v_j \in V_c\}$ are given in advance, the *offline charging throughput maximization problem* can be defined as follows.

Given a set $V_c \subseteq V$ of sensors to be recharged, a tour $P = \{(v_j, t_j)\}_{j=0}^{m}$ is a sequence of pairs (v_j, t_j), where $v_j \in V_c \cup \{v_0\}$ and t_j is the arrival time when a mobile charger visits v_j. Noticing that v_0 is the depot of the mobile charger, the feasibility constraint for a tour is

$$t_0 = 0 \tag{5.1}$$

$$t_1 = t_0 + l(v_0, v_1) \tag{5.2}$$

$$t_{j+1} = t_j + C + l(v_j, v_{j+1}),\ 1 \le j < m \tag{5.3}$$

$$t_j \ge r_j,\ 1 \le j < m \tag{5.4}$$

$$t_m + C + l(v_m, v_0) \le T \tag{5.5}$$

where $l(v_j, v_{j+1})$ is the travel time of the mobile charger from v_j to v_{j+1}, C is a constant charging time, and T is a given finite-horizon time period. Constraint (5.4) ensures that a sensor should be charged only after it sends out a request. Constraint (5.5) ensures that the mobile charger will return to v_0 ultimately. The goal is to find a tour with the maximum charging throughput. Different from most existing studies, we here consider finding a charging tour to maximize the charging throughput, other than finding a charging tour to keep all sensors alive, since in reality it could be impossible to keep all sensors alive due to the limited resource/capability of the mobile charger.

5.3.4 NP-Hardness

We show that the offline charging throughput maximization problem is NP-hard by the following theorem.

Theorem 11 *The offline charging throughput maximization problem is NP-hard.*

Proof We show the claim by a reduction from a well-known NP-hard problem - the orienteering problem [33] which is defined as follows. Given n nodes in the Euclidean plane labeled from 1 to n and each with a score, find a route of the maximum score through these nodes beginning at 1 and ending at n of length (or duration) no greater than a given budget. Clearly, assuming that each recharging request is released at the beginning of the given time period T, it is easy to verify that this special case of the offline charging throughput maximization problem is equivalent to the defined orienteering problem. Hence, the offline charging throughput maximization problem is NP-hard too. □

5.4 Offline Approximation Algorithm

In this section, we first devise an approximation algorithm for the charging throughput maximization problem by assuming that all recharging requests in a given time period T are known in advance. We reduce the problem to the orienteering problem with time windows. The solution to the latter in turn returns an approximate solution to the former.

The orienteering problem with time windows is defined as follows. Given a directed edge weighted graph $G' = (V', A', l')$ with $l'(u, v)$ denoting the length of edge (u, v) from u to v and each node $v' \in V'$ having a time window $[R(v'), D(v')]$ during which it can only be visited no earlier than $R(v')$ and no later than $D(v')$ with $R(v') \leq D(v')$, two nodes $s, t \in V'$ and an integer budget $B > 0$, find an $s - t$ walk of length at most B to maximize the number of vertices covered. Chekuri et al. [17] proposed a recursive greedy algorithm for the orienteering problem.

In the following we reduce the problem of concern to the orienteering problem with time windows. Given a set V_c of sensors to be recharged, we construct a directed graph $G_c = (V'_c \cup \{v_0\}, A_c, l)$ with the budget $T > 0$, where the base station v_0 with a time window $[0, T]$ corresponds to the node s, and the base station v_0 also corresponds to the node t. For each node $v_i \in V_c$, there are two corresponding nodes v'_i with a time window $[r_i, T]$ and v''_i with a time window $[r_i + C, T]$ in V'_c, and an edge from v'_i to v''_i with $l(v'_i, v''_i) = C$, where r_i is the charging request release time of v_i and C is the charging time on v_i. Recall that $l(v_i, v_j)$ is the travel time of the mobile charger from $v_i \in V_c \cup \{v_0\}$ to $v_j \in V_c \cup \{v_0\}$. We then add edges from v_0 to each node $v'_i \in V'_c$ and let $l(v_0, v'_i) = l(v_0, v_i)$. We also add edges from each node $v''_i \in V'_c$ to v_0 and let $l(v''_i, v_0) = l(v_i, v_0)$. We finally add edges from each node $v''_i \in V'_c$ to each different node $v'_j \in V'_c - \{v'_i\}$ and let $l(v''_i, v'_j) = l(v_i, v_j)$. As a result, $G_c = (V'_c \cup \{v_0\}, A_c, l)$ is obtained, where $l(u, v)$ is the length of edge (u, v).

5.4.1 Algorithm

Algorithm 11 `Offline_Appro`$(v_s, v_e, t_s, t_e, V_c', r)$

Input: A directed edge weighted graph $G_c = (V_c \cup \{v_0, t_c\}, A_c, l)$ and a given time budget T.
Output: A tour P starts from v_0.

if $l(v_s, v_e) > t_e - t_s$ **then**
  /* It implies that the time budget is not enough even the mobile charger goes directly from v_s to v_e */
  return Infeasible;
end if;
$P \leftarrow < v_s, v_e >$;
if $r == 0$ **then**
  /* The recursive limit works*/
  return P;
end if;
/* $m(P)$ calculates the number of nodes covered by P */
$max \leftarrow m(P)$;
for each $v \in V_c$ **do**
  /* Guessing the middle node visited */
  $v_m \leftarrow v$;
  for $1 \leq T' \leq (t_e - t_s)$ **do**
    /* Guessing the time budget used */
    $T_m \leftarrow T'$;
    $P_{left} \leftarrow$ Offline_Appro$(v_s, v_m, t_s, t_s + T_m, V_c', r - 1)$;
    $P_{right} \leftarrow$ Offline_Appro$(v_m, v_e, t_s + T_m, t_e, V_c' - V(P_{left}), r - 1)$;
    if $m(P_{left} \cdot P_{right}) > max$ **then**
      /* Concatenation of the two separate tours */
      $P \leftarrow P_{left} \cdot P_{right}$;
      $max \leftarrow m(P_{left} \cdot P_{right})$;
    end if;
  end for;
end for;
return P.

The proposed approximation algorithm is as follows: It first guesses the middle node v_m in a tour of the mobile charger, and the amount of time consumed T_m within the time budget T by the mobile charger from v_0 to v_m, assuming that T is an integer. The guessing step is implemented by enumerating all candidate nodes as the middle node v_m as well as the possible value of T_m, $1 \leq T_m < T$. Notice that we can use standard scaling and rounding ideas to ensure that all values within the total time budget T are integers and polynomially bounded. It then recursively finds a tour P_{left} from v_0 to v_m with budget T_m, which means a tour P_{left} starts at v_0 at time 0 and has to reach v_m no later than time T_m. It also finds another tour P_{right} starting from v_m and ending at v_0 with the budget $T - T_m$ to augment the nodes that are not covered by P_{left}, which means a tour P_{right} starts at v_m no earlier than time T_m and has to reach v_0 at time T. It finally outputs the tour obtained by concatenating P_{left} and P_{right}. Let procedure Offline_Appro$(v_s, v_e, t_s, t_e, V_c', r)$ be used to implement the recursive greedy algorithm mentioned above, where v_s is the start node with starting time t_s, v_e is the end node with ending time t_e, and r indicates the depth of the recursion allowed. Note that v_s and v_e can be the same which implies a close tour. The details are described in **Algorithm** 11.

5.4.2 Complexity Analysis

We now analyze the properties of algorithm `offline_Appro` as follows.

