Energy-efficient Mobile Device-assisted Schemes in Wireless Sensor Networks

by

Qiyue Wu

Thesis submitted in partial fulfillment of the requirements
For Master of Computer Science degree

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

Recently, wireless sensor networks (WSNs), consisted of battery-powered sensor nodes, are widely adopted by various civilian/military applications for implementing real-time monitoring or long-term surveillance tasks. One of the critical issues of WSNs is energy efficiency. Due to the limited battery capacity, the network lifetime and performance of WSNs are constrained. Also, once the sensor is deployed into a risky/remote environment, the replacement of its battery is hard. Therefore, how to improve the energy efficiency of the WSN is a critical issue and has gained tremendous attention from researchers around the world.

To address this problem, by taking advantage of the emerging high-mobility devices (e.g., unmanned aerial vehicle (UAV)), we propose energy-efficient mobile device-assisted schemes in different-scale WSNs. Thanks to the rapid development of wireless techniques, two emerging approaches, i.e., data gathering technique and wireless charging technique, are beneficial to balance the workloads among all sensors or replenish energy to achieve the semi-permanent WSN. First, we design data gathering schemes using the mobile data collector. In order to meet the performance requirements of systems with different scales, our algorithms have two working modes: single- and multiple-data-collector scenarios. For the small-scale system, a single data collector is adopted to access and collect data from the deployed node, and we propose single mobile data collector-assisted (SDCA) data collection schemes for small-scale WSNs. For the large-scale system, multiple data collectors are utilized to gather sensed data from deployed nodes, and two-mode multiple mobile data collector-assisted (MDCA) data collection scheme is designed for balancing between the system energy consumption and the data forwarding latency. Second, the joint data collection and energy charging scheme is developed by adopting mobile chargers (MCs) as mobile devices that are responsible for energy charging and data collection simultaneously. For facing the different performance requirements of systems, a two-mode MC scheduling algorithm is presented. To evaluate our works, extensive simulation experiments are conducted on the OMNeT++ simulator. The results demonstrate that the proposed algorithms achieve better performance than the control group regarding system-wide energy efficiency, network lifetime and average end-to-end delay.

Acknowledgements

First of all, I would like to express my great appreciation to my supervisor, Prof. Azzedine Boukerche who gave me the chance to pursue my graduate study and offered financial support for my research work. His valuable and constructive suggestions during the planning and development of my research work affected me and encouraged me to overcome many challenges. It will become a cherished experience in my life.

I would also like to thank Dr. Peng Sun who always gives me tremendous support. His valuable technical support on my research provided me with enormous assistance. I also appreciate his patience in guiding me in my experiments and thesis writing. His willingness to give his time so generously has been very much appreciated.

I also want to extend my thanks to all team members of PARADISE Laboratory. They encourage and support me during my graduate period. I would like to give special thanks to my friend, Zhijun Hou, who guided me and shared a lot of knowledge on OMNeT++ simulator with me. Additionally, thanks to other members that make the possibility to fulfill my thesis.

Finally, I wish to thank my parents for their encouragement and support throughout my study.

Publications

- Azzedine Boukerche, **Qiyue Wu** and Peng Sun, "A Novel Two-mode QoS-aware Mobile Charger Scheduling Method for Achieving Sustainable Wireless Sensor Networks,“ submitted to *IEEE Transactions on Sustainable Computing*, 2020.
- **Qiyue Wu**, Peng Sun, and Azzedine Boukerche, “SPSS: A Novel Joint Data Gathering and Wireless Charging for Sustainable Wireless Sensor Networks”, accepted by *Proceedings of 2020 IEEE International Conference on Communications (ICC)*, 2020.
- **Qiyue Wu**, Peng Sun, and Azzedine Boukerche, “A Novel Data Collector Path Optimization Method for Lifetime Prolonging in Wireless Sensor Networks”, in *Proceedings of 2019 IEEE Global Communications Conference (GLOBECOM)*, pages 1-6, 2019.
- **Qiyue Wu**, Peng Sun, and Azzedine Boukerche, “Unmanned aerial vehicle-assisted energy-efficient data collection scheme for sustainable wireless sensor networks,” in *Computer Networks*, 165:106927, 2019.
- Azzedine Boukerche, **Qiyue Wu** and Peng Sun, "Efficient Green Protocols for Sustainable Wireless Sensor Networks," in *IEEE Transactions on Sustainable Computing*, 5(1):61-80, 2019.
- **Qiyue Wu**, Peng Sun, and Azzedine Boukerche, “An energy-efficient uav-based data aggregation protocol in wireless sensor networks,” in *Proceedings of the 8th ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications (DIVANet)*, pages 34-40, 2018.
- Peng Sun, Azzedine Boukerche and **Qiyue Wu**, "Theoretical analysis of the target detection rules for the UAV-based wireless sensor networks," in *Proceedings of 2017 IEEE International Conference on Communications (ICC)*, pages 1-6, 2017.

Table of Contents

List of Tables

List of Figures

Nomenclature

AANT	Abnormal Ant
ACO-TCAT	Ant Colony Optimization with Three Classes of Ant Transitions
ASSORT	Asynchronous Sleep-wake Schedules and Opportunistic Routing Technology
AV	Automatic Vehicle
BANT	Backward Ant
CAERP	Cluster Arrangement Energy-efficient Routing Protocol
CBGA	Clustering-based Genetic Algorithm
CDS	Connected Dominating Set
CH	Cluster Head
ECUCF	Energy Conserved Unequal Clusters with Fuzzy logic
EH	Energy Harvesting
EHN	Energy Harvesting Node
ESA	Exponential Smooth Average
FoI	Field-of-Interest
GA	Genetic Algorithm
GCKNA	Geographic Distance-based Connected-k Neighborhood for All Paths Algorithm
GCKNF	Geographic Distance-based Connected-k Neighborhood for the First Path Algorithm

GPS	Global Position System
HTT	Hierarchical Two-Tier
LDS	Linear Distance-based Scheduling
LEACH	Low-energy Adaptive Clustering Hierarchy
LP	Linear Programming
MDCA	Multiple Data Collector-Assisted
MLSS	Multi-Level Sleep Scheduling
MRP	Multipath Routing Protocol
NZS-DCG	Non-Zero-Sum Duty-Cycle Game
OECS	Opportunistic Energy cost with sleep-wake schedule
PDF	Probability Density Function
PSO	Particle Swarm Optimization
QoS-PSO	QoS-based Particle Swarm Optimization Routing Algorithm
RF	Radio Frequency
RkM	Reduced k-Means
RN	Rendezvous Node
RSN	Regular battery-powered Sensor Node
SANT	Search Ant
SCTA	Spanning Covering Tree Algorithm
SCX	Sequential Constructive Crossover
SDCA	Single Data Collector-Assisted
STEM	Sparse Topology and Energy Management
SW-PFR	Sleep-Wake Probabilistic Forwarding Routing
SWIPT	Simultaneous Wireless Information and Power Transfer

TABU-RCC	Tabu-search-based algorithm under Routing and Coverage Constraints
TCOR	Transmission power Control-based Opportunistic Routing
TDMA	Time Division Multiple Access
UAV	Unmanned Aerial Vehicle
WSN	Wireless Sensor Network

Chapter 1

Introduction

Wireless sensor networks (WSNs) consist of a set of battery-powered sensor nodes and sinks that are deployed in the Field-of-Interest (FoI) to monitor and detect special events that are of interest to users [7, 8, 9]. Fig. 1.1 shows the general architecture of the WSN. Many advantages of sensor nodes (e.g., low cost, small size, capable of both sensing and communication, etc.) make the WSN easy to be deployed for various civil and military applications, such as target tracking, environmental surveillance and healthcare monitoring. However, due to the limited battery capacity, the lifetime and performance of the deployed WSN will be limited/constrained, which may lead to the deployed system unable to meet the performance requirements. Therefore, how to improve energy efficiency is a crucial issue and is considered as our motivation in this thesis. In the following, we will introduce motivations and contributions of our work in detail.

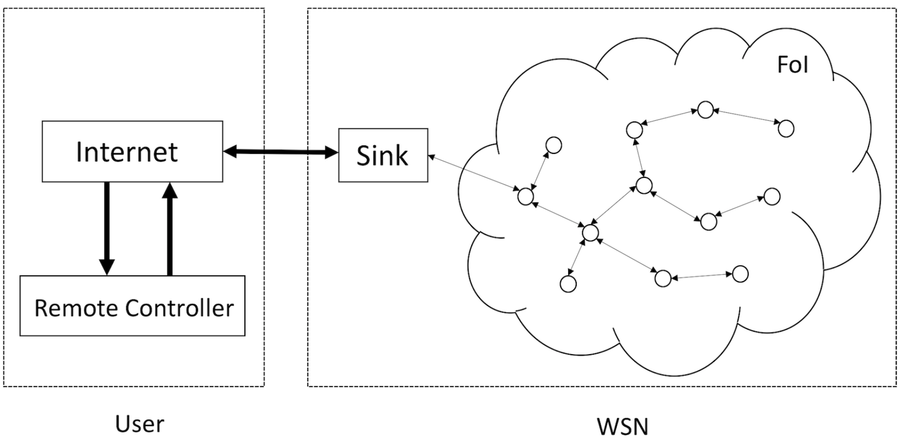


Figure 1.1: The general architecture in the WSN

1.1 Motivation and Objective

WSNs aim to provide real-time monitoring or long-term surveillance of the given FoI, and then forward the gathered information to the designated data sink [10]. Therefore, the desired coverage degree and reliable communication are fundaments to achieve requirements of WSNs [11, 12, 13]. However, the limited available energy budget for sensors directly affects the lifetime and performance of the system [14, 15]. Thus, how to enhance energy efficiency is still a topic that can be further discussed.

The energy of sensors in WSNs is mainly consumed in two ways, i.e., data transceiving and sensing. Moreover, the data transceiving consumes more energy. The nodes which are close to the data sink are prone to deplete their energy because all messages from sensor nodes to the sink have to go through these nodes, which causes a huge amount of energy consumption. Once these nodes exhaust their energy, they cannot communicate with others, which possibly causes that some sensors are isolated and cannot transmit the data to the data sink. It is commonly called energy-hole problem [16]. Many methods are proposed to solve this problem. Conventionally, the deployed sensor nodes in the WSN are static or only have limited mobility (i.e., slow-moving speed and limited/short moving range, etc.). Accordingly, the structure of the deployed WSN is relatively stable, and can only be adjusted in a small scale or within a limited time. Hence, the existing energy-efficient approaches designed for the conventional static WSN concentrate on several aspects, such as node deployment schemes [17, 18, 19], node scheduling schemes [20, 21, 22], energy-efficient routing algorithms [23, 24, 25, 26, 27], etc. However, the limited capacity of the built-in battery is a restraint of the system lifetime of the battery-powered WSN with conventional methods. Fortunately, with the scientific and technological development, many new emerging technologies are applied to WSNs to enhance the energy efficiency of WSNs, e.g., high-mobility data collection devices [28, 29], energy harvesting techniques [30, 31], and cognitive radio [32, 33], etc.

In this thesis, we focus on energy-efficient schemes with new emerging technologies in the WSN. Compared with the traditional mobile sensor nodes [34], the new high-mobility devices (e.g., unmanned aerial vehicle (UAV)) have a much faster-moving speed, a wider deployment range and a relatively long operating time. By installing the data transceiver and the wireless charger, the mobile devices can communicate with sensors and replenish the energy to sensors. Accordingly, the devices can move to specified/designated locations within a restricted period to collect data and replenish the energy for sensors, by which, workloads between the deployed sensors can be balanced, and the energy of the sensors can also be harvested. Moreover, by taking advantage of the long moving range of the mobile device, the sensor nodes deployed in areas further away from the data sink (or isolated nodes) can also be visited by the device. Thereby, the connectivity of the deployed system can be ensured.

Currently, the high-mobility devices for the data collection are primarily classified into the mobile sink and the mobile data collector. The mobile sink is typically considered as a mobile access point providing access to the internet for the deployed sensors and needs to connect the backbone all the time. Accordingly, the sink has to carry extra devices (large payload) to support its functionality. It is normally carried by a vehicle to change its location [35, 36]. Many energy-efficient path planning algorithms (e.g., [37, 38, 39, 40]) have been proposed for the application using the mobile sink in WSNs. However, the mobile sink needs to connect with the backbone all the time. Once the mobile sink changes its location, the topology of the network should be reconstructed. The corresponding cost for network reconstruction (e.g., time cost) could be considerable. Thus, the location of the mobile sink cannot be changed frequently. Moreover, as the carrier of the mobile sink, the mobility of the ground vehicle would be severely affected by the obstacles on the ground. On the contrary, the UAV as a data collector flies in the air and can overcome the adverse effect of the complex geographic surface. To sum up, the mobility of the mobile sink (i.e., the ground vehicle) is relatively lower than the mobile data collector (i.e., the UAV), which may introduce considerable latency [41]. Therefore, adopting a high-mobility data collector is a more practical option for data gathering across the FoI while allowing system-wide data transmission latency to be kept at a relatively low level in WSNs.

Except for the data gathering technique with mobile devices, the energy recharging technique is also considered as a viable method to elongate the lifetime of WSNs [42, 43]. A lot of novel hardware is developed to apply and support the energy recharging technique in the WSN (e.g., wireless identification, sensing platform). There are two main methods regarding the energy recharging technique, i.e., energy harvesting from the ambient environment and wireless charging. A large number of environmental energy harvesting algorithms have been developed in [44, 45, 46, 47, 48]. In these algorithms, sensors can harvest energy from ambient sources, such as wind, thermal, solar, mechanical, temperature variations [49]. However, due to the unstable nature of the ambient energy source, relying solely on energy harvesting does not provide stable energy supply for the system. Therefore, the mobile charger-assisted wireless charging technology is regarded as a more effective energy charging method for the WSN. The mobile charger (MC) as an emerging device can supply the energy to the sensors stably. The sensors can receive energy from MCs to make up the energy consumed on data transmission, which keeps them alive permanently when the number of MCs is adequate. Although many works of wireless charging have been studied, the energy capacity of the MC is ignored in their works. They assumed that the MC has infinite energy, which is not practical in real applications. To enable the approach more reasonable, the energy constraint of the MC will be considered in our proposed scheduling designs.

Typically, a carefully planned moving trajectory is essential for applying the mobile device. The straightforward method is that the mobile device visits every sensor to gather the data from it directly and replenish the energy for it. Whereas, this method is only suitable for sparse networks or networks with a small amount of deployed sensors. It is not effective for large-scale dense WSNs given that the mobile device needs to traverse a plethora of sensors, which leads to a long path for the device, and in turn, introduces a considerable data latency. In practical experience, some applications are delay-sensitive and their data update period should be minimized. Accordingly, to satisfy the system delay constraint, the route of the mobile device should be carefully planned.

Based on the motivation, it is feasible to propose energy-efficient algorithms with mobile devices in cluster-based WSNs. In the data gathering methods, we adopt the mobile data collector as the high-mobility decive. On the other hand, the MC is regarded as the high-mobility device in the joint data collection and energy charging scheme. The components of these proposed algorithms are mainly the topology construction and the path planning of the mobile device. The objectives of our proposed schemes are,

1. Decreasing the system-wide energy consumption;
2. Reducing the delivery latency from sensors to the data sink;
3. Designing the algorithms that can be adopted by various scale WSNs.

1.2 Contribution

The main contributions of this thesis are listed as follows:

1. For the small-scale system, we design a single mobile data collector-assisted (SDCA) data collection scheme. This work aims to solve a joint cluster head (CH) selection and the routing problem of the single data collector. To ensure that the data can be forwarded to the sink within a tolerant delay, we apply a genetic algorithm (GA) to solve the joint problem and obtain the optimal route generated by a GA-based optimization method.
2. In order to improve the energy efficiency and system performance based on the SDCA data collection scheme in WSNs, we propose the improved SDCA data collection

scheme. In detail, the deployed sensors are divided into clusters based on our newly proposed clustering algorithm, and then the optimal path for the data collector is derived by the improved GA optimization method.

3. To improve the energy efficiency for large-scale WSNs, we develop a two-mode multiple mobile data collector-assisted (MDCA) data collection scheme in cluster-based WSNs: for the delay-sensitive application, the proposed scheme runs in gathering-and-carrying mode. Each collector gets back to the sink so that the urgent data can be transmitted to the sink in time; on the other hand, for the delay-tolerant application, the scheme runs in data-relaying mode. The data gathered by a collector will be forwarded to the data sink by intermediate collectors in order to minimize the energy consumed by the collector traveling back to the data sink.

4. To make the WSN semi-permanent, the joint data collection and energy charging scheme is designed by adopting MCs in WSNs. First, the clustering algorithm is improved based on the α-hop clustering algorithm by considering the capacity of sensors. Second, we introduce MCs scheduling schemes by considering two distinctive scenarios, i.e., the delay-tolerant system and the delay-aware system. For the delay-tolerant system, we design a single-path scheduling scheme (SPSS). In this scheme, the genetic algorithm (GA) is adopted to derive a path to achieve the objectives in the delay-tolerant charging problem. Two MCs traverse the derived path in opposite directions to collect data and replenish energy for each sensor with full charge in each cluster. In the delay-aware system, a multiple-path scheduling scheme (MPSS) is designed to schedule multiple MCs to charge sensors. The data collection task will be split and executed by multiple MCs through cooperation. Additionally, the sensors are charged partially to decrease the delay. We combine the minimum spanning tree algorithm and the 2-approximation algorithm [50] to derive a set of paths for MCs.

To verify the effectiveness of the proposed schemes, we also carry out a set of experiments by comparing the proposed works with other existing schemes using mobile devices.

1.3 Outlines

The rest of this thesis is organized as follows. Chapter 2 presents and discusses the existing energy-efficient schemes. The SDCA data collection scheme is proposed in Chapter 3. Chapter 4 describes the details of the improved SDCA data collection scheme. In Chapter 5, we extend the data gathering schemes in WSNs with multiple mobile data collectors.

Chapter 6 develops the two-mode scheduling schemes for the MCs in the WSN. In the end, we conclude this thesis and provide the direction for future work in Chapter 7.

Chapter 2

Literature Review

In this chapter, we summarize and classify the state-of-the-art energy-efficient-related approaches for designing a sustainable WSN. Generally, conventional approaches mainly focus on static WSNs, in which all deployed sensors are stationary and unable to change their locations after being deployed in the FoI. Accordingly, the diversity of the topology in static WSNs is limited, and existing approaches for energy-efficient static WSNs can be roughly categorized into five classes: 1) Clustering-based algorithms; 2) Node deployment strategies; 3) Node scheduling schemes; 4) Data routing algorithms; and 5) Joint energy-efficient designs.

Recently, with scientific and technological development, many new emerging technologies are applied to WSNs. Here, we mainly focus on four types of new techniques-based energy-efficient approaches: 1) High-mobility data collection devices assisted WSNs; 2) Energy harvesting (EH)-aided sustainable WSNs; 3) Machine learning-based approaches; and 4) Cognitive WSNs. We provide a taxonomy of the existing approaches designed to address the problem of energy efficiency in WSNs in Fig. 2.1. The following sections will present conventional methods, emerging techniques-assisted methods, respectively.

2.1 The architecture of WSNs

First of all, we introduce a basic concept in the WSN, i.e., the architecture of WSNs. Typically, to implement a WSN system, a large number of sensors need to be deployed in the FoI. Therefore, the architecture of the system should be considered in every approach based on its performance requirements. There are two main types of architectures that are widely adopted: *flat-based WSNs* and *hierarchical-based WSNs* [51, 52]. We will introduce the details of these two types of architecture in this section.

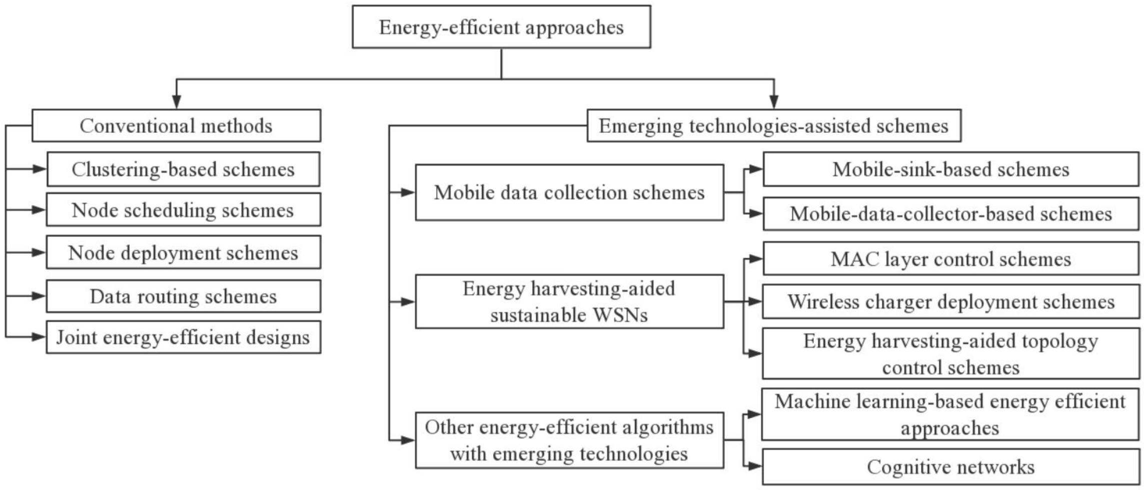
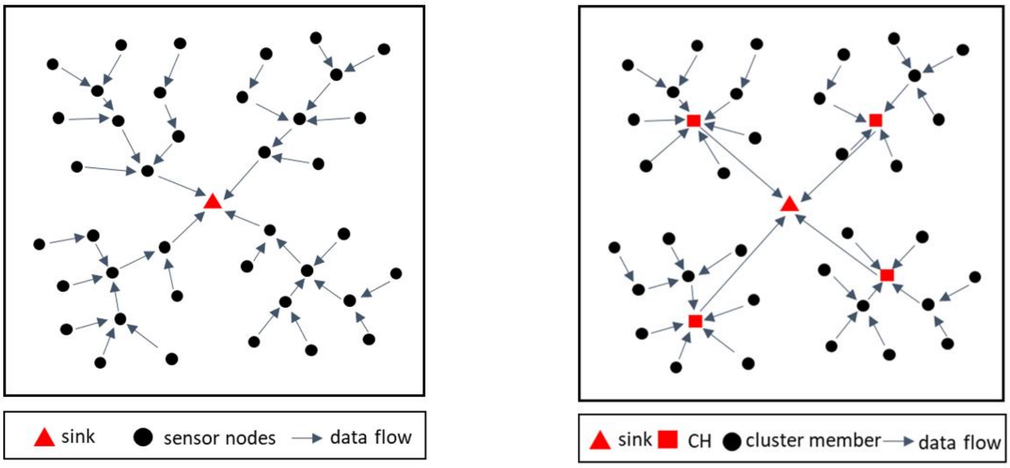


Figure 2.1: Classification of existing approaches proposed for addressing energy efficiency problem in WSNs

Flat-based architecture: For this type of architecture, all nodes operate with the same algorithm, i.e., there is only one tier in the system. A common flat-based architecture is a spanning tree, which is illustrated in Fig. 2.2(a). Each node transmits its sensed data to the data sink by direct communication or by relying on its parent nodes. Since the parent node needs to help its children transmit data while lacking the ability to control the amount of data transmitted by the child nodes, the parent node will consume much more energy than its child nodes. Eventually, the energy-hole problem will occur. Therefore, this type of architecture is unsuitable for the energy-efficient design of WSNs, due to the lack of control over the sensor nodes of the system.

Hierarchical-based architecture: Contrary to the flat-based architecture, hierarchical-based WSNs use multiple-tier architecture [53, 54]. The elements in different tiers have diverse functionalities. Cluster-based architecture is a typical hierarchical-based architecture, which is commonly used in WSNs. An example of a cluster-based WSN deployed in the 2D platform is shown in Fig. 2.2(b). The sensor nodes in the FoI are divided into several groups called clusters. Each cluster has a cluster head (CH) which is responsible for collecting the data from its cluster members and controlling its cluster members, such as changing the working states of its members. Accordingly, to reduce the energy consumption rate of the cluster member, the CH can switch the node into sleep or idle states to conserve its energy. Additionally, sensors rotate the role of the CH in case the CHs run out their energy fast, by which, the probability of producing the energy-hole problem is reduced. Therefore, hierarchical-based architecture is more suitable for designing an energy-efficient WSN.



(a) Flat-based architecture

(b) Cluster-based architecture

Figure 2.2: Two kinds of network architecture

2.2 Conventional Methods

In this section, we will discuss some existing energy-efficient approaches proposed for the "conventional WSN". The term 'conventional WSN' is defined as follows: the components of the deployed WSN are static/stationary or have only limited mobility (i.e., slow-moving speed, limited/short moving range, etc.). Accordingly, the structure/topology of the deployed conventional WSN is relatively stable, and can only be adjusted in small-scale or limited time. For example, in some Virtual-force-based approaches (e.g., [55, 56, 57, 58]), the adopted mobile sensors can move only to improve the coverage degree of a randomly deployed network with initial deployment. Therefore, due to the constraint of the limited mobility, the existing approaches in conventional WSNs mainly focus on the following five aspects: 1) Node deployment strategies; 2) Clustering-based schemes; 3) Node scheduling algorithms; 4) Data routing protocols; 5) Joint energy-efficient designs. In the following parts of this section, we will discuss each type of approaches, based on the state-of-the-art, in detail.

2.2.1 Node Deployment Schemes

To achieve the expected system coverage degree and network connectivity, the intensive node deployment strategy is always adopted to deploy the WSN. Accordingly, the redundant sensor nodes will exist in the system, which may increase the overall deployment cost of the WSN, and adversely affect the performance of the deployed WSN. Due to the limited wireless channel/frequency resource, the intra-system/inter-node interference problem will be aggravated as the number of deployed nodes increases. Consequently, the system-wide link condition may be degraded, and the sensor nodes will have to spend more energy on data retransmission to overcome the potential transmission failures, which in turn increases the energy consumption rate of the system and reduces the lifetime of the deployed system. Therefore, the node deployment scheme, which can reduce the network redundancy while keeping system coverage degree and connectivity, has attracted much attention from researchers. Currently, many approaches have been proposed for addressing the node deployment problem. Here, we will discuss a number of node deployment approaches in detail.

Node deployment strategies need to improve the system's energy efficiency (or reduce the system's energy consumption) as much as possible while ensuring the stable performance of the system. In [59], a novel sensor deployment scheme (ACO-TCAT) was introduced to prolong the network lifetime and ensure the connectivity of the whole network. To achieve these objectives, the authors derived the preferred positions for each node based on the ant colony optimization method with three kinds of ant transitions. The first is called ant transition of Class I (ATC-I). It moves step by step, according to the pheromone intensity of the path and the heuristic value. In addition, the ants move from the sink in the beginning. The second kind of ant transition, named ant transition of Class II (ATC-II), is responsible for effective candidate location selection. The third one is the ant transition of Class III (ATC-III). If both ATC-I and ATC-II are successful, ATC-III will be applied to choose the points from candidates. Nodes with remaining energy will become candidates. To implement the proposed algorithm, the authors assumed that the network was divided into a set of grids with identical size, and the candidate location of the sensor was the vertex of the grid. To guarantee the network connectivity, the candidates of the next step are selected within the transmission range of the sensor in the current step. The algorithm is implemented as follows. Initially, the system sets $\Psi = (\varphi_1, \varphi_2, \varphi_3)$ to represent a sequence of three types of ant transitions. The class of ant transition Ψ at location t is decided by the candidate locations. The pheromone intensity of edge (i, j) after a tour of the ant is updated as follows,

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}(t), \tag{2.1}$$

where ρ is pheromone evaporation parameter and $\Delta\tau_{ij}(t)$ is a parameter related to the number of total points of the solution. In order to narrow the searching area of the algorithm, two boundary parameters, i.e., τ_{min} and τ_{max}, should be predetermined, and T_c is the predefined number of iterations. Moreover, the value of ρ needs to be adjusted in the process of the algorithm. In the first half period, the value should be small so as to create the tiny difference in the pheromone intensity of all edges. On the other hand, in the last half period, the value needs to be large enough to generate a large difference. The advantage of using multiple types of ant transitions is that it decreases the searching range and inferior solutions, which in turn improves the operational speed of the algorithm. Moreover, the authors also considered the effects of the obstacles in this work. Similarly, in [60], an ant colony optimization-based sensor deployment protocol was proposed to prolong the network lifetime and achieve full coverage. The authors used the ant colony optimization to calculate the locations for each sensor and designed a sweep-based protocol to move the sensors to the assigned positions.

Another node deployment scenario was developed by Hao et al. [61]. The motivation of this algorithm was to improve the system's energy efficiency and the system-wide communication link quality based on the optimal locations of sensors derived by considering the effects of the noise interference. The authors presented a link weight model which was formulated on the basis of the current link quality and residual energy level of the node. In this work, each node first exchanges its state information (i.e., its id and initial energy) with its neighbors. By receiving the state information from its neighbors, the node calculated the link weight and sorted its neighbors in the descending order according to their weights. The neighbor with a lower weight is less than the predetermined constant would be removed from the list of neighbors. After updating the neighbors list, it multicasts "link construction" messages to its alive neighbors. If a node receives this link construction message from its neighbor and the neighbor is still on its neighbors' list, it will send back a "ACK" message, and the link between these two nodes will be established. The advantage of the algorithm is to achieve high link quality, low interference and low energy consumption simultaneously.

The authors in [62] designed an energy-efficient connected dominating set (CDS) to improve both the system energy efficiency and achievable system connectivity/reliability. To simultaneously maximize the energy efficiency and reliability, the authors designed a multi-objective optimization model by conducting a probabilistic network model to transform the reliable parameter into a probabilistic parameter to capture the uncertainty of connections between sensors. This is the most significant difference between this work and other existing approaches. The authors showed that system energy efficiency was related to the size of connected dominating sets. The simulation results demonstrated that the presented algorithm could improve the system performance more effectively than other heuristic-based algorithms used as a control group in terms of stable period, link condition and energy consumption rate.