Theorem 12 *Given a set V_c of sensors to be charged within a time period T in the defined wireless sensor network, there is an approximation algorithm* `offline_Appro` *for the offline charging throughput maximization problem with approximation ratio of $O(\log |V_c|)$, which takes $O((|V_c| \cdot T)^{\log |V_c|})$ time.*

Proof Following the classical results in [17], the theorem follows, omitted. □

5.5 Online Heuristics

So far we have provided an offline approximation algorithm by assuming that all recharging requests are given in advance. In reality, it is impossible to know the requests in advance until they are actually received. In the following we develop an online algorithm, where the recharging requests arrive over time. In other words, it is very likely that new recharging requests will be received when the mobile charger moves towards its next charging sensor or is charging the current sensor.

For this online version of the problem, a naive approach is to construct the tour of the mobile charger iteratively. That is, within each iteration, a new recharging request is added to the tour and the mobile charger will serve it. The sum of the traveling time and charging time of charging a sensor can be treated as the processing time of serving a recharging request. This will lead to an online algorithm `Online_SPT` [70]: choose one sensor with the shortest processing time from all available recharging requests. Specifically, assume that the mobile charger currently stays at the location of sensor v_i and finishes its charging. Recall that $l(v_i, v_j)$ is the travel time of the mobile charger from v_i to v_j, and C is the constant charging time. The amount of time for serving the recharging request c_j of sensor v_j is $l(v_i, v_j) + C + l(v_j, v_0) - l(v_i, v_0)$, where v_0 is the depot of the mobile charger. We thus choose a sensor to charge if its recharging request incurs the minimum amount of serving time. This procedure continues until the tour time constraint T is no longer met.

Notice that once the mobile charger visits and charges a sensor, the serving time cost of the mobile charger changes due to the change of the location of the mobile charger. Thus, the solution delivered by algorithm `Online_SPT` may be sub-optimal, which can be illustrated by Figure 5.2.

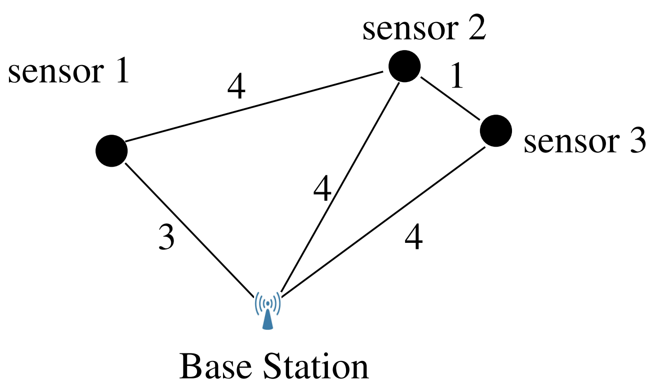


Figure 5.2: An example scenario where the time constraint T is 11, the constant charging time C is 1 and the travel time between nodes is as labeled.

In this example, all three sensors are waiting for charging, the SPT-rule based solution is: Base→1→Base, where only sensor 1 is charged. Notice that although sensor 2 requires longer serving time than sensor 1, it is much closer to sensor 3. Hence it is easy to verify a better solution: Base→2→3→Base, where both sensor 2 and 3 will be charged.

5.5.1 Improved Online Heuristic

Inspired by the illustrated example, we here propose a clustering-based algorithm, which takes both the serving time and the sensor locations into consideration. In general, the proposed algorithm proceeds iteratively. The mobile charger makes its next charging decision only when it finishes recharging the currently chosen sensors already. Within each iteration, it will charge a set of sensors instead of a single sensor. To this end, it first groups recharging requests into different 'clusters' according to the locations of the involving sensors, and then identifies a group as its next charging target with maximizing a metric (to be defined later).

Recall that V_c is the set of sensors to be charged which is updated dynamically. Specifically, within each iteration, for a given integer $K \leq |V_c|$, we first group all sensors to be charged based on their geographical locations, by adopting a well-known K-means clustering algorithm – Lloyd's algorithm [66], which aims to partition $|V_c|$ nodes into K clusters such that each node belongs to the cluster with the nearest mean. Let $V_1, V_2, ...V_K$ be the K clusters formed, where $V_1 \cup V_2 \cup ...V_K = V_c$. Assuming the mobile charger currently stays at the location of sensor v_a, for each cluster obtained, we then find a charging path for the mobile charger that starts from v_a, visits every node in the cluster exactly once and finally returns to the base station v_0 by adopting a MST heuristic for the Traveling Salesman Problem (TSP) [8]. A cluster V_i is a *feasible charging cluster* if the time spent on all previous charging and traveling T', plus the time spent for charging this cluster $|V_i| \cdot C$, and the relevant traveling time $l(V_i)$ is no more than T, i.e., $T' + |V_i| \cdot C + l(V_i) \leq T$, where $l(V_i)$ is the travel time to finish the relevant path from v_a to v_0. If no feasible charging cluster can be found, it implies that the value of K need to be adjusted. We then change the value of K iteratively by setting $K = \min\{\lfloor \beta \cdot K \rfloor, |V_c|\}$ and re-partition until a feasible charging cluster is found, where $\beta = 2$ is the adjusting rate which can also be set as any real number larger than 1. Denote by $\Delta gain(V_i) = \frac{|V_i|}{l(V_i) - l(v_a, v_0) + |V_i| \cdot C}$ the charging gain of cluster V_i. We finally choose a cluster with the maximum charging gain from all feasible clusters as the next charging cluster. That is, the mobile charger will start from v_a and charge all the sensors within the chosen cluster along the found path.

In summary, the algorithm proceeds iteratively. Initially, the mobile charger starts from the base station. Within each iteration, the mobile charger chooses a feasible charging cluster of sensors with maximum charging gain from the K clusters formed to charge. Once no feasible cluster is found, the K is then self-adjusted and re-evaluated iteratively until a feasible cluster is got. This procedure continues until the tour time constraint T is no longer met. The detailed algorithm `Online_K_Cluster` is described in **Algorithm** 12.

Algorithm 12 `Online_K_Cluster`

Input: A set V_c of sensors to be charged which varies over time, a given time period T, and a specified constant K.
Output: A tour P starts from base station v_0.

$P \leftarrow < v_0 >$; $K_{init} \leftarrow K$;
$v_a \leftarrow v_0$; /* the current location of the mobile charger */
$t \leftarrow 0$; /* the current time */
while $t \leq T$ **do**
  Apply a K-means clustering algorithm to partition V_c into K clusters: $V_1, V_2, ...V_K$;
  For each cluster, find a path from v_a that visits every node within this cluster and finally returns to v_0 by adopting a MST heuristic for TSP problem;
  Once no feasible cluster is found, then adjust K by setting $K = \min\{2K, |V_c|\}$ and repartition.
  if $K == |V_c|$ and no feasible cluster found **then**
    Break; /* the mobile charger return to v_0 */
  end if;
  Calculate charging gain for each feasible cluster;
  /* Assuming cluster V_i has maximum charging gain, the mobile charger then goes to charge sensors in this cluster by following the found path */
  Add the charged sensors in P;
  Update sensor set V_c, v_a and t accordingly;
  $K \leftarrow K_{init}$; /* reset K for next iteration */
end while;
return P.

Theorem 13 *Given a time period T per tour and an integer K in a wireless sensor network, there is an online algorithm* `Online_K_Cluster` *for the charging throughput maximization problem, which takes $O(|V|^2 \cdot \log |V| \cdot T)$ time, where $|V|$ is the total number of sensors.*

Proof Clearly, the algorithm `Online_K_Cluster` yields a feasible solution to the charging throughput maximization problem. We then analyze the time complexity in the following. Within each iteration, applying Lloyd's algorithm takes $O(|V_c| \cdot K \cdot l)$ time, where V_c is the set of sensors to be charged and l represents the number of sub-iterations. Calculating the charging gain for a cluster takes $O(|V_c|^2)$. As the value of K may need to be adjusted by setting $K = \min\{2K, |V_c|\}$ and l can be bounded by a pre-defined constant, finding a feasible cluster with maximum charging gain takes $O(|V_c|^2 \cdot \log |V_c|)$ time. It is easy to verify that the number of iterations is bounded by T. The algorithm thus takes $O(|V_c|^2 \cdot \log |V_c| \cdot T) = O(|V|^2 \cdot \log |V| \cdot T)$ time, since $|V_c| \leq |V|$. $\square$

5.6 Performance Study

In this section we evaluate the performance of the proposed algorithms through experimental simulation. We also study the impact of the cluster parameter K on algorithm performance.

5.6.1 Experimental Environment Setting

Two different scale networks are considered in our experiments. One is a small-scale network consisting of 10 to 30 sensors randomly deployed in a $50m \times 50m$ square area, and another is a large-scale network consisting of 100 to $1,000$ sensors randomly deployed in a $500m \times 500m$ square area. The base station (the depot of the mobile charger) is located at one corner of the square. Due to the dynamic nature of sensing activity, each sensor randomly sends its recharging requests within a given time period T. Without loss of generality, we here set $T = 30s$ for a small scale network, and also set the time period for a large scale network at $T = 1,800s$ and $T = 3,600s$, respectively. We further assume that the default constant charging time for each sensor is $2s$, and the mobile charger travels at a constant speed $8m/s$. Each value in figures is the mean of the results by applying each mentioned algorithm to 30 different network topologies of the same network size.

5.6.2 Performance Evaluation of Both Offline Approximation and Online Heuristic Algorithms

In this subsection we first evaluate the performance of the offline approximation algorithm as well as two proposed online heuristics `Online_SPT` and `Online_K_Cluster` in small-scale networks by varying the network size from 10 to 30 and setting the cluster parameter $K = 3$, while the time period T is fixed at $30s$.

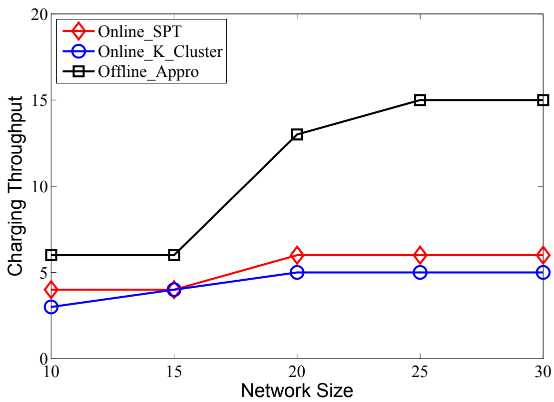
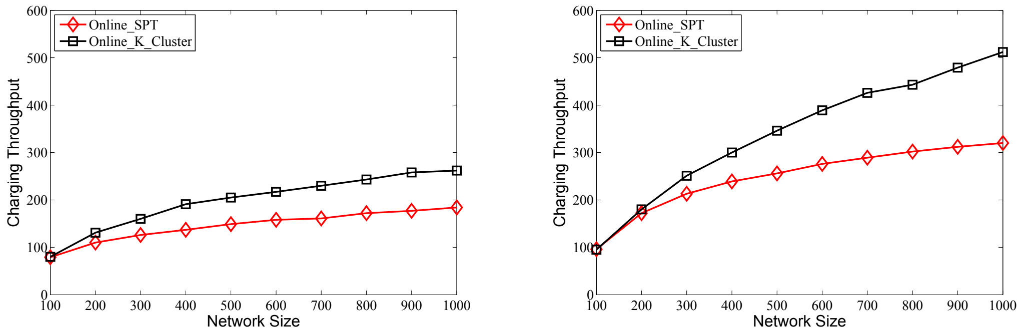


Figure 5.3: The charging throughput performance of both offline approximation and online heuristic algorithms.

Fig. 5.3 clearly shows that offline algorithm `Offline_Appro` outperforms the two online heuristics `Online_SPT` and `Online_K_Cluster` significantly. With the increase on network size, the performance gap becomes larger. The reason behind is that the offline algorithm has the perfect information of all requests. and use a nearly exhaustive search method. Obviously, when there is small-scale recharging requests workload and the global knowledge is available (e.g., by prediction), the offline algorithm is the best choice. However, the offline algorithm is very computationally expensive which makes it impractical in large-scale workload scenario.

5.6.3 Performance Evaluation of Online Heuristic Algorithms

In this subsection we investigate the performance of two online heuristics `Online_SPT` and `Online_K_Cluster` in large-scale networks by varying the network size from 100 to 1,000 and setting the cluster parameter K at 5, while the time period T is fixed at



(a) $T = 1,800s$ (b) $T = 3,600s$

Figure 5.4: The charging throughput performance of online algorithms by varying the network size and setting the given time period T at 1,800s and 3,600s.

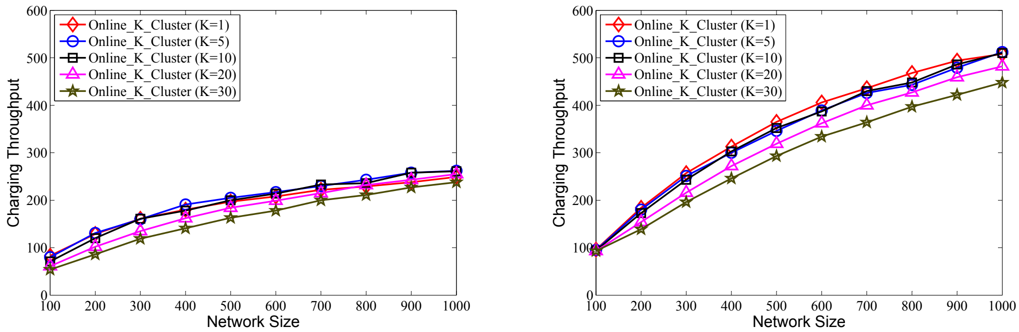
$1,800s$ and $3,600s$, respectively.

Fig. 5.4 demonstrates that the charging throughput of algorithm `Online_K_Cluster` outperforms that of `Online_SPT` with the increase of the network size. For example, in Fig. 5.4(a), when the network size is greater than 100 and T is 1,800s, the charging throughput of `Online_K_Cluster` is at least 20% more than that of `Online_SPT`. When the network size becomes larger, the performance gap also grows upto around 47%. Similarly, in Fig. 5.4(b), when the network size is greater than 200 and T is 3,600s, the charging throughput of `Online_K_Cluster` is at least 19% more than that of `Online_SPT`. It also can be noticed that with a larger time period T, the charging throughput of both `Online_SPT` and `Online_K_Cluster` is increased, as the mobile charger has more time available to serve the recharging requests.

5.6.4 Impact of Cluster Parameter K on Charging Throughput Performance

We finally study the impact of the cluster parameter K on the performance of algorithm `Online_K_Cluster` by setting K at 1, 5, 10, 20, and 30, while the network size varies from 100 to 1,000 and the time period T is fixed at $1,800s$ and $3,600s$, respectively.