As previously mentioned, node deployment strategies aim to reduce the redundant nodes based on the system performance requirements, which commonly improve energy efficiency, enhance coverage and connectivity, reduce the system deployment cost, etc. We summarize some existing node deployment strategies in Table 2.1. Node deployment is the initial phase for deploying a WSN system, and it is often combined with other methods into a joint energy-efficient algorithm. Moreover, the deployment of sensors in some schemes is deterministic, which means these schemes may not be adopted in all scenarios due to the lack of flexibility. Therefore, other energy-efficient methods are proposed, which are introduced in the following section.

Table 2.1: Node Deployment Strategies

Protocol	Performance Requirements	Type of sensors	Techniques	Strengths/Limitations
Liu [59]	Energy efficiency & Connectivity	Homogenous	Ant Colony Optimization	Strengths: high computation speed & taking obstacle into consideration
Yourimet et al. [63]	Energy efficiency & Coverage	Heterogenous	Genetic algorithm	Strengths: low cost & full coverage as with as few nodes as possible
Hao et al. [61]	Energy efficiency & Link quality	Homogenous	Model Formulation	Strengths: long stable period & reliable communication
Tiegang et al. [64]	Energy efficiency & Coverage & Connectivity	Homogenous	Model Formulation	Strength: achieve multiple requirements Limitation: not suitable for all kinds of WSNs
Khalil et al. [62]	Energy efficiency & Connectivity	Heterogenous	Multi-objective Optimization Algorithm	Strength: full coverage with as few nodes as possible
Halder et al. [65]	Energy efficiency	Heterogenous	Model Formulation	Strength: high energy conservation
Liao et al. [60]	Energy efficiency & Coverage	Homogenous	Ant Colony Optimization	Strength: high energy conservation Limitation: high overhead

2.2.2 Clustering-based Schemes

Recall that; the cluster-based architecture of the system is helpful to balance the workloads among all sensors, and can effectively solve the energy-hole problem. Therefore, the clustering algorithm is an essential part of many energy-efficient schemes. For the clustering algorithms, determining how to form the clusters and select the CH of each cluster are two key problems. Many algorithms have been proposed for addressing the clustering problem. One of the most classic clustering methods is Low-Energy Adaptive Clustering Hierarchy (LEACH) [66]. Compared with the traditional clustering algorithm, the CHs are equally rotated to distribute the energy load among all the sensor nodes in WSNs in LEACH. In the setup phase, the CHs are elected with a certain probability. Each sensor has a random probability. If its probability is over a predefined threshold, the sensor is selected to be a CH and broadcasts the message to others. The non-CH chooses one of the CHs which is closed to itself and sends the request message for joining the cluster to its CH. In the steady phase, the cluster members send the sensed data to CHs. The CHs compress the received data and forward the data to the sink. After a predetermined time period, all sensors enter the setup phase again to select a new set of CHs. Although LEACH reduces energy consumption and evenly distributes the workload among all sensors, it still has some drawbacks. The system should consume more overhead while reforming the clusters periodically. Additionally, due to the probability-based CH election, the selected CH might have the low residual energy level and would die quickly. Another drawback is that all CHs need to directly communicate with the sink, in which case the energy expenditure of the CH would be huge because of the long distance between the CH and the sink.

Many protocols based on LEACH have been proposed to improve energy efficiency, e.g., [67, 68, 69, 70, 71, 72, 73, 74]. The improved LEACH-based approaches can be roughly classified into two types. The first type of approaches focus on improving the CH selection method in LEACH. For example, LEACH-centralized (LEACH-C) [67] is a LEACH-based centralized clustering algorithm. In this work, the sink calculates the average residual energy level of sensors according to the energy levels of all sensors. The sensors with energy levels remaining higher than the derived average energy level will become CH candidates. Then, the sink chooses the CHs based on the simulated annealing algorithm. Similarly, in Energy-LEACH protocol [73], the CHs are also selected based on the current residual energy. The nodes with more residual energy become the CHs and inform others, and the nodes with less residual energy turn into cluster members and send a message for joining the cluster to its new CH. To improve LEACH, an adapted approach (LEACH-A) is presented in [74]. In this protocol, the residual energy threshold is predetermined. If its residual energy is over the threshold, the node is elected to be the CH and to communicate to the sink. However, if the residual energies of all sensors are less than the threshold, the node nearest the sink is selected to be a CH. In LEACH, if the CH dies, the cluster cannot communicate with the sink until the system enters the next round. To avoid this issue, in V-LEACH [68], a CH and a vice-CH are elected in each cluster. The vice-CH takes the role of the CH once the CH dies. It can decrease the frequency of CH selection in order to reduce the construction overhead. Another LEACH-based scheme is LEACH-FL [70]. The threshold is calculated by Fuzzy Logic, which is related to battery level, distance and node density. Each node has a random value between 0 to 1. The node with a value less than that of the threshold will be selected to be a CH. The second type improves LEACH regarding the path from CHs to the sink. For instance, TL-LEACH is proposed in [69]. A two-level hierarchy is built for reducing energy consumption, in which, several CHs that can communicate directly with the sink are selected to be heads of CHs. Accordingly, the cluster members in each cluster communicate with their CH, and the CH sends the received data to the designated head of CHs. The head of CHs then forwards all gathered data to the sink.

In addition to the LEACH-based approaches mentioned above, a number of other clustering algorithms have also been proposed. Chamam et al. [75] proposed a clustering algorithm regarding the CH election and cluster formation. The CHs are selected based on three factors, i.e., their residual energies, their distances to cluster members, their positions within the diagram formed by CHs. Meanwhile, there are some constraints regarding routing (connectivity) and coverage within each cluster. The area covered by each cluster needs to exceed a predefined coverage degree, the size of each cluster should be smaller than a predetermined upper bound, and the connectivity between CHs and nodes should be guaranteed. The authors [75] formulated this problem as an Integer Linear Programming (LP) model, which is NP-Complete. Then, they designed a Tabu-search-based algorithm for CHs election under routing and coverage constraints (TABU-RCC) to derive the near-optimal solution. However, in this work, each non-CH node has to connect to at least one CH. This results in too many CHs in the network, especially in large-scale WSNs. Therefore, energy efficiency may not be achieved effectively. A novel algorithm named Energy Conserved Unequal Clusters with Fuzzy Logic (ECUCF) was designed and presented in [76]. The deployed network is first divided into closest, middle and outside sectors based on the distance to the sink and the residual energy of sensors. Prime-CHs are chosen based on the nodes' probability function, and final CHs are selected from the prime-CHs using type-1 Fuzzy Logic with respect to the distance to the sink, the node's residual energy, and the node proximity of the sensor. To improve the performance of the algorithm with type-1 Fuzzy Logic, authors in [77] presented an energy-efficient clustering algorithm for WSNs by type-2 Fuzzy Logic. In this work, the fuzzy input variables are similar to the parameters in [76]. The results showed that the performance of type-2 Fuzzy Logic is always better than type-1 Fuzzy Logic because type-2 Fuzzy Logic can handle the uncertainty in environments with more accuracy than type-1 Fuzzy Logic.

Since the CHs will consume significantly more energy than those co-existing regular sensors, the selection of the set of CHs for improving energy efficiency is another critical problem of the clustering algorithm. A set of approaches have been designed to address the CH selection problem. Boukerche et al. [2] proposed a CH selection for mobile WSNs. The objective was to maximize or minimize a function $F(x)$ which could be a mathematical calculation that could be implemented by a single sensor or a formula capable of capturing characteristics of sensors, e.g., residual energy level, the average distance to its neighbors, etc. The proposed algorithm is implemented in multiple phases. In the beginning, all sensors begin the election process. The nodes are classified into two states through the election process: follower and candidate. If a node's state is a follower, then it has five types of events that can happen: 1) The node receives a "*Tree*" message from a node which is the parent of it. 2) The node receives the "*I_Am_Here*" message. If the node sending this message has a better-evaluated leader, then the merging process is launched. 3) The node receives a "*Merge*" message, after which its leader will choose which cluster to merge.

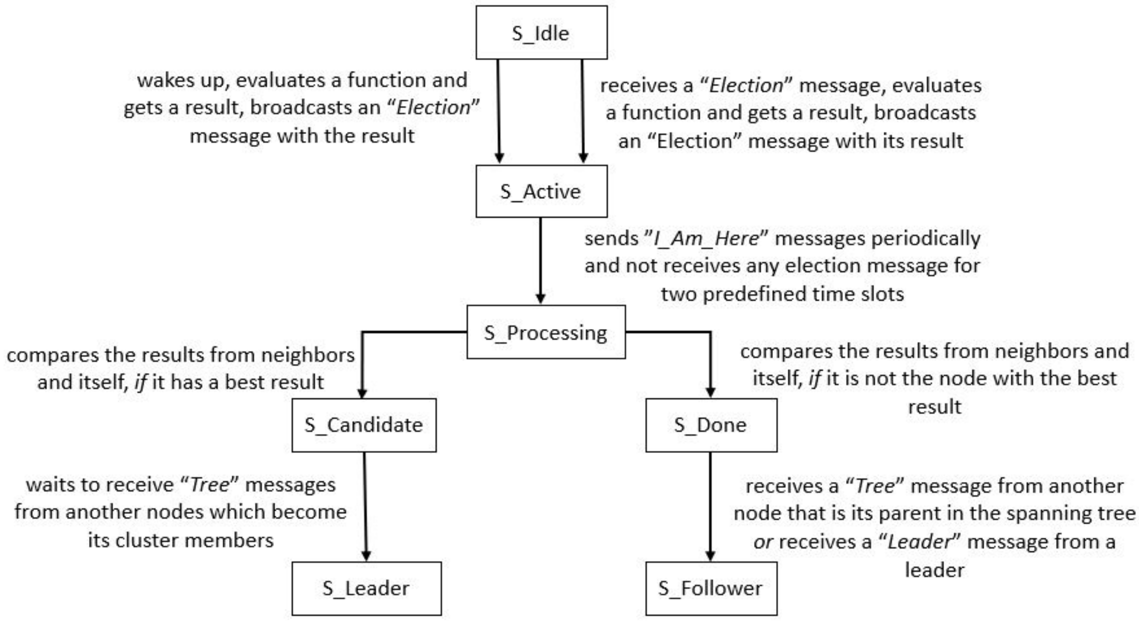


Figure 2.3: The description of the states of nodes in [2]

4) The node receives a "*Leader*" message. It realizes that the leader is changed and the spanning tree is reconstructed. 5) The node receives a "*Tree2*" message. It knows that two clusters are merged, and the directions of the links of the spanning tree are flipped. Finally, the leaders are elected from candidates, and the clusters in the network are formed. After the CH is selected, it informs its cluster members. If it receives a "*Merge*" message, it will determine the cluster merging. The detailed description of the states is illustrated in Fig. 2.3. Similar work can be found in [78]. Han et al. [79] proposed a double-phase CH election scheme to achieve the energy efficiency in heterogeneous WSNs. In the first phase, the temporary CHs are selected based on the relative levels of initial energy and their residual energy. After selecting temporary CHs, the system enters the second phase. The temporary CH with the highest residual energy level in its cluster becomes the final CH. Otherwise, the temporary CH is replaced by its cluster-mates that have higher residual energy levels in their clusters. The advantage of this protocol is the prevention of the nodes with low-level energy becoming CHs, by which the lifetime of WSNs can be prolonged.

2.2.3 Node Scheduling Schemes

In Section 2.2.1, we discussed node deployment schemes, by which sensor nodes can be deployed in a deterministic manner to fulfill the system performance requirements. However, in some specific application areas (e.g., monitoring of volcanic activities, boundary/military-related surveillance, etc.), the deterministic deployment strategies may be not applicable due to environments that are hazardous to human safety. The randomly intensive deployment methods are still necessary. In this case, the number of deployed nodes is definitely larger than the optimal number of nodes needed to achieve the expected system coverage degree and network connectivity. Therefore, overlaps between the coverage areas and communication areas of different nodes are inevitable. The node scheduling scheme, which can periodically switch the state of nodes ON/OFF in turn to reserve the limited energy of each node, is considered a potential solution for coping with this problem. Briefly, the basic logic of the node scheduling scheme is that the system only keeps a minimum number of carefully selected active nodes to satisfy its performance requirements, and switches the rest of the nodes into sleep/idle state to save their limited energy. Once the system performance requirements change or the residual energy levels of some active nodes reduces, the nodes in the sleep/idle state will be triggered. Therefore, the crucial problem of node scheduling lies in creating a balance between energy consumption and network performance. Additionally, in certain cases, the node scheduling scheme needs to face a special delay issue, i.e., wakeup delay [80].

We have introduced two kinds of system architectures in WSNs before, i.e., flat-based and hierarchical-based. Here, we classify the node scheduling schemes based on these two types of system architectures and provide some examples designed for each type of architecture, respectively.

For flat-based WSNs, the most distant node needs to transmit the data to the sink through intermediate nodes. If it is awake and needs to send its sensing data, all the nodes on the route between this further node and the sink should be awake to ensure that the data can be forwarded to the sink. Therefore, to improve energy efficiency, the schemes in flat-based WSNs primarily focus on how to reduce the number of awake nodes and satisfy performance requirements at the same time. Now, we introduce some schemes in flat-based WSNs.

Schurgers et al. [81] proposed a protocol named Sparse Topology and Energy Management (STEM). The authors adopted two types of channels: one is the data channel for data transmission, another is the wakeup channel for waking nodes. When a sensor node needs to transmit data to the sink, it periodically sends a beacon packet to wake its neighbors through the wakeup channel. Once the neighbor receives the beacon packet, it replies with an ACK message and switches its state to awake. The beacon messages are sent periodically until the node receives an ACK packet, or until the waiting time reaches the predetermined threshold.

By considering the homogeneous WSN environment, another node scheduling scheme was presented in [3]. They combined sleep/awake state switching and probabilistic forwarding protocol, and then designed a protocol called the Sleep-Wake Probabilistic Forwarding protocol (SW-PFR). The proposed work has two phases. The initial phase is the front creation phase. The objective of this phase is to ensure a period of reasonable length for implementing the data propagation process. They adopted the "flooding" mechanism with a predetermined number of iterations to create the front. The header of each message has a counter β which contains the hop number needed to pass in the initial phase. Once a sensor receives a message, it reduces the β by 1 and deterministically forwards the message to the sink. The sensor multicasts the packets to its neighbors that lie in the direction of the sink. The second phase is the probabilistic forwarding phase. Each sensor calculates the forwarding probability $\mathbf{P}_{fwd} = \frac{\phi}{\pi}$, where ϕ is the angle defined by two lines: the line connecting this sensor and another sensor which detected the event and another line connecting this sensor and the sink. The sensor with a larger probability value should be chosen because it has a bigger angle, which means that it is located closer to the direct line between the source node and the sink (see Fig. 2.4). The strengths of this protocol include the implementation of a sleep/wakeup mechanism and the targeting of WSNs with multiple events.

Another node scheduling protocol named non-zero-sum duty-cycle game (NZS-DCG) was proposed in [82]. In this protocol, the authors considered the duty cycle assignment and the cooperation of nodes in the overlapping area. They formulated the problem based on game theory to obtain the initial duty cycle value for each sensor. The results showed that the protocol not only reduced energy consumption but also balanced workloads of sensors.

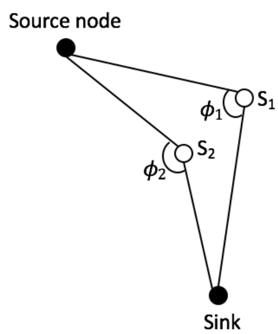


Figure 2.4: Angle and closeness to optimal line [3]

On the other hand, node scheduling approaches in the hierarchical-based WSN mainly concentrate on forming clusters and selecting the set of sleep nodes, in which the system allows the CHs to control the states of its cluster members. By taking advantage of the multiple-tiered structure of the system, the controlling workload can be distributed to CHs. It is helpful for reducing the system computational complexity [83]. In [83], Deng et al. developed a cluster-based node scheduling scheme called Linear Distance-based Scheduling (LDS). The objectives of this approach were to prolong the system lifetime and ensure appropriate sensing coverage degree. In this work, authors assumed that two kinds of sensors were deployed in the WSNs, i.e., predetermined CHs and regular sensors (i.e., cluster members). They demonstrated how the CH chose its cluster members that needed to sleep based on the following criterion: coverage capacity will not be degraded by switching these nodes into the sleep state. By assuming that the distance between the node and its CH was known in advance, the authors showed that the nodes far away from their CHs had a greater chance to be put into sleep, and the presented work achieved high energy conservation.

Lin et al. [84] proposed an optimal node scheduling based on a CH deployment scenario. The proposed work considered three system performance requirements: the coverage requirement of sensors, the routing requirement of sinks (i.e., ensuring each CH can transmit data to the sink in the formed connected network), and the connectivity between each CH and its cluster members. This approach has two steps. In the first step, each sensor calculates the maximum number of disjoint full-coverage sets. Then, in the second step, a CH is deployed into each of the sets to build the set that can meet the coverage requirements as long as possible. The authors introduced a genetic algorithm (GA) to deploy the minimum number of CHs to satisfy the requirements.

Hassan et al. [85] presented an idea of multi-level sleep scheduling (MLSS) for cluster-based WSNs. The duration of sleep is calculated based on the interval between the arrival times of two consecutive packets. The predicted interval is calculated by the exponential smooth average (ESA) scheme. After finishing the calculation of the interval, the system triggers the sleep mode of each node. The drawback of this scheme is that it is not suitable for delay-sensitive WSNs.

We summarize some of the node scheduling schemes in Table 2.2. Node scheduling is a widely-used method for achieving energy-efficient WSNs. Like the examples provided above, the network lifetime will be prolonged by preserving energy in each sensor node. However, certain node scheduling schemes need to consider the delay requirements of the system due to the sleep/wakeup delay. Moreover, the overhead on the wakeup message transmission needs to be decreased. Meanwhile, the topology of the network is changed due to the switch of the node's states. Therefore, the matter of designing a routing protocol that can face frequent-changing topology is a challenge. A collection of routing strategies regarding energy conservation will be introduced and discussed in the following section.

Table 2.2: Node scheduling approaches

Scheme	Performance Requirements	System Architecture	State Exchanging Scheme	Technicals	Strengths/ Limitations
Schurgers et al. [81]	Energy Efficiency & Reasonable Delay	Flat	Event-driven	Model Formulation	Strength: higher energy conservation Limitation: high overhead
Boukerche et al. [3]	Energy Efficiency & Event Detection	Flat	Event-driven	Model Formulation	Strength: suitable for all scale networks
Tseng et al. [82]	Energy Efficiency	Flat	Event-driven & Time-driven	Game theory & Nash Equilibrium	Strength: balance the workloads of sensors Limitation: huge density of sensors
Deng et al. [83]	Energy Efficiency & Coverage	Cluster	Event-driven	Genetic algorithm	Strength: ensure the coverage Limitation: unequal energy consumption
Lin et al. [84]	Energy Efficiency & Coverage & Connectivity	Cluster	Time-driven	Genetic Algorithm	Strength: low computation complexity
Hassan et al. [85]	Energy Efficiency	Cluster	Time-driven	Exponential Smooth Average scheme	Limitation: long data latency
Kang et al. [86]	Energy Efficiency & Coverage & Connectivity	Flat	Time-driven/ Event-driven	duty-cycling technique	Strength: suitable complex systems Limitation: sensors need always to listen to transmission channel

2.2.4 Data Routing Schemes

In WSNs, it is essential to ensure that the data from each sensor can be forwarded to the data sink. In other words, the route from the sensor to the sink should be reliable [87, 88, 89, 90]. For achieving guaranteed and reliable data transmission, a carefully designed routing scheme is necessary. Meanwhile, since the lifetime of a WSN is directly affected by the lifetime of the nodes in the system [7], maximizing the efficiency of nodes' energy usage and the overall energy utilization of network is one of the main objectives of designing a routing scheme for the WSN. One of the most direct ways to design an energy-efficient route is to try to shorten the data transmission distance; this is also one of the most widely adopted routing metrics. Currently, many energy-efficient routing protocols have been proposed. In the following, we will discuss some of the state-of-the-art protocols in detail. Similar to Section 2.2.3, we also categorize the routing protocols based on the system architectures in WSNs.

First, we introduce some routing schemes in the flat-based WSNs. Based on characteristics of the flat-based WSN, routing protocols are generally designed in a hop-based manner, i.e., the objective of these protocols is to find a short path between each pair of sensors until the route reaches the data center.

Chang et al. [91] proposed a distributed shortest path routing algorithm. They formulated the maximum lifetime routing problem as an LP problem. The purpose of their formulation was to find the link cost function to derive the maximum network lifetime. The link cost is calculated based on the energy consumption of data transmissions and the residual energy levels of two end-nodes on a link, which is given by,

$$cost_{ij} = \left(e_{ij}^{t}\right)^{x1} \underline{E}_i^{-x2} E_i^{x3} + \left(e_{ij}^{r}\right)^{x1} \underline{E}_j^{-x2} E_j^{x3}, \tag{2.2}$$

where e_{ij}^{t} is the energy consumption for transmitting a message and e_{ij}^{r} is the energy consumption for receiving a message. E_i and E_j are the initial energy. $\underline{E}_i$ and $\underline{E}_j$ are the residual energy. $x1$, $x2$, $x3$ are non-negative weighting factors for each parameter. To simplify the proposed optimization problem, they adopted a heuristic flow augmentation algorithm. In this work, for each hop, the link cost for each candidate for the next hop should be calculated. The node with the minimum link cost will be chosen as the next hop. Finally, the network can get a routing table with the minimum cost path.

Similarly, Coutinho et al. introduced a transmission power control-based opportunistic routing (TCOR) algorithm in [15]. Opportunistic routing as a new routing paradigm can guarantee reliable communications due to the properties of the wireless medium [92], e.g., broadcast nature, spatial diversity, etc. It is helpful for decreasing the energy consumed by retransmissions and adjusting the workloads of critical sensors dynamically. The objective of the TCOR algorithm is to reduce the energy consumption of data transmissions while ensuring reliable communication. This algorithm has two phases. The first phase is the neighborhood discovery. In this phase, each node will establish its neighboring table, which contains neighbors' location information and the corresponding link condition information. To achieve this goal, each node needs to broadcast beacon messages in its vicinity. Meanwhile, by receiving a beacon message, the node calculates the link error probability between the sender and itself and updates its neighboring table. The second phase is the forwarder set selection. The selected set must satisfy the following conditions: 1) The probability of every node in the forwarder set should be greater than or equal to the predetermined threshold; and 2) The energy consumption of the forwarder set is minimal. Then, the next hop of each sensor is selected through TCOR. The advantage of the proposed algorithm is that it improves energy efficiency and achieves system-wide communication reliability.

Moreover, based on the Particle Swarm Optimization (PSO) method, Liu et al. [93] proposed an agent-assisted Quality-of-Service-based routing algorithm (QoS-PSO) for WSNs to derive the optimal path. This algorithm improves energy efficiency and ensures low data latency. The drawback of QoS-PSO is that it is not applicable to the large-scale WSNs because the system needs to store a significant amount of information for each sensor.

In addition to the routing protocols in flat-based WSNs, many routing protocols have been proposed for hierarchical-based WSNs. In [94], a novel cluster arrangement energy-efficient routing protocol algorithm (CAERP) was proposed. This protocol was composed of three parts: cluster formation algorithm, cluster head selection algorithm and a routing algorithm. For clustering, they introduced an uneven clustering algorithm, i.e., the sizes of clusters in the network are different. Clusters that are closer to the sink are smaller than those that are farther from the sink. The purpose of the uneven clustering is to save cluster heads' energy for communications with inter-clustered nodes since they should spend more energy on communication with other cluster heads. Each cluster head is selected based on both the distance from the sink and the residual energy. For routing algorithms, the CH (or cluster member) simply chooses the nearest CH (or CH/cluster member) as its next hop. The CAERP is simple to implement and can solve the initial dead node problem. However, the node that is far from the sink will run out of its energy quickly.

In [95], a multipath routing protocol (MRP) was proposed. This approach applied dynamic clustering and ant colony optimization that has introduced in [59]. The ants can find food faster over the shortest path, so the pheromone of this path is increased. The best path is eventually chosen according to the highest value of pheromone. The search ant (SANT) is responsible for information gathering about paths to the sink and intermediate nodes on the path. The backward ant (BANT) is responsible for updating the pheromone value of the path from the sink to the source node. The abnormal ant (AANT) is responsible for avoiding the suspension of the scheme. The process of finding the best path from a CH to the sink has two steps: First, the CH generates a set of SANTs to find the sink, and SANTs collect path information. The AANT is created based on the probability that the SANT can get to an intermediate node. Second, the sink generates a BANT once a SANT arrives at the sink. The BANT goes back to the source node while updating the pheromone, and finally brings path information back to the source node. Each node can then get the optimal path information according to the information from BANT. If the number of optimal paths is more than one (i.e., there is more than one path with the same pheromone value), the source node will choose one of them randomly. MRP is helpful for improving data collection efficiency and system connectivity. Here, we summarize the features of the routing protocols discussed above in Table 2.3.

2.2.5 Joint Energy-efficient Designs in Conventional WSNs

The WSN is a somewhat complex system, and many parameters in the system affect the system performance. Typically, to achieve the desired system performance, an applicable WSN system needs to make the trade-off between multiple performance requirements. Accordingly, to improve the overall performance of WSNs, the joint energy-efficient design that considers multiple inter-dependent methods (namely, node deployment, node scheduling, routing, etc.) is necessary. In this section, we will discuss two kinds of joint

Table 2.3: Routing Schemes

Protocol	Performance Requirements	System Architecture	Known Position	Techniques	Strengths/ Limitations
Zeng et al. [96]	Energy efficiency	Flat	No	HS algorithm & Local search strategy	Limitation: does not consider the distance between sensors
Coutinho et al. [15]	Energy efficiency & Reliable Communication	Flat	Yes	Formulation with packet delivery probability estimation	Limitation: long data latency
Liu et al. [93]	Energy efficiency & Sufficient QoS	Flat	Yes	Particle swarm optimization algorithm	Limitation: not suitable to the huge-scale WSNs
Vijayan et al. [94]	Energy efficiency	Cluster	Yes	Uneven clustering mechanism	Limitation: high message overhead
Zungeru et al. [95]	Energy efficiency	Cluster	No	Dynamic clustering & Ant colony optimization	Strength: high data collection efficiency
Chang et al. [91]	Energy efficiency	Flat	No	Linear programming problem & Flow augmentation algorithm	Limitation: high cost on path switching

energy-efficient approaches.

First, we will introduce some schemes by jointly considering node deployment and node scheduling. Mini et al. [97] proposed a novel joint sensor deployment and scheduling scheme to optimize the performance of WSNs. In this work, their goal was to improve energy efficiency and achieve the target coverage mentioned in [7]. They applied the artificial bee colony algorithm [98] to compute the locations of deployed sensors. In the beginning, the system derives the initial solution, in which all targets can be covered by k sensors. To reduce the energy wasted by the task-free sensors, each deployed sensor must cover at least one target. In the artificial bee colony algorithm, the fitness function is formulated to evaluate the quality of the potential/candidate solutions. In this work, the fitness function is related to the network lifetime, which is calculated as,

$$F = \min_j \left\lfloor \frac{\sum_i M_{ij} * b_i}{k} \right\rfloor, \tag{2.3}$$

where M_{ij} is a coverage metric that represents the sensor S_i monitors the target T_j. $b_i = \frac{initial\ energy}{energy\ consumption\ rate}$ is the battery lifetime. The system looks for new solutions by implementing the artificial bee colony algorithm in the neighborhood. The solution is replaced by the new one which is superior to the original solution. The process will repeat until it has satisfied the terminal conditions, and then the near-optimal solution can be obtained. The results showed that system performance was improved regarding the network lifetime and system coverage degree. The authors in [4] developed another joint node deployment and scheduling scheme for group-based industrial WSNs [99]. The sensors in the industrial WSN are divided into several groups. Accordingly, when two groups of nodes need to communicate with each other, the data must be transmitted by the nodes located in the intersection region of these two groups. Consequently, the nodes in the intersection region will consume more energy than other nodes. Therefore, the key point of this scheme is the node deployment in the intersection region. This strategy has two stages. In the first stage, an improved geometric selective harmony search algorithm [100] was adopted to determine the node deployment position and the number of sleep schedules in order to minimize overall energy consumption. Fig. 2.5 shows the flowchart of this work, where $par(g)$ is an increasing linear function of g, and $bw(g)$ is an exponential function of g. Then, the system enters the second stage and checks whether there is an available sensor between the centers of two neighboring groups. If so, the system would determine the sleep schedules in the next round based on the improved geometric selective harmony search algorithm. Then it repeats stage 2 until there is no redundant sensor.