From Fig. 5.5, it can be seen that the charging throughput of algorithm `Online_K_Cluster` with $K = 1$, $K = 5$ and $K = 10$ is at the same level, while algorithm `Online_K_Cluster`



(a) $T = 1,800s$ (b) $T = 3,600s$

Figure 5.5: The impact of cluster parameter K by varying the network size n and setting the tolerant delay T at 1,800s and 3,600s.

with $K = 30$ delivers the worst performance. With the network size grows, the performance gap becomes smaller. Specifically, in Fig. 5.5(a), the charging throughput of algorithm `Online_K_Cluster` with $K = 5$ outperforms that of algorithm `Online_K_Cluster` with $K = 1$ and $K = 10$ slightly, and is more than at least 25% and 19% compared with that of algorithm `Online_K_Cluster` with $K = 20$ and $K = 30$ when the network size is less than 800, respectively. Fig. 5.5(b) also exhibits the similar performance behavior in which algorithm `Online_K_Cluster` with $K = 1$ outperforms algorithm `Online_K_Cluster` with $K = 5$ and $K = 10$ slightly, omitted. In general, the charging throughput of algorithm `Online_K_Cluster` decreases when the K value is too large. In order to achieve a best charging throughput, a proper K should be assigned according to the network size and the tour time bound.

5.7 Conclusions

In this chapter we have studied the problem of finding an optimal close trajectory for a mobile charger in renewable sensor networks, subject to the time duration constraint of the mobile charger per tour. We formulated the problem as a charging throughput maximization problem with an aim of maximizing the number of sensors charged per tour. Due to the NP-hardness of the problem, we then proposed an offline approximation algorithm and two online heuristics. Finally, we evaluated the performance of the proposed algorithms through experimental simulation, and provided numerical results to validate the efficiency of the proposed algorithms. Nevertheless, our work mainly focuses on maximizing the number of sensors charged, which may result in biased charging behavior in some edge cases (e.g. some sensors that are far from the base station or sparsely located have few opportunities to be charged). We will extend our work in future by considering fairness issues as well.

Chapter 6

Conclusions and Future Work

This chapter summarizes the contributions we made in this thesis, followed by discussing several potential research topics derived from this work.

6.1 Summary of Contributions

Several key issues of deploying renewable sensor networks for sustainable monitoring were studied in this thesis. New concepts, models, optimization techniques, and implementations were proposed and evaluated for renewable sensor networks to achieve unattended and continuing quality-aware services. As almost all the formulated problems are NP-hard, approximate solutions with guaranteed performance ratios for gathering data from sensor nodes and replenishing energy to sensor nodes efficiently were developed. Fast and scalable algorithms were devised by exploiting the combinatorial property of resource optimization problems (target coverage, mobile data collection, and energy replenishment). The main contributions of this thesis are summarized as follows.

- We investigated existing energy harvesting prediction approaches. Specifically, we investigated the accuracy of the energy harvesting prediction approach `VEWMA` in comparison with the one of a basic prediction approach `EWMA`, using the real solar data profiles obtained from The National Solar Radiation Data Base in the States [7], which contain the most comprehensive collection of solar data for public access.
- We dealt with the coverage quality efficiency in renewable sensor networks.

We introduced a new metric that is a weighted linear combination of two sub-modular utility functions to measure the coverage quality within different time scales. Based on the proposed metric, we devised an offline algorithm `Greedy_Heuristic` and its distributed implementation `Distributed_Implement` to schedule sensors' duty-cycles within the given energy budget to maximize the coverage quality. We also proposed an adaptive framework `Adaptive_Framework` to deal with harvesting energy prediction fluctuations, and showed that under this adaptive framework, the proposed centralized and distributed algorithms are still applicable.

- We addressed the optimization of data collection by sensor networks in two application scenarios. We first studied the data collection maximization problem in a renewable sensor network with a path-constrained mobile sink, and proposed an offline approximation algorithm `Offline_Appro` and an online distributed algorithm `Online_Appro` to schedule sensors transmitting their data to the mobile sink, through incorporating time-varying sensor energy budgets and employing multi-rate wireless communications. We also investigated the data quality maximization problem in a renewable sensor network, where a mobile sink with controlled mobility is employed for data gathering, and developed a centralized algorithm `Max_Utility` and its distributed implementation `Dis_Max_Utility` for the problem, which find a close trajectory for the mobile sink and schedule the sojourn time at each sojourn location.

- We studied the energy provisioning for renewable sensor networks, by utilizing wireless energy transfer technology. We formulated the charging throughput maximization problem in a renewable sensor network where a mobile charger travels around the sensing field to replenish sensors with energy. We reduced the problem to the orienteering problem with time windows, and proposed an offline approximation algorithm. We also developed two online heuristics `Online_SPT` and `Online_K_Cluster` for it, which construct the tour of the mobile charger iteratively.

- We conducted extensive experiments by simulation to evaluate all proposed algorithms, including investigating the impact of constraint parameters on the performance of the algorithms, and comparing their performance with that of comparable algorithms. Experimental results showed that the proposed algorithms are very promising, in terms of coverage quality, network throughput, network data quality, and charging throughput.

6.2 Future Work

There are several potential research topics that can be explored based on the work in this thesis.

- The use of multiple mobile sinks/chargers can be investigated to further improve the network performance. It is challenging to arrange the motion of multiple sinks/chargers jointly. For example, with a given number of sinks, design multiple closed paths for them to maximize the data quality, subject to the motion constraints. Or, determine the minimum number of motion-constrained chargers and find their trajectories to ensure that none of sensors runs out of energy.

- Several issues such as latency and mobility constraints which are coupled with exploring sink/charger mobility can be considered, to deliver more efficient data gathering or energy recharging solutions. For an instance, the latency of delivering data from sinks to the data center can be considered as an essential performance metric for data collection. Or, jointly considering scheduling and advanced mobility model will enable the developed solution to be more practical.

- The cross-layer design should be researched in the future. Taking into account the techniques in other layers (e.g., MAC layer) will make the solutions more effective and efficient in improving network performance. Moreover, joint optimizing approaches of multiple layers will eliminate any improper assumption raised by designing algorithms from the aspect of a single layer, and also enable the developed solutions more practical in real applications.

References

1. IF. Akyildiz, Weilian Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *Communications Magazine, IEEE*, 40(8):102–114, Aug 2002. (cited on page 1)

2. Y. Alayev, Fangfei Chen, Yun Hou, M.P. Johnson, A. Bar-Noy, T. La Porta, and K.K. Leung. Throughput maximization in mobile wsn scheduling with power control and rate selection. In *Distributed Computing in Sensor Systems (DCOSS), 2012 IEEE 8th International Conference on*, pages 33–40, May 2012. (cited on page 54)

3. H.M. Ammari and S.K. Das. Centralized and clustered k-coverage protocols for wireless sensor networks. *Computers, IEEE Transactions on*, 61(1):118–133, Jan 2012. (cited on pages 18 and 19)

4. Constantinos Marios Angelopoulos, Sotiris Nikoletseas, Theofanis P. Raptis, Christoforos Raptopoulos, and Filippos Vasilakis. Efficient energy management in wireless rechargeable sensor networks. In *Proceedings of the 15th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, MSWiM '12, pages 309–316, New York, NY, USA, 2012. ACM. (cited on pages 10 and 101)

5. R. Bar-Yehuda and S. Even. A local-ratio theorem for approximating the weighted vertex cover problem. In G. Ausiello and M. Lucertini, editors, *Analysis and Design of Algorithms for Combinatorial Problems*, volume 109 of *North-Holland Mathematics Studies*, pages 27 – 45. North-Holland, 1985. (cited on page 58)