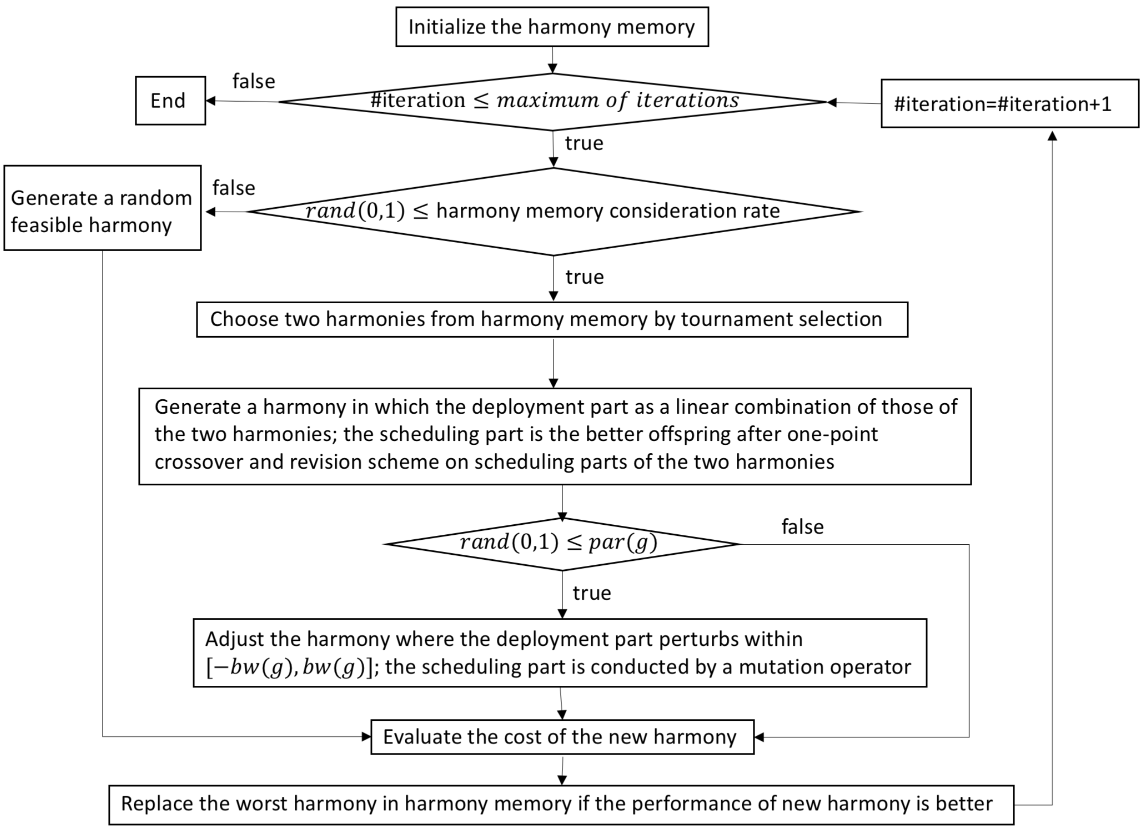


Figure 2.5: The proposed improved geometric selective harmony search algorithm in [4]

In addition to the joint node deployment and scheduling schemes mentioned above, the joint routing and scheduling algorithms also attracted much attention from researchers. Hsu et al. [92] proposed a joint Asynchronous Sleep-wake Schedules and Opportunistic Routing Technology (ASSORT) to prolong the network lifetime. The authors adopted the asynchronous sleep-wake scheduling that could be easily implemented in WSNs and could reduce the overhead on clock synchronization. To simplify the mathematical analysis, the authors adopted the Poisson process. Accordingly, the sleep period is represented by an exponentially random variable with the reciprocal of the wake-up rate λ. The wakeup period is donated by Δ_{wake}. The value of these two parameters is derived by a simple search algorithm introduced in [101]. All sensors will operate the sleep-wakeup scheduling. When a sensor has a message to send, its scheduling is suspended. Then, the node starts the probing period in which the node decides its forwarder. For the routing algorithm, the forwarder for each node is selected from its multiple-forwarder set, which is derived by the proposed metric called opportunistic energy cost with a sleep-wake schedule (OECS). Residual energy is an important factor in the OECS. Briefly, a sensor with less energy should focus on saving energy. Accordingly, the energy cost of the sensor node is proportional to the remaining energy. The OECS value is related to the cost of probing C_{prob}, waking C_{wake}, transmitting data C_{tx}, receiving data C_{rx} and forwarding data to sink C_{fwd}, which is given by

$$OECS_u(F_u, P_u) = \frac{C_{prob} + C_{wake} + C_{tx} + C_{rx} + C_{fwd}}{P_{TS}}, \tag{2.4}$$

where P_{TS} is the probability that means at least one sender will receive data. Initially, the OECS values of the sensors and the sink are infinite and zero, respectively. Once a node receives the OECS value of its neighbors, it updates its OECS value and broadcasts its OECS to its neighbors. Each sensor adds the neighbor node with the smallest OECS value to its multiple-forwarder set. When a sensor has a message to send, its scheduling is suspended. Then, the node starts the probing period in which the node decides its forwarder. This node broadcasts a beacon message to its multiple-forwarder set. By receiving an ACK message from its awake neighbor, the node selects this neighbor as the forwarder and sends the data to it. If the node does not need to work, it will turn to sleep. The advantage of the proposed scheme is the reduction of the system overhead while providing reliable communication.

In [102], authors developed a node scheduling scheme by taking advantage of geographic routing in WSNs. Typically, in geographic routing, the position information is available to all sensors by using the Global Position System (GPS) or other mobility-based localization approaches, e.g., [103]. The authors proposed two kinds of geographic distance-based connected-k neighborhood node scheduling schemes. In this scheme, the forwarding information was decided based on the position of nodes. The first proposed scheme focused on the shorter first transmission path, called the Geographic distance-based Connected-k Neighborhood for the First path (GCKNF) algorithm. Another scheme focuses on all routing paths for multipath transmission, which is called the Geographic-distance-based Connected-k Neighborhood for All paths (GCKNA) algorithm. To implement these two algorithms, the authors made two assumptions: 1) To save energy, the node with at least

k awake neighbors needs to be put into sleep state as well as ensure it is k-connected. 2) The neighbor of each node that is closest to the sink should be awake, which can achieve the shortest path for data transmission. In GCKNF, each node first checks if it has k neighbors, otherwise, it increases its transmission range until it belongs to a connect-k neighborhood. Then, the node chooses its neighbor located closest to the sink as its next hop. Next, the node selects a random rank and records its neighbors whose ranks exceed its own rank into a list named C_u. If the size of C_u of a node exceeds the number k and this node is not the next hop for others, the node will be put to sleep. In GCKNA, the node checks the connected-k neighborhood, which is the same as the first step in GCKNF. Each node then calculates the geographic distance between itself and the sink and records its awake neighbors whose distances from the sink are less than its distance into C_u. The node will be asleep if the size of C_u is greater than k. This scheme can achieve low computation complexity. However, it is not suitable for sparse WSNs. For example, in the worst case, each node in the system is the next hop for one of the other nodes. Thus, all the nodes need to stay in the active state. We summarize the discussed joint energy-efficient schemes in Table 2.4.

Table 2.4: Joint-designed conventional methods

Protocol	Combined Methods	Performance Requirements	System Architecture	Techniques	Strengths/ Limitations
Mini et al. [97]	Node deployment & Node scheduling	Energy efficiency & Target Coverage	Flat	Artificial bee colony algorithm	Strengths: high energy efficiency & achieving coverage
Lin et al. [4]	Node deployment & Node scheduling	Energy efficiency	Flat	Improved geometric selective harmony search algorithm	Strength: high energy efficiency
Hsu et al. [92]	Node scheduling & Routing	Energy efficiency & Reliable communication	Flat	Poisson process	Strength: improving the link reliability
Buratti et al. [104]	Node scheduling & Routing	Energy efficiency & Throughput	Flat	Dijkstra's algorithm & DSATUR algorithm [105]	Strengths: high delivery ratio & low data latency
Zhu et al. [102]	Node scheduling & Routing	Energy efficiency & Connectivity	Flat	*Connect-k neighborhood* basis	Limitation: not suitable to the low density of sensors
Elsersy et al. [106]	Node deployment & Routing	Energy efficiency & Information quality	Flat	Genetic algorithm & Effective independence model	Strength: low computing complexity

For the conventional WSN, we discussed five main types of energy-efficient approaches. The approaches designed for the traditional WSNs are helpful for reducing the energy consumption rate or minimizing the working period for the deployed sensors. However, due to the limited capacity of the built-in battery, the system lifetime of the battery-powered WSN is still limited with conventional methods. To improve energy efficiency more effectively, new techniques are developed and adopted in WSNs, e.g., high-mobility data collectors, energy-harvesting techniques, etc. The approaches with these new techniques can prolong the network lifetime and even make the semi-permanent WSNs. In the next section, we will provide a detailed discussion of some recent energy-efficient approaches designed based on emerging techniques.

2.3 Emerging Techniques-assisted Methods

The emerging techniques-assisted methods for energy efficiency in WSNs are mainly classified into four types: mobile data collection schemes, EH-aided approaches, machine learning-based algorithms, cognitive networks. Mobile data collection schemes can effectively decrease the energy consumption of sensors. Besides, EH-aided approaches make sensors harvest energy from other devices to prolong the lifetime of the WSN. In the following, a set of emerging techniques-assisted methods for energy efficiency in WSNs are presented in detail.

2.3.1 Mobile Data Collection Schemes

In recent years, depending on the booming development of unmanned vehicle techniques, many mobile devices have been developed (e.g., UAV, automatic vehicle (AV), etc.) and widely adopted in both military and civilian applications [107, 108]. Data collection with new mobile devices is an emerging and effective approach for energy conservation. Compared with traditional mobile sensor nodes, the emerging mobile devices (e.g., UAV) have a much faster-moving speed, longer deployment range, and relatively longer operating time. Hence, by installing the sensing and communication equipment, these emerging mobile devices can be considered as sensors with high mobility. Additionally, due to the mobility of the mobile devices, they can arrive at any position and visit any node to collect data from it, which saves more energy than is consumed by forwarding data by sensors. There are two main kinds of approaches with different types of mobile devices: mobile-sink based and mobile-data-collector based. We will survey the related approaches in detail in the following.

Mobile-sink-based Approaches

The greatest benefits of using mobile sinks are as follows: 1) Avoiding long-hop relaying: The mobile sink will move according to different metrics, such as the probability of target detection or the density of sensor nodes. Therefore, the sensor could transmit the data to the sink with a small number of hops. 2) Balancing the workload of the whole network: In this case, no node is always near the sink. It can eliminate the energy-hole problem. Currently, depending on the adopted mobility model, existing mobile-sink based approaches can be categorized into two types: uncontrollable mobility (e.g., random mobility model) and controllable mobility (e.g., unrestricted model, geographically restricted model). We will introduce several schemes according to these two types of mobility models, respectively.

Normally, the existing approaches for the uncontrollable mobile sink adopt the random mobility model which means the data sink will move in a random manner, i.e., randomly selected moving velocity (both direction and speed), or randomly selected moving trajectory. Accordingly, instead of using a guaranteed coverage degree to measure the system performance, the desired coverage probability is adopted in this type of scenario. Based on the adopted random mobility model, the authors in [109] proposed an LP formulation for assigning different staying periods for the mobile sink in different locations of the FoI. In the presented work, authors assumed that the FoI was divided into multiple grids of identical size and all nodes with the identical initial energy e_0 were uniformly located on the vertex of each grid. To maximize the system lifetime, by considering both the energy constraints of each sensor node and the staying period constraint, the proposed LP-based optimization function for maximizing system lifetime z is formulated as follows:

$$\max z = \sum_{k \in N} t_k, \tag{2.5}$$

$$s.t. \quad \sum_{k \in N} c_i^k t_k \leq e_0 \quad i \in N, \tag{2.6}$$

$$t_k \geq 0 \quad k \in N, \tag{2.7}$$

where t_k is the stay period at node k, c_i^k is energy consumption of the node i when the sink stops at node k. Eq. (2.6) expresses that the consumed energy could not exceed the initial energy for each node, and Eq. (2.7) states that the time of the sink stays at one point must be bigger than zero. They analyzed the results of the simulation and concluded that the energy consumption was evenly distributed among all sensors.

Contrary to the uncontrollable mobile sink approaches mentioned above, the mobile data sink in the controllable mobility scenario moves in a deterministic manner, i.e., the mobile data sink moves in a predetermined cruise trajectory to achieve the guaranteed system coverage degree. This kind of approaches can achieve high efficiency for monitoring the FoI with the fixed event occurring probabilities in different areas, or for detecting the static target with pre-known location information. However, for detecting mobile targets or monitoring the FoI with random events occurring probabilities in different areas, the efficiency of this kind of deterministic approaches could be low, since this kind of predetermined algorithms lack the flexibility to cope with unpredictable changes in the system.

Konstantopoulos et al. [5] proposed a rendezvous-based approach with a mobile sink, which belongs to path-restricted mobility. The objectives of this protocol were to minimize the overall network overhead and energy expenditure and balance the energy consumption among sensors to prolong the network lifetime. The authors made several assumptions of the protocol: 1) The location of each sensor is known in advance. 2) The mobile sink is located on a public transportation vehicle, which travels along a fixed path. This presented algorithm has two phases: *setup phase* and *steady phase*. In the first phase, all sensors were divided into multiple clusters of different sizes based on the clustering algorithm introduced in [110], which is similar to the uneven clustering algorithm adopted in [94]. In each cluster, the CH was selected mainly depending on the residual energy of each cluster member and its distance to the data sink's trajectory (derived based on the received strength of the beacon message broadcasted by the data sink with fixed power). The cluster member with the largest residual energy has the highest probability of being selected as the CH. Except for CHs and cluster members, the authors adopted a special type of node: rendezvous node (RN). RN is responsible for sensing and gathering the data, and is ready to send the data to the mobile sink once the sink appears in its communication range. For RN selection, the candidates must be a set of nodes that can receive the beacon message directly from the sink. Each RN candidate calculates its competence value which is related to its residual energy and its communication probability with sinks and channel conditions. The candidate then sends its value to its cluster heads. The cluster head will choose several candidates with high competence value to be RNs in its cluster. After selecting RNs, cluster heads attach to RNs and send the data to them. An example of the network model is depicted in Fig. 2.6. In the next phase, all the sensor nodes start sending gathered data to its CH. Each CH sends the filtered data to its neighboring cluster head until reaching the end cluster head u that sends the collected data to the neighboring RN. Cluster head u should distribute the data according to the transmission capability of RNs, and then forward filtered data to its RNs. Finally, the RN forwards data to the mobile sink once the sink enters its communication range. The advantage of the proposed scheme is that it prolongs network lifetime while keeping a high average residual energy level.

Similarly, Liang et al. [111] developed an approach by adopting a mobile sink in WSNs. This approach was designed based on the location-restricted mobile model in which the mobile sink can only visit certain designated locations. The objective of this work is to design the optimal path for the mobile sink with the constrained moving range. This moving range constraint is used to limit the energy consumption of the mobile sink. In addition, the presented work also aimed to avoid data loss when the sink modified its locations and reduce the overhead of routing path construction. Relying on the proposed mixed-integer LP method, the routing decision was derived based on an optimization func-

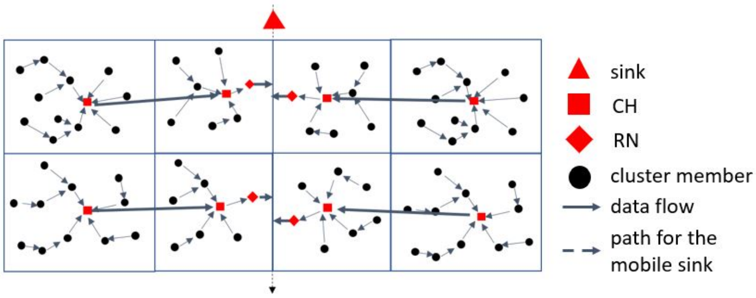


Figure 2.6: The network model introduced in [5]

tion formulated to face the constraint of the mobile sink's moving range. Furthermore, the authors developed a staying time scheduling scheme for the mobile sink with the heuristic algorithm.

The authors in [112] proposed another energy-efficient path selection algorithm with the mobile sink, called reduced k-means (RkM). They adopted k-means clustering scheme [113] to choose a set of candidates of CHs where the mobile sink needs to visit, and then optimized it to derive a minimum number of CHs with respect to three factors: 1) the number of one-hop neighbors; 2) the distance to the extreme sensors that are a set of sensors closest to the border of the FoI; and 3) the average distance to one-hop neighbors. Briefly, the system adds all candidates into a set C, calculates their weight values based on the three parameters mentioned above and sorts them in descending order. The candidate with the highest value is selected to be the CH. The selected CH and its neighbors form a cluster. Then, all the nodes within this cluster are removed from C. This process is repeated until no candidate is left in the C. After selecting all CHs, the system uses Christofias's heuristic algorithm [114] to derive the path of the mobile sink for visiting all CHs. This algorithm can be implemented easily and its computational complexity is low. However, it is not suitable for the large-scale sparse WSNs, since the number of CHs will be large, which in turn increases the moving distance of the mobile sink.

Mobile-data-collector-based Approaches

In addition to mobile-sink based approaches, an alternative method is using the mobile data collector to collect data from the sensor nodes. Using the mobile data collectors is helpful for distributing the workloads more evenly among sensors. The path of the mobile data collector is designated by the data sink or the user in advance. There are two methods of communication in mobile data collection schemes: one-hop communication and multi-hop communication. For one-hop communication, the sensors only transmit the data to the mobile data collector directly, without any intermediate node. On the other hand, the data from one sensor is sent to the mobile data collector through relay nodes in multi-hop communication.

The most straightforward way to design the path for the mobile data collector is to visit all nodes one by one with one-hop communication. However, this kind of schemes may only be suitable for sparse networks or applications with a small number of sensors. It is not practical for large-scale dense WSNs because the data collector needs to visit too many sensors one-by-one, which will introduce a greater traveling distance of the data collector, which, in turn, causes long data latency. Therefore, the cluster-based data collection schemes with one-hop communication are developed by exploring the advantages of wireless communication. More precisely, the sensors within a predetermined range are grouped into a cluster. They will communicate with the mobile data collector once the collector enters the cluster. The advantage of one-hop-based schemes is that they reduce the transmission traffic and energy consumption from forwarding. Here, we will introduce some schemes designed based on one-hop communication.

Liu et al. [115] proposed an energy-efficient data gathering scheme with a mobile data collector in the cluster-based WSN. They developed a cluster-based genetic algorithm to figure out the optimal path for the data collector while improving the network performance. They determined that the network should first be divided into several clusters according to a clustering algorithm based on Euclidean distance introduced in [116]. A virtual cluster head point called waypoint in this work is selected in the overlapped sensing range of all sensors in the cluster. The sensors can transmit the data to the mobile data collector with one-hop communication once the data collector arrives at the waypoint in the cluster. The authors applied GA to calculate the path for the data collector which is composed of all waypoints. This scheme is highly effective for data transmission. However, the average distance of data transmission may be considerable, since the communication link between the data collector and the sensor located far away from the waypoint could be long. In a worst-case scenario, all the nodes located on the boundary of the cluster would have to use the maximum transmission power to transmit data.

Xie et al. [117] introduced a heuristic tour-planning algorithm with multiple mobile data collectors to improve the energy efficiency of WSNs. In this algorithm, they considered the effects of the obstacles located in the FoI. The algorithm has three phases: First, the sensors execute the clustering algorithm. Due to the obstacles, mobile data collectors cannot stay at the position covered by obstacles and have to avoid them. Therefore, the shape of obstacles should be regularized to plan the route for mobile data collectors in the second phase. To find the shortest path between any given pair of sensors, the spanning graph is constructed based on the line sweep algorithm [118]. Then, the complete graph for mobile data collectors is developed on the basis of the spanning graph to derive an obstacle-avoiding shortest tour, which is similar to the traveling salesman problem. Due to the high computational complexity for constructing the complete graph, they applied Warshall-Floyd algorithm [119] to derive the result.

Even though the one-hop-communication-based approaches try to minimize the effect of the relatively long traveling distance to the data center, the data forwarding latency caused by the long route of the mobile data collector is still noticeable for some delay-sensitive applications. Additionally, in order to achieve one-hop communication, some sensors need to enlarge their transmission range [120], which will introduce extra energy consumption in certain nodes. To overcome this limitation, another method of communication is developed, known as multiple-hop communication, as mentioned above. This method is more suitable for a large-scale system with densely deployed nodes. Due to the high node density in the FoI, it is unnecessary to visit every node. The network architecture for multiple-hop communication is typically cluster-based. In contrast to the cluster with one-hop communication, the size of the clusters in multi-hop communication is larger. The cluster members need to send the data to the mobile sinks through their CHs or other relaying nodes. It is good to achieve a short path for mobile devices so that data latency can be decreased.

An approach using multi-hop communication was proposed by Singh et al. in [121]. In the proposed approach, two data mules as mobile devices were applied to data collection. Their paths of data mules are restricted to the leftmost and rightmost of the whole FoI. Additionally, they designed an odd-even round number rule to determine the direction of the data transmission in each round. This approach is divided into three phases: 1) The first phase is clustering and cluster head selection. The whole network is divided into multiple grids with an identical size. To reduce the energy consumption rate, the distance between two sensor nodes should remain less than d_0 which determined by users in advance. In order to ensure that the distance between two cluster heads is less than d_0, the width x of grid is less than $\frac{d_0}{\sqrt{5}}$ and the number of clusters is $N = \frac{network\ area}{x^2}$. The first phase terminates once the system fulfills clustering formation. The node closest to the center of the grid is selected as the CH. The data generated within the grid will be forwarded to the CH. 2) The second phase is the design of the routing algorithm with odd-even round numbers. Cluster members transmit data to their CH in a time division multiple access (TDMA)-based manner. For the CH, the sink notifies CHs about the current round number. If the round number is odd, each CH forwards data to its left neighboring CH. The CHs located close to the leftmost boundary of the FoI gather data sent from other CHs and upload them to the data mule. Otherwise, the procedure operates in the opposite direction. 3) The last step is data collection by data mules. The data mules traverse along their stipulated routes, receive the data from CHs, and then transfer to the sink as they approach it. This step is useful for keeping a balanced workload among all CHs. However, the energy consumption of CHs is still relatively high because they need to do a large number of data transmissions, especially in large-scale networks. Additionally, although the path of the data collector is fixed and short, it must wait at every stop point for the data from all clusters. This may generate additional data latency.

Moreover, some protocols employ the UAV to collect data from sensors. In [122], a joint node scheduling and routing scheme was proposed for the UAV-based WSN to conserve energy within the whole network. First, the authors formulated the problem as a mixed-integer non-convex optimization problem. The authors assumed that there are K sensors in the network. In addition, they assumed that there is at least one route for UAV from the start node q_o to the destination q_F within a predetermined time slot T with the max speed of UAV V_{max}. They also considered the node scheduling with respect to the transmission rate of sensors, which is defined by $R_k[m]$, where m $(1 \leq m \leq M)$ donates the m-th time slot. The formulation with node schedule $\mathbf{X}$ and route of UAV $\mathbf{Q}$ is shown below:

$$\min_{X,Q,\theta} \quad \theta \tag{2.8}$$

$$s.t. \quad \sum_{m=1}^{M} x_k[m]E_T \leq \theta, \qquad \forall k, \tag{2.9}$$

$$\sum_{m=1}^{M} x_k[m]R_k[m] \geq r_k, \qquad \forall k, \tag{2.10}$$

$$\sum_{k=1}^{K} x_k[m] \leq 1, \qquad x_k[m] \in \{0,1\}, \qquad \forall m, \tag{2.11}$$

$$\Big\|q[m] - q[m-1]\Big\| \leq D_{max}, \qquad \forall m \geq 2 \tag{2.12}$$

$$q[1] = q_0, q[M] = q_F, \tag{2.13}$$

where θ is the slack variable indicating the maximum energy that can be consumed in the data transmission to be diminished. The constraint (2.9) ensures the energy consumed by all sensors will not surpass θ. The constraint (2.10) ensures the amount of data from each sensor is collected. The constraint (2.11) guarantees only one sensor can be awake and communicating with the UAV at one time slot, where $x_k[m] = 1$ is the sensor awake at time slot m; otherwise, $x_k[m] = 0$. The constraints (2.12) and (2.13) represent the speed, initial and final position constraints for UAV. However, the optimization problem is non-convex and is hard to be solved optimally. Therefore, they used the successive convex optimization technique [123] to derive a sub-optimal solution for this problem.

Mobile data collection algorithms, considered an emerging method, can balance the workload among all sensors, and relieve traffic flow for some specific nodes. Additionally, mobile devices can gather data consistently because they can recharge their batteries after returning to the sink. Table 2.5 recapitulates the different data collection protocols with mobile elements presented in this section.

Table 2.5: Mobile data collection schemes

Scheme	Mobile element	Network Architecture	Data collection strategy	Mobility mode	Strengths & Limitations
Wang et al. [109]	Mobile sink	Flat	One-hop	Random	Strength: simple implementation Limitation: low time efficiency
Konstantopoulos et al. [5]	Mobile sink	Cluster	Multi-hop	Fix	Strength: relatively balanced workloads of all sensors Limitation: long data update period
Liang et al. [111]	Mobile sink	Flat	Multi-hop	Fix	Strength: suitable for time-sensitive systems
Kaswan et al. [112]	Mobile sink	Cluster	Multi-hop	Fix	Limitation: not suitable for WSNs with low density of sensors
Shi et al. [124]	Mobile sink	Flat	One-hop & Multi-hop	Fix	Limitation: not suitable for large-scale WSNs
Liu et al. [115]	Mobile data collector	Cluster	One-hop	Fix	Limitation: not suitable for sparse WSNs
Singh et al. [121]	Multiple mobile data collectors	Cluster	Multi-hop	Fix	Strengths: achieve high energy efficiency & scalablity
Zhan et al. [122]	Mobile data collector	Flat	Multi-hop	Fix	Strengths: high energy efficiency & combining node scheduling
Xie et al. [117]	Multiple mobile data collectors	Cluster	One-hop	Fix	Strength: considering the obstacle Limitation: long data update period

2.3.2 Energy Harvesting-aided WSNs

Since the approaches described above (i.e., conventional methods and mobile data collection algorithms) were designed based on the battery-powered WSN, and the capacity of the battery is limited, the system lifetime of these proposed approaches is still constrained. Therefore, another new technology known as energy harvesting has drawn more attention and has been introduced into WSNs in recent years. Energy transfer/harvesting is considered as one of the most effective technologies to improve system energy efficiency. It transfers the lifetime of the system from limited to semi-permanent. The sensor can gain energy from various sources which are classified into two categories: ambient sources and external sources. The ambient sources (e.g., radio frequency (RF), solar power, thermal energy, etc.) are no cost, but they are unstable due to the weather and environmental conditions [46]. External sources, such as mechanical- and human-based devices (e.g., MC), are deployed in WSNs for energy harvesting purposes. There are two methods for transferring energy from the energy sources to deployed sensors: direct-contact charging and wireless charging [125]. In direct-contact charging, a node recharges its battery through physical contact. However, this may violate the definition of WSNs. Therefore, WSNs commonly adopt wireless charging methods. In an EH-aided network, the critical problem is how to prolong network lifetime by trading off energy harvesting and data forwarding. A number of approaches have been developed in the EH-aided WSN and can be generally classified into three types: MAC layer control schemes, EH-aided topology control schemes and wireless charger deployment schemes. We present these three types of approaches in the EH-aided WSN, respectively.

MAC Layer Control Schemes

Due to the spatiotemporal fluctuation of renewable energy, the available energy of energy harvesting-nodes (EHNs) remains uncertain. Hence, improving the utilization of the residual energy of the EHN is vital in EH-aided WSNs.

Zheng et al. [126] proposed a distributed optimization algorithm with multi-channel and multi-access in WSNs based on game theory. Each energy harvesting sensor node chooses one channel for its data transmission, while this channel is also used by other sensors for data transmission. This problem is a non-cooperative game with at least one Nash equilibrium, i.e., Pareto optimal. Additionally, they also developed a distributed algorithm for the same objective with online learning, which can converge to the Nash equilibrium of the formulated game. Another MAC layer control approach was proposed in [127]. In this work, the author designed an asynchronous collision avoidance scheme based on the hop-count information to effectively exploit the available energy of EHNs.

Moreover, in [6], a resource allocation algorithm that considered the simultaneous wireless information and power transfer (SWIPT) technique in WSN was proposed. In this protocol, the authors made an assumption regarding data transmission: If the nodes needed to transmit the data through their neighbors, they would send energy to the receivers as well. The received model of EHN is shown in Fig. 2.7. The algorithm is designed for two scenarios which are related to minimum data rate requirements of the system and the power limitation: 1) the received energy is distributed into a continuous set of streams with random ratios; and 2) the received energy is distributed into a discrete set of streams with determined ratios. The authors formulated a resource allocation problem for each scenario, which was non-convex. Thus, the authors proposed a cross-layer resource allocation algorithm. They first transformed the non-convex problem into a convex optimization problem, based on the fractional programming [128]. Then, the formulation was solved by an iterative algorithm that was designed based on Lagrange dual decomposition. Eventually, the optimal solution for distributing and splitting energy was obtained. The advantage of the proposed algorithm is that the relaying nodes can get energy from senders (even interference signals and antenna noises). Furthermore, the energy-hole problem can be avoided.

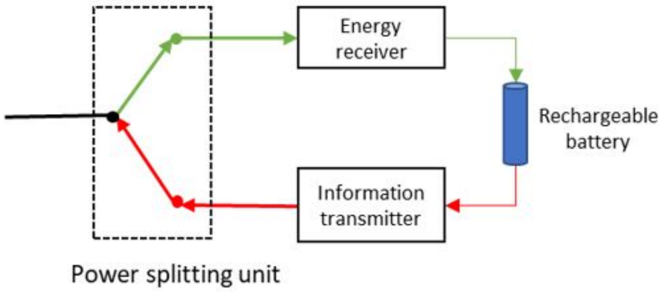


Figure 2.7: The receive model in EH-aided WSN in [6]

Energy Harvesting-aided Topology Control and Routing Schemes

As mentioned before, the ambient energy resource is unevenly distributed in the FoI. Hence, the energy gathered by each energy harvesting sensor is different. For instance, the nodes in direct sunshine can obtain more energy than those in the shadow. Based on the different level of residual energy of sensors, the topology control and routing algorithm in the EH-based WSNs are important for balancing energy consumption and harvesting of all sensors and avoid exhausting critical sensors. A data gathering optimization algorithm in energy harvesting WSNs was proposed in [129]. The objective was to determine the optimal data gathering scenario by considering the energy required for data sensing and transmission. Therefore, they developed a balanced power distribution algorithm to manage power according to the residual energy and energy requirements of sensors. Moreover, they proposed a distributed sensing and routing control scheme based on the proposed power distribution algorithm with the sub-gradient method and double decomposition method [130]. This scheme is beneficial for converging the routing problem of data gathering.

Martinez et al. [131] developed an energy-harvest-aware routing algorithm. In this algorithm, they assumed that multiple traffic flows are dependent. They took the battery capacity into consideration and formulated the routing problem with the multi-commodity to find the trade-off between the maximum the total network lifetime and the maximum minimal residual energy. The algorithm can achieve a higher average residual energy of the whole network.