6. Stefano Basagni, Alessio Carosi, Emanuel Melachrinoudis, Chiara Petrioli, and Z.Maria Wang. Controlled sink mobility for prolonging wireless sensor net-

works lifetime. *Wireless Networks*, 14(6):831–858, 2008. (cited on pages 48 and 79)

7. The National Solar Radiation Data Base. $http : //rredc.nrel.gov/solar/old_data/nsrdb$. (cited on pages 11, 40, and 117)

8. K. Bharath-Kumar and J.M. Jaffe. Routing to multiple destinations in computer networks. *Communications, IEEE Transactions on*, 31(3):343–351, Mar 1983. (cited on page 110)

9. L. Boloni, D. Turgut, S. Basagni, and C. Petrioli. Scheduling data transmissions of underwater sensor nodes for maximizing value of information. In *Global Communications Conference (GLOBECOM), 2013 IEEE*, pages 438–443, Dec 2013. (cited on page 9)

10. L. Boloni, D. Turgut, S. Basagni, and C. Petrioli. Scheduling data transmissions of underwater sensor nodes for maximizing value of information. In *Global Communications Conference (GLOBECOM), 2013 IEEE*, pages 438–443, Dec 2013. (cited on page 50)

11. A. Cammarano, C. Petrioli, and D. Spenza. Pro-energy: A novel energy prediction model for solar and wind energy-harvesting wireless sensor networks. In *Mobile Adhoc and Sensor Systems (MASS), 2012 IEEE 9th International Conference on*, pages 75–83, Oct 2012. (cited on page 20)

12. M. Cardei, M.T. Thai, Yingshu Li, and Weili Wu. Energy-efficient target coverage in wireless sensor networks. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 3, pages 1976–1984 vol. 3, March 2005. (cited on page 7)

13. M. Cardei, M.T. Thai, Yingshu Li, and Weili Wu. Energy-efficient target coverage in wireless sensor networks. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 3, pages 1976–1984 vol. 3, March 2005. (cited on pages 15 and 18)

14. M. Cardei, Jie Wu, Mingming Lu, and M.O. Pervaiz. Maximum network lifetime in wireless sensor networks with adjustable sensing ranges. In *Wireless And Mobile Computing, Networking And Communications, 2005. (WiMob'2005), IEEE International Conference on*, volume 3, pages 438–445 Vol. 3, Aug 2005. (cited on page 6)

15. Mihaela Cardei and Jie Wu. Energy-efficient coverage problems in wireless ad-hoc sensor networks. *Comput. Commun.*, 29(4):413–420, February 2006. (cited on page 18)

16. Arnab Chakrabarti, Ashutosh Sabharwal, and Behnaam Aazhang. Communication power optimization in a sensor network with a path-constrained mobile observer. *ACM Trans. Sen. Netw.*, 2(3):297–324, August 2006. (cited on pages 9, 50, and 51)

17. C. Chekuri and M. Pal. A recursive greedy algorithm for walks in directed graphs. In *Foundations of Computer Science, 2005. FOCS 2005. 46th Annual IEEE Symposium on*, pages 245–253, Oct 2005. (cited on pages 106 and 108)

18. Ai Chen, S. Kumar, and T.H. Lai. Local barrier coverage in wireless sensor networks. *Mobile Computing, IEEE Transactions on*, 9(4):491–504, April 2010. (cited on page 6)

19. W.C. Cheng, Cheng-Fu Chou, L. Golubchik, S. Khuller, and Yung-Chun Wan. A coordinated data collection approach: design, evaluation, and comparison. *Selected Areas in Communications, IEEE Journal on*, 22(10):2004–2018, Dec 2004. (cited on page 8)

20. Reuven Cohen, Liran Katzir, and Danny Raz. An efficient approximation for the generalized assignment problem. *Inf. Process. Lett.*, 100(4):162–166, November 2006. (cited on pages 57, 58, and 60)

21. Haipeng Dai, Lintong Jiang, Xiaobing Wu, D.K.Y. Yau, Guihai Chen, and Shaojie Tang. Near optimal charging and scheduling scheme for stochastic event capture with rechargeable sensors. In *Mobile Ad-Hoc and Sensor Systems (MASS),*

2013 IEEE 10th International Conference on, pages 10–18, Oct 2013. (cited on pages 10 and 101)

22. Haipeng Dai, Xiaobing Wu, Lijie Xu, and Guihai Chen. Practical scheduling for stochastic event capture in wireless rechargeable sensor networks. In *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, pages 986–991, April 2013. (cited on pages 7, 15, and 19)

23. Mario Di Francesco, Sajal K. Das, and Giuseppe Anastasi. Data collection in wireless sensor networks with mobile elements: A survey. *ACM Trans. Sen. Netw.*, 8(1):7:1–7:31, August 2011. (cited on pages 8, 9, and 50)

24. Ling Ding, Weili Wu, J. Willson, Lidong Wu, Zaixin Lu, and Wonjun Lee. Constant-approximation for target coverage problem in wireless sensor networks. In *INFOCOM, 2012 Proceedings IEEE*, pages 1584–1592, March 2012. (cited on pages 7 and 15)

25. P. Dutta, J. Hui, Jaein Jeong, Sukun Kim, C. Sharp, J. Taneja, G. Tolle, K. Whitehouse, and D. Culler. Trio: enabling sustainable and scalable outdoor wireless sensor network deployments. In *Information Processing in Sensor Networks, 2006. IPSN 2006. The Fifth International Conference on*, pages 407–415, 2006. (cited on page 2)

26. Kai-Wei Fan, Zizhan Zheng, and Prasun Sinha. Steady and fair rate allocation for rechargeable sensors in perpetual sensor networks. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*, SenSys '08, pages 239–252, New York, NY, USA, 2008. ACM. (cited on pages 9, 50, and 99)

27. Lingkun Fu, Peng Cheng, Yu Gu, Jiming Chen, and Tian He. Minimizing charging delay in wireless rechargeable sensor networks. In *INFOCOM, 2013 Proceedings IEEE*, pages 2922–2930, April 2013. (cited on pages 10 and 101)

28. Shuai Gao, Hongke Zhang, and S. Das. Efficient data collection in wireless sensor networks with path-constrained mobile sinks. In *World of Wireless, Mo-*

bile and Multimedia Networks Workshops, 2009. WoWMoM 2009. IEEE International Symposium on a, pages 1–9, June 2009. (cited on page 9)

29. Shuai Gao, Hongke Zhang, and S.K. Das. Efficient data collection in wireless sensor networks with path-constrained mobile sinks. *Mobile Computing, IEEE Transactions on*, 10(4):592–608, April 2011. (cited on pages 9, 50, and 51)

30. Michael R. Garey and David S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1990. (cited on pages 85 and 87)

31. B. Gedik, Ling Liu, and P.S. Yu. Asap: An adaptive sampling approach to data collection in sensor networks. *Parallel and Distributed Systems, IEEE Transactions on*, 18(12):1766–1783, Dec 2007. (cited on page 8)

32. Philipp Maria Glatz, Leander Bernd Hörmann, Christian Steger, and Reinhold Weiss. Designing perpetual energy harvesting systems explained with rivermote: A wireless sensor network platform for river monitoring. In *EJSE Special Issue: Wireless Sensor Networks and Practical Applications*, 2010. (cited on page 2)

33. Bruce L. Golden, Larry Levy, and Rakesh Vohra. The orienteering problem. *Naval Research Logistics (NRL)*, 34(3):307–318, 1987. (cited on page 105)