Wireless Charger or Data Relaying Node Deployment Schemes

In order to make the sensor receive the sustained energy from the ambient surrounding, the chargers that can transfer the energy to the sensors are commonly deployed in the FoI. Since the harvested energy is mainly related to the distance to the chargers, the location selection for chargers is a key issue in EH-aided WSNs. Bi et al. [132] proposed a scheme for the simultaneous deployment of access points and wireless chargers in the same network. The access point is a device that collects data from sensors in this work. They assumed the sensors' positions would be known in advance and also adopted the receiver model as described in [6]. The energy harvesting or consumption of the sensor is related to the distance between itself and responding devices. To achieve a long network lifetime, they tried to find the minimal residual energy among all sensors and optimize the locations of chargers and access points. In their work, they first designed the deployment of one kind of devices under the premise that another type of devices had been deployed. For charger placement optimization with a fixed access point location, they used a bi-section search [133] to calculate the optimal location of a single charger. Then, they considered the method for deploying multiple chargers in WSNs based on k-means clustering algorithm [113] and greedy algorithm [133]. Next, they developed a multiple access points deployment scheme based on the trial-and-error method. The solution that yields more energy to sensors is the most optimal.

The authors of [134] presented a mobile energy charger routing problem in the cluster-based WSN. The mobile charger designed the path independently on the basis of the locations and the residual energy level of CHs. Furthermore, the system-wide energy balance was further achieved by bilateral trading between cluster headers with higher levels of residual energy and those with lower levels.

In addition to the wireless charger deployment scenarios discussed above, other related works have focused on EH-aided data relaying node deployment strategies. For example, in [135], a Hierarchical Two-Tier (HTT) node deployment strategy was proposed for the sustainable WSN. In this work, the authors adopted two types of nodes in the system: 1) the regular battery-powered sensor node (RSN), which is randomly deployed for monitoring the FoI; and 2) the data relaying EHN, which is deployed based on a probability density function (PDF)-based EHN deployment strategy. The latter focuses on gathering data from RSNs located in its vicinity and forwarding the collected data to the data sink. Further, an energy-efficient routing scheme was presented for helping each RSN find a forwarding path with the minimum energy consumption rate. Similar approaches can be found in [136, 137].

2.3.3 Other Energy Efficient Algorithms with Emerging Technologies

In this section, we will introduce two popular and effective methods (i.e., machine learning, cognitive network) as follows.

Machine Learning-based Energy-efficient Approaches

With the development of the computation ability of the CPU, machine learning (ML) has gradually become a popular method and adopted in many fields. It has been also applied to WSNs for energy efficiency. Machine learning aims to obtain the solution from the knowledge acquisition problem and improve system performances [138].

For instance, by exploiting the Deep learning (DL) models' (e.g., deep neural networks, etc.) ability to solve non-linear and non-convex problems [139], in [140], a novel deep learning-based channel learning scheme was developed for channel estimation problem that is adopted in multiple channel technique and is non-linear and non-convex. This work aims to minimize the mean square error of channel estimation for reducing the difference between the channel estimator and the channel coefficient. This problem mainly focuses on the channel state so that the future data is predicted for reducing the adverse impact of changes of deployment environment on channel conditions. Therefore, the authors used the deep AutoEncoder to learn the channel state information based on the harvested energy feedback. After the training procedure, the derived optimized pilot signal weight can be used for sensors in the deployed EH-aided WSN. The strengths of the deep learning-based channel estimation scheme are outperforming existing approaches of the channel estimation problem in terms of harvested energy and computing complexity.

Besides the channel condition, in the real world, the deployment environment may be affected by some other unexpected exterior factors (e.g., weather conditions, etc.) and interior factors (e.g., a varying topology caused by random node failure, etc.). Therefore, to accommodate the dynamic environment, researchers need to design approaches that can be self-adjusted as the environment condition changes. Accordingly, Reinforcement learning (RL)-based method is considered a practical solution for addressing this issue, since it can derive the appropriate policy based on the trial-and-error interactions with the environment (considered online training process) and figure out the decision to maximize the cumulative reward/agent's payoff [141]. Q-learning is a simple and typical RL-based method [142]. Q-value function (i.e., Q(s, a)) is formulated in the Q-learning to evaluate the quality of the selected action a in the state s. During the learning process, the node computes rewards based on the action in the specific state and derive the policy for selecting actions to maximize its long term rewards. For example, for the solar energy-powered WSNs, a Q-learning-based node scheduling algorithm was proposed in [143] to maximum network lifetime while maintaining the desired system coverage ratio. More precisely, at the beginning of each scheduling interval, the deployed nodes will be grouped based on the given clustering scheme. Every group member derives updates its Q-value function based on its residual energy level, current solar radiation intensity, and its battery recharging cycle. After the learning process, the node with the highest Q-value in the cluster will be active in the current scheduling period. Similar reinforcement learning-based algorithms for energy efficiency can be found in [144, 145].

Cognitive Networks

Cognitive radio is a potential technology that is helpful for achieving good spectrum utilization [146]. The main point is the cooperative relaying in cognitive networks [147]. Some existing approaches have developed energy-efficient methods in the cognitive network. This is helpful for solving the power allocation problem and improving the performance of EH-aided WSNs. In [32], the authors proposed a novel energy harvesting protocol in an underlay cognitive relay network with multiple transceivers. In another related work, a specific element is adopted in the cognitive network, i.e., spectrum sensors. Spectrum sensors are used to cooperatively detect the licensed spectrum for available channels for data sensors [148]. The authors in [149] presented a resource-allocation algorithm for the EH-aided and cognitive radio network to save the energy for sensors and ensure the sustainability of spectrum sensors. To sum up, the energy-efficient approaches designed for cognitive networks can conserve energy; meanwhile, they also improve channel utilization.

2.4 Summary

In this chapter, we summarized several kinds of state-of-the-art approaches designed for improving the energy efficiency of WSNs. The objectives of all these protocols are to prolong the network lifetime while satisfying the system requirements (e.g., reliable communication, delay tolerance). We first introduced the common architectures used in energy-efficient WSNs. Then we conducted a taxonomy for the existing energy-efficient approaches according to their applied technologies. The representative methods in each category were discussed, and we further summarized the advantages and limitations of different methods.

As far as we know, adopting mobile devices to gather data and wireless charging is a promising and practical method for improving energy efficiency in WSNs. However, the biggest problem with the mobile device-assisted data collection approach is data latency. The mobile device should gather the data from all sensors, return and forward them to the sink. Though people have designed many strategies to reduce the length of the path for the mobile device, e.g., clustering, it still spends considerable time visiting many sensors. Therefore, we will design the novel path planning strategy for a single mobile device or multiple mobile devices to reduce energy efficiency and minimize the delay in our work.

Chapter 3

Single Data Collector-assisted (SDCA) Data Collection Scheme

In this chapter, we propose a novel energy-efficient data aggregation scheme by using a single mobile collector, named single data collector-assisted (SDCA) data collection scheme in small-scale WSNs. The objective of this work is minimizing the energy consumed by sensor nodes and the data collectors in WSNs. In this work, the FoI is initially divided into multiple grids with the identical size, and the nodes in one grid form a cluster. Then we formulate the optimization problem and figure it out with a heuristic approach, i.e., GA, which can derive the optimal path for the data collector with low complexity. The content of this chapter has been published in *An energy-efficient uav-based data aggregation protocol in wireless sensor networks.*

3.1 System models and assumptions

First, we introduce the network model and the energy model of the system. The proposed schemes in the rest chapters will be designed based on the introduced system models and assumptions.

3.1.1 System Model

In our work, all the sensor nodes are randomly deployed with uniform distribution in a two-dimensional rectangle-shaped area. The set $S = \{S_1, S_2, S_3, ..., S_N\}$ denotes the set of N sensor nodes. We make several assumptions for the network model, which are listed as follows:

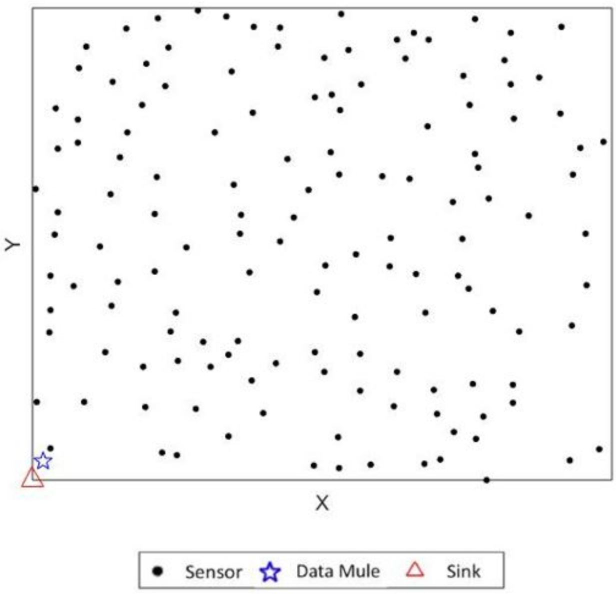


Figure 3.1: Network Model

- The FoI is divided into multiple clusters. To ensure the connectivity of the deployed system, the number of deployed nodes is derived based on the method introduced in [150].

- The sensors are stationary and cannot change their locations after deployment. They are homogeneous, which means that they have the identical sensing range R_s^s, communication range R_s^c, data buffer size Buf_s, and initial energy $IniE_s$. We adopted the same assumption widely-used in many existing approaches (e.g., [151, 152, 153, 154, 155, 156, 157, 158, 159, 160]), i.e., relying on the equipped GPS module, the deployed sensor can be aware of its location information independently, and the data sink can obtain the detailed location information of the deployed sensors in advance by gathering their location information.

- The data sink is deployed outside of the FoI and processes the data from sensors. In addition, it is responsible for figuring out the path for mobile devices.

- The mobile device has high mobility and transmission capacity. It can communicate with the sensor node when it arrives at the position of the node [161].

The network model in the 2D platform is shown in Fig. 3.1.

3.1.2 Energy Model

To evaluate the energy efficiency of WSNs, the appropriate energy model is essential. In our work, we adopted two types energy models, i.e., energy model of sensors on data transmission and energy model of mobile devices on movement. We will discuss the detailed information of these two models as follows:

Energy Model of Sensors on Data Transmission

In the WSN, the energy consumption of the deployed sensor is composed of two main parts: **data transceiving** and **sensing**. Normally, researchers consider that the energy consumed during data transmission is much higher than receiving and sensing. Table 3.1 shows the energy consumption rate for the well-known MICA2 mote sensor in different working states [1]. From this example, we can see that the energy consumption of data transmission is twice as much as that of receiving data, and forty times that of data sensing. Therefore, we will focus on the energy consumed on data transceiving. In our work, we utilize the energy model introduced in [66]. The energy consumption for forwarding a k-bit message between the distance d can be measured by,

$$E_T = E_{elec} \times k + \epsilon_{amp} \times k \times d^2, \tag{3.1}$$

where E_{elec} represents the energy consumption in the transmitter circuit, and ϵ_{amp} donates the energy consumed by the transmit amplifier.

Additionally, the energy consumption for receiving a k-bit message can be derived as,

$$E_R = E_{elec} \times k. \tag{3.2}$$

The energy model presented here indicates that the energy consumption of sensors is primarily related to the distance from a sensor to another sensor (or to the collector). Hence, shortening the transmission distance can effectively reduce the energy consumption on data transceiving.

To sum up, the energy consumption rate of a sensor P_C^s is defined by

$$P_C^s = E_T \times Data_T^s + E_R \times Data_R^s, \tag{3.3}$$

where $Data_T^s$ and $Data_R^s$ represent the transmitting data rate and the received data rate of the sensor, respectively.

Table 3.1: Energy consumption rate of a MICA2 mote sensor [1]

Working State	Energy Consumption Rate (mW)
Transmit	81
Receive	30
Idle	30
Sleep	0.003
Sense	2

Energy Model of Mobile Devices on Movement

The energy E_{move} consumed of the mobile devices to move distance D is

$$E_{move} = P_{move} \times \frac{D}{V}, \tag{3.4}$$

where P_{move} and V are the movement power rate and the velocity of the mobile device, respectively.

3.2 Problem Formulation

In this section, we fomulate the problem in this work. We only consider the energy consumption of data transmission (i.e., E_T) in this work. Recall that; the energy consumption for transmitting a k-bit message for a given distance d is given in Eq. (3.1). Therefore, in each cluster, cluster members send the sensed data to CH. Hence, the total energy consumption in the cluster j when each cluster member forwards one packet to CH is

$$E_T^j = \sum_{u \in C_j} E_{elec} \times k + \epsilon_{amp} \times k \times d^2_{(CH,u)}, \tag{3.5}$$

where CH is the cluster head of cluster j, u is the cluster member in cluster j.

For the mobile data collector, the energy is mainly consumed by movement. According to the dynamics, the energy E_{DC} required for data collector to move form current position to next CH in cluster j is

$$E_{DC}^j = P_{DC} \times \frac{D_{(current,CH_j)}}{v_{DC}}, \tag{3.6}$$

where P_{DC} and v_{DC} represent the power rate and the speed of the data collector, respectively. $D_{(current,CHj)}$ is the distance between the current location of the data collector and next CH.

In order to balance the energy expenditure between the sensor nodes and the data collector, we introduce a user-configurable coefficient σ. The coefficient can be adjusted based on the network requirements. For example, when the sensor nodes are deployed in some remote and hazardous areas, it is hard to change their batteries. In this case, compared to the energy consumption of the data collector, the energy consumption of sensor nodes is more critical. Therefore, the value of the coefficient should be higher. On the other hand, if the system requires a short data update period or less energy consumption of the data collector, the value of the coefficient should be lower. Therefore, the coefficient of energy consumption for sensor nodes should be higher. Moreover, we need normalize E_T^j and E_{DC}^j to make these two values on different scales to a common scale. E_T^j is normalized by E_T^{max} which is the maximum value of E_T^j among all clusters in the deployed WSN. Because P_{DC} and v_{DC} in (3.6) are constants, E_{DC}^j is proportional to the moving distance of the data collector. Thus, E_{DC}^j is normalized by E_{DC}^{max} which is the energy consumed by the data collector when it travels between the pair of CHs with the longest distance.

In general, the objective of our work is to minimize the energy consumption of the whole network. The corresponding optimization functions for the overall energy consumption are given by,

$$min \quad E_{total} \tag{3.7}$$

$$E_{total} = \sum_{j=1}^{M} (\sigma \times \frac{E_T^j}{E_T^{max}} + (1-\sigma) \times \frac{E_{DC}^j}{E_{DC}^{max}}) \tag{3.8}$$

where M is the number of clusters, and E_{total} is the total energy consumption of the entire system.

3.3 The Proposed Scheme

As mentioned previously, the data collector has to visit the sink and CHs of all clusters. In each cluster, a CH has to be chosen among all nodes according to the total energy consumption of all nodes in its cluster calculated by Eq. (3.5) and the distance that the data collector needs to move to it. However, the computational complexity for solving the aforementioned formulation to achieve global optimization could be huge. Moreover, it is important to decrease the computing complexity for solving optimization problem [162]. To reduce complexity, we design a novel protocol based on heuristic algorithm. GA as a heuristic algorithm is an applicable method to derive the local optimum, and it is more suitable for optimization problems in a large-scale WSN [163, 164] than other heurstic algorithms (e.g., greedy algorithm). GA firstly generates the initial population (i.e., initial solutions) and stores the population in a list based on their fitness value. In each iteration, on the basis of the initial population or the subsequent population generated by the previous iteration, new solutions would be generated through three main operations, i.e., selection, crossover, and mutation. After executing the operations with the predetermined number of iterations, the solution with the best fitness value is selected to be the final solution. To define an appropriate fitness function to measure the quality of the solution is essential to implement GA. In what follows, we will introduce the proposed scheme in detail.

3.3.1 Topology Construction Phase

Initially, sensor nodes are deployed randomly in the FoI. The FoI is divided into multiple grids with an identical size. The sensors in each grid form a cluster. Recall that; the global position information is known by the sink, and every node knows its location information. Every node is the candidate of CH in the beginning. For each CH, it gathers data from all nodes in the same cluster and then forwards gathered data to the data collector. In each cluster, to reduce overall energy consumed by sensor nodes for transmitting data to the CH, the aggregated length of the path from cluster members to the CH has to be shortened. Accordingly, the CH in each cluster is chosen by the data sink based on the sum of energy consumed by other nodes in its cluster when these nodes transmit data to it, and we introduce E_T^i to represent it.

3.3.2 Initial Population Generation

After E_T^i of all sensors is calculated, the data sink begins to figure out the path of the data collector by using GA. GA uses chromosomes to encode the solution of data [165]. In this work, each unique chromosome represents an optional path of the data collector. Every gene in the chromosome is denoted by the sink or a sensor node in the system, which is encoded by ID of each node or the sink. In this work, we use a new designed random initial algorithm to generate the initial population. To generate a new chromosome, the system randomly chooses candidate genes and adds them to the chromosome. In our design, for each chromosome, only one sensor node can be selected from each cluster. Let H be the population size. The value of H should be set large enough, which is based on the size of FoI. Assuming that there are M clusters in the area, and each cluster has at least n sensors, the number of optional trajectories for the data collector is $n^M \times M!$. Thus, the probability that we can obtain the optimal solution increases as the value of H increases.

3.3.3 Fitness

The fitness function evaluates the quality of each solution for deriving the optimum. The better chromosome with higher fitness value should be chosen. As the mentioned in [166], for the minimization problem, the fitness function is set to be the reciprocal of the objective function. Therefore, the fitness value of our work is defined as

$$F = \sum_{j=1}^{M} \left(\frac{1}{\sigma} \times \frac{E_T^{max}}{E_T^j} + \frac{1}{(1-\sigma)} \times \frac{E_{DC}^{max}}{E_{DC}^j}\right), \tag{3.9}$$

where E_T^{max} and E_{DC}^{max} is calculated by the data sink in advance.

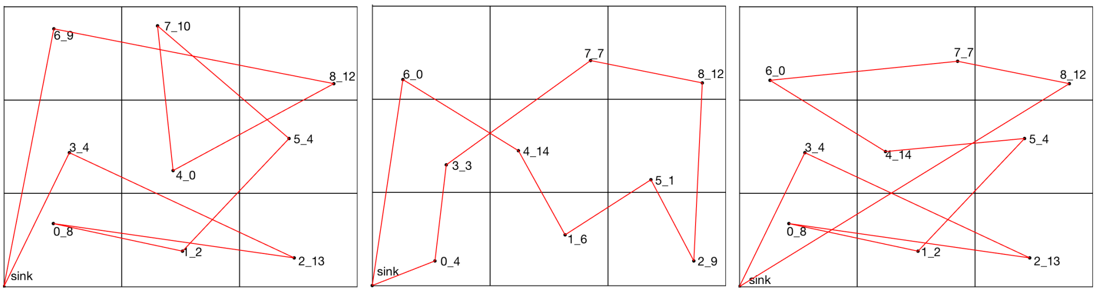
3.3.4 Selection and Crossover

Based on the generated initial population, chromosomes are sorted according to their fitness value. The new chromosome is generated by two steps: 1) Selection operation: Two chromosomes randomly are chosen from the population as two parents; 2) Crossover operation: Based on the chosen parents, one offspring is produced through crossover operation.

In GA, the crossover is an essential operation. The search space of GA is improved by crossover because new offsprings are generated constantly. Thus, the crossover probability P_C should be set high. Similar to [167], we also adopt sequential constructive crossover (SCX) [166] as a crossover operator, which guarantees that the quality of the offspring generated by SCX is higher than its parents. In other words, the offspring is closer to the optimal solution.

Let's introduce the SCX through an example which is based on the assumption in our scheme. The pair of chromosomes as parents are chosen by selection operation among the population, which are shown in Fig. 3.2(a) and Fig. 3.2(b), respectively. Here, the node is defined by i_j, where i indicates the cluster number, and j is the serial number of nodes in its cluster. We set $\sigma = 0.8$ in this example. The candidate gene with a smaller value derived by Eq. (3.8) should be added to the offspring. Recall that, it is necessary to guarantee that only one sensor in each cluster has to be visited by the data collector. The offspring is empty at first. Since the data aggregation procedure always starts from the sink, the sink has to be the initial point of all chromosomes. Starting from the sink, the candidate of the next node of the offspring in parent 1 is node 3_4, and the candidate in parent 2 is node 6_0. The value of node 3_4 is smaller than node 6_0, which are 0.3614 and 0.483, respectively. So, node 3_4 is added to the offspring and set to be the current node now. Furthermore, cluster 3 which has been visited need to be recorded. In the next step, the candidate is 2_13 in the parent 1, and the current node is not existing in the parent 2.

But node 3_3 in the parent 2 is from cluster 3, so the candidate for the parent 2 is node 7_7 which is the next node of node 3_3 in its path. At the end of this step, node 2_13 with small value is added to the offspring. To ensure only one node in each cluster should be visited, all nodes in the visited cluster will not be considered as candidate genes anymore. In the parent, the candidate genes can only be selected from one of the unvisited clusters. For example, the current node is node 0_8. For parent 2, the sink which is the next node of node 0_4 that is from cluster 0 should be chosen for the candidate. However, it has been visited in the beginning, and there are some clusters that have not been visited yet. Hence, according to the order of unvisited clusters (i.e., cluster 1, cluster 2, cluster 4, cluster 5, cluster 6, cluster 7, cluster 8), the candidate in parent 2 is node 1_6. After the offspring is generated, its fitness value should be calculated. In this example, the offspring (Fig. 3.2(c)) has a higher fitness value than its parents, which means the solution represented by the offspring is better than its parents. Then, the parents with the lower fitness value will be replaced by the offspring.



(a) Parent1: Fitness=0.214751 (b) Parent2: Fitness=0.21253 (c) Offspring: Fitness=0.262065

Figure 3.2: An example of the offspring generated by SCX with the pair of parents chromosomes

3.3.5 Mutation

The mutation is another important operation of GA by which, new chromosomes can be generated by replacing some genes in original chromosomes. Consequently, it is helpful to improve the diversity of the population and search space. The mutation probability P_M would be set low in case the optimal search is broken. In this paper, we adopt the Gaussian mutation introduced in [168]. In detail, the i-th gene is mutated on the basis of an offset generated by the Gaussian distribution $N(\mu, \sigma^2)$ with the mean μ and the variance σ^2. Furthermore, the new gene has to be selected from the cluster of the replaced gene. The gaussian mutation is effective for GA to converge towards a better solution.

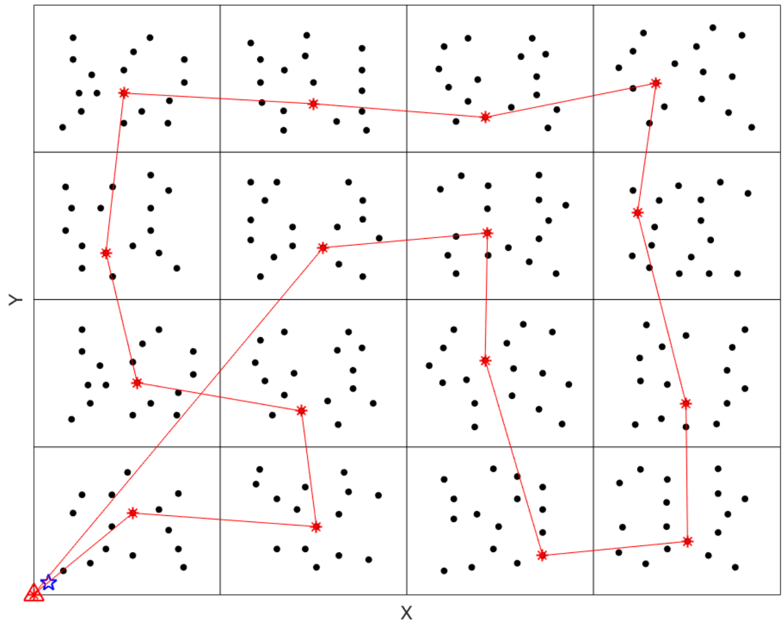


Figure 3.3: An example of the steady phase in the SDCA scheme

3.3.6 Steady Phase

By running GA in a predetermined number of generations, the chromosome with the highest fitness value is chosen as the final solution, which is the optimal path for the data collector. The data sink forwards a message which contains the information about the path and topology of each cluster to the data collector. Then, the data collector goes along the given route to initialize the deployed system. The data collector visits CHs and informs them about corresponding topology information. Once a CH receives the message from the data collector, it broadcasts this message to its cluster members. When cluster members receive the message from the CH, they start to transmit the sensed data to the CH. The CH aggregates the data and forwards the data to the data collector once the data collector enters its transmission range. From the second round, the data collector visits all CHs to gather data and forwards to the data sink at the end of each round. We provide an example of the steady phase in Fig. 3.3. The solid line demonstrates the path of the data collector.

3.4 Performance Evaluation

This section will present the simulation setup and the simulation results of the SDCA data collection scheme.

3.4.1 Experiment Setup

To evaluate the proposed protocol, we implemented a set of experiments on the OMNeT++ simulator. The FoI was divided into multiple grids with the identical size $250m \times 250m$, and we varied the size of the FoI by increasing the number of clusters M from 3×3 to 6×6. The speed and power rate of the data collector were set to $20m/s$ and $178W$ [169]. The value of E_{elec} and ϵ_{amp} were introduced in [66]. The parameters of Gaussian mutation were set according to [63], while the mean and the variance of the Gaussian number were changed based on the size of the FoI. To mainly evaluate the performance on energy efficiency of our proposed work, we set the value of σ to be 0.8. The simulation parameters are listed in Table 3.2.

Table 3.2: Simulation settings of SDCA Data Collection Scheme

Parameter	Description	Value
R_s^c	Communication range	$50m$ [170]
E_{elec}	Energy cost on the transceiver circuit	$50nJ/bit$ [66]
ϵ_{amp}	Energy cost on the transmit amplifier	$100pJ/bit/m^2$ [66]
v_{DC}	Speed of the mobile data collector	$20m/s$ [169]
P_{DC}	Power of the mobile data collector	$178W$ [169]
σ	The coefficient for energy consumption of sensor nodes in Eq. (3.8)	0.8
μ	The mean in the Gaussian distribution	0 [63]
σ^2	The variance in the Gaussian distribution	1 [63]
P_C	Crossover probability	0.8
P_M	Mutation probability	0.2
K	Packet size	4000 bits [136]

3.4.2 Simulation Results of SDCA Data Collection Scheme

In this section, we compared our work with three other different schemes with a single data collector, i.e., centre-based, greedy-based, clustering-based genetic algorithm (CBGA) [167], regarding *moving distance of the data collector* (meter), *data update period* (second), *system throughput* (Mbps) and *system-wide energy consumption* (nJ/bit). In the centre-based scheme, the node located closest to the centroid of each cluster is chosen to be the CH. In this scenario, the data collector visits all CHs according to an S-shaped route that is top-to-bottom and left-to-right. In addition, we developed a greedy-based algorithm as a heuristic algorithm for the data collector routing problem. The CBGA is implemented based on the one-hop data collection scheme [167]. We will analyze the experiment results in the following.

The first experiment compared the data update period of different schemes, which contains two main components, i.e., moving delay of the data collector, data forwarding latency. Due to the limited speed of the data collector, the moving delay is the dominated factor of the update period, which is mainly related to the moving distance of the data collector. This criterion is essential for data aggregation by using the data collector. The objective of deployed WSNs is monitoring the FoI or detecting the intruder. Therefore, the update period should be as short as possible in order to track the target. The results of the moving distance of the data collector and the data update period in different protocols are shown in Table 3.3 and Fig. 3.4, respectively. We can see that the SDCA data collection scheme has a much shorter update period than contrast approaches. The results for the first three schemes are similar. But CBGA has the longest update period among them because the number of waypoints is nearly six times more than other works in the same size as the FoI. It causes much longer moving distance in CBGA. Although the limitation of the operation time of the data collector is not considered in our experiment, if the operation time is too long, the data collector would run out of its energy before returning to the data sink. Consequently, the shorter update period of the system is an advantage for the SDCA scheme.

In the second experiment, we compared the system throughput. Fig. 3.5 shows the results of the system throughput with unit *Mbps*. It depicts that the system throughput increases as the number of sensor nodes increases in all schemes. This is because the amount of data generated by deployed sensors monotonically increases as the increasing size of the FoI.

The last experiment was conducted to compare the performance in terms of system-wide energy consumption, which measures the energy consumed by all sensors per bit. As shown in Fig. 3.6, the SDCA data collection scheme provided 1%-28.4% improvement

Table 3.3: The moving distance of the data collector in different size of the FoI

	The number of nodes			
SCHEMES	150	260	410	590
SDCA scheme	**2310** m	**3987** m	**6661** m	**8465** m
Centre-based scheme	3029 m	4775 m	7653 m	10120 m
Greedy-based scheme	3264 m	5548 m	9878 m	13716 m
CBGA	4673 m	8024 m	12461 m	22892 m

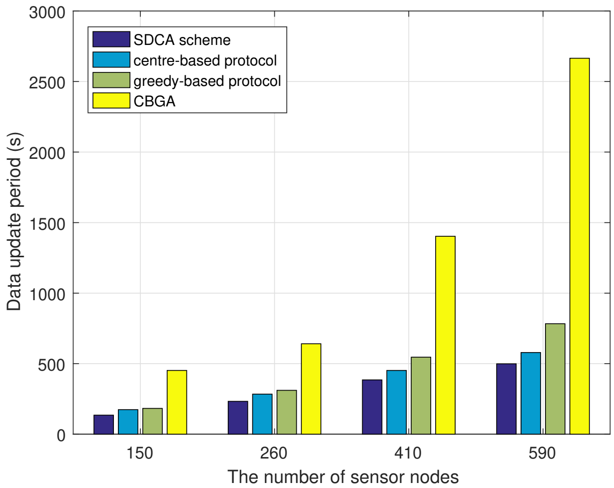


Figure 3.4: Comparison of data update period of the SDCA data collection scheme and the control group

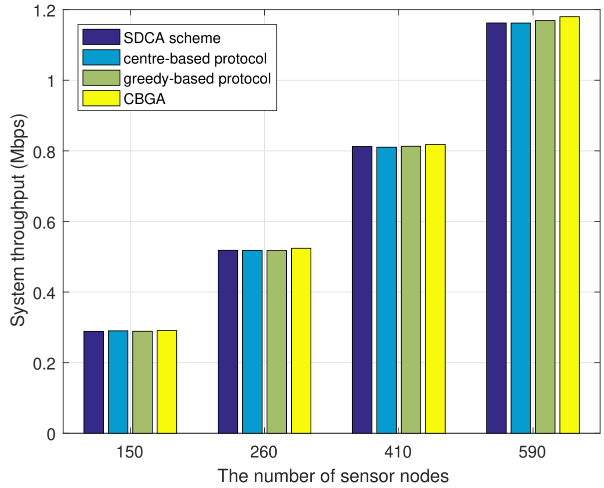


Figure 3.5: Comparison of the system throughput of the SDCA data collection scheme and the control group

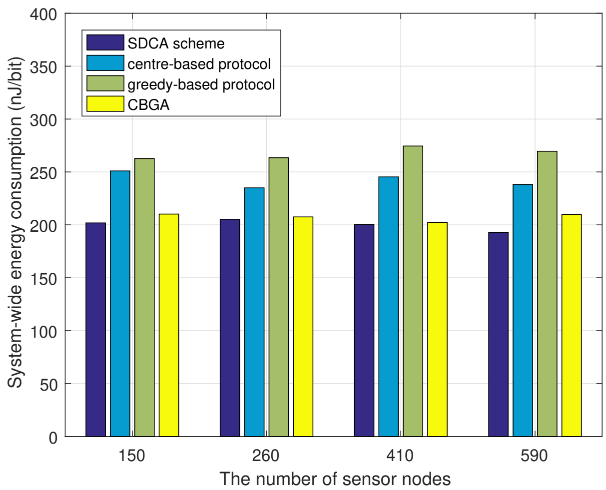


Figure 3.6: Comparison of system-wide energy consumption of the SDCA data collection scheme and the control group

compared with other three schemes. The reason is that the energy consumed by sensors in each cluster is taken into consideration as the objective function in our work. The sum of energy consumed by cluster members is less than centre-based protocol and greedy-based one. For CBGA, it is based on one-hop and without forwarding data by intermediate nodes, but the distance between the waypoint and its sensor nodes might be longer, which introduces extra energy consumption in each sensor. On the contrary, in our work, the transmission distance for each node was minimized. Hence, the energy consumption of each node is lower, and the system-wide energy consumption decreases by 1% to 8% comparing with CBGA.

3.5 Summary

In this chapter, we introduced the SDCA data collection scheme to improve the system-wide energy efficiency of small-scale WSNs. We applied a single data collector to gather the data from sensors. We formalized an optimization problem for minimizing the total energy consumption of the entire system. Due to the high computing complexity of this problem, a GA-based optimization approach has developed to derive the optimal solution. Our scheme has three phases. First, we constructed the topology of the network. Second, the data sink selected the CHs of each cluster and calculated the route for the data collector by executing GA. Third, the system entered the steady phase, and the data collector traversed the designated path and gathered data from each cluster. The simulation results showed that the proposed protocol could achieve a shorter data update period and much less energy consumption in WSNs.

Chapter 4

Improved Single Data Collector-assisted (SDCA) Data Collection Scheme

We introduced the SDCA data collection in the last chapter. The SDCA data collection scheme is helpful to enhance the energy efficiency of the entire system, however, the energy consumed by the intermediate nodes on forwarding is still considerable, which can be improved by reducing the size of clusters or the number of intermediate nodes. Therefore, relying on the energy model discussed in Chapter 3, we design a two-phase data gathering strategy with the mobile data collector in cluster-based WSNs, called improved SDCA data collection scheme, to enhance energy efficiency based on SDCA data collection scheme. The sensors are initially divided into clusters on the basis of the newly proposed clustering algorithm in the first phase. Then, we use GA to derive the shortest path of the mobile data collector. The path of the data collector would eventually be obtained and ensure lower data latency. We introduce the improved SDCA data collection scheme in the following. The content of this chapter has been published in *A novel data collector path optimization method for lifetime prolonging in wireless sensor networks.*

4.1 Cluster-based Topology Construction Algorithm

The first phase in the improved SDCA data collection scheme is cluster formation. To ensure low data latency, we adopt multi-hop communication between the sensors and the collector. The sensors are grouped into a set of clusters. The size of clusters is a crucial factor that affects the performance on energy efficiency and system delay. We propose a α-

hop clustering algorithm. The value of α is determined based on the system requirements. If the system only focuses on improving energy efficiency, the value of α would be small. It is helpful to decrease the energy consumption on data forwarding by relay nodes. On the other hand, if the application is delay sensitive, the number of clusters should be small in order to minimize the tour length. Therefore, the value of α should be relatively large. The CH in each cluster is selected based on the weight with respect to two parameters: the number of neighbors M_i and the average distance from neighbors D_i. The D_i represents the average distance of the node to its one-hop neighbors, which is defined as

$$D_i = \frac{\sum_{j=1}^{M_i} d(i, n_j)}{M_i}, \tag{4.1}$$

where n_j is the one-hop neighbor of the sensor, and $d(i, n_j)$ is the distance between the sensor to its neighbor n_j. We set a weight function for sensors based on the number of neighbors and the average one-hop distance. The CH will be chosen on the basis of the weight value from all sensors. The weight function is defined as follows,

$$Weight_i = M_i \times \frac{1}{D_i}. \tag{4.2}$$

Because of the different range of values for the parameters, i.e., M_i and D_i, we need to normalize the values between the range of 0 to 1. Therefore, the weight function is converted to

$$Weight_i = \frac{M_i}{M_{max}} \times \frac{D_{max}}{D_i}, \tag{4.3}$$

where M_{max} and D_{max} is the maximum of M_i and D_i among all sensors in the system.

The node information is known in advance, and the neighborhood information of each node is obtained by the method proposed in [171]. At first, all sensors are candidates of the CH, and their weights need to be calculated by Eq. (4.3). Then the sensor with the largest weight will be inserted into the list of CH. All nodes within its α hops could be found in the neighbor table and are removed from the candidate list. Next, the rest of the sensors in the candidate list will recalculate their weights. The iteration continues to choose other CHs until the candidate list is empty. Consequently, the list of CH will be obtained. The α-hop clustering algorithm is shown in Alg. 1.

4.2 Path Optimization Method for The Mobile Data Collector

After constructing the topology of the WSN, the path of the data collector which consists of all CHs will be generated. To obtain the optimal path with the low time complexity, we also

Algorithm 1 α-hop clustering algorithm

Require: Set of nodes N, neighbor table nb of N, the hop count in each cluster α

Ensure: Cluster head list CH, the set of sensor in cluster m $Cluster_m$

$Candidate \leftarrow N$
$CH \leftarrow \emptyset$
$m \leftarrow 0$
Calculate weight W_i for all sensors based on Eq. (4.3) in $Candidate$
Sort $Candidate$ in descending order according to W_i
while $Candidate$ is not empty **do**
    Obtain the first node in the $Candidate$ and add into CH
    Add the first node and its α-hop neighbors from the neighbor table nb into $Cluster_m$
    Remove all nodes in the $Cluster_m$ from $Candidate$
    $m \leftarrow m + 1$
end while

apply GA as the heuristic algorithm in the improved SDCA data collection scheme. The process of GA is similar to that in the SDCA data collection scheme, which is introduced in Section 3.3, and is modified according to the requirements in the improved SDCA data collection scheme. Here, we introduce the modified GA briefly. GA applies the chromosome to represent each solution which is the path for the data collector in this work. Each chromosome consists of a set of genes which are denoted by all positions that the collector needs to visit. Additionally, the chromosomes should not be duplicated. Since the goal of the proposed algorithm is to find the shortest path, the fitness function is reciprocal of the path length of each solution.

The data sink first derives the initial solutions and sorts them based on the fitness value in descending order. Then it starts the GA loop which includes three main operations of GA, i.e., selection, crossover, mutation. Here, we primarily introduce the modified SCX. Since the data collector starts from the data sink, the sink is initially added into the offspring and is set to be the current node. The parent chromosomes figure out the candidates for the next node of the offspring. The candidates in both chromosomes are the next node of the current node. However, if the candidate already exists in the offspring, the node from the unvisited cluster with the smallest sequent number will replace it. One of the candidates with a small distance from the current node is selected as the next node of the offspring. SCX will terminate once the offspring is constructed. Then the fitness value of the offspring is calculated. The chromosome with worse fitness value would be removed among two parents and their offspring. Fig. 4.1 depicts the complete process flow of the proposed path planning algorithm.

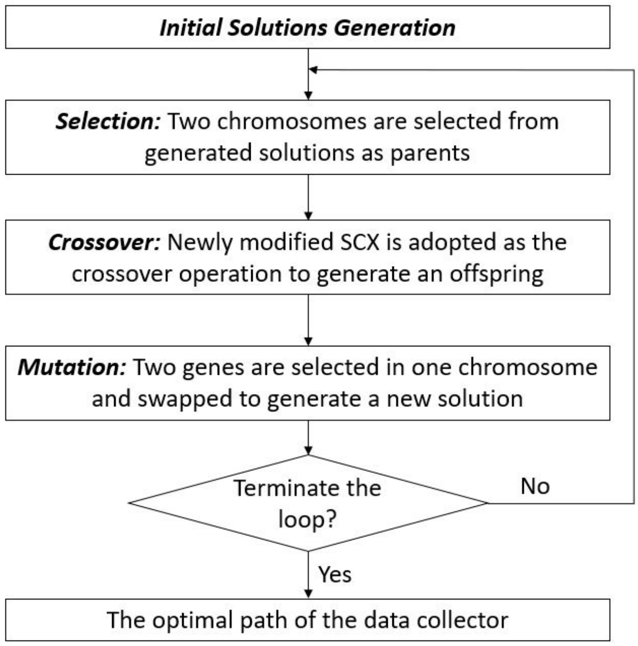


Figure 4.1: The path planning algorithm for the single device workflow

The optimal path of the data collector is finally obtained by GA. The collector receives the path and cluster information from the sink and then begins to collect data. Once the collector arrives at the CH, it will notify the CH. Then the CH broadcasts its cluster members to gather data from them and forwards the data to the collector. Once the collector visits all CHs, it will return to the sink and transmit all collected data. Finally, it recharges its batteries and starts the next tour. We present an example to illustrate the process of data collection in Fig. 4.2.

4.3 Performance Evaluation

The experiment setup and results of the improved SDCA data collection scheme are shown in this section. The improved SDCA data collection scheme has been ported to the OMNeT++ simulator. The simulation parameters are similar to the settings in the experiments of the SDCA data collection scheme, which are listed in Table 3.2.

We first demonstrate the effect of the value of α on the proposed clustering algorithm by the simulation experiments. Fig. 4.3 shows the changing trend of the data update period versus the different value of α. Recall that; the update period is mainly affected by the journey length of the data collector. Additionally, the tour length will be elongated

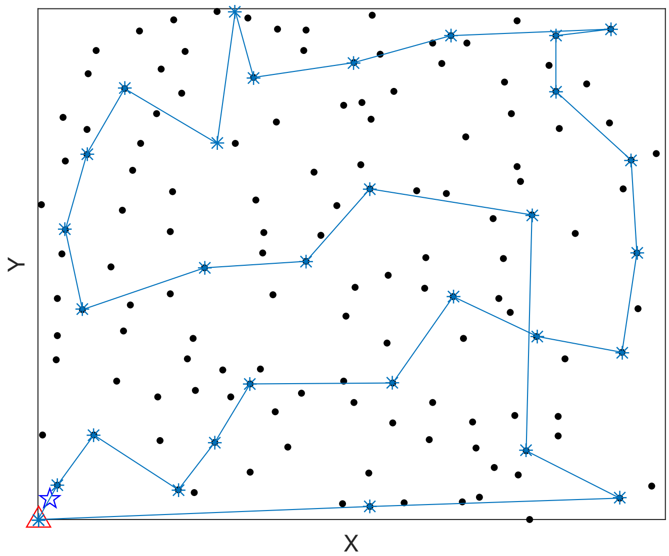


Figure 4.2: The path planning for the mobile data collector in the improved SDCA scheme

as the number of clusters in the FoI increases. In Fig. 4.3, we can see that the update period becomes shorter as the value of α increases. The reason for this result is that the size of clusters will be large when the α is large, and then the number of clusters will be decreased. Therefore, the tour length of the data collector would be shortened when the value of α is set to be large.

On the other hand, the system-wide energy consumption is also be affected by the value of the α. As shown in Fig. 4.4, the energy consumption shows a growing trend with the increasing of the value of α. Because of the big size of clusters introduced by a large value of α, more data should be transmitted to the CH by relaying nodes. Therefore, the energy consumed by sensors as relaying nodes on data transmission increases, which renders relatively higher system-wide energy consumption as the α increases. Thus, the system should balance the number and the size of the clusters and set the value of α based on the system requirements. For a fair comparison, the value of α is set to be 2 in the following experiments.

Next, we conducted a set of experiments to compare the improved SDCA data collection scheme with the other three schemes using mobile elements, i.e., the SDCA data collection scheme, CBGA, and reduced k-means (RkM) [112]. The primary difference among these four schemes is the clustering strategy which affects the system performance. In the improved SDCA data collection scheme, the sensors are divided into clusters based on the

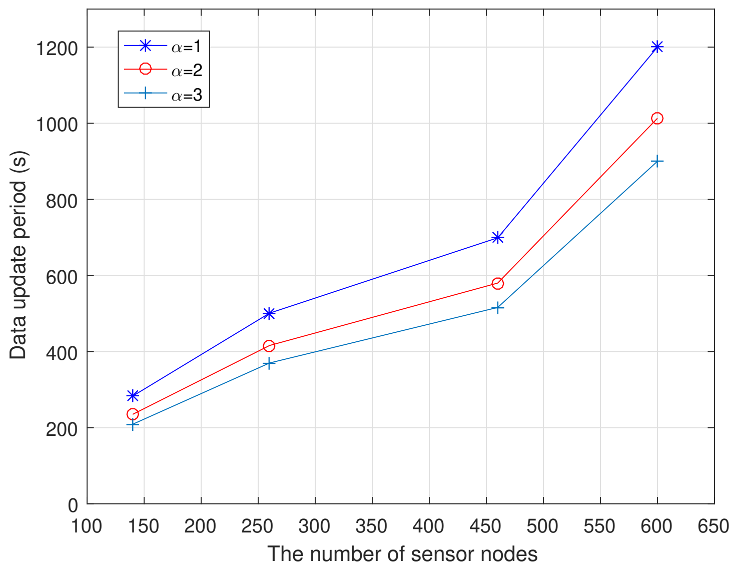


Figure 4.3: Comparison of data update period under different values of α

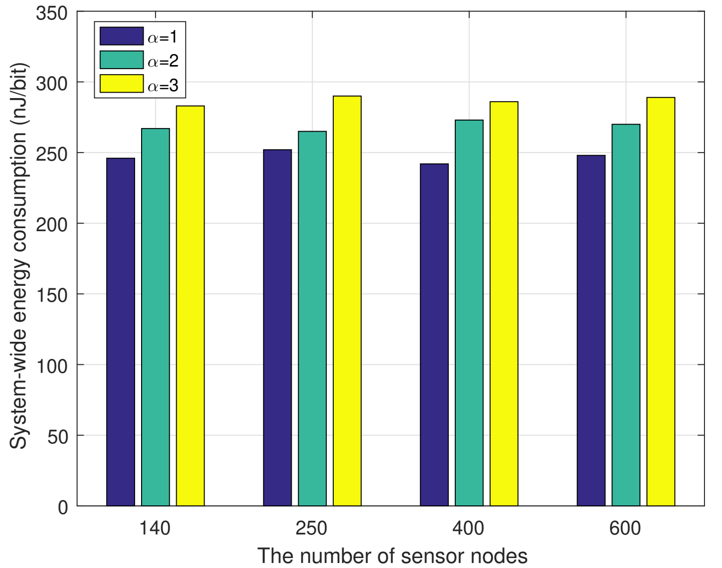


Figure 4.4: Comparison of wide-system energy consumption under different values of α

α-hop clustering algorithm. While, in the SDCA data collection scheme, the FoI is initially divided into several grids with identical size, and the sensors in one grid are grouped into a cluster. CBGA is a data collection scheme with the one-hop communication strategy which means the sensors can communicate with the data collector directly. The sensors, whose transmission disks are overlapped, constitute a cluster. In RkM, the single mobile sink is applied to gather data from sensors. This scheme applied k-means clustering algorithm to select a minimal number of CHs and divide the sensors into the clusters according to the distance to the CHs. Then, the authors utilized Christofides's heuristic algorithm [172] to plan the trajectory of the mobile sink. Here, we compare the performance of improved SDCA data collection scheme with the schemes in the control group regarding *data update period* (second), *system throughput* (Mbps) and the *system-wide energy consumption* (nJ/bit).

Fig. 4.5 compares the *data update period* of these four algorithms. The results are shown in Fig. 4.5 and prove our proposed two SDCA data collection algorithms achieve a shorter data update period because of the reasonable clustering algorithm. This criterion is important for delay-sensitive applications in WSNs given that the urgent data can be transmitted to the data sink within the tolerant period.

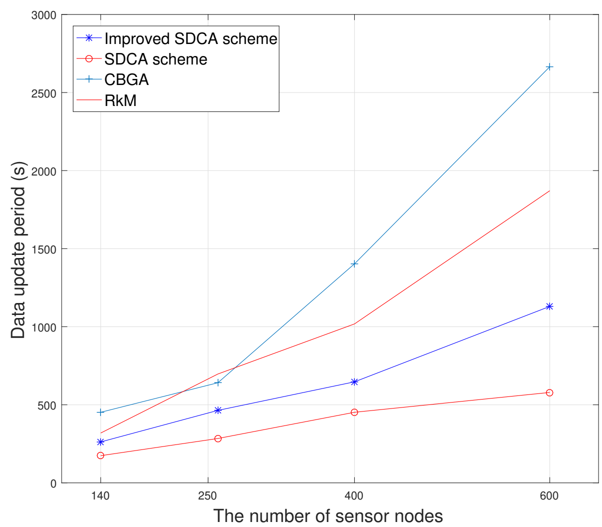


Figure 4.5: Comparison of data update period of the improved SDCA data collection scheme and the control group

In the second experiment, we compare the *system throughput* from the different algorithms. It evaluates the sum of the data rates which are sent to the data sink in WSNs. Fig. 4.6 presents the results of these four data collection schemes. The difference among the

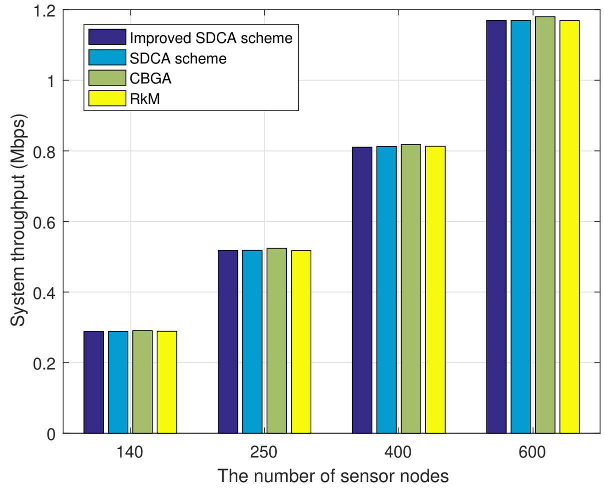


Figure 4.6: Comparison of the system throughput of the improved SDCA data collection scheme and the control group

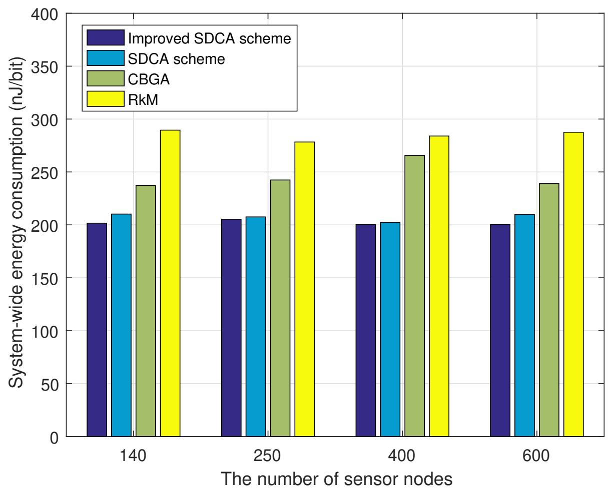


Figure 4.7: Comparison of system-wide energy consumption of the improved SDCA data collection scheme and the control group

four schemes is not substantial. The reason is the system throughput is mainly associated with the size of the FoI and is improved by increasing the number of deployed sensors in the WSN.

The third experiment is conducted to evaluate the algorithms in terms of the *system-wide energy consumption rate*. Recall that the major source of energy consumption by sensors is transceiving. Therefore, this metric represents the energy required for the node to send each bit. The results of energy consumption rates for the four algorithms are shown in Fig. 4.7. The improved SDCA data collection scheme achieves higher energy efficiency than other algorithms since we take the distance of transceiving and the hop count in the clusters into consideration. The energy consumption on sending or forwarding data is minimized. In RkM and CBGA, the communication strategy between the sensor and collector is one-hop. Though the sensor does not need to forward data from others, the distance between the collector and the sensor might be large. It may lead to more energy consumption on transmission than the schemes with multi-hop communication. Additionally, the size of clusters generated in SDCA data collection scheme is big, which means that the average number of hops to CH for each sensor is large, and causes more energy consumed on data forwarding by intermediate nodes. Therefore, the improved SDCA data collection scheme can effectively reduce the system-wide energy consumption.

4.4 Summary

To enhance the energy efficiency of WSNs, we proposed an improved SDCA data collection scheme for balancing the energy efficiency and the delay of data based on the proposed scheme introduced in Chapter 3. The contributions are twofold: we developed a cluster formation algorithm to generate the cluster with the appropriate size and select the CH in each cluster. Moreover, the path of the data collector was obtained by the modified GA with lower complexity. After deriving the optimal path, the data collector started to go through the optimal path and collect the data from sensors. The results from experiments demonstrated that our protocol improved energy efficiency and achieved lower data latency compared to other existing protocols using a single data collector.

Chapter 5

Two-mode Multiple Data Collector-assisted (MDCA) Data Collection Scheme

In large-scale WSNs, the SDCA data collection schemes are not effective because the path length for the single data collector is too long. To shorten the path of the data collector, an alternative approach is to decrease the number of clusters, which could be achieved by increasing the value of α in the α-hop clustering algorithm. However, this approach leads to huge energy consumption on data forwarding and high workloads for CHs. In this chapter, we propose a two-mode multiple data collector-assisted (MDCA) data collection scheme in cluster-based WSNs by adopting multiple data collectors in the large-scale FoI. The content of this chapter has been published in *Unmanned aerial vehicle-assisted energy-efficient data collection scheme for sustainable wireless sensor networks.*

5.1 Problem Statement

In the MDCA data gathering scheme, the entire network is partitioned into a set of subnetworks. Each collector is responsible for collecting data from sensors in a designated subnetwork by traversing a planned path, called subtour. There are two data collection modes for MDCA data collection scheme. One is that data collectors transmit the collected data to the sink directly, called gathering-and-carrying mode. In detail, the starting point of all subtours is the data sink. The collector visits a set of CHs to collect data from their clusters. Then it goes back and forwards the aggregated data to the data sink directly. This mode is suitable for the delay-sensitive applications because the urgent data can arrive at the sink in time. However, the collectors that need to collect data from more distant clusters have to suffer from the longer flight time and more energy consumption for returning to the data sink. Therefore, for these collectors, the path length for each data collector should be constrainted to avoid extremely long subtours.

The other mode for multiple collectors is that data collectors transmit the gathered data to the sink relying on other intermediate collectors, called data-relaying mode. In this mode, the collectors gathering data in the remote area will forward data to their neighboring collector once the distance between them is less than their communication range. It is helpful to reduce much energy on moving between the area and the sink. The drawback of this mode is that two neighboring collectors may meet each other after a long period due to their different motion characteristics, e.g., speed, trajectory, etc. Hence, the data from further clusters cannot arrive at the sink under the time constraints. There is another reason causing long data latency. When a mobile collector receives the gathered data from its neighboring collector, it has to finish its data collection task first, and then forwards the data that it collects and receives from other collectors. As such, the data from the remote area have to suffer from the long forwarding latency before being received and processed by the data sink. Although it is not suitable for the delay-sensitive applications, it can save the energy of the data collectors on moving. Accordingly, the applications need to determine the data collection mode on the basis of their requirements. Besides, the length of subtour should be constrained for avoiding the buffer overflows in sensors or exceeding the maximum operation time of the data collector. The reason for the buffer overflow is the limited data buffer of sensors. Once the buffer is full, the sensed data cannot be saved into the buffer and have to be dropped. Therefore, the tour length of the collector should be predetermined to avoid the buffer overflow problem, which is defined as the limited maximum moving length L_{limit} in our work. In the MDCA data collection scheme, we propose two modes of schemes to meet different system requirements. These two schemes are described in the following.

5.2 Delay-Aware MDCA Data collection Scheme

We first introduce the MDCA data collection scheme with the gathering-and-carrying mode in cluster-based WSNs for delay-sensitive applications. The network is initially divided into a set of clusters by the α-hop clustering algorithm introduced in Section 4.1. Then, the topology of the WSN is constructed. The limited length of the path $L^{(i)}_{limit}$ in this scheme is considered to be related to the buffer size Buf_s, the data generated rate of sensors $DataRate_s$ and the moving speed of mobile collectors v_{DC}. The upper bound of journey time in each round is set based on the buffer size of the sensor to avoid the potential buffer overflow. Therefore, the $L_{limit}^{(i)}$ of colletor i is given by,

$$L_{limit}^{(i)} = \frac{Buf_s}{DataRate_s} \times v_{DC}, \tag{5.1}$$

where the subscript s denotes the ID of the sensor.

To further decrease the data forwarding latency, a heuristic algorithm is a pragmatic method to solve the problem under consideration, since it can obtain the optimal solution with low computing complexity [132]. Accordingly, we develop a delay-aware heuristic MDCA (DA-MDCA) data collection scheme. The subtours for the minimum number of UAVs are derived by the proposed algorithm, which ensures that the lengths of subtours are below the upper bound and as average as possible. Initially, all CHs are set to be unvisited. The start point for each subtour should be the data sink. Thus, the data sink is added into the subtour and is set to be the *Current Node*. The data sink should calculate $L_{limit}^{(i)}$ based on Eq. (5.1) for each collector. Then, it tests the node which is closest to the *Current Node*. If the length of subtour by adding this neighboring node is not over $L_{limit}^{(i)}$, the node will be added into the subtour and will be set as the new *Current Node*. Then, the data sink looks for the next node for the subtour. The searching procedure keeps continuously until the accumulated path length exceeds $L_{limit}^{(i)}$. Then the subtour is obtained and the data sink will start to find the next subtour. The algorithm will terminate until all CHs have been visited. The details of the proposed DA-MDCA data collection scheme are shown in Alg. 2.

5.3 Delay-Tolerent MDCA Data Collection Scheme

The DA-MDCA data collection scheme we proposed above is more suitable for delay-aware applications. However, the energy consumption of the data collector on movement is relatively high because the data collector needs to commute frequently to and from the data sink and the corresponding subnetwork. To improve the system efficiency for the delay-tolerant applications, our MDCA scheme can switch its working status to the newly designed data-relaying mode, called delay-tolerant MDCA (DT-MDCA) data collection scheme. Some related works have been proposed by adopting the data-relaying working mode, such as [173]. However, these existing approaches lead to a long time for data delivery, especially from the remote subnetworks, because the data collector needs to complete its own data collection task before sending the collected data (including data from other data collectors) to the next relay collector. To address this problem, a new data-relaying method for the MDCA scheme is designed. Different from the path planning

Algorithm 2 DA-MDCA Data Collection Scheme

Require: Set of nodes CH and their positions

Ensure: The path for each data collector P_i

```
  Unvisited ← CH
  current ← sink
  i ← 0
  while Unvisited is not empty do
    L ← 0
    P_i ← current
    isfinished ← ture
    while isfinished == true do
      Calculate the L_limit^(i) in this round by Eq. (5.1)
      Sort Unvisited in ascending order according to the distance to current
      candidate ← the first node in Unvisited
      if L+ distance from current to candidate+ distance from candidate to the sink
      ≤ L_limit^(i) then
        L ← L+ distance from current to candidate
        P_i ← candidate
        current ← candidate
        Remove current from Unvisited
      else
        i ← i + +
        current ← sink
        isfinished ← false
      end if
    end while
  end while
```

in the DA-MDCA scheme, the length of subtours is defined based on the distance from the sink to clusters needed to be visited. In our design, we adopt a similar idea of the unequal clustering algorithm introduced in [110] to balance the workloads between data collectors. More precisely, the data collector closer to the data center will be responsible for smaller subnetworks (i.e., the subtour length is shorter) because it also needs to do data relay for the distant data collectors. Accordingly, the limited maximum moving length for each collector $L^{(i)}_{limit}$ is given by

$$L^{(i)}_{limit} = \left(1 - \gamma \times \frac{d_{max} - d_{(s_i,sink)}}{d_{max} - d_{min}}\right) \times \frac{Buf_s \times v_{DC}}{DataRate_s}, \tag{5.2}$$

where d_{min} and d_{max} are the distance from the sink to the closest node and the farthest node, respectively. The $d_{(s_i,sink)}$ is the distance between the start point of subtour i and the sink. The parameter, $\gamma \in [0, 1]$, is a user-defined control parameter. By choosing a different value of γ, the user can adjust the limited maximum moving length according to the system requirements (e.g., the number of mobile data collectors deployed in the WSN).