34. Evren Güney, Necati Aras, İ. Kuban Altinel, and Cem Ersoy. Efficient integer programming formulations for optimum sink location and routing in heterogeneous wireless sensor networks. *Comput. Netw.*, 54(11):1805–1822, August 2010. (cited on page 80)

35. Songtao Guo and Yuanyuan Yang. A distributed optimal framework for mobile data gathering with concurrent data uploading in wireless sensor networks. In *INFOCOM, 2012 Proceedings IEEE*, pages 1305–1313, March 2012. (cited on page 9)

36. H. Gupta, Zongheng Zhou, S.R. Das, and Q. Gu. Connected sensor cover: self-organization of sensor networks for efficient query execution. *Networking, IEEE/ACM Transactions on*, 14(1):55–67, Feb 2006. (cited on page 7)

37. Himanshu Gupta, Samir R. Das, and Quinyi Gu. Connected sensor cover: Self-organization of sensor networks for efficient query execution. In *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking &Amp; Computing*, MobiHoc '03, pages 189–200, New York, NY, USA, 2003. ACM. (cited on pages 7, 15, 18, and 41)

38. Liang He, Yu Gu, Jianping Pan, and Ting Zhu. On-demand charging in wireless sensor networks: Theories and applications. In *Mobile Ad-Hoc and Sensor Systems (MASS), 2013 IEEE 10th International Conference on*, pages 28–36, Oct 2013. (cited on pages 10, 101, 102, and 103)

39. Liang He, Jianping Pan, and Jingdong Xu. A progressive approach to reducing data collection latency in wireless sensor networks with mobile elements. *Mobile Computing, IEEE Transactions on*, 12(7):1308–1320, July 2013. (cited on page 9)

40. W.B. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. *Wireless Communications, IEEE Transactions on*, 1(4):660–670, Oct 2002. (cited on page 8)

41. Sushant Jain, Rahul C. Shah, Waylon Brunette, Gaetano Borriello, and Sumit Roy. Exploiting mobility for energy efficient data collection in wireless sensor networks. *Mob. Netw. Appl.*, 11(3):327–339, June 2006. (cited on page 9)

42. I. Jawhar and N. Mohamed. A hierarchical and topological classification of linear sensor networks. In *Wireless Telecommunications Symposium, 2009. WTS 2009*, pages 1–8, April 2009. (cited on page 48)

43. Byoungwoo Kang and Gerbrand Ceder. Battery materials for ultrafast charging and discharging. *Nature*, 458(7235):190–193, 2009. (cited on pages 100 and 103)

44. Aman Kansal, Jason Hsu, Sadaf Zahedi, and Mani B. Srivastava. Power management in energy harvesting sensor networks. *ACM Trans. Embed. Comput. Syst.*, 6(4), September 2007. (cited on pages 20, 21, 36, 48, 53, and 99)

45. Aman Kansal, Arun A. Somasundara, David D. Jea, Mani B. Srivastava, and Deborah Estrin. Intelligent fluid infrastructure for embedded networks. In *Pro-*

ceedings of the 2Nd International Conference on Mobile Systems, Applications, and Services, MobiSys '04, pages 111–124, New York, NY, USA, 2004. ACM. (cited on pages 9 and 50)

46. P.S. Khairnar and N.B. Mehta. Power and discrete rate adaptation for energy harvesting wireless nodes. In *Communications (ICC), 2011 IEEE International Conference on*, pages 1–5, June 2011. (cited on page 54)

47. Sukun Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon. Health monitoring of civil infrastructures using wireless sensor networks. In *Information Processing in Sensor Networks, 2007. IPSN 2007. 6th International Symposium on*, pages 254–263, April 2007. (cited on page 48)

48. B. Krishnamachari, D. Estrin, and S. Wicker. The impact of data aggregation in wireless sensor networks. In *Distributed Computing Systems Workshops, 2002. Proceedings. 22nd International Conference on*, pages 575–578, 2002. (cited on page 10)

49. André Kurs, Aristeidis Karalis, Robert Moffatt, J. D. Joannopoulos, Peter Fisher, and Marin Soljačić. Wireless power transfer via strongly coupled magnetic resonances. *Science*, 317(5834):83–86, 2007. (cited on pages 10 and 99)

50. Eugene L. Lawler. Fast approximation algorithms for knapsack problems. *Mathematics of Operations Research*, 4(4):339–356, 1979. (cited on page 60)

51. Zi Li, Yang Peng, Wensheng Zhang, and Daji Qiao. J-roc: A joint routing and charging scheme to prolong sensor network lifetime. In *Proceedings of the 2011 19th IEEE International Conference on Network Protocols*, ICNP '11, pages 373–382, Washington, DC, USA, 2011. IEEE Computer Society. (cited on pages 10, 101, and 103)

52. Weifa Liang and Jun Luo. Network lifetime maximization in sensor networks with multiple mobile sinks. In *Local Computer Networks (LCN), 2011 IEEE 36th Conference on*, pages 350–357, Oct 2011. (cited on pages 9, 48, 50, 77, and 80)

53. Weifa Liang, Jun Luo, and Xu Xu. Prolonging network lifetime via a controlled mobile sink in wireless sensor networks. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–6, Dec 2010. (cited on pages 48 and 80)

54. Weifa Liang, Jun Luo, and Xu Xu. Network lifetime maximization for time-sensitive data gathering in wireless sensor networks with a mobile sink. *Wireless Communications and Mobile Computing*, 13(14):1263–1280, 2013. (cited on page 80)

55. Weifa Liang, Xiaojiang Ren, Xiaohua Jia, and Xu Xu. Monitoring quality maximization through fair rate allocation in harvesting sensor networks. *Parallel and Distributed Systems, IEEE Transactions on*, 24(9):1827–1840, Sept 2013. (cited on pages 9, 50, and 99)

56. Weifa Liang, P. Schweitzer, and Zichuan Xu. Approximation algorithms for capacitated minimum forest problems in wireless sensor networks with a mobile sink. *Computers, IEEE Transactions on*, 62(10):1932–1944, Oct 2013. (cited on pages 9, 50, and 51)

57. Longbi Lin, N.B. Shroff, and R. Srikant. Asymptotically optimal energy-aware routing for multihop wireless networks with renewable energy sources. *Networking, IEEE/ACM Transactions on*, 15(5):1021–1034, Oct 2007. (cited on pages 20 and 52)

58. Changlei Liu and Guohong Cao. Distributed critical location coverage in wireless sensor networks with lifetime constraint. In *INFOCOM, 2012 Proceedings IEEE*, pages 1314–1322, March 2012. (cited on pages 6, 7, 15, and 22)

59. Chong Liu, Kui Wu, and Jian Pei. An energy-efficient data collection framework for wireless sensor networks by exploiting spatiotemporal correlation. *Parallel and Distributed Systems, IEEE Transactions on*, 18(7):1010–1023, July 2007. (cited on page 8)

60. Hai Liu, Xiaohua Jia, Peng-Jun Wan, Chih-Wei Yi, S.K. Makki, and N. Pissinou. Maximizing lifetime of sensor surveillance systems. *Networking, IEEE/ACM*

Transactions on, 15(2):334–345, April 2007. (cited on pages 7 and 15)

61. Ren-Shiou Liu, Kai-Wei Fan, Zizhan Zheng, and P. Sinha. Perpetual and fair data collection for environmental energy harvesting sensor networks. *Networking, IEEE/ACM Transactions on*, 19(4):947–960, Aug 2011. (cited on pages 9, 50, and 96)