In this scheme, the sensors are grouped into clusters by our proposed clustering algorithm introduced in Section 4.1 and all CHs are selected. Then, the data sink begins to derive the subtours for each collector. Unlike the DA-MDCA scheme that the subtour construction procedure starts from the data sink, here, the farthest node from the data sink is selected as the *Source Node* of current subtour. The purpose of selecting the farthest node as the *Source Node* first is to achieve a large amount of clusters on the tour of the collectors which are distant to the data sink and a small amount of clusters on the tour of the collectors which are closer to the data sink. Then, the *Source Node* calculates $L^{(i)}_{limit}$ based on Eq. (5.2) and set itself as the *Current Node*. The node which is closest to the *Current Node* is selected. If the length of the subtour does not exceed $L^{(i)}_{limit}$ after adding this node into the subtour, the node will be added into the current subtour and is converted to *Current Node*. After that, the next node in the current node will be derived. The subtour is derived until no node can be added into the subtour due to the tour length limitation. Similar to the previously mentioned DA-MDCA scheme, the subtour construction procedure terminates when all CHs have been visited. After constructing the subtours for collectors, the collectors begin to gather data from their corresponding subnetworks. The collector working in the distant area will transmit its collected data to one of its neighboring collectors which is closer to the sink when the neighboring collector enters its communication disk. We show the DT-MDCA scheme in Alg. 3.

Algorithm 3 DT-MDCA Data Collection Scheme

Require: Set of nodes CH and their positions

Ensure: The path for each data collector P_i

```
Unvisited ← CH
i ← 0
while Unvisited is not empty do
  L ← 0
  Sort Unvisited in descending order according to the distance to the sink
  source ← the first node in Unvisited
  current ← source
  P_i ← current
  Remove current from Unvisited
  Calculate the L_limit^(i) in this round by Eq. (5.2)
  isfinished ← ture
  while isfinished == true do
    Sort Unvisited in ascending order according to the distance to current
    candidate ← the first node in Unvisited
    if L+ distance from current to candidate distance from candidate to source ≤ L_limit^(i)
    then
      L ← L+ distance from current to candidate
      P_i ← candidate
      current ← candidate
      Remove current from Unvisited
    else
      i ← i++
      isfinished ← false
    end if
  end while
end while
```

5.4 Performance Evaluation

This section will present the experiment setup and simulation results of the two-mode MDCA data collection schemes and the control group. To evaluate the performance of MDCA data collection schemes, we also adopted OMNet++ simulator to conduct the simulation experiments. Except for simulation parameters are shown in Table 3.2, other settings are summarized in Table 5.1.

Table 5.1: Simulation settings of MDCA data collection schemes

Parameter	Description	Value
Buf_s	Buffer size of the sensor	64KB/512KB [174]
$IniE_s$	Initial energy of sensor	15J [15]
$DataRate_s$	Data generation rate	2000 bits/s [175]

We evaluate the newly proposed two-mode MDCA data collection scheme and adopt the tour planning algorithm with the spanning covering tree algorithm (SCTA) proposed in [173] as the control group in the simulations. Recall that; the tour planning algorithm with SCTA runs in the data-relaying mode, and it utilizes the clustering algorithm which is similar to CBGA. A virtual point, called the polling point in [173], is initially set in each cluster, and the data collectors will gather data from the sensors in each cluster at the polling point. Then, the minimum spanning tree composed of all polling points is constructed by the SCTA. The optimal subtours of multiple data collectors are derived from the spanning tree by limiting their tour length. The main differences of our proposed MDCA schemes and the tour planning with SCTA are the workloads of data collectors and the path planning method for collectors. We compare these MDCA data collection schemes regarding *system-wide energy efficiency* (nJ/bit), *network lifetime* (second), *energy consumption of the data collector* (kJ/round) and *packet delay* (second).

Fig. 5.1 illustrates the comparison of the *system-wide energy consumption* from these three MDCA data collection schemes. The results from the experiments depict that our proposed MDCA schemes achieve higher system-wide energy efficiency. Moreover, Fig. 5.2 shows the comparison results of the *network lifetime* from these three schemes. The results present the network lifetime in our proposed MDCA schemes is much longer than that of the tour planning algorithm with SCTA (approximately 25%). The achievement is mainly because of the adoption of the newly proposed clustering algorithm based on the multi-hop communication strategy. It effectively reduces the transmission distance from a sensor to another sensor (or to the collector) by selecting the optimal CH. Additionally, it minimizes energy consumption for the sensors on data forwarding by limiting the hop counts in each cluster. On the other hand, with the one-hop communication strategy in the tour planning with SCTA, the transmission distance between the sensor and the polling point is likely to be long, which causes significant energy consumption by the sensors on transmission. Therefore, the newly proposed MDCA data collection schemes significantly improve the system-wide energy efficiency in large-scale WSNs.

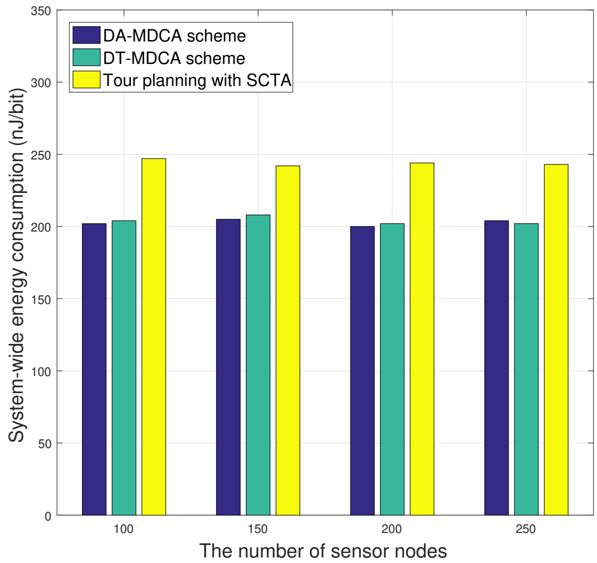


Figure 5.1: Comparison of system-wide energy consumption of MDCA data collection schemes and the control group

In the next experiment, we compare the *energy consumption of the data collector*, which is denoted as the total energy consumption by all collectors finishing the collection task in each round. The number of data collectors deployed in the WSN with the different size is listed in Table 5.2, and the results of the energy consumption by all sensors are shown in Fig. 5.3. The results demonstrate that both the DT-MDCA scheme and the tour planning with the SCTA have better performance than the DA-MDCA scheme in terms of the number of data collectors used and the energy efficiency for the data collectors. In general, the energy is mainly consumed by data collectors on data transceiving and movement. Compared to the energy consumed by data collectors on data transceiving, energy consumed by movement is tremendously dominant in these data collection schemes. Due to the different modes adopted in the schemes, the proportions of the number of the deployed collectors and energy consumption by collectors in different schemes vary. Given that the DA-MDCA scheme runs the gathering-and-carrying mode, the collectors can transmit their gathered data to the data sink directly, which reduces energy consumption

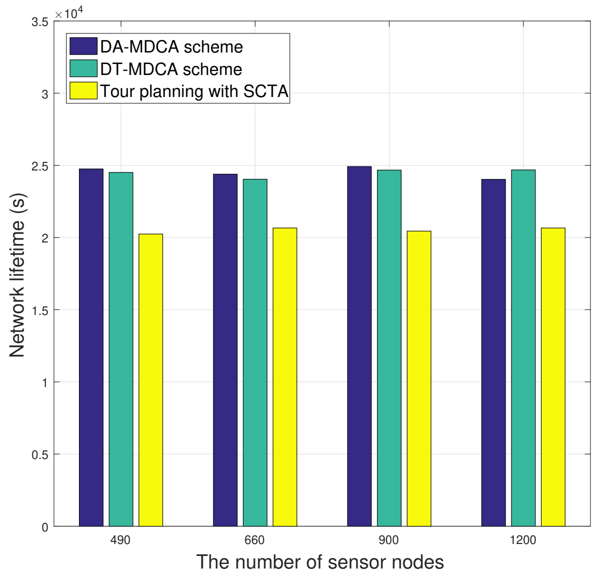


Figure 5.2: Comparison of network lifetime of MDCA data collection schemes and the control group

on data transmission. However, more collectors should be deployed in the FoI to achieve the data collection tasks in delay-sensitive applications due to the relatively short tour length for each collector with the predetermined L_{limit}. Therefore, the collectors in this mode need to consume more energy on movement. On the other hand, the DT-MDCA scheme and the tour planning with SCTA run in data-relaying mode. They consume more energy on data transceiving with the sensors and relaying collectors than the schemes with the gathering-and-carrying mode, however, the number of data collectors is less and they consume less energy on movement. Therefore, the total energy consumption by collectors in these two schemes is less than the energy consumed by collectors in the DA-MDCA scheme.

Table 5.2: The number of data collectors deployed by different MDCA data collection schemes

	The number of collectors in each scheme		
Number of deployed sensors	DA-MDCA	DT-MDCA	Tour planning with SCTA
490	11	6	8
660	15	10	12
900	22	15	16
1200	29	22	23

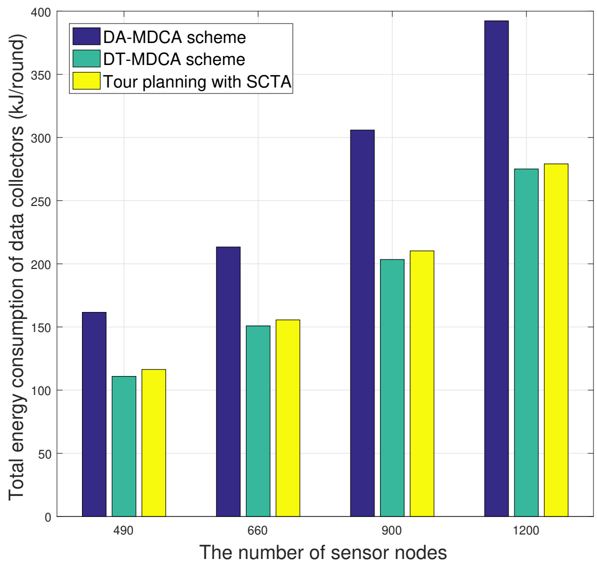


Figure 5.3: Comparison of the total energy consumption of data collectors of MDCA data collection schemes and the control group

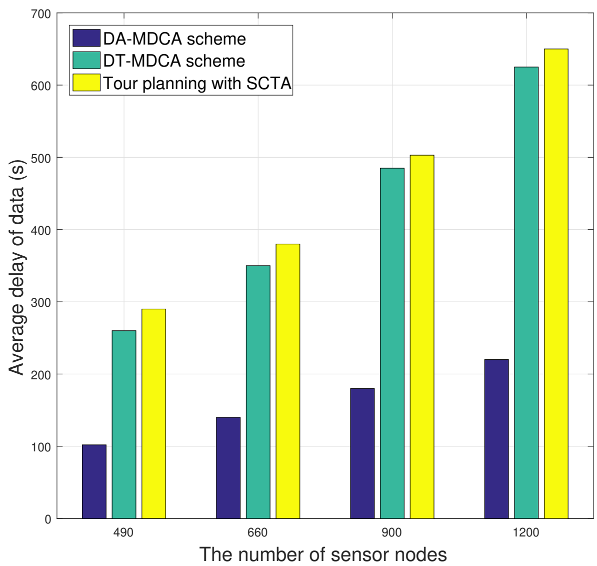


Figure 5.4: Comparison of the average delay of MDCA data collection schemes and the control group

Fig. 5.4 is given to show the *average delay* in different scale WSNs with the different schemes. The delay is defined as the time period from the moment a sensor sends the data to the moment the data sink receives the data. The delay of data enlarges by increasing the number of deployed sensors in the WSNs, which is triggered by the increased average distance from the sensor to the data sink and more generated data messages in the WSNs. In the tour planning with SCTA, the data collector that gathers the data from the distant subnetwork can only transmit its aggregated data to the data sink by its neighboring collector. However, if the tour length of its neighboring collector is long, the delay will be extended. Compared with tour planning with SCTA, the improvement of our proposed DT-MDCA scheme is that the tour length for each collector is planned according to the distance from the clusters needed to be visited the data sink. It can decrease the tour length for the intermediate collectors. Thus, the average delay of the DT-MDCA scheme is shorter than that of the tour planning with SCTA. However, in the data-relaying mode, the data collector might wait for a long time period to communicate with the neighboring collector until its neighboring collector enters its communication range. It will cause a long data latency. In the DA-MDCA scheme, each data collector will return to the data sink after finishing its collection task, which is helpful to reduce the data latency, especially for the data from the distant subnetworks. Because the collectors gathering data from the remote area may only visit a small number of clusters under the length limitation, the data can be transmitted to the data sink within the specified period. Therefore, the average delay of data in the DA-MDCA scheme is reduced significantly compared with other schemes.

5.5 Summary

In order to improve the energy efficiency in large-scale WSNs, we developed a two-mode of data gathering mechanism by adopting multiple collectors, which could switch its status between the delay-aware mode and the delay-tolerant mode. The DA-MDCA data collection scheme ran in the gathering-and-carrying mode in which all collectors transmitted their gathered data to the data sink directly. It was helpful to shorten the data latency. On the other hand, the DT-MDCA data collection scheme ran in the data-relaying mode in which the collectors gathering data from the distant subnetworks transmitted data to the data sink by intermediate collectors. It was beneficial to enhance the energy efficiency of the collectors. Therefore, our proposed two-mode MDCA data collection scheme could satisfy the requirements in different systems. The simulation results demonstrated that the MDCA data collection schemes improved the system-wide energy efficiency and enhanced the system performance in different aspects. The DA-MDCA gathering scheme could shorten the data latency. On the other hand, the DT-MDCA collection scheme was helpful to reduce the budget and energy consumption for multiple data collectors.

Chapter 6

Joint Data Collection and Wireless Charging Scheme

Data gathering using the mobile data collector can save more energy consumed by the sensors on data transmission and address the energy-hole problem of WSNs. However, the network lifetime is still limited. As stated before, sensors are able to receive the energy by the wireless charging, which is effective to prolong the system lifetime, even make the WSN semi-permanent. In this chapter, we discuss the proposed joint data collection and wireless charging schemes in detail. Here, we assume that mobile chargers (MCs) as the high-mobility device to collect data from CHs. Moreover, MCs would replenish energy to sensors in each cluster simultaneously. In this chapter, we first propose an improved clustering algorithm. Then, the scheduling schemes of MCs are proposed. In what follows, the joint data collection and wireless charging scheme will be introduced in detail. The content of this chapter has been published in *A novel two-mode QoS-aware mobile charger scheduling method for achieving sustainable wireless sensor networks.*

6.1 Syetem models and assumptions

In the joint design, we follow the same assumptions of network model and the energy model on data transmission of the sensor and movement of the mobile device introduced in Section 3.1. Moreover, the energy model in terms of wireless charging should also be taken into account in the joint data collection and wireless charging scheme. In our works, we use the charging model introduced in [176]. Let P_t^{MC} denote the charging power of the MC. Then, the energy receiving rate of a sensor P_r^s in the charging range R_c^{MC} can be denoted by

$$P_r^s = \frac{G_s G_r \eta}{L_p} (\frac{\lambda}{4\pi(d+\beta)})^2 P_t^{MC}, \tag{6.1}$$

where G_s and G_r are the antenna gain of transmitter and receiver, respectively. η represents the rectifier efficiency, L_p is the polarity loss, λ is the amplitude, and β is the coefficient for adjusting the Friis' free space equation. As shown in Eq. (6.1), P_r^s is mainly related to the distance between the sensor and the MC. To simplify the description, Eq. (6.1) can be modified to

$$P_r^s = \frac{\alpha}{(d+\beta)^2} P_t^{MC}, \tag{6.2}$$

where $\alpha = \frac{G_s G_r \eta \lambda^2}{L_p (4\pi)^2}$. Therefore, the energy E_H^s received by the sensor s_i from the MC within a time period t is

$$E_H^s = P_r^s \times t = \frac{\alpha}{(d_{(s_i,MC)}+\beta)^2} P_t^{MC} \times t. \tag{6.3}$$

6.2 The Improved Clustering Algorithm

In this section, we introduce an improved clustering algorithm based on the α-hop clustering algorithm proposed in 4.1. Except for the number of neighbors M_i and the average distance from neighbors D_i introduced in the α-hop clustering algorithm, the residual energy of the sensor E_{Res}^i is taken into consideration in the improved clustering algorithm. The sensor with high residual energy has a high probability to be the CH. Therefore, the weight function is modified as follows,

$$Weight = \lambda \frac{M_i}{M_{max}} + (1-\lambda) \left[\mu \frac{E_{Res}^i}{E_{cap}^i} + (1-\mu) \frac{D_{max}}{D_i} \right], \tag{6.4}$$

where λ, μ are two coefficients between 0 to 1 to adjust the weight of the three parameters in the weight function. E_{cap}^i denotes the energy capacity of the sensor.

After sorting all sensors according to their weights, the cluster construction would be executed. The process of the cluster construction in the improved clustering algorithm is similar to the α-hop clustering algorithm. The main difference from the α-hop clustering algorithm is the size of clusters. In the α-hop clustering algorithm, the size of clusters is determined by the value of α. However, in the joint design, considering that the MC needs to replenish energy to all sensors in each cluster, the size of each cluster should be no bigger than the charging range of the MC while decreasing the number of clusters. Initially, all sensors are candidates of the CH. The sensor with the highest weight is selected as the CH currently. Then, the sensors whose distance to the current CH is less than the charging range need to attach to the CH. Next, the sensors in the formed cluster are removed from the table of CH candidates, and the current iteration ends. In the next iteration, the rest of the candidates will calculate their weight again and repeat the process of cluster formation. Once the table of CH candidates is empty, all clusters are formed in the WSN. An illustration that the clusters are generated by the improved clustering algorithm is shown in Fig. 6.1.

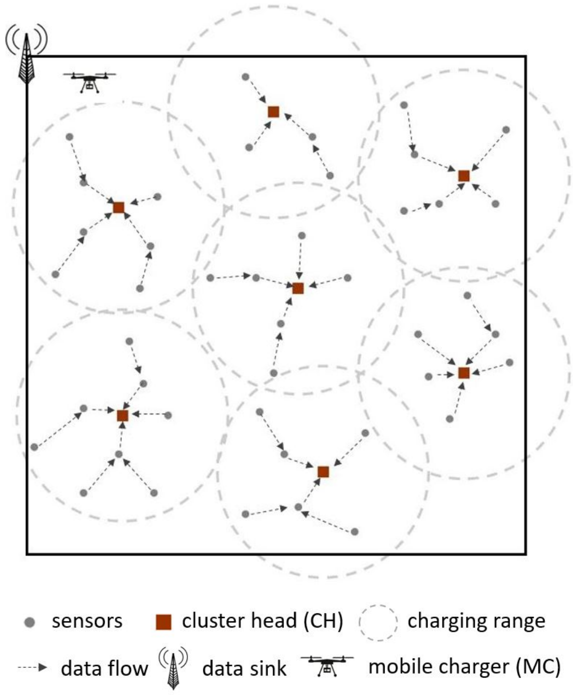


Figure 6.1: Example of the clusters generated by the improved clustering algorithm

6.3 Problem Statement

The scheduling scheme of MCs focuses on addressing two subproblems: the path planning and the sojourn period calculation in the cluster. To face the diverse system requirements, we design the scheduling scheme for two scenarios, i.e., delay-tolerant scenario and delay-aware scenario. Additionally, the cost of the MC is also considered in our work, in other words, the number of MCs deployed in the WSN is as few as possible. In this section, we will present the charging problems in these two scenarios.

6.3.1 Problem Statement for the Delay-tolerant Scenario

In the delay-tolerant scenario, we assume that the MC stays the location of the CH in each cluster until the energy of the CH and its cluster members is fully charged to make the WSN semi-permanent. For the delay-tolerant scenario, the charging utility which measures the quality of sensor charging plays a vital role [177, 178, 179, 180]. Accordingly, the objective of the delay-tolerant charging scheme is maximizing the charging utility of sensors while minimizing the energy consumption by the MC on movement. In this work, we adopt the charging utility metric introduced in [180] to evaluate derived potential paths. The charging utility for the path P of the MC is the sum of the energy provided to the sensors s_i in the cluster k_i in the path P, which can be formulated by

$$U^P = \sum_{k_i \in P} \sum_{s_i \in k_i} E_H^{s_i}. \tag{6.5}$$

Therefore, we define the delay-tolerant charging problem of the MC as follows:

Problem 1 *Given a set of sensor nodes $S = \{s_1, s_2, ..., s_N\}$, and a set of CHs ($CH \subseteq S$) for K clusters which are sojourn locations of MCs in each cluster, we assume that there are two MCs, that work corporately, in the deployed WSN, and they traverse the given path in the opposite directions. The delay-tolerant charging scheduling problem is constructing a path p including all CHs for two MCs such that the charging utility of all sensors is maximized while the length of the path is minimized. The objective function and constraints of the problem are formulated as follows:*

$$max \quad U^P - E_{move}^{MC}, \tag{6.6}$$

$$s.t. \quad \tau_{k+1} \geq \tau_k + t_c^k + \tfrac{D_{(k,k+1)}}{v^{MC}}, \quad \forall k \in P \tag{6.7}$$

where τ_k is the arrival time of the MC at the cluster k, and t_c^k is the time period of charging in the cluster k. Constraint (6.7) ensures that each sensor node should be charged before depleting its battery.

6.3.2 Problem Statement for the Delay-aware Scenario

Different from the delay-tolerant scenario, the data latency is a main factor for the delay-aware scenario. To decrease the data latency, multiple MCs are deployed in the FoI, and each one has responsibility for energy replenishment and data collection in the specified area where a collection of clusters locate. Therefore, multiple paths of the MCs should be constructed. For the delay-aware scheduling scheme, it not only focuses on the charging utility but also the data update period and the number of the MCs. Due to the requirements of the delay-aware scenario, the objectives of the scheduling scheme are stated in the following:

- *Charging utility maximization:* Due to the multiple paths of the MCs in the WSN, the system-wide charging utility in the delay-aware charging problem is formulated by
$$U = \sum_{p_i \in P} U^{p_i}. \tag{6.8}$$
- *Delay minimization:* There are two factors affecting the delay in the system, i.e., sojourn time/charging time of the MC in each round T_c^{MC} and the traveling time of the MC in each round T_{move}^{MC}. Therefore, the formulation representing the data latency is given by
$$T_{delay}^{MC} = \sum_{k=1}^{K} T_c^{MC} + T_{move}^{MC}, \tag{6.9}$$
where K denotes the number of the clusters in the path of the MC.
- *The number of MCs minimization:* It is necessary to deploy the minimum number of MCs to cover all sensors in the WSN. However, the confined energy capacity of the MC is a critical constraint in the scheme. The less the MCs are adopted in the WSN, the heavier the workload of each one is responsible for. It is unwise for the MC return to the sink to replenish its battery before visiting all clusters in its given path, which causes an additional delay in the delay-aware system.

Problem 2 *Given a set of sensor nodes $S = \{s_1, s_2, ..., s_N\}$, a set of CHs ($CH \subseteq S$) for K clusters, and M MCs, the delay-aware charging scheduling problem aims to obtain M paths P=$\{p_1$,p_2,. . . ,$p_M\}$ for M MCs whose start point and end point are the data sink. The objective of the problem is to maximize the charging utility while minimizing the data latency and cost of the WSN, subject to the energy capacity of the MCs and sensors. We formulate the delay-aware problem in the following:*

$$max \quad \sum_{i=1}^{M} U^{p_i} - \sum_{i=1}^{M} T_{delay}^{MC_i} - M, \tag{6.10}$$

$$s.t. \quad \tau_{k+1} \geq \tau_k + t_c^k + \tfrac{D_{(k,k+1)}}{V^{MC}}, \quad \forall k \in p_i, i = 1, \ldots, M \tag{6.11}$$

$$E_{cap}^{MC_i} \geq E_c^{MC_i} + E_{move}^{MC_i}. \quad i = 1, ..., M \tag{6.12}$$

Same as Constraint (6.7), Constraint (6.11) prevents all sensors from exhausting their energy. Moreover, Constraint (6.12) guarantees that each MC has sufficient energy for energy replenishment and movement in each round.

6.3.3 Problem Hardness Analysis

Recall that; the objective of Problem 1 is maximizing the charging utility while minimizing the energy consumption of the MC. However, the charging utility maximization problem can be proved by a reduction from the orienteering problem [180], and the orienteering problem is a well-known NP-hard problem [181]. Hence, Problem 1 is NP-hard.

On the other hand, Problem 2 aims to derive multiple paths for M MCs, which is in the form of a multiple travel salesman problem. However, the multiple travel salesman problem is also a well-known NP-hard problem [173]. Due to the NP-hardness of these two optimization problems, it is infeasible to derive optimal solutions with brute force or traditional optimization methods. Therefore, we will address these proposed problems with heuristic algorithms.

6.4 Delay-tolerant Scheduling Scheme

In the delay-tolerant system, the charging utility and energy consumption of MCs are crucial factors to affect the charging efficiency in the WSN. As proved above, the delay-tolerant charging problem is NP-hard. Therefore, we design a heuristic algorithm to address the delay-tolerant charging, named single-path scheduling scheme (SPSS). In the scheme, we assume that SPSS implements the full charge to make the WSN semi-permanent. Besides, two MCs are assigned to traverse a single path simultaneously in the opposite directions in order to avoid the sensors exhausting their energy before the MC visits their clusters.

In this scheme, we adopt GA to derive the near-optimal path for the deployed two MCs. In the SPSS, the fitness function is defined based on the objectives of the scheme, i.e., the charging utility and the energy consumed by the MC. Based on the fitness function rule in [166], the fitness function for SPSS is formulated by

$$F = \frac{U^P}{E_{move}^{MC}} = \frac{\sum_{k_i \in P} \sum_{s_i \in k_i} E_H^{s_i}}{E_{move}^{MC}}. \tag{6.13}$$

Here, we assume that sensors should receive the energy to make up the energy consumed on data transceiving. Therefore, the fitness function can be expanded in the following form, i.e., Eq. (6.14).

$$F = \frac{\sum_{k_i \in P} \sum_{s_i \in k_i} E_T^{s_i}}{E_{move}^{MC}} = \frac{\sum_{k_i \in P} \sum_{s_i \in k_i} P_C^{s_i} \times \sum_{k=1}^{K} (T_c^{MC} + T_{move}^{MC})}{P_{move}^{MC} \times \sum_{k=1}^{K} T_{move}^{MC}} \tag{6.14}$$

In our work, the sum of energy consumption rate of all sensors $\sum_{k_i \in P} \sum_{s_i \in k_i} P_C^{s_i}$ and the movement power rate of the MC P_{move}^{MC} are invariable. To simply the fitness function, it is rewritten as follows:

$$F = \omega \times \frac{\sum_{k=1}^{K} T_c^{MC}}{\sum_{k=1}^{K} T_{move}^{MC}}, \tag{6.15}$$

where the value of ω is setting based on $\sum_{k_i \in P} \sum_{s_i \in k_i} P_C^{s_i}$ and P_{move}^{MC}.

Recall that; chromosomes are adopted to represent the solutions in GA, and each one is comprised of a set of genes. In SPSS, the chromosome denotes the path of MCs, and each gene in the chromosome represents a CH selected by the proposed clustering algorithm or the data sink. In order to generate new solutions, the initial population should be generated firstly. The process of the initial population generation is the same as the process in the improved SDCA data collection scheme. The first gene and the last gene in each chromosome are the sink because the MC should start to traverse the path from the sink and return to the sink at the end of each round. All CHs are randomly arranged into a chromosome for many times to create the initial population. The repetition of chromosomes should be prohibited. The fitness values of the initial population will be calculated, and chromosomes are sorted by the fitness value in descending order. After generating the initial population, new solutions will be derived by the main operations of GA, i.e., selection, crossover, mutation. In the SPSS, we utilize methods of GA operations different from the improved SDCA data collection scheme to generate the path of MCs. The main operations in the SPSS are introduced as follows.

1. Selection: In order to derive a new chromosome with high fitness value, the better chromosomes should be selected as parents with high probability. In the SPSS, we apply the roulette wheel selection approach [182]. The probability for each chromosome with fitness value f_i is calculated by

   $$P = \frac{f_i}{\Sigma_{j=1}^{H} f_j}, \tag{6.16}$$

   where H is the size of the population. Two chromosomes with higher fitness values are selected by the roulette wheel selection.

2. Crossover: The improved heuristic crossover operator in [183] is advantageous to generate a better offspring with high probability and is employed as the crossover operation. Two parents are selected by the selection operation, and the offspring will be created by crossover operation. Because the sink is the start point of the chromosome, the sink is the first gene in the offspring and is set to be the current gene. The candidates of the next gene are selected from the two parents. In each loop, the ratio of the charging time in the candidate of the next gene to the moving time from the current gene to the candidate gene in two parents will be compared.

The gene with a bigger ratio will be added into offspring. The current gene is removed from both two parents, and the selected gene will be the current gene in the next loop. The parent whose gene is not selected needs to right-circumvolve until the first gene is the new current gene and enter the next loop. The offspring will be generated once all CHs have been added into it. Then, the fitness value of the offspring will be calculated. The chromosome with the lowest fitness value among the parents and the offspring should be replaced.

3. Mutation: In the SPSS, the exchange mutation approach [184] is used as a mutation operation. Two random selected genes swap their locations in the chromosome. Then, its fitness value should be recalculated.

After the predetermined generations, the chromosome with the highest fitness value will be selected as the path of MCs. Then, the information regarding the path and the clusters is sent to the MCs. They start to go along the path in opposite directions and visit each cluster. Once the MC arrives at the CH in each cluster, it will sojourn and charge all sensors in the cluster. Additionally, the MC will communicate with the CH for data collection. Once the MC has passed by the half of distance of the given road, it will return to the sink, which can prevent the data latency. Due to the finite energy capacity of the MC, when it is ready to move to the next cluster, it needs to check its residual energy in case it will run out of its energy before returning to the data sink. If its residual energy is sufficient to move to the next CH and return to the sink, it will continue executing its tasks until it completes its tour and goes back to the sink; otherwise, it should return to the data sink immediately, forward the gathered data to the sink and charge its battery. Because of the long charging period of the MC, the sink will arrange a new one to take over its work. The new MC first goes back to the last cluster that the original one visits and keeps on traversing the given path, collecting data and charging energy to the sensors. At the end of each tour, the MC returns to the sink, forward the data and charge its battery. To avoid waiting for the charging period of the MC, a new MC with full energy will replace the origin one to go on the next round.

6.5 Delay-aware Scheduling Scheme

From the formulations of the multiple objective problem presented above, we can see that three main components affect the performance of the scheduling scheme, i.e., the sojourn time in each cluster, the path of MCs and the number of MCs. To achieve an effective schedule with low time complexity, we proposed a multiple-path scheduling scheme (MPSS).

6.5.1 Sojourn Period in Each Cluster

For the delay-aware systems, it is infeasible to recharge the sensors fully when the MC arrives and sojourns in each cluster, which is implemented in the SPSS. In the WSN with wireless charging, the energy received by sensors is proportion to the sojourn period for MC in the cluster, which is presented in Eq. (6.3). If the sensors need to receive more energy from the MC, the MC should stay in the cluster for a longer period. However, once the sojourn time in the cluster is too short, the sensors cannot receive sufficient energy from the MC. The sensors might drain their energy before accomplishing the data collection task. To achieve a longer operational time while reducing the data latency and charging delay, we assume that the residual energy of each sensor in the cluster should come up to the predetermined threshold $E^s_{th} = \delta E^s_{cap}$ during the energy charging process, where δ is a ratio between 0 and 1 to determine the E^s_{th} and defined by users. Once the MC arrives in the cluster, all sensors in the cluster will be charged until their residual energy is up to E^s_{th}. Additionally, the MC collects the data in the meanwhile. The MC will depart from the current cluster after fulfilling the data collection and energy charging tasks.

6.5.2 Path Planning of the MCs

To plan the multiple paths for MCs with low complexity, we develop a heuristic algorithm combined the minimum spanning tree and the 2-approximation algorithm. In the proposed scheme, each MC sets out from the data sink and return to the sink at the end of each round so that the MC can transmit the collected data timely and replenish its battery. Moreover, the energy consumed by the MC for each tour should not exceed the energy capacity of the MC E^{MC}_{cap}. Due to the limited energy capacity, the MCs far away from the data sink should be responsible for fewer clusters, because they need to spend more energy on movement. In [173], the authors designed the data gathering algorithm with multiple data collectors. The proposed algorithm is inspired by the covering salesman problem [185] including two subproblems, i.e., obtaining a minimum sojourn point set and planning the shortest path for visiting all sojourn points. In our work, all CHs are assigned as the sojourn points. Therefore, the MPSS is designed by improving the shortest path planning method introduced in [173]. The improved path planning algorithm under the assumptions of MPSS is described in the following. To avoid draining the MC's energy, E^{MC}_{cap} is set to be the upper bound of the energy consumed by the MC in each round. First, each CH calculates the maximum energy E^k_{max} that needs to be replenished by the MC in each cluster, where $E^k_{max} = \sum_{u \epsilon V_i}^{|V|} \delta E^{V_i}_{cap}$. Next, the minimum spanning tree $T(V, E)$ on all CHs is constructed. Each node in T calculates its weight $Weight_v$ which is the sum costs in the subtree whose root is v. The sum costs are comprised of the needed charging energy of each node in the subtree, and the energy consumed on movement between each node and its children. After generating the topology of T, a set of subtrees will be built. Initially, the deepest node u in T is found, and it is assigned as the root of current subtree t which is denoted by $Root_t$. Recall that the start point and the end point in each round are the sink. The energy consumed on movement between the data sink and the subnetwork should be cut. Therefore, the upper bound on energy consumption E_{Res}^{MC} in the current subtree is calculated by

$$E_{Res}^{MC} = E_{cap}^{MC} - 2 * \frac{D_{(u,sink)}}{V^{MC}} P_{move}^{MC}. \tag{6.17}$$

To extend the subtree t, the parent of the root of subtree t in T becomes the candidate, and its weight $Weight_{Parent(Root_t)}$ should be checked if it satisfies $Weight_{Parent(Root_t)} \leq \frac{E_{Res}^{MC}}{2}$. If the parent node meets the requirement, it becomes the root of the current subtree t, and repeat the process of subtree extension; otherwise, the process of subtree extension stops, and the subtree whose root is $Root_t$ in T becomes the current subtree t. After that, the subtree t is removed from T. Due to the changes of the children of the nodes, each node should recalculate its weight, and the new subtree would be built from the remaining T in the next loop. All subtrees are generated until there is no node in T. Then, the path of each subtree can be planned by executing the 2-approximation algorithm for TSP [50]. Besides, the sink should be added into the path, and the CH which is the closest to the sink among all CHs in the path is selected as the adjacent sojourn point with the sink. Fig. 6.2 depicts the workflow of the proposed path planning algorithm in MPSS.

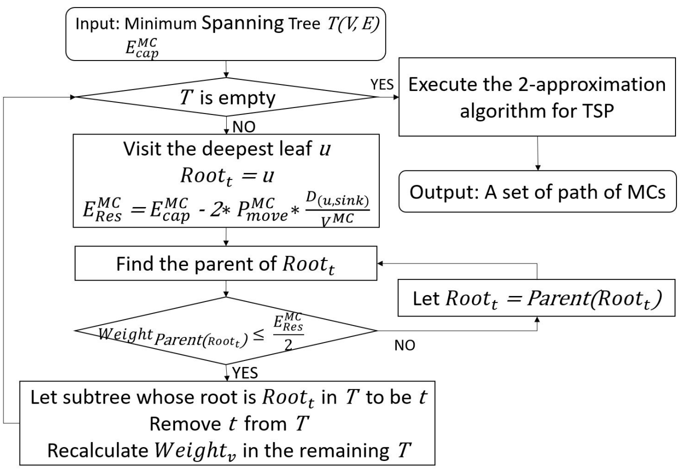


Figure 6.2: The workflow of the proposed path planning algorithm in MPSS

6.6 Performance Evaluation

In this section, we evaluate the performance of our proposed schemes by extensive experiments conducted on the OMNeT++ simulator. Moreover, the result analysis is provided, which can make a better understanding of our proposed schemes.

6.6.1 Simulation Setup

Different from the experiments in other algorithms introduced above, we also should focus on the settings of MCs. In these experiments, we deployed 100 to 300 sensors with uniform distribution in the FoI. All sensors transmit the message within their fixed transmission range R_t^s that was set to be $25m$, and the energy consumption on data transmission was based on the energy model proposed in [66]. The initial energy of the sensor was $2J$. Moreover, the velocity and the power rate of the MC were $5m/s$ and $5W$, respectively. Considering that the MC has finite energy, we set the energy capacity of the MC to be $10800J$. Additionally, the charging range of the MC R_c^{MC} was set to be $50m$, which means that all sensors in the charging range can be recharged by the mobile charger based on the charging model introduced in [176]. All simulation settings are listed in Table 6.1.

Table 6.1: Simulation settings of joint data gathering and wireless charging schemes

Parameter	Description	Value
E_{cap}^s	Energy capacity of sensor	$2J$ [186]
E_{cap}^{MC}	Energy capacity of the MC	$10800J$ [186]
P_t^{MC}	Charging power rate of the MC	$5W$ [177]
P_{move}^{MC}	Movement power rate of the MC	$180W$ [186]
V^{MC}	Velocity of the MC	$18m/s$ [186]
α	The parameter of the charging model	0.864×10^{-4} [176]
β	The parameter of the charging model	0.2316 [176]

6.6.2 Results Analysis

We conducted a collection of simulations to evaluate the performance of the proposed schemes and compared them with another joint energy charging and data collection algorithms, i.e., a joint energy replenish and data collection algorithm [186]. All schemes are cluster-based and multi-hop communication. The differences of these algorithms are the clustering algorithms and the scheduling schemes for the MC. In our proposed scheduling schemes, we adopt the proposed clustering algorithm to select the CHs based on the attributes of sensors and form the clusters in the WSN. Then the schedule for the MCs is designed by SPSS or MPSS on the basis of the delay requirement of the system. In the joint algorithm proposed in [186], the K-means clustering algorithm is used to divide the sensors into a set of clusters, and the CHs are selected according to their residual energy and the distance from them to the center of their own clusters. Two MCs works simultaneously and walk along the shortest Hamiltonian cycle for energy charging and data collection. Considering the size of the WSN and the battery capacities of the sensor and the MC in our experiments, we set the value of δ is 0.9. In the following, we will present and analyze the simulation results from extensive experiments.

The first experiment evaluated the average delay in different schemes. In Fig. 6.3, the result shows the data latency in MPSS is the lowest among all schemes. The average delay is affected by two parameters, i.e., charging time in each cluster, the journey time in each round. The charging time depends on the size of the cluster and the charging energy threshold. Because of the partial charging, the charging time of the MPSS is shorter than the others. In our experiments, compared with the charging time, the delay is mainly affected by the journey time. The journey time is related to the length of paths of MCs. Fig. 6.4 presents the journey length in these charging schemes. Due to the similar objectives proposed in SPSS and the scheme in [186], the result on average delay and the journey length is approximate. The reason for the slight difference in the result is the different clustering algorithms in two schemes. The length of paths generated by the MPSS is relatively shorter. In the MPSS, the data latency should be diminished. Therefore, more MCs are deployed in the WSN for task splitting regarding data collection and energy replenishment, and the length of the tour for each MC is average and short. Except for the short paths of the MCs, another reason for the low data update period of the MPSS is that the length of paths for the MCs is limited based on the energy capacity of the MC. It ensures that each MC can complete its tasks in each round and avoids that the MC returns to the data sink before visiting all clusters. On the contrary, though the SPSS and the algorithm in [186] adopt two MCs to co-operate the tasks while minimizing the energy consumption of MCs on movement, the energy capacity of the MC is not considered. In these cases, the energy consumed on movement and charging may exceed its energy capacity, especially in the large-scale WSN. It causes that the MC needs to return to the sink in advance. Therefore, the MC wastes an amount of time and energy on the movement between the sink and the clusters. Consequently, the MPSS achieves the shortest data update period compared with other schemes.

In the second experiment, we compared the average residual energy. Fig. 6.5 shows the results of the residual energy among all sensors after a period of time. It depicts that the

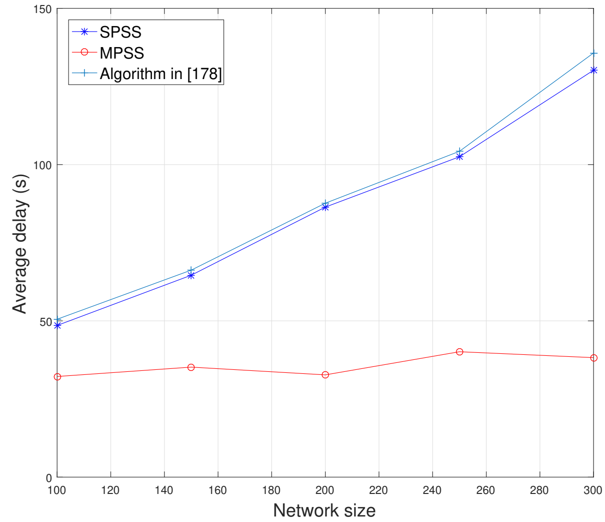


Figure 6.3: Comparison of the average delay of joint data gathering and wireless charging schemes

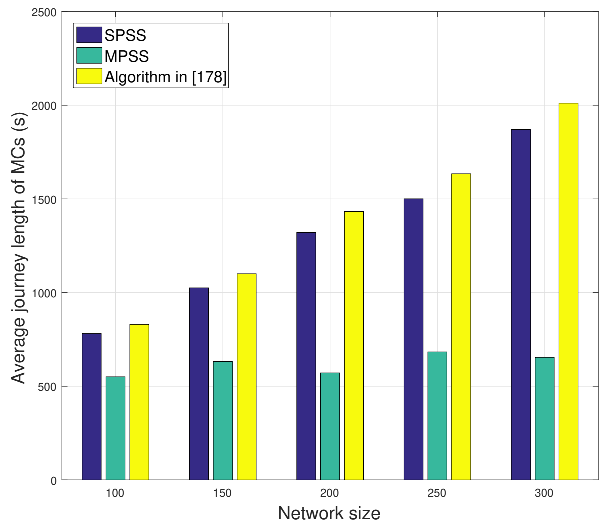


Figure 6.4: The average length of journey paths for MCs

results of the SPSS and the algorithm in [186] are approximate, and the residual energy in both schemes is high. It verifies that these two methods can prolong the network lifetime, even remain sensors alive permanently. The residual energy in the MPSS is lower than the other two schemes, which is determined by the predetermined energy threshold. However, the ratio between the residual energy and the threshold of energy capacity in the MPSS is relatively higher than the other two schemes. The average ratio of the SPSS, the MPSS, and the algorithm in [186] are 0.982, 0.993, 0.978, respectively. Because of the short path of each MC, the charging frequency for each cluster is high. On the other hand, the traveling distance of MCs in the algorithm in [186] is longer than others, which may render a long charging delay, especially in the large-scale WSN.

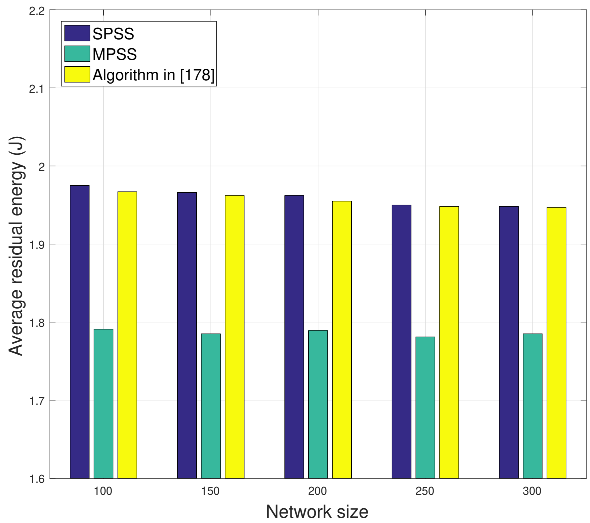


Figure 6.5: Comparison of the average residual energy of joint data gathering and wireless charging schemes

The last experiment was conducted to compare the performance on the charging utility in the different schemes. As shown in Fig. 6.6, the charging utility gain by the SPSS is the highest among all schemes. The reason is that the ratio of the sum of the energy received by sensors to the energy consumed by MCs on movement is set as the fitness function to assess the solutions. It leads to that the MCs in the SPSS consume more energy and time on sensor charging instead of movement. On the contrary, the algorithm in [186] only considers the movement distance of the MCs, and the path distance of MCs is relatively longer than SPSS. Therefore, the MCs will spend more energy on movement. Due to the partial charging, the MCs begin to replenish the sensors when their residual energy is below the energy threshold. Hence, the SPSS achieves high charging utility and charging efficiency.

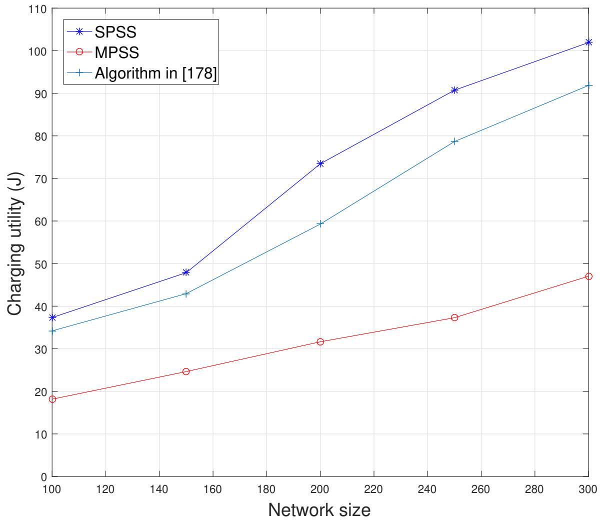


Figure 6.6: Comparison of the charging utility of joint data gathering and wireless charging schemes

6.7 Summary

In this chapter, we studied the problem of the joint data collection and energy replenishment scheme for solving the energy constraint issue in WSNs. Except for energy efficiency, the data latency is also considered in the joint design. Therefore, we proposed a two-mode joint design for facing different system requirements. Due to the NP-hardness of the proposed charging problems, we designed heuristic charging schemes to solve charging problems with low complexity, i.e., SPSS and MPSS. First, the network topology was constructed, and the CHs were selected by the improved clustering algorithm. Second, two modes of scheduling schemes were designed by heuristic algorithms based on their objectives under different scenarios. For the SPSS, the sensors were fully charged to make them alive permanently while improving the charging utility. On the other hand, the MPSS implemented partial charging to decrease the data latency while prolonging the network lifetime. The simulation results showed that the SPSS achieved high charging utility and charging efficiency, and the MPSS can reduce the data latency significantly.

Chapter 7

Conclusion and Future Work

In this section, we conclude the proposed energy efficient algorithms and provide the direction of future work.

7.1 Conclusion

In this thesis, we mainly discussed the energy-efficient issue in WSNs. It is crucial since it affects the network lifetime and system performance in WSNs because of the finite energy capacity of sensors. The delay issue is also critical in WSNs, especially in the delay-sensitive applications. To improve energy efficiency and reduce the data latency, we proposed mobile device-assisted energy-efficient algorithms in WSNs. There are four proposed works in our work: an SDCA data collection scheme, an improved SDCA data collection scheme for the small-scale WSN, a two-mode MDCA data collection mechanism for the large-scale WSN and a joint data collection and energy charging scheme.

In the SDCA data collection scheme, the FoI was divided into the square-shaped grids with the same size, and the sensors in one grid formed a cluster. The optimal path for the data collector was generated by GA with low complexity to reduce the energy consumption of sensors and the data collector. In the improved SDCA scheme, the sensors initially were grouped into clusters based on our newly proposed clustering algorithm. Then, the path for the single collector was derived by modified GA.

For the MDCA data collection scheme, we developed a two-mode data gathering mechanism by adopting multiple collectors, which could switch its status between the delay-aware mode and the delay-tolerant mode. The delay-aware MDCA scheme ran in the gathering-and-carrying mode in which all collectors transmitted their gathered data to the data sink directly. It could help to shorten the data latency. The delay-tolerant MDCA schemes ran in the data-relaying mode in which the collectors gathered data from the distant subnetworks and transmitted data to the data sink by intermediate collectors. It was helpful to enhance the energy efficiency of the collectors.

Similar to the MDCA data collection scheme, two types of MC scheduling schemes in the joint data collection and energy charging design based on different delay requirements of systems. In the delay-tolerant scenario, SPSS is developed by adopting two MCs to traverse a single derived path by the heuristic algorithm in order to execute the data collection and wireless charging cooperatively. On the other hand, the MPSS implements a minimal number of MCs to operate the data gathering and wireless charging. Each MC is responsible for a set of designated clusters, which ensures that the MC fulfills its task before running out of its battery and does not need to return to the sink in advance.

7.2 Future Work

There are several aspects of our proposed works that can be extended to further improve the system performance regarding energy efficiency, data latency and other practical problems.

Though our work has designed many methods to reduce the traveling length of the mobile device (such as clustering, heuristic-based the optimal path planning for the mobile device), they still spend considerable time visiting many sensors. WSNs are sometimes used for event detection or target tracking, and the sensors need not work or will not generate any sensed data all the time. Therefore, the mobile device doesn't have to visit these sensors in each round. For this situation, we can combine the proposed energy-efficient schemes and the node scheduling approach to design a new path planning algorithm by considering the event prediction [187, 188, 189]. The mobile data collector will traverse some specific locations where the target will appear with a high probability, which can significantly shorten the data update period.

In real life, the nodes usually are deployed in diverse environments where conditions are very complex [190]. For example, there are many obstacles in the FoI [117, 191, 192, 193, 194], and the mobile device cannot traverse the region of the obstacle. Therefore, we need to improve our scheme and take these regions that cannot be passed through by devices into consideration. Besides, we can implement the proposed energy efficient algorithms in other situations by considering different assumptions, which makes our schemes more practical, e.g. underwater networks [195, 196, 197, 198, 199], smart dust networks [200].

References

[1] T. Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall, 2 edition, 2001.

[2] A. Boukerche and K. Abrougui. An efficient leader election protocol for mobile networks. In *Proceedings of ACM international conference on Wireless communications and mobile computing (IWCMC)*, pages 1129–1134, 2006.

[3] A. Boukerche, I. Chatzigiannakis, and S. Nikoletseas. Power-efficient data propagation protocols for wireless sensor networks. *Simulation*, 81(6):399–411, 2005.

[4] C. C. Lin, D. J. Deng, Z. Y. Chen, and K. C. Chen. Key design of driving industry 4.0: joint energy-efficient deployment and scheduling in group-based industrial wireless sensor networks. *IEEE Communications Magazine*, 54(10):46–52, 2016.

[5] C. Konstantopoulos, G. Pantziou, D. Gavalas, A. Mpitziopoulos, and B. Mamalis. A rendezvous-based approach enabling energy-efficient sensory data collection with mobile sinks. *IEEE Transactions on Parallel and Distributed Systems*, 23(5):809–817, 2012.

[6] S. Guo, Y. Shi, Y. Yang, and B. Xiao. Energy efficiency maximization in mobile wireless energy harvesting sensor networks. *IEEE Transactions on Mobile Computing*, 17(7):1524–1537, 2018.

[7] A. Boukerche and P. Sun. Connectivity and coverage based protocols for wireless sensor networks. *Ad Hoc Networks*, 80:54–69, 2018.

[8] Y. Ren, R. Werner, N. Pazzi, and A. Boukerche. Monitoring patients via a secure and mobile healthcare system. *IEEE Wireless Communications*, 17(1):59–65, 2010.

[9] A. Boukerche, S. Hong, and T. Jacob. An efficient synchronization scheme of multimedia streams in wireless and mobile systems. *IEEE Transactions on Parallel and Distributed Systems*, 13(9):911–923, 2002.

[10] A. Boukerche. A simulation based study of on-demand routing protocols for ad hoc wireless networks. In *Proceedings of 34th Annual Simulation Symposium (SS)*, pages 85–92, 2001.

[11] A. Boukerch, L. Xu, and K. El-Khatib. Trust-based security for wireless ad hoc and sensor networks. *Computer Communications*, 30(11-12):2413–2427, 2007.

[12] A. Boukerche and Y. Ren. A security management scheme using a novel computational reputation model for wireless and mobile ad hoc networks. In *Proceedings of the 5th ACM International Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (PE-WASUN)*, pages 88–95, 2008.

[13] A. Boukerche, Y. Du, J. Feng, and R. W. N. Pazzi. A reliable synchronous transport protocol for wireless image sensor networks. In *Proceedings of the 13th IEEE Symposium on Computers and Communications (ISCC)*, pages 1083–1089, 2008.

[14] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella. Energy conservation in wireless sensor networks: A survey. *Ad Hoc Networks*, 7(3):537–568, 2009.

[15] R. W. L. Coutinho, A. Boukerche, L. F. M. Vieira, and A. A. F. Loureiro. Transmission power control-based opportunistic routing for wireless sensor networks. In *Proceedings of ACM international conference on Modeling, analysis and simulation of wireless and mobile systems (MSWiM)*, pages 219–226, 2014.

[16] J. Li and P. Mohapatra. An analytical model for the energy hole problem in many-to-one sensor networks. In *Proceedings of IEEE Vehicular Technology Conference (VTC)*, volume 4, pages 2721–2725, 2005.

[17] A. Liu, X. Jin, G. Cui, and Z. Chen. Deployment guidelines for achieving maximum lifetime and avoiding energy holes in sensor network. *Information Sciences*, 230:197–226, 2013.

[18] H. A. Hashim, B. O. Ayinde, and M. A. Abido. Optimal placement of relay nodes in wireless sensor network using artificial bee colony algorithm. *Journal of Network and Computer Applications*, 64:239 – 248, 2016.

[19] A. Boukerche, N. J. McGraw, C. Dzermajko, and K. Lu. Grid-filtered region-based data distribution management in large-scale distributed simulation systems. In *Proceedings of 38th Annual Simulation Symposium (ANSS)*, pages 259–266, 2005.

[20] J. Long, M. Dong, K. Ota, and A. Liu. A green tdma scheduling algorithm for prolonging lifetime in wireless sensor networks. *IEEE Systems Journal*, 11(2):868–877, 2017.

[21] J. Lee and J. Lee. Ant-colony-based scheduling algorithm for energy-efficient coverage of wsn. *IEEE Sensors Journal*, 12(10):3036–3046, 2012.

[22] A. Boukerche and D. Turgut. Secure time synchronization protocols for wireless sensor networks. *IEEE Wireless Communications*, 14(5):64–69, 2007.

[23] P. S. Mann and S. Singh. Energy-efficient hierarchical routing for wireless sensor networks: a swarm intelligence approach. *Wireless Personal Communications*, 92(2):785–805, 2017.

[24] C. G. Rezende, A. Mammeri, A. Boukerche, and A. A. F. Loureiro. A receiver-based video dissemination solution for vehicular networks with content transmissions decoupled from relay node selection. *Ad Hoc Networks*, 17:1–17, 2014.

[25] H. Xie, A. Boukerche, and A. A. F. Loureiro. A multipath video streaming solution for vehicular networks with link disjoint and node-disjoint. *IEEE Transactions on Parallel and Distributed Systems*, 26(12):3223–3235, 2015.

[26] A. Boukerche and S. K. Das. Dynamic load balancing strategies for conservative parallel simulations. In *Proceedings of the 11th Workshop Parallel and Distributed Simulation (PADS)*, pages 20–28, 1997.

[27] E. El Ajaltouni, A. Boukerche, and M. Zhang. An efficient dynamic load balancing scheme for distributed simulations on a grid infrastructure. In *Proceedings of the 12th IEEE/ACM International Symposium on Distributed Simulation and Real-Time Applications*, pages 61–68, 2008.

[28] B. Prince and S. Gupta. Load balanced and energy efficient data collection scheme using data mule for wsns. In *Proceedings of IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, pages 1651–1655, 2016.

[29] C. Wang, S. Guo, and Y. Yang. An optimization framework for mobile data collection in energy-harvesting wireless sensor networks. *IEEE Transactions on Mobile Computing*, 15(12):2969–2986, 2016.

[30] M. H. Anisi, G. Abdul-Salaam, M. Y. I. Idris, A. W. A. Wahab, and I. Ahmedy. Energy harvesting and battery power based routing in wireless sensor networks. *Wireless Networks*, 23(1):249–266, 2017.

[31] D. Zhang, Z. Chen, H. Zhou, L. Chen, and X. S. Shen. Energy-balanced cooperative transmission based on relay selection and power control in energy harvesting wireless sensor network. *Computer Networks*, 104:189–197, 2016.

[32] Y. Liu, S. A. Mousavifar, Y. Deng, C. Leung, and M. Elkashlan. Wireless energy harvesting in a cognitive relay network. *IEEE Transactions on Wireless Communications*, 15(4):2498–2508, 2016.

[33] D. Jiang, X. Ying, Y. Han, and Z. Lv. Collaborative multi-hop routing in cognitive wireless networks. *Wireless Personal Communications*, 86(2):901–923, 2016.

[34] M. Elhadef, A. Boukerche, and H. Elkadiki. Performance analysis of a distributed comparison-based self-diagnosis protocol for wireless ad-hoc networks. In *Proceedings of 9th International Symposium on Modeling Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, pages 165–172, 2006.

[35] G. Xing, T. Wang, W. Jia, and M. Li. Rendezvous design algorithms for wireless sensor networks with a mobile base station. In *Proceedings of ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*, pages 231–240, 2008.

[36] N. Kumar and D. Dash. Mobile data sink-based time-constrained data collection from mobile sensors: a heuristic approach. *IET Wireless Sensor Systems*, 8(3):129–135, 2018.

[37] H. Zhao, S. Guo, X. Wang, and F. Wang. Energy-efficient topology control algorithm for maximizing network lifetime in wireless sensor networks with mobile sink. *Applied Soft Computing*, 34:539–550, 2015.

[38] A. W. Khan, A. H. Abdullah, M. A. Razzaque, and J. I. Bangash. Vgdra: a virtual grid-based dynamic routes adjustment scheme for mobile sink-based wireless sensor networks. *IEEE Sensors Journal*, 15(1):526–534, 2015.

[39] J. Chang and T. Shen. An efficient tree-based power saving scheme for wireless sensor networks with mobile sink. *IEEE Sensors Journal*, 16(20):7545–7557, 2016.

[40] A. Boukerche and A. Martirosyan. An energy efficient and low latency multiple events' propagation protocol for wireless sensor networks with multiple sinks. In *Proceedings of ACM workshop on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks (PE-WASUN)*, pages 82–86, 2007.

[41] X. Ma, R. Kacimi, and R. Dhaou. Fairness-aware uav-assisted data collection in mobile wireless sensor networks. In *Proceedings of IEEE International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 995–1001, 2016.

[42] T. Rault, A. Bouabdallah, and Y. Challal. Energy efficiency in wireless sensor networks: A top-down survey. *Computer Networks*, 67:104–122, 2014.

[43] A. Boukerche, A. Zarrad, and R. B. Araujo. A cross-layer approach-based gnutella for collaborative virtual environments over mobile ad hoc networks. *IEEE Transactions on Parallel and Distributed Systems*, 21(7):911–924, 2010.

[44] K. Zeng, K. Ren, W. Lou, and P. J. Moran. Energy aware efficient geographic routing in lossy wireless sensor networks with environmental energy supply. *Wireless Networks*, 15(1):39–51, 2009.

[45] W. K. G. Seah, Z. A. Eu, and H. Tan. Wireless sensor networks powered by ambient energy harvesting (wsn-heap) - survey and challenges. In *Proceedings of IEEE International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology (VITAE)*, pages 1–5, 2009.

[46] A. Cammarano, C. Petrioli, and D. Spenza. Online energy harvesting prediction in environmentally powered wireless sensor networks. *IEEE Sensors Journal*, 16(17):6793–6804, 2016.

[47] S. Kosunalp. An energy prediction algorithm for wind-powered wireless sensor networks with energy harvesting. *Energy*, 139:1275 – 1280, 2017.

[48] M. Ayaz, E. Farjah, and T. Ghanbari. An efficient power supply for wireless sensor networks through environmental energy harvesting. In *Proceedings of IEEE Iranian Conference on Renewable Energy & Distributed Generation (ICREDG)*, pages 1–6, 2016.

[49] F. K. Shaikh and S. Zeadally. Energy harvesting in wireless sensor networks: A comprehensive review. *Renewable and Sustainable Energy Reviews*, 55:1041 – 1054, 2016.