62. Ren-Shiou Liu, Kai-Wei Fan, Zizhan Zheng, and P. Sinha. Perpetual and fair data collection for environmental energy harvesting sensor networks. *Networking, IEEE/ACM Transactions on*, 19(4):947–960, Aug 2011. (cited on page 20)

63. Ren-Shiou Liu, P. Sinha, and C.E. Koksal. Joint energy management and resource allocation in rechargeable sensor networks. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9, March 2010. (cited on pages 9, 50, and 70)

64. Yuzhen Liu and Weifa Liang. Approximate coverage in wireless sensor networks. In *Local Computer Networks, 2005. 30th Anniversary. The IEEE Conference on*, pages 68–75, Nov 2005. (cited on pages 7, 15, 18, and 19)

65. Yuzhen Liu and Weifa Liang. Prolonging network lifetime for target coverage in sensor networks. In *Wireless Algorithms, Systems, and Applications, Third International Conference, WASA 2008, Dallas, TX, USA, October 26-28, 2008. Proceedings*, pages 212–223, 2008. (cited on page 19)

66. S. Lloyd. Least squares quantization in pcm. *Information Theory, IEEE Transactions on*, 28(2):129–137, Mar 1982. (cited on page 110)

67. Jun Luo and J.-P. Hubaux. Joint mobility and routing for lifetime elongation in wireless sensor networks. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 3, pages 1735–1746 vol. 3, March 2005. (cited on page 48)

68. Jun Luo, Jacques Panchard, MichałPiórkowski, Matthias Grossglauser, and Jean-Pierre Hubaux. Mobiroute: Routing towards a mobile sink for improving lifetime in sensor networks. In *Proceedings of the Second IEEE International Conference*

on Distributed Computing in Sensor Systems, DCOSS'06, pages 480–497, Berlin, Heidelberg, 2006. Springer-Verlag. (cited on page 79)

69. Ming Ma and Yuanyuan Yang. Data gathering in wireless sensor networks with mobile collectors. In *Parallel and Distributed Processing, 2008. IPDPS 2008. IEEE International Symposium on*, pages 1–9, April 2008. (cited on pages 8 and 77)

70. Weizhen Mao, RexK. Kincaid, and Adam Rifkin. On-line algorithms for a single machine scheduling problem. In StephenG. Nash, Ariela Sofer, WilliamR. Stewart, and EdwardA. Wasil, editors, *The Impact of Emerging Technologies on Computer Science and Operations Research*, volume 4 of *Operations Research/Computer Science Interfaces Series*, pages 157–173. Springer US, 1995. (cited on page 109)

71. Silvio Micali and Vijay V. Vazirani. An o(v, v, c , e,) algoithm for finding maximum matching in general graphs. In *Proceedings of the 21st Annual Symposium on Foundations of Computer Science*, SFCS '80, pages 17–27, Washington, DC, USA, 1980. IEEE Computer Society. (cited on page 68)

72. D. Musiani, K. Lin, and T.S. Rosing. Active sensing platform for wireless structural health monitoring. In *Information Processing in Sensor Networks, 2007. IPSN 2007. 6th International Symposium on*, pages 390 –399, april 2007. (cited on page 2)

73. Hidehisa Nakayama, Nirwan Ansari, Abbas Jamalipour, and Nei Kato. Fault-resilient sensing in wireless sensor networks. *Comput. Commun.*, 30(11-12):2375–2384, September 2007. (cited on page 77)

74. Dong Kun Noh, Lili Wang, Yong Yang, Hieu Khac Le, and Tarek Abdelzaher. Minimum variance energy allocation for a solar-powered sensor system. In *Proceedings of the 5th IEEE International Conference on Distributed Computing in Sensor Systems*, DCOSS '09, pages 44–57, Berlin, Heidelberg, 2009. Springer-Verlag. (cited on page 21)

75. Yang Peng, Zi Li, Wensheng Zhang, and D. Qiao. Prolonging sensor network lifetime through wireless charging. In *Real-Time Systems Symposium (RTSS), 2010*

IEEE 31st, pages 129–139, Nov 2010. (cited on pages 10 and 101)

76. Volodymyr Pryyma, Damla Turgut, and Ladislau Bölöni. Active time scheduling for rechargeable sensor networks. *Comput. Netw.*, 54(4):631–640, March 2010. (cited on pages 7 and 15)

77. Xiaojiang Ren and Weifa Liang. Delay-tolerant data gathering in energy harvesting sensor networks with a mobile sink. In *Global Communications Conference (GLOBECOM), 2012 IEEE*, pages 93–99, Dec 2012. (cited on page 99)

78. Xiaojiang Ren and Weifa Liang. The use of a mobile sink for quality data collection in energy harvesting sensor networks. In *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, pages 1145–1150, April 2013. (cited on pages 70 and 99)

79. Xiaojiang Ren, Weifa Liang, and Wenzheng Xu. Use of a mobile sink for maximizing data collection in energy harvesting sensor networks. In *Parallel Processing (ICPP), 2013 42nd International Conference on*, pages 439–448, Oct 2013. (cited on page 99)

80. Anna Scaglione and Sergio D. Servetto. On the interdependence of routing and data compression in multi-hop sensor networks. In *Proceedings of the 8th Annual International Conference on Mobile Computing and Networking*, MobiCom '02, pages 140–147, New York, NY, USA, 2002. ACM. (cited on page 8)

81. C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379–423, 1948. (cited on page 54)

82. Yi Shi and Y.T. Hou. Theoretical results on base station movement problem for sensor network. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages –, April 2008. (cited on pages 48 and 77)

83. Yi Shi, Liguang Xie, Y.T. Hou, and H.D. Sherali. On renewable sensor networks with wireless energy transfer. In *INFOCOM, 2011 Proceedings IEEE*, pages 1350–1358, April 2011. (cited on pages 10, 101, and 103)

84. AA Somasundara, A Kansal, D.D. Jea, D. Estrin, and M.B. Srivastava. Controllably mobile infrastructure for low energy embedded networks. *Mobile Computing, IEEE Transactions on*, 5(8):958–973, Aug 2006. (cited on pages 9 and 50)

85. S. Sudevalayam and P. Kulkarni. Energy harvesting sensor nodes: Survey and implications. *Communications Surveys Tutorials, IEEE*, 13(3):443 –461, quarter 2011. (cited on page 2)

86. Shaojie Tang, Mo Li, Xinfa Shen, Jianhui Zhang, Guojun Dai, and S.K. Das. Cool: On coverage with solar-powered sensors. In *Distributed Computing Systems (ICDCS), 2011 31st International Conference on*, pages 488–496, June 2011. (cited on pages 7, 15, 19, and 25)

87. Shaojie Tang and Jing Yuan. Damson: On distributed sensing scheduling to achieve high quality of monitoring. In *INFOCOM, 2013 Proceedings IEEE*, pages 155–159, April 2013. (cited on page 19)

88. Xueyan Tang and Jianliang Xu. Adaptive data collection strategies for lifetime-constrained wireless sensor networks. *Parallel and Distributed Systems, IEEE Transactions on*, 19(6):721–734, June 2008. (cited on page 8)

89. CC2420 RF Transceiver. $http://www.ti.com/lit/gpn/cc2420$. (cited on page 40)

90. CC2500 RF Transceiver. $http://www.ti.com/lit/ds/symlink/cc2500.pdf$. (cited on page 54)

91. D. Turgut and L. Boloni. Ive: Improving the value of information in energy-constrained intruder tracking sensor networks. In *Communications (ICC), 2013 IEEE International Conference on*, pages 6360–6364, June 2013. (cited on page 50)