[50] S. S. Skiena. Traveling salesman problem. *The Algorithm Design Manual*, pages 319–322, 1997.

[51] N. Sabor, S. Sasaki, M. Abo-Zahhad, and S. M. Ahmed. A comprehensive survey on hierarchical-based routing protocols for mobile wireless sensor networks: review, taxonomy, and future directions. *Wireless Communications and Mobile Computing*, 2017(5):1–23, 2017.

[52] A. Boukerche and X. Fei. A coverage-preserving scheme for wireless sensor network with irregular sensing range. *Ad Hoc Networks*, 5(8):1303–1316, 2007.

[53] A. Boukerche and C. Dzermajko. Performance evaluation of data distribution management strategies. *Concurrency and Computation: Practice and Experience*, 16(15):1545–1573, 2004.

[54] S. Correia, A. Boukerche, and R. I. Meneguette. An architecture for hierarchical software-defined vehicular networks. *IEEE Communications Magazine*, 55(7):80–86, 2017.

[55] A. Howard, M. J. Mataric, and G. S. Sukhatme. Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem. In *Proceedings of the 6th International Symposium on Distributed Autonomous Robotic Systems (DARS)*, pages 299–308, 2002.

[56] Y. Zou and K. Chakrabarty. Sensor deployment and target localization based on virtual forces. In *Proceedings of Annual Joint Conference of the IEEE Computer and Communications Societies (INFCOM)*, volume 2, pages 1293 – 1303, 2003.

[57] Y. Zou and K. Chakrabarty. Sensor deployment and target localization in distributed sensor networks. *ACM Transactions on Embedded Computing Systems*, 3(1):61–91, 2004.

[58] K. Derr and M. Manic. Wireless sensor network configuration - part ii: Adaptive coverage for decentralized algorithms. *IEEE Transactions on Industrial Informatics*, 9(3):1728–1738, 2013.

[59] X. Liu. Sensor deployment of wireless sensor networks based on ant colony optimization with three classes of ant transitions. *IEEE Communications Letters*, 16(10):1604–1607, 2012.

[60] W. Liao, S. Kuai, and M. Lin. An energy-efficient sensor deployment scheme for wireless sensor networks using ant colony optimization algorithm. *Wireless Personal Communications*, 82(4):2135–2153, 2015.

[61] X. Hao, W. Liu, N. Yao, D. Geng, and X. Li. Distributed topology construction algorithm to improve link quality and energy efficiency for wireless sensor networks. *Journal of Network and Computer Applications*, 72:162–170, 2016.

[62] E. A. Khalil and S. Ozdemir. Reliable and energy efficient topology control in probabilistic wireless sensor networks via multi-objective optimization. *The Journal of Supercomputing*, 73(6):2632–2656, 2017.

[63] Y. Yoon and Y. H. Kim. An efficient genetic algorithm for maximum coverage deployment in wireless sensor networks. *IEEE Transactions on Cybernetics*, 43(5):1473–1483, 2013.

[64] T. Fan, G. Teng, and L. Huo. Deployment strategy of wsn based on minimizing cost per unit area. *Computer Communications*, 38:26 – 35, 2014.

[65] S. Halder and S. Das Bit. Enhancement of wireless sensor network lifetime by deploying heterogeneous nodes. *Journal of Network and Computer Applications*, 38:106–124, 2014.

[66] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of IEEE Annual Hawaii International Conference on System Sciences (HICSS)*, pages 1–10, 2000.

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

[68] M. B. Yassein, Y. Khamayseh, and W. Mardini. Improvement on leach protocol of wireless sensor network (vleach). *International Journal of Digital Content Technology and its Applications*, 3:132–136, 2009.

[69] V. Loscri, G. Morabito, and S. Marano. A two-levels hierarchy for low-energy adaptive clustering hierarchy (tl-leach). In *Proceedings of IEEE Vehicular Technology Conference (VTC)*, volume 3, pages 1809–1813, 2005.

[70] J. S. Lee and W. L. Cheng. Fuzzy-logic-based clustering approach for wireless sensor networks using energy predication. *IEEE Sensors Journal*, 12(9):2891–2897, 2012.

[71] G. Ran, H. Zhang, and S. Gong. Improving on leach protocol of wireless sensor networks using fuzzy logic. *Journal of Information and Computational Science*, 7(3):767–775, 2010.

[72] M. Tong and M. Tang. Leach-b: An improved leach protocol for wireless sensor network. In *Proceedings of IEEE International Conference on Wireless Communications Networking and Mobile Computing (WiCOM)*, pages 1–4, 2010.

[73] F. Xiangning and S. Yulin. Improvement on leach protocol of wireless sensor network. In *Proceedings of IEEE International Conference on Sensor Technologies and Applications (SENSORCOMM)*, pages 260–264, 2007.

[74] J. Zhao and L. Yang. Leach-a: An adaptive method for improving leach protocol. *Sensors and Transducers*, 162:136–140, 2014.

[75] A. Chamam and S. Pierre. On the planning of wireless sensor networks: Energy-efficient clustering under the joint routing and coverage constraint. *IEEE Transactions on Mobile Computing*, 8(8):1077–1086, 2009.

[76] K. Sundaran, V. Ganapathy, and P. Sudhakara. Fuzzy logic based unequal clustering in wireless sensor network for minimizing energy consumption. In *Proceedings of IEEE International Conference on Computing and Communications Technologies (ICCCT)*, pages 304–309, 2017.

[77] P. Nayak and B. Vathasavai. Energy efficient clustering algorithm for multi-hop wireless sensor network using type-2 fuzzy logic. *IEEE Sensors Journal*, 17(14):4492–4499, 2017.

[78] A. Boukerche and K. Abrougui. An efficient leader election protocol for wireless quasi-static mesh networks: Proof of correctness. In *Proceedings of IEEE International Conference on Communications (ICC)*, pages 3491–3496, 2007.

[79] R. Han, W. Yang, Y. Wang, and K. You. Dce: A distributed energy-efficient clustering protocol for wireless sensor network based on double-phase cluster-head election. *Sensors*, 17(5):998:1–998:15, 2017.

[80] W. Ye, J. Heidemann, and D. Estrin. An energy-efficient mac protocol for wireless sensor networks. In *Proceedings of Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies (INFCOM)*, volume 3, pages 1567–1576, 2002.

[81] C. Schurgers, V. Tsiatsis, and M. B. Srivastava. Stem: Topology management for energy efficient sensor networks. In *Proceedings IEEE Aerospace Conference*, volume 3, pages 1099–1108, 2002.

[82] F. Tseng, H. Cho, L. Chou, T. K. Shih, and H. Chao. An efficient power conservation scheme in non-zero-sum duty-cycle game for wireless sensor networks. *International Journal of Sensor Networks*, 21(4):242–251, 2016.

[83] J. Deng, Y. S. Han, W. B. Heinzelman, and P. K. Varshney. Scheduling sleeping nodes in high density cluster-based sensor networks. *Mobile Networks and Applications*, 10(6):825–835, 2005.

[84] Y. Lin, X. M. Hu, J. Zhang, O. Liu, and H. l. Liu. Optimal node scheduling for the lifetime maximization of two-tier wireless sensor networks. In *Proceedings of IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8, 2010.

[85] S. Hassan, M. S. Nisar, and H. Jiang. *Multi-level sleep scheduling for heterogeneous wireless sensor networks*, chapter 2, pages 227–233. 2016.

[86] Y. Kang, B. Hu, Y. Ding, and J. Tan. A hybrid node scheduling approach based on energy efficient chain routing for wsn. *Advances in Mechanical Engineering*, 6:1–12, 2014.

[87] Y. Wang, Y. Zhang, J. Liu, and R. Bhandari. *Coverage, connectivity, and deployment in wireless sensor networks*, pages 25–44. 2015.

[88] A. Boukerche, R. W. N. Pazzi, and R. B. de Araujo. A fast and reliable protocol for wireless sensor networks in critical conditions monitoring applications. In *Proceedings of the 7th International Symposium on Modeling Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, pages 157–164, 2004.

[89] A. Boukerche and S. Samarah. A novel algorithm for mining association rules in wireless ad hoc sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 19(7):865–877, 2008.

[90] R. Oliveira, C. Montez, A. Boukerche, and M. S. Wangham. Reliable data dissemination protocol for VANET traffic safety applications. *Ad Hoc Networks*, 63:30–44, 2017.

[91] J. Chang and L. Tassiulas. Maximum lifetime routing in wireless sensor networks. *IEEE/ACM Transactions on Networking*, 12(4):609–619, 2004.

[92] C. C. Hsu, M. S. Kuo, S. C. Wang, and C. F. Chou. Joint design of asynchronous sleep-wake scheduling and opportunistic routing in wireless sensor networks. *IEEE Transactions on Computers*, 63(7):1840–1846, 2014.

[93] M. Liu, S. Xu, and S. Sun. An agent-assisted qos-based routing algorithm for wireless sensor networks. *Journal of Network and Computer Applications*, 35(1):29–36, 2012.

[94] K. Vijayan and A. Raaza. A novel cluster arrangement energy efficient routing protocol for wireless sensor networks. *Indian Journal of science and Technology*, 9(2), 2016.

[95] A. M. Zungeru, K. P. Seng, L. Ang, and W. C. Chia. Energy efficiency performance improvements for ant-based routing algorithm in wireless sensor networks. *Journal of Sensors*, 2013:1–17, 2013.

[96] B. Zeng and Y. Dong. An improved harmony search based energy-efficient routing algorithm for wireless sensor networks. *Applied Soft Computing*, 41:135–147, 2016.

[97] S. Mini, S. K. Udgata, and S. L. Sabat. Sensor deployment and scheduling for target coverage problem in wireless sensor networks. *IEEE Sensors Journal*, 14(3):636–644, 2014.

[98] D. Karaboga and B. Akay. A survey: algorithms simulating bee swarm intelligence. *Artificial intelligence review*, 31(1-4):61–85, 2009.

[99] G. Han, L. Liu, J. Jiang, L. Shu, and G. Hancke. Analysis of energy-efficient connected target coverage algorithms for industrial wireless sensor networks. *IEEE Transactions on Industrial Informatics*, 13(1):135–143, 2017.

[100] M. Castelli, S. Silva, L. Manzoni, and L. Vanneschi. Geometric selective harmony search. *Information Sciences*, 279:468–482, 2014.

[101] C. C. Hsu, M. S. Kuo, S. C. Wang, and C. F. Chou. Joint design of asynchronous sleep-wake scheduling, and opportunistic routing in wireless sensor networks. Technical report, 2012.

[102] C. Zhu, L. T. Yang, L. Shu, V. C. M. Leung, J. J. P. C. Rodrigues, and L. Wang. Sleep scheduling for geographic routing in duty-cycled mobile sensor networks. *IEEE Transactions on Industrial Electronics*, 61(11):6346–6355, 2014.

[103] S. Guo, L. He, Y. Gu, B. Jiang, and T. He. Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links. *IEEE Transactions on Computers*, 63(11):2787–2802, 2014.

[104] C. Buratti and R. Verdone. Joint scheduling and routing with power control for centralized wireless sensor networks. *Wireless Networks*, 24(5):1699–1714, 2018.

[105] Z. Walczak and J. M. Wojciechowski. Transmission scheduling in packet radio networks using graph coloring algorithm. In *Proceedings of IEEE International Conference on Wireless and Mobile Communications (ICWMC)*, pages 46–46, 2006.

[106] M. Elsersy, T. M. Elfouly, and M. H. Ahmed. Joint optimal placement, routing, and flow assignment in wireless sensor networks for structural health monitoring. *IEEE Sensors Journal*, 16(12):5095–5106, 2016.

[107] A. Boukerche, S. K. Das, and A. Fabbri. Swimnet: A scalable parallel simulation testbed for wireless and mobile networks. *Wireless Networks*, 7(5):467–486, 2001.

[108] A. Boukerche, R. W. N. Pazzi, and J. Feng. An end-to-end virtual environment streaming technique for thin mobile devices over heterogeneous networks. *Computer Communications*, 31(11):2716–2725, 2008.

[109] Z. M. Wang, S. Basagni, E. Melachrinoudis, and C. Petrioli. Exploiting sink mobility for maximizing sensor networks lifetime. In *Proceedings of IEEE Annual Hawaii International Conference on System Sciences (HICSS)*, pages 287a–295a, 2005.

[110] G. Chen, C. Li, M. Ye, and J. Wu. An unequal cluster-based routing protocol in wireless sensor networks. *Wireless Networks*, 15(2):193–207, 2009.

[111] W. Liang, J. Luo, and X. Xu. Prolonging network lifetime via a controlled mobile sink in wireless sensor networks. In *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM)*, pages 1–6, 2010.

[112] A. Kaswan, K. Nitesh, and P. K. Jana. Energy efficient path selection for mobile sink and data gathering in wireless sensor networks. *AEU - International Journal of Electronics and Communications*, 73:110–118, 2017.

[113] J. A Hartigan and M. A Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C*, 28(1):100–108, 1979.

[114] D. S. Johnson, G. Gutin, L. A. McGeoch, A. Yeo, W. Zhang, and A. Zverovitch. Experimental analysis of heuristics for the atsp. In *The traveling salesman problem and its variations*, pages 445–487. 2007.

[115] J. S. Liu, S. Y. Wu, and K. M. Chiu. Path planning of a data mule in wireless sensor network using an improved implementation of clustering-based genetic algorithm. In *Proceedings of IEEE Symposium on Computational Intelligence in Control and Automation (CICA)*, pages 30–37, 2013.

[116] K. M. Chiu and J. S. Liu. Robot routing using clustering-based parallel genetic algorithm with migration. In *Proceedings of IEEE Workshop On Merging Fields Of Computational Intelligence And Sensor Technology (MFCIST)*, pages 42–49, 2011.

[117] G. Xie, K. Ota, M. Dong, F. Pan, and A. Liu. Energy-efficient routing for mobile data collectors in wireless sensor networks with obstacles. *Peer-to-Peer Networking and Applications*, 10(3):472–483, 2017.

[118] H. Zhou, N. Shenoy, and W. Nicholls. Efficient minimum spanning tree construction without delaunay triangulation. In *Proceedings of the Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 192–197, 2001.

[119] Jan L. A. van de Snepscheut. A derivation of a distributed implementation of warshall's algorithm. *Science of Computer Programming*, 7(1):55–60, 1986.

[120] G. Abdul-Salaam, A. H. Abdullah, M. H. Anisi, A. Gani, and A. Alelaiwi. A comparative analysis of energy conservation approaches in hybrid wireless sensor networks data collection protocols. *Telecommunication Systems*, 61(1):159–179, 2016.

[121] S. K. Singh, P. Kumar, and J. P. Singh. An energy efficient odd-even round number based data collection using mules in wsns. In *Proceedings of IEEE International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pages 1255–1259, 2016.

[122] C. Zhan, Y. Zeng, and R. Zhang. Energy-efficient data collection in uav enabled wireless sensor network. *IEEE Wireless Communications Letters*, 7(3):328–331, 2018.

[123] Y. Zeng and R. Zhang. Energy-efficient uav communication with trajectory optimization. *IEEE Transactions on Wireless Communications*, 16(6):3747–3760, 2017.

[124] Y. Shi and Y. T. Hou. Theoretical results on base station movement problem for sensor network. In *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, pages 1–5, 2008.

[125] F. Akhtar and M. H. Rehmani. Energy replenishment using renewable and traditional energy resources for sustainable wireless sensor networks: A review. *Renewable and Sustainable Energy Reviews*, 45:769–784, 2015.

[126] J. Zheng, H. Zhang, Y. Cai, R. Li, and A. Anpalagan. Game-theoretic multi-channel multi-access in energy harvesting wireless sensor networks. *IEEE Sensors Journal*, 16(11):4587–4594, 2016.

[127] X. Fafoutis, A. Di Mauro, C. Orfanidis, and N. Dragoni. Energy-efficient medium access control for energy harvesting communications. *IEEE Transactions on Consumer Electronics*, 61(4):402–410, 2015.

[128] W. Dinkelbach. On nonlinear fractional programming. *Management science*, 13(7):492–498, 1967.

[129] Y. Zhang, S. He, and J. Chen. Data gathering optimization by dynamic sensing and routing in rechargeable sensor networks. *IEEE/ACM Transactions on Networking*, 24(3):1632–1646, 2016.

[130] M. Chiang. Balancing transport and physical layers in wireless multihop networks: Jointly optimal congestion control and power control. *IEEE Journal on Selected Areas in Communications*, 23(1):104–116, 2005.

[131] G. Martinez, S. Li, and C. Zhou. Multi-commodity online maximum lifetime utility routing for energy-harvesting wireless sensor networks. In *Proceedings of IEEE Global Communications Conference (GLOBECOM)*, pages 106–111, 2014.

[132] S. Bi and R. Zhang. Placement optimization of energy and information access points in wireless powered communication networks. *IEEE Transactions on Wireless Communications*, 15(3):2351–2364, 2016.

[133] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[134] C. Moraes and D. Har. Charging distributed sensor nodes exploiting clustering and energy trading. *IEEE Sensors Journal*, 17(2):546–555, 2017.

[135] A. Boukerche and P. Sun. A novel hierarchical two-tier node deployment strategy for sustainable wireless sensor networks. *IEEE Transactions on Sustainable Computing*, 3(4):236–247, 2018.

[136] D. Wu, J. He, H. Wang, C. Wang, and R. Wang. A hierarchical packet forwarding mechanism for energy harvesting wireless sensor networks. *IEEE Communications Magazine*, 53(8):92–98, 2015.

[137] A. Sunny and J. Kuri. A framework for designing multihop energy harvesting sensor networks. *IEEE Journal on Selected Areas in Communications*, 34(5):1491–1501, 2016.

[138] A. H. B. Duffy. The "what" and "how" of learning in design. *IEEE Expert*, 12(3):71–76, 1997.

[139] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–44, 2015.

[140] J. Kang, C. Chun, and I. Kim. Deep-learning-based channel estimation for wireless energy transfer. *IEEE Communications Letters*, 22(11):2310–2313, 2018.

[141] L. P. Kaelbling, M. L. Littman, and A. W. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.

[142] P. Dayan and C. J. C. H. Watkins. Q-learning. *Machine learning*, 8(3):279–292, 1992.

[143] H. Chen, X. Li, and F. Zhao. A reinforcement learning-based sleep scheduling algorithm for desired area coverage in solar-powered wireless sensor networks. *IEEE Sensors Journal*, 16(8):2763–2774, 2016.

[144] D. Ye and M. Zhang. A self-adaptive sleep/wake-up scheduling approach for wireless sensor networks. *IEEE Transactions on Cybernetics*, 48(3):979–992, 2018.

[145] F. A. Aoudia, M. Gautier, and O. Berder. Rlman: An energy manager based on reinforcement learning for energy harvesting wireless sensor networks. *IEEE Transactions on Green Communications and Networking*, 2(2):408–417, 2018.

[146] A. Boukerche, S. Hong, and T. Jacob. A distributed algorithm for dynamic channel allocation. *Mobile Networks and Applications*, 7(2):115–126, 2002.

[147] J. Mitola and G. Q. Maguire. Cognitive radio: making software radios more personal. *IEEE Personal Communications*, 6(4):13–18, 1999.

[148] N. Hoven, R. Tandra, and A. Sahai. Some fundamental limits on cognitive radio. In *Proceeding of Allerton Conference on Communication, Control, and Computing (ALLERTON)*, 2004.

[149] D. Zhang, Z. Chen, J. Ren, N. Zhang, M. K. Awad, H. Zhou, and X. S. Shen. Energy-harvesting-aided spectrum sensing and data transmission in heterogeneous cognitive radio sensor network. *IEEE Transactions on Vehicular Technology*, 66(1):831–843, 2017.

[150] P. Sun and N. Samaan. Random node failures and wireless networks connectivity: Theoretical analysis. *IEEE Wireless Communications Letters*, 4(5):461–464, 2015.

[151] A. Boukerche, C. G. Rezende, and R. W. N. Pazzi. Improving neighbor localization in vehicular ad hoc networks to avoid overhead from periodic messages. In *Proceedings of the Global Communications Conference (GLOBECOM)*, pages 1–6, 2009.

[152] A. Boukerche and S. Rogers. GPS query optimization in mobile and wireless networks. In *Proceedings of the Sixth IEEE Symposium on Computers and Communications (ISCC)*, pages 198–203, 2001.

[153] A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro. Vehicular ad hoc networks: A new challenge for localization-based systems. *Computer Communications*, 31(12):2838–2849, 2008.

[154] H. A. B. F. De Oliveira, A. Boukerche, E. F. Nakamura, and A. A. F. Loureiro. An efficient directed localization recursion protocol for wireless sensor networks. *IEEE Transactions on Computers*, 58(5):677–691, 2008.

[155] A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro. Secure localization algorithms for wireless sensor networks. *IEEE Communications Magazine*, 46(4):96–101, 2008.

[156] A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro. Localization systems for wireless sensor networks. *IEEE Wireless Communications*, 14(6):6–12, 2007.

[157] O. Abumansoor and A. Boukerche. A secure cooperative approach for nonline-of-sight location verification in vanet. *IEEE Transactions on Vehicular Technology*, 61(1):275–285, 2012.

[158] R. Wang, C. G. Rezende, H. S. Ramos, R. W. Pazzi, A. Boukerche, and A. A. F. Loureiro. LIAITHON: A location-aware multipath video streaming scheme for urban vehicular networks. In *Proceedings of 2012 IEEE Symposium on Computers and Communications (ISCC)*, pages 436–441, 2012.

[159] K. Abrougui, A. Boukerche, and R. W. N. Pazzi. Design and evaluation of context-aware and location-based service discovery protocols for vehicular networks. *IEEE Transactions on Intelligent Transportation Systems*, 12(3):717–735, 2011.

[160] H. A. B. F. de Oliveira, E. F. Nakamura, A. A. F. Loureiro, and A. Boukerche. Error analysis of localization systems for sensor networks. In *Proceedings of 13th ACM International Workshop on Geographic Information Systems (ACM-GIS)*, pages 71–78, 2005.

[161] A. Boukerche, S.l K. Das, A. Fabbri, and O. Yildiz. Exploiting model independence for parallel PCS network simulation. In *Proceedings of the Thirteenth Workshop on Parallel and Distributed Simulation (PADS)*, pages 166–173, 1999.

[162] A. Boukerche and C. Tropper. A distributed graph algorithm for the detection of local cycles and knots. *IEEE Transactions on Parallel and Distributed Systems*, 9(8):748–757, 1998.

[163] R. B. Batista, A. Boukerche, and A. C. M. A. de Melo. A parallel strategy for biological sequence alignment in restricted memory space. *Journal of Parallel and Distributed Computing*, 68(4):548–561, 2008.

[164] A. Boukerche, J. M. Correa, A. C. M. A. de Melo, and R. P. Jacobi. A hardware accelerator for the fast retrieval of DIALIGN biological sequence alignments in linear space. *IEEE Transactions on Computers*, 59(6):808–821, 2010.

[165] M. Alnuaimi, K. Shuaib, K. Alnuaimi, and M. Abdel-Hafez. Data gathering in delay tolerant wireless sensor networks using a ferry. *Sensors*, 15(10):25809–25830, 2015.

[166] Z. H. Ahmed. Genetic algorithm for the traveling salesman problem using sequential constructive crossover operator. *Journal of Biometrics & Bioinformatics*, 3:96–105, 2010.

[167] S. Y. Wu and J. S. Liu. Evolutionary path planning of a data mule in wireless sensor network by using shortcuts. In *Proceedings of IEEE Congress on Evolutionary Computation (CEC)*, pages 2708–2715, 2014.

[168] R. Hinterding. Gaussian mutation and self-adaption for numeric genetic algorithms. In *Proceedings of IEEE International Conference on Evolutionary Computation (ICEC)*, volume 1, pages 384–389, 1995.

[169] SZ DJI Technology Co., Ltd. Phantom 4 pro specs, 2018.

[170] S. Yu and C. S. G. Lee. Mobility and routing joint design for lifetime maximization in mobile sensor networks. In *Proceedings of 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2300–2305, 2011.

[171] A. A. A. Ari, B. O. Yenke, N. Labraoui, I. Damakoa, and A. Gueroui. A power efficient cluster-based routing algorithm for wireless sensor networks: Honeybees swarm intelligence based approach. *Journal of Network and Computer Applications*, 69:77–97, 2016.

[172] D. S. Johnson and L. A. McGeoch. *Experimental Analysis of Heuristics for the STSP*, pages 369–443. 2007.

[173] M. Ma, Y. Yang, and M. Zhao. Tour planning for mobile data-gathering mechanisms in wireless sensor networks. *IEEE Transactions on Vehicular Technology*, 62(4):1472–1483, 2013.

[174] Y. Li, R. Bartos, and J. Swan. Transport protocol with acknowledgement-assisted storage management for intermittently connected wireless sensor networks. In *Proceedings of International Conference on Distributed Computing and Networking (ICDCN)*, volume 7730, pages 57–71, 2013.

[175] P. Thulasiraman and K. A. White. Topology control of tactical wireless sensor networks using energy efficient zone routing. *Digital Communications and Networks*, 2(1):1 – 14, 2016.

[176] S. He, J. Chen, F. Jiang, D. K. Y. Yau, G. Xing, and Y.n Sun. Energy provisioning in wireless rechargeable sensor networks. *IEEE Transactions on Mobile Computing*, 12(10):1931–1942, 2013.

[177] Y. Ma, W. Liang, and W. Xu. Charging utility maximization in wireless rechargeable sensor networks by charging multiple sensors simultaneously. *IEEE/ACM Transactions on Networking*, 26(4):1591–1604, 2018.

[178] T. Wu, P. Yang, H. Dai, W. Xu, and M. Xu. Charging oriented sensor placement and flexible scheduling in rechargeable wsns. In *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, pages 73–81, 2019.

[179] W. Liang, Z. Xu, W. Xu, J. Shi, G. Mao, and S. K. Das. Approximation algorithms for charging reward maximization in rechargeable sensor networks via a mobile charger. *IEEE/ACM Transactions on Networking*, 25(5):3161–3174, 2017.

[180] X. Ye and W. Liang. Charging utility maximization in wireless rechargeable sensor networks. *Wireless Networks*, 23(7):2069–2081, 2017.

[181] C. Chekuri and M. Pal. A recursive greedy algorithm for walks in directed graphs. In *Proceedings of Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 245–253, 2005.

[182] J. Zhong, X. Hu, J. Zhang, and M. Gu. Comparison of performance between different selection strategies on simple genetic algorithms. In *Proceedings of IEEE International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC)*, volume 2, pages 1115–1121, 2005.

[183] C. Zhou, J. Zheng, and W. Li. An improved heuristic crossover operator for tsp. In *Proceedings of IEEE International Conference on Natural Computation (ICNC)*, volume 1, pages 541–545, 2008.

[184] Emile Aarts and Jan Karel Lenstra. *Local search in combinatorial optimization.* Princeton University Press, 2003.

[185] J. R. Current and D. A. Schilling. The covering salesman problem. *Transportation Science*, 23(3):208–213, 1989.

[186] G. Han, X. Yang, L. Liu, and W. Zhang. A joint energy replenishment and data collection algorithm in wireless rechargeable sensor networks. *IEEE Internet of Things Journal*, 5(4):2596–2604, 2017.

[187] A. Boukerche and S. Samarah. An efficient data extraction mechanism for mining association rules from wireless sensor networks. In *Proceedings of IEEE International Conference on Communications (ICC)*, pages 3936–3941, 2007.

[188] A. Mammeri, A. Boukerche, and G. Lu. Lane detection and tracking system based on the MSER algorithm, hough transform and kalman filter. In *Proceedings of 17th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, pages 259–266, 2014.

[189] A. Mammeri, A. Boukerche, and Z. Tang. A real-time lane marking localization, tracking and communication system. *Computer Communications*, 73:132–143, 2016.

[190] M. B. Younes and A. Boukerche. A performance evaluation of an efficient traffic congestion detection protocol (ECODE) for intelligent transportation systems. *Ad Hoc Networks*, 24:317–336, 2015.

[191] C. Chang, C. Chang, Y. Chen, and H. Chang. Obstacle-resistant deployment algorithms for wireless sensor networks. *IEEE Transactions on Vehicular Technology*, 58(6):2925–2941, 2009.

[192] T. Chin. Sensor deployment for collaborative target detection in the presence of obstacles. In *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM)*, pages 1–5, 2009.

[193] H. P. Gupta, S. V. Rao, A. K. Yadav, and T. Dutta. Geographic routing in clustered wireless sensor networks among obstacles. *IEEE Sensors Journal*, 15(5):2984–2992, 2015.

[194] R. W. N. Pazzi and A. Boukerche. PROPANE: A progressive panorama streaming protocol to support interactive 3d virtual environment exploration on graphics-constrained devices. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 11(1):5:1–5:22, 2014.

[195] R. W. L. Coutinho, A. Boukerche, L. F. M. Vieira, and A. A. F. Loureiro. Geographic and opportunistic routing for underwater sensor networks. *IEEE Transactions on Computers*, 65(2):548–561, 2015.

[196] R. W. L. Coutinho, A. Boukerche, L. F. M. Vieira, and A. A. F. Loureiro. Design guidelines for opportunistic routing in underwater networks. *IEEE Communications Magazine*, 54(2):40–48, 2016.

[197] A. Darehshoorzadeh and A. Boukerche. Underwater sensor networks: a new challenge for opportunistic routing protocols. *IEEE Communications Magazine*, 53(11):98–107, 2015.

[198] R. W. L. Coutinho, A. Boukerche, L. F. M. Vieira, and A. A. F. Loureiro. A novel void node recovery paradigm for long-term underwater sensor networks. *Ad Hoc Networks*, 34:144–156, 2015.

[199] R. W. L. Coutinho, A. Boukerche, L. F. M. Vieira, and A. A. F. Loureiro. Underwater wireless sensor networks: A new challenge for topology control-based systems. *ACM Computing Surveys*, 51(1):19:1–19:36, 2018.

[200] A. Boukerche, I. Chatzigiannakis, and S. E. Nikoletseas. A new energy efficient and fault-tolerant protocol for data propagation in smart dust networks using varying transmission range. *Computer Communications*, 29(4):477–489, 2006.