92. Damla Turgut and Ladislau Bölöni. Heuristic approaches for transmission scheduling in sensor networks with multiple mobile sinks. *Comput. J.*, 54(3):332–344, March 2011. (cited on pages 9 and 50)

93. TurtleNet. $https://dome.cs.umass.edu/turtlenet$. (cited on page 2)

94. Bang Wang. Coverage problems in sensor networks: A survey. *ACM Comput. Surv.*, 43(4):32:1–32:53, October 2011. (cited on page 18)

95. Cong Wang, Yuanyuan Yang, and Ji Li. Stochastic mobile energy replenishment and adaptive sensor activation for perpetual wireless rechargeable sensor networks. In *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, pages 974–979, April 2013. (cited on pages 10, 101, and 102)

96. Qing Wang, D.O. Wu, and Pingyi Fan. Delay-constrained optimal link scheduling in wireless sensor networks. *Vehicular Technology, IEEE Transactions on*, 59(9):4564–4577, Nov 2010. (cited on page 52)

97. You-Chiun Wang, Chun-Chi Hu, and Yu-Chee Tseng. Efficient placement and dispatch of sensors in a wireless sensor network. *Mobile Computing, IEEE Transactions on*, 7(2):262–274, Feb 2008. (cited on page 6)

98. Z.M. Wang, S. Basagni, E. Melachrinoudis, and C. Petrioli. Exploiting sink mobility for maximizing sensor networks lifetime. In *System Sciences, 2005. HICSS '05. Proceedings of the 38th Annual Hawaii International Conference on*, pages 287a–287a, Jan 2005. (cited on pages 48 and 79)

99. Lidong Wu, Hongwei Du, Weili Wu, Deying Li, Jing Lv, and Wonjun Lee. Approximations for minimum connected sensor cover. In *INFOCOM, 2013 Proceedings IEEE*, pages 1187–1194, April 2013. (cited on pages 7, 15, and 18)

100. L. Xie, Y. Shi, Y. T. Hou, and H. D. Sherali. Making sensor networks immortal: An energy-renewal approach with wireless power transfer. *Networking, IEEE/ACM Transactions on*, 20(6):1748–1761, Dec 2012. (cited on page 101)

101. Liguang Xie, Yi Shi, Y. Thomas Hou, Wenjing Lou, and Hanif D. Sherali. On traveling path and related problems for a mobile station in a rechargeable sensor network. In *Proceedings of the Fourteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '13, pages 109–118, New York, NY, USA, 2013. ACM. (cited on pages 10 and 101)

102. Liguang Xie, Yi Shi, Y.T. Hou, Wenjing Lou, H.D. Sherali, and S.F. Midkiff. On renewable sensor networks with wireless energy transfer: The multi-node case. In *Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2012 9th Annual IEEE Communications Society Conference on*, pages 10–18, June 2012. (cited on page 101)

103. Liguang Xie, Yi Shi, Y.T. Hou, Wenjing Lou, H.D. Sherali, and S.F. Midkiff. Bundling mobile base station and wireless energy transfer: Modeling and optimization. In *INFOCOM, 2013 Proceedings IEEE*, pages 1636–1644, April 2013. (cited on page 101)

104. Guoliang Xing, Tian Wang, Weijia Jia, and Minming Li. Rendezvous design algorithms for wireless sensor networks with a mobile base station. In *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '08, pages 231–240, New York, NY, USA, 2008. ACM. (cited on pages 80, 81, and 82)

105. W. Xu, W. Liang, and X. Lin. Approximation algorithms for min-max cycle cover problems. *Computers, IEEE Transactions on*, PP(99):1–1, 2013. (cited on page 50)

106. Xing Xu, Ji Luo, and Qian Zhang. Delay tolerant event collection in sensor networks with mobile sink. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9, March 2010. (cited on pages 48, 77, and 80)

107. Xu Xu and Weifa Liang. Monitoring quality optimization in wireless sensor networks with a mobile sink. In *Proceedings of the 14th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, MSWiM '11, pages 77–84, New York, NY, USA, 2011. ACM. (cited on pages 9 and 50)

108. Xu Xu, Weifa Liang, and T. Wark. Data quality maximization in sensor networks with a mobile sink. In *Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011 International Conference on*, pages 1–8, June 2011. (cited on page 48)

109. Zichuan Xu, Weifa Liang, and Yinlong Xu. Network lifetime maximization in delay-tolerant sensor networks with a mobile sink. In *Distributed Computing in Sensor Systems (DCOSS), 2012 IEEE 8th International Conference on*, pages 9–16, May 2012. (cited on pages 9, 48, 50, 77, and 80)

110. Changlin Yang and Kwan-Wu Chin. Novel algorithms for complete targets coverage in energy harvesting wireless sensor networks. *Communications Letters, IEEE*, 18(1):118–121, January 2014. (cited on pages 7, 15, and 19)

111. David K. Y. Yau, Nung Kwan Yip, Chris Y. T. Ma, Nageswara S. Rao, and Mallikarjun Shankar. Quality of monitoring of stochastic events by periodic & proportional-share scheduling of sensor coverage. In *Proceedings of the 2008 ACM CoNEXT Conference*, CoNEXT '08, pages 26:1–26:12, New York, NY, USA, 2008. ACM. (cited on page 24)

112. David K. Y. Yau, Nung Kwan Yip, Chris Y. T. Ma, Nageswara S. V. Rao, and Mallikarjun Shankar. Quality of monitoring of stochastic events by periodic and proportional-share scheduling of sensor coverage. *ACM Trans. Sen. Netw.*, 7(2):18:1–18:49, September 2010. (cited on page 24)

113. YoungSang Yun and Y. Xia. Maximizing the lifetime of wireless sensor networks with mobile sink in delay-tolerant applications. *Mobile Computing, IEEE Transactions on*, 9(9):1308–1318, Sept 2010. (cited on page 80)

114. Bo Zhang, Robert Simon, and Hakan Aydin. Maximum utility rate allocation for energy harvesting wireless sensor networks. In *Proceedings of the 14th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, MSWiM '11, pages 7–16, New York, NY, USA, 2011. ACM. (cited on page 50)

115. Pei Zhang, Christopher M. Sadler, Stephen A. Lyon, and Margaret Martonosi. Hardware design experiences in zebranet. In *Proceedings of the 2Nd International Conference on Embedded Networked Sensor Systems*, SenSys '04, pages 227–238, New York, NY, USA, 2004. ACM. (cited on page 2)

116. Sheng Zhang, Jie Wu, and Sanglu Lu. Collaborative mobile charging for sensor networks. In *Mobile Adhoc and Sensor Systems (MASS), 2012 IEEE 9th International Conference on*, pages 84–92, Oct 2012. (cited on pages 10 and 101)

117. Jerry Zhao and Ramesh Govindan. Understanding packet delivery performance in dense wireless sensor networks. In *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, SenSys '03, pages 1–13, New York, NY, USA, 2003. ACM. (cited on page 10)

118. Miao Zhao, Ji Li, and Yuanyuan Yang. Joint mobile energy replenishment and data gathering in wireless rechargeable sensor networks. In *Teletraffic Congress (ITC), 2011 23rd International*, pages 238–245, Sept 2011. (cited on pages 10, 101, and 102)

119. Miao Zhao and Yuanyuan Yang. A framework for mobile data gathering with load balanced clustering and mimo uploading. In *INFOCOM, 2011 Proceedings IEEE*, pages 2759–2767, April 2011. (cited on page 81)